

# COMPUTATIONAL BIOLOGY - GENOMES, NETWORKS, AND EVOLUTION

*iSi!*

*Manolis Kellis et al.*  
Massachusetts Institute of Technology

Massachusetts Institute of Technology  
Computational Biology - Genomes, Networks,  
and Evolution

Manolis Kellis et al.

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (<https://LibreTexts.org>) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by [NICE CXone Expert](#) and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptions contact [info@LibreTexts.org](mailto:info@LibreTexts.org) or visit our main website at <https://LibreTexts.org>.

This text was compiled on 11/17/2025

## TABLE OF CONTENTS

### Licensing

### Materia Frontal

- [TitlePage](#)
- [InfoPage](#)
- [Tabla de Contenidos](#)

### 1: Introducción al Curso

- [1.1: Introducción y Objetivos](#)
- [1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional](#)
- [1.3: Materiales adicionales](#)
- [1.4: Curso Crash en Biología Molecular](#)
- [1.5: Introducción a algoritmos e inferencia probabilística](#)

### 2: Alineación de Secuencias y Programación Dinámica

- [2.1: Introducción](#)
- [2.2: Alineación de secuencias](#)
- [2.3: Formulaciones de problemas](#)
- [2.4: Programación dinámica](#)
- [2.5: El algoritmo de Needleman-Wunsch](#)
- [2.6: Alineación múltiple](#)
- [2.7: Herramientas y Técnicas](#)
- [2.8: Apéndice](#)
- [2.9: Bibliografía](#)

### 3: Alineación rápida de secuencias y búsqueda de bases de datos

- [3.1: ¿Qué hemos aprendido?](#)
- [3.2: Introducción](#)
- [3.3: Alineación global vs. alineación local vs. alineación semi-global](#)
- [3.4: Coincidencia exacta de cadenas en tiempo lineal](#)
- [3.5: El algoritmo BLAST \(Herramienta Básica de Búsqueda de Alineación Local\)](#)
- [3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal](#)
- [3.7: Fundamentos probabilísticos del alineamiento de secuencias](#)

### 4: Genómica Comparada I- Anotación del Genoma

- [4.1: Introducción](#)
- [4.2: Conservación de secuencias genómicas](#)
- [4.3: Restricción en exceso](#)
- [4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección](#)
- [4.5: Firmas de codificación de proteínas](#)
- [4.6: Firmas génicas de microARN \(miARN\)](#)
- [4.7: Motivos Regulatorios](#)
- [4.8: Lectura adicional](#)
- [4.9: Herramientas y Técnicas](#)

- Bibliografía

## 5: Ensamblaje del Genoma y Alineación del Genoma

- 5.1: Introducción
- 5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso
- 5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas
- 5.4: Alineación del Genoma Completo
- 5.5: Alineación regional basada en genes
- 5.6: Mecanismos de Evolución Genómica
- 5.7: Duplicación del genoma completo
- 5.8: Recursos adicionales y bibliografía
- Bibliografía

## 6: Genómica Bacteriana—Evolución Molecular a Nivel de Ecosistemas

- 6.1: Introducción
- 6.2: Estudio 1- Evolución de la vida en la tierra
- 6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros
- 6.4: Estudio 3- Proyecto de Ecología Gut Humana (HuGE)
- 6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo
- 6.6: Estudio 5- Transferencia Génica Horizontal (HGT) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos
- 6.7: Estudio 6- Identificación de factores de virulencia en Meningitis
- 6.08: Q
  - 6.8: Q/A
- 6.9: Direcciones actuales de investigación
- 6.10 Lectura adicional
- 6.12 ¿Qué hemos aprendido?
- Bibliografía

## 7: Modelos ocultos de Markov I

- 7.1: Introducción
- 7.2: Motivación
- 7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización
- 7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología
- 7.5: Ajustes algorítmicos para HMM
- 7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo?
- 7.7: Lectura adicional, ¿qué hemos aprendido?

## 8: Modelos Ocultos de Markov II-Decodificación posterior y aprendizaje

- 8.1: Revisión de la conferencia anterior
- 8.2: Decodificación posterior
- 8.3: Memoria de codificación en un HMM- Detección de islas CpG
- 8.4: Aprendizaje
- 8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín
- 8.6: Direcciones actuales de investigación, ¿qué hemos aprendido? , Bibliografía
- 8.9 ¿Qué hemos aprendido?
- Bibliografía

## 9: Identificación Génica- Estructura Génica, Semi-Markov, CRFS

- 9.1: Introducción
- 9.2: Descripción general de los contenidos del capítulo
- 9.3: Genes eucariotas: una introducción
- 9.4: Supuestos para la identificación computacional de genes
- 9.5: Cadenas Ocultas de Markov
- 9.6: Campos aleatorios condicionales
- 9.7: Otros métodos
- 9.8: Conclusión, Bibliografía
- Bibliografía

## 10: Plegamiento de ARN

- 10.1: Motivación y Propósito
- 10.2: Química del ARN
- 10.3: Origen y Funciones del ARN
- 10.4: Estructura del ARN
- 10.5: Problema de plegamiento de ARN y enfoques
- 10.6: Evolución del ARN
- 10.7: Aproximación probabilística al problema del plegamiento del ARN
- 10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía

## 11: Modificaciones de ARN

- 11.1: Introducción
- 11.2: Regulación Postranscripcional
- 11.3: ¿Qué hemos aprendido?

## 12: ARN intergénicos grandes no codificantes

- 12.1: Bibliografía
- 12.2: Introducción
- 12.3: ARN no codificantes de plantas a mamíferos
- 12.4: Tema práctico- RNaseQ
- 12.5: ARN largos no codificantes en la regulación epigenética
- 12.6: ARN intergenéticos no codificantes: ¿faltan lincs en células madre o cancerosas?
- 12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas?

## 13: ARN pequeño

- 13.1: Introducción
- 13.2: Interferencia de ARN
- 13.3: Bibliografía

## 14: Secuenciación de ARNm para análisis de expresión y descubrimiento de transcritos

- 14.1: Introducción
- 14.2: Microarrays de expresión
- 14.3: La biología de la secuenciación de ARNm
- 14.4: Mapeo de Lectura - Alineación Espaciada de
- 14.5: Reconstrucción

- [14.6: Cuantificación](#)

## 15: Regulación Génica I - Agrupación de Expresión Génica

- [15.1: Introducción](#)
- [15.2: Métodos para medir la expresión génica](#)
- [15.3: Algoritmos de Clustering](#)
- [15.4: Direcciones actuales de investigación](#)
- [15.5: Lectura adicional](#)
- [15.6: Recursos](#)
- [15.7: Qué hemos aprendido, Bibliografía](#)

## 16: Regulación Génica II - Clasificación

- [16.1: Introducción](#)
- [16.2: Clasificación—Técnicas Bayesianas](#)
- [16.3: Máquinas vectoriales de soporte de clasificación](#)
- [16.4: Clasificación Tumoral con SVMs](#)
- [16.5: Aprendizaje Semi-Supervisado](#)
- [16.6: Lectura adicional, Recursos, Bibliografía](#)

## 17: Motivos Regulatorios, Muestreo de Gibbs y EM

- [17.1: Representación de Motivos y Contenido de Información](#)
- [17.2: Introducción a los motivos reguladores y la regulación génica](#)
- [17.3: Maximización de expectativas](#)
- [17.4: Muestreo de Gibbs- Muestra de distribución conjunta \( \$M, Z\_{ij}\$ \)](#)
- [17.5: Descubrimiento del motivo de novo](#)
- [17.6: Posiblemente cosas en desuso por debajo-](#)
- [17.7: Comparando diferentes métodos](#)
- [17.8: OOPS, ZOOPS, MTC](#)
- [17.9: Ampliación del Enfoque EM](#)

## 18: Genómica Regulatoria

- [18.1: Introducción a la Genómica Regulatoria](#)
- [18.2: Descubrimiento de Motivos De Novo](#)
- [18.3: Predecir objetivos regulares](#)
- [18.4: Genes y dianas de microARN](#)

## 19: Epigenómica

- [19: Epigenómica/Estados de cromatina](#)
  - [19.1: Introducción](#)
  - [19.2: Información Epigenética en Nucleosomas](#)
  - [19.3: Ensayos Epigenómicos](#)
  - [19.4: Procesamiento primario de datos de ChIP](#)
  - [19.5: Anotar el genoma usando firmas de cromatina](#)
  - [19.6: Direcciones actuales de investigación](#)
  - [19.7: Lectura adicional, herramientas y técnicas](#)
  - [19.8: ¿Qué hemos aprendido? , Bibliografía](#)

## 20: Redes I- Inferencia, Estructura, Métodos Espectrales

- 20.1: Introducción
- 20.2: Medidas de Centralidad de Red
- 20.3: Revisión de álgebra lineal
- 20.4: Análisis de componentes principales dispersos
- 20.5: Comunidades y Módulos de Red
- 20.6: Núcleo de Difusión en Red
- 20.7: Redes neuronales
- 20.8: Temas abiertos y desafíos
- 20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía
- 20.10: ¿Qué hemos aprendido?
- Bibliografía

## 21: Redes Regulatorias- Inferencia, Análisis, Aplicación

- 21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación
- 21.2: Inferencia de estructura
- 21.3: Visión general de la tarea de aprendizaje PGM
- 21.4: Aplicación de Redes
- 21.5: Propiedades Estructurales de Redes
- 21.6: Clustering de redes, Bibliografía
- Bibliografía

## 22: Interacciones de cromatina

- 22.1: Introducción
- 22.2: Terminología relevante
- 22.3: Métodos moleculares para estudiar la organización del genoma nuclear
- 22.4: Mapeo de interacciones genoma-lámina nuclear (LADs)
- 22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear
- 22.6: Arquitectura de la Organización del Genoma
- 22.7: Comprensión mecanicista de la arquitectura del genoma
- 22.8: Direcciones actuales de investigación

## 23: Introducción al Modelado Metabólico en Estado Estable

- 23.1: Introducción
- 23.2: Construcción de modelos
- 23.3: Análisis de Flujo Metabólico
- 23.4: Aplicaciones
- 23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía
- 23.6: Herramientas y Techniques
- Bibliografía

## 24: El Proyecto Encode- Experimentación Sistemática y Genómica Integrativa

- 24.1: Introducción
- 24.2: Técnicas Experimentales
- 24.3: Técnicas Computacionales
- 24.4: Direcciones actuales de investigación

- 24.5: Lectura adicional, Herramientas y técnicas, Bibliografía
- 24.6: Herramientas y Técnicas
- Bibliografía
- Sección 7: ¿Qué hemos aprendido?

## 25: Biología Sintética

- 25.1: Introducción a la Biología Sintética
- 25.2: Direcciones actuales de investigación
- 25.3: Herramientas y Técnicas
- 25.4: ¿Qué hemos aprendido? , Bibliografía
- Bibliografía

## 26: Evolución Molecular y Filogenética

- 26.1: Introducción
- 26.2: Fundamentos de la Filogenia
- 26.3: Métodos basados en la distancia
- 26.4: Métodos basados en caracteres
- 26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido
- 26.6: Hacia el proyecto final
- 26.7: ¿Qué hemos aprendido?
- Bibliografía

## 27: Filogenómica II

- 27.1: Introducción
- 27.2: SPIDR
- 27.3: Gráficas de Recombinación Ancestral
- 27.4: Conclusión
- 27.05: Inferir ortológicos
  - 27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética
- 27.6: Reconstrucción
- 27.7: Modelización de Frecuencias de Poblaciones y Alelos
- 27.10 ¿Qué hemos aprendido?
- 27.9 Lectura adicional
- Bibliografía

## 28: Historia de la población

- 28.1: Introducción
- 28.2: Encuesta Rápida de Variación Genética Humana
- 28.3: Flujo genético africano y europeo
- 28.4: Flujo de genes en el subcontinente indio
- 28.5: Flujo de genes entre poblaciones humanas arcaicas
- 28.6: Herramientas y Técnicas
- 28.7: Direcciones de investigación, lecturas adicionales, bibliografía
- 28.8: Ascendencia Europea y Migraciones

## 29: Variación genética poblacional

- 29.1: Introducción
- 29.2: Conceptos básicos de selección de población
- 29.3: Vinculación genética
- 29.4: Selección natural
- 29.5: Evolución Humana
- 29.6: Investigación actual
- 29.7: Lectura adicional

## 30: Genética médica: el pasado hasta el presente

- 30.1: Bibliografía
- 30.2: Introducción
- 30.3: Objetivos de investigar las bases genéticas de la enfermedad
- 30.4: Rasgos mendelianos
- 30.5: Rasgos Complejos
- 30.6: Estudios de Asociación en todo el genoma
- 30.7: Direcciones actuales de investigación
- 30.8: Herramientas y Técnicas
- 30.9: ¿Qué hemos aprendido?

## 31: Variación 2- Mapeo cuantitativo de rasgos, eQTLs, Variación de Rasgo Molecular

- 31.1: Introducción
- 31.2: Conceptos básicos de eQTL
- 31.3: Estructura de un estudio eQTL
- 31.4: Direcciones actuales de investigación
- 31.5: ¿Qué hemos aprendido?
- 31.6: Lectura adicional
- 31.7: Herramientas y Recursos
- 31.8: Bibliografía

## 32: Genomas Personales, Genomas Sintéticos, Computación en C vs Si

- 32.1: Introducción
- 32.2: Genomas de Lectura y Escritura
- 32.3: Genomas personales
- 32.4: Lectura adicional
- 32.5: Bibliografía

## 33: Genómica personal

- 33.1: Introducción
- 33.2: Epidemiología- Una visión general
- 33.3: Epidemiología Genética
- 33.4: Epidemiología Molecular
- 33.5: Modelado y Pruebas de Causalidad
- 33.6: ¿Qué hemos aprendido?

## 34: Genómica del Cáncer

- [Sección 1: Introducción](#)
- [Sección 2: Caracterización](#)
- [Sección 3: Interpretación](#)
- [Sección 5: Lectura adicional](#)
- [Sección 6: ¿Qué hemos aprendido?](#)

## 35: Edición del genoma

- [1: Introducción](#)
- [2: Direcciones actuales de investigación](#)
- [3: ¿Qué hemos aprendido?](#)

[Index](#)

[Glossary](#)

[Detailed Licensing](#)

[Volver Materia](#)

- [Índice](#)
- [Índice](#)
- [Glosario](#)

## Licensing

A detailed breakdown of this resource's licensing can be found in [Back Matter/Detailed Licensing](#).

## CHAPTER OVERVIEW

### Materia Frontal

[TitlePage](#)

[InfoPage](#)

[Tabla de Contenidos](#)

---

This page titled [Materia Frontal](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

Texto por Defecto

---

This page titled [TitlePage](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [TitlePage](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (<https://LibreTexts.org>) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by [NICE CXone Expert](#) and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptations contact [info@LibreTexts.org](mailto:info@LibreTexts.org) or visit our main website at <https://LibreTexts.org>.

This text was compiled on 11/17/2025

## TABLE OF CONTENTS

### Licensing

### Materia Frontal

- [TitlePage](#)
- [InfoPage](#)
- [Tabla de Contenidos](#)

### 1: Introducción al Curso

- [1.1: Introducción y Objetivos](#)
- [1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional](#)
- [1.3: Materiales adicionales](#)
- [1.4: Curso Crash en Biología Molecular](#)
- [1.5: Introducción a algoritmos e inferencia probabilística](#)

### 2: Alineación de Secuencias y Programación Dinámica

- [2.1: Introducción](#)
- [2.2: Alineación de secuencias](#)
- [2.3: Formulaciones de problemas](#)
- [2.4: Programación dinámica](#)
- [2.5: El algoritmo de Needleman-Wunsch](#)
- [2.6: Alineación múltiple](#)
- [2.7: Herramientas y Técnicas](#)
- [2.8: Apéndice](#)
- [2.9: Bibliografía](#)

### 3: Alineación rápida de secuencias y búsqueda de bases de datos

- [3.1: ¿Qué hemos aprendido?](#)
- [3.2: Introducción](#)
- [3.3: Alineación global vs. alineación local vs. alineación semi-global](#)
- [3.4: Coincidencia exacta de cadenas en tiempo lineal](#)
- [3.5: El algoritmo BLAST \(Herramienta Básica de Búsqueda de Alineación Local\)](#)
- [3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal](#)
- [3.7: Fundamentos probabilísticos del alineamiento de secuencias](#)

### 4: Genómica Comparada I- Anotación del Genoma

- [4.1: Introducción](#)
- [4.2: Conservación de secuencias genómicas](#)
- [4.3: Restricción en exceso](#)
- [4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección](#)
- [4.5: Firmas de codificación de proteínas](#)
- [4.6: Firmas génicas de microARN \(miARN\)](#)
- [4.7: Motivos Regulatorios](#)
- [4.8: Lectura adicional](#)
- [4.9: Herramientas y Técnicas](#)

- Bibliografía

## 5: Ensamblaje del Genoma y Alineación del Genoma

- 5.1: Introducción
- 5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso
- 5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas
- 5.4: Alineación del Genoma Completo
- 5.5: Alineación regional basada en genes
- 5.6: Mecanismos de Evolución Genómica
- 5.7: Duplicación del genoma completo
- 5.8: Recursos adicionales y bibliografía
- Bibliografía

## 6: Genómica Bacteriana—Evolución Molecular a Nivel de Ecosistemas

- 6.1: Introducción
- 6.2: Estudio 1- Evolución de la vida en la tierra
- 6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros
- 6.4: Estudio 3- Proyecto de Ecología Gut Humana (HuGE)
- 6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo
- 6.6: Estudio 5- Transferencia Génica Horizontal (HGT) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos
- 6.7: Estudio 6- Identificación de factores de virulencia en Meningitis
- 6.08: Q
  - 6.8: Q/A
- 6.9: Direcciones actuales de investigación
- 6.10 Lectura adicional
- 6.12 ¿Qué hemos aprendido?
- Bibliografía

## 7: Modelos ocultos de Markov I

- 7.1: Introducción
- 7.2: Motivación
- 7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización
- 7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología
- 7.5: Ajustes algorítmicos para HMM
- 7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo?
- 7.7: Lectura adicional, ¿qué hemos aprendido?

## 8: Modelos Ocultos de Markov II-Decodificación posterior y aprendizaje

- 8.1: Revisión de la conferencia anterior
- 8.2: Decodificación posterior
- 8.3: Memoria de codificación en un HMM- Detección de islas CpG
- 8.4: Aprendizaje
- 8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín
- 8.6: Direcciones actuales de investigación, ¿qué hemos aprendido? , Bibliografía
- 8.9 ¿Qué hemos aprendido?
- Bibliografía

## 9: Identificación Génica- Estructura Génica, Semi-Markov, CRFS

- 9.1: Introducción
- 9.2: Descripción general de los contenidos del capítulo
- 9.3: Genes eucariotas: una introducción
- 9.4: Supuestos para la identificación computacional de genes
- 9.5: Cadenas Ocultas de Markov
- 9.6: Campos aleatorios condicionales
- 9.7: Otros métodos
- 9.8: Conclusión, Bibliografía
- Bibliografía

## 10: Plegamiento de ARN

- 10.1: Motivación y Propósito
- 10.2: Química del ARN
- 10.3: Origen y Funciones del ARN
- 10.4: Estructura del ARN
- 10.5: Problema de plegamiento de ARN y enfoques
- 10.6: Evolución del ARN
- 10.7: Aproximación probabilística al problema del plegamiento del ARN
- 10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía

## 11: Modificaciones de ARN

- 11.1: Introducción
- 11.2: Regulación Postranscripcional
- 11.3: ¿Qué hemos aprendido?

## 12: ARN intergénicos grandes no codificantes

- 12.1: Bibliografía
- 12.2: Introducción
- 12.3: ARN no codificantes de plantas a mamíferos
- 12.4: Tema práctico- RNaseQ
- 12.5: ARN largos no codificantes en la regulación epigenética
- 12.6: ARN intergenéticos no codificantes: ¿faltan lincs en células madre o cancerosas?
- 12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas?

## 13: ARN pequeño

- 13.1: Introducción
- 13.2: Interferencia de ARN
- 13.3: Bibliografía

## 14: Secuenciación de ARNm para análisis de expresión y descubrimiento de transcritos

- 14.1: Introducción
- 14.2: Microarrays de expresión
- 14.3: La biología de la secuenciación de ARNm
- 14.4: Mapeo de Lectura - Alineación Espaciada de
- 14.5: Reconstrucción

- [14.6: Cuantificación](#)

## 15: Regulación Génica I - Agrupación de Expresión Génica

- [15.1: Introducción](#)
- [15.2: Métodos para medir la expresión génica](#)
- [15.3: Algoritmos de Clustering](#)
- [15.4: Direcciones actuales de investigación](#)
- [15.5: Lectura adicional](#)
- [15.6: Recursos](#)
- [15.7: Qué hemos aprendido, Bibliografía](#)

## 16: Regulación Génica II - Clasificación

- [16.1: Introducción](#)
- [16.2: Clasificación—Técnicas Bayesianas](#)
- [16.3: Máquinas vectoriales de soporte de clasificación](#)
- [16.4: Clasificación Tumoral con SVMs](#)
- [16.5: Aprendizaje Semi-Supervisado](#)
- [16.6: Lectura adicional, Recursos, Bibliografía](#)

## 17: Motivos Regulatorios, Muestreo de Gibbs y EM

- [17.1: Representación de Motivos y Contenido de Información](#)
- [17.2: Introducción a los motivos reguladores y la regulación génica](#)
- [17.3: Maximización de expectativas](#)
- [17.4: Muestreo de Gibbs- Muestra de distribución conjunta \( \$M, Z\_{ij}\$ \)](#)
- [17.5: Descubrimiento del motivo de novo](#)
- [17.6: Posiblemente cosas en desuso por debajo-](#)
- [17.7: Comparando diferentes métodos](#)
- [17.8: OOPS, ZOOPS, MTC](#)
- [17.9: Ampliación del Enfoque EM](#)

## 18: Genómica Regulatoria

- [18.1: Introducción a la Genómica Regulatoria](#)
- [18.2: Descubrimiento de Motivos De Novo](#)
- [18.3: Predecir objetivos regulares](#)
- [18.4: Genes y dianas de microARN](#)

## 19: Epigenómica

- [19: Epigenómica/Estados de cromatina](#)
  - [19.1: Introducción](#)
  - [19.2: Información Epigenética en Nucleosomas](#)
  - [19.3: Ensayos Epigenómicos](#)
  - [19.4: Procesamiento primario de datos de ChIP](#)
  - [19.5: Anotar el genoma usando firmas de cromatina](#)
  - [19.6: Direcciones actuales de investigación](#)
  - [19.7: Lectura adicional, herramientas y técnicas](#)
  - [19.8: ¿Qué hemos aprendido? , Bibliografía](#)

## 20: Redes I- Inferencia, Estructura, Métodos Espectrales

- 20.1: Introducción
- 20.2: Medidas de Centralidad de Red
- 20.3: Revisión de álgebra lineal
- 20.4: Análisis de componentes principales dispersos
- 20.5: Comunidades y Módulos de Red
- 20.6: Núcleo de Difusión en Red
- 20.7: Redes neuronales
- 20.8: Temas abiertos y desafíos
- 20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía
- 20.10: ¿Qué hemos aprendido?
- Bibliografía

## 21: Redes Regulatorias- Inferencia, Análisis, Aplicación

- 21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación
- 21.2: Inferencia de estructura
- 21.3: Visión general de la tarea de aprendizaje PGM
- 21.4: Aplicación de Redes
- 21.5: Propiedades Estructurales de Redes
- 21.6: Clustering de redes, Bibliografía
- Bibliografía

## 22: Interacciones de cromatina

- 22.1: Introducción
- 22.2: Terminología relevante
- 22.3: Métodos moleculares para estudiar la organización del genoma nuclear
- 22.4: Mapeo de interacciones genoma-lámina nuclear (LADs)
- 22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear
- 22.6: Arquitectura de la Organización del Genoma
- 22.7: Comprensión mecanicista de la arquitectura del genoma
- 22.8: Direcciones actuales de investigación

## 23: Introducción al Modelado Metabólico en Estado Estable

- 23.1: Introducción
- 23.2: Construcción de modelos
- 23.3: Análisis de Flujo Metabólico
- 23.4: Aplicaciones
- 23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía
- 23.6: Herramientas y Techniques
- Bibliografía

## 24: El Proyecto Encode- Experimentación Sistemática y Genómica Integrativa

- 24.1: Introducción
- 24.2: Técnicas Experimentales
- 24.3: Técnicas Computacionales
- 24.4: Direcciones actuales de investigación

- 24.5: Lectura adicional, Herramientas y técnicas, Bibliografía
- 24.6: Herramientas y Técnicas
- Bibliografía
- Sección 7: ¿Qué hemos aprendido?

## 25: Biología Sintética

- 25.1: Introducción a la Biología Sintética
- 25.2: Direcciones actuales de investigación
- 25.3: Herramientas y Técnicas
- 25.4: ¿Qué hemos aprendido? , Bibliografía
- Bibliografía

## 26: Evolución Molecular y Filogenética

- 26.1: Introducción
- 26.2: Fundamentos de la Filogenia
- 26.3: Métodos basados en la distancia
- 26.4: Métodos basados en caracteres
- 26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido
- 26.6: Hacia el proyecto final
- 26.7: ¿Qué hemos aprendido?
- Bibliografía

## 27: Filogenómica II

- 27.1: Introducción
- 27.2: SPIDR
- 27.3: Gráficas de Recombinación Ancestral
- 27.4: Conclusión
- 27.05: Inferir ortológicos
  - 27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética
- 27.6: Reconstrucción
- 27.7: Modelización de Frecuencias de Poblaciones y Alelos
- 27.10 ¿Qué hemos aprendido?
- 27.9 Lectura adicional
- Bibliografía

## 28: Historia de la población

- 28.1: Introducción
- 28.2: Encuesta Rápida de Variación Genética Humana
- 28.3: Flujo genético africano y europeo
- 28.4: Flujo de genes en el subcontinente indio
- 28.5: Flujo de genes entre poblaciones humanas arcaicas
- 28.6: Herramientas y Técnicas
- 28.7: Direcciones de investigación, lecturas adicionales, bibliografía
- 28.8: Ascendencia Europea y Migraciones

## 29: Variación genética poblacional

- 29.1: Introducción
- 29.2: Conceptos básicos de selección de población
- 29.3: Vinculación genética
- 29.4: Selección natural
- 29.5: Evolución Humana
- 29.6: Investigación actual
- 29.7: Lectura adicional

## 30: Genética médica: el pasado hasta el presente

- 30.1: Bibliografía
- 30.2: Introducción
- 30.3: Objetivos de investigar las bases genéticas de la enfermedad
- 30.4: Rasgos mendelianos
- 30.5: Rasgos Complejos
- 30.6: Estudios de Asociación en todo el genoma
- 30.7: Direcciones actuales de investigación
- 30.8: Herramientas y Técnicas
- 30.9: ¿Qué hemos aprendido?

## 31: Variación 2- Mapeo cuantitativo de rasgos, eQTLs, Variación de Rasgo Molecular

- 31.1: Introducción
- 31.2: Conceptos básicos de eQTL
- 31.3: Estructura de un estudio eQTL
- 31.4: Direcciones actuales de investigación
- 31.5: ¿Qué hemos aprendido?
- 31.6: Lectura adicional
- 31.7: Herramientas y Recursos
- 31.8: Bibliografía

## 32: Genomas Personales, Genomas Sintéticos, Computación en C vs Si

- 32.1: Introducción
- 32.2: Genomas de Lectura y Escritura
- 32.3: Genomas personales
- 32.4: Lectura adicional
- 32.5: Bibliografía

## 33: Genómica personal

- 33.1: Introducción
- 33.2: Epidemiología- Una visión general
- 33.3: Epidemiología Genética
- 33.4: Epidemiología Molecular
- 33.5: Modelado y Pruebas de Causalidad
- 33.6: ¿Qué hemos aprendido?

## 34: Genómica del Cáncer

- Sección 1: Introducción
- Sección 2: Caracterización
- Sección 3: Interpretación
- Sección 5: Lectura adicional
- Sección 6: ¿Qué hemos aprendido?

## 35: Edición del genoma

- 1: Introducción
- 2: Direcciones actuales de investigación
- 3: ¿Qué hemos aprendido?

[Index](#)

[Glossary](#)

[Detailed Licensing](#)

## Volver Materia

- Índice
- Índice
- Glosario

---

Tabla de Contenidos is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## CHAPTER OVERVIEW

### 1: Introducción al Curso

- 1.1: Introducción y Objetivos
- 1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional
- 1.3: Materiales adicionales
- 1.4: Curso Crash en Biología Molecular
- 1.5: Introducción a algoritmos e inferencia probabilística

---

This page titled [1: Introducción al Curso](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 1.1: Introducción y Objetivos

### curso de biología computacional

Estas notas de clase tienen como objetivo ser impartidas como un curso de término sobre biología computacional, cada conferencia de 1.5 horas cubriendo un capítulo, junto con tareas quincenales y sesiones de tutoría para ayudar a los estudiantes a lograr sus propios proyectos de investigación independientes. Las notas surgieron del curso del MIT 6.047/6.878, y reflejan muy de cerca la estructura de las conferencias correspondientes.

### Dualidad de Metas: Fundamentos y Fronteras

Hay dos metas para este curso. El primer objetivo es introducirte en las bases del campo de la biología computacional. A saber, introducir los problemas biológicos fundamentales del campo, y aprender las técnicas algorítmicas y de aprendizaje automático necesarias para abordarlos. Esto va más allá de aprender a usar los programas y herramientas en línea que son populares en cualquier año dado. En cambio, el objetivo es que entiendas los principios subyacentes de las técnicas más exitosas que están actualmente en uso, y brindarte la capacidad de diseñar e implementar la próxima generación de herramientas. Esa es la razón por la que una clase de algoritmos introductorios se establece como pre-req; la mejor manera de obtener una comprensión más profunda de los algoritmos presentados es implementarlos usted mismo.

El segundo objetivo del curso es abordar las fronteras de investigación de la biología computacional, y de eso se tratan realmente todos los temas avanzados y tareas prácticas. De hecho, nos gustaría darte una idea de cómo funciona la investigación, exponerte a las direcciones de investigación actuales, guiarte para encontrar los problemas más interesantes para ti y ayudarte a convertirte en un practicante activo en el campo. Esto se logra a través de conferencias invitadas, conjuntos de problemas, laboratorios y, lo más importante, un proyecto de investigación independiente a largo plazo, donde realiza su investigación independiente.

Los módulos del curso siguen ese patrón, cada uno consistente en conferencias que cubren los fundamentos y las fronteras de cada tema. Las conferencias fundacionales introducen los problemas clásicos en el campo. Estos problemas se entienden muy bien y ya se han encontrado soluciones elegantes; algunos incluso se han enseñado desde hace más de una década. La parte de fronteras del módulo abarca temas avanzados, generalmente abordando cuestiones centrales que aún permanecen abiertas en el campo. Estos capítulos suelen incluir conferencias invitadas de algunos de los pioneros en cada área que hablan tanto sobre el estado general del campo como de la investigación de su propio laboratorio.

Las asignaciones para el curso siguen el mismo patrón de fundación/fronteras. La mitad de las tareas van a consistir en elaborar los métodos con lápiz sobre papel y profundizar en las nociones algorítmicas y de aprendizaje automático de los problemas. La otra mitad en realidad van a ser preguntas prácticas consistentes en asignaciones de programación, donde se proporcionan conjuntos de datos reales. Analizarás estos datos usando las técnicas que has aprendido e interpretarás tus resultados, dándote una experiencia real. Las tareas se construyen hasta el proyecto final, donde propondrás y llevarás a cabo un proyecto de investigación original, y presentarás tus hallazgos en formato conferencia. En general, las asignaciones están diseñadas para darle la oportunidad de aplicar métodos de biología computacional a problemas reales en biología.

### Dualidad de disciplinas: Computación y Biología

Además de apuntar a cubrir tanto fundaciones como fronteras, la otra dualidad importante de este curso es entre cómputos y biología.

Desde la perspectiva biológica del curso, nuestro objetivo es enseñar temas que son fundamentales para nuestra comprensión de la biología, la medicina y la salud humana. Por lo tanto, rehuimos cualquier problema computacionalmente interesante que sea de inspiración biológica, pero que no sea relevante para la biología. No solo vamos a ver algo en biología, a inspirarnos, y luego ir a la informática y hacer muchas cosas que a la biología nunca le importarán. En cambio, nuestro objetivo es trabajar en problemas que puedan hacer un cambio significativo en el campo de la biología. Nos gustaría que publicaran artículos que realmente importan a la comunidad biológica y que tengan un impacto biológico real. Por lo tanto, este objetivo ha guiado la selección de temas para el curso, y cada capítulo se centra en un problema biológico fundamental.

Desde la perspectiva computacional del curso, siendo después de todo una clase de informática, nos enfocamos en explorar técnicas y principios generales que sin duda son importantes en la biología computacional, pero que no obstante pueden aplicarse en

cualquier otro campo que requiera análisis e interpretación de datos. De ahí que si lo que quieras es adentrarte en cosmología, meteorología, geología, o algo así, esta clase ofrece técnicas computacionales que probablemente serán útiles cuando se trata de conjuntos de datos del mundo real relacionados con esos campos.

## ¿Por qué Biología Computacional?

lecture1\_transcript.html #Motivations

Son muchas las razones por las que la Biología Computacional ha surgido como una disciplina importante en los últimos años, y quizás algunas de estas te llevan a recoger este libro o registrarte en esta clase. A pesar de que tenemos nuestra propia opinión sobre cuáles son estas razones, año con año hemos pedido a los estudiantes su propia visión sobre lo que ha permitido que el campo de la Biología Computacional se expanda tan rápidamente en los últimos años. Sus respuestas caen en varios temas amplios, que resumimos aquí.

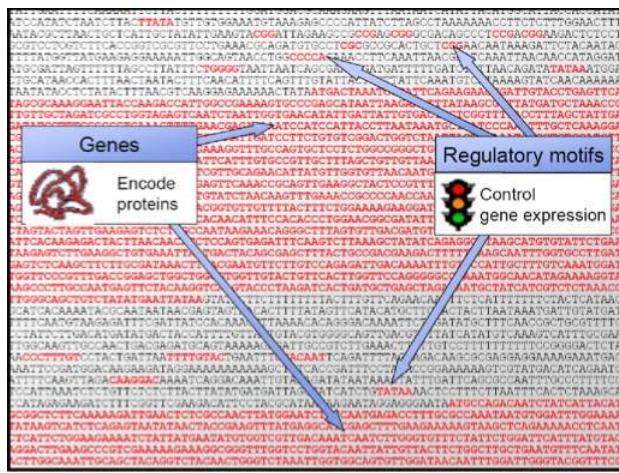
1. Quizás la razón más fundamental por la que los enfoques computacionales son tan adecuados para el estudio de los datos biológicos es que en su núcleo, los sistemas biológicos son fundamentalmente de naturaleza digital. Para ser contundentes, los humanos no son los primeros en construir una computadora digital; nuestros antepasados son la primera computadora digital, ya que las primeras formas de vida basadas en ADN ya estaban almacenando, copiando y procesando información digital codificada en las letras A, C, G y T. La mayor ventaja evolutiva de un medio digital para almacenar información genética es que puede persistir a lo largo de miles de generaciones, mientras que las señales analógicas se diluirían de generación en generación a partir de la difusión química básica.
2. Además del ADN, muchos otros aspectos de la biología son digitales, como los interruptores biológicos, que aseguran que solo se logren dos estados discretos posibles mediante bucles de retroalimentación y procesos metaestables, aunque estos sean implementados por niveles de moléculas. Los extensos circuitos de retroalimentación y otros circuitos regulatorios diversos implementan decisiones discretas a través de componentes que de otro modo serían inestables, nuevamente con principios de diseño similares a la práctica de ingeniería, haciendo que nuestra búsqueda por comprender los sistemas biológicos desde una perspectiva de ingeniería sea
3. Las ciencias que se benefician mucho del procesamiento de datos, como la Biología Computacional, siguen un ciclo virtuoso que involucra los datos disponibles para su procesamiento. Cuanto más se pueda hacer procesando y analizando los datos disponibles, más financiamiento se destinará al desarrollo de tecnologías para obtener, procesar y analizar aún más datos. Las nuevas tecnologías, como la secuenciación y las técnicas experimentales de alto rendimiento, como los ensayos de micromatrices, dos híbridos de levadura y Chip-chip, están creando cantidades enormes y crecientes de datos que pueden analizarse y procesarse usando técnicas computacionales. Los proyectos genómicos de \$1000 y \$100 son evidencia de este ciclo. Hace más de diez años, cuando comenzaron estos proyectos, hubiera sido absurdo incluso imaginar procesar cantidades tan masivas de datos. Sin embargo, a medida que se idearon más ventajas potenciales a partir del procesamiento de estos datos, se dedicó más financiamiento al desarrollo de tecnologías que harían factibles estos proyectos.
4. La capacidad de procesar datos ha mejorado mucho en los últimos años, debido a: 1) el enorme poder computacional disponible en la actualidad (debido a la ley de Moore, entre otras cosas), y 2) los avances en las técnicas algorítmicas en cuestión.
5. Los enfoques de optimización se pueden utilizar para resolver, a través de técnicas computacionales, que de otra manera son problemas in- tratables.
6. Las consideraciones de tiempo de ejecución y memoria son críticas cuando se trata de enormes conjuntos de datos. Un algoritmo que funcione bien en un genoma pequeño (por ejemplo, una bacteria) podría ser demasiado ineficiente en el tiempo o en el espacio para ser aplicado a 1000 genomas de mamíferos. Además, las preguntas combinatorias aumentan drásticamente la complejidad algorítmica.
7. Los conjuntos de datos biológicos pueden ser ruidosos y filtrar la señal del ruido es un problema computacional.
8. Los enfoques de aprendizaje automático son útiles para hacer inferencias, clasificar características biológicas e identificar señales robustas.
9. A medida que se profundiza nuestra comprensión de los sistemas biológicos, hemos comenzado a darnos cuenta de que tales sistemas no pueden analizarse aisladamente. Estos sistemas han demostrado estar entrelazados de formas antes inauditas, y hemos comenzado a cambiar nuestros análisis a técnicas que los consideran todos como un todo.

10. Es posible utilizar enfoques computacionales para encontrar correlaciones de manera imparcial, y llegar a conclusiones que transformen el conocimiento biológico y faciliten el aprendizaje activo. Este enfoque se llama descubrimiento basado en datos.
11. Los estudios computacionales pueden predecir hipótesis, mecanismos y teorías para explicar observaciones experimentales. Estas hipótesis falsificables pueden entonces ser probadas experimentalmente.
12. Los enfoques computacionales pueden ser utilizados no solo para analizar datos existentes sino también para motivar la recolección de datos y sugerir experimentos útiles. Además, el filtrado computacional puede estrechar el espacio de búsqueda experimental para permitir diseños experimentales más enfocados y eficientes.
13. La biología tiene reglas: La evolución está impulsada por dos reglas simples: 1) mutación aleatoria y 2) selección brutal. Los sistemas biológicos están limitados a estas reglas, y al analizar datos, buscamos encontrar e interpretar el comportamiento emergente que estas reglas generan.
14. Los conjuntos de datos se pueden combinar utilizando enfoques computacionales, de modo que la información recopilada a través de múltiples experimentos y el uso de diversos enfoques experimentales se pueda llevar a cabo en cuestiones de interés.
15. Las visualizaciones efectivas de los datos biológicos pueden facilitar el descubrimiento.
16. Los enfoques computacionales se pueden usar para simular y modelar datos biológicos.
17. Los enfoques computacionales pueden ser más éticos. Por ejemplo, algunos experimentos biológicos pueden ser poco éticos para realizar en sujetos vivos pero podrían ser simulados por una computadora.
18. A gran escala, los enfoques de ingeniería de sistemas se ven facilitados por la técnica computacional para obtener visiones globales sobre el organismo que son demasiado complejas para analizarlas de otra manera.

## Encontrar elementos funcionales: una pregunta de biología computacional

lecture1\_transcript.html #Codons

Varios problemas de biología computacional se refieren a encontrar señales biológicas en datos de ADN (por ejemplo, regiones codificantes, promotores, potenciadores, reguladores,...).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 1.1: En este problema de biología computacional, se nos proporciona una secuencia de bases, y se desea localizar genes y motivos reguladores.

Luego discutimos una pregunta específica que la biología computacional puede ser utilizada para abordar: ¿cómo se pueden encontrar elementos funcionales en una secuencia genómica? La Figura 1.1 muestra parte de la secuencia del genoma de la levadura. Dada esta secuencia, podemos preguntar:

P: ¿Cuáles son los genes que codifican las proteínas?

R: Durante la traducción, el codón de inicio marca el primer aminoácido en una proteína, y el codón de parada indica el final de la proteína. Sin embargo, como se indica en el portaobjetos “Extracción de señal del ruido”, solo algunas de estas secuencias ATG en el ADN marcan realmente el inicio de un gen que se expresará como proteína. Los otros son “ruido”; por ejemplo, pueden haber formado parte de intrones (secuencias no codificantes que se empalman después de la transcripción).

P: ¿Cómo podemos encontrar características (genes, motivos reguladores y otros elementos funcionales) en la secuencia genómica?

R: Estas preguntas podrían abordarse ya sea experimentalmente o computacionalmente. Un enfoque experimental del problema sería crear un knockout, y ver si se ve afectada la aptitud del organismo. También podríamos abordar la cuestión computacionalmente viendo si la secuencia se conserva a través de los genomas de múltiples especies. Si la secuencia se conserva significativamente a lo largo del tiempo evolutivo, es probable que realice una función importante.

Hay advertencias en ambos enfoques. Quitar el elemento puede no revelar su función, incluso si no hay diferencia aparente con respecto al original, esto podría deberse simplemente a que no se han probado las condiciones adecuadas. Además, el simple hecho de que un elemento no se conserve no significa que no sea funcional. (También, tenga en cuenta que “elemento funcional” es un término ambiguo. Ciertamente, hay muchos tipos de elementos funcionales en el genoma que no son codificantes de proteínas. Curiosamente, 90-95% del genoma humano se transcribe (se usa como molde para hacer ARN). No se sabe cuál es la función de la mayoría de estas regiones transcritas, o de hecho si son funcionales).

---

This page titled [1.1: Introducción y Objetivos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1: Introduction and Goals** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional

### Objetivos finales del proyecto

Un componente importante de ser biólogo computacional es la capacidad de realizar investigaciones independientes en el área. Las habilidades para un investigador exitoso difieren de una persona a otra, pero en el proceso de impartir este curso, hemos identificado varios aspectos que todos son necesarios, y hemos establecido actividades para un proyecto a largo plazo, que permitan a los estudiantes llevar a cabo su investigación independiente.

El proyecto refleja el proceso científico del mundo real: idear una idea → enmarcarla → proponerla → revisarla → llevarla a cabo → presenta tus resultados. Se espera que los estudiantes piensen críticamente sobre su propio proyecto, y también evalúen propuestas de investigación entre pares y, por último, respondan a los comentarios de sus compañeros.

Se espera que los estudiantes utilicen datos reales y presenten sus resultados en formato de conferencia. El objetivo final es la investigación publicable. Se anima a los estudiantes a platicar con el personal del curso mientras formulan una idea final del proyecto, mirar de cabeza a través de los diversos capítulos y módulos, y hacerse una idea de qué áreas le interesarán más.

### Hitos finales del proyecto

En lugar de esperar hasta el final del trimestre para comenzar la lluvia de ideas o proporcionar retroalimentación, comenzamos las actividades del proyecto con el primer conjunto de problemas, para identificar problemas de interés y tipos de proyectos, encontrar socios, hablar con estudiantes actuales y posdoctorados en biología computacional que puedan servir como mentores, y diseñar un plan de investigación al estilo de una propuesta de los NIH para identificar los posibles escollos tempranamente y abordarlos o resolverlos antes de que se conviertan en un cuello de botella.

Al establecer varios hitos de progreso incremental a lo largo del trimestre, junto con la tutoría y retroalimentación a lo largo del semestre, hemos logrado avances consistentes en años anteriores, lo que puede ser útil para los estudiantes que asuman un nuevo proyecto en cualquier etapa de su carrera. Los proyectos de investigación de este curso en el pasado han sido utilizados como punto de partida para un artículo publicado, han dado lugar a tesis de maestría y doctorado, y han obtenido premios tanto académicamente como en conferencias.

El cronograma para el proyecto final es el siguiente:

1. Configuración: una breve descripción de su experiencia e interés. Vencido 9/29
2. Lluvia de ideas: una lista de ideas y socios iniciales del proyecto. Vencido 10/6
3. Propuesta: presentar una propuesta de proyecto en forma de propuesta de NIH. Vencido 10/20
4. Presentación de la propuesta: presentar diapositivas a clase y mentores sobre la propuesta. Vencido 10/23 5. Revisión y crítica de 3 propuestas de pares. Vencido 10/30
6. Informe de avance de mitad de período: escribir esquema del informe final. Vencido 11/19
7. Informe Final del Proyecto: escribir informe en formato de ponencia de conferencia. Vencimiento 12/6
8. Presentación de Clase Final: plática de conferencia de 10min. Vencimiento 12/10

Habrá sesiones de mentoría los viernes antes de que venza cada parte del proyecto final, y se le anima a encontrar un mentor en las primeras sesiones que esté activamente interesado en su proyecto y pueda ayudarlo con más frecuencia. Las sesiones de tutoría pueden ser útiles para identificar si los resultados inesperados son el resultado de un error o, en cambio, son un descubrimiento.

Asegúrate de comenzar a trabajar en el proyecto incluso mientras esperas las revisiones por pares, para que tengas 4-5 semanas para completar la investigación en sí.

### Entregables del proyecto

El proyecto final incluirá los siguientes dos entregables:

1. Una presentación escrita, con vencimiento el lunes a las 8 pm, la semana pasada de clases. La presentación escrita puede contener los siguientes elementos:

- Quién hizo qué (para reflejar la tendencia en las publicaciones)
  - La experiencia general del proyecto
  - Tus descubrimientos
  - Lo que aprendiste de la experiencia (introspección)
2. Una presentación oral, que vence el jueves después de la presentación escrita. Esto permite a los estudiantes tres días para preparar la presentación oral.

## Graduación de proyectos

Seleccionar un proyecto que tenga éxito puede ser difícil. Para ayudar a los estudiantes a optimizar para un proyecto exitoso, les informamos de antemano el esquema de calificación, diseñado para maximizar el impacto del proyecto al ser original, desafiante y relevante para el campo, pero por supuesto, la calificación depende en última instancia del logro general y la claridad de presentación.

Brevemente, la ecuación de calificación para el proyecto final es:

$$\text{min } (O, C, R) \times A + P$$

donde

Originalidad: los experimentos computacionales no originales no se publican

Desafío: el proyecto tiene que ser lo suficientemente difícil

Relevancia - tiene que ser de la biología, no puede simplemente reutilizar algo de otro campo

Logro - si no logras nada no obtendrás una buena calificación

Presentación - aunque hayas logrado un buen proyecto tienes que poder presentarlo para que todos lo sepan, y que se vea fácil. La presentación debe mostrar cómo el proyecto es O, C y R.

Originalidad, Desafío, Relevancia son cada uno de 5 puntos, Logro y Presentación son cada uno de 10.

---

This page titled [1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [1.2: Final Project - Introduction to Research In Computational Biology](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 1.3: Materiales adicionales

### Materiales en línea para el otoño de 2015

lecture1\_transcript.html #Handouts

Además de estas notas estáticas, el curso cuenta con varios recursos en línea:

- El calendario del curso en Google Calendar. Puedes agregar “6.047 Conferencias”, un calendario público.
- El sistema de toma de notas NB para anotar estas notas <http://nb.mit.edu/>

### Libros de texto

lecture1\_transcript.html #CourseInformation Se recomiendan los siguientes tres libros de texto de referencia (opcionales) para la clase.

1. Richard Durbin, Sean R. Eddy, Anders Krogh y Graeme Mitchison, Análisis de Secuencia Biológica: Modelos probabilísticos de proteínas y ácidos nucleicos.
2. Neil Jones y Pavel Pevzner, Una introducción a los algoritmos bioinformáticos.
3. Richard Duda, Peter Hart, David Stork, Clasificación de Patrones.

Cada libro tiene una ventaja diferente. El primer libro es clásico. Es pesado en matemáticas y cubre gran parte de lo que hay en clase. El libro se centra en la alineación de secuencias. Como parte de la teoría de alineamiento de secuencias, el libro aborda Modelos Ocultos de Markov (HMM), métodos de alineación por pares y múltiples, árboles filogenéticos, así como un breve trasfondo en teoría de probabilidad.

El segundo libro pretende equilibrar el rigor matemático y la relevancia biológica. Según el autor, es un buen libro para estudiantes de pregrado. El libro incluye una tabla que asocia algoritmos a problemas biológicos.

El tercer libro trata sobre el aprendizaje automático. Se necesita más un enfoque de ingeniería. Incluye teoría del aprendizaje automático, redes neuronales y, como su nombre indica, reconocimiento de patrones.

---

This page titled [1.3: Materiales adicionales](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [1.3: Additional materials](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 1.4: Curso Crash en Biología Molecular

lecture1\_transcript.html #CentralDogma ADN → ARN → Proteína

El dogma central de la biología molecular describe cómo se almacena e interpreta la información genética en la célula: El código genético de un organismo se almacena en el ADN, que se transcribe en ARN, que finalmente se traduce en proteína. Las proteínas llevan a cabo la mayoría de las funciones celulares como la motilidad, la regulación del ADN y la replicación.

Aunque el dogma central es cierto en la mayoría de las situaciones, hay una serie de notables excepciones al modelo. Por ejemplo, los retrovirus son capaces de generar ADN a partir de ARN mediante transcripción inversa. Además, algunos virus son tan primitivos que ni siquiera tienen ADN, sino que solo usan ARN a proteína.

### 1.4.2 ADN

#### ¿Sabías?

El dogma central a veces se interpreta **incorrectamente** con demasiada fuerza en el sentido de que el ADN solo almacena información inmutable de una generación a otra que permanece idéntica dentro de una generación, el ARN solo se usa como medio de transferencia de información temporal y las proteínas son la única molécula que puede llevar a cabo acciones complejas.

Nuevamente, hay muchas excepciones a esta interpretación, por ejemplo:

- Las mutaciones somáticas pueden alterar el ADN dentro de una generación, y diferentes células pueden tener diferentes contenidos de ADN.
  - Algunas células experimentan alteraciones programadas del ADN durante la maduración, lo que resulta en diferentes contenidos de ADN, la más famosa la inmunidad B y T mientras que las células sanguíneas
  - Las modificaciones epigenéticas del ADN pueden heredarse de una generación a otra
  - El ARN puede desempeñar muchos papeles diversos en la regulación génica, la detección metabólica y la reacción enzimática
- ciones, funciones que antes se pensaba que estaban reservadas a las proteínas.
- Las proteínas en sí mismas pueden sufrir cambios conformacionales que se heredan epigenéticamente sin- modo estados de priones que fueron famosos responsables de la enfermedad de las vacas locas

ADN → ARN → Proteína

Función de ADN

La molécula de ADN almacena la información genética de un organismo. El ADN contiene regiones llamadas genes, que codifican para que se produzcan proteínas. Otras regiones del ADN contienen elementos reguladores, que influyen parcialmente en el nivel de expresión de cada gen. Dentro del código genético del ADN se encuentran tanto los datos sobre las proteínas que necesitan ser codificadas, como los circuitos de control, en forma de motivos reguladores.

Estructura del ADN

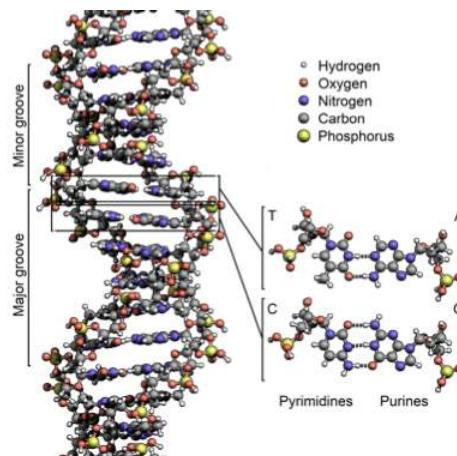
El ADN está compuesto por cuatro nucleótidos: A (adenina), C (citosina), T (timina) y G (guanina). A y G son purinas, las cuales tienen dos anillos, mientras que C y T son pirimidinas, con un anillo. A y T están conectados por dos enlaces de hidrógeno, mientras que C y G están conectados por tres enlaces. Por lo tanto, el emparejamiento A-T es más débil que el emparejamiento C-G. (Por esta razón, la composición genética de las bacterias que viven en aguas termales es 80% G-C). lecture1\_transcript.html #Complementarity

Las dos cadenas de ADN en la doble hélice son complementarias, lo que significa que si hay una A en una hebra, se unirá a una T en la otra, y si hay una C en una hebra, se unirá a una G en la otra. Las cadenas de ADN también tienen dirección, lo que se refiere a las posiciones del anillo de pentosa donde se conecta la cadena principal de fosfato. Esta convención de dirección proviene del hecho de que la ADN y ARN polimerasa sintetizan en la dirección 5' a 3'. Con esto en mente, podemos decir que las cadenas de ADN son antiparalelas, ya que el extremo 5' de una hebra es adyacente al extremo 3' de la otra. Como resultado, el ADN se puede leer tanto en la dirección 3' a 5' como en la dirección 5' a 3', y los genes y otros elementos funcionales se pueden encontrar

en cada una. Por convención, el ADN se escribe de 5' a 3'. Las direcciones 5' y 3' se refieren a las posiciones en el anillo de pentosa donde se conecta la cadena principal de fosfato.

El emparejamiento de bases entre nucleótidos del ADN constituye su estructura primaria y secundaria. Además de la estructura secundaria del ADN, existen varios niveles adicionales de estructura que permiten compactar fuertemente el ADN e influir en la expresión génica (Figura 3). La estructura terciaria describe la torsión en la escalera de ADN que forma una forma helicoidal. En la estructura cuaternaria, el ADN está fuertemente enrollado alrededor de pequeñas proteínas llamadas histonas. Estos complejos de ADN-histona se enrollan adicionalmente en estructuras más ajustadas que se ven en la cromatina.

Antes de que el ADN pueda replicarse o transcribirse en ARN, la estructura de la cromatina debe estar localmente “desempaquetada”. Así, la expresión génica puede ser regulada por modificaciones en la estructura de la cromatina, que hacen más fácil o más difícil que el ADN sea desempaquetado. Esta regulación de la expresión génica a través de la modificación de la cromatina es un ejemplo de epigenética.



© Zephyris en Wikipedia. Algunos derechos reservados. Licencia: CC BY-SA. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 1.2: La estructura de doble hélice del ADN. Los nucleótidos están en el centro, y el esqueleto de azúcar-fosfato se encuentra en el exterior.

### Replicación de ADN

La estructura del ADN, con sus débiles enlaces de hidrógeno entre las bases en el centro, permite separar fácilmente las cadenas con el propósito de replicación del ADN (la capacidad de separar las cadenas de ADN también permite la transcripción, traducción, recombinación y reparación del ADN, entre otras). Esto fue señalado por Watson y Crick como “No ha escapado a nuestro aviso que el emparejamiento específico que hemos postulado de inmediato sugiere un posible mecanismo de copia para el material genético”. En la replicación del ADN, las dos cadenas complementarias se separan, y cada una de las cadenas se utiliza como plantillas para la construcción de una nueva hebra.

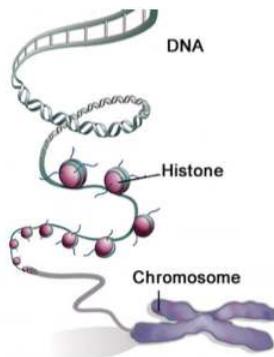
Las ADN polimerasas se unen a cada una de las cadenas en el origen de la replicación, leyendo cada hebra existente desde la dirección 3' a 5' y colocando bases complementarias de manera que la nueva cadena crezca en la dirección 5' a 3'. Debido a que la nueva hebra debe crecer de 5' a 3', una hebra (la hebra principal) puede copiarse continuamente, mientras que la otra (la hebra rezagada) crece en trozos que luego son pegados entre sí por la ADN ligasa. El resultado final son 2 piezas bicatenarias de ADN, donde cada una está compuesta por 1 hebra vieja, y 1 nueva hebra; por esta razón, la replicación del ADN es semiconservativa.

Muchos organismos tienen su ADN roto en varios cromosomas. Cada cromosoma contiene dos hebras de ADN, que son complementarias entre sí pero que se leen en direcciones opuestas. Los genes pueden aparecer en cualquiera de las cadenas de ADN. El ADN antes de un gen (en la región 5') se considera “aguas arriba” mientras que el ADN después de un gen (en la región 3') se considera “aguas abajo”.

### 1.4.3 Transcripción

[lecture1\\_transcript.html #Transcription](#)

ADN → ARN → Proteína



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Qiu, Jane. "Epigenética: Sinfonía inconclusa". *Naturaleza* 441, núm. 7090 (2006): 143-45.

Figura 1.3: El ADN se empaqueta sobre varias capas de organización en un cromosoma compacto

### Generación de ARNm

La transcripción es el proceso mediante el cual se produce ARN usando un molde de ADN. El ADN se desenrolla parcialmente para formar una “burbuja”, y la ARN polimerasa es reclutada en el sitio de inicio de la transcripción (TSS) por complejos proteicos reguladores. La ARN polimerasa lee el ADN desde la dirección 3' a 5' y coloca bases complementarias para formar ARN mensajero (ARNm). El ARN usa los mismos nucleótidos que el ADN, excepto que se usa Uracilo en lugar de Timina.

### Modificaciones postranscripcionales

El ARNm en eucariotas experimenta modificaciones postraduccionales, o procesos que editan aún más la cadena de ARNm. Lo más notable es que un proceso llamado splicing elimina intrones, interviniendo regiones que no codifican para proteínas, de modo que solo quedan las regiones codificantes, los exones. Diferentes regiones del transcripto primario pueden ser empalmadas para conducir a diferentes productos proteicos (corte y empalme alternativo). De esta manera, se puede generar un enorme número de moléculas diferentes en base a diferentes permutaciones de corte y empalme.

Además del corte y empalme, se procesan ambos extremos de la molécula de ARNm. El extremo 5' está tapado con un nucleótido de guanina modificado. En el extremo 3', se agregan aproximadamente 250 residuos de adenina para formar una cola de poli (A).

## RNA

[lecture1\\_transcript.html #RNA](#)

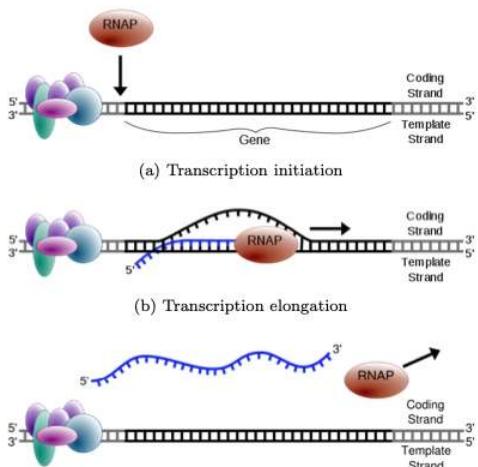
ADN → ARN → Proteína El

ARN se produce cuando se transcribe ADN. Es estructuralmente similar al ADN, con las siguientes diferencias principales:

1. Se utiliza el nucleótido uracilo (U) en lugar de la timina del ADN (T).
2. El ARN contiene ribosa en lugar de desoxirribosa (la desoxirribosa carece de la molécula de oxígeno en la posición 2' que se encuentra en la ribosa).
3. El ARN es monocatenario, mientras que el ADN es bicatenario.

Las moléculas de ARN son el paso intermedio para codificar una proteína. Las moléculas de ARN también tienen funciones catalíticas y reguladoras. Un ejemplo de función catalítica es en la síntesis de proteínas, donde el ARN es parte del ribosoma.

Hay muchos tipos diferentes de ARN, incluyendo:



Cortesía de Forluvoft en Wikipedia. Imágenes en el dominio público.

Figura 1.4: El ARN se produce a partir de un molde de ADN durante la transcripción. Se abre una “burbuja” en el ADN, permitiendo que la ARN polimerasa entre y coloque bases complementarias al ADN.

#### a) Iniciación de la transcripción

#### b) Alargamiento de la transcripción

#### c) Terminación de la transcripción

1. El ARNm (ARN mensajero) contiene la información para elaborar una proteína y se traduce en secuencia proteica.
2. El ARNt (ARN de transferencia) especifica la traducción de codón a aminoácido. Contiene un anti-codón de 3 pares de bases complementario a un codón en el ARNm, y lleva el aminoácido correspondiente a su anticodón unido a su extremo 3'.
3. El ARNr (RBA ribosómico) forma el núcleo del ribosoma, el orgánulo responsable de la traducción del ARNm a proteína.
4. El ARNsA (ARN nuclear pequeño) está involucrado en el corte y empalme (eliminación de intrones de) pre- ARNm, así como otras funciones.

Existen otros tipos funcionales de ARN y aún se están descubriendo. Aunque generalmente se piensa que las proteínas llevan a cabo funciones celulares esenciales, las moléculas de ARN pueden tener estructuras tridimensionales complejas y realizar diversas funciones en la célula.

Según la hipótesis del “mundo del ARN”, los primeros años de vida se basaban enteramente en el ARN. El ARN sirvió tanto como repositorio de información (como el ADN hoy en día) como el caballo de batalla funcional (como la proteína hoy en día) en los primeros organismos. Se cree que la proteína surgió después a través de los ribosomas, y se cree que el ADN surgió en último lugar, vía transcripción inversa.

## Traducción

lecture1\_transcript.html #Translation

ADN → ARN → Proteína

### Traducción

A diferencia de la transcripción, en la que los nucleótidos permanecieron como medio de codificación de información tanto en ADN como en ARN, cuando el ARN se traduce en proteína, la estructura primaria de la proteína está determinada por la secuencia de aminoácidos de la que está compuesta. Dado que existen 20 aminoácidos y solo 4 nucleótidos, las secuencias de 3 nucleótidos en el ARNm, conocidas como codones, codifican para cada uno de los 20 aminoácidos.

Cada una de las 64 posibles 3 secuencias de nucleótidos (codón) especifica de manera única un aminoácido particular, o es un codón de parada que termina la traducción de proteínas (el codón de inicio también codifica metionina). Dado que hay 64 posibles secuencias de codones, el código es degenerado, y algunos aminoácidos se especifican mediante múltiples codificaciones. La mayor parte de la degeneración ocurre en la posición del 3er codón.

### Modificaciones postraduccionales

Al igual que el ARNm, la proteína también sufre modificaciones adicionales que afectan su estructura y función. Un tipo de modificación postraduccional (PTM) implica la introducción de nuevos grupos funcionales a los aminoácidos. Más notablemente, la fosforilación es el proceso mediante el cual se añade un grupo fosfato a un aminoácido que puede activar o desactivar la proteína por completo. Otro tipo de PTM es la escisión de enlaces peptídicos. Por ejemplo, la hormona insulina se escinde dos veces después de la formación de enlaces disulfuro dentro de la proteína original.

SECOND POSITION					
	U	C	A	G	
U	phenylalanine	serine	tyrosine	cysteine	U
	leucine		stop	stop	C
			stop	tryptophan	A
					G
C	leucine	proline	histidine	arginine	U
			glutamine		C
A	isoleucine	threonine	asparagine	serine	A
	* methionine		lysine	arginine	G
G	valine	alanine	aspartic acid	glycine	U
			glutamic acid		C
					A
					G

\* and start

Figura 1.5: Esta tabla de codones muestra a cuál de los 20 aminoácidos se traduce cada uno de los codones de 3 nucleótidos en ARNm. En rojo están los codones de parada, que terminan la traducción.

## Proteína

ADN → ARN → Proteína

La proteína es la molécula responsable de llevar a cabo la mayoría de las tareas de la célula, y puede tener muchas funciones, como enzimática, contráctil, transporte, sistema inmune, señal y receptor por nombrar algunas. Al igual que el ARN y el ADN, las proteínas son polímeros elaborados a partir de subunidades repetitivas. En lugar de nucleótidos, sin embargo, las proteínas están compuestas por aminoácidos.

Cada aminoácido tiene propiedades especiales de tamaño, carga, forma y acidez. Como tal, la estructura adicional emerge más allá simplemente de la secuencia de aminoácidos (la estructura primaria), como resultado de las interacciones entre los aminoácidos. Como tal, la forma tridimensional, y por lo tanto la función, de una proteína está determinada por su secuencia. Sin embargo, determinar la forma de una proteína a partir de su secuencia es un problema sin resolver en biología computacional.

## Regulación: de las moléculas a la vida

lecture1\_transcript.html #Regulation

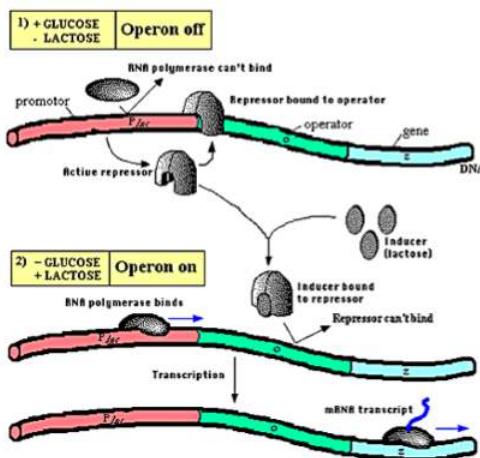
No todos los genes se expresan al mismo tiempo en una célula. Por ejemplo, las células desperdiciarían energía si producían transportador de lactosa en ausencia de lactosa. Es importante que una célula sepa qué genes debe expresar y cuándo. Se involucra una red reguladora para controlar el nivel de expresión de genes en una circunstancia específica.

La transcripción es una de las etapas en las que se pueden regular los niveles de proteína. La región promotora, un segmento de ADN que se encuentra aguas arriba (más allá del extremo 5') de los genes, funciona en la regulación transcripcional. La región promotora contiene motivos que son reconocidos por proteínas llamadas factores de transcripción. Cuando se unen, los factores de transcripción pueden reclutar ARN polimerasa, lo que lleva a la transcripción génica. Sin embargo, los factores de transcripción también pueden participar en complejas interacciones reguladoras. Puede haber múltiples sitios de unión en un promotor, que pueden actuar como una puerta lógica para la activación génica. La regulación en eucariotas puede ser extremadamente compleja, con la expresión génica afectada no solo por la región promotora cercana, sino también por potenciadores y represores distantes.

Podemos usar modelos probabilísticos para identificar genes que están regulados por un factor de transcripción dado. Por ejemplo, dado el conjunto de motivos que se sabe que se unen a un factor de transcripción dado, podemos calcular la probabilidad de que un

motivo candidato también se une al factor de transcripción (ver las notas para el precepto #1). También se puede utilizar el análisis comparativo de secuencias para identificar motivos reguladores, ya que los motivos reguladores muestran patrones característicos de conservación evolutiva.

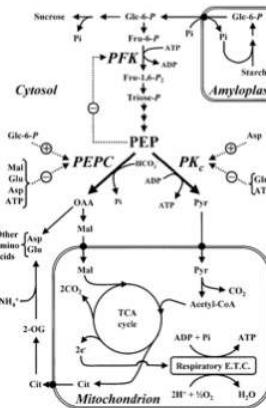
El operón lac en *E. coli* y otras bacterias es un ejemplo de un circuito regulador simple. En las bacterias, los genes con funciones relacionadas a menudo se localizan uno al lado del otro, controlados por la misma región reguladora, y se transcriben juntos; este grupo de genes se llama operón. El operón lac funciona en el metabolismo de la lactosa de azúcar, la cual puede ser utilizada como fuente de energía. Sin embargo, las bacterias prefieren usar la glucosa como fuente de energía, por lo que si hay glucosa presente en el ambiente las bacterias no quieren producir las proteínas que están codificadas por el operón lac. Por lo tanto, la transcripción del operón lac está regulada por un circuito elegante en el que la transcripción se produce sólo si hay lactosa pero no glucosa presente en el ambiente.



### Induction of the Lac Operon

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 1.6: El operón Lac ilustra un sistema de regulación biológica simple. En presencia de glucosa, los genes al metabolismo de la lactosa resultan porque la glucosa inactiva una proteína activadora. En ausencia de lactosa, una proteína represora también resulta el operón. Los genes del metabolismo de la lactosa se expresan únicamente en presencia de lactosa y ausencia de glucosa.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 1.7: Las vías metabólicas y la regulación pueden ser estudiadas por biología computacional. Los modelos se elaboran a partir de información a escala genómica y se utilizan para predecir la función metabólica y para la ingeniería Un ejemplo de ingeniería biológica es modificar el genoma de las bacterias para sobreexpresar artemesina, un antibiótico utilizado para tratar la malaria.

## Metabolismo

lecture1\_transcript.html#

Los organismos vivos están hechos de bloques de construcción autoorganizados. La fuente de energía es necesaria para organizar los bloques. El mecanismo básico involucrado en los bloques de construcción es degradar las moléculas pequeñas para obtener energía para construir moléculas grandes. El proceso de degradar moléculas para liberar energía se llama catabolismo y el proceso de usar energía para ensamblar moléculas más complejas se llama anabolismo. El anabolismo y el catabolismo son ambos procesos metabólicos. El metabolismo regula el flujo de masa y energía para mantener un organismo en un estado de baja entropía.

Las enzimas son un componente crítico de las reacciones metabólicas. La gran mayoría de (¡pero no todos!) las enzimas son proteínas. Muchas reacciones biológicamente críticas tienen altas energías de activación, por lo que la reacción no catalizada ocurriría extremadamente lenta o nada en absoluto. Las enzimas aceleran estas reacciones, para que puedan ocurrir a un ritmo que sea sustentable para la célula. En las células vivas, las reacciones se organizan en vías metabólicas. Una reacción puede tener muchos pasos, con los productos de un paso sirviendo como sustrato para el siguiente. Además, las reacciones metabólicas a menudo requieren una inversión de energía (notablemente como una molécula llamada ATP), y la energía liberada por una reacción puede ser capturada por una reacción posterior en la ruta. Las vías metabólicas también son importantes para la regulación de las reacciones metabólicas si se inhibe cualquier paso, los pasos posteriores pueden carecer del sustrato o la energía que necesitan para proceder. A menudo, los puntos de control regulatorios aparecen temprano en las vías metabólicas, ya que si es necesario detener la reacción, obviamente es mejor detenerla antes de que se haya invertido mucha energía.

## Biología de Sistemas

lecture1\_transcript.html #SystemsBiology

La biología de sistemas se esfuerza por explorar y explicar el comportamiento que surge de las complejas interacciones entre los componentes de un sistema biológico. Un artículo reciente interesante en biología de sistemas es “Metabolic gene regulation in a dynamically changing environment” (Bennett et al., 2008). Este trabajo hace la suposición de que la levadura es un sistema lineal, invariable en el tiempo, y ejecuta una señal (glucosa) a través del sistema para observar la respuesta. Se observa una respuesta periódica a las fluctuaciones de baja frecuencia en el nivel de glucosa, pero hay poca respuesta a las fluctuaciones de alta frecuencia en el nivel de glucosa. Así, este estudio encuentra que la levadura actúa como un filtro de paso bajo para las fluctuaciones en el nivel de glucosa.

## Biología Sintética

lecture1\_transcript.html #SyntheticBiology

No solo podemos usar enfoques computacionales para modelar y analizar datos biológicos recopilados de células, sino que también podemos diseñar celdas que implementen circuitos lógicos específicos para llevar a cabo funciones novedosas. La tarea de diseñar nuevos sistemas biológicos se conoce como biología sintética.

Un éxito particularmente notable de la biología sintética es la mejora de la producción de artemesinina. Artemesinina es un medicamento utilizado para tratar la malaria. Sin embargo, la artemisinina era bastante cara de producir. Recientemente, se ha diseñado una cepa de levadura para sintetizar un precursor del ácido artemisínico a la mitad del costo anterior.

## Organismos modelo y biología humana

Existen diversos organismos modelo para todos los aspectos de la biología humana. Importancia del uso de organismos modelo a un nivel apropiado de complejidad.

Nota: En este libro en particular, nos centraremos en la biología humana, y usaremos ejemplos de la levadura de panadero *Saccharomyces cerevisiae*, la mosca de la fruta *Drosophila melanogaster*, el gusano nematodo *Caenorhabditis elegans* y el ratón casero *Mus musculus*. Trataremos la evolución bacteriana solo en el contexto de la metagenómica del microbioma humano.

---

This page titled [1.4: Curso Crash en Biología Molecular](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.4: Crash Course in Molecular Biology** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 1.5: Introducción a algoritmos e inferencia probabilística

1. Rápidamente revisaremos alguna probabilidad básica considerando una forma alternativa de representar motivos: una matriz de peso de posición (PWM). Nos gustaría modelar el hecho de que las proteínas pueden unirse a motivos que no están completamente especificados. Es decir, algunas posiciones pueden requerir un cierto nucleótido (por ejemplo, A), mientras que otras posiciones son libres para ser un subconjunto de los 4 nucleótidos (por ejemplo, A o C). Un PWM representa el conjunto de todas las secuencias de ADN que pertenecen al motivo mediante el uso de una matriz que almacena la probabilidad de encontrar cada uno de los 4 nucleótidos en cada posición en el motivo. Por ejemplo, considere el siguiente PWM para un motivo con longitud 4:

	1	2	3	4
A	0.6	0.25	0.10	1.0
G	0.4	0.25	0.10	0.0
T	0.0	0.25	0.40	0.0
C	0.0	0.25	0.40	0.0

Decimos que este motivo puede generar secuencias de longitud 4. Los PWM suelen suponer que la distribución de una posición no está influenciada por la base de otra posición. Observe que cada posición está asociada con una distribución de probabilidad sobre los nucleótidos (suman a 1 y no son negativos).

2. También podemos modelar la distribución de fondo de los nucleótidos (la distribución que se encuentra a través del genoma):

A	0.1
G	0.4
T	0.1
C	0.4

Observe cómo las probabilidades para A y T son las mismas y las probabilidades de G y C son las mismas. Esto es consecuencia de la complementariedad del ADN que asegura que la composición global de A y T, G y C es la misma en general en el genoma.

3. Considera la secuencia  $S = GCAA$ .

- La probabilidad de que el motivo genere esta secuencia es

$$P(S|M) = 0.4 \times 0.25 \times 0.1 \times 1.0 = 0.01.$$

- La probabilidad de que el fondo genere esta secuencia

$$P(S|B) = 0.4 \times 0.4 \times 0.1 \times 0.1 = 0.0016.$$

4. Solo esto no es particularmente interesante. Sin embargo, dada la fracción de secuencias que son generadas por el motivo, por ejemplo  $P(M) = 0.1$ , y suponiendo que todas las demás secuencias son generadas por el fondo ( $P(B) = 0.9$ ) podemos calcular la probabilidad de que el motivo genere la secuencia usando la Regla de Bayes:

$$\begin{aligned} P(M|S) &= \frac{P(S|M)P(M)}{P(S)} \\ &= \frac{P(S|M)P(M)}{P(S|B)P(B) + P(S|M)P(M)} \\ &= \frac{0.01 \times 0.1}{0.0016 \times 0.9 + 0.01 \times 0.1} = 0.40984 \end{aligned}$$

This page titled [1.5: Introducción a algoritmos e inferencia probabilística](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [1.5: Introduction to algorithms and probabilistic inference](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 2: Alineación de Secuencias y Programación Dinámica

- 2.1: Introducción
- 2.2: Alineación de secuencias
- 2.3: Formulaciones de problemas
- 2.4: Programación dinámica
- 2.5: El algoritmo de Needleman-Wunsch
- 2.6: Alineación múltiple
- 2.7: Herramientas y Técnicas
- 2.8: Apéndice
- 2.9: Bibliografía

---

This page titled [2: Alineación de Secuencias y Programación Dinámica](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 2.1: Introducción

La alineación de secuencias es una poderosa herramienta capaz de revelar los patrones y funciones de los genes. Si dos regiones genéticas son similares o idénticas, el alineamiento de secuencias puede demostrar los elementos conservados o diferencias entre ellas. La evolución ha conservado dos amplias clases de elementos funcionales en el genoma. Dichos elementos pre-servidos entre especies suelen ser homólogos<sup>1</sup>, ya sean secuencias ortólogas o parálogas (consulte el Apéndice 2.11.1). Ambas clases de elementos conservados pueden ayudar a demostrar la función o historia evolutiva de una secuencia génica. Resuelta principalmente mediante métodos computacionales (la mayoría de las veces programación dinámica), la alineación de secuencias es una forma rápida y poderosa de encontrar similitudes entre genes o genomas. Estas notas discuten el problema de alineación de secuencias, la técnica de programación dinámica y una solución específica al problema usando esta técnica.

---

<sup>1</sup> Las secuencias homólogas son secuencias genómicas descendientes de un ancestro común.

This page titled [2.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **2.1: Introduction** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.2: Alineación de secuencias

La alineación de secuencias representa el método de comparación de dos o más cadenas genéticas, como ADN o ARN. Estas comparaciones ayudan con el descubrimiento de puntos en común genéticos y con el rastreo (implícito) de la evolución de las hebras. Hay dos tipos principales de alineación:

- Alineación global: un intento de alinear cada elemento de una cadena genética, más útil cuando las hebras genéticas consideradas son de aproximadamente el mismo tamaño. La alineación global también puede terminar en brechas.
- Alineación local: un intento de alinear regiones de secuencias que contienen motivos de secuencia similares dentro de un contexto más amplio.

### Ejemplo de Alineación

Dentro de las secuencias génicas ortólogas, existen islas de conservación, o tramos relativamente grandes de nucleótidos que se conservan entre generaciones. Estas regiones conservadas suelen implicar elementos funcionales y viceversa. Como ejemplo, se consideró el alineamiento de la región intergénica Gal10-Gal1 para cuatro especies diferentes de levaduras, la primera alineación cruzada del genoma completo de especies (Figura 2.1). Al mirar este alineamiento, observamos que algunas áreas son más similares que otras, lo que sugiere que estas áreas se han conservado a través de la evolución. En particular, observamos algunos pequeños motivos conservados como CGG y CGC, que de hecho son elementos funcionales en la unión de Gal4 [8].<sup>2</sup> Este ejemplo destaca cómo los datos evolutivos pueden ayudar a localizar áreas funcionales del genoma: los niveles de conservación por nucleótido denotan la importancia de cada nucleótido, y los exones se encuentran entre los elementos más conservados en el genoma.

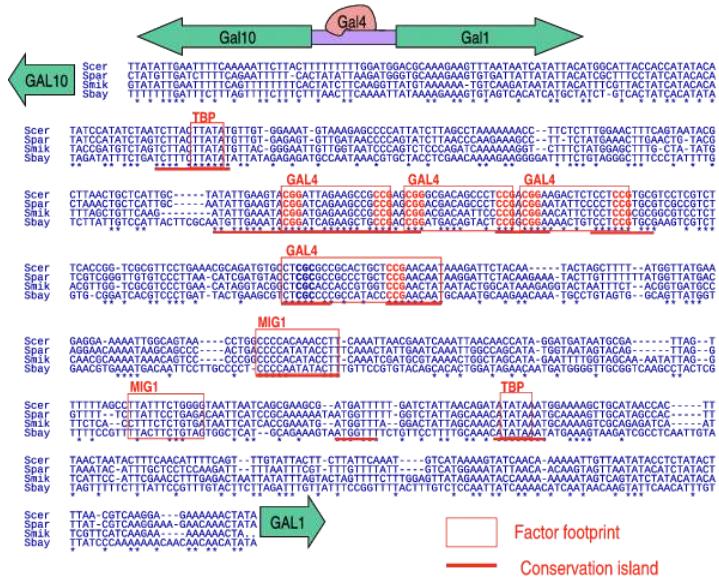
Tenemos que ser cautelosos con nuestras interpretaciones, sin embargo, porque la conservación a veces ocurre por casualidad aleatoria. Para extraer información biológica precisa de alineaciones de secuencias tenemos que separar las firmas verdaderas del ruido. El enfoque más común de este problema implica modelar el proceso evolutivo. Mediante el uso de frecuencias de sustitución de codones conocidas y restricciones de estructura secundaria de ARN, por ejemplo, podemos calcular la probabilidad de que la evolución actuó para preservar una función biológica. Ver Capítulo?? para una discusión en profundidad sobre el modelado evolutivo y la conservación funcional en el contexto de la anotación genómica.

### Resolución de alineación de secuencias

Los genomas cambian con el tiempo, y la escasez de genomas antiguos hace prácticamente imposible comparar los genomas de las especies vivas con los de sus antepasados. Así, nos limitamos a comparar solo los genomas de descendientes vivos. El objetivo del alineamiento de secuencias es inferir las 'operaciones de edición' que cambian un genoma observando solo estos puntos finales.

Debemos hacer algunas suposiciones a la hora de realizar el alineamiento de secuencias, aunque sólo sea porque debemos transformar un problema biológico en uno computacionalmente factible y requerimos un modelo con relativa simplicidad y trazabilidad. En la práctica, la evolución de la secuencia se debe principalmente a mutaciones, delecciones e inserciones de nucleótidos (Figura 2.2). Por lo tanto, nuestro modelo de alineación de secuencias solo considerará estas tres operaciones e ignorará otros eventos realistas que ocurren con menor probabilidad (por ejemplo, duplicaciones).<sup>3</sup>

1. Una mutación nucleotídica ocurre cuando algún nucleótido en una secuencia cambia a otro nucleótido durante el curso de la evolución.
2. Una delección de nucleótidos ocurre cuando se elimina algún nucleótido de una secuencia durante el curso de la evolución.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 2.1: Alineación de secuencias de Gal10-Gal1 entre cuatro cepas de levadura. Los asteriscos marcan los nucleótidos conservados.

3. Una inserción de nucleótidos ocurre cuando se agrega algún nucleótido a una secuencia durante el curso de la evolución.

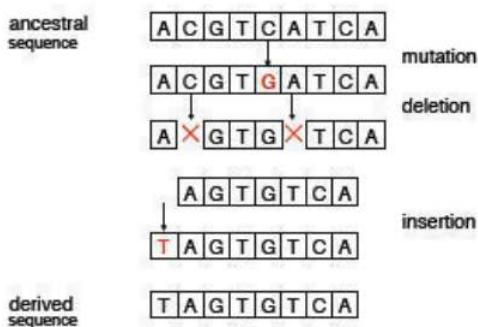


Figura 2.2: Cambios evolutivos de una secuencia genética

Tenga en cuenta que estos tres eventos son todos reversibles. Por ejemplo, si un nucleótido N muta en algún nucleótido M, también es posible que el nucleótido M pueda mutar al nucleótido N. De manera similar, si se elimina el nucleótido N, el evento puede revertirse si el nucleótido N es (re) insertado. Claramente, un evento de inserción se invierte por un evento de eliminación correspondiente.

Esta reversibilidad es parte de una suposición de diseño mayor: la reversibilidad en el tiempo. Específicamente, cualquier evento en nuestro modelo es reversible en el tiempo. Por ejemplo, una delección de nucleótidos que avanza en el tiempo puede verse como una inserción de nucleótidos que retrocede en el tiempo. Esto es útil porque estaremos alineando secuencias que ambas existen en el presente. Para comparar la relación evolutiva, pensaremos en nosotros mismos siguiendo una secuencia hacia atrás en el tiempo a un ancestro común y luego continuaremos adelante en el tiempo a la otra secuencia. Al hacerlo, podemos evitar el problema de no tener una secuencia de nucleótidos ancestrales.

Tenga en cuenta que la reversibilidad en el tiempo es útil para resolver algunos problemas biológicos, pero en realidad no se aplica a



Figura 2.3: La alineación de secuencias humana con ratón es análoga al rastreo hacia atrás del ser humano a un ancestro común, luego hacia adelante al ratón

sistemas biológicos. Por ejemplo, CpG<sup>4</sup> puede emparejarse incorrectamente con un TpG o CpA durante la replicación del ADN, pero la operación inversa no puede ocurrir; de ahí que esta transformación no sea reversible en el tiempo. Para ser muy claros, la reversibilidad en el tiempo es simplemente una decisión de diseño en nuestro modelo; no es inherente a la biología<sup>5</sup>.

También necesitamos alguna manera de evaluar nuestras alineaciones. Hay muchas secuencias posibles de eventos que podrían cambiar un genoma en otro. Quizás los más obvios minimizan el número de eventos (es decir, mutaciones, inserciones y delecciones) entre dos genomas, pero también son posibles secuencias de eventos en los que muchas inserciones son seguidas de delecciones correspondientes. Queremos establecer un criterio de optimalidad que nos permita escoger la “mejor” serie de eventos que describan los cambios entre genomas.

Elegimos invocar la navaja de Occam y seleccionar un método de máxima parsimonia como nuestro criterio de optimalidad. Es decir, en general, deseamos minimizar el número de eventos utilizados para explicar las diferencias entre dos secuencias de nucleótidos. En la práctica, encontramos que es más probable que ocurran mutaciones puntuales que inserciones y delecciones, y ciertas mutaciones son más probables que otras [11]. Nuestro método de parsimonia debe tener en cuenta estas y otras desigualdades a la hora de maximizar la parsimonia. Esto lleva a la idea de una matriz de sustitución y una penalización por gap, las cuales se desarrollan en los siguientes apartados. Tenga en cuenta que no fue necesario elegir un método de parsimonia máxima para nuestro criterio de optimalidad. Podríamos elegir un método probabilístico, por ejemplo usando Modelos Ocultos de Markov (HMM), que asignaría una medida de probabilidad sobre el espacio de posibles rutas de eventos y usaría otros métodos para evaluar alineaciones (por ejemplo, métodos bayesianos). Observe la dualidad entre estos dos enfoques: nuestro método de parsimonia máxima refleja la creencia de que los eventos de mutación tienen baja probabilidad, por lo que al buscar soluciones que minimicen el número de eventos estamos maximizando implícitamente su probabilidad.

---

2. Gal4 de hecho muestra una estructura particular, que comprende dos brazos que cada uno se une a la misma secuencia, en orden inverso.

3. Curiosamente, las decisiones de modelado tomadas para mejorar la trazabilidad no necesariamente resultan en una relevancia disminuida; por ejemplo, tener en cuenta la direccionalidad en el estudio de las inversiones cromosómicas produce soluciones polinomiales en tiempo para un problema de NP de otro modo. [6]

4. p denota la cadena principal de fosfato en una cadena de ADN

5. Este es un ejemplo donde entender la biología ayuda mucho al diseño, e ilustra el principio general de que el éxito en la biología computacional requiere un fuerte conocimiento de los fundamentos tanto de la CS como de la biología. Advertencia: los informáticos que ignoran la biología trabajarán demasiado duro.

---

This page titled [2.2: Alineación de secuencias](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.2: Aligning Sequences](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.3: Formulaciones de problemas

En esta sección, presentamos un problema simple, lo analizamos y aumentamos iterativamente su complejidad hasta que se parezca mucho al problema de alineación de secuencias. Esta sección debe ser vista como un calentamiento para la Sección 2.5 sobre el algoritmo Needleman-Wunsch.

### Formulación 1: Subcadena común más larga

Como primer intento, supongamos que tratamos las secuencias de nucleótidos como cadenas sobre el alfabeto A, C, G y T. Dadas dos de estas cadenas, S1 y S2, podríamos intentar alinearlas encontrando la subcadena común más larga entre ellas. En particular, estas subcadenas no pueden tener brechas en ellas.

Como ejemplo, si  $S1 = \text{ACGTCA}$  y  $S2 = \text{TAGTGTCA}$  (consulte la Figura 2.4), la subcadena común más larga entre ellos es GTCA. Entonces en esta formulación, podríamos alinear S1 y S2 a lo largo de su subcadena común más larga, GTCA, para obtener la mayor cantidad de coincidencias. Un algoritmo simple sería intentar alinear S1 con diferentes desplazamientos de S2 y realizar un seguimiento de la coincidencia de subcadenas más larga encontrada hasta ahora. Tenga en cuenta que este algoritmo es cuadrático en la longitud de la secuencia más corta, que es más lenta de lo que preferiríamos para un problema tan simple.

Figura 2.4: Ejemplo de formulación de subcadena común más larga

### Formulación 2: Subsecuencia común más larga (LCS)

Otra formulación es permitir brechas en nuestras subsecuencias y no limitarnos solo a subcadenas sin huecos. Dada una secuencia,  $X = (x_1, \dots, x_m)$  definimos formalmente  $Z = (z_1, \dots, z_k)$  que es una subsecuencia de X si existe una secuencia estrictamente creciente  $i_1 < i_2 < \dots < i_k$  de índices de X tal que para todos  $j, x_{i_j} = z_j$  (CLRS 350-1).

En el problema de la subsecuencia común más larga (LCS), se nos dan dos secuencias X e Y y queremos encontrar la subsecuencia común Z de longitud máxima. Consideraremos el ejemplo de las secuencias  $S1 = \text{ACGTCA}$  y  $S2 = \text{TAGTGTCA}$  (consulte la Figura 2.5). La subsecuencia común más larga es AGTTCA, una coincidencia más larga que solo la subcadena común más larga.

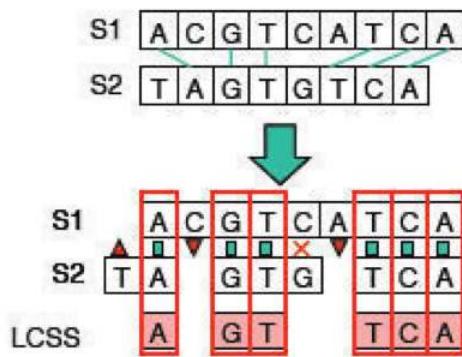


Figura 2.5: Ejemplo de formulación de subsecuencia común más larga

### Formulación 3: Alineación de Secuencias como Distancia de Edición

#### Formulación

La formulación anterior de LCS está cerca del problema de alineación de secuencias completas, pero hasta ahora no hemos especificado ninguna función de costo que pueda diferenciar entre los tres tipos de operaciones de edición (inserción, eliminación y sustitución). Implícitamente, nuestra función de costo ha sido uniforme, lo que implica que todas las operaciones son igualmente probables. Dado que las sustituciones son mucho más probables, queremos sesgar nuestra solución LCS con una función de costo que prefiera las sustituciones sobre las inserciones y eliminaciones.

Refundimos la alineación de secuencias como un caso especial del problema clásico Edit-Distance<sup>6</sup> en informática (CLRS 366). Agregamos penalizaciones variables para diferentes operaciones de edición para reflejar ocurrencias biológicas. Un razonamiento biológico para esta decisión de puntuación es la probabilidad de que las bases se transcriban incorrectamente durante la polimerización. De las cuatro bases nucleotídicas, A y G son purinas (más grandes, dos anillos fusionados), mientras que C y T son

pirimidinas (más pequeñas, un anillo). Así, la ADN polimerasa <sup>7</sup> es mucho más probable que confunda dos purinas o dos pirimidinas ya que son similares en estructura. La matriz de puntuación en la Figura 2.6 modela las consideraciones anteriores. Tenga en cuenta que la tabla es simétrica, esto es compatible con nuestro diseño reversible en el tiempo.

	A	G	T	C
A	+1	-½	-1	-1
G	-½	+1	-1	-1
T	-1	-1	+1	-½
C	-1	-1	-½	+1

Figura 2.6: Matriz de costos para coincidencias y desajustes

El cálculo de las puntuaciones implica alternar entre la interpretación probabilística de la frecuencia con la que ocurren los eventos biológicos y la interpretación algorítmica de asignar una puntuación para cada operación. El problema es encontrar la secuencia de operación menos costosa (según la matriz de costos) que pueda transformar la secuencia inicial de nucleótidos en la secuencia de nucleótidos final.

## Complejidad de la distancia de edición

Todos los algoritmos para resolver la distancia de edición entre dos cadenas operan en tiempo casi polinómico. En 2015, Backurs e Indyk [?] publicó una prueba de que la distancia de edición no se puede resolver más rápido que  $O(n^2)$  en el caso general. Este resultado depende de la Hipótesis Fuerte del Tiempo Exponencial (SETH), que establece que los problemas NP-completos no pueden resolverse en tiempo subexponencial en el peor de los casos.

#### 2.3.4 Formulación 4: Modelos de Costos Variables

Biológicamente, el costo de crear una brecha es más caro que el costo de extender una brecha ya creada. Así, podríamos crear un modelo que dé cuenta de esta variación de costos. Hay muchos modelos de este tipo que podríamos usar, incluyendo los siguientes:

- Penalización por hueco lineal: Costo fijo para todos los huecos (igual que la formulación 3).
  - Penalización por brecha afín: Imponga un costo inicial grande para abrir una brecha, luego un costo incremental pequeño para cada extensión de brecha.
  - Penalización por brecha general: Permitir cualquier función de costo. Tenga en cuenta que esto puede cambiar el tiempo de ejecución asintótico de nuestro algoritmo.
  - Penalización por brecha consciente de fotogramas: Adapte la función de costo para tener en cuenta las interrupciones del marco de codificación (los indels que causan cambios de fotograma en los elementos funcionales generalmente causan modificaciones fenotípicas importantes).

### 2.3.5 Enumeración

Recordemos que para resolver la formulación de la Subcadena Común Más Larga, simplemente podríamos enumerar todas las alineaciones posibles, evaluar cada una y seleccionar la mejor. Esto se debió a que solo hubo alineaciones O ( $n$ ) de las dos secuencias. Una vez que permitimos brechas en nuestra alineación, sin embargo, ya no es así. Es un problema conocido que no se puede enumerar el número de todas las alineaciones con huecos posibles (al menos cuando las secuencias son largas). Por ejemplo, con dos secuencias de longitud 1000, el número de posibles alineaciones supera el número de átomos en el universo.

Dada una métrica para puntuar una alineación dada, el algoritmo simple de fuerza bruta enumera todas las alineaciones posibles, calcula la puntuación de cada una y elige la alineación con la puntuación máxima. Esto lleva a la pregunta: '¿Cuántas alineaciones posibles hay?' Si considera solo NBA<sup>8</sup> n > m, el número de alineaciones es

```
\left(\begin{array}{c} n+m \\ m \end{array}\right) = \frac{(n+m)!}{n! m!} \approx \frac{(2n)!}{(n!)^2} \approx \frac{\sqrt{4\pi n}}{2^n} \frac{(2n)^{2n}}{(n)^n} e^{2n} \approx \frac{e^{2n}}{\sqrt{\pi n}}
```

Este número crece extremadamente rápido, y para valores de  $n$  tan pequeños 30 es demasiado grande ( $> 10^{17}$ ) para que esta estrategia de enumeración sea factible. Por lo tanto, usar un algoritmo mejor que la fuerza bruta es una necesidad.

<sup>6</sup> La distancia de edición o distancia Levenshtein es una métrica para medir la cantidad de diferencia entre dos secuencias (por ejemplo, la distancia Levenshtein aplicada a dos cadenas representa el número mínimo de ediciones necesarias para transformar una cadena en otra).

<sup>7</sup> La ADN polimerasa es una enzima que ayuda a copiar una cadena de ADN durante la replicación.

<sup>8</sup> Alineaciones no aburridas, o alineaciones donde los huecos siempre se emparejan con nucleótidos.

---

This page titled [2.3: Formulaciones de problemas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.3: Problem Formulations](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.4: Programación dinámica

Antes de proceder a una solución del problema de alineación de secuencias, primero discutimos la programación dinámica, un método general y poderoso para resolver problemas con ciertos tipos de estructura.

### Teoría de la Programación Dinámica

La programación dinámica se puede utilizar para resolver problemas con:

1. Subestructura Óptima: La solución óptima a una instancia del problema contiene soluciones óptimas a los subproblemas.
2. Subproblemas superpuestos: Hay un número limitado de subproblemas, muchos/la mayoría de los cuales se repiten muchas veces.

La programación dinámica suele utilizarse, pero no siempre, para resolver problemas de optimización, similar a los algoritmos codiciosos. A diferencia de los algoritmos codiciosos, que requieren que una propiedad de elección codiciosa sea válida, la programación dinámica funciona en una variedad de problemas en los que las elecciones localmente óptimas no producen resultados óptimos a nivel mundial. El Apéndice 2.11.3 analiza con más detalle la distinción entre algoritmos codiciosos y programación dinámica; en términos generales, los algoritmos codiciosos resuelven una clase de problemas más pequeña que la programación dinámica.

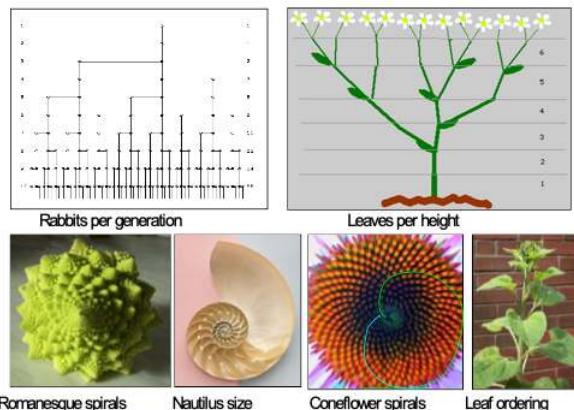
En la práctica, resolver un problema usando programación dinámica implica dos partes principales: Configurar la programación dinámica y luego realizar cálculos. La configuración de la programación dinámica suele requerir los siguientes 5 pasos:

1. Encontrar una parametrización 'matricial' del problema. Determinar el número de dimensiones (variables).
2. Asegurar que el espacio subproblemático sea polinomial (no exponencial). Tenga en cuenta que si se usa una pequeña porción de subproblemas, entonces la memorización puede ser mejor; de manera similar, si la reutilización de subproblemas no es extensa, la programación dinámica puede no ser la mejor solución para el problema.
3. Determinar un orden transversal efectivo. Los subproblemas deben estar listos (resueltos) cuando sean necesarios, por lo que el orden de cálculos importa.
4. Determinar una fórmula recursiva: Un problema mayor generalmente se resuelve en función de sus subpartes.
5. Recuerde opciones: Por lo general, la fórmula recursiva implica un paso de minimización o maximización. Además, a menudo se necesita una representación para almacenar punteros transversales, y la representación debe ser polinómica.

Una vez que se configura la programación dinámica, el cálculo suele ser sencillo:

1. Completar sistemáticamente la tabla de resultados (y generalmente los punteros de seguimiento) y encontrar una puntuación óptima.
2. Trazback desde la puntuación óptima a través de los punteros para determinar una solución óptima.

## Números de Fibonacci



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 2.7: Los ejemplos de números de Fibonacci en la naturaleza son ubicuos.

Los números de Fibonacci proporcionan un ejemplo instructivo de los beneficios de la programación dinámica. La secuencia de Fibonacci se define recursivamente como  $F_0 = F_1 = 1$ ,  $F_n = F_{n-1} + F_{n-2}$  para  $n \leq 2$ . Desarrollamos un algoritmo para calcular el número  $n$  de Fibonacci, y luego refinarlo primero usando memoización y luego usando programación dinámica para ilustrar conceptos clave.

### La Solución Náive

El enfoque simple de arriba hacia abajo es simplemente aplicar la definición recursiva. El Listado 1 muestra una implementación simple de Python.

```

1 # Assume n is a non-negative integer.
2 def fib(n):
3     if n == 0 or n == 1:
4         return 1
5     else:
6         return fib(n - 1) + fib(n - 2)

```

Listado 2.1: Implementación de Python para computar números de Fibonacci recursivamente.

Pero este algoritmo de arriba hacia abajo se ejecuta en tiempo exponencial. Es decir, si  $T(n)$  es el tiempo que lleva computar el  $n^{\circ}$  número de Fibonacci, tenemos ese  $T(n) = T(n - 1) + T(n - 2) + O(1)$ , así  $T(n) = O(\phi^n)$ <sup>9</sup>. El problema es que estamos repitiendo el trabajo resolviendo el mismo subproblema muchas veces.

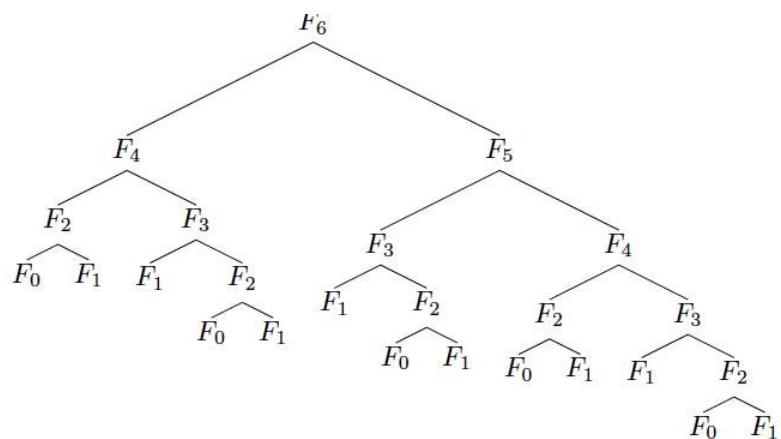


Figura 2.8: El árbol de recursión para el procedimiento de fib que muestra subproblemas repetidos. El tamaño del árbol es  $O(\phi^n)$ , donde  $\phi$  es la proporción áurea.

### La solución de memorización

Una mejor solución que todavía utiliza el enfoque de arriba hacia abajo es memoizar las respuestas a los subproblemas. El Listado 2 da una implementación de Python que usa memorización.

```
1 # Assume n is a non-negative integer.
2 fibs = {0: 1, 1: 1} # stores subproblem answers
3 def fib(n):
4     if n not in fibs:
5         x = fib(n - 2)
6         y = fib(n - 1)
7         fibs[n] = x + y
8
9 return fibs[n]
```

Listado 2.2: Implementación de Python para calcular números de Fibonacci mediante memorización.

Tenga en cuenta que esta implementación ahora se ejecuta en  $T(n) = O(n)$  tiempo porque cada subproblema se calcula como máximo una vez.

La solución de programación dinámica

Para calcular el  $n$ ésimo número de Fibonacci, en lugar de comenzar con  $F(n)$  y usar recursión, podemos iniciar el cálculo desde abajo ya que sabemos que vamos a necesitar todos los subproblemas de todos modos. De esta manera, omitiremos gran parte del trabajo repetido que haría el enfoque de arriba hacia abajo naïve, y podremos calcular el  $n$ ésimo número de Fibonacci en  $O(n)$  tiempo.

Como ejercicio formal, podemos aplicar los pasos señalados en la sección 2.4.1:

1. Encuentra una parametrización 'matricial': En este caso, la matriz es unidimensional; solo hay una a cualquier subproblema  $F(x)$ .
2. Asegúrese de que el espacio del subproblema sea polinomio: Dado que solo hay  $n - 1$  subproblemas, el espacio es polinomio.
3. Determinar un orden transversal efectivo: Como se mencionó anteriormente, aplicaremos un orden transversal de abajo hacia arriba (es decir, computar los subproblemas en orden ascendente).
4. Determine una fórmula recursiva: Esta es simplemente la conocida recurrencia  $F(n) = F(n-1) + F(n-2)$ .
5. Recordar elecciones: En este caso no hay nada que recordar, ya que no se tomaron decisiones en la fórmula recursiva.

El Listado 3 muestra una implementación de Python de este enfoque.

Este método está optimizado para usar solo espacio constante en lugar de una tabla completa ya que solo necesitamos la respuesta a cada subproblema una vez. Pero en general las soluciones de programación dinámica, queremos almacenar las soluciones a los subproblemas en una tabla ya que es posible que necesitemos utilizarlas varias veces sin volver a calcular sus respuestas. Dichas soluciones se parecerían algo a la solución de memorización en el Listado 2, pero generalmente serán de abajo hacia arriba en lugar de de arriba hacia abajo. En este ejemplo particular, la distinción entre la solución de memorización y la solución de programación dinámica es mínima ya que ambos enfoques calculan todas las soluciones de subproblemas y las utilizan el mismo número de veces. En general, la memorización es útil cuando no se computarán todos los subproblemas, mientras que la programación dinámica guarda la sobrecarga de las llamadas a funciones recursivas, y por lo tanto es preferible cuando todas las soluciones de subproblemas deben calcularse<sup>10</sup>. Se pueden encontrar ejemplos adicionales de programación dinámica en línea [7].

## Alineación de Secuencias mediante Programación Dinámica

Ahora estamos listos para resolver el problema más difícil de la alineación de secuencias mediante programación dinámica, la cual se presenta en profundidad en la siguiente sección. Tenga en cuenta que la visión clave para resolver el problema de alineación de secuencias es que las puntuaciones de alineación son aditivas. Esto nos permite crear una matriz  $M$  indexada por  $i$  y  $j$ , que son posiciones en dos secuencias  $S$  y  $T$  a alinear. La mejor alineación de  $S$  y  $T$  corresponde con la mejor trayectoria a través de la matriz  $M$  después de rellenarla usando una fórmula recursiva.

Mediante el uso de programación dinámica para resolver el problema de alineación de secuencias, logramos una solución demostrablemente óptima, que es mucho más eficiente que la enumeración de fuerza bruta.

<sup>9</sup>  $\phi$  es la proporción áurea, i.e.  $\frac{1+\sqrt{5}}{2}$

<sup>10</sup> En algunos casos, la programación dinámica es prácticamente la única solución aceptable; este es el caso en particular cuando las cadenas de dependencia entre subproblemas son largas: en este caso, la solución basada en memorización recurre demasiado profundamente y provoca un desbordamiento de pila

---

2.4: Programación dinámica is shared under a CC BY-NC-SA license and was authored, remixed, and/or curated by LibreTexts.

- 2.4: Dynamic Programming is licensed CC BY-NC-SA 4.0.

## 2.5: El algoritmo de Needleman-Wunsch

Ahora utilizaremos la programación dinámica para abordar el problema más difícil de la alineación general de secuencias. Dadas dos cadenas  $S = (S_1, \dots, S_n)$  y  $T = (T_1, \dots, T_m)$ , queremos encontrar la subsecuencia común más larga, que puede o no contener huecos. En lugar de maximizar la longitud de una subsecuencia común, queremos calcular la subsecuencia común que optimice la puntuación según lo definido por nuestra función de puntuación. Que  $d$  denote el costo de penalización por gap y  $s(x; y)$  la puntuación de alinear una base  $x$  y una base  $y$ , estos se deducen de las probabilidades de inserción/deletión y sustitución que se pueden determinar experimentalmente o observando secuencias que sabemos que están estrechamente relacionadas. El algoritmo que desarrollaremos en las siguientes secciones para resolver la alineación de secuencias se conoce como el algoritmo Needleman-Wunsch.

### Programación dinámica vs. memorización

Antes de sumergirnos en el algoritmo, una nota final sobre la memorización está en orden. Al igual que el problema de Fibonacci, el problema de alineación de secuencias se puede resolver en un enfoque de arriba hacia abajo o de abajo hacia arriba.

En un *enfoque recursivo de arriba hacia abajo* podemos usar la memoización para crear un diccionario potencialmente grande indexado por cada uno de los subproblemas que estamos resolviendo (secuencias alineadas). Esto requiere  $O(n^2m^2)$  espacio si indexamos cada subproblema por los puntos inicial y final de las subsecuencias para las que se necesita calcular una alineación óptima. La ventaja es que resolvemos cada subproblema a lo sumo una vez: si no está en el diccionario, el problema se calcula y luego se inserta en el diccionario para mayor referencia.

En un *enfoque iterativo de abajo hacia arriba* podemos usar programación dinámica. Definimos el orden de los subproblemas de computación de tal manera que una solución a un problema se computa una vez que se han resuelto los subproblemas relevantes. En particular, los subproblemas más simples vendrán antes que los más complejos. Esto elimina la necesidad de realizar un seguimiento de qué subproblemas se han resuelto (el diccionario en memorización se convierte en una matriz) y asegura que no haya trabajo duplicado (cada subalineación se calcula solo una vez).

Así, en este caso particular, la única diferencia práctica entre memorización y programación dinámica es el costo de las llamadas recursivas incurridas en el caso de memorización (el uso del espacio es el mismo).

### Declaración del problema

Supongamos que tenemos un alineamiento óptimo para dos secuencias  $S_{1\dots n}$  y  $T_{1\dots m}$  en el que Si coincide con  $T_j$ . La visión clave es que esta alineación óptima se compone de una alineación óptima entre  $(S_1, \dots, S_{i-1})$  y  $(T_1, \dots, T_{i-1})$  y una alineación óptima entre  $(S_{i+1}, \dots, S_n)$  y  $(T_{j+1}, \dots, T_m)$ . Esto se desprende de un argumento de cortar y pegar: si una de estas alineaciones parciales es subóptima, entonces cortamos y pegamos una mejor alineación en lugar de la subóptima. Esto logra una mayor puntuación de la alineación general y, por lo tanto, contradice la optimalidad de la alineación global inicial. En otras palabras, cada subrayectoria en una ruta óptima también debe ser óptima. Observe que las puntuaciones son aditivas, por lo que la puntuación de la alineación general es igual a la suma de las puntuaciones de las alineaciones de las subsecuencias. Esto supone implícitamente que los subproblemas de cálculo de las alineaciones de puntuación óptimas de las subsecuencias son independientes. Necesitamos motivar biológicamente que tal suposición conduzca a resultados significativos.

### Espacio de índice de subproblemas

Ahora necesitamos indexar el espacio de subproblemas. Dejar  $F_{i,j}$  ser la puntuación de la alineación óptima de  $(S_1, \dots, S_i)$  y  $(T_1, \dots, T_i)$ . El espacio de los subproblemas es  $\{F_{i,j}, i \in [0, |S|], j \in [0, |T|]\}$ . Esto nos permite mantener una  $(m+1) \times (n+1)$  matriz F con las soluciones (es decir, puntuaciones óptimas) para todos los subproblemas.

### Optimalidad local

Podemos calcular la solución óptima para un subproblema haciendo una elección local óptima basada en los resultados de los subproblemas más pequeños. Así, necesitamos establecer una función recursiva que muestre cómo la solución a un problema determinado depende de sus subproblemas. Y usamos esta definición recursiva para llenar la tabla F de una manera ascendente.

Podemos considerar las 4 posibilidades (insertar, eliminar, sustituir, emparejar) y evaluar cada una de ellas en función de los resultados que hemos calculado para subproblemas menores. Para inicializar la tabla, establecemos  $F_{0,j} = -j \cdot d$  y  $F_{i,0} = -i \cdot d$  ya que esas son las puntuaciones de alinear  $(T_1, \dots, T_i)$  con  $j$  huecos y  $(S_1, \dots, S_i)$  con  $i$  huecos (también conocido como superposición cero entre las dos secuencias). Luego atravesamos la matriz columna por columna calculando la puntuación óptima para cada subproblema de alineación considerando las cuatro posibilidades:

- La Secuencia S tiene un hueco en la posición de alineación actual.
- La Secuencia T tiene un hueco en la posición de alineación actual.
- Hay una mutación (sustitución de nucleótidos) en la posición actual.
- Hay un partido en la posición actual.

Luego utilizamos la posibilidad que produce la puntuación máxima. Expresamos esto matemáticamente por la fórmula recursiva para  $F_{i,j}$ :

$$F(0,0) = 0$$

$$\text{Initialization : } F(i,0) = F(i-1,0) - d$$

$$F(0,j) = F(0,j-1) - d$$

```
\text{\text{Iteración}}:\quad F(i,j) = \max\left\{ \begin{array}{l} \text{izquierda} \\ F(i-1,j) - d \\ \text{insertar hueco en S} \\ F(i,j-1) - d \\ \text{insertar hueco en T} \\ F(i-1,j-1) + s \left( x_{-i}, y_{-j} \right) \\ \text{coincidencia o mutación} \end{array} \right\}
```

Terminación: Inferior derecha

Después de atravesar la matriz, la puntuación óptima para la alineación global viene dada por  $F_{m,n}$ . El orden transversal tiene que ser tal que tengamos soluciones a subproblemas dados cuando los necesitamos. Es decir, para calcular  $F_{i,j}$ , necesitamos conocer los valores a la izquierda, arriba y diagonalmente arriba  $F_{i,j}$  en la tabla. Así podemos atravesar la tabla en orden mayor de fila o columna o incluso diagonalmente desde la celda superior izquierda hasta la celda inferior derecha. Ahora bien, para obtener la alineación real solo tenemos que recordar las elecciones que hicimos en cada paso.

## Solución Óptima

Las trayectorias a través de la matriz  $F$  corresponden a alineaciones de secuencia óptimas. Al evaluar cada celda  $F_{i,j}$  hacemos una elección seleccionando el máximo de las tres posibilidades. Así, el valor de cada celda (no inicializada) en la matriz se determina ya sea por la celda a su izquierda, por encima de ella, o diagonalmente a la izquierda por encima de ella. Una coincidencia y una sustitución se representan como viajando en la dirección diagonal; sin embargo, se puede aplicar un costo diferente para cada uno, dependiendo de si los dos pares de bases que estamos alineando coinciden o no. Para construir la alineación óptima real, necesitamos rastrear a través de nuestras elecciones en la matriz. Es útil mantener un puntero para cada celda mientras se llena la tabla que muestra qué elección se hizo para obtener la puntuación para esa celda. Entonces podemos simplemente seguir nuestros punteros hacia atrás para reconstruir la alineación óptima.

## Análisis de soluciones

El análisis de tiempo de ejecución de este algoritmo es muy sencillo. Cada actualización lleva  $O(1)$  tiempo, y como hay  $mn$  elementos en la matriz  $F$ , el tiempo total de ejecución es  $O(mn)$ . De igual manera, el espacio total de almacenamiento es  $O(mn)$ . Para el caso más general donde la regla de actualización es más complicada, el tiempo de ejecución puede ser más costoso. Por ejemplo, si la regla de actualización requiere probar todos los tamaños de huecos (por ejemplo, el costo de un hueco no es lineal), entonces el tiempo de ejecución sería  $O(mn(m+n))$ .

## Needleman-Wunsch en la práctica

Supongamos que queremos alinear dos secuencias  $S$  y  $T$ , donde

$$S = AGT$$

T = AAGC

El primer paso es colocar las dos secuencias a lo largo de los márgenes de una matriz e inicializar las celdas de la matriz. Para inicializar asignamos un 0 a la primera entrada de la matriz y luego rellenamos la primera fila y columna con base en la adición incremental de penalizaciones por brecha, como en la Figura 2.9 a continuación. Aunque el algoritmo podría llenar la primera fila y columna a través de la iteración, es importante definir claramente y establecer límites sobre el problema.

El siguiente paso es la iteración a través de la matriz. El algoritmo procede a lo largo de filas o a lo largo de columnas, considerando una celda a la vez. Para cada celda se calculan tres puntuaciones, dependiendo de las puntuaciones de tres celdas de matriz adyacentes (específicamente la entrada anterior, la diagonal hacia arriba y hacia la izquierda, y la de la izquierda). La puntuación máxima de estos tres trazados posibles se asigna a la entrada y también se almacena el puntero correspondiente. La terminación ocurre cuando el algoritmo llega a la esquina inferior derecha. En la Figura 2.10 la matriz de alineación para las secuencias S y T se ha llenado con puntajes y punteros.

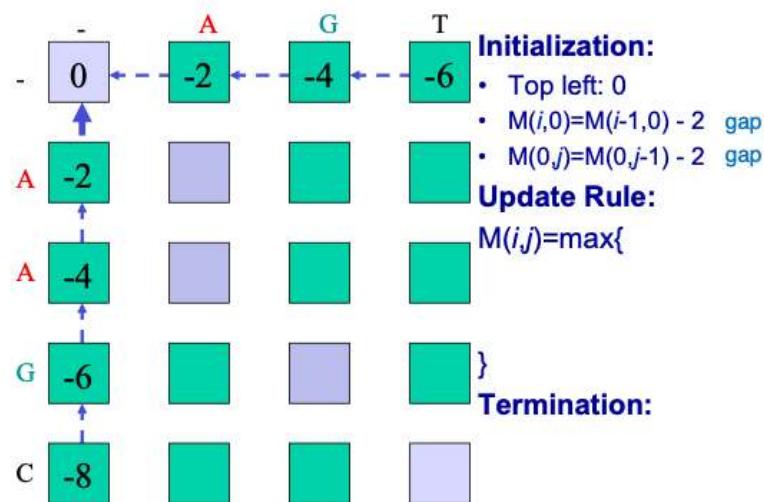


Figura 2.9: (Ejemplo) Configuración inicial para Needleman-Wunsch

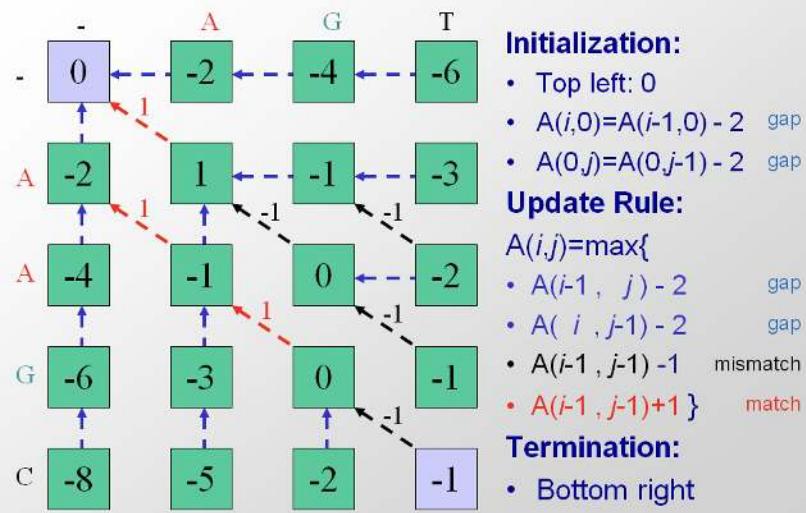


Figura 2.10: (Ejemplo) A mitad del segundo paso de Needleman-Wunsch

El paso final del algoritmo es la trazabilidad óptima de la ruta. En nuestro ejemplo comenzamos en la esquina inferior derecha y seguimos los punteros disponibles hasta la esquina superior izquierda. Al registrar las decisiones de alineación tomadas en cada celda durante el rastreo, podemos reconstruir la alineación de secuencia óptima de extremo a principio y luego invertirla. Obsérvese

que en este caso particular existen múltiples vías óptimas (Figura 2.11). Una implementación de pseudocódigo del algoritmo Needleman-Wunsch se incluye en el Apéndice 2.11.4

## Optimizaciones

El algoritmo dinámico que presentamos es mucho más rápido que la estrategia de fuerza bruta de enumerar alineaciones y funciona bien para secuencias de hasta 10 kilobases de longitud. Sin embargo, a la escala de alineaciones del genoma completo el algoritmo dado no es factible. Para alinear secuencias mucho más grandes podemos hacer modificaciones al algoritmo y mejorar aún más su rendimiento.

### Programación dinámica acotada

Una posible optimización es ignorar las alineaciones de mandrinado leve (MBAs) o las alineaciones que tienen demasiados huecos. Explícitamente, podemos limitarnos a cierta distancia  $W$  de la diagonal en la matriz

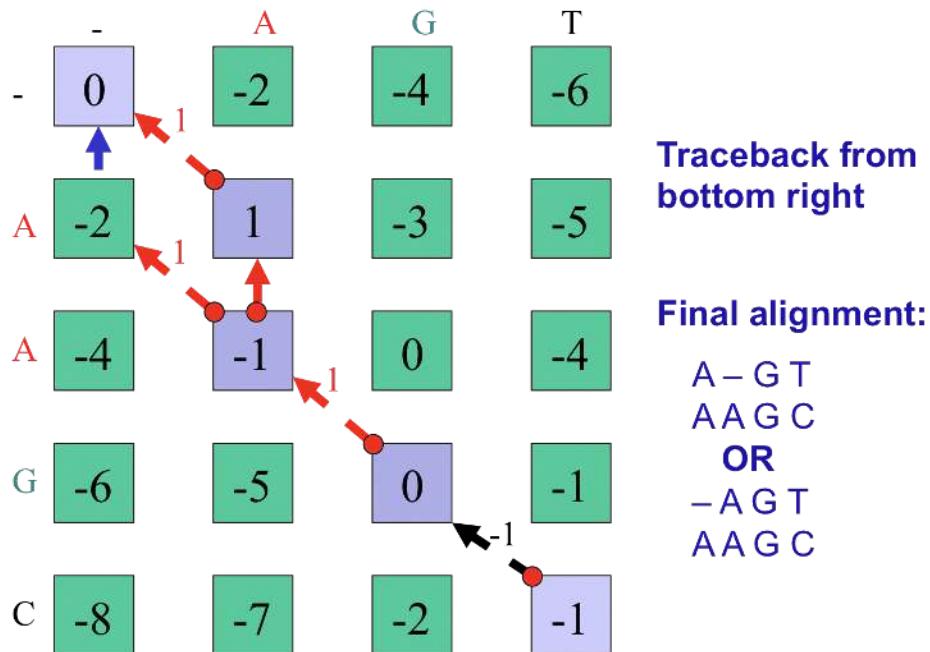
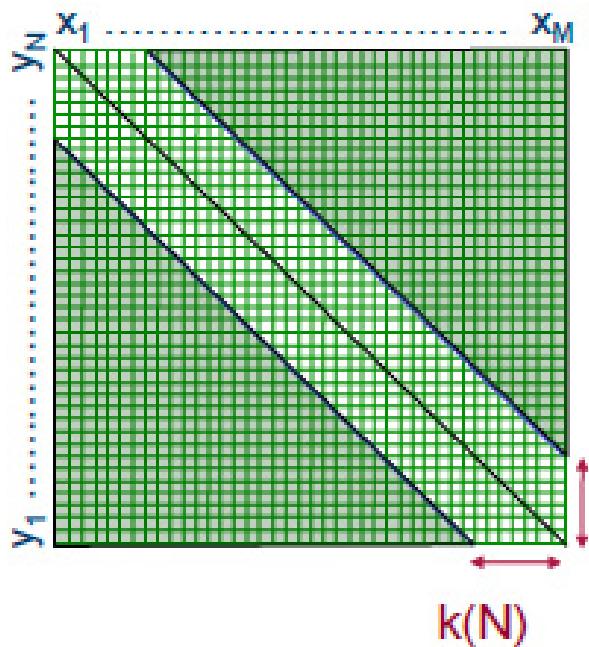


Figura 2.11: (Ejemplo) Rastreo de la alineación óptima

$F$  de subproblemas. Es decir, suponemos que la ruta de optimización en  $F$  de  $F_{0,0}$  a  $F_{m,n}$  está dentro de la distancia  $W$  a lo largo de la diagonal. Esto significa que la recursión (2.2) solo necesita aplicarse a las entradas en  $F$  dentro de la distancia  $W$  alrededor de la diagonal, y esto produce un costo de tiempo/espacio de  $O((m + n)W)$  (consulte la Figura 2.12).



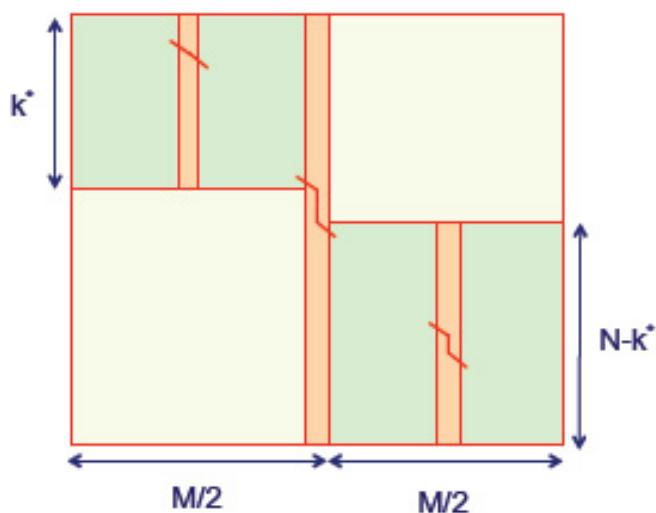
© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 2.12: Ejemplo de programación dinámica acotada

Tenga en cuenta, sin embargo, que esta estrategia es heurística y ya no garantiza una alineación óptima. En cambio, alcanza un límite inferior sobre la puntuación óptima. Esto se puede utilizar en un paso posterior donde descartamos las recursiones en la matriz F que, dado el límite inferior, no puede conducir a una alineación óptima.

#### Alineación lineal de espacios

La recursión (2.2) se puede resolver usando solo espacio lineal: actualizamos las columnas en F de izquierda a derecha durante las cuales solo realizamos un seguimiento de la última columna actualizada que cuesta  $O(m)$  espacio. Sin embargo, además  $F_{m,n}$  de la puntuación de la alineación óptima, también queremos calcular una alineación correspondiente. Si usamos trace back, entonces necesitamos almacenar punteros para cada una de las entradas en F, y esto cuesta  $O(mn)$  espacio.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 2.13: Recuperando el alineamiento de secuencia con  $O(m + n)$  el espacio

¡También es posible encontrar una alineación óptima usando solo espacio lineal! El objetivo es usar dividir y conquistar para calcular la estructura de la alineación óptima para una entrada de matriz en cada paso. La Figura 2.13 ilustra el proceso. La idea clave es que una alineación dinámica de programación pueda proceder con la misma facilidad en la dirección inversa, comenzando en la esquina inferior derecha y terminando en la parte superior izquierda. Entonces, si la matriz se divide por la mitad, entonces tanto un pase directo como un paso inverso pueden correr al mismo tiempo y converger en la columna media. En el punto de cruce podemos sumar las dos puntuaciones de alineación juntas; la celda en la columna media con la puntuación máxima debe caer en el camino óptimo general.

Podemos describir este proceso de manera más formal y cuantitativa. Primero calcula el índice de fila  $u \in 1, \dots, m$  que está en la ruta óptima mientras cruza la  $\frac{n}{2}^{\text{th}}$  columna. For  $1 \leq i \leq m$  y  $n \leq j \leq n$  let  $C_{i,j}$  denotar el índice de fila que está en el camino óptimo  $F_{i,j}$  al cruzar la  $\frac{n}{2}^{\text{th}}$  columna. Después, mientras actualizamos las columnas de F de izquierda a derecha, también podemos actualizar las columnas de C de izquierda a derecha. Entonces, en  $O(mn)$  tiempo y  $O(m)$  espacio somos capaces de calcular la puntuación  $F_{m,n}$  y también  $C_{m,n}$ , que es igual al índice de fila  $u \in 1, \dots, m$  que está en el camino óptimo al cruzar la  $\frac{n}{2}^{\text{th}}$  columna. Ahora entra en juego la idea de dividir y conquistar. Repetimos el procedimiento anterior para la  $u \times \frac{n}{2}$  submatriz superior izquierda de F y también repetimos el procedimiento anterior para la  $(m-u) \times \frac{n}{2}$  submatriz inferior derecha de F. Esto se puede hacer usando el espacio lineal  $O(m+n)$  asignado. El tiempo de ejecución para la submatriz superior izquierda es  $O(\frac{mn}{2})$  y el tiempo de ejecución para la submatriz inferior derecha es  $O(\frac{(m-u)n}{2})$ , lo que sumado da un tiempo de ejecución de  $O(\frac{mn}{2}) = O(mn)$ .

Seguimos repitiendo el procedimiento anterior para submatrices cada vez más pequeñas de F mientras recolectamos más y más entradas de una alineación con puntaje óptimo. El tiempo total de ejecución es  $O(mn) + O(\frac{mn}{2}) + O(\frac{mn}{4}) + \dots = O(2mn) = O(mn)$ . Entonces, sin sacrificar el tiempo de funcionamiento general (hasta un factor constante), dividir y conquistar conduce a una solución espacial lineal (ver también la Sección?? en la Conferencia 3).

---

This page titled [2.5: El algoritmo de Needleman-Wunsch](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.5: The Needleman-Wunsch Algorithm](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.6: Alineación múltiple

### Alineación de tres secuencias

Ahora que hemos visto cómo alinear un par de secuencias, es natural extender esta idea a *múltiples* secuencias. Supongamos que nos gustaría encontrar el alineamiento óptimo de 3 secuencias. ¿Cómo podríamos proceder?

Recordemos que cuando alineamos dos secuencias S y T, elegimos el máximo de tres posibilidades para la posición final del alineamiento (secuencia T alineada contra un hueco, secuencia S alineada contra un hueco, o secuencia S alineada contra secuencia T):

```
\[F_{i,j} = \max\ izquierda\begin{array}{l}
F_{i,j-1} + d \\
F_{i-1,j} + d \\
F_{i-1,j-1} + s\ izquierda(S_{i}, T_{j})\ derecha
\end{array}\ derecha.\ nonumber]
```

Para tres secuencias S, T y U, hay siete posibilidades para la posición final del alineamiento. Es decir, hay tres formas de tener dos huecos en la posición final, tres formas de tener un hueco, y una manera de tener las tres secuencias alineadas\left(\begin{array}{l}

```
3\\
1\\
\end{array}\right) + \left(\begin{array}{l}
3\\
2\\
\end{array}\right) + \left(\begin{array}{l}
3\\
3\\
\end{array}\right) = 7\right)). La regla de actualización es ahora:
```

```
\[F_{i,j,k} = \max\ izquierda\begin{array}{l}
F_{i-1,j,k} + s\ izquierda(S_{i}, -, -)\ derecha\\
F_{i-1,j-1,k} + s\ izquierda(S_{i}, -, T_{j}, -)\ derecha\\
F_{i-1,j,k-1} + s\ izquierda(-, -, U_{k})\ derecha\\
F_{i-1,j-1,k-1} + s\ izquierda(S_{i}, T_{j}, -)\ derecha\\
F_{i-1,j,k-1} + s\ izquierda(S_{i}, -, U_{k})\ derecha\\
F_{i-1,j-1,k-1} + s\ izquierda(-, T_{j}, U_{k})\ derecha\\
F_{i-1,j-1,k-1} + s\ izquierda(S_{i}, T_{j}, U_{k})\ derecha
\end{array}\ derecha.\ nonumber]
```

donde s es la función que describe las puntuaciones de brecha, coincidencia y desajuste.

Este enfoque, sin embargo, es exponencial en el número de secuencias que estamos alineando. Si tenemos k secuencias de longitud  $n$ , calcular la alineación óptima usando una matriz de programación dinámica k-dimensional lleva  $O((2n)^k)$  tiempo (el factor de 2 resulta del hecho de que un k-cubo tiene  $2^k$  vértices, por lo que necesitamos tomar el máximo de  $2^k - 1$  celdas vecinas para cada entrada en la matriz de puntuación). Como puedes imaginar, este algoritmo rápidamente se vuelve poco práctico a medida que aumenta el número de secuencias.

### Alineación múltiple heurística

Un enfoque comúnmente utilizado para el alineamiento de múltiples secuencias se llama alineación múltiple progresiva. Consiste en que conocemos el árbol evolutivo relacionando cada una de nuestras secuencias. Luego comenzamos realizando una alineación por pares de las dos secuencias más estrechamente relacionadas. Esta alineación inicial se llama alineación de semillas. Luego procedemos a alinear la siguiente secuencia más cercana a la semilla, y esta nueva alineación reemplaza a la semilla. Este proceso continúa hasta que se produce la alineación final.

En la práctica, generalmente no conocemos el árbol evolutivo (o árbol guía), esta técnica generalmente se empareja con algún tipo de algoritmo de agrupamiento que puede usar una medida de similitud de baja resolución para generar una estimación del árbol.

Si bien el tiempo de ejecución de este enfoque heurístico se mejora mucho con respecto al método anterior (polinomio en el número de secuencias en lugar de exponencial), ya no podemos garantizar que el alineamiento final sea óptimo.

Tenga en cuenta que aún no hemos explicado cómo alinear una secuencia contra una alineación existente. Un enfoque posible sería realizar alineaciones por pares de la nueva secuencia con cada secuencia ya en el alineamiento semilla (suponemos que cualquier posición en el alineamiento de semillas que ya sea un hueco seguirá siendo una). Luego podemos agregar la nueva secuencia al alineamiento de semillas basado en el mejor alineamiento por pares (este enfoque fue descrito previamente por Feng y Doolittle [4]). Alternativamente, podemos idear una función para puntuar la alineación de una secuencia con otra alineación (tales funciones de puntuación a menudo se basan en la suma por pares de las puntuaciones en cada posición).

El diseño de mejores herramientas de alineación de secuencias múltiples es un área activa de investigación. En la sección 2.7 se detallan algunos de los trabajos actuales en este campo.

---

This page titled [2.6: Alineación múltiple](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.6: Multiple alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.7: Herramientas y Técnicas

Lalign encuentra alineaciones locales entre dos secuencias. Dotlet es un applet Java basado en navegador para visualizar la alineación de dos secuencias en una matriz de puntos.

Las siguientes herramientas están disponibles para la alineación de múltiples secuencias:

- [Clustal Omega](#) - Un programa de alineación de secuencias múltiples que utiliza árboles guía sembrados y HMM técnicas perfil-perfil para generar alineaciones. [10]
- [MUSCLE](#) - Comparación de Secuencias Múltiples por Log-Expectativa [3]
- [T-Coffee](#) - Permite combinar los resultados obtenidos con varios métodos de alineación [2]
- [MAFFT](#) - (Alineación Múltiple mediante Transformada Rápida de Fourier) es un programa de alineación de secuencias múltiples de alta velocidad [5]
- [Kalign](#) - Un algoritmo de alineación de secuencias múltiples rápido y preciso [9]

---

This page titled [2.7: Herramientas y Técnicas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.7: Tools and Techniques](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.8: Apéndice

### Homología

Uno de los objetivos clave del alineamiento de secuencias es identificar secuencias homólogas (por ejemplo, genes) en un genoma. Dos secuencias homólogas están relacionadas evolutivamente, específicamente por descendencia de un ancestro común. Los dos tipos primarios de homólogos son ortólogos y parálogos (consultar la Figura 2.14<sup>11</sup>). Existen otras formas de homología (por ejemplo, xenólogos), pero están fuera del alcance de estas notas.

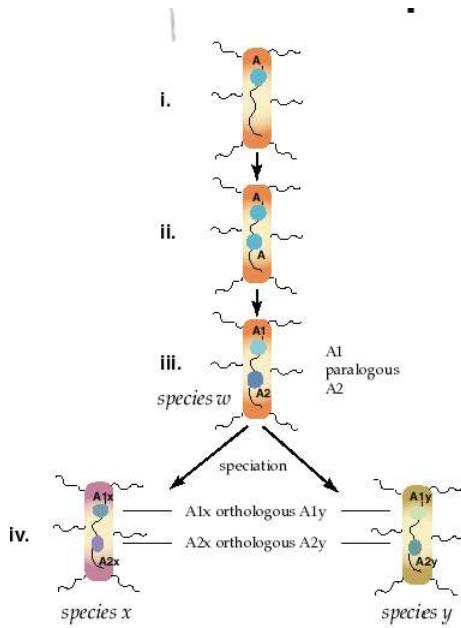


Figura 2.14: Secuencias ortólogas y parálogas © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Los ortólogos surgen de eventos de especiación, dando lugar a dos organismos con una copia del mismo gen. Por ejemplo, cuando una sola especie A se especia en dos especies B y C, hay genes en las especies B y C que descienden de un gen común en la especie A, y estos genes en B y C son ortólogos (los genes continúan evolucionando independientemente entre sí, pero aún realizan la misma función relativa).

Los parálogos surgen de eventos de duplicación dentro de una especie. Por ejemplo, cuando se produce una duplicación génica en algunas especies A, la especie tiene un gen B original y una copia génica B', y los genes B y B' son parálogos. Generalmente, las secuencias ortólogas entre dos especies estarán más estrechamente relacionadas entre sí que las secuencias parálogas. Esto ocurre porque los ortólogos suelen conservar (aunque no siempre) la función a lo largo del tiempo, mientras que los parálogos suelen cambiar con el tiempo, por ejemplo especializando la (sub) función de un gen o evolucionando una nueva función. Como resultado, determinar secuencias ortólogas es generalmente más importante que identificar secuencias parálogas cuando se mide la relación evolutiva.

### Selección Natural

El tema de la selección natural es un tema demasiado grande para resumirlo efectivamente en pocos párrafos cortos; en cambio, este apéndice introduce tres tipos amplios de selección natural: selección positiva, selección negativa y selección neutra.

- La selección positiva ocurre cuando un rasgo es evolutivamente ventajoso y aumenta la aptitud de un individuo, de modo que un individuo con el rasgo es más probable que tenga descendencia (robusta). A menudo se asocia con el desarrollo de nuevos rasgos.
- La selección negativa ocurre cuando un rasgo es evolutivamente desventajoso y disminuye la condición física de un individuo. La selección negativa actúa para reducir la prevalencia de alelos genéticos que reducen la aptitud de una especie. La selección

negativa también se conoce como selección purificadora debido a su tendencia a 'purificar' los alelos genéticos hasta que solo los alelos más exitosos existen en la población.

- La selección neutra describe la evolución que ocurre aleatoriamente, como resultado de que los alelos no afectan a la aptitud de un individuo. En ausencia de presiones selectivas, no se produce ninguna selección positiva o negativa, y el resultado es la selección neutra.

## Programación dinámica v. Algoritmos codiciosos

La programación dinámica y los algoritmos codiciosos son algo similares, y le corresponde a uno conocer las distinciones entre los dos. Los problemas que pueden resolverse mediante programación dinámica suelen ser problemas de optimización que presentan dos rasgos: 1. subestructura óptima y 2. subproblemas superpuestos.

Los problemas solucionables por algoritmos codiciosos requieren tanto estos rasgos como (3) la propiedad de elección codiciosa. Cuando se trata de un problema “en la naturaleza”, a menudo es fácil determinar si satisface (1) y (2) pero difícil determinar si debe tener la propiedad de elección codiciosa. No siempre está claro si las elecciones óptimas a nivel local producirán una solución óptima a nivel mundial.

Para los biólogos computacionales, hay dos puntos útiles a tener en cuenta con respecto a si emplear programación dinámica o programación codiciosa. Primero, si un problema puede resolverse usando un algoritmo codicioso, entonces puede resolverse usando programación dinámica, mientras que lo contrario no es cierto. En segundo lugar, las estructuras problemáticas que permiten algoritmos codiciosos normalmente no aparecen en la biología computacional.

Para dilucidar este segundo punto, podría ser útil considerar las estructuras que permiten que la programación codiciosa funcione, pero tal discusión nos llevaría demasiado lejos. El estudiante interesado (preferiblemente uno con antecedentes matemáticos) debe mirar a los matroides y avaridos, que son estructuras que tienen la propiedad de elección codiciosa. Para nuestros propósitos, simplemente vamos a afirmar que los problemas biológicos suelen involucrar a entidades que son altamente sistémicas y que hay pocas razones para sospechar suficiente estructura en la mayoría de los problemas para emplear algoritmos codiciosos.

## Pseudocódigo para el algoritmo Needleman-Wunsch

El primer problema en el primer conjunto de problemas le pide que termine una implementación del algoritmo Needleman-Wunsch (NW), y el código Python de trabajo para el algoritmo se omite intencionalmente. En cambio, este apéndice resume los pasos generales del algoritmo NW (Sección 2.5) en un solo lugar.

Problema: Dadas dos secuencias S y T de longitud m y n, una matriz de sustitución vU de puntuaciones coincidentes, y una penalización por hueco G, determinan la alineación óptima de S y T y la puntuación del alineamiento.

Algoritmo:

- Crear dos  $m + 1$  por  $n + 1$  matrices A y B. A será la matriz de puntuación y B será la matriz de seguimiento. La entrada  $(i, j)$  de la matriz A contendrá la puntuación de la alineación óptima de las secuencias  $S[1, \dots, i]$  y  $T[1, \dots, j]$ , y la entrada  $(i, j)$  de la matriz B contendrá un puntero a la entrada a partir de la cual se construyó el alineamiento óptimo.
- Inicializar la primera fila y columna de la matriz de puntuación A de tal manera que las puntuaciones tengan en cuenta las penalizaciones por brecha, e inicializar la primera fila y columna de la matriz de trazabilidad B de la manera obvia.
- Pasar por las entradas  $(i, j)$  de la matriz A en algún orden razonable, determinando el alineamiento óptimo de las secuencias  $S[1, \dots, i]$  y  $T[1, \dots, j]$  usando las entradas  $(i - 1, j - 1)$ ,  $(i - 1, j)$  y  $(i, j - 1)$ . Establezca el puntero en la matriz B a la entrada correspondiente a partir de la cual se construyó la alineación óptima en  $(i, j)$ .
- Una vez completadas todas las entradas de las matrices A y B, la puntuación de la alineación óptima se puede encontrar en la entrada  $(m, n)$  de la matriz A.
- Construir la alineación óptima siguiendo la trayectoria de los punteros comenzando en la entrada  $(m, n)$  de la matriz B y terminando en la entrada  $(0, 0)$  de la matriz B.

<sup>11</sup> R.B. - BIOS 60579

- **2.8: Appendix** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 2.9: Bibliografía

- 
- [1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest y Clifford Stein. Introducción a los ritmos algorítmicos. The MIT Press, Londres, tercera edición, 1964.
  - [2] Paolo Di Tommaso, Sébastien Moretti, Ioannis Xenarios, Miquel Orobio, Alberto Montanyola, Jia-Ming Chang, Jean-François Taly, y Cedric Notredame. T-Coffee: un servidor web para el alineamiento múltiple de secuencias de proteínas y ARN utilizando información estructural y extensión de homología. Nucleic Acids Research, 39 (edición del servidor web) :W13—W17, 2011.
  - [3] Robert C. Edgar. MUSCLE: alineación de múltiples secuencias con alta precisión y alto rendimiento. Investigación de ácidos nucleicos, 32 (5) :1792—7, enero de 2004.
  - [4] D F Feng y R F Doolittle. Alineación progresiva de secuencias como requisito previo para corregir árboles filogenéticos. Revista de Evolución Molecular, 25 (4) :351—360, 1987.
  - [5] Kazutaka Katoh, George Asimenos y Hiroyuki Toh. Alineación múltiple de secuencias de ADN con MAFFT. Métodos En Biología Molecular Clifton Nj, 537:39 —64, 2009.
  - [6] John D. Kececioglu y David Sankoff. Límites eficientes para la distancia de inversión cromosómica orientada. En Proceedings of the 5th Annual Symposium on Combinatorial Pattern Matching, CPM '94, páginas 307—325, Londres, Reino Unido, 1994. Springer-Verlag.
  - [7] Manolis Kellis. Problemas de práctica de programación dinámica. <http://people.csail.mit.edu/bdean/6.046/dp/>, septiembre de 2010.
  - [8] Manolis Kellis, Nick Patterson, Matthew Endrizzi, Bruce Birren y Eric S Lander. Secuenciación y comparación de especies de levaduras para identificar genes y elementos reguladores. Naturaleza, 423 (6937) :241—254, 2003.
  - [9] Timo Lassmann y Erik L L Sonnhammer. Kalign—un algoritmo de alineación de múltiples secuencias preciso y rápido. BMC Bioinformática, 6 (1) :298, 2005.
  - [10] Fabian Sievers, Andreas Wilm, David Dineen, Toby J Gibson, Kevin Karplus, Weizhong Li, Rodrigo López, Hamish McWilliam, Michael Remmert, Johannes Söding, Julie D Thompson y Desmond G Higgins. Generación rápida y escalable de alineaciones de secuencias múltiples de proteínas de alta calidad usando Clustal Omega. Biología de Sistemas Moleculares, 7 (539) :539, 2011.
  - [11] Zhaolei Zhang y Mark Gerstein. Patrones de sustitución, inserción y delección de nucleótidos en el genoma humano inferidos a partir de pseudogenes. Nucleic Acids Research, 31 (18) :5338—5348, 2003.
- 

This page titled [2.9: Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [2.9: Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 3: Alineación rápida de secuencias y búsqueda de bases de datos

- 3.1: ¿Qué hemos aprendido?
- 3.2: Introducción
- 3.3: Alineación global vs. alineación local vs. alineación semi-global
- 3.4: Coincidencia exacta de cadenas en tiempo lineal
- 3.5: El algoritmo BLAST (Herramienta Básica de Búsqueda de Alineación Local)
- 3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal
- 3.7: Fundamentos probabilísticos del alineamiento de secuencias

---

This page titled [3: Alineación rápida de secuencias y búsqueda de bases de datos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 3.1: ¿Qué hemos aprendido?

En esta sección exploramos algoritmos de alineación más allá de la alineación global. Comenzamos revisando nuestro uso de la programación dinámica para resolver problemas de alineación global usando el algoritmo Needleman-Wunsch. Luego exploramos alternativas de alineaciones locales (Smith-Waterman) y semi-globales. Luego discutimos el uso de la función hash para hacer coincidir cadenas exactas en tiempo lineal (Karp-Rabin) así como hacer una búsqueda de vecindario, investigando secuencias similares en tiempo lineal probabilístico (principio de encasillado, peines, explosión de 2 golpes, proyecciones aleatorias). También hemos abordado el uso del preprocesamiento para la coincidencia lineal de cadenas de tiempo, así como el fondo probabilístico para la alineación de secuencias.

---

This page titled 3.1: ¿Qué hemos aprendido? is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

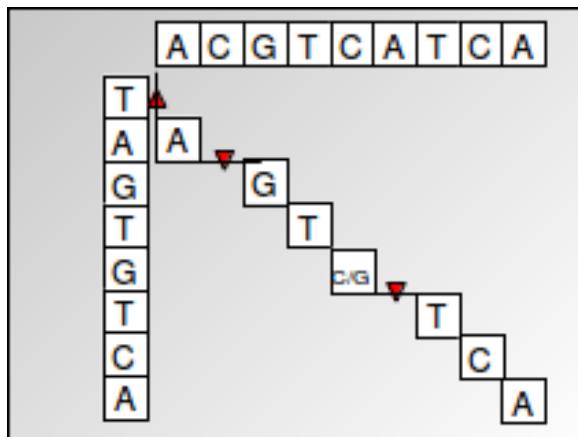
- [3.1: What Have We Learned?](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 3.2: Introducción

En el capítulo anterior, utilizamos programación dinámica para calcular alineaciones de secuencias en  $O(n^2)$ . En particular, aprendimos el algoritmo de alineación global, que empareja secuencias completas entre sí a nivel de nucleótidos. Normalmente aplicamos esto cuando se sabe que las secuencias son homólogas (es decir, las secuencias provienen de organismos que comparten un ancestro común).

La importancia biológica de encontrar alineamientos de secuencias es poder inferir el conjunto más probable de eventos evolutivos como mutaciones puntuales/desapareamientos y huecos (inserciones o delecciones) que ocurrieron para transformar una secuencia en otra. Para ello, primero asumimos que el conjunto de transformaciones con el menor costo es la secuencia de transformaciones más probable. Al asignar costos a cada tipo de transformación (desajuste o brecha) que reflejen sus respectivos niveles de dificultad evolutiva, encontrar una alineación óptima se reduce a encontrar el conjunto de transformaciones que dan como resultado el menor costo general.

Esto lo conseguimos usando un algoritmo de programación dinámica conocido como el algoritmo Needleman-Wunsch. La programación dinámica utiliza subestructuras óptimas para descomponer un problema en subproblemas similares. El problema de encontrar una alineación de secuencias se puede expresar muy bien como un algoritmo de programación dinámica ya que las puntuaciones de alineación son aditivas, lo que significa que encontrar el alineamiento de una secuencia más grande se puede encontrar encontrando recursivamente las alineaciones de subsecuencias más pequeñas. Las puntuaciones se almacenan en una matriz, con una secuencia correspondiente a las columnas y la otra secuencia correspondiente a las filas. Cada célula representa la transformación requerida entre dos nucleótidos correspondientes a la fila y columna de la célula. Se recupera una alineación trazando de nuevo a través de la matriz de programación dinámica (que se muestra a continuación). El enfoque de programación dinámica es preferible a un algoritmo codicioso que simplemente elige la transición con un costo mínimo en cada paso porque un algoritmo codicioso no garantiza que el resultado general dará la alineación óptima o de menor costo.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 3.1: Alineación global

Para resumir el algoritmo de Needleman-Wunsch para la alineación global:

Calculamos las puntuaciones correspondientes a cada celda de la matriz y registramos nuestra elección (memorización) en ese paso, es decir, cuál de las celdas superior, izquierda o diagonal condujo a la puntuación máxima para la celda actual. Nos quedamos con una matriz llena de puntuaciones óptimas en cada posición de celda, junto con punteros en cada celda que reflejan la elección óptima que conduce a esa celda en particular.

Luego podemos recuperar la alineación óptima trazando desde la celda en la esquina inferior derecha (que contiene la puntuación de alinear una secuencia completa con la otra) siguiendo los punteros que reflejan elecciones localmente óptimas, y luego construyendo la alineación correspondiente a una ruta óptima seguida en la matriz.

El tiempo de ejecución del algoritmo Needleman-Wunsch es  $O(n^2)$  ya que para cada celda de la matriz, hacemos una cantidad finita de cálculo. Calculamos 3 valores usando puntuaciones ya calculadas y luego tomamos el máximo de esos valores para

encontrar la puntuación correspondiente a esa celda, que es una operación de tiempo constante ( $O(1)$ ).

Para garantizar la corrección, es necesario computar el costo para cada celda de la matriz. Es posible que la alineación óptima pueda estar compuesta por una mala alineación (consistente en brechas y desajustes) al inicio, seguida de muchos partidos, convirtiéndola en la mejor alineación general. Estos son los casos que atraviesan el límite de nuestra matriz de alineación. Así, para garantizar la óptima alineación global, necesitamos computar cada entrada de la matriz.

El alineamiento global es útil para comparar dos secuencias que se cree que son homólogas. Es menos útil para comparar secuencias con reordenamientos o inversiones o alinear un gen recién secuenciado contra genes de referencia en un genoma conocido, conocido como búsqueda en bases de datos. En la práctica, a menudo también podemos restringir el espacio de alineación a explorar si sabemos que algunas alineaciones son claramente subóptimas.

En este capítulo se abordarán otras formas de algoritmos de alineación para abordar tales escenarios. Primero introducirá el algoritmo de Smith-Waterman para el alineamiento local para alinear subsecuencias en lugar de secuencias completas, en contraste con el algoritmo de Needleman-Wunsch para alineación global. Más adelante, se dará una visión general de los métodos hash y semi-numéricos como el algoritmo de Karp-Rabin para encontrar la subcadena común más larga (contigua) de nucleótidos. Estos algoritmos son implementados y ampliados para emparejamientos inexactos en el programa BLAST, una de las herramientas más citadas y exitosas en biología computacional. Finalmente, este capítulo repasará BLAST para la búsqueda de bases de datos, así como la base probabilística del alineamiento de secuencias y cómo las puntuaciones de alineación pueden interpretarse como razones de verosimilitud.

Esquema:

1. Introducción

- Revisión del alineamiento global (Needleman-Wunsch)

2. Alineación global vs. alineación local vs. alineación semi-global

- Reglas de inicialización, terminación y actualización para alineación global (Needleman-Wunsch) vs. alineación local (Smith-Waterman) vs. alineación semi-global
- Penalizaciones por brecha variable, aceleraciones algorítmicas

3. Coincidencia exacta de cadenas en tiempo lineal

- Algoritmo Karp-Rabin y métodos semi-numéricos • Funciones hash y algoritmos aleatorios

4. El algoritmo BLAST y coincidencia inexacta • Hashing con búsqueda de vecindarios

- Explosión de dos golpes y hash con peines

5. Preprocesamiento para la coincidencia de cadenas en tiempo lineal

- Preprocesamiento fundamental
- Árboles de sufijos
- Matrices de sufijos
- Transformación de Madriás-Wheeler

6. Fundamentos probabilísticos del alineamiento de secuencias • Penalizaciones por desajuste, matrices BLOSUM y

- Significancia estadística de una puntuación de alineación

---

This page titled [3.2: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [3.2: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

### 3.3: Alineación global vs. alineación local vs. alineación semi-global

Una alineación global se define como la alineación de *extremo a extremo* de dos cadenas  $s$  y  $t$ .

Una alineación local de cadenas  $s$  y  $t$  es una alineación de *subcadenas* de  $s$  con subcadenas de  $t$ .

En general se utilizan para encontrar regiones de alta similitud local. A menudo, estamos más interesados en encontrar locales

alineaciones porque normalmente no conocemos los límites de los genes y sólo se puede conservar un pequeño dominio del gen.

En tales casos, no queremos hacer cumplir que otras partes (potencialmente no homólogas) de la secuencia también se alineen. El alineamiento local también es útil cuando se busca un gen pequeño en un cromosoma grande o para detectar cuándo una secuencia larga puede haber sido reordenada (Figura 4).

Una alineación semi-global de cadena  $s$  y  $t$  es una alineación de una subcadena de  $s$  con una subcadena de  $t$ .

Esta forma de alineación es útil para la detección de solapamientos cuando no deseamos penalizar las brechas iniciales o finales.

Para encontrar una alineación semi-global, las distinciones importantes son inicializar la fila superior y la columna más a la izquierda a cero y terminar el extremo en la fila inferior o la columna más a la derecha.

El algoritmo es el siguiente:

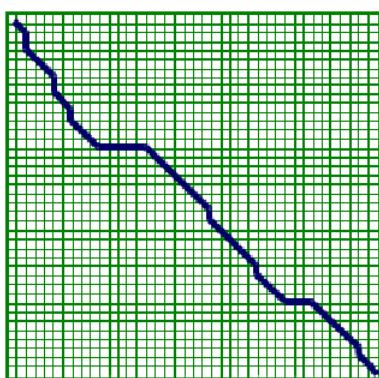


Figura 3.2: Alineación global

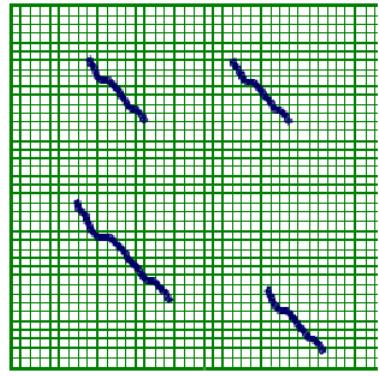


Figura 3.3: Alineación local

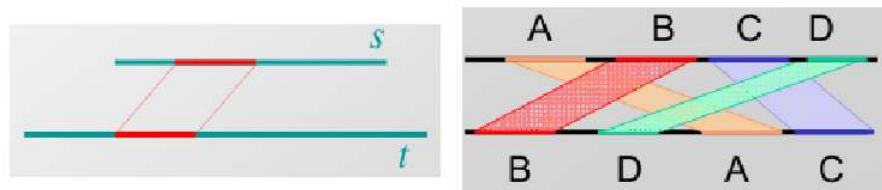


Figura 3.4: Alineaciones locales para detectar reordenamientos

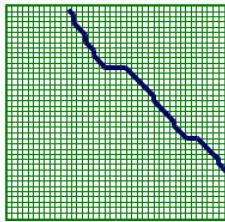


Figura 3.5: Alineación semi-global

```
\[
\begin{array}{l}
\text{Inicialización}:\begin{aligned}
F(i, 0) &= 0 \\
F(0, j) &= 0
\end{aligned} \\
\end{array}
\quad
\begin{array}{l}
\text{iteración}:\ & F(i, j) = \max \left\{ \begin{aligned}
& F(i - 1, j) - d \\
& F(i, j - 1) - d \\
& F(i - 1, j - 1) + s(x_i, y_j)
\end{aligned} \right. \\
& \left. \begin{aligned}
& \text{izquierda } (x_i, y_j) \text{ derecha} \\
& \text{derecha}
\end{aligned} \right. \\
\end{array}
\quad
\begin{array}{l}
\text{Termination : Bottom row or Right column}
\end{array}
```

### Uso de la programación dinámica para alineaciones locales

En esta sección veremos cómo encontrar alineaciones locales con una modificación menor del algoritmo de Needleman-Wunsch que se discutió en el capítulo anterior para encontrar alineaciones globales.

Para encontrar alineaciones globales, se utilizó el siguiente algoritmo de programación dinámica (algoritmo Needleman-Wunsch):

**Initialization :**  $F(0,0)=0$

$$\text{Iteration : } F(i,j) = \max \begin{cases} F(i-1,j) - d \\ F(i,j-1) - d \\ F(i-1,j-1) + s(x_i,y_j) \end{cases} \quad (3.3.1)$$

**Termination :** Bottom right

Para encontrar alineaciones locales solo necesitamos modificar ligeramente el algoritmo de Needleman-Wunsch para comenzar de nuevo y encontrar una nueva alineación local siempre que la puntuación de alineación existente sea negativa. Dado que una alineación local puede comenzar en cualquier lugar, inicializamos la primera fila y columna de la matriz a ceros. El paso de iteración se modifica para incluir un cero para incluir la posibilidad de que iniciar una nueva alineación sea más económico que tener muchos desajustes. Además, dado que la alineación puede terminar en cualquier lugar, necesitamos atravesar toda la matriz para encontrar la puntuación de alineación óptima (no solo en la esquina inferior derecha). El resto del algoritmo, incluido el rastreo, permanece sin cambios, con rastreo indicando un final en cero, indicando el inicio de la alineación óptima.

Estos cambios dan como resultado el siguiente algoritmo de programación dinámica para la alineación local, que también se conoce como:

```
\begin{array}{l}
\begin{array}{l}
\text{Inicialización}: & F(i, 0) = 0 \\
& F(0, j) = 0
\end{array} \\
\end{array}
```

```
\[ \text{Iteración} : \quad F(i, j) = \max\{ izquierda, \begin{array}{c} 0 \\ F(i-1, j) - d \\ F(i, j-1) - d \\ F(i-1, j-1) + s \left( x_i, y_j \right) \end{array} \right] derecha. \nonumber \]
```

*Termination : Anywhere*

## Variaciones algorítmicas

A veces puede resultar costoso tanto en tiempo como en espacio ejecutar estos algoritmos de alineación. Por lo tanto, en esta sección se presentan algunas variaciones algorítmicas para ahorrar tiempo y espacio que funcionan bien en la práctica.

Un método para ahorrar tiempo, es la idea de delimitar el espacio de alineaciones a explorar. La idea es que las buenas alineaciones generalmente se mantengan cerca de la diagonal de la matriz. Así podemos simplemente explorar celdas de matriz dentro de un radio de  $k$  desde la diagonal. El problema con esta modificación es que se trata de una heurística y puede conducir a una solución subóptima ya que no incluye los casos límite mencionados al inicio del capítulo. Sin embargo, esto funciona muy bien en la práctica. Además, dependiendo de las propiedades de la matriz de puntuación, puede ser posible argumentar la corrección del algoritmo de espacio rebatido. Este algoritmo requiere  $O(k * m)$  espacio y  $O(k * m)$  tiempo.

Anteriormente vimos que para calcular la solución óptima, necesitábamos almacenar la puntuación de alineación en cada celda así como el puntero reflejando la elección óptima que conduce a cada celda. Sin embargo, si solo nos interesa la *puntuación óptima de alineación*, y no la alineación real en sí, existe un método para calcular la solución mientras se ahorra espacio. Para calcular la puntuación de cualquier celda solo necesitamos las puntuaciones de la celda anterior, a la izquierda, y a la diagonal izquierda de la celda actual. Al guardar la columna anterior y actual en la que estamos calculando puntuaciones, la solución óptima se puede calcular en el espacio lineal.

Si utilizamos el principio de dividir y conquistar, en realidad podemos encontrar la alineación óptima con el espacio lineal. La idea es que calculemos las alineaciones óptimas desde ambos lados de la matriz, es decir, de izquierda a derecha, y viceversa. Vamos  $u = \lfloor \frac{n}{2} \rfloor$ . Digamos que podemos identificar  $v$  tal que la célula  $(u, v)$  está en el óptimo

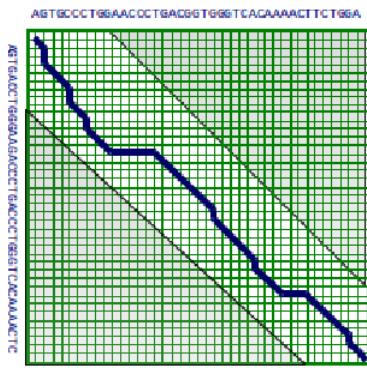


Figura 3.6: Cálculo de espacio rebatido

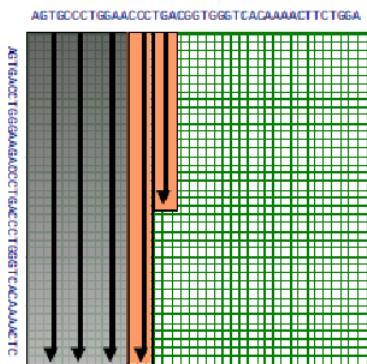
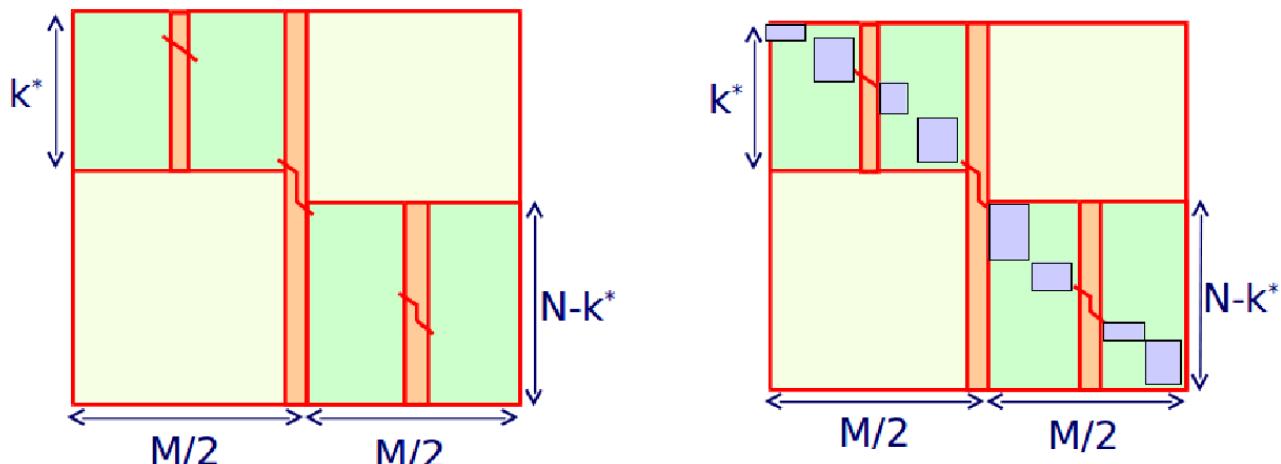


Figura 3.7: Cálculo de espacio lineal para una puntuación de alineación óptima

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

trayectoria de alineación. Eso significa que  $v$  es la fila donde la alineación cruza la columna  $u$  de la matriz. Podemos encontrar la alineación óptima concatenando las alineaciones óptimas de  $(0,0)$  a  $(u, v)$  más la de  $(u, v)$  a  $(m, n)$ , donde  $m$  y  $n$  es la celda inferior derecha (nota: las puntuaciones de alineación de las subalineaciones concatenadas usando nuestro esquema de puntuación son aditivas. Así que hemos aislado nuestro problema a dos problemas separados en las esquinas superior izquierda e inferior derecha de la matriz DP). Entonces podemos seguir dividiendo recursivamente estos subproblemas en subproblemas más pequeños, hasta que estemos abajo a alinear secuencias de longitud 0 o nuestro problema sea lo suficientemente pequeño como para aplicar el algoritmo DP regular. Para encontrar  $v$  la fila en la columna central donde se cruza la alineación óptima simplemente agregamos las puntuaciones entrantes y salientes para esa columna.

Un inconveniente de este enfoque de dividir y conquistar es que tiene un tiempo de ejecución más largo. Sin embargo, el tiempo de ejecución no se incrementa drásticamente. Dado que  $v$  se puede encontrar usando una pasada de DP regular, podemos encontrar  $v$  para cada columna en  $O(mn)$  tiempo y espacio lineal ya que no necesitamos realizar un seguimiento de los punteros de rastreo para este paso. Luego al aplicar el enfoque dividir y conquistar, los subproblemas tardan la mitad del tiempo ya que solo necesitamos hacer un seguimiento de las celdas diagonalmente a lo largo de la trayectoria de alineación óptima (la mitad de la matriz del paso anterior). Eso da un tiempo de ejecución total de  $O(mn(1 + \frac{1}{2} + \frac{1}{4} + \dots)) = O(2MN) = O(mn)$  (usando la suma de series geométricas), para darnos un tiempo de ejecución cuadrático (dos veces más lento que antes, pero sigue siendo el mismo comportamiento asintótico). El tiempo total nunca superará  $2MN$  (el doble del tiempo que el algoritmo anterior). Aunque el tiempo de ejecución se incrementa en un factor constante, una de las grandes ventajas del enfoque de dividir y conquistar es que el espacio se reduce drásticamente a  $O(N)$ .



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 3.9: Divide y vencerás

P: ¿Por qué no usar la variación de espacio rebotado sobre la variación de espacio lineal para obtener tanto el tiempo lineal como el espacio lineal?

R: La variación del espacio rebotado es un enfoque heurístico que puede funcionar bien en la práctica pero no garantiza la alineación óptima.

### Sanciones por brecha generalizada

Las penalizaciones por brecha determinan la puntuación calculada para una subsecuencia y así afectan qué alineación se selecciona. El modelo normal es usar  $a$  donde cada hueco individual en una secuencia de huecos de longitud  $k$  se penaliza por igual con valor  $p$ . Esta penalización puede modelarse como  $w(k) = k * p$ . Dependiendo de la situación, podría ser una buena idea penalizar de manera diferente por, digamos, brechas de diferentes longitudes. Un ejemplo de esto es un en el que la penalización incremental disminuye cuadráticamente a medida que crece el tamaño de la brecha. Esto se puede modelar como  $w(k) = p + q * k + r * k^2$ . Sin embargo, la compensación es que también hay costos asociados con el uso de funciones de penalización de brecha más complejas al aumentar sustancialmente el tiempo de ejecución. Este costo puede mitigarse usando aproximaciones más simples a las funciones de penalización por brecha. Es un intermedio fino: se tiene una penalización fija para iniciar una brecha y un costo lineal para agregar a una brecha; esto se puede modelar como  $w(k) = p + q * k$ .

También se pueden considerar funciones más complejas que tomen en consideración las propiedades de las secuencias codificadoras de proteínas. En el caso del alineamiento de regiones codificadoras de proteínas, un hueco de longitud mod 3 puede ser menos penalizado porque no resultaría en un desplazamiento de marco.

---

This page titled [3.3: Alineación global vs. alineación local vs. alineación semi-global](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [3.3: Global alignment vs. Local alignment vs. Semi-global alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

### 3.4: Coincidencia exacta de cadenas en tiempo lineal

Si bien hemos analizado diversas formas de alineación y algoritmos utilizados para encontrar tales alineaciones, estos algoritmos no son lo suficientemente rápidos para algunos fines. Por ejemplo, podemos tener una secuencia de 100 nucleótidos que queremos buscar en todo el genoma, que puede tener más de mil millones de nucleótidos de largo. En este caso, queremos un algoritmo con un tiempo de ejecución que dependa de la longitud de la secuencia de consulta, posiblemente con algún preprocesamiento en la base de datos, ya que procesar todo el genoma para cada consulta sería extremadamente lento. Para tales problemas, entramos en el ámbito de los algoritmos aleatorios donde en lugar de preocuparnos por el peor desempeño, estamos más interesados en asegurarnos de que el algoritmo sea lineal en el caso esperado. Al buscar coincidencias exactas (consecutivas) de una secuencia, el algoritmo de Karp-Rabin interpreta dicha coincidencia numéricamente. Hay muchas otras soluciones a este problema y algunas de ellas que pueden asegurar que el problema sea lineal en el peor de los casos como: el algoritmo Z, el algoritmo Boyer-Moore y Knuth-Morris-Pratt, algoritmos basados en árboles de sufijos, matrices de sufijos, etc. (discutido en las diapositivas de “Adición de la conferencia 3”)

#### Algoritmo Karp-Rabin

Este algoritmo intenta hacer coincidir un patrón particular con una cadena, que es el principio básico de la búsqueda en la base de datos. El problema es el siguiente: en texto T de longitud n estamos buscando patrón P de longitud m. Las cadenas se mapean a números para permitir una comparación rápida. Una versión ingenua del algoritmo implica mapear las subcadenas P y m-length de T números sinto  $x$  e  $y$ , respectivamente, deslizando  $x$  a lo largo de T en cada desplazamiento hasta que haya una coincidencia de los números.

Figura 3.10: Algoritmo de Karp-Rabin ingenuo

Sin embargo, se puede ver que el algoritmo, como se ha dicho, es de hecho no lineal por dos razones:

1. Calcular cada  $y_i$  toma más que un tiempo constante (de hecho es lineal si calculamos ingenuamente cada número desde cero para cada subsecuencia)
2. Comparar  $x$  e  $y_i$  puede ser costoso si los números son muy grandes, lo que podría suceder si el patrón a emparejar es muy largo

Para que el algoritmo sea más rápido, primero modificamos el procedimiento para calcular  $y_i$  en tiempo constante utilizando el número previamente calculado,  $y_{i-1}$ . Podemos hacer esto usando algunas operaciones de bit: una resta para eliminar el bit de orden superior, una multiplicación para desplazar los caracteres a la izquierda, y una suma para agregar el dígito de orden bajo. Por ejemplo, en la Figura 10, podemos calcular  $y_2$  desde  $y_1$  por

- eliminando el bit de orden más alto:  $23590 \bmod 10000 = 3590$
- desplazamiento a la izquierda:  $3590 * 10 = 35900$
- sumando el nuevo dígito de orden bajo:  $35900 + 2 = 35902$

Nuestro siguiente problema surge cuando tenemos secuencias muy largas para comparar. Esto hace que nuestros cálculos sean con números muy grandes, lo que ya no se convierte en tiempo lineal. Para mantener los números pequeños para asegurar una comparación eficiente, hacemos todos nuestros cálculos módulo p (una forma de hash), donde p refleja la longitud de palabra disponible para nosotros para almacenar números, pero es lo suficientemente pequeña como para que la comparación entre  $x$  y  $y_i$  sea realizable en tiempo constante.

: Usar una función para mapear valores de datos a un conjunto de datos de tamaño fijo.

Debido a que estamos usando hash, mapear al espacio de números módulo p puede resultar en golpes espurios debido a colisiones hash, y así modificamos el algoritmo para tratar tales golpes espurios verificando explícitamente los hits reportados de los valores hash. De ahí que la versión final del algoritmo Karp-Rabin sea:

Para calcular el tiempo de ejecución esperado de Karp-Rabin, debemos tener en cuenta el costo esperado de verificación. Si podemos mostrar que la probabilidad de golpes espurios es pequeña, el tiempo de ejecución esperado es lineal.

#### Preguntas:

P: ¿Y si hay más de 10 caracteres en el alfabeto?

**R:** En tal caso, podemos simplemente modificar el algoritmo anterior incluyendo más dígitos, es decir, trabajando en una base que no sea 10, por ejemplo, digamos base 256. Pero en general, cuando se usa hash, las cadenas se mapean en un espacio de números y de ahí las cadenas se interpretan numéricamente.

**P:** ¿Cómo aplicamos esto al texto?

Figura 3.11: Algoritmo final de Karp-Rabin

**R:** Se utiliza una función hash que cambia el texto en números que son más fáciles de comparar. Por ejemplo, si se usa todo el alfabeto, a las letras se les puede asignar un valor entre 0 y 25, y luego usarse de manera similar a una cadena de números.

**P:** ¿Por qué el uso del módulo disminuye el tiempo de cálculo?

**R:** El módulo se puede aplicar a cada parte individual en el cálculo conservando la respuesta. Por ejemplo: imagina que nuestro texto actual es "314152" y la longitud de la palabra es 5. Despues de hacer nuestro primer cálculo en "31415", movemos nuestro marco para hacer nuestro segundo cálculo, que es:

$$\begin{aligned} 14152 &= (31415 - 3 * 10000) * 10 + 2 \pmod{13} \\ &= (7 - 3 * 3) * 10 + 2 \pmod{13} \\ &= 8 \pmod{13} \end{aligned}$$

Este cálculo se puede hacer ahora en tiempo lineal.

**P:** ¿Hay disposiciones en el algoritmo para coincidencias inexactas?

**R:** El algoritmo anterior solo funciona cuando hay regiones de similitud exacta entre la secuencia de consulta y la base de datos. Sin embargo, el algoritmo BLAST, que veremos más adelante, extiende las ideas anteriores para incluir la noción de buscar en un vecindario biológicamente significativo de la secuencia de consultas para dar cuenta de algunas coincidencias inexactas. Esto se hace buscando en la base de datos no solo la secuencia de consultas, sino también algunas variantes de la secuencia hasta algún número fijo de cambios.

En general, para reducir el tiempo de operaciones sobre argumentos como números o cadenas que son realmente largos, es necesario reducir el rango de números a algo más manejable. El hash es una solución general a esto e implica mapear claves  $k$  de un gran universo  $U$  de cadenas o números en un hash de la clave  $h(k)$  que se encuentra en un rango menor, digamos  $[1\dots m]$ . Hay muchas funciones hash que se pueden utilizar, todas con diferentes propiedades teóricas y prácticas. Las dos propiedades clave que necesitamos para el hash son:

1. Reproducibilidad si  $x = y$ , entonces  $h(x) = h(y)$ . Esto es esencial para que nuestro mapeo tenga sentido.
2. Distribución uniforme de salida Esto implica que independientemente de la distribución de entrada, la distribución de salida es uniforme. es decir  $x = y$ , si, entonces  $P(h(x) = h(y)) = 1/m$ , independientemente de la distribución de entrada. Esta es una propiedad deseable para reducir la posibilidad de golpes espurios.

Una idea interesante que se planteó fue que podría ser útil tener funciones hash sensibles a la localidad desde el punto de vista de uso en búsquedas vecinales, de tal manera que los puntos en  $U$  que están cerca entre sí son mapeados a puntos cercanos por la función hash. La noción de proyecciones aleatorias, como extensión del algoritmo BLAST, se basa en esta idea. Además, hay que señalar que el módulo no satisface la propiedad 2 anterior porque es posible tener distribuciones de entrada (por ejemplo, todos los múltiplos del número vis—vis que se toma el módulo) que resultan en muchas colisiones. Sin embargo, elegir un número aleatorio como divisor del módulo puede evitar muchas colisiones.

Trabajar con hash aumenta la complejidad de analizar el algoritmo ya que ahora necesitamos calcular el tiempo de ejecución esperado al incluir el costo de verificación. Para demostrar que el tiempo de ejecución esperado es lineal, necesitamos demostrar que la probabilidad de golpes espurios es pequeña.

---

This page titled [3.4: Coincidencia exacta de cadenas en tiempo lineal](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts

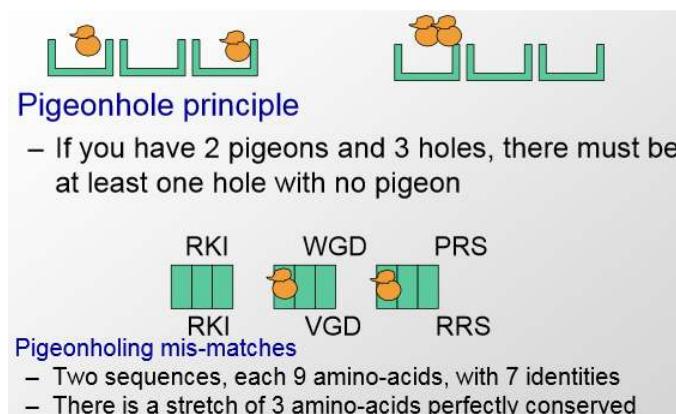
platform.

- **3.4: Linear-time exact string matching** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

### 3.5: El algoritmo BLAST (Herramienta Básica de Búsqueda de Alineación Local)

El algoritmo BLAST analiza el problema de la búsqueda de bases de datos de secuencias, en donde tenemos una consulta, que es una nueva secuencia, y un objetivo, que es un conjunto de muchas secuencias antiguas, y nos interesa saber a cuál (si alguna) de las secuencias diana es la consulta relacionada. Una de las ideas clave de BLAST es que no requiere que las alineaciones individuales sean perfectas; una vez que se identifica una coincidencia inicial, podemos afinar las coincidencias más tarde para encontrar una buena alineación que cumpla con una puntuación umbral. Además, BLAST explota una característica distinta de los problemas de búsqueda de bases de datos: la mayoría de las secuencias diana no estarán completamente relacionadas con la secuencia de consulta y muy pocas secuencias coincidirán.

Sin embargo, las alineaciones correctas (casi perfectas) tendrán largas subcadenas de nucleótidos que coinciden perfectamente. Por ejemplo, si buscamos secuencias de longitud 100 y vamos a rechazar coincidencias que sean menos de 90% idénticas, no necesitamos mirar secuencias que ni siquiera contienen un tramo consecutivo de menos de 10 nucleótidos coincidentes en una fila. Basamos esta suposición en el: si  $m$  artículos se ponen en  $n$  contenedores y  $m > n$ , se deben poner al menos 2 artículos en uno de los  $n$  contenedores.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 3.12: Principio de encasillamiento

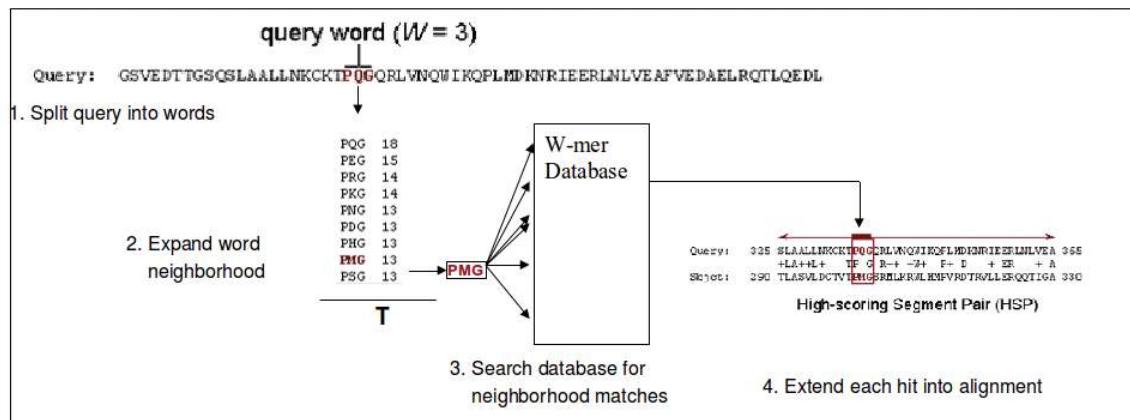
Además, en biología, es más probable que el ADN funcional se conserve, y por lo tanto las mutaciones que encontramos no se distribuirán realmente al azar, sino que se agruparán en regiones no funcionales del ADN dejando intactas largas extensiones de ADN funcional. Por lo tanto, debido al principio del encasillamiento y debido a que secuencias muy similares tendrán tramos de similitud, podemos preseleccionar las secuencias para tramos largos comunes. Esta idea se usa en BLAST dividiendo la secuencia de consulta en W-meros y preseleccionando las secuencias diana para todas las posibles  $W - mers$  limitando nuestras semillas para que estén  $W - mers$  en el vecindario que cumplen un cierto umbral.

El otro aspecto de BLAST que nos permite acelerar consultas repetidas es la capacidad de preprocesar una gran base de datos de ADN fuera de línea. Después del preprocesamiento, la búsqueda de una secuencia de longitud  $m$  en una base de datos de longitud  $n$  tomará solo  $O(m)$  tiempo. Las ideas clave en las que se basa BLAST son las ideas de hash y búsqueda de vecindarios que permiten buscar  $W - mers$ , incluso cuando no hay coincidencias precisas.

#### El algoritmo BLAST

Los pasos son los siguientes:

- Dividir la consulta en palabras superpuestas de longitud  $W$  (los  $W$ -mers)
- Encuentra un “barrio” de palabras similares para cada palabra (ver abajo)
- Busque cada palabra en el vecindario en una tabla hash para encontrar la ubicación en la base de datos donde aparece cada palabra. Llama a estas las *semillas*, y deja que  $S$  sea la colección de semillas.
- Extiende las semillas en  $S$  hasta que la puntuación de la alineación descienda por debajo de algún umbral  $X$ .
- Informar coincidencias con puntuaciones más altas en general



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 3.13: El algoritmo BLAST

El paso de preprocessamiento de BLAST asegura que todas las subcadenas de  $W$  nucleótidos se incluirán en nuestra base de datos (o en una tabla hash). Estos se llaman los *W-meros* de la base de datos. Al igual que en el paso 1, primero dividimos la consulta observando todas las subcadenas de  $W$  nucleótidos consecutivos en la consulta. Para encontrar la vecindad de estos *W-meros*, luego modificamos estas secuencias cambiándolas ligeramente y calculando su similitud con la secuencia original. Generamos progresivamente palabras más disímiles en nuestro vecindario hasta que nuestra medida de similitud desciende por debajo de algún umbral  $T$ . Esto nos brinda flexibilidad para encontrar coincidencias que no tengan exactamente  $W$  caracteres coincidentes consecutivos seguidos, pero que tengan suficientes coincidencias para considerarse similares, es decir, para cumplir con una puntuación umbral de certificación.

Luego, buscamos todas estas palabras en nuestra tabla hash para encontrar semillas de  $W$  nucleótidos coincidentes consecutivos. Luego extendemos estas semillas para encontrar nuestra alineación usando el algoritmo Smith-Waterman para la alineación local, hasta que la puntuación caiga por debajo de cierto umbral  $X$ . Dado que la región que estamos considerando es un segmento mucho más corto, esto no será tan lento como ejecutar el algoritmo en toda la base de datos de ADN.

También es interesante observar la influencia de diversos parámetros de BLAST en el desempeño del algoritmo frente al tiempo de ejecución y la sensibilidad:

- **W** Aunque  $W$  grande resultaría en menos golpes/collisiones espurias, haciéndolo así más rápido, también hay compensaciones asociadas, a saber: un gran vecindario de secuencias de consulta ligeramente diferentes, una tabla hash grande y muy pocos aciertos. Por otro lado, si  $W$  es demasiado pequeño, es posible que obtengamos demasiados aciertos, lo que empuja los costos de tiempo de ejecución al paso de extensión/alineación semilla.
- **T** Si  $T$  es más alto, el algoritmo será más rápido, pero es posible que se pierdan secuencias que son más distantes evolutivamente. Si comparas dos especies relacionadas, probablemente puedas establecer una  $T$  más alta ya que esperas encontrar más coincidencias entre secuencias que sean bastante similares.
- **X** Su influencia es bastante similar a la  $T$  en que ambos controlarán la sensibilidad del algoritmo. Si bien  $W$  y  $T$  afectan el número total de aciertos que uno recibe, y por lo tanto afectan drásticamente el tiempo de ejecución del algoritmo, establecer una  $X$  realmente estricta a pesar de  $W$  y  $T$  menos estrictos, resultará en costos de tiempo de ejecución al probar secuencias innecesarias que no cumplen con la rigurosidad de  $X$ . Por lo tanto, es importante igualar el rigurosidad de  $X$  con la de  $W$  y  $T$  para evitar tiempos de cómputos innecesarios.

## Extensões a BLAST

- **Filtrado** Las regiones de baja complejidad pueden causar impactos espurios. Por ejemplo, si nuestra consulta tiene una cadena de copias del mismo nucleótido, por ejemplo, repeticiones de AC o solo G, y la base de datos tiene un largo tramo del mismo nucleótido, entonces habrá muchos éxitos inútiles. Para evitar esto, podemos intentar filtrar partes de baja complejidad de la consulta o podemos ignorar partes excesivamente representadas de la base de datos irrazonablemente.

- BLAST de dos golpes La idea aquí es usar hash doble en el que en lugar de hash un W -mer largo, vamos a hash dos W-mers pequeños. Esto nos permite encontrar pequeñas regiones de similitud ya que es mucho más probable que tenga dos W-meros más pequeños que coincidan en lugar de un W-mer largo. Esto nos permite obtener una mayor sensibilidad con un W más pequeño, a la vez que seguimos podando golpes espurios. Esto significa que pasaremos menos tiempo tratando de extender partidos que en realidad no coincidan. Así, esto nos permite mejorar la velocidad manteniendo la sensibilidad.

P: Por un W lo suficientemente largo, ¿tendría sentido considerar más de 2 W-mers más pequeños?

R: Sería interesante ver cómo el número de tales W-meros influye en la sensibilidad del algoritmo. Esto es similar a usar un peine, descrito a continuación.

- Peines Esta es la idea de usar W-mers no consecutivos para el hash. Recuerde de sus clases de biología que el tercer nucleótido en un triplete generalmente no tiene realmente un efecto sobre qué aminoácido está representado. Esto significa que cada tercer nucleótido de una secuencia es menos probable que se conserve por evolución, ya que a menudo no importa. Por lo tanto, podríamos querer buscar W-meros que se vean similares excepto en cada tercer codón. Este es un ejemplo particular de un peine. Un peine es simplemente una máscara de bits que representa qué nucleótidos nos importan cuando tratamos de encontrar coincidencias. Nosotros explicamos anteriormente por qué 110110110. (ignorando cada tercer nucleótido) podría ser un buen peine, y resulta serlo. Sin embargo, otros peines también son útiles. Una forma de elegir un peine es simplemente escoger algunos nucleótidos al azar. En lugar de elegir solo un peine para una proyección, es posible elegir aleatoriamente un conjunto de tales peines y proyectar los W-mers a lo largo de cada uno de estos peines para obtener un conjunto de bases de datos de búsqueda. Entonces, la cadena de consulta también se puede proyectar aleatoriamente a lo largo de estos peines para buscar en estas bases de datos, aumentando así la probabilidad de encontrar una coincidencia. Esto se llama Proyección Aleatoria. Ampliando esto, una idea interesante para un proyecto final es pensar en diferentes técnicas de proyección o hashing que tengan sentido biológicamente. Una adición a esta técnica es analizar falsos negativos y falsos positivos, y cambiar el peine para que sea más selectivo. Algunos artículos que exploran adiciones a esta búsqueda incluyen Califino-Rigoutsos'93, Buhler'01 e Indyk-Motwani'98.
- PSI-BLAST Iterativo Específico de Posición BLAST crea perfiles resumidos de proteínas relacionadas usando BLAST. Después de una ronda de BLAST, actualiza la matriz de puntuación de la alineación múltiple, y luego ejecuta rondas posteriores de BLAST, actualizando iterativamente la matriz de puntuación. Construye un Modelo Hidden Markov para rastrear la conservación de aminoácidos específicos. PSI-BLAST permite la detección de proteínas relacionadas a distancia.

---

This page titled [3.5: El algoritmo BLAST \(Herramienta Básica de Búsqueda de Alineación Local\)](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [3.5: The BLAST algorithm \(Basic Local Alignment Search Tool\)](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal

La técnica de hash en el núcleo del algoritmo BLAST es una poderosa forma de cadena para una búsqueda rápida. Se invierte un tiempo sustancial para procesar todo el genoma, o un gran conjunto de genomas, antes de obtener una secuencia de consulta. Una vez obtenida la secuencia de consulta, se puede procesar de manera similar y sus partes se pueden buscar en la base de datos indexada en tiempo lineal.

En esta sección, describimos brevemente cuatro formas adicionales de preprocesar una base de datos para una búsqueda rápida de cadenas, cada una de las cuales tiene importancia práctica y teórica.

### Árboles de sufijos

Los árboles de sufijos proporcionan una poderosa representación de árboles de subcadenas de una secuencia diana T, capturando todos los sufijos de T en un árbol de base.

Representación de una secuencia en un árbol de sufijos

Buscar una nueva secuencia contra un árbol de sufijos

Construcción lineal de árboles de sufijos

### Matrices de sufijos

Para muchas aplicaciones genómicas, los árboles de sufijos son demasiado caros de almacenar en la memoria, y se necesitaron repeticiones más eficientes. Las matrices de sufijos se desarrollaron específicamente para reducir el consumo de memoria de los árboles de sufijos y lograr los mismos objetivos con una necesidad de espacio significativamente reducida.

Usando matrices de sufijos, cualquier subcadena se puede encontrar haciendo una búsqueda binaria en la lista ordenada de sufijos. Al explorar así el prefijo de cada sufijo, terminamos buscando en todas las subcadenas.

### La transformación de Madrows-Wheeler

Una representación aún más eficiente que los árboles de sufijos viene dada por la transformada de Burrows-Wheeler (BWT), que permite almacenar toda la cadena hash en el mismo número de caracteres que la cadena original (e incluso de manera más compacta, ya que contiene series frecuentes de homopolímeros de caracteres que pueden ser más fáciles comprimida). Esto ha ayudado a hacer programas que pueden funcionar de manera aún más eficiente.

Primero consideraremos la matriz BWT, que es una extensión de una matriz de sufijos, ya que contiene no solo todos los sufijos en orden ordenado (lexicográfico), sino que añade a cada sufijo comenzando en la posición i el prefijo que termina en la posición i - 1, conteniendo así cada fila una rotación completa de la cadena original. Esto permite todas las operaciones de matriz de sufijos y árbol de sufijos, de encontrar la posición de los sufijos en el tiempo lineal en la cadena de consulta.

La diferencia clave con las matrices de sufijos es el uso del espacio, donde en lugar de almacenar todos los sufijos en la memoria, lo que incluso para matrices de sufijos es muy costoso, solo se almacena la última columna de la matriz BWT, en base a la cual se puede recuperar la matriz original.

Una matriz auxiliar se puede utilizar para acelerar aún más las cosas y evitar tener que repetir operaciones de encontrar la primera aparición de cada carácter en la matriz de sufijos modificada.

Por último, una vez que se encuentran las posiciones de 100.000s de subcadenas en la cadena modificada (la última columna de la matriz BTW), estas coordenadas se pueden transformar a las posiciones originales, ahorrando tiempo de ejecución al amortizar el costo de la transformación a través de las muchas lecturas.

El BWT ha tenido un impacto muy fuerte en los algoritmos de coincidencia de cadenas cortas, y casi todos los mapeadores de lectura más rápida se basan actualmente en la transformación Burrows-Wheeler.

### Pre-procesamiento fundamental

Esta es una variación del procesamiento que tiene interés teórico pero que ha encontrado relativamente poco uso práctico en bioinformática. Se basa en el vector Z, que contiene en cada posición i la longitud del prefijo más largo de una cadena que también

coincide con la subcadena comenzando en  $i$ . Esto permite calcular los vectores L y R (Izquierda y Derecha) que denotan el final de las subcadenas duplicadas más largas que contienen la posición actual  $i$ .

## Coincidencia de cuerdas educadas

El algoritmo Z permite un cálculo sencillo de los algoritmos Boyer-Moore y Knuth-Morris-Pratt para la coincidencia de cadenas en tiempo lineal. Estos algoritmos utilizan la información recopilada en cada comparación al hacer coincidir cadenas para mejorar la coincidencia de cadenas con  $O(n)$ . El algoritmo ingenuo es el siguiente: compara su cadena de longitud  $m$  carácter por carácter con la secuencia. Después de comparar toda la cadena, si hay algún desajuste, se mueve al siguiente índice y vuelve a intentarlo. Esto se completa en  $O(m * n)$  el tiempo.

Una mejora de este algoritmo es descontinuar la comparación actual si se encuentra un desajuste. Sin embargo, esto todavía se completa en el  $O(m * n)$  tiempo cuando la cadena que estamos comparando coincide con la secuencia completa.

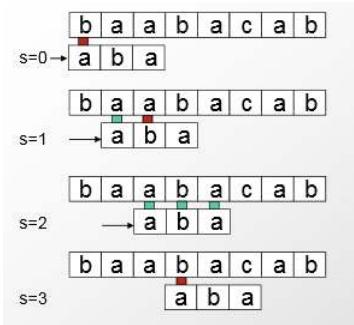


Figura 3.14: Coincidencia de cuerdas educada

La visión clave proviene de aprender de la redundancia interna en la cadena para comparar, y usarlo para hacer cambios más grandes hacia abajo en la secuencia objetivo. Cuando se comete un error, todas las bases en la comparación actual se pueden utilizar para mover el fotograma considerado para la siguiente comparación más abajo. Como se ve a continuación, esto reduce en gran medida el número de comparaciones requeridas, disminuyendo el tiempo de ejecución a  $O(n)$ .

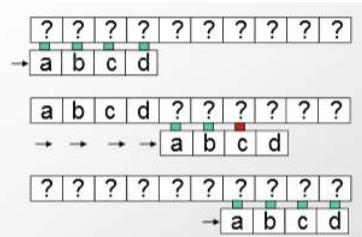


Figura 3.15: Coincidencia final de cadenas

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

---

This page titled [3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [3.6: Pre-processing for linear-time string matching](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

### 3.7: Fundamentos probabilísticos del alineamiento de secuencias

Como se describió anteriormente, el algoritmo BLAST utiliza una matriz de puntuación (sustitución) para expandir la lista de *W-meros* con el fin de buscar y determinar una secuencia aproximadamente coincidente durante la extensión de la semilla. Además, se utiliza una matriz de puntuación en la evaluación de coincidencias o desajustes en los algoritmos de alineación. Pero, ¿cómo construimos esta matriz en primer lugar? ¿Cómo se determina el valor de  $\delta(x_i, y_j)$  en alineación global/local?

La idea detrás de la matriz de puntuación es que la puntuación de alineación debe reflejar la probabilidad de que dos secuencias similares sean homólogas, es decir, la probabilidad de que dos secuencias que tienen un montón de nucleótidos en común también comparten una ascendencia común. Para ello, nos fijamos en los cocientes de verosimilitud entre dos hipótesis.

1. **Hipótesis 1:** — Que el alineamiento entre las dos secuencias se debe al azar y las secuencias son, de hecho, no relacionadas.
  2. **Hipótesis 2:** — Que el alineamiento se debe a la ascendencia común y las secuencias están realmente relacionadas.

Luego, calculamos la probabilidad de observar una alineación de acuerdo a cada hipótesis.  $Pr(x, y|u)$  es la probabilidad de alinear  $x$  con  $y$  asumiendo que no están relacionados, mientras que  $Pr(x, y|R)$  es la probabilidad de la

	A	G	T	C
A	+1	-½	-1	-1
G	-½	+1	-1	-1
T	-1	-1	+1	-½
C	-1	-1	-½	+1

Figura 3.16: Puntuaciones de coincidencia de nucleótidos

alineación, suponiendo que estén relacionados. Luego, definimos la puntuación de alineación como el log de la relación de verosimilitud entre los dos:

```
\begin{equation}
S \equiv \log \frac{P(\mathbf{x},\mathbf{y} \mid R)}{P(\mathbf{x},\mathbf{y} \mid U)}
\end{equation}\nonumber
```

Dado que una suma de registros es un registro de productos, podemos obtener la puntuación total de la alineación sumando las puntuaciones de las alineaciones individuales. Esto nos da la probabilidad de toda la alineación, asumiendo que cada alineación individual es independiente. Así, una puntuación de matriz aditiva nos da exactamente la probabilidad de que las dos secuencias estén relacionadas, y el alineamiento no se debe al azar. De manera más formal, considerando el caso de alinear proteínas, para secuencias no relacionadas, la probabilidad de tener un alineamiento de  $n$  residuos entre  $x$  e  $y$  es un producto simple de las probabilidades de las secuencias individuales ya que los emparejamientos de residuos son independientes.

Es decir,

```

\begin{comenzar}{ecuación}
\comenzar{alineado}
\mathbf{x} &= \text{izquierda} \{x_1\} \lpuntos x_n \derecha \\
\mathbf{y} &= \text{izquierda} \{y_1\} \lpuntos x_n \derecha \\
q_a &= P(\text{texto aminoácido} a) \\
P(x, y | \text{media } U) &= \prod_{i=1}^n q_{x_i} \prod_{i=1}^n q_{y_i} \\
\end{alineado}
\end{ecuación}\nonumber

```

Para secuencias relacionadas, los emparejamientos de residuos ya no son independientes por lo que debemos usar una articulación diferente

probabilidad, suponiendo que cada par de aminoácidos alineados evolucionó a partir de un ancestro común:

```
\begin{equation}
\begin{aligned}
p_{(a,b)} &= P(\text{evolución dio lugar a } a | \text{text{ in}} \mathbf{x} | \text{text{ y}} b | \text{text{ in}} \mathbf{y}) \\
P(\mathbf{x}, \mathbf{y} | R) &= \prod_{i=1}^n p_{x_i y_i}
\end{aligned}
\end{equation}
```

Entonces, la razón de verosimilitud entre los dos viene dada por:

```
\begin{equation}
\begin{aligned}
&\frac{P(\mathbf{x}, \mathbf{y} | R)}{P(\mathbf{x}, \mathbf{y} | U)} = \frac{\prod_{i=1}^n p_{x_i y_i}}{\prod_{i=1}^n q_{x_i} q_{y_i}}
&= \frac{\prod_{i=1}^n p_{x_i y_i}}{\prod_{i=1}^n q_{x_i} q_{y_i}}
\end{aligned}
\end{equation}
```

Como finalmente queremos calcular una suma de puntajes y probabilidades requieren agregar productos, tomamos el registro del producto para obtener una suma útil:

```
\begin{equation}
\begin{aligned}
S &\equiv \log \frac{P(\mathbf{x}, \mathbf{y} | R)}{P(\mathbf{x}, \mathbf{y} | U)} \\
&\equiv \sum_i \log \left( \frac{p_{x_i y_i}}{q_{x_i} q_{y_i}} \right) \text{derecha} \\
&\equiv \sum_i s_{izquierda(x_i, y_i)} \text{derecha}
\end{aligned}
\end{equation}
```

Así, la puntuación de la matriz de sustitución para un par dado a, b es dada por

```
\begin{equation}
s(a, b) = \log \left( \frac{p_{a b}}{q_a q_b} \right) \text{derecha}
\end{equation}
```

La expresión anterior se usa entonces para producir una matriz de sustitución como el BLOSUM62 para aminoácidos. Es interesante señalar que la puntuación de una coincidencia de un aminoácido consigo mismo depende del aminoácido en sí mismo porque la frecuencia de ocurrencia aleatoria de un aminoácido afecta los términos utilizados en el cálculo de la puntuación de la relación de verosimilitud de alineación. De ahí que estas matrices capturen no sólo la similitud de secuencia de los alineamientos, sino también la similitud química de diversos aminoácidos.

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																			C	
S	-1	4																		S	
T	-1	1	5																	T	
P	-3	-1	-1	7																P	
A	0	1	0	-1	4															A	
G	-3	0	-2	-2	0	6														G	
N	-3	1	0	-2	-2	0	6													N	
D	-3	0	-1	-1	-2	-1	1	6												D	
E	-4	0	-1	-1	-1	-2	0	2	5											E	
Q	-3	0	-1	-1	-1	-2	0	0	2	5										Q	
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8									H	
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5								R	
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5							K	
M	-1	-1	-1	-2	-1	-3	-2	-3	0	-2	-1	-1	5							M	
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	1	4					I		
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	2	2	4				L		
V	-1	-2	0	-2	0	-3	-3	-2	-2	-3	-3	-2	1	3	1	4			V		
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6		F		
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	2	-2	-2	-1	-1	-1	-1	3	7	<th>Y</th>	Y	
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	W
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 3.17: Matriz BLOSUM62 para aminoácidos

Lectura adicional:

Algoritmos relacionados con BLAST: Califino-Rigoutsos'93, Buhler'01 e Indyk-Motwani'98

This page titled [3.7: Fundamentos probabilísticos del alineamiento de secuencias](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [3.7: Probabilistic Foundations of Sequence Alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 4: Genómica Comparada I- Anotación del Genoma

En este capítulo exploraremos el campo emergente de la genómica comparativa, principalmente a través de ejemplos de alineaciones genómicas de múltiples especies (trabajo realizado por el laboratorio de Kellis). Un enfoque para el análisis de genomas es inferir funciones génicas importantes a través de la aplicación de una comprensión de la evolución para buscar patrones evolutivos esperados. Otro enfoque es descubrir las tendencias evolutivas mediante el estudio de los propios genomas. En conjunto, el conocimiento evolutivo y los grandes conjuntos de datos genómicos ofrecen un gran potencial para el descubrimiento de nuevos fenómenos biológicos.

- [4.1: Introducción](#)
- [4.2: Conservación de secuencias genómicas](#)
- [4.3: Restricción en exceso](#)
- [4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección](#)
- [4.5: Firmas de codificación de proteínas](#)
- [4.6: Firmas génicas de microARN \(miARN\)](#)
- [4.7: Motivos Regulatorios](#)
- [4.8: Lectura adicional](#)
- [4.9: Herramientas y Técnicas](#)
- [Bibliografía](#)

---

This page titled [4: Genómica Comparada I- Anotación del Genoma](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 4.1: Introducción

Un tema recurrente de este trabajo es adoptar un enfoque computacional global para analizar elementos de genes y ARN codificados en el genoma y utilizarlo para encontrar nuevos fenómenos biológicos interesantes. Podemos hacer esto viendo cómo los ejemplos individuales “divergen” o difieren del caso promedio. Por ejemplo, al examinar muchos genes que codifican proteínas, podemos identificar características representativas de esa clase de loci. Luego podemos elaborar pruebas de alta precisión para distinguir genes que codifican proteínas de genes que no codifican proteínas. A menudo, estas pruebas computacionales, basadas en miles de ejemplos, serán mucho más definitivas que las pruebas convencionales de laboratorio húmedo de bajo rendimiento. (Tales pruebas pueden incluir espectrometría de masas para detectar productos proteicos, en los casos en los que queremos saber si un locus en particular es codificador de proteínas).

### Motivación y Desafío

A medida que el costo de la secuenciación genómica continúa bajando, la disponibilidad de datos genómicos secuenciados se ha disparado. Sin embargo, el análisis de los datos no se ha mantenido al día, mientras que hay muchos fenómenos biológicos interesantes que no se han descubierto en las interminables cadenas de ATGCs. El objetivo de la genómica comparativa es aprovechar la gran cantidad de información disponible para buscar patrones biológicos.

Como su nombre indica, la genómica comparada no se enfoca en un conjunto específico de genomas. El problema de centrarse puramente en el nivel del genoma único es que se pierden las firmas evolutivas clave. La genómica comparativa resuelve este problema comparando genomas de muchas especies que evolucionaron a partir de un ancestro común. A medida que la evolución cambia el genoma de una especie, deja huellas de su presencia. Veremos más adelante en este capítulo que la evolución discrimina entre porciones de un genoma sobre la base de la función biológica. Al explotar esta correlación entre las huellas dactilares evolutivas y el papel biológico de una subsecuencia genómica, la genómica comparativa es capaz de dirigir la investigación de laboratorio húmedo a partes interesantes del genoma y descubrir nuevos fenómenos biológicos.

#### FAQ

P: ¿Por qué las mutaciones solo se acumulan en ciertas regiones del genoma, mientras que otras regiones se conservan?

R: En regiones no funcionales del ADN, las mutaciones acumuladas se mantienen porque no perturban la función del ADN. En las regiones funcionales, estas mutaciones pueden conducir a una disminución de la aptitud; estas mutaciones luego se descartan de la especie por selección natural.

Podemos obtener mucha información sobre la evolución a través del estudio de la genómica y, de manera similar, podemos aprender sobre el genoma a través del estudio de la evolución. Por ejemplo, a partir del principio de “supervivencia del más apto”, podemos comparar especies relacionadas para descubrir qué partes del genoma son elementos funcionales. El proceso evolutivo introduce mutaciones en cualquier genoma. En regiones no funcionales del ADN, las mutaciones acumuladas se mantienen porque no perturban la función del ADN. Sin embargo, en regiones funcionales, las mutaciones acumuladas a menudo conducen a una disminución de la aptitud. Por lo tanto, no es probable que estas mutaciones decrecientes del estado físico se perpetúen a las generaciones futuras. A medida que avanza el tiempo, es probable que los organismos evolutivamente inadecuados no sobrevivan y sus genes se diluyan. Al comparar los genomas de las especies supervivientes con los genomas de sus antepasados, podemos ver qué porciones constituyen elementos funcionales y cuáles constituyen “ADN basura”.

A la fecha se han descubierto diversos marcadores y fenómenos biológicos importantes a través de métodos de genómica comparativa. Por ejemplo, CRISPRs (Clustered Regularly Interspaced Short Palindromic Repeats), que se encuentran en bacterias y arqueas, se descubrieron por primera vez a través de la genómica comparativa. Los experimentos de seguimiento revelaron que proporcionan inmunidad adaptativa a plásmidos y fagos. Otro ejemplo, que veremos más adelante en este capítulo, es el fenómeno de la lectura de codones de parada, donde ocasionalmente se ignoran los codones de parada durante el proceso de la fase de traducción de la biosíntesis de proteínas. Sin una genómica comparada que los guíe, los experimentalistas podrían haber ignorado ambas características durante muchos años.

Sin un sistema para interpretar e identificar características importantes en los genomas, todas las secuencias de ADN en la tierra son solo un mar de datos sin sentido. Sin embargo, no podemos ignorar la importancia tanto de la informática como de la biología.

en la genómica comparada. Sin conocimiento de biología, uno podría pasar por alto las firmas de sustituciones sinónimos o mutaciones de cambio de marco. Por otro lado, ignorar los enfoques computacionales conduciría a la incapacidad de analizar conjuntos de datos cada vez más grandes que emergen de los centros de secuenciación. La genómica comparada requiere habilidades y conocimientos multidisciplinarios poco frecuentes.

Este es un momento particularmente emocionante para ingresar al campo de la genómica comparada, porque el campo es lo suficientemente maduro como para que existan herramientas y datos disponibles para hacer descubrimientos. Pero es lo suficientemente joven como para que probablemente se sigan haciendo hallazgos importantes durante muchos años.

### Importancia de muchos genomas estrechamente relacionados

Para resolver características biológicas significativas necesitamos tanto similitud suficiente para permitir la comparación como suficiente divergencia para identificar firmas de cambio a lo largo del tiempo evolutivo. Esto es difícil de lograr en una comparación por pares. Mejoramos la resolución de nuestro análisis extendiendo el análisis a muchos genomas simultáneamente con algunos grupos de organismos similares y algunos organismos diferentes. Una simple analogía es la de observar una orquesta. Si colocas un solo micrófono, será difícil descifrar la señal proveniente de todo el sistema, ya que se verá abrumada por el ruido local desde el único punto de observación, el instrumento más cercano. Si colocas muchos micrófonos distribuidos por la orquesta a distancias razonables, entonces obtienes una perspectiva mucho mejor no solo de la señal general, sino también de la estructura del ruido local. Del mismo modo, al secuenciar muchos genomas a través del árbol de la vida, podemos distinguir las señales biológicas de los elementos funcionales del ruido de las mutaciones neutras. Esto se debe a que la naturaleza selecciona para la conservación de elementos funcionales a través de grandes distancias filogenéticas mientras constantemente introduce ruido a través de procesos mutagénicos que operan en escalas de tiempo más cortas

En este capítulo, asumiremos que ya tenemos una alineación genómica completa de múltiples especies estrechamente relacionadas, abarcando regiones codificantes y no codificantes. En la práctica, construir ensamblajes genómicos completos y alineaciones de genoma completo es un problema muy desafiante; ese será el tema del próximo capítulo.

#### FAQ

P: ¿Por qué hay más poder de resolución cuando aumenta la distancia evolutiva o la longitud de las ramas entre especies?

R: Si estamos comparando dos especies como el humano y el chimpancé que están muy cerca entre sí, esperamos ver pocas o ninguna mutación. Esto nos da poco poder discriminativo porque no vemos diferencia entre el número de mutaciones en los elementos funcionales frente al número de mutaciones en elementos no funcionales. Sin embargo, a medida que aumentamos el tiempo evolutivo entre especies, esperamos ver más mutaciones, pero lo que en realidad vemos es una disminución notable en el número observado de mutaciones en ciertas regiones del genoma. Podemos concluir que estas regiones son regiones funcionales. Por lo tanto, nuestra confianza en los elementos funcionales percibidos aumenta a medida que aumenta la longitud de

#### FAQ

P: ¿Por qué es mejor tener muchas especies estrechamente relacionadas para la misma longitud de rama en lugar de una especie lejanamente relacionada?

R: A medida que aumenta la longitud de las ramas entre especies distantes relacionadas, incluso los elementos funcionales no se conservan. Además, alinear de manera confiable genes de parientes lejanos de la misma especie es difícil, si no imposible, usando la tecnología actual como BLAST.

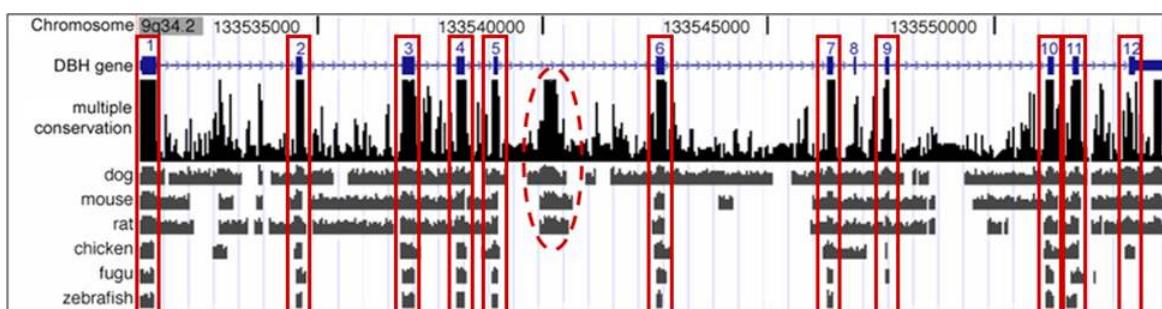
### Genómica comparada y firmas evolutivas

Dado un alineamiento de todo el genoma, posteriormente podemos analizar el nivel de conservación de los elementos funcionales en cada uno de los genomas considerados. Usando el navegador del genoma UCSC, uno puede ver un nivel de conservación para cada gen en el genoma humano derivado de alinear los genomas de muchas otras especies. En la Figura 4.1 a continuación, vemos una secuencia de ADN representada en el eje x, mientras que cada “fila” representa una especie diferente. El eje y dentro de cada

fila representa la cantidad de conservación para esa especie en esa parte del cromosoma (aunque también se utilizaron otras especies que no se muestran para calcular la conservación). Barras superiores corresponden con mayor conservación.

A partir de esta cifra, podemos ver que hay bloques de conservación separados por regiones que no están conservadas. Los 12 exones (resaltados por rectángulos rojos) se conservan en su mayoría entre especies, pero a veces faltan ciertos exones; por ejemplo, al pez cebra le falta el exón 9. Sin embargo, también vemos que hay un pico en algunas especies (como un círculo en rojo) que no corresponde a un gen codificante de proteínas conocido. Esto nos dice que algunas regiones intrónicas también se han conservado evolutivamente, ya que las regiones de ADN que no codifican proteínas aún pueden ser importantes como elementos funcionales, como ARN, microARN y motivos reguladores. Al observar cómo se conservan las regiones, en lugar de solo mirar la cantidad de conservación, podemos observar 'firmas evolutivas' de conservación para diferentes elementos funcionales.

El patrón de mutación/inserción/deleción puede ayudarnos a distinguir diferentes tipos de elementos funcionales en el genoma. Diferentes elementos funcionales están bajo diferentes presiones selectivas y al considerar a qué presiones selectivas se encuentra cada elemento, podemos desarrollar firmas evolutivas características de cada función. Por ejemplo, vemos la diferencia en las firmas evolutivas tal como muestran los genes codificantes de proteínas frente a los motivos reguladores... etc.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.1: Los exones (en caja en rojo) están profundamente conservados de mamíferos a peces. Otros elementos también están fuertemente conservados, como el pico circular cerca del centro de la gráfica. Este puede ser un elemento regulador que se encuentra en mamíferos pero no en aves o peces.

## FAQ

P: Dada una alineación de genes de múltiples especies, ¿qué se puede medir para determinar el nivel de conservación de uno o varios genes específicos?

R: Un método simple es solo observar la puntuación de alineación para cada gen. Si se quiere distinguir entre segmentos codificantes de proteínas altamente conservados de segmentos que no codifican proteínas, también se puede observar la conservación de codones. Sin embargo, en ambos enfoques, tenemos que considerar la posición de cada especie comparada en el árbol filogenético. Una puntuación de comparación por pares que sea menor entre dos especies separadas por una mayor distancia en el árbol filogenético que la puntuación por pares entre dos especies estrechamente relacionadas no implicaría necesariamente una menor conservación.

This page titled [4.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [4.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.2: Conservación de secuencias genómicas

### Elementos funcionales en Drosophila

En un artículo<sup>1</sup> de 2007, Stark et al. identificaron firmas evolutivas de diferentes elementos funcionales y función predicha usando firmas conservadas. Un hallazgo importante es que a lo largo del tiempo evolutivo, los genes tienden a permanecer en una ubicación similar. Esto se ilustra en la Figura 4.2, que muestra el resultado de un alineamiento múltiple en segmentos ortólogos de genomas de doce especies de Drosophila. Cada genoma está representado por una línea azul horizontal, donde la línea superior representa la secuencia de referencia. Las líneas grises conectan elementos funcionales ortólogos, y es evidente que sus posiciones generalmente se conservan en las diferentes especies.

#### FAQ

P: ¿Por qué es significativo que se conserve la posición de los elementos ortólogos?

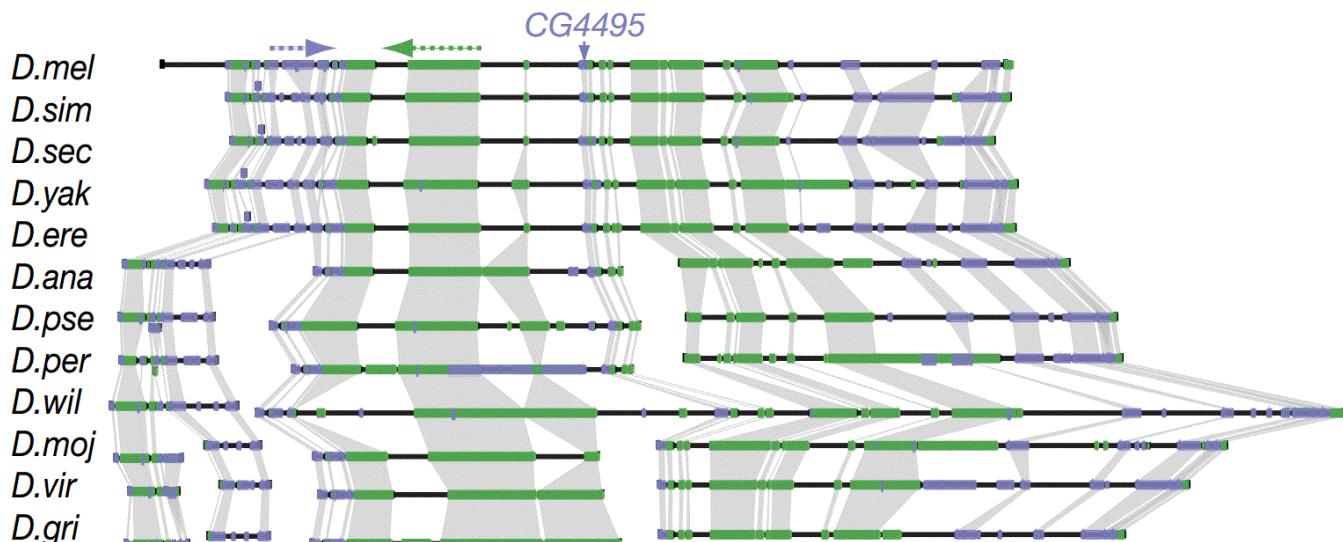
R: El hecho de que se conserven las posiciones es lo que nos permite hacer comparaciones entre especies. De lo contrario, no seríamos capaces de alinear regiones no codificantes de manera confiable.

Drosophila es una gran especie para estudiar porque, de hecho, la separación de las moscas de la fruta es mayor que la de los mamíferos. Esto nos lleva a una interesante nota al margen, la de qué especies seleccionar al mirar las firmas de conservación. No se quiere tener especies muy similares (como los humanos y los chimpancés, que comparten 98% del genoma), porque sería difícil distinguir regiones que son distintas de las que son iguales. Al comparar especies con humanos, el nivel adecuado de conservación a tener en cuenta son los mamíferos. Específicamente, la mayoría de las investigaciones realizadas en este campo se realizan utilizando 29 mamíferos euterianos (mamíferos placentarios, sin marsupiales ni monotremos) para estudiar. Otra de las cosas a tener en cuenta son las diferencias de longitud de ramificación entre dos especies. Sus sujetos de estudio ideales serían algunas especies estrechamente relacionadas (ramificación corta), para evitar problemas de interpretación que surgen con mutaciones largas de longitud de ramificación, como retromutaciones.

### Tarifas y patrones de selección

Ahora que hemos establecido que hay estructura para la evolución de las secuencias genómicas, podemos comenzar a analizar características específicas de la conservación. Para esta sección, consideraremos datos genómicos a nivel de nucleótidos individuales. Más adelante en este capítulo veremos que también podemos analizar secuencias de aminoácidos.

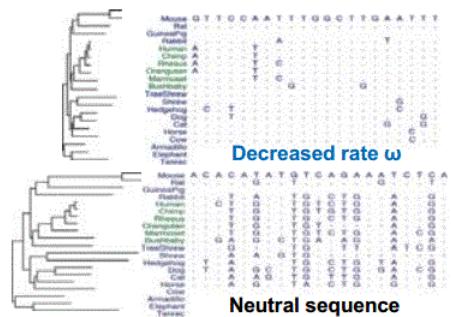
Podemos estimar la intensidad de una restricción de selección  $\omega$  haciendo un modelo de probabilidades de la tasa de sustitución inferida a partir de los datos de alineación genómica. El uso de una estimación de probabilidad máxima (ML) de  $\omega$  puede proporcionarnos la tasa de selección  $\omega$  así como la puntuación de probabilidades logarítmicas de que la tasa no es natural.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.2: Identificación comparativa de elementos funcionales en 12 especies de *Drosophila*. Las líneas grises indican la alineación de regiones ortólogas. El color indica la dirección de la transcripción.

Una propiedad que esto mide que podemos considerar es la tasa de sustitución de nucleótidos en un genoma. La Figura 4.3 muestra dos secuencias de nucleótidos de una colección de mamíferos. Una de las secuencias está sujeta a tasas normales de cambio, mientras que la otra demuestra una tasa reducida. De ahí que podamos plantear la hipótesis de que esta última secuencia está sujeta a un mayor nivel de restricción evolutiva, y puede representar una sección más importante biológicamente del genoma.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.3: Comparación entre dos regiones genómicas con diferentes tasas de selección  $\omega$ . La secuencia de la izquierda demuestra tasas normales de mutación, mientras que la secuencia de la derecha muestra un alto nivel de conservación, como lo demuestra el reducido número de mutaciones.

Podemos detectar patrones inusuales de selección  $\pi$  observando un modelo probabilístico de una distribución estacionaria que es diferente de la distribución de fondo. La estimación ML de  $\pi$  nos proporciona la matriz de peso de probabilidad (PWM) para cada k-mer en el genoma, así como el log odds score para sustituciones que son inusuales (por ejemplo, una base que cambia a una y solo otra base). Como se puede ver en la Figura 4.4, las letras específicas importan porque algunas bases cambian selectivamente a una (o dos otras bases), y la base específica a la que cambia puede sugerir cuál puede ser la función de la secuencia.

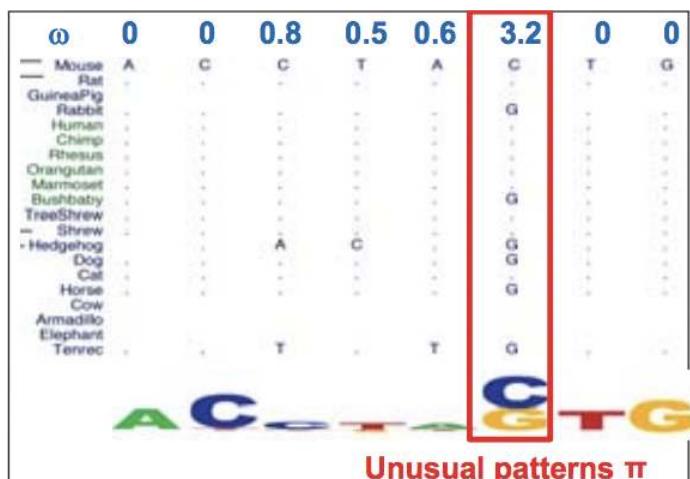
Podemos aumentar nuestro poder de detección de elementos de restricción observando más especies, como se muestra en la Figura 4.5 donde vemos un aumento dramático en la potencia para detectar pequeños elementos restringidos.

<sup>1</sup> [www.nature.com/nature/journal...ture06340.html](http://www.nature.com/nature/journal...ture06340.html)

- **4.2: Conservation of genomic sequences** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.3: Restricción en exceso

En la mayoría de las regiones del genoma donde vemos conservación entre especies, esperamos que haya al menos alguna cantidad de sustitución sinónima. Se trata de sustituciones de nucleótidos “silenciosas” que modifican un codón de tal manera que el aminoácido que codifica no se modifica. En un artículo<sup>2</sup> de 2011, Lindblad-Toh et al. estudiaron la restricción evolutiva en el genoma humano haciendo análisis comparativo de 29 especies de mamíferos. Encontraron que entre los 29 genomas, el sitio nucleotídico promedio mostró 4.5 sustituciones por sitio.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.4: Esta secuencia muestra una tasa de sustitución inusual de sustituir C por G y viceversa.

Dada una tasa de sustitución promedio tan alta, no esperamos ver una conservación perfecta en todas las regiones que se conservan. Por ejemplo, ignorando todos los demás efectos, la probabilidad de que un 12-mer permanezca fijo en todas las 29 especies es menor que  $10^{-25}$ . Así, las regiones que están casi perfectamente conservadas en múltiples especies se destacan por ser únicas y dignas de estudio adicional. Una de esas regiones se muestra en la Figura 4.6.

### Causas del exceso de restricción

La pregunta es ¿qué presiones evolutivas hacen que ciertas regiones estén tan perfectamente conservadas? Las siguientes fueron todas mencionadas en clase como posibilidades:

- ¿Podría ser que exista una estructura especial de ADN que proteja esta zona de la mutación?
- ¿Hay alguna maquinaria especial para corregir errores que se asiente en este lugar?
- ¿Puede la célula utilizar el estado de metilación de las dos copias de ADN como mecanismo de corrección de errores? Este mecanismo se basaría en el hecho de que la nueva copia de ADN no está metilada, y por lo tanto la maquinaria de replicación del ADN podría verificar la nueva copia contra la copia vieja metilada.
- ¿Quizás la próxima generación no pueda sobrevivir si esta región está mutada?

Otra posible explicación es que se está produciendo selección para conservar codones específicos. Algunos codones son más eficientes que otros: por ejemplo, las proteínas de mayor abundancia que necesitan una traducción rápida pueden seleccionar codones que dan la tasa de traducción más eficiente, mientras que otras proteínas pueden seleccionar codones que dan una traducción menos eficiente.

Aún así, estas regiones parecen estar demasiado perfectamente conservadas para ser explicadas solo por el uso de codones. ¿Qué más puede explicar el exceso de restricción? Debe haber algún grado de precisión necesario a nivel de nucleótidos que impida que estas secuencias diverjan.

Podría ser que estemos viendo la misma región en dos especies que apenas han divergido recientemente o que existe un mecanismo genético específico que proteja esta zona. Sin embargo, es más probable que tanta conservación sea un signo de regiones codificantes de proteínas que codifican simultáneamente otros elementos funcionales. Por ejemplo, el gen HOXB5 muestra un

obvio exceso de restricción, y hay evidencia de que el extremo 5' del ORF de HOXB5 codifica tanto proteína como una estructura secundaria de ARN.

	$\pi$ log-odds (12mers)	$\pi$ log-odds (50mers)	$\omega$ (12mers)	$\omega$ (50mers)
<b>29 mammals</b>	<b>7.1/1.5/4.6</b>	<b>6.8/1.8/4.1</b>	<b>5.7/ 1.1/3.8</b>	<b>5.7/1.8/3.0</b>
<b>(HMRD) Human Mouse Rat Dog</b>	<b>4.2/0.0/0.0</b>	<b>5.3/0.1/0.3</b>	<b>4.5/0.0/0.0</b>	<b>5.1/0.6/1.7</b>

### Estimated / kmers detectable at 5% FDR / base pairs detectable at 5% FDR

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.5: Al aumentar el número de mamíferos estudiados, vemos un aumento en los k-meros restringidos y pares de bases que son detectables.

Las regiones que codifican más de un tipo de elemento funcional están bajo presiones selectivas superpuestas. Podría haber presión en el espacio de codificación de proteínas para mantener la secuencia de aminoácidos correspondiente a esta región igual, combinada con la presión del espacio de ARN para mantener una secuencia de nucleótidos que preserva la estructura secundaria del ARN. Como resultado de estas dos presiones para mantener codones para los mismos aminoácidos y producir la misma estructura de ARN, es probable que la región muestre mucha menos tolerancia para cualquier patrón de sustitución sinónimo.

El proceso de estimación de restricción evolutiva a partir de datos de alineamiento genómico en múltiples especies sigue los siguientes pasos:

- Contar el número de operaciones de edición (es decir, el número de sustituciones y/o eliminaciones/inserciones)
- Estimar el número de mutaciones incluyendo retromutaciones
- Incorporar información sobre los elementos vecinales del elemento conservado mirando “ventanas de conservación”
- Estimar la probabilidad de un “estado oculto” restringido mediante el uso de Modelos Ocultos de Markov
- Utilice la filogenia para estimar la tasa de mutación del árbol (es decir, rechazar sustituciones que deberían ocurrir a lo largo del árbol)
- Permitir que diferentes porciones del árbol tengan diferentes tasas de mutación

### Modelado de restricción de exceso

Para estudiar mejor la región de restricción excesiva, desarrollamos modelos matemáticos para medir sistemáticamente la cantidad de conservación sinónica y no sinónica de diferentes regiones. Mediremos dos tasas: conservación de codones y nucleótidos.

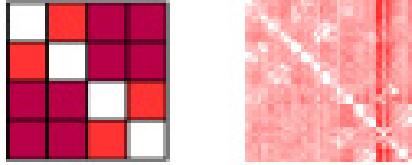
Para representar el modelo nulo, podemos construir matrices de velocidad ( $4 \times 4$  en el caso de nucleótidos y  $64 \times 64$  para el caso del codón) que dan las tasas de sustituciones entre codones o nucleótidos por unidad de tiempo. Estimamos las tasas en el modelo nulo observando una tonelada de datos y estimando las probabilidades de cada tipo de sustitución. Véase la Figura 4.18a en 4.5.2 para un ejemplo de una matriz nula para el caso del codón.

- $\lambda_s$ : la tasa de sustituciones sinónimas

anc_aa	M	B	S	F	P	V	N	S	G	R	Y	P	N	G	O	D	Y	Q	L	N	T	G	S	S	S	L	S	G	S	S	N	R	D	P	A	M	T	G	S	T	G	Y	N	N
Human	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Chimp	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Rhesus	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Mouse	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Lemur	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Bushbaby	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Tree shrew	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
House	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Ray	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Manatee	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Guinea_pig	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Squirrel	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Rabbit	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Platypus	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Dolphin	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Cow	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Horse	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Car	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Dog	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Porc	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Meerkat	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Megabat	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Hedgehog	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Elephant	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Rock	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Terrier	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Armadillo	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A
Sloth	A	T	G	C	T	C	A	T	G	C	T	T	G	G	C	G	C	G	A	C	T	T	A	G	G	A	T	C	A	C	G	C	T	C	T	G	G	T	A	A	T	T	A	A

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.6: Muchas regiones genómicas, como HOXB5, muestran más conservación de la que esperaríamos en regiones codificantes conservadas normales. Entre las 29 especies en estudio, todas menos 7 de ellas tenían exactamente la misma secuencia de nucleótidos. Las áreas verdes son áreas que han sufrido mutaciones evolutivas.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.7: Podemos modelar mutaciones usando matrices de velocidad, como se muestra aquí para las sustituciones de nucleótidos a la izquierda y las sustituciones de codones a la derecha. En cada matriz, la celda en la fila m y la enésima columna representa la probabilidad de que el símbolo m mute en el símbolo n. Cuanto más oscuro es el color, menos probable es la mutación.

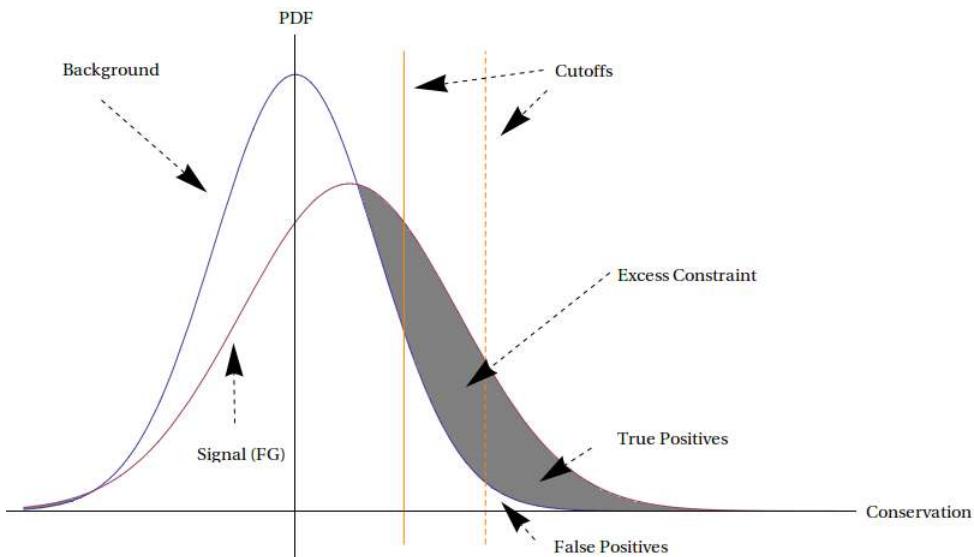
Por ejemplo, si  $\lambda_s = 0.5$ , entonces la tasa de sustituciones sinónimas es la mitad de lo que se espera del modelo nulo en esa región. Luego podemos evaluar la significancia estadística de las estimaciones de tasa que obtenemos, y encontrar regiones donde la tasa de sustitución es mucho menor de lo esperado.

El uso de un modelo nulo aquí nos ayuda a dar cuenta de los sesgos en la cobertura de alineación de ciertos codones y también da cuenta de la posibilidad de degeneración de codones, en cuyo caso esperaríamos ver una tasa mucho mayor de sustituciones. Aprenderemos a combinar dichos modelos con métodos filogénicos cuando hablaremos de árboles filogénicos y evolución más adelante en el curso.

La aplicación de este modelo muestra que las secuencias en los primeros codones traducidos, exones de casete (exones que están presentes en un transcripto de ARNm pero ausentes en una isoforma del transcripto) y, alternativamente, regiones empalmadas tienen tasas especialmente bajas de sustituciones sinónimas.

## Exceso de restricción en el genoma humano

En esta sección, examinaremos el problema de determinar la proporción total del genoma humano bajo restricción excesiva. En particular, revisitaremos el trabajo de Lindblad-Toh et al. (2011), que compararon 29 genomas de mamíferos. Midieron los niveles de conservación a lo largo del genoma aplicando el proceso descrito en la sección anterior a 50—meros. Al considerar solo 50—meros que formaban parte de repeticiones ancestrales, es posible determinar un nivel de fondo de conservación. Podemos imaginar que las intensidades de conservación entre los 50 meros se distribuyen de acuerdo con una distribución de probabilidad oculta, como se ilustra en la Figura 4.8. En la figura, la curva de fondo representa la distribución de restricción en ausencia de mecanismos especiales para el exceso de restricción, según se determina al observar repeticiones ancestrales, mientras que la curva de señal (primer plano) representa la distribución real del genoma. La curva de señal tiene más conservación en general debido a los efectos purificadores de la selección natural.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

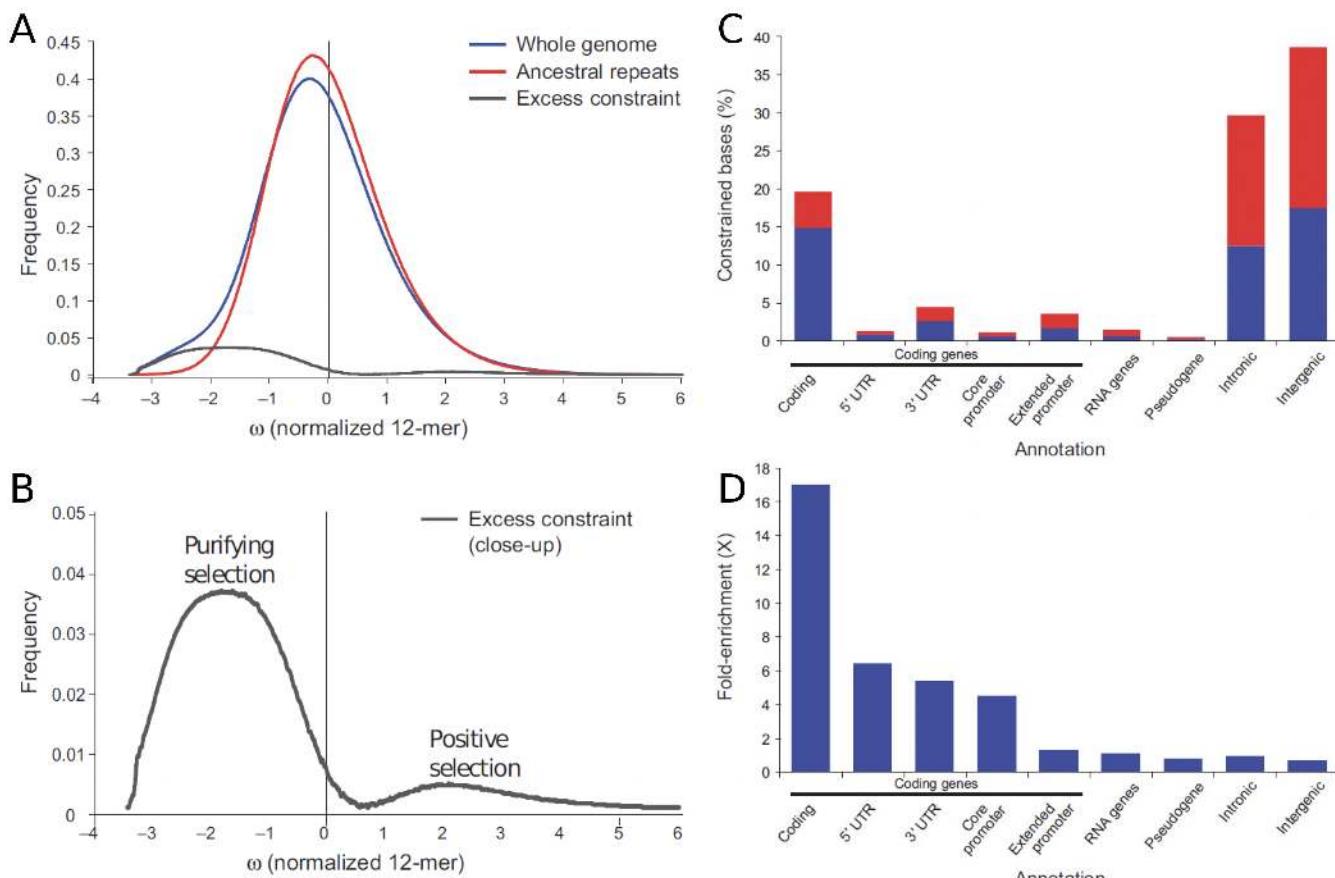
Figura 4.8: Medición del exceso de restricción genómica amplia. Consulte el texto adjunto para una explicación.

Es posible que deseemos investigar regiones específicas del genoma que están bajo restricciones excesivas estableciendo un nivel umbral de conservación y examinando las regiones que están más conservadas. En la ilustración, esto corresponde a considerar todos los 50—mers que caen a la derecha de una de las líneas anaranjadas. Vemos que si bien este método efectivamente nos da regiones bajo restricción excesiva, también nos da falsos positivos. Esto se debe a que incluso en ausencia de selección purificadora y otros efectos, ciertas regiones estarán fuertemente conservadas, simplemente por casualidad aleatoria. Establecer el umbral más alto, como usar la línea punteada naranja como nuestro umbral, reduce la proporción de falsos positivos (FP) a verdaderos positivos (TP), al tiempo que disminuye el número de verdaderos positivos detectados, negociando así mayor especificidad por menor sensibilidad.

No obstante, no toda esperanza está perdida. Es posible medir empíricamente las curvas de señal tanto de fondo (BG) como de primer plano (FG), como se describió anteriormente. Una vez hecho esto, el área de la región entre ellos, que está sombreada en gris en la Figura 4.8, se puede determinar por integración. Esta área representa la proporción del genoma que está bajo restricción excesiva. Debido a que las curvas se superponen, no podemos detectar todos los elementos conservados pero podemos estimar la cantidad total de restricción excesiva. Este número de restricción estimada resulta ser alrededor del 5% del genoma humano, dependiendo de qué tan grande se use una ventana. Es probable que todas esas regiones sean funcionales, pero dado que alrededor del 1.5% del genoma humano es codificante de proteínas, podemos inferir que el 3.5% restante consiste en elementos funcionales no codificantes, la mayoría de los cuales probablemente desempeñan funciones reguladoras.

Hemos visto que la restricción evolutiva sobre todo el genoma se puede estimar evaluando la restricción genómica contra una distribución de fondo. Lindblad-Toh et al. (2011) comparan la conservación del genoma en 29 mamíferos frente a un fondo calculado a partir de elementos de repetición ancestrales para encontrar regiones con restricción excesiva (Figura 4.9A y B). La anotación de bases evolutivamente restringidas revela que la mayoría de las regiones descubiertas son intergénicas e intrónicas y demuestra que pasar de cuatro (HMRD) a 29 genomas de mamíferos aumenta el poder de este análisis principalmente en regiones no codificantes (Figura 4.9C). Las regiones más restringidas en el genoma son regiones codificantes (Figura 4.9D).

Como se muestra en la Figura 4.9, el aumento de HMRD a una alineación de 29 genomas mejora enormemente la potencia



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.9: Detectando elementos funcionales a partir de su firma evolutiva. A Distribución de restricción para todo el genoma frente a repeticiones ancestrales (fondo). B Diferencia entre el genoma completo y la restricción de fondo. C Descubrimiento de elementos funcionales a partir del exceso de restricción. Los elementos novedosos se muestran en rojo. D Enriquecimiento de elementos para regiones de restricción excesiva.

de este análisis. Sin embargo, si bien la cantidad de elementos intergénicos detectados aumentó significativamente, la detección aún está limitada por el hecho de que los elementos no funcionales tienen una profundidad de cobertura de especies mucho menor en múltiples alineamientos que las regiones funcionales (Figura 4.10). Por ejemplo, las repeticiones ancestrales (AR,  $\mu = 11.4$ ) tienen una profundidad de cobertura promedio mucho menor que los exones ( $\mu = 20.9$ ). Por un lado, esto muestra evidencia de selección contra inserciones y delecciones en elementos funcionales, las cuales no son examinadas en el análisis de restricción base. Por otro lado, también complica el análisis de la restricción evolutiva, ya que dicho trabajo debe entonces manejar una cobertura variable a lo largo del genoma.

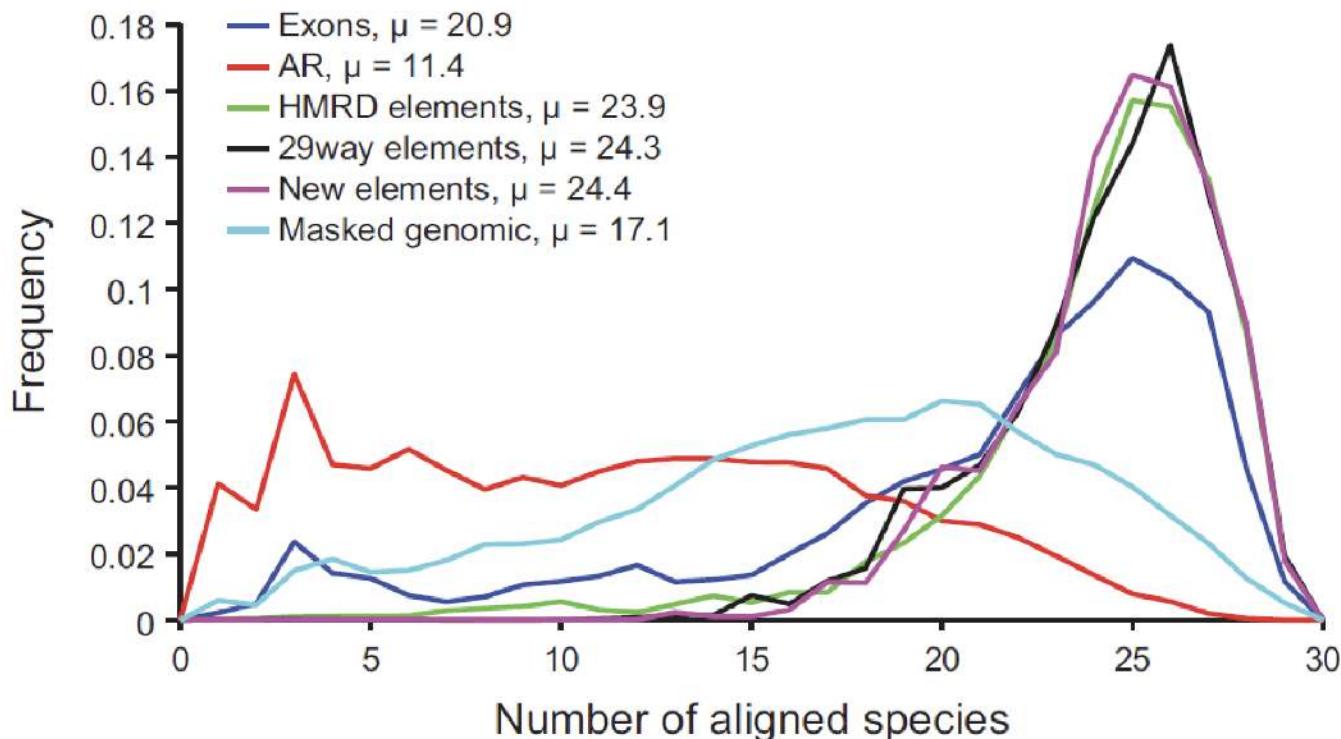
## Ejemplos de Restricción Excesiva

Se han encontrado ejemplos de restricción excesiva en los siguientes casos:

- La mayoría de los genes Hox muestran regiones de restricción superpuestas. En particular, como se mencionó anteriormente los primeros 50 aminoácidos de HOXB5 están casi completamente conservados. Además, HOXA2 muestra módulos regulatorios superpuestos. Estos dos loci codifican potenciadores del desarrollo, proporcionando un mecanismo para la expresión específica del tejido.
- ADAR: el principal regulador de la edición de ARNm, tiene una variante de empalme donde se encontró una baja tasa de sustitución sinónima a una resolución de 9 codones.
- BRCA1: Hurst y Pal (2001) encontraron una baja tasa de sustituciones sinónimas en ciertas regiones del BRCA1, el principal gen involucrado en el cáncer de mama. Ellos plantearon la hipótesis de que la selección purificadora se produce- anillo en estas

regiones. (Esta afirmación fue refutada por Schmid y Yang (2008) quienes afirman que este fenómeno es el artefacto de un análisis de ventana deslizante).

- THRA/NR1D1: estos genes, también involucrados en el cáncer de mama, forman parte de una región codificadora dual que codifica para ambos genes y está altamente conservada.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.10: Profundidad de cobertura a través de diferentes conjuntos de elementos.

- SEPHS2: tiene una horquilla involucrada en la recodificación de selenocisteína. Debido a que esta región debe seleccionar codones tanto para conservar la secuencia de aminoácidos de la proteína como los nucleótidos para mantener la misma estructura secundaria de ARN, muestra un exceso de restricción.

### Medición de restricción en nucleótidos individuales

Al medir la restricción evolutiva en nucleótidos individuales en lugar de bloques de la secuencia, podemos encontrar sitios de unión a factores de transcripción individuales, sesgo específico de posición dentro de instancias de motivos y revelar consenso de motivo entre la mayoría de las especies. Específicamente, podemos detectar SNP que interrumpen los motivos reguladores conservados y determinar el nivel de evolución observando cada nucleótido en el gen. Al observar los nucleótidos individualmente, podemos encontrar SNP que son importantes en la función de una secuencia específica.

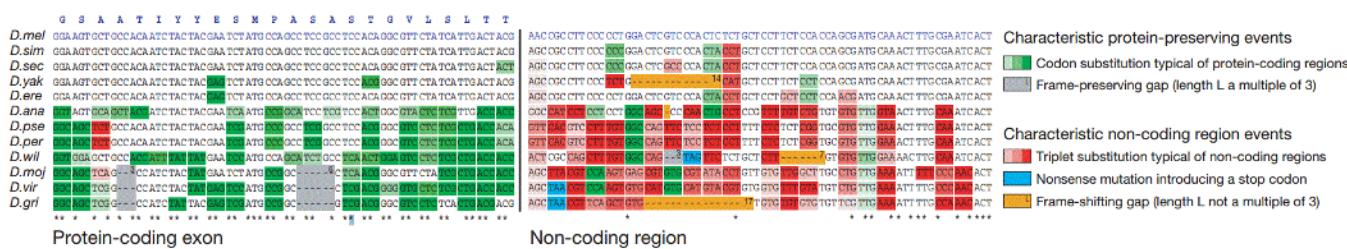
<sup>2</sup> [www.nature.com/nature/journal...ture10530.html](http://www.nature.com/nature/journal...ture10530.html)

This page titled 4.3: Restricción en exceso is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Manolis Kellis et al. (MIT OpenCourseWare) via source content that was edited to the style and standards of the LibreTexts platform.

- 4.3: Excess Constraint by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección

Independientemente de la tasa de sustitución, también podemos considerar el patrón de sustituciones en una subsecuencia de nucleótidos particular. Considerar una secuencia de nucleótidos que codifica una proteína. Debido a la oscilación del ARNt, una mutación en el tercer nucleótido de un codón es menos probable que afecte a la proteína final que una mutación en las otras posiciones. Por lo tanto, esperamos ver un patrón de sustituciones incrementadas en la tercera posición cuando se observan subsecuencias codificantes de proteínas del genoma. Esto efectivamente se verifica experimentalmente, como se muestra en la Figura 4.11.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

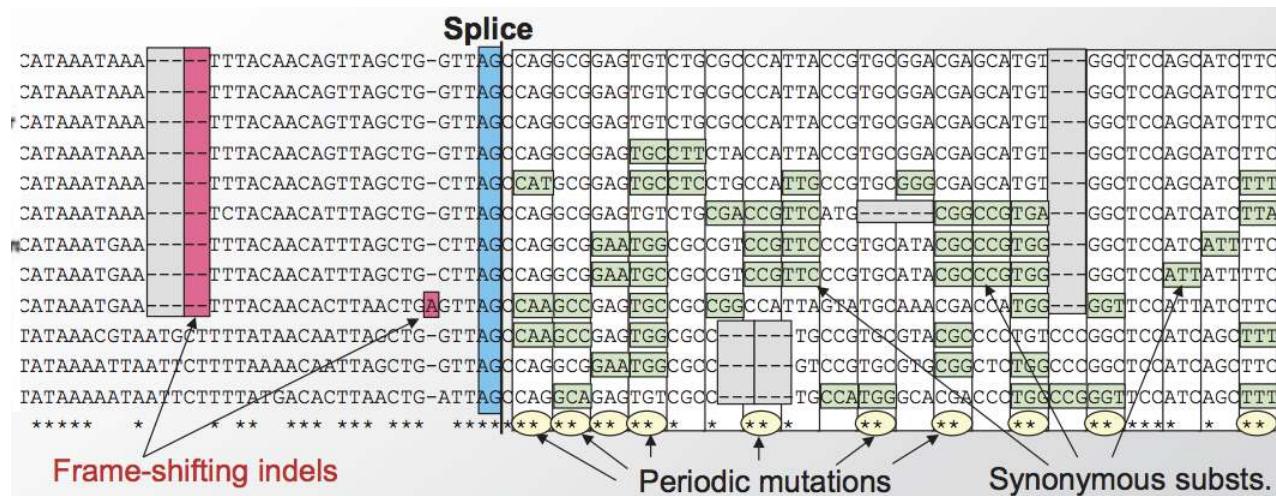
Figura 4.11: Diferentes patrones de mutación en regiones codificadoras de proteínas y no codificadoras de proteínas. Los asteriscos indican que el nucleótido se conservó en todas las especies. Obsérvese que dentro del exón que codifica la proteína, los nucleótidos 1 y 2 de cada codón tienden a conservarse, mientras que el codón 3 puede variar más, lo cual es consistente con el fenómeno de bamboleo.

### Preguntas frecuentes

P: En la Figura 4.11, también vemos sustituciones de nucleótidos en grupos de tres o seis. ¿Por qué es este el caso?

R: Las inserciones y delecciones en grupos de tres y seis también contribuyen a preservar el marco de lectura. Si todos los nucleótidos se eliminan en un codón, el resto de los codones no se ven afectados durante la traducción de aminoácidos. Sin embargo, si eliminamos un número de nucleótidos que no es un múltiplo de tres (es decir, solo eliminamos parte de algún codón), entonces la traducción del resto de los codones se vuelve sin sentido ya que el marco de lectura se ha desplazado.

En la Figura 4.12, podemos ver una característica más de los genes que codifican proteínas. Los límites de conservación son muy distintos y se encuentran cerca de sitios de empalme. Las mutaciones periódicas (en múltiplos de tres) comienzan a ocurrir después del límite del sitio de empalme.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.12: Además de la conservación del marco de lectura y las sustituciones cada tercer nucleótido, también vemos límites de conservación definidos que señalan los sitios de corte y empalme.

Como podemos ver con la detección de genes codificantes de proteínas, no solo es importante considerar la tasa de sustitución sino también el patrón de sustituciones. Al observar cómo se conservan las regiones, en lugar de solo mirar la cantidad de conservación, podemos observar 'firmas evolutivas' de conservación para diferentes elementos funcionales.

### Presiones selectivas en diferentes elementos funcionales

Diferentes elementos funcionales tienen diferentes presiones selectivas (debido a su estructura y otras características); algunos cambios ( inserciones, delecciones o mutaciones) que pueden ser extremadamente dañinos para un elemento funcional pueden ser inocuos para otro. Al averiguar cuáles son las "firmas" para diferentes elementos, podemos anotar con mayor precisión una región observando los patrones de conservación que muestra.

Tal patrón se llama una firma evolutiva: un patrón de cambio que se tolera dentro de elementos que aún conservan su función. Una firma evolutiva es diferente del grado de conservación en que tolera la mutación, pero solo tipos específicos de mutaciones en lugares específicos. Las firmas evolutivas surgen porque la evolución y la selección natural están actuando en diferentes niveles en ciertos elementos funcionales. Por ejemplo, en un gen codificador de proteínas la evolución está actuando a nivel de aminoácidos, por lo que la selección natural no filtrará los cambios de nucleótidos que no afectan a la secuencia de aminoácidos. Mientras que un ARN estructural tendrá presión para preservar pares de nucleótidos, pero no necesariamente nucleótidos individuales.

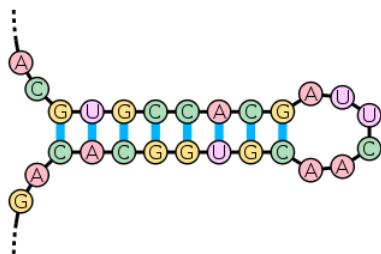
Es importante destacar que el patrón de conservación tiene una estructura filogenética distinta. Más especies similares (mamíferos) se agrupan junto con dominios conservados compartidos de los que carecen los peces, lo que sugiere una innovación específica de mamíferos, tal vez para elementos reguladores no compartidos por peces. Mientras tanto, algunas características se conservan globalmente, lo que sugiere una significación universal, como la codificación de proteínas. La anotación aproximada inicial de las regiones codificadoras de proteínas en el genoma humano fue posible usando la heurística simple que si se conservaba de humano a pescado probablemente sirvió como región codificante de proteínas.

Una idea interesante para un proyecto final sería mapear las divergencias en la alineación múltiple y llamar a estos eventos "nacimientos" de nuevos elementos de codificación. Al enfocarse en un elemento particular (digamos microARN) se podrían identificar períodos de innovación y aislar porciones de un árbol filogenético enriquecidas para ciertas clases de estos elementos.

El resto del capítulo se centrará en cuantificar el grado en que una secuencia sigue un patrón dado. Kellis comparó el proceso de evolución con la exploración de un paisaje de fitness, con la puntuación de aptitud de una secuencia particular restringida por la función que codifica. Por ejemplo, los genes codificantes de proteínas están restringidos por la selección en el producto traducido, por lo que se toleran sustituciones sinónimas en el tercer par de bases de un codón.

A continuación se muestra un resumen de los patrones esperados seguidos de varios elementos funcionales:

- Los genes que codifican proteínas exhiben frecuencias particulares de sustitución de codones, así como conservación del marco de lectura. Esto tiene sentido porque la importancia de los genes son las proteínas para las que codifican; por lo tanto, los cambios que dan como resultado aminoácidos iguales o similares pueden tolerarse fácilmente, mientras que un pequeño cambio que cambia drásticamente la proteína resultante puede considerarse desastroso. Además de la corrección de errores del sistema de reparación de desapareamientos y la propia ADN polimerasa, la redundancia del código genético proporciona un nivel adicional de corrección/tolerancia intrínseca de errores.
- El ARN estructural se selecciona en base a la secuencia secundaria del ARN transcrita, y por lo tanto requiere cambios compensatorios. Por ejemplo, algunos ARN tienen una estructura secundaria tallo-bucle tal que secciones de su secuencia se unen a otras secciones de su secuencia en su “tallo”, como se muestra en la figura 4.13.



Cortesía de Sakurambo en Wikipedia. Imagen en el dominio público.

Figura 4.13: ARN con estructura secundaria tallo-bucle

Imagínese que un nucleótido (A) y su compañero (T) se unen entre sí en el tallo, y luego (A) muta a a (C). Esto arruinaría la estructura secundaria del ARN. Para corregir esto, o el (C) mutaría de nuevo a un (A), o el (T) mutaría a a (G). Entonces el par (C) - (G) mantendría la estructura secundaria. A esto se le llama una mutación compensatoria. Por lo tanto, en las estructuras de ARN, la cantidad de cambio en la estructura secundaria (por ejemplo, tallo-bucle) es más importante que la cantidad de cambio en la estructura primaria (solo la secuencia). Comprender los efectos de los cambios en la estructura del ARN requiere el conocimiento de la estructura secundaria. La probable estructura secundaria de un ARN se puede determinar modelando la estabilidad de muchas conformaciones posibles y eligiendo la conformación más probable.

- El microARN es una molécula que se expulsa del núcleo al citoplasma. Su rasgo característico es que también tienen la estructura de horquilla (tallos-bucle) ilustrada en la Figura 4.13, pero una sección del tallo es complementaria a una porción de ARNm.
- Cuando el microARN une su secuencia complementaria a la porción respectiva del ARNm, degrada el ARNm. Esto quiere decir que es un regulador posttranscripcional, ya que se está utilizando para limitar la producción de una proteína (traducción) después de la transcripción. El microARN se conserva de manera diferente al ARN estructural. Debido a su unión a una diana de ARNm, la región de unión está mucho más conservada para mantener la especificidad de la diana.
- Finalmente, los motivos reguladores se conservan en secuencia (para unirse a parejas proteicas interaccionantes particulares) pero no necesariamente en ubicación. Los motivos regulatorios pueden moverse ya que solo necesitan reclutar un factor para una región en particular. Se toleran pequeños cambios (inserciones y delecciones) que preservan el consenso del motivo, así como los cambios aguas arriba y aguas abajo que mueven la ubicación del motivo.

Al tratar de entender el papel de la conservación en la predicción de clases funcionales, una pregunta importante es qué tanto de la conservación observada puede explicarse por patrones conocidos. Incluso después de dar cuenta de la conservación “aleatoria”, aproximadamente el 60% de la conservación no aleatoria en el genoma de la mosca no se tuvo en cuenta, es decir, no pudimos identificarla como un gen codificador de proteínas, ARN, microARN o motivo regulador. Sin embargo, el hecho de que permanezcan conservados sugiere un papel funcional. Esa secuencia tan conservada sigue siendo poco entendida subraya que quedan muchas preguntas emocionantes por responder. Un proyecto final para 6.047 en el pasado fue usar clustering (aprendizaje no supervisado) para dar cuenta de la otra conservación. Se convirtió en un proyecto M.Eng, y se identificaron algunos clusters, pero la función de estos clusters era, y es, aún no está clara. ¡Es un problema abierto!

This page titled [4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **4.4: Diversity of evolutionary signatures- An Overview of Selection Patterns** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.5: Firmas de codificación de proteínas

En la diapositiva 12, vemos tres ejemplos de conservación: una secuencia intrónica con mala conservación, una región codificante con alta conservación y una región no codificante con alta conservación, lo que significa que probablemente sea un elemento funcional. Como vimos al inicio de esta sección, la característica importante de las regiones codificadoras de proteínas para recordar es que los codones (triples de nucleótidos) codifican los aminoácidos, que componen las proteínas. Esto da como resultado la firma evolutiva de las regiones codificadoras de proteínas, como se muestra en la diapositiva 13: (i) conservación del marco de lectura y (ii) patrones de sustitución de codones. La intuición para esta firma es relativamente sencilla.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.14: Algunas mutaciones puntuales a la secuencia de ADN no cambian la traducción de proteínas

En primer lugar, la conservación del marco de lectura tiene sentido, ya que una inserción o delección de uno o dos nucleótidos “desplazará” la forma en que se leen todos los siguientes codones. Sin embargo, si ocurre una inserción o eliminación en un múltiplo de 3, los otros codones seguirán siendo leídos de la misma manera, por lo que este es un cambio menos significativo. En segundo lugar, tiene sentido que algunas mutaciones sean menos dañinas que otras, ya que diferentes tripletes pueden codificar para los mismos aminoácidos (una sustitución conservadora, como se desprende de la matriz de abajo), e incluso las mutaciones que dan como resultado un aminoácido diferente pueden ser evolutivamente neutras si se producen las sustituciones con aminoácidos similares en un dominio de la proteína donde no se requieren las propiedades exactas de los aminoácidos. Estos patrones distintivos nos permiten “colorear” el genoma y ver claramente dónde están los exones, como se muestra en la Figura 4.15.

Al usar estos patrones para distinguir las firmas evolutivas, tenemos que asegurarnos de considerar las siguientes ideas:

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.15: Al colorear los tipos de inserciones/delecciones/sustituciones que ocurren en una secuencia, podemos ver patrones o firmas evolutivas que distinguen una región conservada codificante de proteína de una región conservada que no codifica proteínas.

- Cuantificar la distinción de las  $64^2$  posibles sustituciones de codones considerando regiones sinónimas (frecuentes en secuencias codificantes de proteínas) y sin sentido (más frecuentes en secuencias no codificantes que codificantes).
- Modelar la relación filogenética entre las especies: múltiples sustituciones aparentes pueden ser ex- plaqueadas por un evento evolutivo.
- Tolerar la incertidumbre en la entrada como secuencias ancestrales desconocidas y brechas en alineación (datos faltantes).
- Informar la certeza o incertidumbre del resultado: cuantificar la confianza de que una alineación dada es codificadora de proteínas utilizando diversas unidades como valor p, bits, decibans... etc.

### Lectura: conservación de marco (RFC)

Ahora que conocemos este patrón de conservación en genes codificadores de proteínas, podemos desarrollar métodos para determinar si un gen es codificante de proteínas o si no lo es.

Al puntuar la presión para permanecer en el mismo marco de lectura, podemos cuantificar la probabilidad de que una región sea codificadora de proteínas o no. Como se muestra en la diapositiva 20, podemos hacer esto teniendo una secuencia diana (Scer, el genoma de *S. cerevisiae*), y luego alinear una secuencia de selección (Spar, *S. paradoxus*) con ella y calculando qué proporción del tiempo la secuencia seleccionada coincide con el marco de lectura de la secuencia diana.

Como no sabemos dónde comienza el marco de lectura en la secuencia seleccionada, alineamos tres veces para probar todas las compensaciones posibles:

(Sparf1, Sparf2, Sparf3)

A partir de estos, elegimos el alineamiento donde la secuencia seleccionada suele estar sincronizada con la secuencia diana. Por ejemplo, podemos comenzar a numerar los nucleótidos “1, 2, 3... etc.” hasta llegar a una brecha que no numeramos. O podemos comenzar a numerar los nucleótidos “2, 3, 1... etc.” donde cada triplete de “1,2,3” representa un codón.

Finalmente, para el mejor alineamiento, calculamos el porcentaje de nucleótidos que están fuera de marco —si está por encima de un límite, esta especie seleccionada “vota” que esta región es una región codificadora de proteínas, y si es baja, esta especie “vota” que se trata de una región intergénica. Se contabilizan los “votos” de todas las especies para sumarse al puntaje RFC.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.16: Dos alineamientos que muestran diferencias de patrones de conservación entre secuencias génicas e intergénicas. Los cuadros rojos representan huecos que desplazan el marco de codificación, y los cuadros grises son huecos que no cambian de cuadro (en múltiplos de tres). Las regiones verdes se conservan y las amarillas están mutadas. Obsérvese el patrón de “coincidencia, coincidencia, falta de coincidencia” en la secuencia codificante de proteínas que indica mutaciones sinónimas.

Este método no es robusto al error de secuenciación. Podemos compensar estos errores usando una ventana de escaneo más pequeña y observando la conservación del marco de lectura local.

Se demostró que el método tiene 99.9% de especificidad y 99% de sensibilidad cuando se aplica al genoma de levadura. Cuando se aplicó a 2000 ORF hipotéticos (marcos de lectura abiertos, o genes propuestos)<sup>3</sup> en levaduras, rechazó 500 de estos supuestos genes codificantes de proteínas por no ser codificantes de proteínas.

De igual manera, 4000 genes hipotéticos en el genoma humano fueron rechazados por este método. Este modelo creó una hipótesis específica (que era poco probable que estas secuencias de ADN codificaran proteínas) que posteriormente se ha apoyado con la confirmación experimental de que las regiones no codifican proteínas *in vivo*.<sup>4</sup>

Esto representa un importante paso adelante para la anotación del genoma, ya que anteriormente era difícil concluir que una secuencia de ADN no codificaba simplemente por falta de evidencia. Al reducir el enfoque y crear una nueva hipótesis nula (que el gen en cuestión parece ser un gen no codificador) se hizo mucho más fácil no solo aceptar genes codificantes, sino rechazar los genes no codificantes con soporte computacional. Durante la discusión sobre la conservación del marco de lectura en clase, identificamos una idea emocionante para un proyecto final que sería buscar el nacimiento de nuevas proteínas funcionales resultantes de mutaciones de cambio de marco.

## Frecuencias de sustitución de codones (CSF)

La segunda firma de las regiones codificadoras de proteínas, las frecuencias de sustitución de codones, actúa sobre múltiples niveles de conservación. Para explorar estas frecuencias, es útil recordar que la evolución de codones puede ser modelada

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.17: Los cuadros rojos representan huecos de desplazamiento de marco, y los huecos en múltiplos de tres son incoloros. Las regiones conservadas y mutadas son verdes y amarillas, respectivamente.

por distribuciones de probabilidad condicional (CPD) — la probabilidad de que un descendiente tenga un codón b donde un antepasado tenía el codón a una cantidad de tiempo t ago.

El evento más conservador es el mantenimiento exacto del codón. Una mutación que codifica para el mismo aminoácido puede ser conservadora pero no totalmente sinónimo, debido a los sesgos de uso de codones específicos de especies. Incluso las mutaciones que alteran la identidad del aminoácido podrían ser conservadoras si codifican aminoácidos con propiedades bioquímicas similares.

Utilizamos un DPC para capturar el efecto neto de todas estas consideraciones. Para calcular estos CPD, necesitamos una matriz de “tasa”, Q, que mida el tipo de cambio para una unidad de tiempo; es decir, indica con qué frecuencia el codón a en la especie 1 se sustituye por el codón b en la especie 2, por una longitud de rama unitaria. Entonces, usando  $e^{Qt}$ , podemos estimar la frecuencia de sustitución en el tiempo t.

Cuando el CPD se considera en conjunto con la topología de una gráfica de red que representa el árbol evolutivo, tiene aproximadamente  $(2L - 2) \cdot 64^2$  parámetros, donde L es el número de hojas en el árbol (especies en la filogenia evolutiva). Este número de parámetros se deriva del número de entradas en Q y el número de longitudes de rama independientes, t. Las estimaciones de estos parámetros pueden ser determinadas por MLE a partir de datos de entrenamiento.

El CPD se define en términos de  $e^{Qt}$  de la siguiente manera:

```
\begin{ecuación}
\operatorname{Pr} (\text{child} = a \mid \text{parent} = b; t) = \left[ e^{\sum Q_{tij}} \right]_{a,b}
\end{ecuación}
```

La intuición, es que a medida que aumenta el tiempo, aumenta la probabilidad de sustituciones, mientras que en el tiempo “inicial” ( $t = 0$ ),  $e^{Qt}$  es la matriz de identidad, ya que se garantiza que cada codón sea él mismo. Pero, ¿cómo obtenemos la matriz de tarifas?

- $Q$  se “aprende” de las secuencias, mediante el uso de Expectación-Maximización, por ejemplo. Muchas secuencias codificadoras de proteínas conocidas se utilizan como datos de entrenamiento (o regiones no codificantes al generar ese modelo).
- Dados los parámetros del modelo, podemos usar el algoritmo de Felsenstein [1] para calcular la probabilidad de cualquier alineación, teniendo en cuenta la filogenia, dado el modelo de sustitución (el paso E).

```
\begin{ecuación}
\text{Probabilidad} (\boldsymbol{Q}) = \operatorname{Pr} (\text{Datos de entrenamiento}; \boldsymbol{Q}, t)
\end{ecuación}
```

- Entonces, dadas las alineaciones y la filogenia, podemos elegir los parámetros (la matriz de tasa:  $Q$ , y longitudes de rama:  $t$ ) que maximicen la probabilidad de esas alineaciones en el paso M; por ejemplo, para estimar  $Q$ , podemos contar el número de veces que un codón es sustituido por otro en el alineamiento. El espacio de argumento consiste en miles de posibilidades para  $Q$  y  $t$ . Este espacio está representado por  $Q$ .
- $$\hat{Q} = \arg \max_{\boldsymbol{Q}} \text{Probabilidad} (\boldsymbol{Q})$$
- Este espacio está representado por  $Q$ .  $\hat{Q}$  es el parámetro que maximiza la probabilidad:

```
\begin{ecuación}
\hat{Q} = \arg \max_{\boldsymbol{Q}} \text{Probabilidad} (\boldsymbol{Q})
\end{ecuación}
```

Otras estrategias de maximización incluyen: maximización de expectativas, ascenso de gradiente, recocido simulado, descomposición espectral. La longitud de la rama,  $t$ , se puede optimizar usando el mismo método simultáneamente.

## FAQ

P: ¿Cómo contribuye la longitud de la rama a determinar la matriz de tasas?

R: Las longitudes de las ramas especifican cuánto “tiempo” pasó entre dos nodos cualesquiera. La matriz de velocidad describe las frecuencias relativas de sustituciones de codones por unidad de longitud de rama.

Con dos matrices de tasas estimadas, las probabilidades calculadas de cualquier alineación dada son diferentes para cada matriz. Ahora, podemos comparar la razón de verosimilitud  $\frac{\Pr(\text{Leaves}; Q_{C,t})}{\Pr(\text{Leaves}; Q_{N,t})}$ , que el alineamiento vino de una región codificante de proteínas en lugar de provenir de una región que no codifica proteínas.

(a) Matriz de tasas  $Q_N$  estimada a partir de regiones no codificantes

(b) Matriz de tasas  $Q_C$  estimada a partir de regiones codificantes conocidas.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.18: Matrices de tasas para los modelos nulos y alternos. Un color más claro significa que la sustitución es más probable.

Ahora que sabemos cómo obtener nuestro modelo, observamos que, dado el patrón específico de frecuencias de sustitución de codones para la codificación de proteínas, queremos dos modelos para que podamos distinguir entre regiones codificantes y no codificantes. Las figuras 4.18a y 4.18b muestran matrices de tasas para regiones intergénicas y génicas, respectivamente. Una serie de características sobresalientes se presentan en la matriz de sustitución de codones (CSM) para genes. Obsérvese que se ha eliminado el elemento diagonal principal, ya que la frecuencia de un triplete que se intercambia por sí mismo obviamente será mucho mayor que cualquier otro intercambio. Sin embargo,

1. inmediatamente es obvio que existe un fuerte elemento diagonal en las regiones codificantes de proteínas.
2. También observamos ciertos elementos diagonales de alta puntuación en el CSM codificador: estas son sustituciones que tienen una función más cercana que en secuencia, como codones degenerados de 6 veces o aminoácidos muy similares.
3. También observamos franjas verticales oscuras, que indican que estas sustituciones son especialmente improbables. Estas columnas corresponden a codones de parada, ya que las sustituciones a este triplete alterarían significativamente la función de la proteína, y por lo tanto se seleccionan fuertemente contra.

Por otro lado, en la matriz para regiones intergénicas, los tipos de cambio son más uniformes. En estas regiones, lo que importa es la proximidad mutacional, es decir, la distancia de edición o el número de cambios de una secuencia a otra. Las regiones genéticas están dictadas por la proximidad selectiva, o la similitud en la secuencia de aminoácidos de la proteína resultante del gen.

Ahora que tenemos las dos matrices de tasas para las dos regiones, podemos calcular las probabilidades de que cada matriz genere los genomas de las dos especies. Esto se puede hacer usando el algoritmo de Felsenstein, y sumando la “puntuación” para cada par de codones correspondientes en las dos especies. Finalmente, podemos calcular la razón de verosimilitud de que el alineamiento vino de una región codificante a una región no codificante dividiendo las dos puntuaciones, esto demuestra nuestra confianza en nuestra anotación de la secuencia. Si la relación es mayor que 1, podemos adivinar que es una región codificante, y si es menor que 1, entonces es una región no codificante. Por ejemplo, en la Figura 4.16, tenemos mucha confianza en las respectivas clasificaciones de cada región.

Cabe señalar, sin embargo, que aunque la “coloración” de las secuencias confirma nuestras clasificaciones, las razones de verosimilitud se calculan independientemente de la ‘coloración’, que utiliza nuestro conocimiento de sustituciones sinónimas o conservadoras. Esto implica además que este método infiere automáticamente el código genético a partir del patrón de sustituciones que se produce, simplemente observando las sustituciones de alta puntuación. En especies con un código genético diferente, los patrones de intercambio de codones serán diferentes; por ejemplo, en la albúmina de Candida, el CTG codifica serina (polar) en lugar de leucina (hidrófoba), y esto se puede deducir de los CSM. Sin embargo, el método no requiere ningún conocimiento de esto; en cambio, podemos deducir esto a posteriori del CSM.

En resumen, podemos distinguir entre regiones no codificantes y codificantes del genoma en función de sus firmas evolutivas, mediante la creación de dos matrices separadas de 64 por 64 tasas: una que mide la tasa de sustituciones de codones en regiones codificantes y la otra en regiones no codificantes. La matriz de tasas da la tasa de cambio de codones o nucleótidos a lo largo de una unidad de tiempo.

Se utilizaron las dos matrices para calcular dos probabilidades para cualquier alineación dada: la probabilidad de que proviniera de una región codificante y la probabilidad de que proviniera de una región no codificante. Tomando la razón de verosimilitud de estas dos probabilidades da una medida de confianza de que el alineamiento es proteína, codificando como estrato demoníaco en la Figura 4.19. Mediante este método podemos seleccionar regiones del genoma que evolucionan de acuerdo con la firma codificante de la proteína.

puntuaciones — esto demuestra nuestra confianza en nuestra anotación de la secuencia. Si la relación es mayor que 1, podemos adivinar que es una región codificante, y si es menor que 1, entonces es una región no codificante. Por ejemplo, en la Figura 4.16, tenemos mucha confianza en las respectivas clasificaciones de cada región.

Cabe señalar, sin embargo, que aunque la “coloración” de las secuencias confirma nuestras clasificaciones, las razones de verosimilitud se calculan independientemente de la ‘coloración’, que utiliza nuestro conocimiento de sustituciones sinónimas o conservadoras. Esto implica además que este método infiere automáticamente el código genético a partir del patrón de sustituciones que se produce, simplemente observando las sustituciones de alta puntuación. En especies con un código genético diferente, los patrones de intercambio de codones serán diferentes; por ejemplo, en la albúmina de Candida, el CTG codifica serina (polar) en lugar de leucina (hidrófoba), y esto se puede deducir de los CSM. Sin embargo, el método no requiere ningún conocimiento de esto; en cambio, podemos deducir esto a posteriori del CSM.

En resumen, podemos distinguir entre regiones no codificantes y codificantes del genoma en función de sus firmas evolutivas, mediante la creación de dos matrices separadas de 64 por 64 tasas: una que mide la tasa de sustituciones de codones en regiones codificantes y la otra en regiones no codificantes. La matriz de tasas da la tasa de cambio de codones o nucleótidos a lo largo de una unidad de tiempo.

Se utilizaron las dos matrices para calcular dos probabilidades para cualquier alineación dada: la probabilidad de que proviniera de una región codificante y la probabilidad de que proviniera de una región no codificante. Tomando la razón de verosimilitud de estas dos probabilidades da una medida de confianza de que el alineamiento es proteína, codificando como estrato demoníaco en la Figura 4.19. Mediante este método podemos seleccionar regiones del genoma que evolucionan de acuerdo con la firma codificante de la proteína.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.19: Como podemos ver en la figura que la razón de verosimilitud es positiva para secuencias que probablemente sean codificantes de proteínas y negativas para secuencias que probablemente no sean codificantes de proteínas.

Veremos más adelante cómo combinar este enfoque de cociente de verosimilitud con métodos filogenéticos para encontrar patrones evolutivos de regiones codificantes de proteínas.

Sin embargo, este método solo nos permite encontrar regiones que se seleccionan a nivel traslacional. El punto clave es que aquí estamos midiendo solo para la selección de codificación de proteínas. Veremos hoy cómo podemos buscar otros elementos funcionales conservados que exhiban sus propias firmas únicas.

## Clasificación de las secuencias del genoma de *Drosophila*

Hemos visto que el uso de estas métricas de RFC y CSF nos permite clasificar exones e intrones con especificidad y sensibilidad extremadamente altas. Los clasificadores que utilizan estas medidas para clasificar secuencias se pueden implementar usando un campo aleatorio condicional (SMCRF) HMM o Semi-Markov. Los CRF permiten la integración de diversas características que no necesariamente tienen una naturaleza probabilística, mientras que los HMM requieren que modelaremos todo como probabilidades de transición y emisión. Los CRF serán discutidos en una próxima conferencia. Uno podría preguntarse por qué es necesario implementar estos métodos más complejos, cuando el método más simple de verificar la conservación del marco de lectura funcionó bien. La razón es que en regiones muy cortas, las inserciones y delecciones serán muy poco frecuentes, incluso por casualidad, por lo que no habrá suficiente señal para hacer la distinción entre regiones codificantes de proteínas y no codificantes de proteínas. En la siguiente figura, vemos una secuencia de ADN a lo largo del eje x, con las filas que representan un gen anotado, cantidad de conservación, cantidad de proteína que codifica la firma evolutiva y el resultado de la decodificación de Viterbi usando el SMCRF, respectivamente.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.20: Las firmas evolutivas pueden predecir nuevos genes y exones. La estrella denota un nuevo exón, el cual se predijo mediante las tres pruebas de genómica comparativa, y posteriormente se verificó mediante secuenciación de ADNc.

Este es un ejemplo de cómo la utilización de la firma codificante de proteínas para clasificar regiones ha demostrado ser muy exitosa. La identificación de regiones que se habían pensado que eran genes pero que no tenían alto contenido proteico, las firmas codificantes nos permitieron rechazar fuertemente 414 genes en el genoma de la mosca previamente clasificados como CGID, solo genes, lo que llevó a los curadores de FlyBase a eliminar 222 de ellos y marcar otros 73 como inciertos. Además, también hubo falsos negativos definitivos, ya que existía evidencia funcional para los genes bajo examen. Finalmente, en los datos, también vemos regiones con ambas conversaciones, así como una gran firma codificante de proteínas, pero no se habían marcado previamente como partes de genes, como en la Figura 4.20. Algunos de estos han sido probados experimentalmente y se ha demostrado que son partes de nuevos genes o extensiones de genes existentes. Esto subraya la utilidad de la biología computacional para apalancar y dirigir el trabajo experimental.

## Codones de parada con fugas

Los codones de parada (TAA, TAG, TGA en ADN y UAG, UAA, UGA en ARN) suelen señalar el final de un gen. Reflejan claramente la terminación de la traducción cuando se encuentran en el ARNm y liberan la cadena de aminoácidos del ribosoma. Sin embargo, en algunos casos inusuales, la traducción se observa más allá del primer codón de parada. En casos de lectura única, hay un codón de terminación que se encuentra dentro de una región con una proteína clara que codifica la firma seguida de un segundo codón de parada a corta distancia. Un ejemplo de esto en el genoma humano se da en la Figura 4.21. Esto sugiere que la traducción

continúa a través del primer codón de parada. También se han observado casos de doble lectura, donde dos codones de parada se encuentran dentro de una región codificante de proteínas. En estos casos de supresión de codones de terminación, se encuentra que el codón de parada está altamente conservado, lo que sugiere que estos codones de parada omitidos juegan un papel biológico importante.

La lectura translacional se conserva en ambas moscas, que tienen 350 proteínas identificadas que exhiben lectura de codones de parada, y en humanos, que tienen 4 instancias identificadas de tales proteínas. Se observan principalmente en proteínas neuronales en cerebros adultos y proteínas expresadas en el cerebro en *Drosophila*.

El gen kelch exhibe otro ejemplo de supresión de codones de parada en el trabajo. El gen codifica dos ORF con un único codón de parada UGA entre ellos. De esta secuencia se traducen dos proteínas, una del primer ORF y otra de la secuencia completa. La proporción de las dos proteínas está regulada de una manera específica del tejido. En el caso del gen kelch, una mutación del codón de parada de UGA a UAA da como resultado una pérdida de función, lo que sugiere que la supresión del ARNt es el mecanismo detrás de la supresión del codón de parada.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 4.21: Neurotransmisor OPRL1: uno de los cuatro nuevos candidatos de lectura translacional en el genoma humano. Nótese que la región después del primer codón de parada exhibe una firma evolutiva similar a la de la región codificante antes del codón de terminación, lo que indica que el codón de terminación está “suprimido”.

Un ejemplo adicional de supresión de codones de parada es Caki, una proteína activa en la regulación de la liberación de neurotransmisores en *Drosophila*. Los marcos de lectura abiertos (ORF) son secuencias de ADN que contienen un codón de inicio y terminación. En Caki, leer el gen en el primer marco de lectura (Frame 0) da como resultado significativamente más ORF que leer en Frame 1 o Frame 2 (un exceso de ORF 440). En la Figura 4.22 se enumeran doce posibles interpretaciones para el exceso de ORF. Sin embargo, debido a que el exceso se observa solo en el Cuadro 0, solo son probables las primeras 4 interpretaciones:

- Lectura de codones de parada: el codón de parada se suprime cuando el ribosoma extrae ARNt que se empareja incorrectamente con el codón de parada.
- Tonterías recientes: Quizás alguna mutación sin sentido reciente está causando la lectura de codones de parada.
- A a I edición: A diferencia de lo que pensábamos anteriormente, el ARN todavía se puede editar después de la transcripción. En algunos casos la base A se cambia a una I, que puede leerse como una G. Esto podría cambiar un codón de parada de TGA a un TGG, que codifica un aminoácido. Sin embargo, este fenómeno sólo se encuentra en un par de casos.
- Selenocisteína, el “21º aminoácido”: A veces, cuando el codón TGA es leído por un cierto bucle que conduce a un pliegue específico del ARN, se puede decodificar como selenocisteína. Sin embargo, esto solo ocurre en cuatro proteínas de mosca, por lo que no se puede explicar toda la supresión de codones de parada.

Entre estos cuatro, tres de ellos (tonterías recientes, edición A a I, y selenocisteína) representan sólo 17 de los casos. De ahí que parezca que la lectura completa debe ser responsable de la mayoría, si no de todos, de los casos restantes. Además, se observa el uso de codones de parada sesgados, por lo que se descartan otros procesos como el corte y empalme alternativo (donde los exones de ARN después de la transcripción se reconectan de múltiples maneras que conducen a múltiples proteínas) o ORF independientes.

Las regiones de lectura pueden determinarse en una sola especie en función de su patrón de uso de codones. La curva Z, como se muestra en la Figura 4.23, mide los patrones de uso de codones en una región de ADN. A partir de la figura, se puede observar que la región de lectura coincide con la distribución antes del codón de parada regular. Sin embargo, después de la segunda parada, la región coincide con las regiones encontradas después de las paradas regulares.

Otra sugerencia ofrecida en clase fue la posibilidad de deslizamiento del ribosoma, donde el ribosoma salta algunas bases durante la traducción. Esto podría hacer que el ribosoma se salte más allá de un codón de parada. Este evento ocurre en genomas bacterianos y virales, los cuales tienen una mayor presión para mantener sus genomas pequeños, y por lo tanto pueden usar esta técnica de deslizamiento para leer una sola transcripción en cada marco de lectura diferente. Sin embargo, los humanos y las moscas no están bajo una presión tan extrema para mantener sus genomas pequeños. Adicionalmente, se demostró anteriormente que el exceso que observamos más allá del codón de parada es específico del marco 0, lo que sugiere que el deslizamiento del ribosoma no es responsable.

Las células son estocásticas en general y la mayoría de los procesos toleran errores a bajas frecuencias. El sistema no es perfecto y ocurren fugas de codones de parada. Sin embargo, la siguiente evidencia sugiere que la lectura de codones de parada no es aleatoria sino que está sujeta a control regulador:

- La conservación perfecta de los codones de parada de lectura se observa en 93% de los casos, lo que es muy superior al 24% encontrado en el fondo.
- Se observa un aumento de la conservación aguas arriba del codón de terminación de lectura.

Figura 4.22: Diversas interpretaciones de la supresión del codón de parada. Ver texto para explicación.

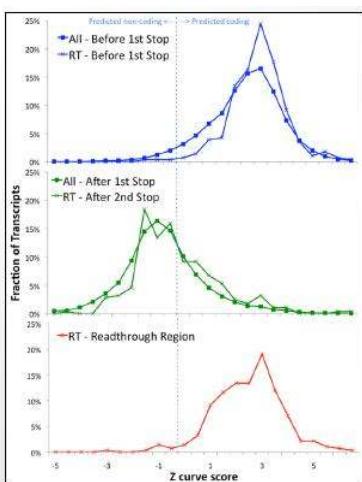


Figura 4.23: Curva Z para Caki. Obsérvese que el uso de codones en la región de lectura es similar al de la región anterior al primer codón de terminación.

Se observa sesgo de codón de parada. El TGAC es la secuencia más frecuente que se encuentra en el codón de parada en la lectura y la menos frecuente en los codones de terminación normales. Se sabe que es un codón de parada “permeable”. El TAAA se encuentra casi universalmente solo en instancias no leídas.

- Números inusualmente altos de repeticiones de GCA observados a través de codones de parada de lectura.
- El aumento de la estructura secundaria del ARN se observa después de la transcripción, lo que sugiere horquillas conservadas evolutivamente.

<sup>3</sup> Kelis M, Patterson N, Endrizzi M, Birren B, Lander E. S. 2003. Secuenciación y comparación de especies de levaduras para identificar genes y elementos reguladores. Ciencia. 423:241—254.

<sup>4</sup> Abrazadera M et al. 2007. Distinguir genes codificantes y no codificantes de proteínas en el genoma humano. PNAS. 104:19428 —19433.

---

This page titled [4.5: Firmas de codificación de proteínas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [4.5: Protein-Coding Signatures](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.6: Firmas génicas de microARN (miARN)

Un ejemplo de regiones genómicas funcionales sujetas a altos niveles de conservación son las secuencias que codifican microARN (miARN). Los miARN son moléculas de ARN que se unen a secuencias complementarias en la región 3' no traducida de moléculas de ARNm dirigidas, causando silenciamiento génico. ¿Cómo encontramos las firmas evolutivas para los genes de miARN y sus dianas, y podemos utilizarlas para obtener nuevos conocimientos sobre sus funciones biológicas? Veremos que esta es una tarea desafiante, ya que los miARN dejan una señal evolutiva altamente conservada pero muy sutil.

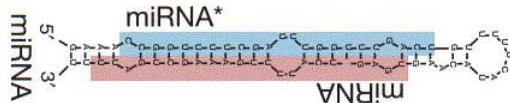


Figura 4.24: La estructura en horquilla de un microARN. Tenga en cuenta que miRNA\* denota la hebra en el lado opuesto de la horquilla, que tiene la misma secuencia que las moléculas de ARNm que son suprimidas por el miARN. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

### Desafío Computacional

Predecir la ubicación de los genes de miARN y sus objetivos es un problema computacionalmente desafiante. Podemos buscar regiones “horquilla”, donde encontramos secuencias de nucleótidos que son complementarias entre sí y predicen una estructura de horquilla. Pero de 760,355 horquillas similares a miARN que se encuentran en la célula, solo 60-100 eran verdaderos miARN. Entonces, para hacer cualquier prueba que nos dé regiones estadísticamente probables de ser miARN, necesitamos una prueba con 99.99% de especificidad.

La Figura 4.25 es un ejemplo del patrón de conservación para genes de miARN. Se pueden observar las dos estructuras en horquilla conservadas en las regiones roja y azul, con una región de baja conservación en el medio. Este patrón es característico de los miARN.



Figura 4.25: Patrón de conservación característico de los miARN. El número de asteriscos debajo de un nucleótido indica el número de especies donde se conserva. Las regiones azul y rojo altamente conservadas representan las cadenas complementarias del miARN, como en la figura 4.24. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Al analizar las características evolutivas y estructurales específicas del miARN, podemos usar combinaciones de estas características para seleccionar regiones de miARN con un enriquecimiento >4.500 veces en comparación con horquillas aleatorias. Los siguientes son ejemplos de características que ayudan a seleccionar los miARN:

- Los miARN se unen a motivos diana altamente conservados en la UTR 3'
- Los miARN se pueden encontrar en intrones de genes conocidos

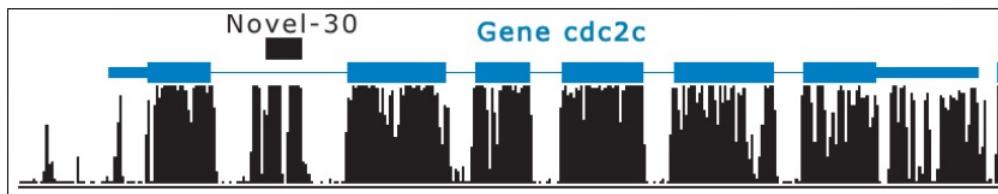


Figura 4.26: Nuevo miARN en intrón© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

- Los miARN tienen preferencia por la cadena positiva del ADN y por los factores de transcripción
- Los miARN normalmente no se encuentran en elementos exónicos y repetitivos del genoma (contraejemplo en la Figura 4.29).
- Los nuevos miARN pueden agruparse con miARN conocidos, especialmente si están en la misma familia o tienen un origen común

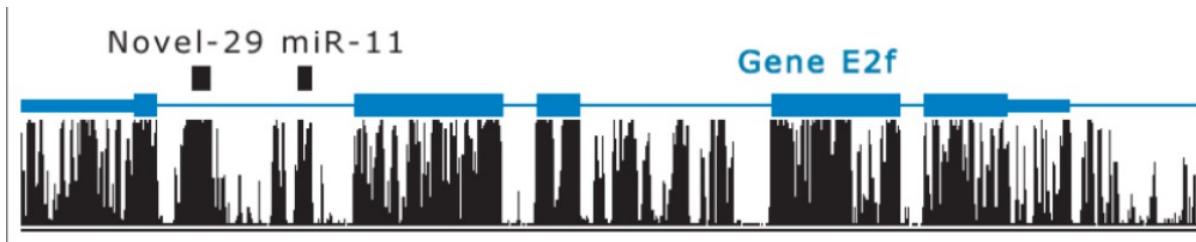


Figura 4.27: Los miARN novedosos y conocidos agrupados. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Estas características de las regiones codificadoras de miARN se pueden agrupar en familias estructurales, lo que permite construir clasificadores basados en ARN conocidos en cada familia. Las consideraciones energéticas para la estructura del ARN pueden ser utilizadas para apoyar esta clasificación en familias. Dentro de cada familia, se produce la conservación ortóloga (genes en diferentes especies para una misma función con el gen ancestral común) y la conservación paráloga (genes duplicados dentro de una misma especie que evolucionaron para servir a diferentes funciones).

### Evolutiva

- Correlación con el perfil de conservación
- MFE del pliegue de consenso
- Índice de conservación de la estructura

### Estructural

- Estabilidad de horquilla (puntuación z MFE)
- Número de bucles asimétricos
- Número de bucles simétricos

Podemos combinar varias características en una prueba usando un árbol de decisión, como se ilustra en la Figura 4.28. En cada nodo del árbol, se aplica una prueba que determina qué rama se seguirá a continuación. El árbol se recorre comenzando desde la raíz hasta que se alcanza un nodo terminal, momento en el que el árbol emitirá una clasificación. Un árbol de decisión puede ser entrenado usando un cuerpo de subsecuencias genómicas clasificadas, después de lo cual se puede usar para predecir si las nuevas subsecuencias son miARN o no. Además, muchos árboles de decisión se pueden combinar en un “bosque aleatorio”, donde se entrena varios árboles de decisión. Cuando se necesita clasificar una nueva secuencia de nucleótidos, cada árbol vota sobre si es o no un miARN, y luego se agregan los votos para determinar la clasificación final.

La aplicación de esta técnica al genoma de la mosca mostró 101 horquillas por encima del punto de corte de 0.95, redescubriendo 60 de 74 miARN conocidos, prediciendo 24 nuevos miARN que fueron validados experimentalmente, y encontrando 17 candidatos adicionales que mostraron evidencia de diversa función.

## Genes de miARN inusuales

Se encontraron las siguientes cuatro “sorpresas” al observar genes específicos de miARN:

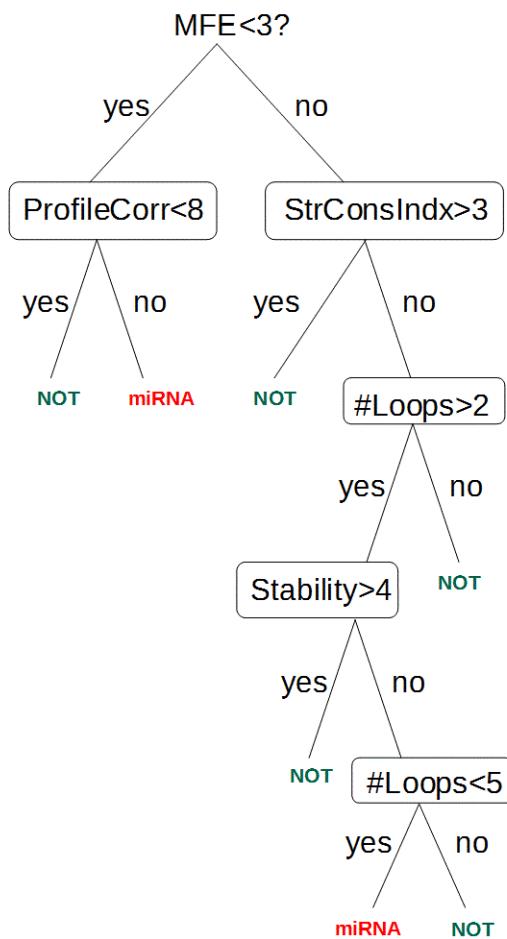


Figura 4.28: Un posible árbol de decisión para la detección de miARN. Las características utilizadas en este árbol son la energía mínima libre, correlación del perfil de conservación, índice de conservación de la estructura, número de bucles y estabilidad.

Sorpresa 1 Ambas hebras podrían ser expresadas y funcionales. Por ejemplo, en el gen miR-Iab-4, la expresión de las cadenas sentido y antisentido se ve en distintos dominios embrionarios. Ambas hebras puntuán > 0.95 para la predicción de miARN.

Sorpresa 2 Algunos miARN pueden tener múltiples extremos 5' para un solo brazo de miARN, lo que da evidencia de un sitio de inicio impreciso. Esto podría dar lugar a múltiples productos maduros, cada uno potencialmente con sus propios objetivos funcionales.

Sorpresa 3 Las regiones de miRNA\* de alta puntuación (el brazo estelar es complementario a la secuencia real de miARN) están muy altamente expresadas, dando lugar a regiones del genoma que están altamente expresadas y contienen elementos funcionales.

Sorpresa 4 Se ha demostrado que tanto miR—10 como miR-10\* son reguladores Hox muy importantes, lo que lleva a la predicción de que los miRNAs podrían ser “reguladores maestros de Hox”. Las páginas 10 y 11 del primer conjunto de conferencias 5 diapositivas muestran la importancia de los miARN que forman una red de regulación para diferentes genes Hox.

### Ejemplo: Reexamen de genes codificadores de proteínas 'dudosos'

Dos genes, CG31044 y CG33311 fueron rechazados independientemente porque sus patrones de conservación no coincidieron con los característicos de las firmas evolutivas de una proteína (ver Sección 4.5). Se identificaron como miARN precursores con base en propiedades genómicas y altos niveles de expresión (Lin et al.). Este es un raro ejemplo de miARN que se encuentra en secuencias previamente exónicas e ilustra el desafío de identificar firmas evolutivas de miARN.

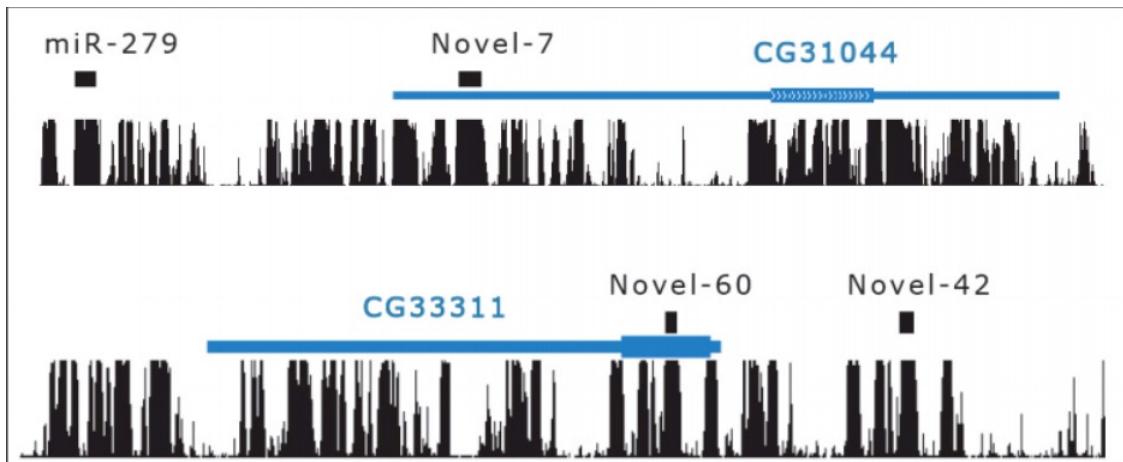


Figura 4.29: Anotaciones y niveles de transcripción existentes en regiones codificadoras de proteínas 'dudosas'. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

This page titled [4.6: Firmas génicas de microARN \(miARN\)](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [4.6: microRNA \(miRNA\) Gene Signatures](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.7: Motivos Regulatorios

Otra clase de elemento funcional que está altamente conservado en muchos genomas contiene motivos reguladores. Un motivo regulador es una secuencia altamente conservada de nucleótidos que ocurre muchas veces a lo largo del genoma y cumple alguna función reguladora. Por ejemplo, estos motivos pueden caracterizar potenciadores, promotores u otros elementos genómicos.

```

D.mel  CAGCT - -AGCC - AACTCTC TAATTACGACTAAGTC - CAAGTC
D.sim  CAGCT - -AGCC - AACTCTC TAATTACGACTAAGTC - CAAGTC
D.sec  CAGCT - -AGCC - AACTCTC TAATTACGACTAAGTC - CAAGTC
D.yak  CAGC - -TAGCC - AACTCTC TAATTACGACTAAGTC - CAAGTC
D.ere  CAGCGGTGCCAAACTCTC TAATTACGACCAAGTC - CAAGTC
D.anu  CACTAGTTCCCTAGGCACTC TAATTACAAGTTAGTCTCTAGAG
***   *   *   *   *   *   *   *   *   *   *   *   *   *   *

```

Figura 4.30: TAATTA es un hexámero que aparece como un elemento conservado a lo largo del genoma en muchos elementos funcionales diferentes, incluyendo aquí. Es un ejemplo de un motivo regulatorio.

### Detectar computacionalmente motivos reguladores

Se han desarrollado métodos computacionales para medir la conservación de motivos reguladores en todo el genoma y encontrar nuevos motivos no anotados de novo. Los motivos conocidos a menudo se encuentran en regiones con alta conservación, por lo que podemos aumentar nuestro poder de prueba probando para la conservación y luego encontrando firmas para motivos regulatorios.

Evaluar el patrón de conservación para motivos conocidos versus el “modelo nulo” de regiones sin motivos da la siguiente firma:

Conservación dentro de:	Gal4 (región de motivo conocida)	Controles
Todas las regiones intergénicas	13%	2%
Intergénico: codificación	13%: 3%	2%: 7%
Upstream: downstream	12:0	1:1

Entonces, como podemos ver, las regiones con motivos reguladores muestran un grado mucho mayor de conservación en regiones intergénicas y aguas arriba del gen de interés.

Para descubrir nuevos motivos, podemos utilizar el siguiente pipeline:

- Elija un motivo “semilla” que consta de dos grupos de tres caracteres no degenerados con un espacio de tamaño variable en el medio.
- Usar una relación de conservación para clasificar los motivos de semillas
- Expandir los motivos de las semillas para llenar las bases alrededor de las semillas usando un algoritmo de escalada de colinas.
- Cluster para eliminar redundancia.

Descubrir motivos y realizar agrupamientos ha llevado al descubrimiento de muchas clases de motivos, como motivos específicos de tejido, motivos específicos de función y módulos de motivos cooperantes.

### Instancias Individuales de Motivos Regulatorios

Para buscar regiones de motivo esperadas, primero podemos calcular una puntuación de ramificación-longitud para una región sospechosa de ser un motivo regulador, y luego usar esta puntuación para darnos un nivel de confianza de cuán probable es que algo sea un motivo real.

La puntuación de longitud de rama (BLS) suma la evidencia de un motivo dado sobre las ramas de un árbol filogenético. Dado el patrón de presencia o ausencia de un motivo en cada especie en el árbol, esta puntuación evalúa la longitud total de la rama del subárbol que conecta las especies que contienen el motivo. Si todas las especies tienen el motivo, el BLS es 100%. Tenga en cuenta que las especies más distamente relacionadas reciben puntuaciones más altas, ya que abarcan una distancia evolutiva más larga. Si un motivo predicho ha abarcado un marco de tiempo evolutivo tan largo, es probable que sea un elemento funcional en lugar de solo una región conservada por casualidad aleatoria.

Para crear un modelo nulo, podemos elegir motivos de control. Los motivos modelo nulos deben elegirse para que tengan la misma composición que el motivo original, para que no sean demasiado similares entre sí y que sean diferentes de los motivos conocidos. Podemos obtener una puntuación de confianza comparando la fracción de instancias de motivo para controlar motivos en una puntuación BLS dada.

---

This page titled [4.7: Motivos Regulatorios](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [4.7: Regulatory Motifs](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.8: Lectura adicional

1. Para más información sobre los cálculos de restricciones y la identificación, consulte “Un mapa de alta resolución de restricción evolutiva humana usando 29 mamíferos” de Lindblad-Toh et. al.
2. Para más información sobre la lectura traduccional y la firma evolutiva, consulte “Revisiting the protein-coding gene catalog of Drosophila melanogaster using 12 fly genomes” de Lin et. al.

---

This page titled [4.8: Lectura adicional](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **4.8: Further Reading** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 4.9: Herramientas y Técnicas

1. Para el alineamiento de secuencias de proteínas, consulte <http://mafft.cbrc.jp/alignment/software/>.
2. Para la predicción de genes mediante desfases de marco en procariotas, ver GeneTack.

---

4.9: Herramientas y Técnicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 4.9: Tools and Techniques has no license indicated.

## Bibliografía

[1] Joseph Felsenstein. Árboles evolutivos a partir de secuencias de ADN: Un enfoque de máxima verosimilitud. Revista de Evolución Molecular, 17:368 —376, 1981. 10.1007/BF01734359.

---

Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 5: Ensamblaje del Genoma y Alineación del Genoma

- 5.1: Introducción
- 5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso
- 5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas
- 5.4: Alineación del Genoma Completo
- 5.5: Alineación regional basada en genes
- 5.6: Mecanismos de Evolución Genómica
- 5.7: Duplicación del genoma completo
- 5.8: Recursos adicionales y bibliografía

#### Bibliografía

---

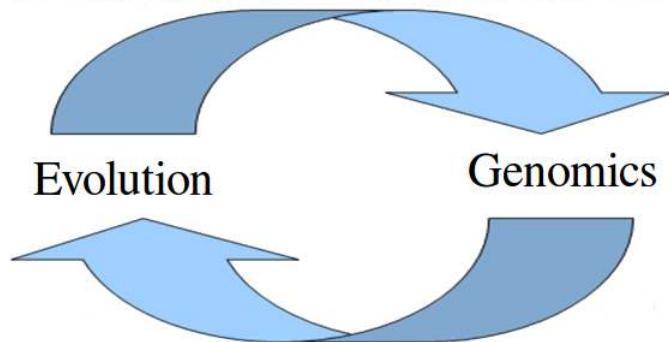
This page titled [5: Ensamblaje del Genoma y Alineación del Genoma](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 5.1: Introducción

En el capítulo anterior, vimos la importancia del análisis genómico comparativo para descubrir elementos funcionales. En la “parte IV” de este libro, veremos cómo podemos utilizar la genómica comparada para estudiar la evolución genética entre especies e individuos. En ambos casos, sin embargo, asumimos que teníamos acceso a genomas completos y alineados en múltiples especies.

En este capítulo, estudiaremos los desafíos del ensamblaje del genoma y la alineación del genoma completo que son los cimientos de las metodologías de genómica comparativa del genoma completo. Primero, estudiaremos los principios algorítmicos centrales que subyacen a muchos de los métodos de ensamblaje del genoma más populares disponibles en la actualidad. En segundo lugar, estudiaremos el problema de la alineación del genoma completo, que requiere comprender los mecanismos de reordenamiento del genoma (por ejemplo, duplicación segmentaria y otras translocaciones). Los dos problemas del ensamblaje del genoma y la alineación del genoma completo son similares en naturaleza, y cerramos discutiendo algunos de los paralelismos entre ellos.

Part I: Using evolution to characterize genomic functional elements



Part II: Using genomic features to discover mechanisms of evolution

Figura 5.1: Podemos usar firmas evolutivas para encontrar elementos funcionales genómicos, y a su vez podemos estudiar mecanismos de evolución observando patrones de variación y cambio genómico.

This page titled [5.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **5.1: Introduction** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso

Muchas áreas de investigación en biología computacional se basan en la disponibilidad de datos completos de la secuencia del genoma completo. Sin embargo, el proceso para secuenciar un genoma completo no es trivial y un área de investigación activa. El problema radica en el hecho de que las tecnologías actuales de secuenciación genómica no pueden leer continuamente de un extremo de una larga secuencia genómica al otro; solo pueden secuenciar con precisión pequeñas secciones de pares de bases (que van de 100 a unos pocos miles, dependiendo del método), llamadas lecturas. Por lo tanto, para construir una secuencia de millones o miles de millones de pares de bases (como el genoma humano), los biólogos computacionales deben encontrar formas de combinar lecturas más pequeñas en secuencias de ADN continuas más grandes. Primero, examinaremos aspectos de la configuración experimental para el enfoque de superposición, diseño y consenso, y luego avanzaremos hacia el aprendizaje sobre cómo combinar lecturas y aprender información de ellas.

### Configuración del experimento

El primer reto que se debe abordar a la hora de poner en marcha este experimento es que necesitamos comenzar con muchas copias de cada cromosoma para poder utilizar este enfoque. Este número es del orden de  $10^5$ . Es importante señalar que la forma en que obtengamos estas copias es muy importante y afectará nuestros resultados más adelante ya que muchas de las comparaciones que hagamos dependerán de datos consistentes. La primera forma en que podemos pensar para obtener tantos datos es amplificar un genoma dado. Sin embargo, la amplificación hace daño lo que arrojará nuestros algoritmos en pasos posteriores y causará peores resultados. Otro método posible sería congrudear el genoma para obtener muchas copias de cada cromosoma. Si buscas deshacerte del polimorfismo, esta puede ser una buena técnica, pero también perdemos datos valiosos de los sitios polimórficos cuando nos cruzamos. Un método sugerido para obtener estos datos es usar un individuo, aunque el organismo tendría que ser bastante grande. También podríamos usar técnicas como la progenie de uno o progenie de dos para obtener la menor cantidad posible de versiones de cada cromosoma. Esto conseguirá una alta profundidad de secuenciación en cada cromosoma, que es la razón por la que queremos que todos los cromosomas sean lo más similares posible.

A continuación, veamos cómo podríamos decidir sobre nuestras longitudes de lectura dada la tecnología actual. Mirando (Figura 5.2), podemos ver que se debe hacer un análisis costo-beneficio para decidir qué plataforma usar en un proyecto determinado. Con la tecnología actual, comúnmente usamos HiSeq2500 con una longitud de lectura de aproximadamente 250, aunque esto está cambiando rápidamente.

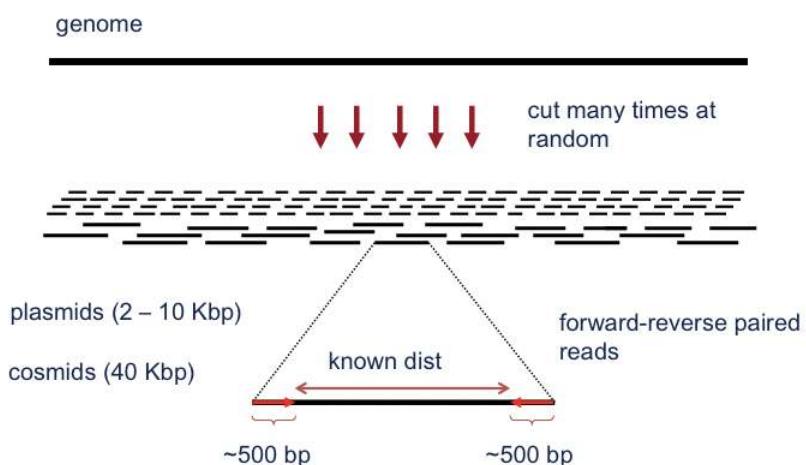
<b>capability</b>	<b>platform</b>	<b>read length</b>	<b>scale</b>	<b>cost</b>	<b>other</b>
short reads	HiSeq 2500	125	high	medium	high output mode
longer reads	HiSeq 2500	250	high	medium	rapid run mode
	MiSeq	300	low	high	
cheap human reads	HiSeq X	150	high	low	human only min 10 purchase

Figura 5.2: Aquí hay un vistazo rápido a algunas plataformas que se pueden usar para leer genomas.

Por último, veamos algunas secuencias que causan problemas al usar plataformas con lecturas cortas. Las secuencias con alto contenido de GC (por ejemplo, GGCGCGATC), bajo contenido de GC (por ejemplo, AAATAATCAA) o baja complejidad (por ejemplo, ATATATA) pueden causar problemas con lecturas cortas. Esta sigue siendo un área activa de investigación, pero algunas explicaciones posibles incluyen el deslizamiento de la polimerasa y la desnaturización del ADN con demasiada facilidad o no lo suficientemente fácil.

En esta sección se examinará uno de los métodos tempranos más exitosos para ensamblar computacionalmente un genoma a partir de un conjunto de lecturas de ADN, llamado secuenciación de escopeta (Figura 5.3). La secuenciación de escopeta implica cortar aleatoriamente múltiples copias del mismo genoma en muchos fragmentos pequeños, como si el ADN fuera disparado con una escopeta. Típicamente, el ADN se fragmenta realmente usando sonicación (breves ráfagas de un ultrasonido) o una enzima dirigida diseñada para escindir el genoma en motivos de secuencia específicos. Ambos métodos se pueden ajustar para crear fragmentos de diferentes tamaños.

Después de que el ADN ha sido amplificado y fragmentado, se utiliza la técnica desarrollada por Frederick Sanger en 1977 llamada secuenciación de terminación de cadena (también llamada secuenciación de Sanger) para secuenciar los fragmentos. En resumen, los fragmentos son extendidos por la ADN polimerasa hasta que se incorpora un didesoxinucleotifosfato; estos nucleótidos especiales provocan la terminación de la extensión de un fragmento. Por lo tanto, la longitud del fragmento se convierte en un proxy para donde se agregó un ddNTP dado en la secuencia. Se pueden ejecutar cuatro reacciones separadas, cada una con un ddNTP diferente (A, G, C, T) y luego ejecutar los resultados en un gel para determinar el orden relativo de las bases. El resultado son muchas secuencias de bases con puntuaciones de calidad por base correspondientes, lo que indica la probabilidad de que cada base haya sido llamada correctamente. Los fragmentos más cortos se pueden secuenciar completamente, pero los fragmentos más largos solo se pueden secuenciar en cada uno de sus extremos ya que la calidad disminuye significativamente.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.3: La secuenciación de escopeta implica cortar aleatoriamente un genoma en pequeños fragmentos para que puedan ser secuenciados y luego reensamblarlos computacionalmente en una secuencia continua.

después de aproximadamente 500-900 pares de bases. Estas lecturas de extremos pareados se denominan pares de relaciones de pareja. En el resto de esta sección, discutimos cómo usar las lecturas para construir secuencias mucho más largas, hasta el tamaño de cromosomas completos.

### Encontrar lecturas superpuestas

Para combinar los fragmentos de ADN en segmentos más grandes, debemos encontrar lugares donde dos o más lecturas sobre- lap, es decir, donde la secuencia inicial de un fragmento coincide con la secuencia final de otro fragmento. Por ejemplo, dados dos fragmentos como ACGTTGACCGCATTGCCATA y GACCGCATTGCCATACGCATGCCATACGGCATT, podemos construir una secuencia mayor basada en el solapamiento: ACGTTGACCGCATTGCCATACGCATGCCATACGGCATT (Figura 5.4).

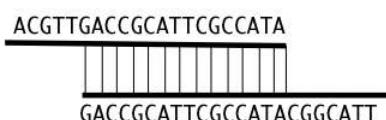


Figura 5.4: Construcción de una secuencia a partir de la superposición de lectura

Un método para encontrar secuencias coincidentes es el algoritmo de programación dinámica de Needleman-Wunsch, que se discutió en el capítulo 2. El método Needleman-Wunsch no es práctico para el ensamblaje del genoma, sin embargo, ya que necesitaríamos realizar millones de alineamientos de pares, cada uno tomando  $O(n^2)$  tiempo, para construir un genoma completo a partir de los fragmentos de ADN.

Un mejor enfoque es usar el algoritmo BLAST (discutido en el capítulo 3) para hash todos los k-mers (secuencias únicas de longitud k) en las lecturas y encontrar todas las ubicaciones donde dos o más lecturas tienen uno de los k-mers en común. Esto nos permite lograr  $O(k^n)$  eficiencia en lugar de  $O(n^2)$  comparaciones por parejas. k puede ser cualquier número menor que el tamaño de las lecturas, pero varía dependiendo de la sensibilidad y especificidad deseadas. Ajustando la longitud de lectura para abarcar las regiones repetitivas del genoma, podemos resolver correctamente estas regiones y acercarnos mucho al ideal de un genoma completo y continuo. Un ensamblador popular de superposición-diseño-consenso llamado Arachne usa k = 24 [2].

Dados los k-mers coincidentes, podemos alinear cada una de las lecturas correspondientes y descartar cualquier coincidencia que sea inferior al 97% similar. No requerimos que las lecturas sean idénticas ya que permitimos la posibilidad de errores de secuenciación y heterocigosis (es decir, un organismo diploide como un ser humano puede tener dos variantes diferentes en un sitio polimórfico).

### Fusionando lecturas en cóntigs

Utilizando las técnicas descritas anteriormente para encontrar superposiciones entre fragmentos de ADN, podemos juntar segmentos más grandes de secuencias continuas llamadas *cóntigs*. Una forma de visualizar este proceso es crear una gráfica en la que todos los nodos representen lecturas, y los bordes representan superposiciones entre las lecturas (Figura 5.5). Nuestra gráfica tendrá *superposición transitiva*; es decir, algunos bordes conectarán nodos dispares que ya están conectados por nodos intermedios. Al eliminar las superposiciones transitoriamente inferidas, podemos crear una cadena de lecturas que han sido ordenadas para formar un cóntigo más grande. Estas transformaciones gráficas se discuten con mayor profundidad en la sección 5.3.1 a continuación. Para entender mejor el tamaño de los cóntigos, calculamos algo conocido como *N50*. Debido a que las medidas de longitud del cóntigo tienden a ser altamente sensibles al corte del cóntigo más pequeño, *N50* se calcula como la mediana ponderada por longitud. Para un ser humano, *N50* suele estar cerca de 125 kb.

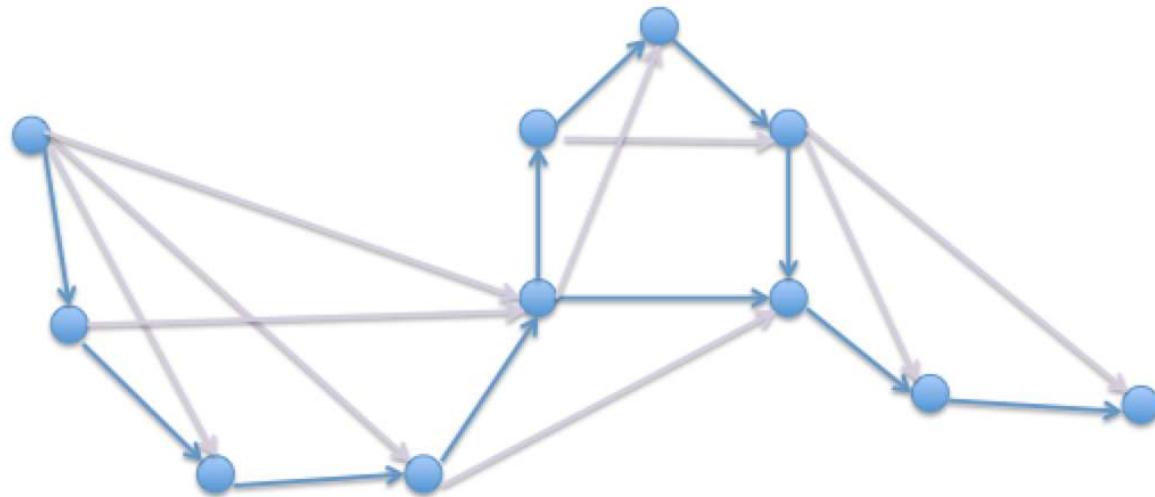
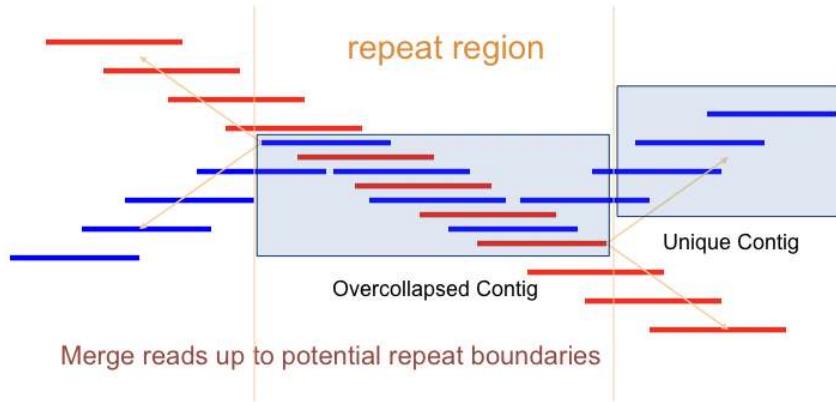


Figura 5.5: Podemos visualizar el proceso de fusión de fragmentos dejando que los nodos en una gráfica representen lecturas y los bordes representen superposiciones. Al eliminar los bordes transitivamente inferibles (los bordes rosados en esta imagen), nos quedan cadenas de lecturas ordenadas para formar cóntigos.

En teoría, deberíamos poder utilizar el enfoque anterior para crear grandes cóntigos a partir de nuestras lecturas siempre y cuando tengamos una cobertura adecuada de la región dada. En la práctica, a menudo nos encontramos con grandes secciones del genoma que son extremadamente repetitivas y como resultado son difíciles de ensamblar. Por ejemplo, no está claro exactamente cómo alinear las dos secuencias siguientes: ATATAT y ATATATAT. Debido al contenido de información extremadamente bajo en el patrón de secuencia, podrían superponerse en cualquier número de formas. Además, estas regiones repetitivas pueden aparecer en

múltiples localizaciones del genoma, y es difícil determinar qué lecturas provienen de qué ubicaciones. Los cónigos formados por estas lecturas ambiguas y repetitivas se denominan cónigos sobrecolapsados.

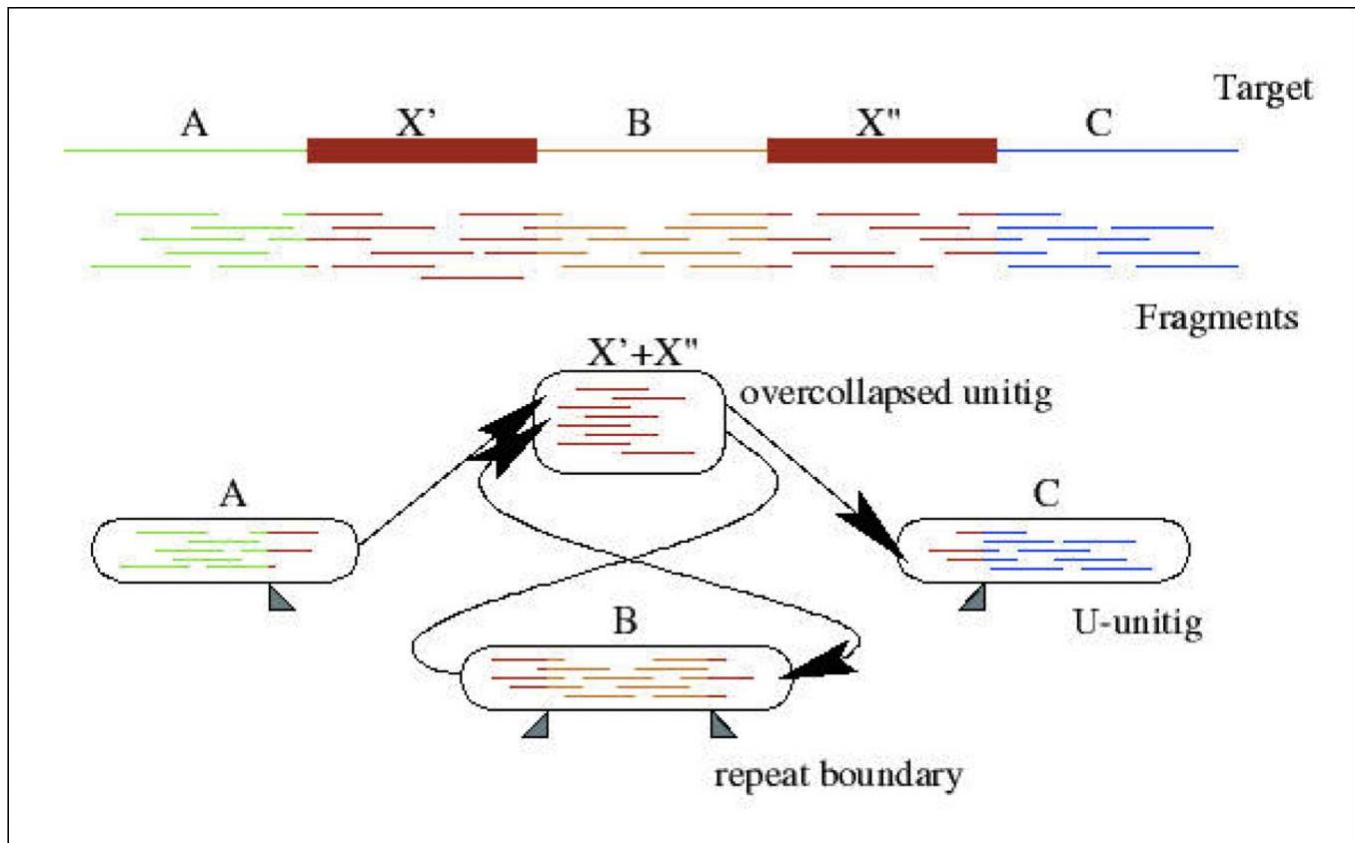
Para determinar qué secciones están sobrecolapsadas, a menudo es posible cuantificar la profundidad de cobertura de los fragmentos que componen cada cónig. Si un cónig tiene significativamente más cobertura que los otros, es probable que sea un candidato para una región sobrecolapsada. Adicionalmente, varios cónigos únicos pueden superponerse a un cónig en la misma ubicación, lo que es otra indicación de que el cónig puede estar sobrecolapsado (Figura 5.6).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.6: Los cónigos sobrecolapsados son causados por regiones repetitivas del genoma que no se pueden extraer entre sí durante la secuenciación. Los patrones de ramificación de alineación que surgen durante el proceso de fusión de fragmentos en cónigos son una fuerte indicación de que una de las regiones puede estar sobrecolapsada.

Después de que los fragmentos han sido ensamblados en cónigos hasta el punto de una posible sección repetida, el resultado es una gráfica en la que los nodos son cónigos, y los bordes son enlaces entre cónigos únicos y cónigos sobrecolapsados (Figura 5.7).

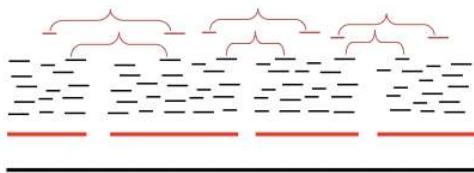


© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.7: En esta gráfica que conecta cōntigos, la región repetida X tiene indegree y outdegree igual a 2. La secuencia objetivo que se muestra en la parte superior se puede inferir a partir de los enlaces en la gráfica.

### Colocación de gráficos de cōntigos en andamios

Una vez que nuestros fragmentos se ensamblan en cōntigos y gráficos de cōntigos, podemos usar los pares de relaciones de pareja más grandes para unir cōntigos en supercōntigs o andamios. Los pares de mate son útiles tanto para orientar los cōntigos como para colocarlos en el orden correcto. Si los pares de relaciones son lo suficientemente largos, a menudo pueden abarcar regiones repetitivas y ayudar a resolver las ambigüedades descritas en la sección anterior (Figura 5.8).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.8: Los pares de mate nos ayudan a determinar el orden relativo de los cōntigos para vincularlos en supercōntigos.

A diferencia de los cōntigos, los supercōntigos pueden contener algunos huecos en la secuencia debido a que los pares de relaciones de pareja que conectan los cōntigos solo se secuencian en los extremos. Como generalmente sabemos cuánto tiempo tiene un par de parejas dado, podemos estimar cuántos pares de bases faltan, pero debido a la aleatoriedad de los cortes en la secuenciación de escopeta, es posible que no tengamos los datos disponibles para completar la secuencia exacta. Llenar cada hueco puede ser extremadamente costoso, por lo que incluso los genomas más completamente ensamblados suelen contener algunos huecos.

## Derivar secuencia consenso

El objetivo del ensamblaje del genoma es crear una secuencia continua, por lo que después de que las lecturas se hayan alineado en cíntigos, necesitamos resolver cualquier diferencia entre ellas. Como se mencionó anteriormente, algunas de las lecturas superpuestas pueden no ser idénticas debido a errores de secuenciación o polimorfismo. A menudo podemos determinar cuándo ha habido un error de secuenciación cuando una base no está de acuerdo con todas las otras bases alineadas a ella. Teniendo en cuenta los puntajes de calidad en cada una de las bases, normalmente podemos resolver estos conflictos con bastante facilidad. Este método de resolución de conflictos se denomina votación ponderada (Figura 5.9). Otra alternativa es ignorar las frecuencias de cada base y tomar como consenso la letra de máxima calidad. En ocasiones, querrás conservar todas las bases que forman un conjunto polimórfico porque puede ser información importante. En este caso, no podríamos utilizar estos métodos para derivar una secuencia consensuada.

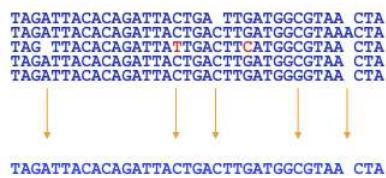


Figura 5.9: Derivamos la secuencia consenso de alineamiento múltiple mediante votación ponderada en cada base.

En algunos casos, no es posible derivar un consenso si, por ejemplo, el genoma es heterocigoto y hay números iguales de dos bases diferentes en una ubicación. En este caso, el ensamblador deberá elegir un representante.

### ¿Sabías?

Dado que el polimorfismo puede complicar significativamente el ensamblaje de genomas diploides, algunos investigadores inducen varias generaciones de endogamia en las especies seleccionadas para reducir la cantidad de heterocigosidad antes de intentar secuenciar el genoma.

En esta sección, vimos un algoritmo para hacer ensamblaje del genoma dadas lecturas. Sin embargo, este algoritmo funciona bien cuando las lecturas tienen una longitud de 500 a 900 bases o más, lo que es típico de la secuenciación de Sanger. Se requieren algoritmos alternos de ensamblaje del genoma es que las lecturas que obtenemos de nuestros métodos de secuenciación son mucho más cortas.

This page titled [5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

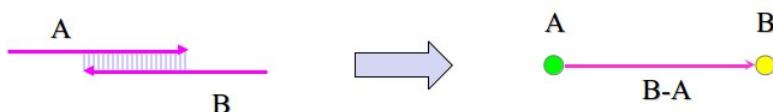
- [5.2: Genome Assembly I- Overlap-Layout-Consensus Approach](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas

La secuenciación de escopeta, que es un método de secuenciación más moderno y económico, da lecturas de alrededor de 100 bases de longitud. La longitud más corta de las lecturas da como resultado muchas más repeticiones de longitud mayor que la de las lecturas. Por lo tanto, necesitamos algoritmos nuevos y más sofisticados para hacer el ensamblaje del genoma correctamente.

### Definición y construcción del gráfico de cuerdas

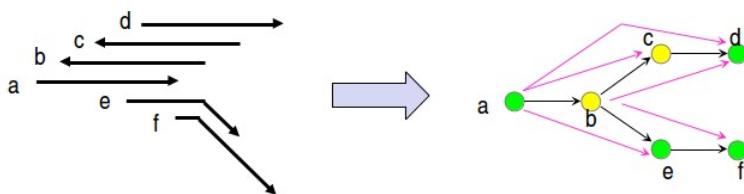
La idea detrás del ensamblaje del gráfico de cuerdas es similar a la gráfica de lecturas que vimos en la sección 5.2.2. En definitiva, estamos construyendo una gráfica en la que los nodos son datos de secuencia y los bordes se superponen, para luego tratar de encontrar la ruta más robusta a través de todos los bordes para representar nuestra secuencia subyacente.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.10: Construcción de un gráfico de cadenas.

A partir de las lecturas que obtenemos de la secuenciación de Shotgun, se construye un gráfico de cadenas agregando un borde para cada par de lecturas superpuestas. Tenga en cuenta que los vértices de la gráfica denotan cruces, y los bordes corresponden a la cadena de bases. Un solo nodo corresponde a cada lectura, y llegar a ese nodo mientras recorre la gráfica equivale a leer todas las bases hasta el final de la lectura correspondiente al nodo. Por ejemplo, en la figura 5.10, tenemos dos lecturas superpuestas A y B y son las únicas lecturas que tenemos. El gráfico de cadena correspondiente tiene dos nodos y dos aristas. Un borde no tiene un vértice en su extremo de cola, y tiene A en su extremo de cabeza. Este borde denota todas las bases en la lectura A. El segundo borde va del nodo A al nodo B, y solo denota las bases en B-A (la parte de lectura B que no se solapa con A). De esta manera, cuando atravesamos los bordes una vez, leemos toda la región exactamente una vez. En particular, observe que no atravesamos el solapamiento de la lectura A y la lectura B dos veces.

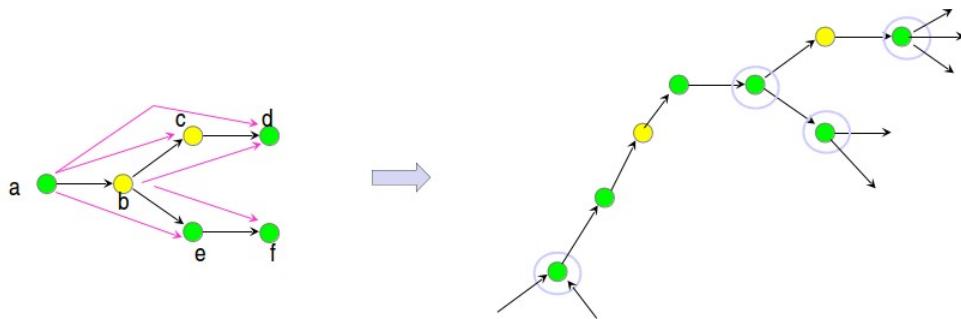


© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.11: Construyendo un gráfico de cadenas 99

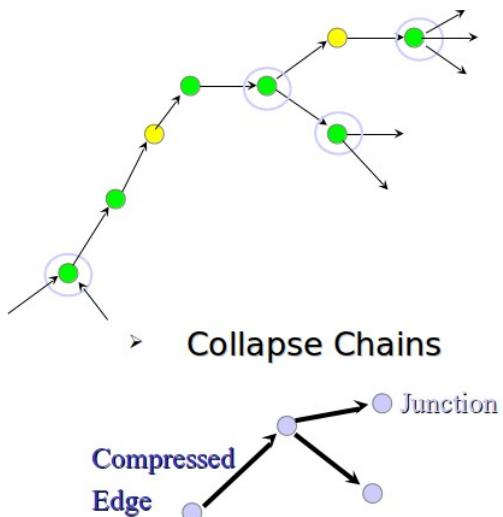
Hay un par de sutilezas en el gráfico de cuerdas (figura 5.11) que deben mencionarse:

- Tenemos dos colores diferentes para los nodos ya que el ADN se puede leer en dos direcciones. Si la superposición es entre las lecturas tal cual, entonces los nodos reciben los mismos colores. Y si el solapamiento es entre una lectura y las bases complementarias de la otra lectura, entonces reciben diferentes colores.
- En segundo lugar, si A y B se superponen, entonces hay ambigüedad en si dibujamos un borde de A a B, o de B a A. Tal ambigüedad necesita resolverse de manera consistente en los cruces causados por repeticiones.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.12: Ejemplo de gráfico de cuerdas sometido a remoción de bordes transitivos.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.13: Ejemplo de gráfico de cadenas en proceso de colapso de cadena.

Después de construir el gráfico de cadenas a partir de lecturas superpuestas, nosotros: -

- Eliminar bordes transitivos: Los bordes transitivos son causados por superposiciones transitivas, es decir, una superposición B se superpone a C de tal manera que A se superpone a C. Hay algoritmos aleatorios que eliminan bordes transitivos en O (E) tiempo de ejecución esperado. En la figura 5.12, se puede ver el ejemplo de eliminación de bordes transitivos.
- Contraer cadenas: Después de eliminar los bordes transitivos, el gráfico que construimos tendrá muchas cadenas donde cada nodo tiene un borde entrante y un borde saliente. Derrumbamos todas estas cadenas a un solo borde. Un ejemplo de ello se muestra en la figura 5.13.

## Flujos y consistencia gráfica

Después de hacer todo lo mencionado anteriormente obtendremos una gráfica bastante compleja, es decir, seguirá teniendo una serie de uniones debido a repeticiones relativamente largas en el genoma en comparación con la longitud de las lecturas. Ahora veremos cómo se pueden utilizar los conceptos de flujos para hacer frente a las repeticiones.

Primero, estimamos el peso de cada borde por el número de lecturas que obtenemos corresponde al borde. Si tenemos el doble del número de lecturas para algún borde que el número de ADN que secuenciamos, entonces es justo suponer que esta región del genoma se repite. Sin embargo, esta técnica por sí misma no es lo suficientemente precisa. De ahí que a veces podamos hacer estimaciones diciendo que el peso de algún borde es  $\geq 2$ , y no asignarle un número particular.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.14: Izquierda: Concepto de resolución de flujo. Derecha: Ejemplo de resolución de flujo

Utilizamos el razonamiento a partir de flujos para resolver tales ambigüedades. Necesitamos satisfacer la restricción de flujo en cada cruce, es decir, el peso total de todos los bordes entrantes debe ser igual al peso total de todos los bordes salientes. Por ejemplo, en la figura 5.14 hay una unión con un borde entrante de peso 1, y dos bordes salientes de peso  $\geq 0$  y  $\geq 1$ . De ahí que podamos inferir que los pesos de los bordes salientes son exactamente iguales a 0 y 1 respectivamente. De esta manera se pueden inferir muchos pesos aplicando iterativamente este mismo proceso a lo largo de toda la gráfica.

## Flujo factible

Una vez que tenemos la gráfica y los pesos de borde, ejecutamos un algoritmo de flujo de costos mínimos en la gráfica. Dado que los genomas más grandes pueden no tener un flujo de costo mínimo único, iterativamente hacemos lo siguiente:

- Agregar penalización  $\epsilon$  a todos los bordes en solución
- Resolver flujo nuevamente - si hay un flujo de costo mínimo alternativo, ahora tendrá un costo menor en relación con el flujo anterior
- Repita hasta que no encontremos nuevos bordes

Después de hacer lo anterior, podremos etiquetar cada borde como uno de los siguientes

- *Requerido*: bordes que formaban parte de todas las soluciones
- *No confiables*: bordes que formaban parte de algunas de las soluciones
- *No requeridos*: bordes que no formaban parte de ninguna solución

## Cómo lidiar con errores de secuenciación

Existen diversas fuentes de errores en el procedimiento de secuenciación del genoma. Los errores son generalmente de dos tipos diferentes, locales y globales.

Los errores locales incluyen inserciones, delecciones y mutaciones. Dichos errores locales se tratan cuando buscamos lecturas superpuestas. Es decir, mientras verificamos si las lecturas se superponen, verificamos si hay superposiciones mientras somos tolerantes hacia los errores de secuenciación. Una vez que hemos calculado las superposiciones, podemos derivar un consenso mediante mecanismos como la eliminación de indels y mutaciones que no son apoyadas por ninguna otra lectura y son contradicidas por al menos 2.

Los errores globales son causados por otros mecanismos como dos secuencias diferentes que se combinan antes de ser leídas, y de ahí obtenemos una lectura que es de diferentes lugares del genoma. Tales lecturas se llaman chimers. Estos errores se resuelven mientras se busca un flujo factible en la red. Cuando el borde correspondiente a la quimera está en uso, la cantidad de flujo que atraviesa este borde es menor en comparación con la capacidad de flujo. De ahí que el borde pueda ser detectado y luego ignorado.

Cada paso del algoritmo se hace lo más robusto y resistente posible a los errores de secuenciación. Y el número de ADN divididos y secuenciados se decide de una manera para que seamos capaces de construir la mayor parte del ADN (es decir, cumplir con alguna garantía de calidad como 98% o 95%).

## Recursos

Algunos ensambladores de genoma populares que utilizan gráficos de cadena se enumeran a continuación

- Euler (Pevzner, 2001/06): Indización → Debruijn grafos → picking paths → consenso
- Valvel (Birney, 2010): Lecturas cortas → genomas pequeños → simplificación → corrección de errores
- ALLPATH (Gnerre, 2011): Lecturas cortas → genomas grandes → saltar datos → incertidumbre

---

This page titled [5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

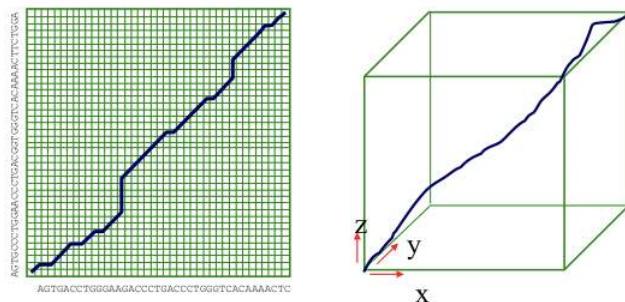
- **5.3: Genome Assembly II- String graph methods** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.4: Alineación del Genoma Completo

Una vez que tenemos acceso a secuencias del genoma completo para varias especies diferentes, podemos intentar alinearlas para inferir el camino que tomó la evolución para diferenciar estas especies. En esta sección discutimos algunos de los métodos para realizar alineaciones de genoma completo entre múltiples especies.

### Alineación global, local y 'glocal'

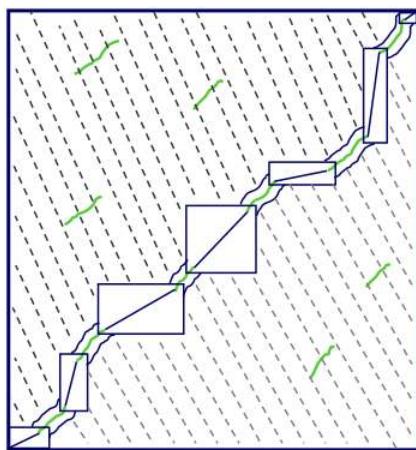
El algoritmo de Needleman-Wunsch discutido en el capítulo 2 es la mejor manera de generar un alineamiento óptimo entre dos o más secuencias genómicas de tamaño limitado. A nivel de genomas completos, sin embargo, el límite de tiempo  $O(n^2)$  no es práctico. Además, para encontrar un alineamiento óptimo entre  $k$  especies diferentes, el tiempo para el algoritmo de Needleman-Wunsch se extiende a  $O(n^k)$ . Para genomas que tienen millones de bases de largo, este tiempo de ejecución es prohibitivo (Figura 5.15).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.15: El algoritmo de Needleman-Wunsch para alineaciones de 2 y 3 genomas

Una alternativa es usar una herramienta de alineación local eficiente como BLAST para encontrar todas las alineaciones locales y luego encadenarlas a lo largo de la diagonal para formar alineaciones globales. Este enfoque puede ahorrar una cantidad significativa de tiempo, ya que el proceso de búsqueda de alineaciones locales es muy eficiente, y entonces solo necesitamos realizar el algoritmo de Needleman-Wunsch que consume mucho tiempo en los pequeños rectángulos entre alineaciones locales (Figura 5.16).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

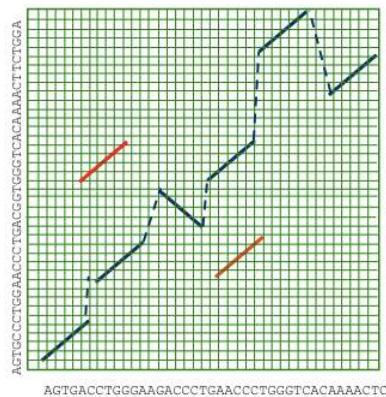
Figura 5.16: Podemos ahorrar tiempo al realizar una alineación global al encontrar primero todas las alineaciones locales y luego encadenarlas a lo largo de la diagonal con programación dinámica restringida.

Otro enfoque novedoso para el alineamiento del genoma completo es extender la búsqueda de alineamiento local para incluir inversiones, duplicaciones y translocaciones. Entonces podemos encadenar estos elementos usando las transformaciones de menor

costo entre secuencias. Este enfoque se denomina comúnmente alineación glocal, ya que busca combinar lo mejor de la alineación local y global para crear la imagen más precisa de cómo evolucionan los genomas a lo largo del tiempo (Figura 5.17).

## Lagan: Encadenamiento de alineaciones locales

LAGAN es un popular kit de herramientas de software que incorpora muchas de las ideas anteriores y se puede utilizar para alineaciones locales, globales, glocales y múltiples entre especies.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.17: La alineación glocal permite la posibilidad de duplicaciones, inversiones y translocaciones.

El algoritmo LAGAN regular consiste en encontrar alineaciones locales, encadenar alineaciones locales a lo largo de la diagonal y luego realizar programación dinámica restringida para encontrar la ruta óptima entre alineaciones locales.

Multi-lagan utiliza el mismo enfoque que el LAGAN regular pero lo generaliza a la alineación de múltiples especies. En este algoritmo, el usuario debe proporcionar un conjunto de genomas y un árbol filogenético correspondiente. Multi- LAGAN realiza alineamiento por pares guiado por el árbol filogenético. Primero compara especies altamente relacionadas, y luego compara iterativamente especies cada vez más distantes.

Shuffle-lagan es una herramienta de alineación glocal que encuentra alineaciones locales, construye un mapa de homología aproximado y luego alinea globalmente cada una de las partes consistentes (Figura 5.18). Para construir un mapa de homología, el algoritmo elige el subconjunto de puntuación máxima de alineaciones locales en función de ciertas penalizaciones de brecha y transformación, que forman una cadena no decreciente en al menos una de las dos secuencias. A diferencia del LAGAN regular, todas las secuencias de alineación local posibles se consideran como pasos en el alineamiento glocal, ya que podrían representar translocaciones, inversiones y translocaciones invertidas, así como secuencias regulares no transformadas. Una vez que se ha construido el mapa de homología aproximado, el algoritmo divide las regiones homólogas en trozos de alineaciones locales que están aproximadamente a lo largo de la misma ruta continua. Finalmente, se aplica el algoritmo LAGAN a cada fragmento para vincular las alineaciones locales mediante programación dinámica restringida.

Al ejecutar Shuffle-lagan u otras herramientas de alineación glocal, podemos descubrir inversiones, translocaciones y otras relaciones homólogas entre diferentes especies. Al mapear las conexiones entre estos rangos traseros, podemos obtener una idea de cómo evolucionó cada especie a partir del ancestro común (Figura 5.19).

---

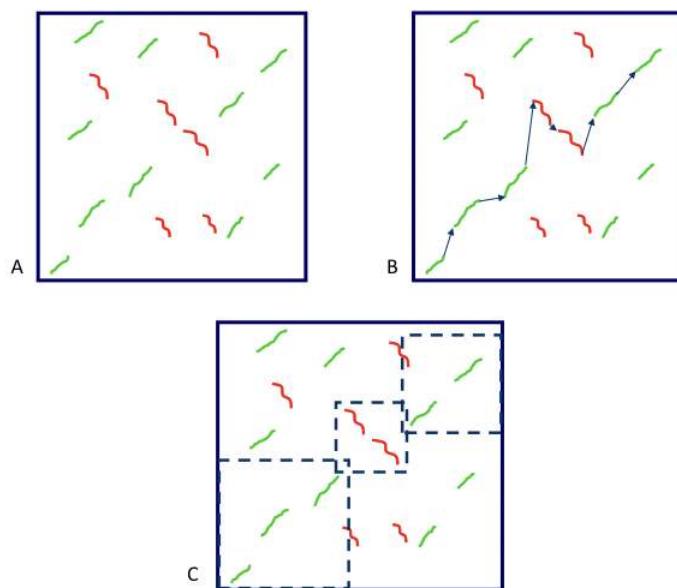
This page titled [5.4: Alineación del Genoma Completo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [5.4: Whole-Genome Alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.5: Alineación regional basada en genes

Una forma alternativa de alinear múltiples genomas ancla segmentos genómicos basados en los genes que contienen, y utiliza la correspondencia de genes para resolver las regiones correspondientes en cada par de especies. Luego se construye un alineamiento a nivel de nucleótidos basado en métodos descritos anteriormente en cada región conservada multiplicada.

Debido a que no todas las regiones tienen correspondencia uno a uno y la secuencia no es estática, esto es más difícil: los genes experimentan divergencia, duplicación y pérdidas y genomas completos sufren reordenamientos. Para ayudar a superar estos desafíos, los investigadores analizan la similitud de aminoácidos de los pares de genes en los genomas y la ubicación de los genes dentro de cada genoma.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.18: Los pasos para ejecutar el algoritmo SLAGAN son A. Encontrar todas las alineaciones locales, B. Construir un mapa de homología aproximada, y C. alinear globalmente las partes consistentes usando el algoritmo LAGAN regular

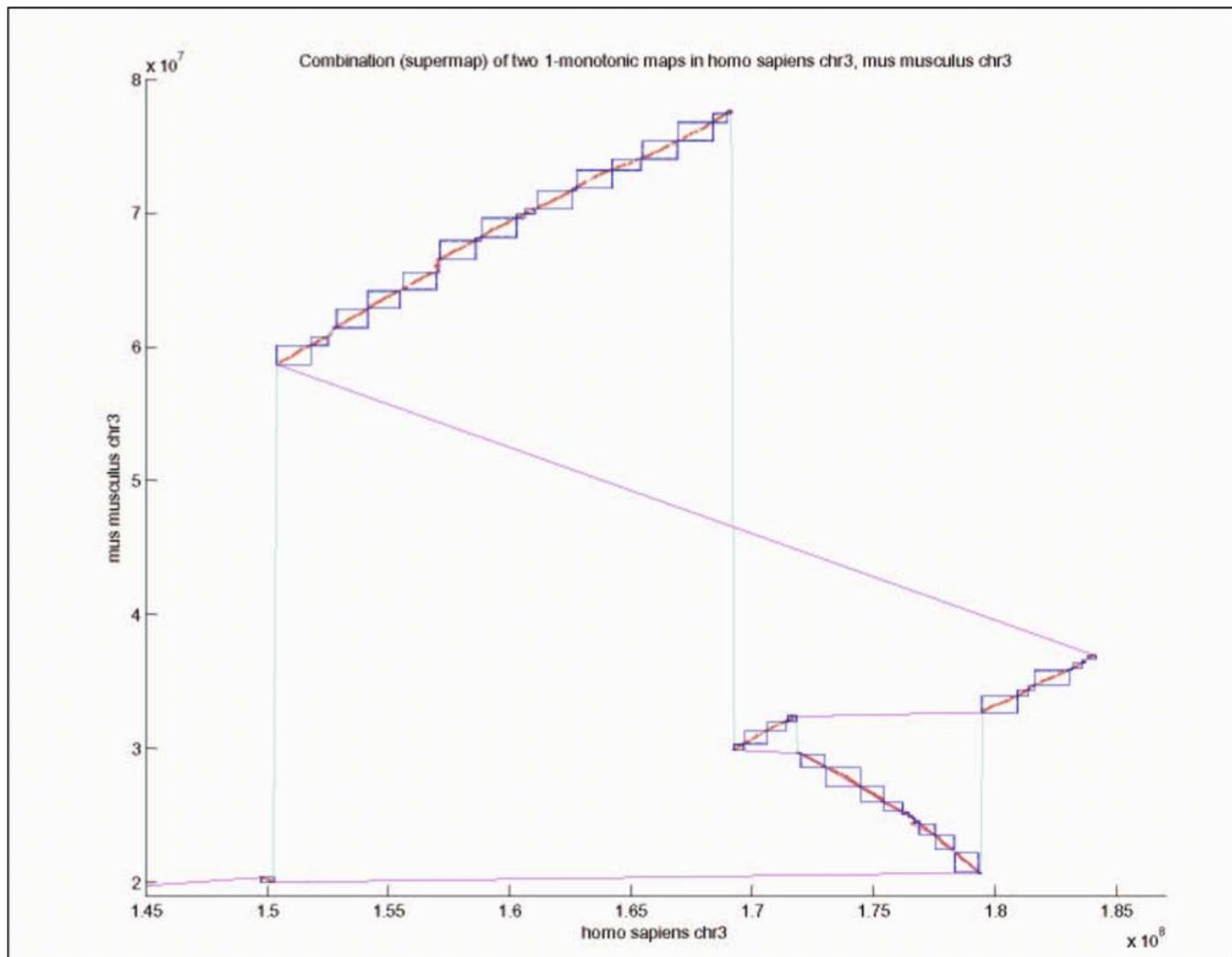
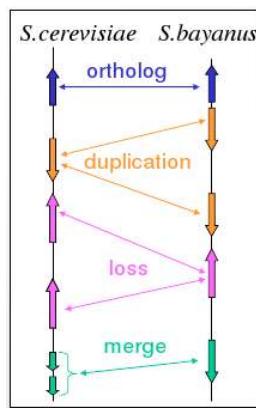


Figura 5.19: Utilizando los conceptos de alineación glocal, podemos descubrir inversiones, translocaciones y otras relaciones homólogas entre diferentes especies como el humano y el ratón.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.20: Gráfica de correspondencia génica de *S. cerevisiae* y *S. bayanus*.

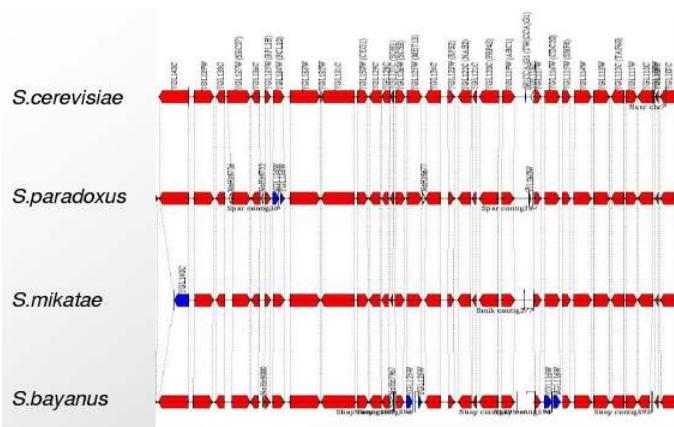
La correspondencia génica puede representarse mediante una gráfica bipartita ponderada con nodos que representan genes con coordenadas y bordes que representan similitud de secuencia ponderada (Figura 5.20). Las relaciones ortólogas son coincidencias uno a uno y las relaciones parálogas son coincidencias de uno a muchos o de muchos a muchos. La gráfica se simplifica primero

eliminando bordes espurios y luego los bordes se seleccionan en base a la información disponible, como bloques de orden génico conservado y similitud de secuencia de proteínas.

El algoritmo Best Unambiguous Subgroups (BUS) se puede utilizar entonces para resolver la correspondencia de genes y regiones. BUS extiende el concepto de mejores aciertos bidireccionales y utiliza refinamiento iterativo con un umbral relativo creciente. Utiliza la conectividad completa de gráficos bipartitos con similitud de aminoácidos integrada e información de orden de genes.

### ¿Sabías?

La gráfica bipartita es una gráfica cuyos vértices se pueden dividir en dos conjuntos disjuntos U y V de tal manera que cada borde conecta un vértice en U a un vértice en V.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.21: Ilustración de la correspondencia génica para el Cromosoma VI de *S.cerevisiae* (250-300pb).

En el ejemplo de una correspondencia génica correctamente resuelta de *S.cerevisiae* con otras tres especies relacionadas, más del 90% de los genes tuvieron una correspondencia uno a uno y se identificaron regiones y familias de proteínas de cambio rápido.

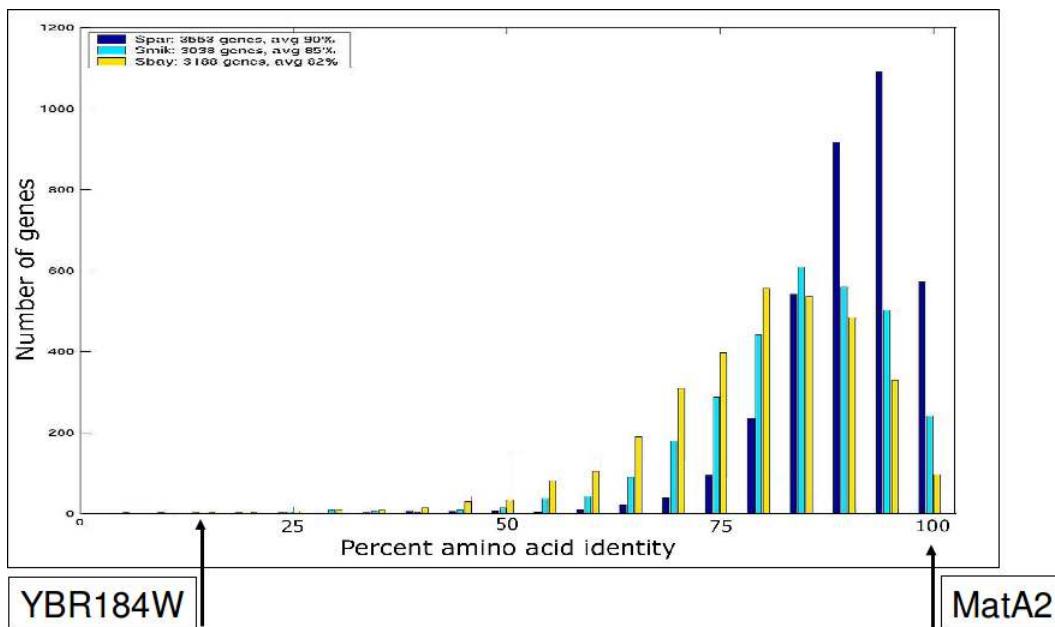
This page titled [5.5: Alineación regional basada en genes](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [5.5: Gene-based region alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.6: Mecanismos de Evolución Genómica

Una vez que tenemos alineamientos de grandes regiones genómicas (o genomas completos) a través de múltiples especies relacionadas, podemos comenzar a hacer comparaciones para inferir las historias evolutivas de esas regiones.

Las tasas de evolución varían entre especies y regiones genómicas. En *S. cerevisiae*, por ejemplo, el 80% de las ambigüedades se encuentran en el 5% del genoma. Los telómeros son secuencias repetitivas de ADN al final de los cromosomas que protegen los extremos de los cromosomas del deterioro. Las regiones teloméricas son inherentemente inestables, tienden a sufrir una rápida evolución estructural, y el 80% de variación corresponde a 31 de las 32 regiones teloméricas. Las familias de genes contenidas dentro de estas regiones como HXT, FLO, COS, PAU e YRF muestran una evolución significativa en número, orden y orientación. Varias secuencias novedosas y codificantes de proteínas se pueden encontrar en estas regiones. Dado que se encuentran muy pocos reordenamientos genómicos aparte de las regiones teloméricas, las regiones de cambio rápido pueden identificarse por expansiones de la familia de proteínas en los extremos cromosómicos.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.22: Vista dinámica de un gen cambiante.

Genes evolucionan a diferentes ritmos. Por ejemplo, como se ilustra en la Figura 5.22, en un extremo, hay YBR184W en la levadura que muestra una conservación de secuencia inusualmente baja y exhibe numerosas inserciones y delecciones entre especies. En el otro extremo se encuentra MatA2, que muestra una conservación perfecta de aminoácidos y nu-cleotide. Las tasas de mutación a menudo también varían según la clasificación funcional. Por ejemplo, las proteínas ribosómicas mitocondriales están menos conservadas que las proteínas ribosómicas.

El hecho de que algunos genes evolucionen más lentamente en una especie frente a otra puede deberse a factores como ciclos de vida más largos. La falta de cambio evolutivo en genes específicos, sin embargo, sugiere que existen funciones biológicas adicionales que son responsables de la presión para conservar la secuencia de nucleótidos. La levadura puede cambiar los tipos de apareamiento cambiando todos sus genes A y α y matA2 es uno de los cuatro genes de tipo apareamiento de levadura (matA2, Matα2, MatA1, Matα1). Su papel podría ser revelado potencialmente por análisis de conservación de nucleótidos.

Los genes de rápida evolución también pueden ser biológicamente significativos. Los mecanismos de cambio rápido de proteínas incluyen:

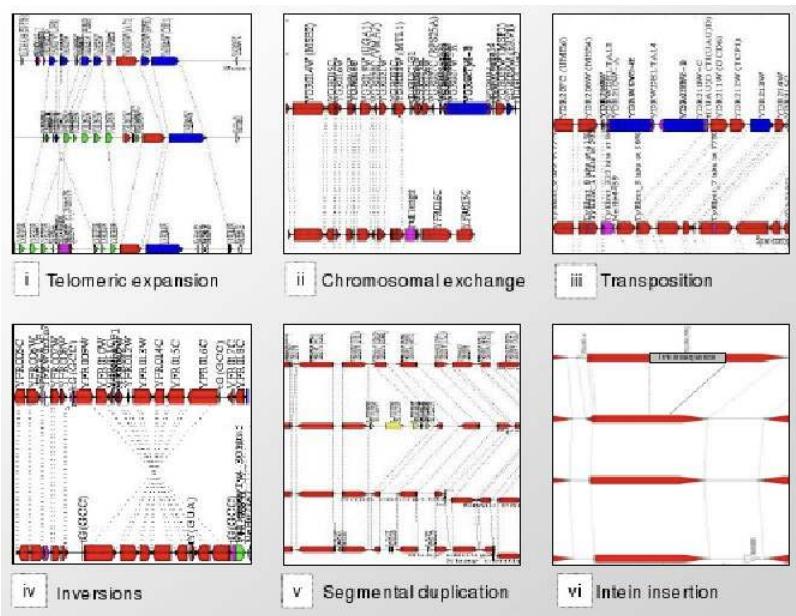
- Creación de dominios proteicos a través de tramos de Glutamina (Q) y Asparagina (N) e interacciones proteína-proteína,
- Desplazamientos compensatorios de fotogramas que permiten la exploración de nuevos marcos de lectura y la lectura/creación de señales de edición de ARN,

- Variaciones de codones de parada y lectura regulada donde las ganancias permiten cambios rápidos y las pérdidas pueden dar como resultado una nueva diversidad
- Las inteínas, que son segmentos de proteínas que pueden retirarse de una proteína y luego volver a unirse a la proteína restante, obtienen ganancias de transferencias horizontales de inteínas autoempalmantes postraduccionales.

Ahora analizamos las diferencias en el contenido de genes entre diferentes especies (*S.cerevisiae*, *S.paradoxus*, *S.mikatae* y *S.bayanus*.) Se puede revelar mucho sobre la pérdida y conversión de genes observando las posiciones de los parálogos a través de especies relacionadas y observando las tasas de cambio de los parálogos. Hay 8-10 genes únicos para cada genoma que están involucrados principalmente con el metabolismo, la regulación y el silenciamiento, y la respuesta al estrés. Además, hay cambios en la dosificación génica con duplicaciones tanto en tandem como en segmentos. Las expansiones de la familia de proteínas también están presentes con 211 genes con correspondencia ambigua. Sin embargo, en general, hay pocos genes novedosos en las diferentes especies.

## Reordenamientos cromosómicos

Estos suelen estar mediados por mecanismos específicos como se ilustra para *Saccharomyces* en la Figura 5.23. [MattFox] Fig11cromovolimageissuperborrosasfasasicanse.dondequiero que se encontrara esto habría sido encontrado, debería ser colocado con un alto



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

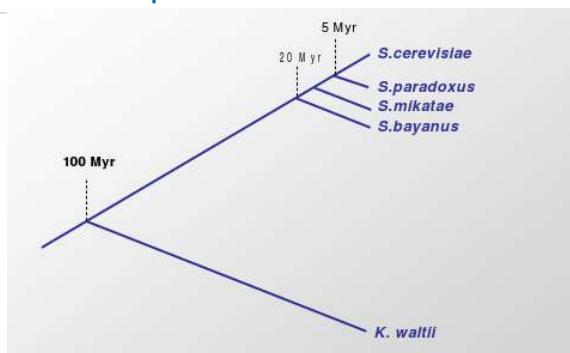
Figura 5.23: Mecanismos de evolución cromosómica.

Las translocaciones a través de genes diferentes a menudo ocurren a través de elementos genéticos transponibles (elementos Ty en levadura, por ejemplo). Las ubicaciones del transposón se conservan con inserciones recientes que aparecen en ubicaciones antiguas y restos de repeticiones terminales largas encontrados en otros genomas. Sin embargo, son evolutivamente activos (por ejemplo, siendo recientes los elementos Ty en la levadura), y normalmente aparecen en un solo genoma. La ventaja evolutiva de tales transposones conservados locacionalmente puede estar en la posibilidad de mediar arreglos reversibles. Las inversiones a menudo están flanqueadas por genes de ARNt en orientación transcripcional opuesta. Esto puede sugerir que se originan a partir de la recombinación entre genes de ARNt.

This page titled [5.6: Mecanismos de Evolución Genómica](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **5.6: Mechanisms of Genome Evolution** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.7: Duplicación del genoma completo



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.24: Retrocediendo en el tiempo evolutivo para *Saccharomyces*.

A medida que trazas especies más atrás en el tiempo evolutivo, tienes la capacidad de hacer diferentes conjuntos de preguntas. En clase, el ejemplo utilizado fue *K. waltii*, que data de unos 95 millones de años antes que *S.cerevisiae* y 80 millones de años antes que *S.bayanus*.

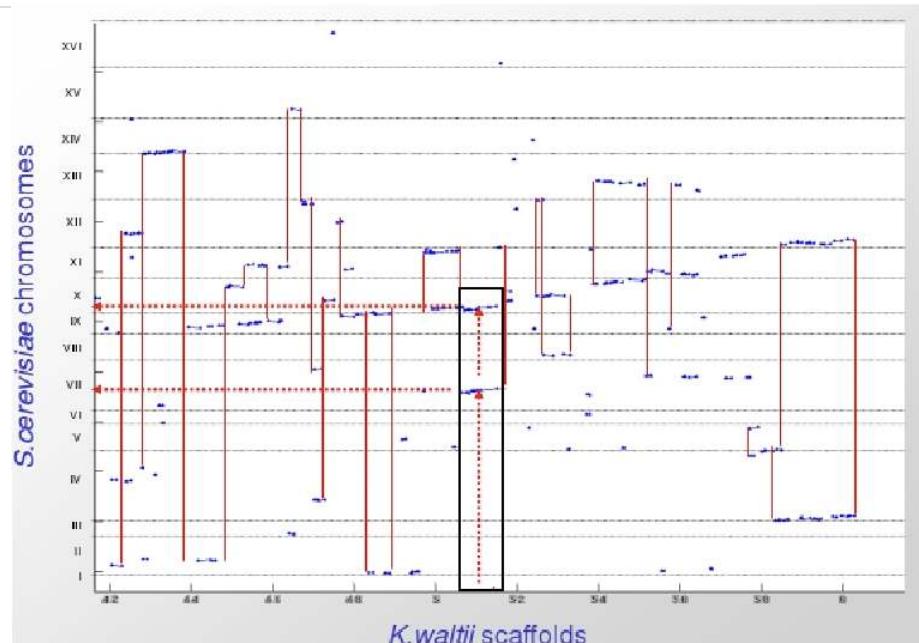
Al observar la gráfica de puntos de los cromosomas de *S.cerevisiae* y los andamios de *K.waltii*, se observó una divergencia a lo largo de la diagonal en el centro de la parcela, mientras que la mayoría de los pares de regiones conservadas presentan una gráfica de puntos con una diagonal clara y recta. Al ver el segmento a mayor aumento (Figura 5.25), parece que los fragmentos hermanos de *S.cerevisiae* se mapean a los correspondientes andamios de *K.waltii*.

Esquemáticamente (Figura 5.26) las regiones hermanas muestran entrelazado génico. En el mapeo duplicado de centromeros, las regiones hermanas pueden reconocerse en base al orden de los genes. Este entrelazado de genes observado proporciona evidencia de duplicación completa del genoma.

This page titled [5.7: Duplicación del genoma completo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

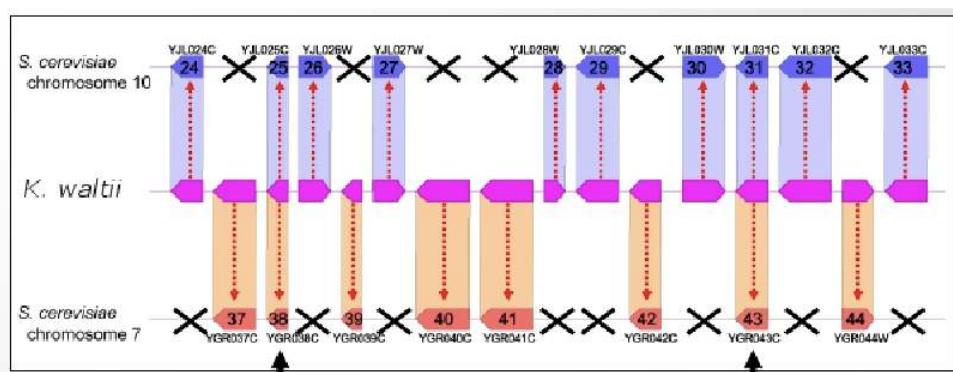
- **5.7: Whole Genome Duplication** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 5.8: Recursos adicionales y bibliografía



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.25: Correspondencia génica para cromosomas de *S.cerevisiae* y andamios de *K.waltii*.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 5.26: Intercalado génico mostrando regiones hermanas en la bibliografía de *K.waltii* y *S.cerevisiae*

This page titled [5.8: Recursos adicionales y bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [5.8: Additional Resources and Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## Bibliografía

1. [1] Base de datos Embl allelectron - exones de cassette.
2. [2] Batzoglou S et al. Arachne: un ensamblador de escopeta de genoma completo. *Genome Res*, 2002.
3. [3] Manolis Kellis. Diapositivas 04: Genómica comparada i. 21 de septiembre de 2010.
4. [4] Manolis Kellis. Diapositivas de conferencias 05.1: Genómica comparada ii. 23 de septiembre de 2010.
5. [5] Manolis Kellis. Diapositivas de conferencias 05.2: Genómica comparada iii, evolución. 25 de septiembre de 2010.
6. [6] Nikolaus Rajewsky Kevin Chen. Evolución de la regulación génica por factores de transcripción y micrornas. *Nature Reviews Genetics*, 2007.

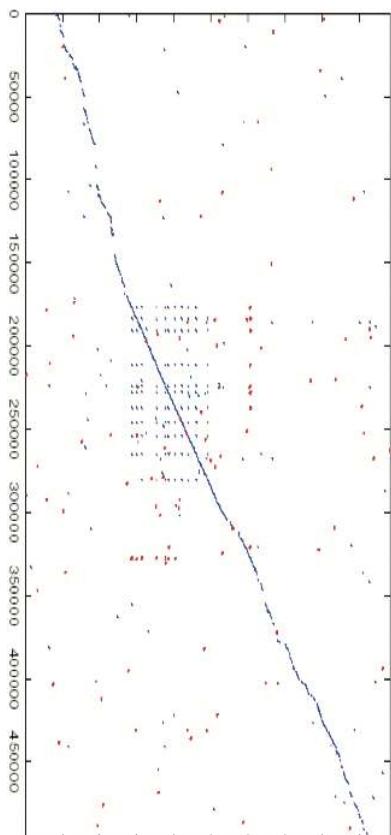
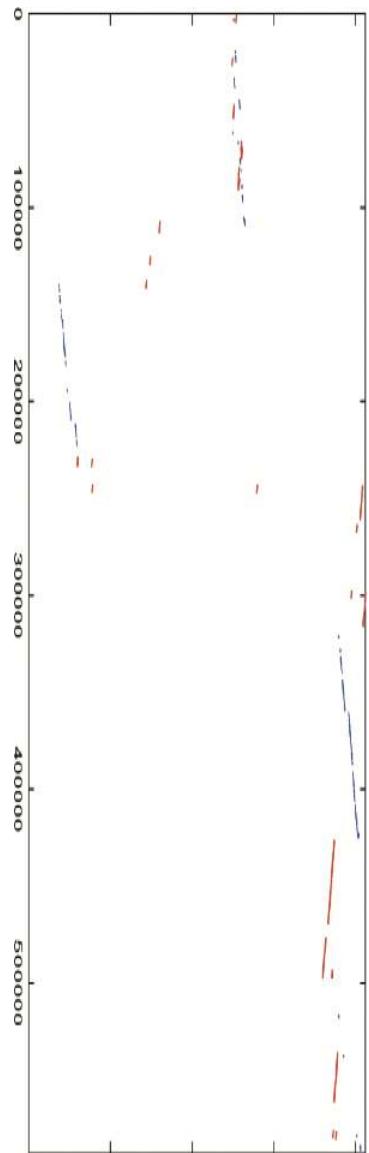


Figura 5.27: Resultados de S-LAGAN.



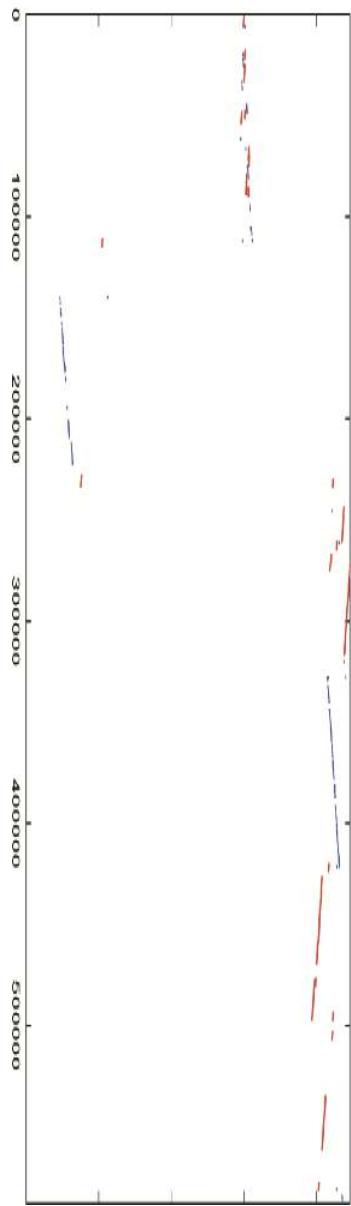
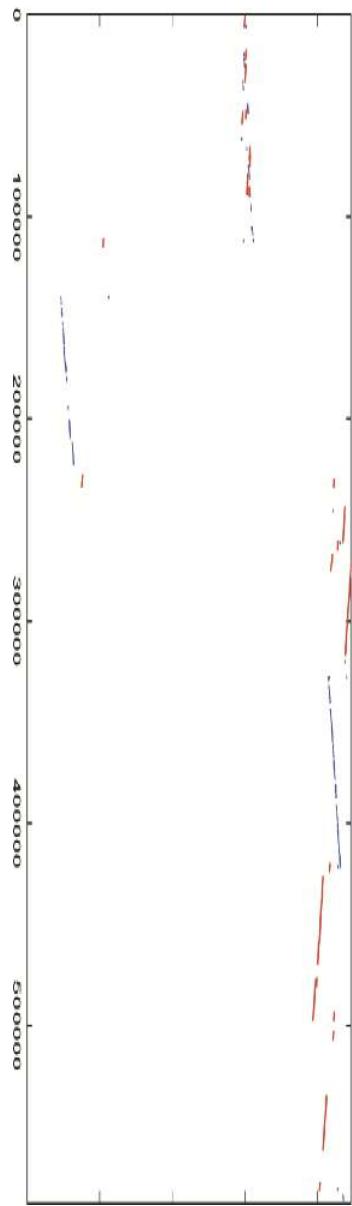


Figura 5.28: Resultados de S-LAGAN para locus IGF.



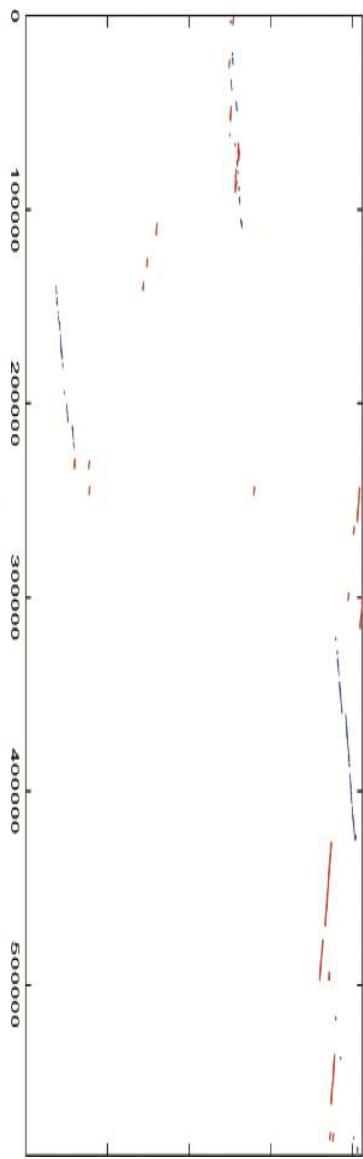


Figura 5.29: Resultados de S-LAGAN para locus IGF.

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

- [7] Douglas Robinson y Lynn Cooley. Examen de la función de dos proteínas kelch generadas por supresión de codones de parada. Desarrollo, 1997.
- [8] Stark. Descubrimiento de elementos funcionales en 12 genomas de drosophila usando firmas evolutivas. Naturaleza, 2007.
- [9] Angela Tan. Conferencia 15 notas: Genómica comparada i: Anotación del genoma. 4 de noviembre de 2009.

---

Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 6: Genómica Bacteriana—Evolución Molecular a Nivel de Ecosistemas

- 6.1: Introducción
- 6.2: Estudio 1- Evolución de la vida en la tierra
- 6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros
- 6.4: Estudio 3- Proyecto de Ecología Gut Humana (HuGE)
- 6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo
- 6.6: Estudio 5- Transferencia Génica Horizontal (HGT) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos
- 6.7: Estudio 6- Identificación de factores de virulencia en Meningitis
- 6.08: Q
- 6.8: Q/A
- 6.9: Direcciones actuales de investigación
- 6.10 Lectura adicional
- 6.12 ¿Qué hemos aprendido?
- Bibliografía

---

This page titled 6: Genómica Bacteriana—Evolución Molecular a Nivel de Ecosistemas is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 6.1: Introducción

Con la magnitud y diversidad de las poblaciones bacterianas en el cuerpo humano, el microbioma humano tiene muchas propiedades comunes con los ecosistemas naturales investigados en biología ambiental. Como un campo con un gran número de problemas cuantitativos que abordar, la genómica bacteriana ofrece una oportunidad para que el biólogo computacional participe activamente en el avance de esta área de investigación.

Hay aproximadamente  $10^{14}$  células microbianas en un intestino humano promedio, mientras que solo hay  $10^{13}$  células humanas en un cuerpo humano en total. Además, hay  $10^{12}$  células microbianas externas que viven en nuestra piel. Desde la perspectiva del recuento celular, esto corresponde a 10 veces más células bacterianas en nuestro cuerpo que nuestras propias células. Desde la perspectiva del recuento de genes, hay 100 veces más genes pertenecientes a las bacterias que viven en/sobre nosotros que a nuestras propias células. Por esta razón, estas comunidades microbianas que viven en nuestros cuerpos son parte integral de lo que nos hace humanos y debemos investigar sobre estos genes que no están codificados directamente en nuestro genoma, pero que aún tienen un efecto significativo en nuestra fisiología.

### Evolución de la investigación del microbioma

Las primeras etapas de la investigación de microbiomas se basaron principalmente en la recolección de datos y el análisis de encuestas de grupos bacterianos presentes en un ecosistema particular. Además de recolectar datos, este tipo de investigación también involucró la secuenciación de genomas bacterianos y la identificación de marcadores génicos para determinar diferentes grupos bacterianos presentes en la muestra. El marcador más utilizado para este propósito es el gen ARNr 16S, que es una sección del ADN procariota que codifica ARN ribosómico. Tres características principales del gen 16S que lo convierten en un marcador muy efectivo para estudios de microbiomas son: (1) su tamaño corto ( $\sim 1500$  bases) que lo hace más barato de secuenciar y analizar, (2) alta conservación debido a los requisitos exactos de plegamiento del ARN ribosómico para el que codifica, y (3) su especificidad para procariota organismos que nos permiten diferenciar de ADN contaminante protista, fúngico, vegetal y animal.

Otra dirección en la investigación microbiana temprana fue inferir reglas a partir de conjuntos de datos generados sobre ecosistemas microbianos. Estos estudios investigaron inicialmente datos microbianos generados e intentaron comprender las reglas de abundancia microbiana en diferentes tipos de ecosistemas e inferir redes de poblaciones bacterianas en cuanto a su co-ocurrencia, correlación y causalidad entre sí.

Un tipo más reciente de investigación microbiana adopta un enfoque predictivo y tiene como objetivo modelar el cambio de poblaciones bacterianas en un ecosistema a través del tiempo haciendo uso de ecuaciones diferenciales. Por ejemplo, podemos modelar la tasa de cambio para el tamaño de la población de un grupo bacteriano particular en el intestino humano como una ecuación diferencial ordinaria (ODE) y usar este modelo para predecir el tamaño de la población en un momento futuro integrándolo en el intervalo de tiempo.

Podemos modelar el cambio de poblaciones bacterianas con respecto a múltiples parámetros, como el tiempo y el espacio. Cuando tenemos suficientes datos para representar poblaciones microbianas temporal y espacialmente, podemos modelizarlas usando ecuaciones diferenciales parciales (PDE) para hacer predicciones usando funciones multivariadas.

### Generación de datos para la investigación de microbiomas

La generación de datos para la investigación de microbiomas generalmente sigue el siguiente flujo de trabajo: (1) se toma una muestra de ecosistema microbiano del sitio particular que se estudia (por ejemplo, la piel de un paciente o un lago), (2) se extraen los ADN de las bacterias que viven en la muestra, (3) se secuencian genes de ARNr 16S, (4) se conservan los motivos en alguna fracción del gen 16S (códigos de barras de ADN) se agrupan en unidades taxonómicas operacionales (OTU), y (5) se construye un vector de abundancia para todas las especies de la muestra. En microbiología, las bacterias se clasifican en OTU de acuerdo a sus propiedades funcionales y no a las especies, debido a la dificultad de aplicar la definición de especie convencional al mundo bacteriano.

En el resto de la conferencia se describen una serie de estudios recientes que están relacionados con el campo de la genómica bacteriana y los estudios del microbioma humano.

This page titled [6.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

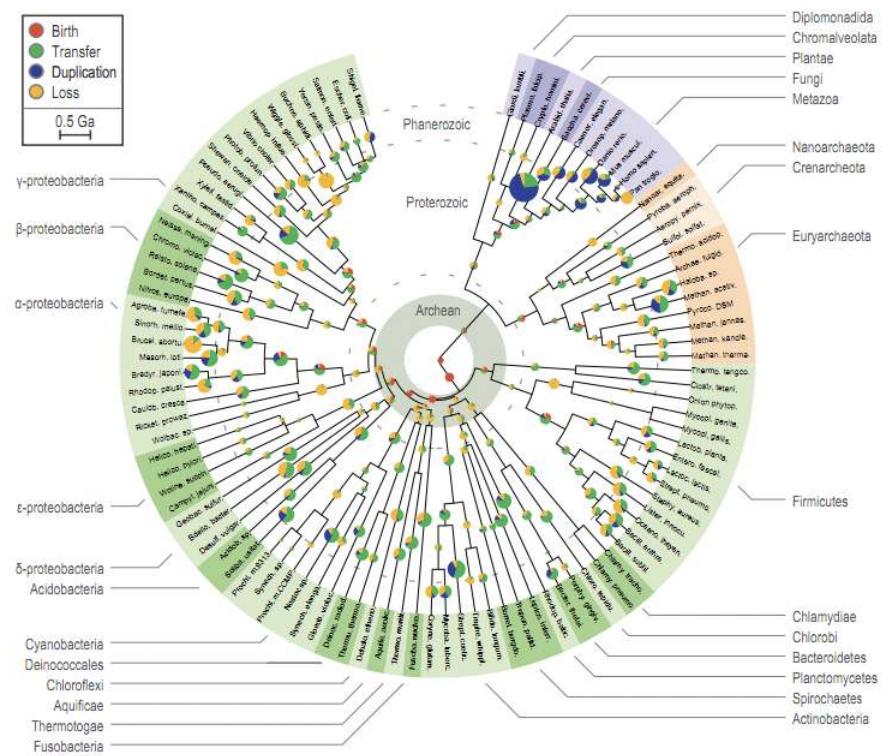
- [6.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.2: Estudio 1- Evolución de la vida en la tierra

Este estudio [2] se inspira en una cita de Max Delbrück: "Cualquier célula viva lleva consigo la experiencia de mil millones de años de experimentación por parte de sus antepasados". En esta dirección, es posible encontrar evidencia en los genomas de organismos vivos de cambios ambientales antiguos con grandes impactos biológicos. Por ejemplo, el oxígeno que la mayoría de los organismos utilizan actualmente habría sido extremadamente tóxico para casi toda la vida en la tierra antes de la acumulación de oxígeno a través de la fotosíntesis oxigénica. Se sabe que este suceso ocurrió hace aproximadamente 2.4 mil millones de años y provocó una dramática transformación de la vida en la tierra.

Se desarrolló un algoritmo de programación dinámica para inferir eventos de nacimiento, duplicación, pérdida y transferencia génica horizontal dada la filogenia de especies y filogenia de diferentes genes. La transferencia horizontal de genes es el evento en el que las bacterias transfieren una porción de su genoma a otras bacterias de diferentes grupos taxonómicos.

La Figura 6.1 muestra una visión general de estos eventos inferidos en un árbol filogenético enfocándose en la vida procariota. En cada nodo, el tamaño del gráfico circular representa la cantidad de cambio genético entre dos ramas y cada porción coloreada representa la tasa de un evento de modificación genética particular. Partiendo de la raíz del árbol, vemos que casi todo el gráfico circular está representado por genes recién nacidos representados por rojo. Sin embargo, hace alrededor de 2.5 mil millones de años los cortes verdes y azules se vuelven más prevalentes, lo que representa la tasa de transferencia génica horizontal y eventos de duplicación génica.



© Lawrence David. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 6.1: Árbol de la vida que muestra tasas de nacimiento, duplicación, pérdida y transferencia génica horizontal en cada punto de ramificación.

En la Figura 6.2, se puede observar un pico grande durante el eón arcaico que representa una gran cantidad de cambio genético en la tierra que ocurre durante este período de tiempo en particular. Este estudio buscó la actividad enzimática de genes que nacieron en este eón diferentes a los genes que ya estaban presentes. En el lado derecho de la Figura 6.2, se muestran los niveles logarítmicos de enriquecimiento de diferentes metabolitos. Se descubrió que la mayoría de los metabolitos enriquecidos producidos por estos genes son funcionales en la reducción de oxidación y el transporte de electrones. En general, este estudio sugiere que la

vida inventó la cadena moderna de transporte de electrones hace alrededor de 3.3 mil millones de años y hace alrededor de 2.8 mil millones de años, los organismos evolucionaron para usar las mismas proteínas que se utilizan para producir oxígeno también para respirar oxígeno.

---

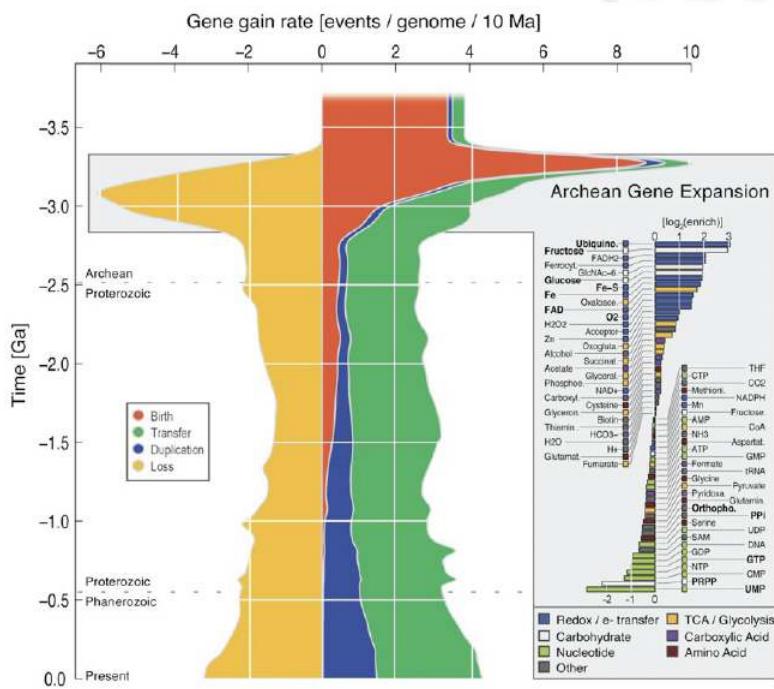
This page titled [6.2: Estudio 1- Evolución de la vida en la tierra](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [6.2: Study 1- Evolution of life on earth](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros

En algunas enfermedades como la Enfermedad Inflamatoria Intestinal (EII); si la enfermedad no es diagnosticada y monitoreada de cerca, los resultados pueden ser muy severos, como la extirpación del colon del paciente. Por otro lado, actualmente los métodos de diagnóstico más confiables existentes son muy invasivos (por ejemplo, colonoscopia). Un enfoque alternativo para el diagnóstico puede ser el análisis de abundancia de la muestra microbiana tomada del colon de los pacientes. Este estudio tiene como objetivo predecir el estado de enfermedad del sujeto a partir de abundancias bacterianas en muestras de heces tomadas del paciente.

Se colectaron 105 muestras para este estudio entre los pacientes del Dr. Athos Boudvaros; algunos de ellos



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: David, Lawrence A. y Eric J. Alm. "Innovación Evolutiva Rápida durante una Expansión Genética Arcaica". *Naturaleza* 469, núm. 7328 (2011): 93-96.

Figura 6.2: Tasas de nacimiento de nuevos genes, duplicación, pérdida y transferencia horizontal de genes durante la expansión génica arcaica

mostrando síntomas de EII y otras enfermedades diferentes (grupo control). En la Figura 6.3, cada bloque de fila representa un conjunto de grupos bacterianos a nivel taxonómico (nivel de filo en la parte superior y nivel de género en la parte inferior) y cada bloque de columna representa un grupo de pacientes diferente: pacientes control, enfermedad de Crohn (EC) y colitis ulcerosa (CU). El único biomarcador significativo fue *E. Coli*, que no se ve en pacientes control y EC sino en aproximadamente un tercio de los pacientes con CU. No parece haber otro grupo bacteriano único que dé una clasificación significativa entre los grupos de pacientes a partir de estas medidas de abundancia.

Dado que la abundancia de *E. Coli* no es un biomarcador bacteriano simple, su uso como herramienta diagnóstica produciría una clasificación de baja precisión. Por otro lado, podemos tomar la distribución de abundancia del grupo bacteriano completo y alimentarlos en un bosque aleatorio y estimar la precisión de validación cruzada. Despues de que se empleó el método de clasificación, fue capaz de decir con 90% de precisión si el paciente está enfermo o no. Esto sugiere que es un método competitivo con respecto a otros enfoques diagnósticos no invasivos que generalmente son altamente específicos pero no lo suficientemente sensibles.

Una diferencia clave entre los grupos control y los grupos de enfermedades es la disminución de la diversidad del ecosistema. Esto sugiere que el estado de la enfermedad no está controlado por un solo germe sino por la robustez general y la resiliencia del ecosistema. Cuando la diversidad en el ecosistema disminuye, el paciente puede comenzar a mostrar síntomas de enfermedad.

This page titled [6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **6.3: Study 2- Pediatric IBD study with Athos Boudvaros** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.4: Estudio 3- Proyecto de Ecología Gut Humana (HuGE)

Este estudio tiene como objetivo identificar más de trescientos factores dietéticos y ambientales que afectan al microbioma humano. Los factores, que fueron seguidos regularmente por una App para iPhone, fueron los alimentos que el sujeto comió, cuánto dormía, el estado de ánimo en el que estaban etc. Además, se tomaron muestras de heces de los sujetos todos los días durante un año para realizar análisis de secuencia de las abundancias del grupo bacteriano para un día específico



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 6.3: Niveles de abundancia de diferentes grupos bacterianos en pacientes control, pacientes con enfermedad de Crohn y pacientes con colitis ulcerosa.

relevante para un factor ambiental particular. La motivación detrás de llevar a cabo este estudio es que, por lo general, es muy difícil obtener una señal fuerte entre las abundancias bacterianas y el estado de la enfermedad. La exploración de los efectos dietéticos sobre el microbioma humano podría dilucidar algunos de estos factores de confusión en el análisis de abundancia bacteriana. Sin embargo, este estudio analizó factores dietéticos y ambientales en los ecosistemas intestinales de solo dos sujetos; inferir correlaciones estadísticamente significativas con factores ambientales requeriría grandes cohortes de sujetos.

La Figura 6.4 muestra los niveles de abundancia de diferentes grupos bacterianos en el intestino de los dos donantes a lo largo del experimento. Un punto clave a notar es que dentro de un individuo, la abundancia bacteriana es muy similar a través del tiempo. Sin embargo, las abundancias de grupos bacterianos en el intestino difieren significativamente de persona a persona.

Un factor dietético estadísticamente significativo que se descubrió como marcador predictivo de abundancias de población bacteriana es el consumo de fibra. Se dedujo que el consumo de fibra está altamente correlacionado con la abundancia de grupos bacterianos como Lachnospiraceae, Bifidobacteria y Ruminococcaceae. En el Donante B, el incremento de 10g en el consumo de fibra incrementó la abundancia general de estos grupos bacterianos en 11%.

En la Figura 6.6 y la Figura 6.7, se muestra una gráfica de horizonte de los dos donantes B y A respectivamente. En la Figura 6.5 se da una leyenda para leer estas gráficas de horizonte. Para cada grupo bacteriano se muestra el gráfico abundancia-tiempo con diferentes colores para diferentes capas de abundancia, los segmentos de diferentes capas se colapsan en la altura de una sola capa mostrando solo el color con la mayor diferencia de valor absoluto de la abundancia normal, y finalmente el negativo los picos se cambian a picos positivos conservando su color original.

En la Figura 6.6, vemos que durante el viaje del donante a Tailandia, se produce un cambio significativo en su ecosistema bacteriano intestinal. Un gran número de grupos bacterianos desaparecen (mostrados en la mitad inferior de la parcela del horizonte) tan pronto como el donante comienza a vivir en Tailandia. Y tan pronto como el donante regresa a Estados Unidos, los niveles de abundancia de estos grupos bacterianos vuelven rápidamente a sus niveles normales. Además, algunos grupos bacterianos que normalmente se consideran patógenos (los primeros 8 grupos se muestran en la parte superior) aparecen en el ecosistema del donante casi tan pronto como el donante se traslada a Tailandia y en su mayoría desaparece cuando regresa a Estados Unidos. Esto indica que los factores ambientales (como la ubicación) pueden causar cambios importantes en nuestro ecosistema intestinal mientras que el factor ambiental está presente pero puede desaparecer después de que se elimine el factor.

Se retiraron cifras del laboratorio David debido a restricciones de derechos de autor.

Figura 6.4: Abundancias bacterianas intestinales trazadas a lo largo del tiempo para los dos donantes participantes en el proyecto HuGE.

Se retiraron cifras del laboratorio David debido a restricciones de derechos de autor.

Figura 6.5: Descripción de cómo leer una gráfica de horizonte.

Se retiraron cifras del laboratorio David debido a restricciones de derechos de autor.

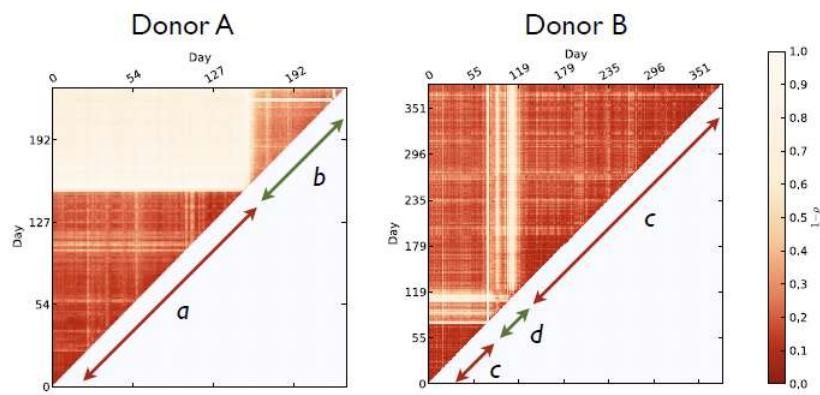
Figura 6.6: Gráfica de horizonte del Donante B en el estudio HuGE.

En la Figura 6.7, vemos que después de que el donante se infecta con salmonela, una porción significativa de su ecosistema intestinal es reemplazada por otros grupos bacterianos. Un gran número de grupos bacterianos desaparecen permanentemente durante la infección y otros grupos bacterianos reemplazan sus nichos ecológicos. En otras palabras, la introducción de un nuevo factor ambiental lleva el ecosistema bacteriano en el intestino del donante de un punto de equilibrio a otro completamente diferente. A pesar de que la población bacteriana consiste principalmente en salmonela durante la infección, antes y después de la infección el recuento bacteriano permanece más o menos igual. El escenario que ocurrió aquí es que la salmonela llevó a algunos grupos bacterianos a la extinción en el intestino y grupos bacterianos similares se apoderaron de sus nichos ecológicos vacíos.

En la Figura 6.8, se muestran valores p para los niveles de correlación de abundancia bacteriana del día a día para el Donante A y B. En la matriz de correlación del Donante A, existe una alta correlación dentro del intervalo de tiempo a correspondiente a la preinfección y dentro del intervalo b correspondiente a la posinfección. Sin embargo, entre a y b casi no hay correlación alguna. Por otro lado, en la matriz de correlación del donante B, vemos que los intervalos de tiempo previos a Tailandia y post-Tailandia, c, tienen una alta correlación dentro y entre ellos. Sin embargo, el intervalo d que corresponde al periodo de tiempo del viaje del Donante B a Tailandia, vemos relativamente poca correlación con c. Esto sugiere que las perturbaciones en el ecosistema bacteriano del Donante B no fueron suficientes para causar un desplazamiento permanente del equilibrio de abundancia como en el caso del Donante A debido a infección por salmonela.

Se retiraron cifras del laboratorio David debido a restricciones de derechos de autor.

Figura 6.7: Gráfica de horizonte del Donante A en el estudio HuGE.



Cortesía de Lawrence David. Usado con permiso.

Figura 6.8: Matrices diarias de correlación de abundancia bacteriana del Donante A y el Donante B.

This page titled [6.4: Estudio 3- Proyecto de Ecología Gut Humana \(HuGE\)](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [6.4: Study 3- Human Gut Ecology \(HuGE\) project](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo

En un estudio de Mozaffarian et al. [4] se analizaron más de cien mil pacientes con el objetivo de descubrir el efecto de las elecciones de dieta y estilo de vida sobre el aumento de peso y la obesidad a largo plazo. Este estudio construyó un modelo para predecir los pesos de los pacientes en función de los tipos y cantidades de alimentos que consumieron durante un cierto período de tiempo. Descubrieron que el tipo de comida rápida (carnes procesadas, papas fritas, bebidas azucaradas) estaban más correlacionados con la obesidad. Por otro lado, el nivel de consumo de yogur se correlacionó inversamente con la obesidad.

Otros experimentos con cohortes de ratones y humanos mostraron que, tanto dentro del grupo control como en el grupo de comida rápida, el aumento del consumo de yogur conduce a la pérdida de peso. En el experimento con ratones, a algunos ratones hembra se les administró *Lactobacillus reuteri* (un grupo de bacterias que se encuentra en el yogur) y se les permitió comer tanta comida regular o comida rápida que quisieran. Esto resultó en una pérdida de peso significativa en el grupo de ratones a los que se les administró el extracto bacteriano purificado.

Se descubrió que un efecto fenotípico inesperado del consumo de yogur orgánico era el pelaje más brillante de los ratones y perros a los que se les administró yogur como parte de su dieta. Un análisis histológico de la biopsia de piel de los ratones control y alimentados con yogur demuestra que los ratones que fueron alimentados con la bacteria en yogur tenían folículos pilosos que están activos, lo que lleva al desarrollo activo de pelaje y pelo más sanos y brillantes.

---

This page titled [6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [6.5: Study 4- Microbiome as the connection between diet and phenotype](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.6: Estudio 5- Transferencia Génica Horizontal (HGT) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos

Un estudio de Hehemann et al. [3] descubrió un gen específico que digiere un tipo de carbohidrato sulfonado que solo se encuentra en envolturas de sushi de algas marinas. Este gen se encuentra en los microbios intestinales de los japoneses pero no en los norteamericanos. El estudio concluyó que este gen específico se ha transferido en algún momento de la historia de las algas mismas a las bacterias que viven en él y luego al microbioma intestinal de una persona japonesa por transferencia génica horizontal. Este estudio también sugiere que, aunque algún grupo bacteriano pueda vivir en nuestro intestino durante toda nuestra vida, pueden obtener nuevas funcionalidades a lo largo de nuestra vida al recoger nuevos genes dependiendo del tipo de alimento que comamos.

En esta dirección, un estudio en el Laboratorio de Alm investigó alrededor de 2000 genomas bacterianos publicados en [1] con el objetivo de detectar genes que son 100% similares pero que pertenecen a bacterias en diferentes grupos taxonómicos. Cualquier gen que sea exactamente el mismo entre diferentes grupos bacterianos indicaría un evento de transferencia génica horizontal. En este estudio, se descubrieron alrededor de 100000 casos de este tipo.

Al observar ambientes específicos, se descubrió que las bacterias aisladas de los humanos comparten genes principalmente con otras bacterias aisladas de sitios humanos. Si nos enfocamos en sitios más específicos, vemos que los genomas bacterianos aislados del intestino humano comparten genes principalmente con otras bacterias que están aisladas del intestino, y los genomas bacterianos aislados de piel humana comparten genes principalmente con otros aislados de piel humana. Este hallazgo sugiere que independientemente de la filogenia de los grupos bacterianos, la ecología es el factor más importante que determina la cantidad de instancias de transferencia génica entre grupos bacterianos.

En la Figura 6.9, vemos que entre diferentes grupos bacterianos tomados de humanos que tienen al menos 3% de distancia del gen 16S, hay alrededor de 23% de probabilidad de que compartan un gen idéntico en su genoma. Además, hay más del 40% de probabilidad de que compartan un gen idéntico si también se muestran del mismo sitio.

Por otro lado, la Figura 6.10 muestra que la geografía es una influencia débil en la transferencia génica horizontal. Las poblaciones bacterianas muestreadas del mismo continente y diferentes continentes tuvieron poca diferencia en cuanto a la cantidad de transferencia genética horizontal detectada.

La Figura 6.11 muestra una matriz codificada por colores de los niveles de HGT entre diversos entornos humanos y no humanos; el triángulo superior derecho representa la cantidad de transferencias de genes horizontales y el triángulo inferior izquierdo que muestra el porcentaje de genes de resistencia a antibióticos (AR) entre los genes transferidos. En la esquina superior derecha, vemos que existe un ligero exceso de instancias de HGT entre microbioma humano y muestras bacterianas tomadas de animales de granja. Y cuando miramos los porcentajes correspondientes de genes de resistencia a antibióticos, vemos que más del 60% de las transferencias son genes AR. Este resultado muestra el efecto directo de la alimentación de antibióticos subterapéuticos al ganado sobre la aparición de genes de resistencia a antibióticos en las poblaciones bacterianas que viven en el intestino humano.

---

This page titled [6.6: Estudio 5- Transferencia Génica Horizontal \(HGT\) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [6.6: Study 5- Horizontal Gene Transfer \(HGT\) between bacterial groups and its effect on antibiotic resistance](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.7: Estudio 6- Identificación de factores de virulencia en Meningitis

La meningitis bacteriana es una enfermedad que es causada por bacterias muy diversas que son capaces de entrar en el torrente sanguíneo y cruzar la barrera hematoencefálica. Este estudio tuvo como objetivo investigar los factores de virulencia que pueden convertir las bacterias en un tipo que puede causar meningitis.

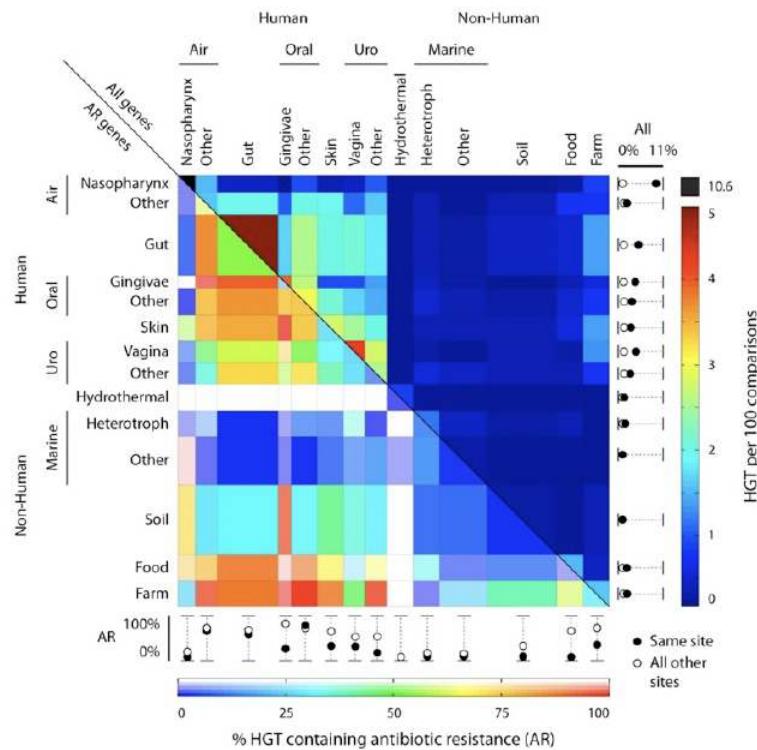
Cifras eliminadas por restricciones de derechos de autor. Ver cifras similares en este artículo de revista: Smillie, Chris S. et al. "La ecología impulsa una red global de intercambio de genes que conecta el microbioma humano". Naturaleza 480, núm. 7376 (2011): 241-244.

Figura 6.9: Tasa de transferencia génica horizontal entre diferentes grupos bacterianos tomados de sitios no humanos, sitios humanos, mismo sitio dentro del ser humano y diferentes sitios dentro del ser humano.

Cifras eliminadas por restricciones de derechos de autor. Ver cifras similares en este artículo de revista: Smillie, Chris S. et al. "La ecología impulsa una red global de intercambio de genes que conecta el microbioma humano". Naturaleza 480, núm. 7376 (2011): 241-244.

Figura 6.10: Tasa de transferencia horizontal de genes entre grupos bacterianos muestrados del mismo continente y de diferentes continentes.

El estudio involucró 70 cepas bacterianas aisladas de pacientes con meningitis, que comprenden 175172 genes en total. Alrededor de 24000 de estos genes no tenían ninguna función conocida. Podría haber algunos genes entre estos 24000



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Smillie, Chris S., et al. "La ecología impulsa una red global de intercambio de genes que conecta el microbioma humano". Naturaleza 480, núm. 7376 (2011): 241-4.

Figura 6.11: Tasa de transferencia génica horizontal entre diferentes sitios humanos y no humanos (arriba a la derecha) y porcentaje de genes de resistencia antiboítica entre las transferencias de genes horizontales (abajo a la izquierda)

que podrían estar llevando a bacterias causantes de meningitis y podrían ser buenos objetivos de drogas. Además, se descubrió que 82 genes estaban involucrados en la transferencia horizontal de genes. 69 de estos tenían funciones conocidas y 13 de ellos pertenecían a los 24000 genes que no tenemos ninguna información funcional. Entre los genes con función conocida, algunos de ellos estaban relacionados con AR, desintoxicación, y también algunos se relacionaron con factores de virulencia conocidos como

la hemolisina que deja que las bacterias vivan en el torrente sanguíneo y la adesina que ayuda a que las bacterias se enganchen en la vena y potencialmente crucen la barrera hematoencefálica.

---

This page titled [6.7: Estudio 6- Identificación de factores de virulencia en Meningitis](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [6.7: Study 6- Identifying virulence factors in Meningitis](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.08: Q

Esta página se ha generado automáticamente porque un usuario ha creado una subpágina de esta página.

---

6.08: Q is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 6.8: Q/A

P: ¿Crees que después de algún tiempo el Donante A en el Estudio 3 tendrá su ecosistema bacteriano regresando a su estado original de preinfección?

R: La infección por salmonela provocó la eliminación de ciertos nichos del ecosistema bacteriano del Donante A que luego fueron llenados por bacterias similares y alcanzadas a un ecosistema diferente en un nuevo equilibrio. Dado que estos nichos están dominados por los nuevos grupos de bacterias, no sería posible que los grupos bacterianos anteriores los reemplazaran sin un cambio a gran escala en su ecosistema intestinal.

P: ¿La muerte de ciertos grupos bacterianos en el intestino durante la infección por salmonela es causada directamente por la infección o es una respuesta inmune para curar la enfermedad?

R: Pueden ser ambas, pero es muy difícil de decir a partir de los datos en el Estudio 3 ya que es sólo un punto de datos que corresponde al evento que podemos observar. Un estudio futuro que trate de averiguar qué sucede en nuestro sistema inmunológico durante la infección se puede observar mediante la extracción de sangre de los pacientes durante la infección.

P: ¿Existe una conexión particular entre el genoma de un individuo y los grupos bacterianos dominantes en el ecosistema bacteriano? ¿Los gemelos mostrarían ecosistemas bacterianos más similares?

R: Los gemelos en general tienen ecosistemas bacterianos similares independientemente de si viven juntos o están separados. Aunque esto parece ser un factor genético al principio, los gemelos monocigóticos y dicigóticos tienen exactamente el mismo efecto, además de mostrar similitud con el ecosistema bacteriano de sus madres. La razón de esto es que a partir del nacimiento hay un periodo de tiempo en el que se programa el ecosistema bacteriano. El efecto de similitud entre gemelos se basa en esto más que en factores genéticos.

---

This page titled 6.8: Q/A is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- 6.8: Q/A by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.9: Direcciones actuales de investigación

### Direcciones actuales de investigación

Una extensión adicional al estudio HuME podría observar el microbioma intestinal de ratones durante una infección por salmonela y observar el proceso de algunos grupos bacterianos siendo conducidos a la extinción y otros tipos de bacterias reemplazando los nichos ecológicos que son vaciados por ellos. Una observación de mayor resolución de este fenómeno en ratones podría iluminar cómo los ecosistemas bacterianos cambian de un equilibrio a otro.

This page titled 6.9: Direcciones actuales de investigación is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **6.9: Current Research Directions** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 6.10 Lectura adicional

- Visión general del Proyecto Microbioma Humano: commonfund.nih.gov/hmp/overview.aspx
- Lawrence A. David y Eric J. Alm. (2011). Rápida innovación evolutiva durante un Archaean expansión genética. *Naturaleza*, 469 (7328) :93-96.
- Un tutorial sobre el gen ARNr 16S y su uso en la investigación del microbioma: [http://greengenes.lbl.gov/cgi-bin/JD\\_Tutorial/nph-Tutorial\\_2Main2.cgi](http://greengenes.lbl.gov/cgi-bin/JD_Tutorial/nph-Tutorial_2Main2.cgi)
- Dariush Mozaffarian, Tao Hao, Eric B. Rimm, Walter C. Willett y Frank B. Hu. (2011). Cambios en la dieta y estilo de vida y aumento de peso a largo plazo en mujeres y hombres. *The New England journal of medicine*, 364 (25) :2392-2404.
- JH Hehemann, G Corrc, T Barbeyron, W Helbert, M Czjzek, y G Michel. (2010). Transferencia de enzimas carbohidrato activas de bacterias marinas a microbiota intestinal japonesa. *Naturaleza*, 464 (5) :908-12.
- El Consorcio de Cepas de Referencia Jumpstart Microbioma Humano. (2010). Un Catálogo de Genomas de Referencia del Microbioma Humano. *Ciencia*, 328 (5981) :994-999

6.10 Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [6.10 Further Reading](#) has no license indicated.

## 6.12 ¿Qué hemos aprendido?

En esta conferencia, aprendimos sobre el campo de la genómica bacteriana en general y cómo los ecosistemas bacterianos pueden ser utilizados para verificar cambios ambientales importantes en etapas tempranas de evolución (Estudio 1), pueden actuar como una herramienta diagnóstica no invasiva (Estudio 2), se ven afectados temporal o permanentemente por diferentes y factores dietéticos (Estudio 3), pueden actuar como el vínculo entre la dieta y el fenotipo (Estudio 4), pueden hacer que genes de resistencia a antibióticos sean transportados entre el microbioma de diferentes especies a través de la transferencia horizontal de genes (Estudio 5), y pueden ser utilizados para identificar factores de virulencia significativos en estados de enfermedad (Estudio 6).

---

6.12 ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [6.12 What have we learned?](#) has no license indicated.

## Bibliografía

- 
- [1] Consorcio de Cepas de Referencia Jumpstart del Microbioma Humano. Un Catálogo de Genomas de Referencia del Microbioma Humano. *Ciencia*, 328 (5981) :994—999, mayo de 2010.
  - [2] Lawrence A. David y Eric J. Alm. Rápida innovación evolutiva durante una expansión genética arcaica. *Nature*, 469 (7328) :93—96, enero de 2011.
  - [3] JH Hehemann, G Corrc, T Barbeyron, W Helbert, M Czjzek y G Michel. Transferencia de enzimas carbohidrato activas de bacterias marinas a microbiota intestinal japonesa. *Naturaleza*, 464 (5) :908—12, 2010 Abr 8.
  - [4] Dariush Mozaffarian, Tao Hao, Eric B. Rimm, Walter C. Willett y Frank B. Hu. Cambios en la dieta y estilo de vida y aumento de peso a largo plazo en mujeres y hombres. *The New England journal of medicine*, 364 (25) :2392—2404, junio de 2011.
- 

Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 7: Modelos ocultos de Markov I

Los modelos ocultos de Markov (HMM) son una herramienta fundamental del aprendizaje automático que es ampliamente utilizada en biología computacional. Usando HMM, podemos explorar la estructura subyacente de secuencias de ADN o polipéptidos, detectando regiones de especial interés. Por ejemplo, podemos identificar subsecuencias conservadas o descubrir regiones con diferentes distribuciones de nucleótidos o aminoácidos tales como regiones promotoras e islas CpG. Mediante este modelo probabilístico, podemos iluminar las propiedades y componentes estructurales de las secuencias y localizar genes y otros elementos funcionales.

[7.1: Introducción](#)

[7.2: Motivación](#)

[7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#)

[7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología](#)

[7.5: Ajustes algorítmicos para HMM](#)

[7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo?](#)

[7.7: Lectura adicional, ¿qué hemos aprendido?](#)

---

This page titled [7: Modelos ocultos de Markov I](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 7.1: Introducción

En esta conferencia definiremos Cadenas de Markov y HMM, aportando una serie de ejemplos motivadores. En la segunda mitad de esta conferencia, discutiremos la puntuación y la decodificación. Aprenderemos a calcular la probabilidad de la combinación de una combinación particular de observaciones y estados. Presentaremos el Algoritmo Forward, un método para calcular la probabilidad de una secuencia dada de observaciones, permitiendo todas las secuencias de estados. Finalmente, discutiremos el problema de determinar el camino más probable de los estados correspondientes a las observaciones dadas, objetivo que se logra mediante el algoritmo de Viterbi.

En la segunda conferencia sobre HMM, continuaremos nuestra discusión sobre la decodificación explorando la decodificación posterior, lo que nos permite calcular el estado más probable en cada punto de la secuencia. Luego exploraremos cómo aprender un modelo oculto de Markov. Cubrimos tanto el aprendizaje supervisado como no supervisado, explicando cómo usar cada uno para aprender los parámetros del modelo. En el aprendizaje supervisado, tenemos datos de entrenamiento disponibles que etiquetan secuencias con modelos particulares. En el aprendizaje no supervisado, no tenemos etiquetas por lo que debemos buscar particionar los datos en categorías discretas basadas en similitudes probabilísticas descubiertas. En nuestra discusión sobre el aprendizaje no supervisado introduciremos el algoritmo general y ampliamente aplicable de Maximización de Expectativas (EM).

---

This page titled [7.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.2: Motivación

### Tienes una nueva secuencia de ADN, ¿y ahora qué?

#### 1. Alinear:

- con cosas que conocemos (búsqueda en bases de datos).
- con cosas desconocidas (montar/clustering)

#### 2. Visualízala: “Regla genómica #1”: ¡Mira tus datos!

- Busque composiciones de nucleótidos no estándar.
- Busque frecuencias k-mer que estén asociadas con regiones codificadoras de proteínas, datos recurrentes, alto contenido de GC, etc.
- Busca motivos, firmas evolutivas.
- Traducir y buscar marcos abiertos de lectura, codones de parada, etc.
- Busque patrones, luego desarrolle herramientas de aprendizaje automático para determinar modelos probabilísticos razonables. Por ejemplo, al observar una serie de cuatrilpes decidimos codificarlos por colores para ver dónde ocurren con mayor frecuencia.

#### 3. Modelarlo:

1. Hacer hipótesis.
2. Construir un modelo generativo para describir la hipótesis.
3. Usa ese modelo para encontrar secuencias de tipo similar.

No estamos buscando secuencias que necesariamente tengan ancestros comunes. Más bien, nos interesan secuencias con propiedades similares. En realidad no sabemos modelar genomas enteros, pero podemos modelar pequeños aspectos de genomas. La tarea requiere comprender todas las propiedades de las regiones genómicas y construir computacionalmente modelos generativos para representar hipótesis. Para una secuencia dada, queremos anotar regiones ya sean intrones, exones, intergénicas, promotoras o regiones clasificables de otro modo.

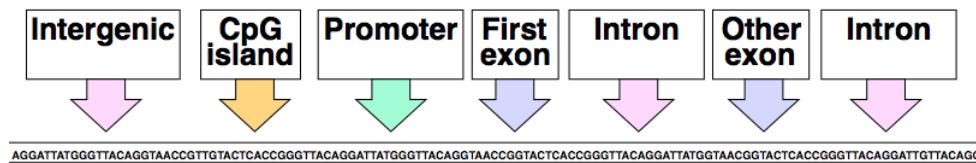


Figura 7.1: Modelado de secuencias biológicas © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Construir este marco nos dará la capacidad de:

- Emitir (generar) secuencias de tipo similar según el modelo generativo
- Reconocer el estado oculto que probablemente ha generado la observación
- Aprenda (entrene) grandes conjuntos de datos y aplique tanto a datos etiquetados previamente (aprendizaje supervisado) como a datos no etiquetados (aprendizaje no supervisado).

En esta conferencia se discuten algoritmos de emisión y reconocimiento.

### ¿Por qué modelar secuencias probabilísticas?

- Los datos biológicos son ruidosos.
- Actualizar conocimientos previos sobre secuencias biológicas.
- La probabilidad proporciona un cálculo para manipular modelos.
- No limitado a respuestas sí/no, puede proporcionar grados de creencia.
- Muchas herramientas computacionales comunes se basan en modelos probabilísticos.
- Nuestras herramientas: Cadenas Markov y HMM.

This page titled [7.2: Motivación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.2: Motivation](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización

### Ejemplo Motivador: Predicción del Tiempo

La predicción del tiempo siempre ha sido difícil, sobre todo cuando nos gustaría pronosticar el clima muchos días, semanas o incluso meses después. Sin embargo, si solo necesitamos predecir el clima del día siguiente, podemos alcanzar una precisión de predicción decente usando algunos modelos bastante simples como Markov Chain y Hidden Markov Model construyendo modelos gráficos en la Figura 7.2.

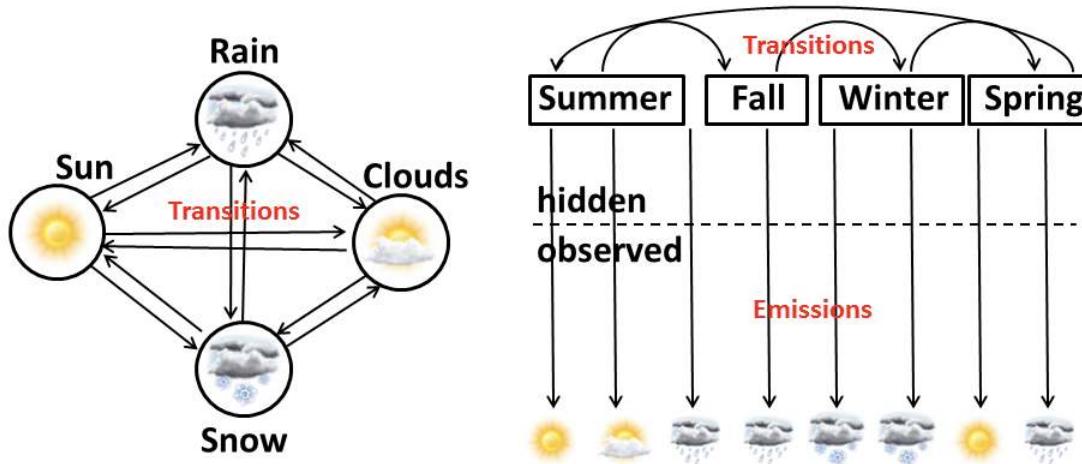


Figura 7.2: Modelos de predicción usando la cadena de Markov y HMM © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

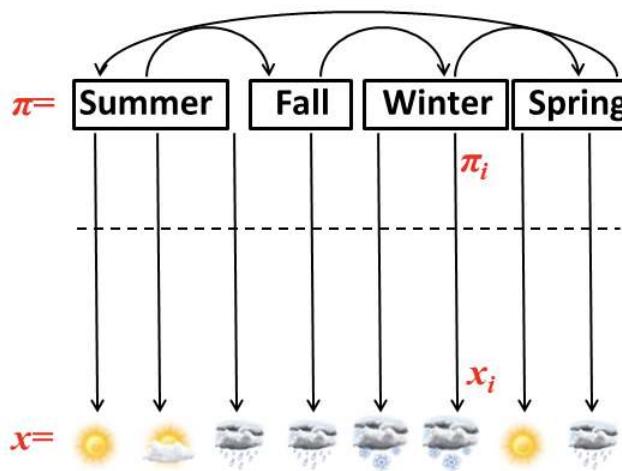
Para el modelo Markov Chain de la izquierda, cuatro tipos de clima (Sol, Lluvia, Nubes y Nieve) pueden pasar directamente de uno a otro. Esto es un “lo que ves es lo que obtienes” en que el siguiente estado sólo depende del estado actual y no hay memoria del estado anterior. Sin embargo para HMM a la derecha, todos los tipos de clima se modelan como la emisión (o resultado) de las estaciones ocultas (Verano, Otoño, Invierno y Primavera). La visión clave detrás es que los estados ocultos del mundo (por ejemplo, estación o sistema de tormentas) determinan las probabilidades de emisión mientras que las transiciones de estado están gobernadas por una cadena de Markov.

### Formalización de la Cadena de Markov y HMMS

Para echar un vistazo más de cerca al modelo Hidden Markov, primero definamos los parámetros clave en la Figura 7.3. El vector  $x$  representa la secuencia de observaciones. El vector  $\pi$  representa la ruta oculta, que es la secuencia de estados ocultos. Cada entrada  $a_{kl}$  de la matriz de transición  $A$  denota la probabilidad de transición del estado  $k$  al estado  $l$ . Cada entrada  $e_k(x_i)$  del vector de emisión denota la probabilidad de observar  $x_i$  desde el estado  $k$ . Y finalmente con estos parámetros y la regla de Bayes, podemos usar  $p(x_i | \pi_i = k)$  para estimar  $p(\pi_i = k | x_i)$ .

### Cadenas Markov

Una [Cadena de Markov](#) viene dada por un conjunto finito de estados y probabilidades de transición entre los estados. En cada paso de tiempo, la Cadena Markov se encuentra en un estado particular y experimenta una transición a otro estado. La probabilidad de transición a otro estado depende únicamente del estado actual, y en particular es independiente de cómo se alcanzó el estado actual. Más formalmente, una Cadena de Markov es un triplete  $(Q, p, A)$  que consiste en:



**Transitions:**  $a_{kl} = P(\pi_i=l|\pi_{i-1}=k)$

**Transition probability  
from state  $k$  to state  $l$**

**Emissions:**  $e_k(x_i) = P(x_i|p_i=k)$

**Emission probability of  
symbol  $x_i$  from state  $k$**

Figura 7.3: Parametrización del Modelo de Predicción HMM. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Un conjunto de estados Q.

- Una matriz de transición A cuyos elementos corresponden a la probabilidad  $A_{ij}$  de transición del estado i al estado j.
- Un vector p de probabilidades de estado inicial.

La propiedad clave de las Cadenas de Markov es que no tienen memoria, es decir, cada estado depende únicamente del estado anterior. Entonces podemos definir inmediatamente una probabilidad para el siguiente estado, dado el estado actual:

$$P(x_i | x_{i-1}, \dots, x_1) = P(x_i | x_{i-1}) \quad (7.3.1)$$

De esta manera, la probabilidad de la secuencia se puede descomponer de la siguiente manera:

$$P(x) = P(x_L, x_{L-1}, \dots, x_1) = P(x_L | x_{L-1})P(x_{L-1} | x_{L-2}) \dots P(x_2 | x_1)P(x_1) \quad (7.3.2)$$

$P(x_L)$  también se puede calcular a partir de las probabilidades de transición: Si multiplicamos las probabilidades de estado iniciales en el tiempo  $t = 0$  por la matriz de transición A, obtenemos las probabilidades de los estados en el tiempo  $t = 1$ . Multiplicando por la potencia apropiada AL de la matriz de transición, obtenemos las probabilidades de estado en el tiempo  $t = L$ .

### Modelos ocultos de Markov

Los modelos ocultos de Markov se utilizan como representación de un espacio problemático en el que las observaciones surgen como resultado de estados de un sistema que no podemos observar directamente. Estas observaciones, o emisiones, resultan de un estado particular basado en un conjunto de probabilidades. Así, los HMM son Modelos Markov donde los estados están ocultos al observador y en su lugar tenemos observaciones generadas con ciertas probabilidades asociadas a cada estado. Estas probabilidades de observaciones se conocen como probabilidades de emisión.

Formalmente, un modelo oculto de Markov es una tupla de 5 ( $Q, A, p, V, E$ ) que consta de los siguientes parámetros:

- Una serie de estados, Q.
- Una matriz de transición, A
- Un vector de probabilidades de estado inicial, p.
- Un conjunto de símbolos de observación, V, por ejemplo {A, T, C, G} o el conjunto de aminoácidos o palabras en un diccionario de inglés.
- Una matriz de probabilidades de emisión, E: Para cada s, t, en Q, la probabilidad de emisión es  $e_{sk} = P(v_k \text{ en el tiempo } t | q_t = s)$

La propiedad clave de la falta de memoria se hereda de Markov Models. Las emisiones y transiciones dependen únicamente del estado actual y no de la historia pasada.

This page titled [7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **7.3: Markov Chains and HMMS - From Example to Formalizing** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología

### El Casino Deshonesto

Imagina el siguiente escenario: Entras a un casino que ofrece un juego de dados rodando. Apostas \$1 y luego tú y un repartidor tiran un dado. Si rotas un número más alto ganas \$2. Ahora hay un giro en este juego aparentemente sencillo. Eres consciente de que el casino tiene dos tipos de dados:

1. Muere justo:  $P(1) = P(2) = P(3) = P(4) = P(5) = P(6) = 1/6$
2. Troquel cargado:  $P(1) = P(2) = P(3) = P(4) = P(5) = 1/10$  y  $P(6) = 1/2$

El crupier puede cambiar entre estos dos dados en cualquier momento sin que usted lo sepa. La única información que tienes son los rollos que observas. Podemos representar el estado del casino muere con un sencillo modelo de Markov:

El modelo muestra los dos estados posibles, sus emisiones y probabilidades de transición entre ellos. Las probabilidades de transición son conjeturas educadas en el mejor de los casos. Suponemos que el cambio entre los estados no ocurre con demasiada frecuencia, de ahí la probabilidad .95 de permanecer en el mismo estado con cada tirada.

### Mantenerse en contacto con la biología: una analogía

A modo de comparación, la Figura 7.5 a continuación da un modelo similar para una situación en biología donde una secuencia de ADN tiene dos fuentes potenciales: inyección por un virus versus producción normal por el propio organismo:

Dado este modelo como hipótesis, observaríamos las frecuencias de C y G para darnos pistas sobre la fuente de la secuencia en cuestión. Este modelo asume que los insertos virales tendrán mayor prevalencia de CpG, lo que conduce a las mayores probabilidades de ocurrencia de C y G.

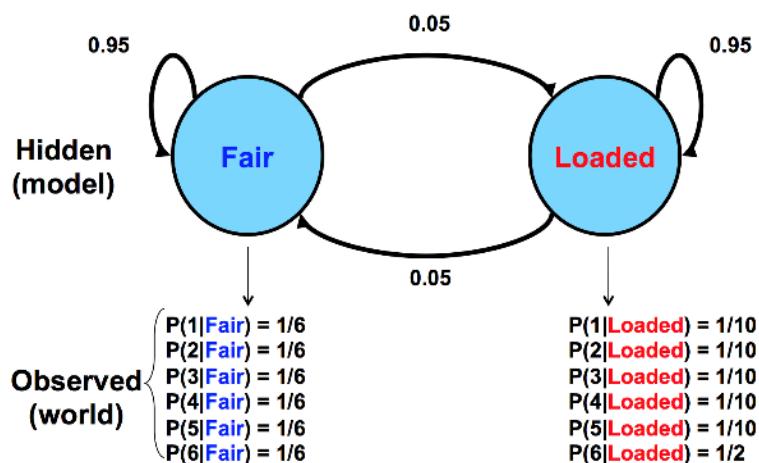


Figura 7.4: Estado de un dado de casino representado por un modelo Hidden Markov

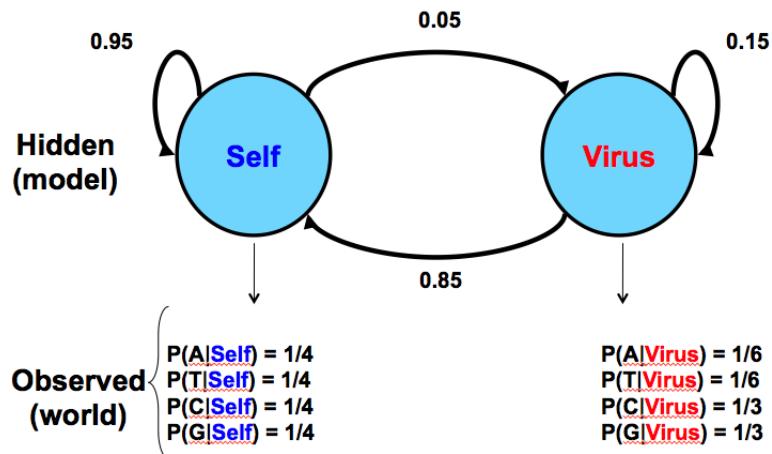


Figura 7.5: Fuentes potenciales de ADN: inyección viral vs producción normal. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

### Ejecución del modelo

Digamos que estamos en el casino y observar la secuencia de rollos que se da en la Figura 7.6. Nos gustaría saber si es más probable que el casino esté usando el dado justo o el dado cargado.



Figura 7.6: Una posible secuencia de rodillos de troquel observados.

Veamos una secuencia particular de rollos.

Por lo tanto, consideraremos dos posibles secuencias de estados en el HMM subyacente, una en la que el distribuidor siempre está usando un dado justo, y la otra en la que el distribuidor siempre está usando un dado cargado. Consideraremos cada ruta de ejecución para entender las implicaciones. Para cada caso, calculamos la probabilidad conjunta de un resultado observado con esa secuencia de estados subyacentes.

En el primer caso, donde suponemos que el distribuidor siempre está utilizando un dado justo, las probabilidades de transición y emisión se muestran en la Figura 7.7. La probabilidad de esta secuencia de estados y emisiones observadas es producto de términos que pueden agruparse en tres componentes:  $1/2$ , la probabilidad de comenzar con el dado justo;  $(1/6)^{10}$ , la probabilidad de la secuencia de rollos si siempre usamos el dado justo; y por último  $(0.95)^9$ , la probabilidad de que siempre sigamos usando el dado justo.

En este modelo, asumimos  $\pi = F, F, F, F, F, F, F, F, F, F$ , y observamos  $x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$ .

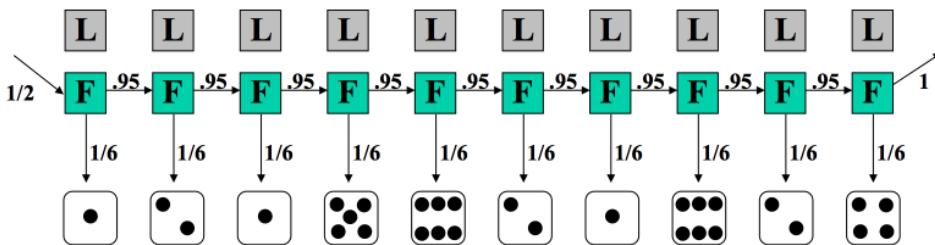


Figura 7.7: Ejecutando el modelo: probabilidad de una secuencia, la ruta dada consiste en todos los dados justos

Ahora podemos calcular la probabilidad conjunta de  $x$  y  $\pi$  de la siguiente manera:

```
\begin{aligned}
&\text{P}(x, \pi) = \text{P}(x | \pi) \text{P}(\pi) \\
&= \frac{1}{2} \times \left( \frac{1}{6} \right)^{10} \times (0.95)^9
\end{aligned}
```

$\&=5.2 \times 10^{-9}$   
 $\backslash end{alineado} \nonumber]$

Con una probabilidad tan pequeña, este podría parecer un caso extremadamente improbable. En la actualidad, la probabilidad es baja porque hay muchas posibilidades igualmente probables, y ningún resultado es probable a priori. La cuestión no es si esta secuencia de estados ocultos es probable, sino si es más probable que las alternativas.

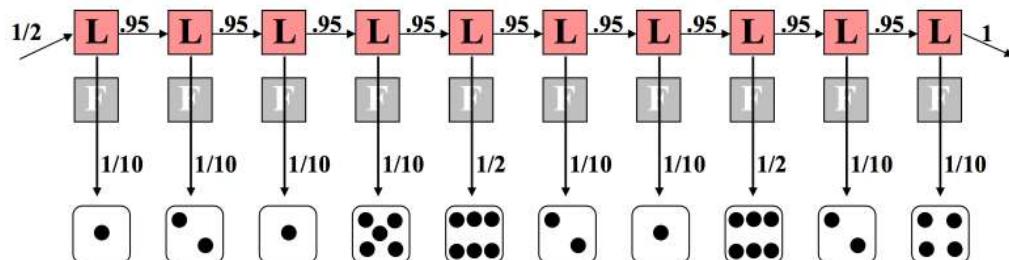


Figura 7.8: Ejecutando el modelo: probabilidad de una secuencia, la ruta dada consiste en todos los dados cargados

Consideremos el extremo opuesto donde el distribuidor siempre usa un dado cargado, como se muestra en la Figura 7.8. Esto tiene un cálculo similar excepto que observamos una diferencia en el componente de emisión. Esta vez, 8 de los 10 rollos llevan una probabilidad de  $1/10$  porque el dado cargado desfavorece a los no seis. Los dos rollos restantes de seis tienen cada uno una probabilidad de  $1/2$  de ocurrir. Nuevamente multiplicamos todas estas probabilidades juntas según principios de independencia y condicionamiento. En este caso, los cálculos son los siguientes:

```
\[ start {align*}
P(x,\ pi) &=\frac{1}{2} \times P(1 \mid L) \times P(L \mid L) \times P(2 \mid L) \dots \\
&=\frac{1}{2} \times \left(\frac{1}{10}\right)^8 \times \left(\frac{1}{2}\right)^2 \times (0.95)^9 \times 7.9 \times 10^{-10}
\end{align*}]
```

Anote la diferencia en los exponentes. Si hacemos una comparación directa, podemos decir que la situación en la que se usa un dado justo a lo largo de la secuencia es de  $52 \times 10^{-10}$  (en comparación con  $7.9 \times 10^{-10}$  con el dado cargado).

Por lo tanto, es seis veces más probable que se utilizó el dado justo que el dado cargado. Esto no es demasiado sorprendente—dos rollos de cada diez que arrojan un 6 no está muy lejos del esperado número 1.7 con el dado justo, y más lejos del esperado número 5 con el dado cargado.

### Agregando complejidad

Ahora imagina el caso más complejo, e interesante, donde el distribuidor cambia el dado en algún momento durante la secuencia. Hacemos una conjectura a un modelo subyacente basado en esta premisa en la Figura 7.9.

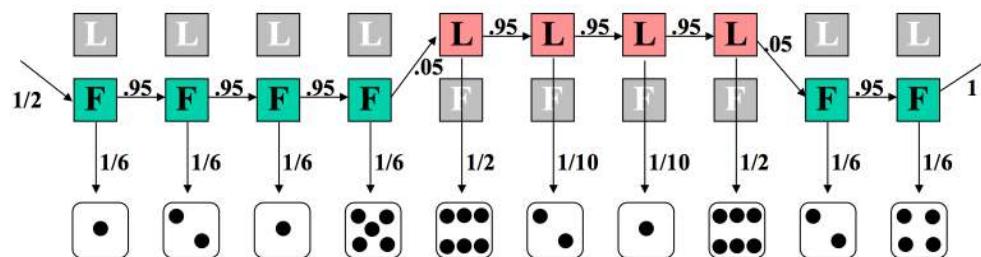


Figura 7.9: Ejecuciones parciales y commutación de matrices

Nuevamente, podemos calcular la probabilidad de la probabilidad conjunta de esta secuencia de estados y observaciones. Aquí, seis de los rollos se calculan con el troquel justo, y cuatro con el cargado. Además, ya no todas las probabilidades de transición son del 95%. Los dos swaps (entre justos y cargados) tienen cada uno una probabilidad del 5%.

```
\[ start {alineado}
P(x,\ pi) &=\frac{1}{2} \times P(1 \mid L) \times P(L \mid L) \times P(2 \mid L) \dots
```

$\&= \frac{1}{2} \times \left( \frac{1}{10} \times 0.95 \times 0.05 \right)^2 \times \frac{1}{6} \times 0.95 \times 0.05 \times 2 \times 10^{-11}$   
 $\&= 4.67 \times 10^{-11}$   
 $\backslash end{alineado}\nonumber\]$

## Volver a Biología

Ahora que hemos formalizado los HMM, queremos utilizarlos para resolver algunos problemas biológicos reales. De hecho, los HMM son una gran herramienta para el análisis de secuencias génicas, porque podemos ver una secuencia de ADN como emitida por una mezcla de modelos. Estos pueden incluir intrones, exones, factores de transcripción, etc. Si bien podemos tener algunos datos de muestra que emparejan modelos con secuencias de ADN, en el caso de que empiezamos de nuevo con una nueva pieza de ADN, podemos usar HMM para atribuir algunos modelos potenciales al ADN en cuestión. Primero presentaremos un ejemplo sencillo y lo pensaremos un poco. Luego, discutiremos algunas aplicaciones de HMM en la resolución de preguntas biológicas interesantes, antes de finalmente describir las técnicas HMM que resuelven los problemas que surgen en tal análisis de primer trato/nativo.

Un ejemplo sencillo: Encontrar regiones ricas en GC

Imagínese el siguiente escenario: estamos tratando de encontrar regiones ricas en GC modelando secuencias de nucleótidos extraídas de dos distribuciones diferentes: fondo y promotor. Las regiones de fondo tienen una distribución uniforme de 0.25 para cada una de A, T, G, C. Las regiones promotoras tienen probabilidades: A: 0.15, T: 0.13, G: 0.30, C: 0.42. Dado un nucleótido observado, no podemos decir nada sobre la región de la que se originó, ya que cualquiera de las regiones emitirá cada nucleótido con cierta probabilidad. Podemos aprender estas probabilidades de estado inicial con base en probabilidades de estado estacionario. Al observar una secuencia, queremos identificar qué regiones se originan a partir de una distribución de fondo (B) y qué regiones son de un modelo promotor (P).

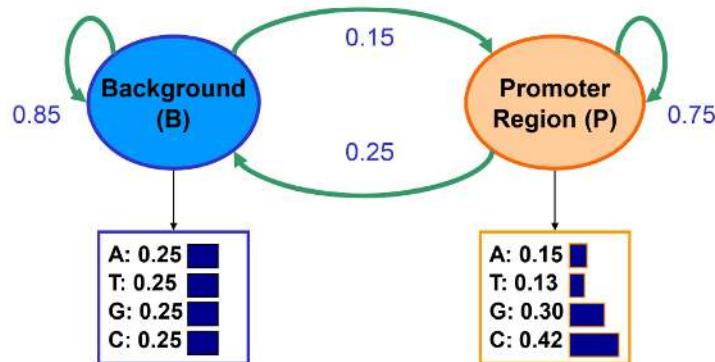


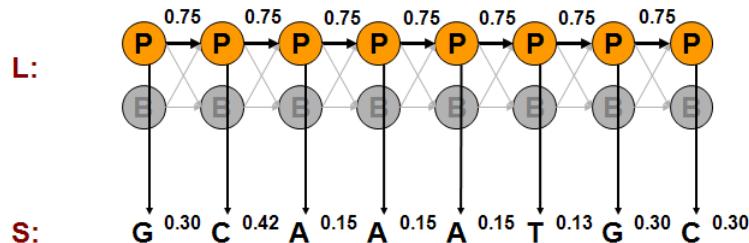
Figura 7.10: HMMS como modelo generativo para encontrar regiones ricas en GC. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Se nos dan las probabilidades de transición y emisión basadas en abundancia relevante y longitud promedio de regiones donde  $x$  = vector de emisiones observables que consiste en símbolos del alfabeto {A, T, G, C};  $\pi$  = vector de estados en una ruta (e.g. BPPBP);  $\pi^*$  = probabilidad máxima de generar ese camino. En nuestra interpretación de secuencia, la ruta de máxima verosimilitud se encontrará incorporando todas las probabilidades de emisión y transición por programación dinámica.

Los HMM son modelos generativos, en que un HMM da la probabilidad de emisión dado un estado (usando la Regla de Bayes), esencialmente diciéndote cuán probable es que el estado genere esas secuencias. Así que siempre podemos ejecutar un modelo generativo para las transiciones entre estados y comenzar en cualquier lugar. En Cadenas de Markov, el próximo estado dará diferentes resultados con diferentes probabilidades. No importa qué estado sea el siguiente, en el siguiente estado, el siguiente símbolo seguirá saliendo con diferentes probabilidades. Los HMM son similares: Se puede elegir un estado inicial basado en el vector de probabilidad inicial. En el ejemplo anterior, comenzaremos en el estado B con alta probabilidad ya que la mayoría de las localizaciones no corresponden a regiones promotoras. Luego se dibuja una emisión de la P ( $X|B$ ). Cada nucleótido ocurre con probabilidad 0.25 en el estado de fondo. Digamos que el nucleótido muestreado es una G. La distribución de los estados

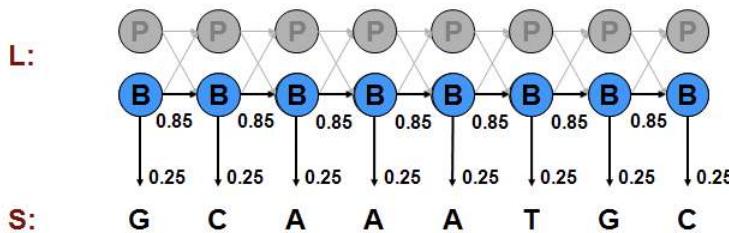
subsiguientes depende únicamente del hecho de que estamos en estado de fondo y es independiente de esta emisión. Entonces tenemos que la probabilidad de permanecer en el estado B es de 0.85 y la probabilidad de transición al estado P es de 0.15, y así sucesivamente.

Podemos calcular la probabilidad de una de esas generaciones multiplicando las probabilidades de que el modelo haga exactamente las elecciones que asumimos. Considere los ejemplos mostrados en las Figuras 7.11, 7.12 y 7.13.



$$\begin{aligned}
 P(x, \pi) &= a_p * e_p(G) * a_{pp} * e_p(G) * a_{pp} * e_p(C) * a_{pp} * e_p(A) * a_{pp} * \dots \\
 &= a_p * (0.75)^7 * (0.15)^3 * (0.13)^1 * (0.30)^2 * (0.42)^2 \\
 &= 9.3 * 10^{-7}
 \end{aligned}$$

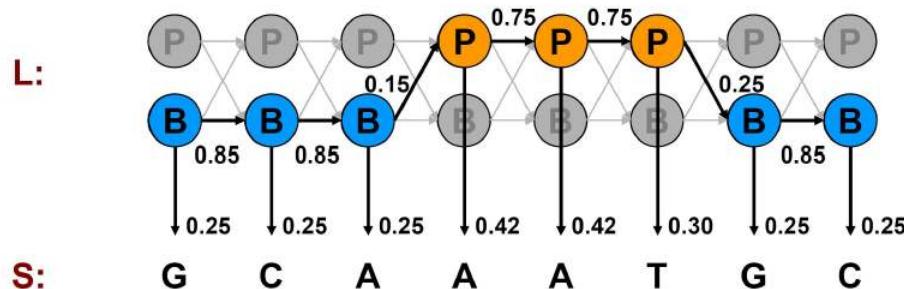
Figura 7.11: Probabilidad de seq, ruta si todo promotor



$$\begin{aligned}
 P &= P(G | B)P(B_1 | B_0)P(C | B)P(B_2 | B_1)P(A | B)P(B_3 | B_2) \dots P(C | B_7) \\
 &= (0.85)^7 \times (0.25)^8 \\
 &= 4.9 \times 10^{-6}
 \end{aligned}$$

A: 0.25	
T: 0.25	
G: 0.25	
C: 0.25	

Figura 7.12: Probabilidad de seq, ruta si todo fondo



$$\begin{aligned}
 P &= P(G|B)P(B_1|B_0)P(C|B)P(B_2|B_1)P(A|B)P(P_3|B_2)\dots P(C|B_7) \\
 &= (0.85)^3 \times (0.25)^6 \times (0.75)^2 \times (0.42)^2 \times 0.30 \times 0.15 \\
 &= 6.7 \times 10^{-7}
 \end{aligned}$$

Figura 7.13: Probabilidad de seq, secuencia de ruta si se mezcla

Podemos calcular la probabilidad conjunta de una secuencia particular de estados correspondientes a las emisiones observadas como hicimos en los ejemplos del Casino:

```
\[ comenzar {alineado}
P\ izquierda (x,\ pi_{\ P}\ derecha) &= a_{\ P}\ veces e_{\ P} (G)\ veces a_{\ P}\ veces e_{\ P} (G)\ veces\ cdots\
&= a_{\ P}\ veces (0.75)^7 \times (0.15)^3 \times (0.13)^2 \times (0.3)^2 \times (0.42)^2 \
&= 9.3 \times 10^{-7}
P\ izquierda (x,\ pi_{\ B}\ derecha) &= (0.85)^7 \times (0.25)^8 \
&= 4.9 \times 10^{-6}
P\ izquierda (x,\ pi_{\ \text{mixto}}\ derecha) &= (0.85)^3 \times (0.25)^6 \times (0.75)^2 \times (0.42)^2 \times 0.3 \times 0.15 \
&= 6.7 \times 10^{-7}
\ end {alineado}\ nonumber]
```

La alternativa de fondo puro es la opción más probable de las posibilidades que hemos examinado. Pero, ¿cómo sabemos si es la opción más probable de todos los caminos posibles de los estados haber generado la secuencia observada?

El enfoque de fuerza bruta consiste en examinar en todos los caminos, probando todas las posibilidades y calculando sus probabilidades conjuntas  $P(x, \pi)$  como hicimos anteriormente. La suma de probabilidades de todas las alternativas es 1. Por ejemplo, si todos los estados son promotores,  $P(x, \pi) = 9.3 \times 10^{-7}$ . Si todas las emisiones son Gs,  $P(x, \pi) = 4.9 \times 10^{-6}$ . la mezcla de B's y P's como en la Figura 7.13,  $P(x, \pi) = 6.7 \times 10^{-7}$ ; que es pequeña porque se paga mucha penalización por las transiciones entre B's y P's que son exponenciales en longitud de secuencia. Por lo general, si observas más G, es más probable que esté en la región promotora y si observas más A y Ts, entonces es más probable que esté en segundo plano. Pero necesitamos algo más que solo observación para apoyar nuestra creencia. Veremos cómo podemos apoyar matemáticamente nuestra intuición en las siguientes secciones.

## Aplicación de HMM en Biología

Los HMM se utilizan para responder muchas preguntas biológicas interesantes. Algunas aplicaciones biológicas de los HMM se resumen en la Figura 7.14.

Application	Detection of GC-rich region	Detection of Conserved region	Detection of Protein coding exons	Detection of Protein coding conservation	Detection of Protein coding gene structures	Detection of chromatin states
Topology / Transitions	2 states, different nucleotide composition	2 states, different conservation levels	2 states, different tri-nucleotide composition	2 states, different evolutionary signatures	~20 states, different composition / conservation, specific structure	40 states, different chromatin mark combinations
Hidden States / Annotation	GC-rich / AT-rich	Conserved / non-Conserved	Coding (exon) / non-Coding (intron or intergenic)	Coding (exon) / non-Coding (intron or intergenic)	First / last / middle coding exon, UTRs, intron 1/2/3, intergenic, *(+,-) strand	Enhancer / Promoter / Transcribed / Repressed / Repetitive
Emissions / Observations	Nucleotides	Level of conservation	Triplets of nucleotides	64 x 64 matrix of codon substitution frequencies	Codons, nucleotides, splice sites, start/stop codons	Vector of chromatin mark frequencies

Figura 7.14: Algunas aplicaciones biológicas de HMM

This page titled [7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.4: Apply HMM to Real World- From Casino to Biology](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.5: Ajustes algorítmicos para HMM

Utilizamos HMM para tres tipos de operación: puntuación, decodificación y aprendizaje. Hablaremos de puntuación y decodificación en esta conferencia. Estas operaciones pueden ocurrir para una sola ruta o todas las rutas posibles. Para las operaciones de ruta única, nuestro enfoque está en descubrir el camino con la máxima probabilidad. Sin embargo, estamos interesados en una secuencia de observaciones o emisiones para todas las operaciones de ruta independientemente de sus trayectorias correspondientes.

### Apuntar

Apuntar en una sola trayectoria

El problema del casino deshonesto y el problema de Predicción de regiones ricas en GC descritos en la sección 7.4 son ejemplos de encontrar la puntuación de probabilidad correspondiente a una sola ruta. Para una sola ruta definimos el problema de puntuación de la siguiente manera:

- Entrada: Una secuencia de observaciones  $x = x_1 x_2 \dots x_n$  generada por un HMM  $M (Q, A, p, V, E)$  y una ruta de estados  $\pi = \pi_1 \pi_2 \dots \pi_n$ .
- Salida: Probabilidad conjunta,  $P(x, \pi)$  de observar  $x$  si la secuencia de estado oculto es  $\pi$ .

	<b>One path</b>	<b>All paths</b>
Scoring	1. Scoring $x$ , one path $P(x, \pi)$ Prob of a path, emissions	2. Scoring $x$ , all paths $P(x) = \sum_{\pi} P(x, \pi)$ Prob of emissions, over all paths
Decoding	3. Viterbi decoding $\pi^* = \operatorname{argmax}_{\pi} P(x, \pi)$ Most likely path	4. Posterior decoding $\pi^A = \{\pi_i \mid \pi_i = \operatorname{argmax}_k \sum_{\pi} P(\pi_i=k x)\}$ Path containing the most likely state at any time point.
Learning	5. Supervised learning, given $\pi$ $\Lambda^* = \operatorname{argmax}_{\Lambda} P(x, \pi   \Lambda)$ 6. Unsupervised learning. $\Lambda^* = \operatorname{argmax}_{\Lambda} \max_{\pi} P(x, \pi   \Lambda)$ Viterbi training, best path	6. Unsupervised learning $\Lambda^* = \operatorname{argmax}_{\Lambda} \sum_{\pi} P(x, \pi   \Lambda)$ Baum-Welch training, over all paths

Figura 7.15: Los seis ajustes algorítmicos para HMMS

El cálculo de ruta única es esencialmente la probabilidad de observar la secuencia dada sobre una ruta particular usando la siguiente fórmula:

$$P(x, \pi) = P(x|\pi) P(\pi)$$

Ya hemos visto los ejemplos de puntuación de ruta única en nuestro Casino Deshonesto y región rica en GC ejemplos.

Apuntar en todas las rutas

Definimos la versión all paths del problema de puntuación de la siguiente manera:

- Entrada: Una secuencia de observaciones  $x = x_1 x_2 \dots x_n$  generada por un HMM  $M (Q, A, p, V, E)$ .

- Salida: La probabilidad conjunta,  $P(x, \pi)$  de observar  $x$  sobre todas las secuencias posibles de estados ocultos  $\pi$ .

La probabilidad sobre todos los caminos  $\pi$  de estados ocultos de la secuencia dada de observaciones viene dada por la siguiente fórmula.

$$P(x) = \sum_{\pi} P(x, \pi)$$

Utilizamos esta puntuación cuando estamos interesados en conocer la probabilidad de una secuencia particular para un HMM dado. Sin embargo, el cálculo ingenuo de esta suma requiere considerar un número exponencial de caminos posibles. Posteriormente en la conferencia veremos cómo calcular esta cantidad en tiempo polinomial.

### 7.5.2 Decodificación

La decodificación responde a la pregunta: Dada alguna secuencia observada, ¿qué camino nos da la máxima probabilidad de observar esta secuencia? Formalmente definimos el problema de la siguiente manera:

- Decodificación sobre una sola ruta:

— Entrada: Una secuencia de observaciones  $x = x_1 x_2 \dots x_N$  generada por un HMM  $M(Q, A, p, V, E)$ .

— Salida: El camino más probable de los estados,  $\pi^* = \pi_1^* \pi_2^* \dots \pi_N^*$

- Decodificación en todas las rutas:

— Entrada: Una secuencia de observaciones  $x = x_1 x_2 \dots x_N$  generada por un HMM  $M(Q, A, p, V, E)$ .

— Salida: La ruta de los estados,  $\pi^* = \pi_1^* \pi_2^* \dots \pi_N^*$  que contiene el estado más probable en cada punto temporal.

En esta conferencia, veremos únicamente el problema de la decodificación a través de una sola ruta. El problema de la decodificación en todos los caminos se discutirá en la próxima conferencia.

Para el problema de decodificación de ruta única, podemos imaginar un enfoque de fuerza bruta donde calculamos las probabilidades conjuntas de una secuencia de emisión dada y todas las trayectorias posibles y luego seleccionamos la ruta con la probabilidad conjunta máxima. El problema es que hay un número exponencial de caminos y usar tal búsqueda de fuerza bruta para el camino de máxima verosimilitud entre todos los caminos posibles es muy lento y poco práctico. La programación dinámica se puede utilizar para resolver este problema. Formulemos el problema en el enfoque de programación dinámica.

Nos gustaría conocer la secuencia más probable de estados con base en la observación. Como entradas, se nos dan los parámetros del modelo  $e_i(s)$ , las probabilidades de emisión para cada estado,  $y_{ij}$ , las probabilidades de transición. También se da la secuencia de emisiones  $x$ . El objetivo es encontrar la secuencia de estados ocultos,  $\pi^*$ , que maximice la probabilidad conjunta con la secuencia dada de emisiones. Es decir,

$$\pi^* = \arg \max_{\pi} P(x, \pi)$$

Dada la secuencia emitida  $x$  podemos evaluar cualquier ruta a través de estados ocultos. No obstante, estamos buscando el mejor camino. Comenzamos por buscar la subestructura óptima de este problema.

Para un mejor camino, podemos decir que, el mejor camino a través de un estado dado debe contener dentro de él lo siguiente: • El mejor camino al estado anterior

- La mejor transición del estado anterior a este estado
- El mejor camino al estado final

Por lo tanto, el mejor camino se puede obtener en base al mejor camino de los estados anteriores, es decir, podemos encontrar una recurrencia para el mejor camino. El algoritmo Viterbi es un algoritmo de programación dinámica que se utiliza comúnmente para obtener la mejor ruta.

Ruta de estado más probable: el algoritmo de Viterbi

Supongamos que  $v_k(i)$  es la probabilidad conocida de que la ruta más probable termine en la posición (o instancia de tiempo)  $i$  en el estado  $k$  para cada  $k$ . Entonces podemos calcular las probabilidades correspondientes en el tiempo  $i + 1$  mediante la siguiente recurrencia.

$$v_i(i+1) = e_i(x_{i+1}) \max_k (a_{ki} v_k(i))$$

La ruta más probable  $\pi^*$ , o la P máxima ( $x, \pi$ ), se puede encontrar recursivamente. Suponiendo que conocemos  $v_j(i-1)$ , la puntuación de la ruta máxima hasta el tiempo  $i-1$ , necesitamos aumentar el cálculo para el siguiente paso de tiempo. La nueva ruta de puntuación máxima para cada estado depende de

- La puntuación máxima de los estados anteriores
- La probabilidad de transición
- La probabilidad de emisión.

En otras palabras, la nueva puntuación máxima para un estado particular en el tiempo  $i$  es la que maximiza la transición de todos los estados previos posibles a ese estado particular (la penalización de transición multiplicada por sus puntuaciones previas máximas multiplicadas por la probabilidad de emisión en el momento actual).

Todas las secuencias tienen que comenzar en el estado 0 (el estado de inicio). Al mantener los punteros hacia atrás, la secuencia de estado real se puede encontrar retrocediendo. La solución de este problema de Programación Dinámica es muy similar a los algoritmos de alineación que se presentaron en conferencias anteriores.

A continuación se resumen los pasos del algoritmo Viterbi [2]:

1. Inicialización( $i=0$ ) :  $v_0(0) = 1, v_k(0) = 0$  for  $k > 0$
2. Recursión( $i = 1 \dots N$ ) :  $v_k(i) = e_k(x_i) \max_j (a_{jk} v_j(i-1)); ptr_i(l) = \arg \max_j (a_{jk} v_j(i-1))$
3. Terminación:  $P(x, \pi^*) = \max_k v_k(N); \pi_N^* = \arg \max_k v_k(N)$
4. Traceback( $i = N \dots 1$ ) :  $\pi_{i-1}^* = ptr_i(\pi_i^*)$

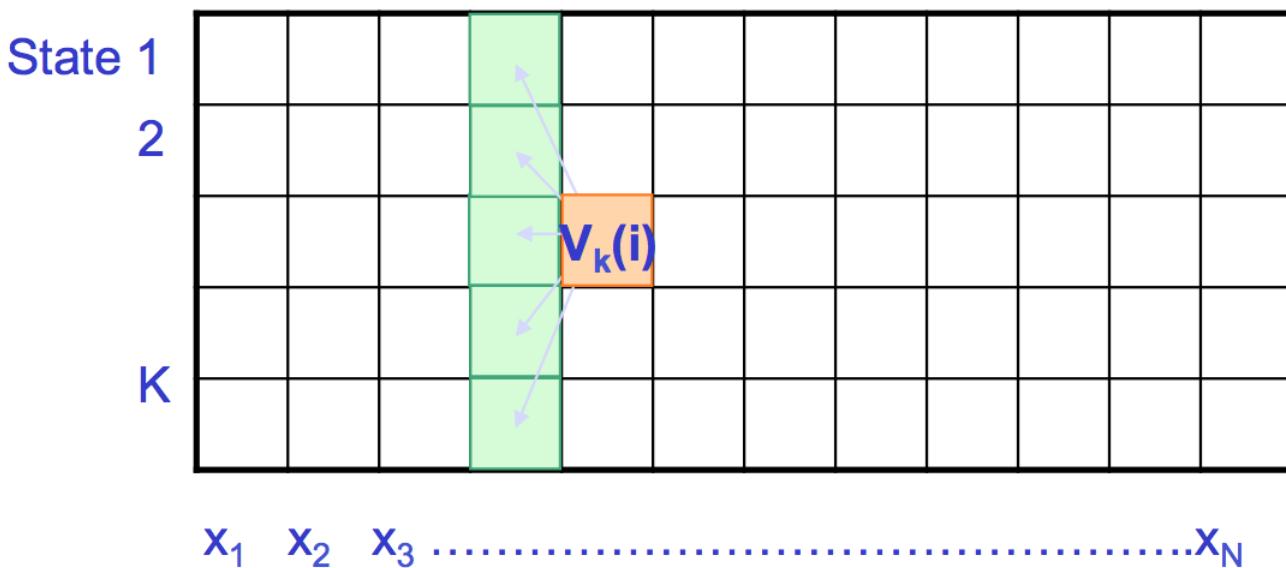


Figura 7.16: El algoritmo de Viterbi

Como podemos ver en la Figura 7.16, llenamos la matriz de izquierda a derecha y trazamos hacia atrás. Cada posición en la matriz tiene  $K$  estados a considerar y hay celdas  $KN$  en la matriz, por lo que el tiempo de cálculo requerido es  $O(K^2N)$  y el espacio requerido es  $O(KN)$  para recordar los punteros. En la práctica, utilizamos puntuaciones logarítmicas para el cálculo. Tenga en cuenta que el tiempo de ejecución se ha reducido de exponencial a polinomio.

## Evaluación

La evaluación se trata de responder a la pregunta: ¿Qué tan bien nuestro modelo de los datos capta los datos reales? Dada una secuencia  $x$ , muchas rutas pueden generar esta secuencia. La pregunta es ¿qué tan probable es la secuencia dada el modelo? En otras palabras, ¿es este un buen modelo? O bien, ¿qué tan bien captura el modelo las características exactas de una secuencia en

particular? Utilizamos la evaluación de HMM para responder a estas preguntas. Adicionalmente, con la evaluación podemos comparar diferentes modelos.

Demos primero una definición formal del problema de Evaluación.

- Entrada: Una secuencia de observaciones  $x = x_1 x_2 \dots x_N$  y un HMM  $M (Q, A, p, V, E)$ .
- Salida: La probabilidad de que  $x$  fue generada por  $M$  sumada en todas las rutas.

Sabemos que si se nos da un HMM podemos generar una secuencia de longitud  $n$  usando los siguientes pasos:

- Iniciar en el estado  $\pi_1$  según la probabilidad  $a_{0\pi_1}$  (obtenida usando vector,  $p$ ).
- Emite letra  $x_1$  según probabilidad de emisión  $e_{\pi_1}(x_1)$ .
- Ir al estado  $\pi_2$  según la probabilidad de transición  $a_{\pi_1|\pi_2}$
- Siga haciendo esto hasta emitir  $x_N$ .

Así podemos emitir cualquier secuencia y calcular su verosimilitud. Sin embargo, muchas secuencias de estados pueden emitir la misma  $x$ . Entonces, ¿cómo calculamos la probabilidad total de generar una  $x$  dada sobre todos los caminos? Es decir, nuestro objetivo es obtener la siguiente probabilidad:

$$P(x | M) = P(x) = \sum_{\pi} P(x, \pi) = \sum_{\pi} P(x | \pi)P(\pi)$$

El reto de obtener esta probabilidad es que hay demasiados caminos (un número exponencial) y cada camino tiene una probabilidad asociada. Un enfoque puede ser usar solo el camino de Viterbi e ignorar los demás, ya que ya sabemos cómo obtener este camino. Pero su probabilidad es muy pequeña ya que es sólo uno de los muchos caminos posibles. Es una buena aproximación sólo si tiene alta densidad de probabilidad. En otros casos, el camino Viterbi nos dará una aproximación inexacta. Alternativamente, el enfoque correcto para calcular la suma exacta iterativamente es a través del uso de programación dinámica. El algoritmo que hace esto se conoce como **Algoritmo Forward**.

### El algoritmo Forward

Primero derivamos la fórmula para la probabilidad hacia delante  $f(i)$ .

```
\begin{aligned}
f_{\{1\}}(i) &= P(\text{izquierda}(x_{\{1\}}) \text{lpuntos } x_{\{i\}}, p_i = \text{derecha}) \\
&\&= \sum_{\{1\}} p_{\{1\}} \text{lpuntos } p_{\{i-1\}} P(\text{izquierda}(x_{\{1\}}) \text{ldots } x_{\{i-1\}}, p_{\{1\}}, \text{ldots}, p_{\{i-2\}}, p_{\{i-1\}}, p_{\{i\}} = \text{derecha}) e_{\{1\}} \text{izquierda}(x_{\{i\}}) \text{derecha}) \\
&\&= \sum_{\{k\}} \sum_{\{1\}} p_{\{1\}} \text{lpuntos } p_{\{i-2\}} P(\text{izquierda}(x_{\{1\}}) \text{ldots } x_{\{i-1\}}, p_{\{1\}}, \text{ldots}, p_{\{i-2\}}, p_{\{i-1\}} = k) \text{derecha}) a_{\{k\}} e_{\{1\}} \text{izquierda}(x_{\{i\}}) \text{derecha}) \\
&\&= \sum_{\{k\}} f_{\{k\}}(i-1) a_{\{k\}} e_{\{1\}} \text{izquierda}(x_{\{i\}}) \text{derecha}) \\
&\&= e_{\{1\}} \text{izquierda}(x_{\{i\}}) \text{derecha}) \sum_{\{k\}} f_{\{k\}}(i-1) a_{\{k\}} \\
\end{aligned}
```

El algoritmo completo [2] se resume a continuación:

- Inicialización( $i = 0$ ) :  $f_0(0) = 1, f_k(0) = 0$  for  $k > 0$
- Iteración( $i = 1 \dots N$ ) :  $f_k(i) = e_k(x_i) \sum_j f_j(i-1) a_{jk}$
- Terminación:  $P(x, \pi^*) = \sum_k f_k(N)$

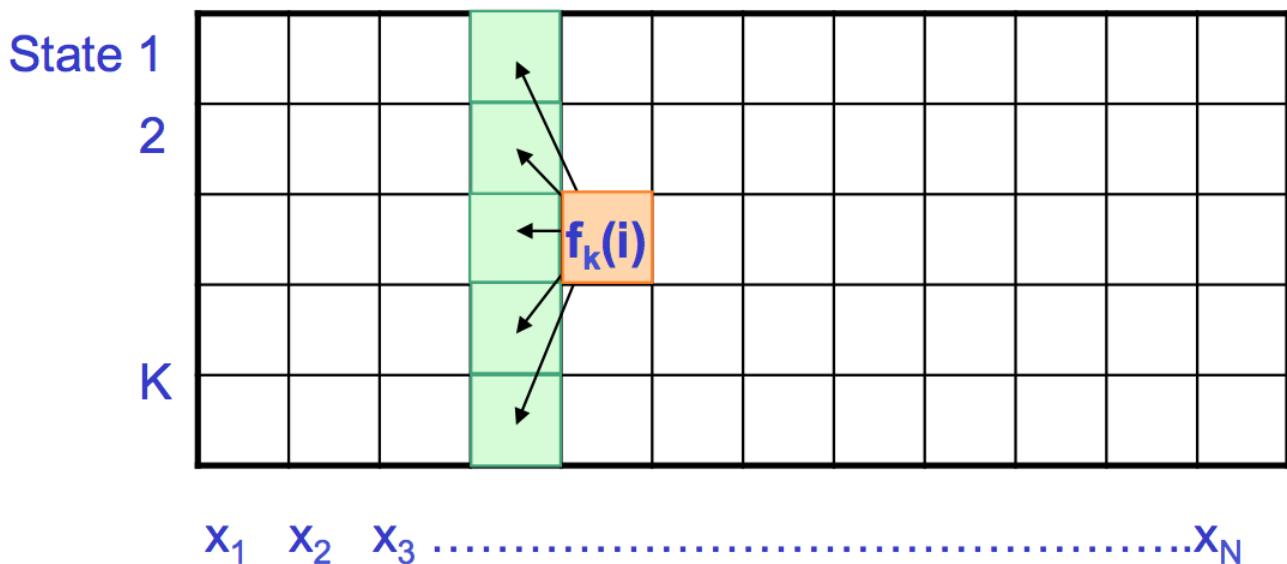


Figura 7.17: El algoritmo Forward

De la Figura 7.17, se puede observar que el algoritmo Forward es muy similar al algoritmo de Viterbi. En el algoritmo Forward, se utiliza la suma en lugar de la maximización. Aquí podemos reutilizar cálculos del problema anterior incluyendo penalización de emisiones, penalización de transiciones y sumas de estados anteriores. El tiempo de cálculo requerido es  $O(K^2 N)$  y el espacio requerido es  $O(KN)$ . El inconveniente de este algoritmo es que en la práctica, tomar la suma de logs es difícil; por lo tanto, en su lugar se utilizan aproximaciones y escalado de probabilidades.

This page titled [7.5: Ajustes algorítmicos para HMM](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.5: Algorithmic Settings for HMMs](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo?

La respuesta a esta pregunta es - ¡Sí, podemos! Pero, ¿cómo? Recordemos que, los modelos de Markov no tienen memoria. Es decir, toda la memoria del modelo está encerrada en estados. Entonces, para almacenar información adicional, debemos incrementar el número de estados. Ahora, volvamos al ejemplo biológico que dimos en la Sección 7.4.2. En nuestro modelo, las emisiones estatales dependían únicamente del estado actual. Y, el estado actual codificaba sólo un nucleótido. Pero, ¿y si queremos que nuestro modelo cuente frecuencias de di-nucleótidos (para islas CpG <sup>1</sup>), o, frecuencias de tri-nucleótidos (para codones), o frecuencias de di-codones que involucran seis nucleótidos? Tenemos que ampliar el número de estados.

Por ejemplo, el último nucleótido visto puede incorporarse a la “memoria” del HMM dividiendo los estados más y menos de nuestro HMM de alto GC/bajo GC en múltiples estados: uno por cada combinación nucleótido/región, como en la Figura 7.18.

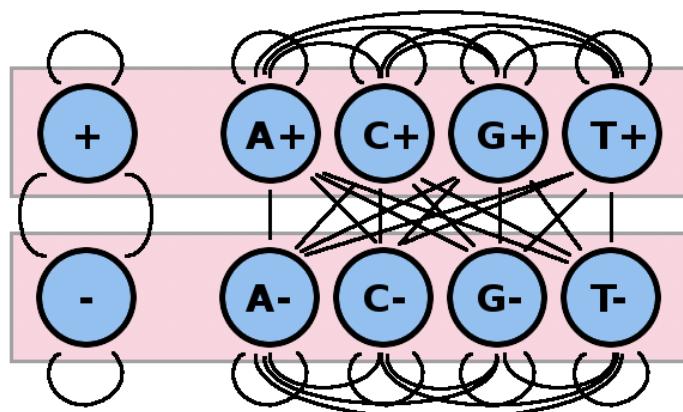


Figura 7.18: Islas CpG - Incorporación de Memoria

Pasar de dos a ocho estados permite conservar la memoria del último nucleótido observado, a la vez que se distingue entre dos regiones distintas. Cuatro nuevos estados corresponden ahora a cada uno de los dos estados originales en el HMM de GC alto/bajo. Mientras que los pesos de transición en el HMM más pequeño se basaron puramente en las frecuencias de los nucleótidos individuales, ahora en el más grande, se basan en frecuencias de di-nucleótidos.

Con esta potencia añadida, ciertas secuencias de di-nucleótidos, como las islas CpG, se pueden modelar específicamente: a la transición de C+ a G+ se le puede asignar mayor peso que la transición de A+ a G+. Además, las transiciones entre + y - se pueden modelar más específicamente para reflejar la frecuencia (o infrecuencia) de secuencias de di-nucleótidos particulares dentro de una u otra.

El proceso de agregar memoria a un HMM puede generalizarse y se puede agregar más memoria para permitir el reconocimiento de secuencias de mayor longitud. Por ejemplo, podemos detectar tripletes de codones con 32 estados, o sextupletos de di-codones con 2048 estados. La memoria dentro del HMM permite una especificidad cada vez más personalizada en el escaneo.

<sup>1</sup> CpG significa C-fosfato-g. Entonces, isla CpG se refiere a una región donde el di-nucleótido GC aparece en la misma cadena.

This page titled [7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo?](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.6: An Interesting Question- Can We Incorporate Memory in Our Model?](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#).  
Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 7.7: Lectura adicional, ¿qué hemos aprendido?

### Distribuciones de Longitud de Estados y Modelos Generalizados de Markov Ocultos

Dada una cadena de Markov con la transición de cualquier estado al estado final teniendo probabilidad  $\tau$ , la probabilidad de generar una secuencia de longitud  $L$  (y luego terminar con una transición al estado final) viene dada por:

$$\tau(1 - \tau)^{L-1}$$

De igual manera, en los HMM que hemos estado examinando, la longitud de los estados será exponencialmente distribuida, lo que no es apropiado para muchos fines. (Por ejemplo, en una secuencia genómica, una distribución exponencial no captura con precisión las longitudes de genes, exones, intrones, etc.). ¿Cómo podemos construir un modelo que no emita secuencias de estados con una distribución exponencial de longitudes? Supongamos que queremos asegurarnos de que nuestra secuencia tiene una longitud exactamente 5. Podríamos construir una secuencia de cinco estados con un solo camino permitido por las probabilidades de transición. Si incluimos un bucle self en uno de los estados, emitiremos secuencias de longitud mínima 5, con secuencias más largas distribuidas exponencialmente. Supongamos que tenemos una cadena de  $n$  estados, con todas las cadenas comenzando con el estado  $\pi_1$  y haciendo la transición a un estado final después de  $\pi_n$ . También supongamos que la probabilidad de transición entre el estado  $\pi_i$  y  $\pi_{i+1}$  es  $1-p$ , mientras que la probabilidad de autotransición del estado  $\pi_i$  es  $p$ . La probabilidad de que una secuencia generada por esta cadena de Markov tenga longitud  $L$  viene dada por:

$$\begin{aligned} & \left[ \begin{array}{l} L-1 \\ n-1 \end{array} \right] \\ & p^{L-n} (1-p)^n \end{aligned}$$

A esto se le llama distribución binomial negativa.

De manera más general, podemos adaptar HMM para producir secuencias de salida de longitud arbitraria. En un Modelo Generalizado de Markov Ocultos [1] (también conocido como modelo semi-Markov oculto), la salida de cada estado es una cadena de símbolos, en lugar de un símbolo individual. La longitud y el contenido de esta cadena de salida se pueden elegir en función de una distribución de probabilidad. Muchas herramientas de búsqueda de genes se basan en modelos generalizados ocultos de Markov.

### Campos aleatorios condicionales

El modelo de campo aleatorio condicional es un modelo gráfico probabilístico discriminativo no dirigido que se usa alternativamente a los HMM. Se utiliza para codificar relaciones conocidas entre observaciones y construir interpretaciones consistentes. A menudo se usa para etiquetar o analizar datos secuenciales. Es ampliamente utilizado en la búsqueda de genes. Los siguientes recursos pueden ser útiles para aprender más sobre los CRF:

- Conferencia sobre Campos Aleatorios Condicionales a partir de Modelos Gráficos Probabilísticos Curso: class. coursera.org/pgm/lecture/preview/33. Para antecedentes, es posible que también desee ver los dos segmentos anteriores, en redes de Markov por pares y distribuciones generales de Gibbs.
- Campos aleatorios condicionales en biología: www.cis.upenn.edu/~pereira/papers/crf.pdf
- Campos aleatorios condicionales tutorial: <http://people.cs.umass.edu/~mccallum...f-tutorial.pdf>

### ¿Qué hemos aprendido?

En esta sección, los principales contenidos que cubrimos son los siguientes:

- Primero, introdujimos la motivación detrás de la adopción de Modelos Ocultos de Markov en nuestro análisis de anotación genómica.
- Segundo, formalizamos Markov Chains y HMM bajo la luz del ejemplo de predicción del clima.
- Tercero, tenemos una idea de cómo aplicar HMM en datos del mundo real al observar los problemas de Casino Deshonesto y región rica en CG.
- En cuarto lugar, introdujimos sistemáticamente ajustes algorítmicos de HMM y entramos en detalles de tres de ellos:

- Scoring: scoring over single path
- Scoring: scoring over all paths
- Decodificación: codificación Viterbi para determinar el camino más probable
- Finalmente, discutimos la posibilidad de introducir la memoria en el análisis de HMM y brindamos lecturas adicionales para los lectores interesados.

## Bibliografía

[1] Introducción a los GHMM: [www.cs.tau.ac.il/~rshamir/algmb/00/scribe00/html/lec07/node28.html](http://www.cs.tau.ac.il/~rshamir/algmb/00/scribe00/html/lec07/node28.html).

[2] R. Durbin, S. Eddy, A. Krogh y G. Mitchison. Análisis de secuencias biológicas. undécima edición, 2006.

---

This page titled [7.7: Lectura adicional, ¿qué hemos aprendido?](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [7.7: Further Reading, What Have We Learned?](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 8: Modelos Ocultos de Markov II-Decodificación posterior y aprendizaje

- 8.1: Revisión de la conferencia anterior
- 8.2: Decodificación posterior
- 8.3: Memoria de codificación en un HMM- Detección de islas CpG
- 8.4: Aprendizaje
- 8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín
- 8.6: Direcciones actuales de investigación, ¿qué hemos aprendido? , Bibliografía
- 8.9 ¿Qué hemos aprendido?

Bibliografía

---

This page titled [8: Modelos Ocultos de Markov II-Decodificación posterior y aprendizaje](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 8.1: Revisión de la conferencia anterior

### Introducción a los modelos ocultos de Markov

En la última conferencia, nos familiarizamos con el concepto de cadenas de Markov de tiempo discreto y Modelos Ocultos de Markov (HMM). En particular, una cadena de Markov es un proceso aleatorio discreto que se ajusta a la propiedad de Markov, es decir, que la probabilidad del siguiente estado depende únicamente del estado actual; esta propiedad también se denomina frecuentemente “falta de memoria”. Para modelar cómo los estados cambian de paso a paso, la cadena de Markov utiliza una matriz de probabilidades de transición. Además, se caracteriza por una correspondencia uno a uno entre los estados y los símbolos observados; es decir, el estado determina completamente todos los observables relevantes. De manera más formal, una cadena de Markov está completamente definida por las siguientes variables:

- $\pi_i \in Q$ , el estado en el  $i^{\text{ésimo}}$  paso en una secuencia de estados finitos  $Q$  de longitud  $N$  que puede contener un valor de un alfabeto finito  $\sigma$  de longitud  $K$
- $a_{jk}$ , la probabilidad de transición de pasar del estado  $j$  al estado  $k$ ,  $P(\pi_i = k | \pi_{i-1} = j)$ , para cada  $j, k \in Q$
- $a_{0j} \in P$ , la probabilidad de que el estado inicial sea  $j$

Los ejemplos de cadenas de Markov abundan en la vida cotidiana. En la última conferencia, consideramos el ejemplo canónico de un sistema meteorológico en el que cada estado es lluvia, nieve, sol o nubes y los observables del sistema corresponden exactamente al estado subyacente: no hay nada que desconozcamos al hacer una observación, como la observación, es decir, si es soleado o lloviendo, determina completamente el estado subyacente, es decir, si está soleado o lloviendo. Supongamos, sin embargo, que estamos considerando el clima ya que está determinado probabilísticamente por las estaciones -por ejemplo, nieva más a menudo en invierno que en primavera- y supongamos además que estamos en la antigüedad y aún no tuvimos acceso al conocimiento sobre cuál es la temporada actual. Consideremos ahora el problema de tratar de inferir la temporada (el estado oculto) del clima (lo observable). Existe alguna relación entre temporada y clima tal que podemos usar información sobre el clima para hacer inferencias sobre qué estación es (si nieva mucho, probablemente no sea verano); esta es la tarea que los HMM buscan emprender. Así, en esta situación, los estados, las estaciones, se consideran “ocultos” y ya no comparten una correspondencia uno a uno con los observables, el clima. Este tipo de situaciones requieren una generalización de las cadenas de Markov conocidas como Modelos Ocultos de Markov (HMM).

#### ¿Sabías?

Las cadenas de Markov pueden ser consideradas como WYSIWYG - Lo que ves es lo que obtienes

Los HMM incorporan elementos adicionales para modelar la desconexión entre los observables de un sistema y los estados ocultos. Para una secuencia de longitud  $N$ , cada estado observable es reemplazado por un estado oculto (la temporada) y un carácter emitido desde ese estado (el clima). Es importante señalar que los caracteres de cada estado se emiten de acuerdo a una serie de probabilidades de emisión (digamos que hay un 50% de probabilidad de nieve, 30% de probabilidad de sol y 20% de probabilidad de lluvia durante el invierno). Más formalmente, los dos descriptores adicionales de un HMM son:

- $x_i \in X$ , la emisión en el paso  $i$  en una secuencia de caracteres finitos  $X$  de longitud  $N$  que puede contener un carácter de un conjunto finito de símbolos de observación  $v_i \in V$
- $e_k(v_i) \in E$ , la probabilidad de emisión de emitir carácter  $v_i$  cuando el estado es  $k$ ,  $P(x_i = v_i | \pi_i = k)$  En resumen, un HMM se define por las siguientes variables:
  - $a_{jk}, e_k(v_i)$  y  $a_{0j}$  que modelan el proceso aleatorio discreto
  - $\pi_i$ , la secuencia de estados ocultos
  - $x_i$ , la secuencia de emisiones observadas

### Aplicaciones genómicas de los HMM

La siguiente figura muestra algunas aplicaciones genómicas de los HMM

Application	Detection of GC-rich region	Detection of Conserved region	Detection of Protein coding exons	Detection of Protein coding conservation	Detection of Protein coding gene structures	Detection of chromatin states
<b>Topology / Transitions</b>	2 states, different nucleotide composition	2 states, difference conservation levels	2 states, different tri-nucleotide composition	2 states, different evolutionary signatures	~20 states, different composition / conservation, specific structure	40 states, different chromatin mark combinations
<b>Hidden States / Annotation</b>	GC-rich / AT-rich	Conserved/ non-Conserved	Coding (exon) / non-Coding (intron or intergenic)	Coding (exon) / non-Coding (intron or intergenic)	First / last / middle coding exon, UTRs, intron 1/2/3, intergenic, *(+,-) strand	Enhancer / Promoter / Transcribed / Repressed / Repetitive
<b>Emissions / Observations</b>	Nucleotides	Level of conservation	Triplets of nucleotides	64 x 64 matrix of codon substitution frequencies	Codons, nucleotides, splice sites, start/stop codons	Vector of chromatin mark frequencies

Figura 8.1: Aplicaciones genómicas de los HMM

Las sutilezas de algunas de las aplicaciones que se muestran en la figura 8.1 incluyen:

- Detección de conservación de codificación de proteínas

Esto es similar a la aplicación de detectar exones codificantes de proteínas porque las emisiones tampoco son nucleótidos sino diferentes en el sentido de que, en lugar de emitir codones, se emiten frecuencias de sustitución de los codones.

- Detección de estructuras génicas codificantes de proteínas

Aquí, es importante que diferentes estados modele los exones primero, último y medio de forma independiente, porque tienen distintas características estructurales relevantes: por ejemplo, el primer exón en una transcripción pasa por un codón de inicio, el último exón pasa por un codón de parada, etc., y para hacer las mejores predicciones, nuestro modelo debe codificar estas características. Esto difiere de la aplicación de detección de exones codificantes de proteínas porque en este caso, la posición del exón no es importante.

También es importante diferenciar entre los intrones 1,2 y 3 para que pueda recordarse el marco de lectura entre un exón y el siguiente exón, por ejemplo, si un exón se detiene en la posición del segundo codón, el siguiente tiene que comenzar en la posición del tercer codón. Por lo tanto, los estados intrónicos adicionales codifican la posición del codón.

- Detección de estados de cromatina Los modelos de estado de

cromatina son dinámicos y varían de un tipo de celda a otro por lo que cada tipo de celda tendrá su propia anotación. Serán discutidos con más detalle en la conferencia de genómica incluyendo estrategias para apilar/concatenar tipos de células.

## Descodificación Viterbi

Anteriormente, demostramos que cuando se le daba un HMM completo ( $Q, A, X, E, P$ ), la probabilidad de que el proceso aleatorio discreto produjera la serie proporcionada de estados ocultos y emisiones viene dada por:

$$P(x_1, \dots, x_N, \pi_1, \dots, \pi_N) = a_{0\pi_1} \prod_i e_{\pi_i}(x_i) a_{\pi_i \pi_{i+1}} \quad (8.1.1)$$

Esto corresponde a la probabilidad conjunta total,  $P(x, \pi)$ . Por lo general, sin embargo, los estados ocultos no se dan y deben inferirse; no nos interesa conocer la probabilidad de la secuencia observada dado un modelo subyacente de estados ocultos, sino que queremos que la secuencia observada infiera los estados ocultos, como cuando usamos la secuencia genómica de un organismo para inferir la ubicación de sus genes. Una solución a este problema de decodificación se conoce como el algoritmo de decodificación de Viterbi. Corriendo en  $O(K^2 N)$  tiempo y  $O(KN)$  espacio, donde  $K$  es el número de estados y  $N$  es la longitud de la secuencia observada, este algoritmo determina la secuencia de estados ocultos (la ruta  $\pi^*$ ) que maximiza la probabilidad conjunta de los observables y estados, es decir,  $P(x, \pi)$ . Esencialmente, este algoritmo define  $V_k(i)$  como la probabilidad de que la ruta más probable termine en el estado  $\pi_i = k$ , y utiliza el argumento de subestructura óptima que vimos en el módulo de alineación de secuencias del curso para calcular recursivamente  $V_k(i) = e_k(x_i) \times \max_j (V_{j-1}(i-1) a_{jk})$  en un algoritmo de programación dinámica.

## Algoritmo delantero

Volviendo por un momento al problema de 'puntuar' en lugar de 'decodificar', otro problema que podríamos querer abordar es el de, en lugar de calcular la probabilidad de que una sola ruta de estado oculto emita la secuencia observada, calcular la probabilidad total de que la secuencia sea producida por todos los posibles caminos. Por ejemplo, en el ejemplo del casino, si la secuencia de rollos es lo suficientemente larga, la probabilidad de cualquier secuencia observada y trayectoria subyacente es muy baja, incluso si es la combinación única secuencia-trayectoria más probable. En cambio, podemos querer tomar una actitud agnóstica hacia el camino y evaluar la probabilidad total de que la secuencia observada surja de alguna manera.

Para ello, proponemos el algoritmo Forward, que se describe en la Figura 8.2

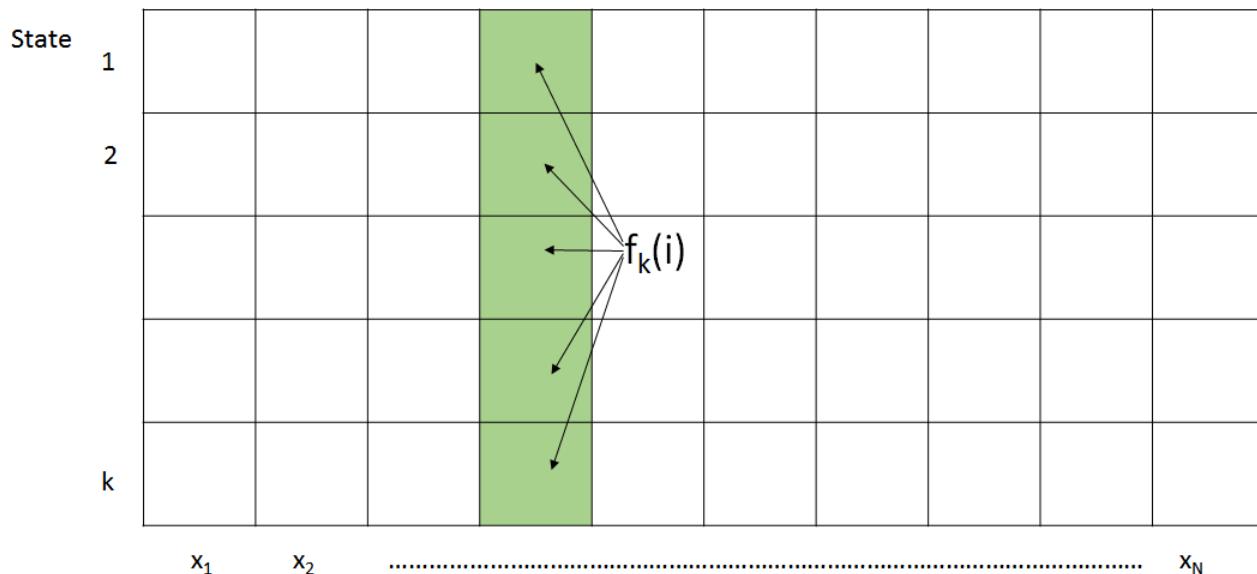


Figura 8.2: El Algoritmo Forward

```
\begin{array}{l}
\begin{array}{l}
\text{Entrada: } x=x_1 \dots x_N \\
\text{Inicialización:} \\
\quad f_0(0)=1, f_k(0)=0, \text{ para todos } k>0 \\
\text{Iteración:} \\
\quad f_k(i) = e_k \text{ izquierda}(x_i \text{ derecha}) \text{ veces} \sum_j a_{jk} f_j(i-1) \\
\end{array} \\
\text{Terminación:} \\
\quad \text{comenzar ecuación} \\
P \text{ izquierda}(x, \pi^*) \text{ derecha} = \sum_k f_k(N) \\
\text{final ecuación nonumber}
\end{array}
```

El algoritmo forward calcula primero la probabilidad conjunta de observar los primeros  $t$  caracteres emitidos y estar en estado  $k$  en el tiempo  $t$ . Más formalmente,

```
\begin{array}{l}
f_k(t) = P \left( \pi_t = k, x_1, \dots, x_t \right) \\
\end{array}
```

Dado que el número de caminos es exponencial en  $t$ , se debe emplear programación dinámica para resolver este problema. Podemos desarrollar una recursión simple para el algoritmo forward empleando la propiedad Markov de la siguiente manera:

```
\begin{array}{l}
f_k(t) = \sum_l P \left( \pi_t = k, x_1, \dots, x_t \mid \pi_{t-1} = l \right) = \sum_l P \left( \pi_t = k \mid \pi_{t-1} = l \right) f_l(t-1)
\end{array}
```

```
\ldots, x_{t-1},\pi_{t-1} = l\derecha) * P\izquierda(x_t,\pi_t\mediados\pi_{t-1}\derecha)
\final{ecuación}]
```

Reconociendo que el primer término corresponde a  $f_l(t - 1)$  y que el segundo término puede expresarse en términos de probabilidades de transición y emisión, esto lleva a la recursión final:

```
\[\comenzar{ecuación}
f_{\{k\}}(t) = e_{\{k\}}\izquierda(x_t\derecha)\sum_{\{l\}} f_{\{l\}}(t-1) * a_{\{l\}k}
\final{ecuación}]\]
```

Intuitivamente, se puede entender esta recursión de la siguiente manera: Cualquier ruta que esté en el estado  $k$  en el tiempo  $t$  debe haber venido de una ruta que estaba en el estado  $l$  en el tiempo  $t - 1$ . La contribución de cada uno de estos conjuntos de rutas es ponderada por el costo de la transición del estado  $l$  al estado  $k$ . También es importante señalar que el algoritmo Viterbi y el algoritmo forward comparten en gran medida la misma recursión. La única diferencia entre los dos algoritmos radica en el hecho de que el algoritmo de Viterbi, que busca encontrar solo la ruta más probable, utiliza una función de maximización, mientras que el algoritmo forward, que busca encontrar la probabilidad total de la secuencia sobre todas las rutas, usa una suma.

Ahora podemos calcular  $f_k(t)$  en base a una suma ponderada de todos los resultados del algoritmo forward tabulados durante el paso de tiempo anterior. Como se muestra en la Figura 8.2, el algoritmo directo se puede implementar fácilmente en una tabla de programación dinámica  $K \times N$ . La primera columna de la tabla se inicializa de acuerdo con las probabilidades de estado inicial  $a_{i0}$  y el algoritmo luego procede a procesar cada columna de izquierda a derecha. Debido a que hay entradas  $KN$  y cada entrada examina un total de  $K$  otras entradas, esto lleva a  $O(K^2N)$  complejidad de tiempo y  $O(KN)$  espacio.

Para ahora calcular la probabilidad total de una secuencia de caracteres observados bajo el HMM actual, necesitamos expresar esta probabilidad en términos del algoritmo forward da de la siguiente manera:

```
\[\start{ecuación}
P\izquierda(x_1,\ldots,x_n\derecha) = \sum_{\{l\}} P\izquierda(x_1,\ldots,x_n,\pi_{\{N\}} = l\derecha) = \sum_{\{l\}} f_{\{l\}}(N)
\end{ecuación}]\]
```

De ahí que la suma de los elementos en la última columna de la tabla de programación dinámica proporciona la probabilidad total de una secuencia observada de caracteres. En la práctica, dada una secuencia suficientemente larga de caracteres emitidos, las probabilidades de avance disminuyen muy rápidamente. Para eludir los problemas asociados con el almacenamiento de pequeños números de coma flotante, se utilizan probabilidades logarítmicas en los cálculos en lugar de las probabilidades mismas. Esta alteración requiere un ligero ajuste al algoritmo y el uso de una expansión de la serie Taylor para la función exponencial.

## Esta conferencia

- Esta conferencia discutirá la decodificación posterior, un algoritmo que nuevamente inferirá la secuencia de estado oculto  $\pi$  que maximiza una métrica diferente. En particular, encuentra el estado más probable en cada posición sobre todas las rutas posibles y lo hace usando tanto el algoritmo hacia adelante como hacia atrás.
- Posteriormente, mostraremos cómo codificar “memoria” en una cadena de Markov agregando más estados para buscar en un genoma islas CpG de dinucleótidos.
- Luego discutiremos cómo usar la estimación de parámetros de máxima probabilidad para el aprendizaje supervisado con un conjunto de datos etiquetado
- También veremos brevemente cómo usar el aprendizaje de Viterbi para la estimación no supervisada de los parámetros de un conjunto de datos sin etiquetar
- Finalmente, aprenderemos a usar la Maximización de Expectativa (EM) para la estimación no supervisada de parámetros de un conjunto de datos sin etiquetar donde el algoritmo específico para HMM se conoce como el algoritmo Baum-Welch.

---

This page titled [8.1: Revisión de la conferencia anterior](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [8.1: Review of previous lecture](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.2: Decodificación posterior

### Motivación

Aunque el algoritmo de decodificación de Viterbi proporciona un medio para estimar los estados ocultos subyacentes a una secuencia de caracteres observados, otro medio válido de inferencia es proporcionado por la decodificación posterior.

La decodificación posterior proporciona el estado más probable en cualquier momento. Para ganar algo de intuición para la decodificación posterior, veamos cómo se aplica a la situación en la que un casino deshonesto alterna entre un dado justo y cargado. Supongamos que entramos al casino sabiendo que el injusto muere se usa el 60 por ciento del tiempo. Con este conocimiento y sin troqueles, nuestra mejor suposición para el dado actual es obviamente el cargado. Despues de un rollo, la probabilidad de que se utilizó el dado cargado viene dada por

```
\begin{equation}
P(\text{die} = \text{loaded} | \text{roll} = k) = \frac{P(\text{die} = \text{loaded}) * P(\text{roll} = k | \text{die} = \text{loaded})}{P(\text{roll} = k)}
\end{equation}
```

Si en cambio observamos una secuencia de  $N$  troqueles, ¿cómo se realiza una inferencia similar? Al permitir que la información fluya entre los  $N$  rollos e influya en la probabilidad de cada estado, la decodificación posterior es una extensión natural de la inferencia anterior a una secuencia de longitud arbitraria. De manera más formal, en lugar de identificar una sola ruta de máxima verosimilitud, la decodificación posterior considera la probabilidad de que cualquier camino se encuentre en el estado  $k$  en el tiempo  $t$  dados todos los caracteres observados, es decir,  $P(\pi_t = k | x_1, \dots, x_n)$ . El estado que maximiza esta probabilidad para un tiempo dado se considera entonces como el estado más probable en ese momento.

Es importante señalar que además de que la información fluya hacia adelante para determinar el estado más probable en un punto, la información también puede fluir hacia atrás desde el final de la secuencia a ese estado para aumentar o reducir la probabilidad de cada estado en ese punto. Esto es en parte una consecuencia natural de la reversibilidad de la regla de Bayes: nuestras probabilidades cambian de probabilidades previas a probabilidades posteriores al observar más datos. Para dilucidar esto, imagina de nuevo el ejemplo del casino. Como se dijo anteriormente, sin observar ningún rollo, lo más probable es que el estado 0 sea injusto: esta es nuestra probabilidad previa. Si el primer rollo es un 6, nuestra creencia de que state1 es injusto se refuerza (si rodar seis es más probable en un dado injusto). Si se vuelve a rodar un 6, la información fluye hacia atrás desde el segundo rollo de troqueles y refuerza aún más nuestro estado1 la creencia de un dado injusto. Cuantos más rollos tengamos, más información fluye hacia atrás y refuerza o contrasta nuestras creencias sobre el estado, ilustrando así la forma en que la información fluye hacia atrás y hacia adelante para afectar nuestra creencia sobre los estados en la Decodificación Posterior.

Usando algunas manipulaciones elementales, podemos reorganizar esta probabilidad en la siguiente forma usando la regla de Bayes:

```
\begin{equation}
\pi_t^* = \operatorname{argmax}_k P(\text{izquierda}(\pi_{t-1} = k, x_1, \dots, x_{t-1}, x_t) * P(\text{izquierda}(x_1, \dots, x_{t-1}, x_t) / \operatorname{argmax}_k P(\text{izquierda}(\pi_{t-1} = k, x_1, \dots, x_{t-1}, x_t) * P(\text{izquierda}(x_1, \dots, x_{t-1}, x_t))
\end{equation}
```

Debido a que  $P(x)$  es una constante, podemos descuidarla a la hora de maximizar la función. Por lo tanto,

```
\begin{equation}
\pi_t^* = \operatorname{argmax}_k P(\text{izquierda}(\pi_{t-1} = k, x_1, \dots, x_{t-1}, x_t) * P(\text{izquierda}(x_1, \dots, x_{t-1}, x_t) / \operatorname{argmax}_k P(\text{izquierda}(\pi_{t-1} = k, x_1, \dots, x_{t-1}, x_t) * P(\text{izquierda}(x_1, \dots, x_{t-1}, x_t))
\end{equation}
```

Usando la propiedad Markov, podemos simplemente escribir esta expresión de la siguiente manera:

```
\begin{equation}
\pi_t^* = \operatorname{argmax}_k P(\text{izquierda}(\pi_{t-1} = k, x_1, \dots, x_{t-1}, x_t) * P(\text{izquierda}(x_1, \dots, x_{t-1}, x_t))
\end{equation}
```

```
\{t+1},\ldots, x_{n}\ mediados\ pi_{t} =k\ derecha) =\ nombreoperador\{argmax\}_{k} f_{k}(t) * b_{k}(t)
\ final\{ecuación\}
```

Aquí, hemos definido\begin{ecuación}

```
f_{k}(t) =P\ izquierda(\pi_{t} =k, x_{1},\ldots, x_{t}\ derecha)\text{y} b_{k}(t) =P\left(x_{t+1},\ldots, x_{n}\mid\pi_{t} =k\right)\ derecha)
```

\end{ecuación}). Como veremos en breve, estos parámetros se calculan utilizando el algoritmo forward y el algoritmo back respectivamente. Para resolver el problema de decodificación posterior, solo necesitamos resolver cada uno de estos subproblemas. El algoritmo forward se ha ilustrado en el capítulo anterior y en la revisión al inicio de este capítulo y el algoritmo hacia atrás se explicará en la siguiente sección.

## Algoritmo hacia

Como se describió anteriormente, el algoritmo hacia atrás se utiliza para calcular la siguiente probabilidad:

```
\[\comenzar\{ecuación\}
b_{k}(t) =P\ izquierda(x_{t+1},\lpuntos, x_{n}\ mediados\pi_{t} =k\right)\ derecha)
\ final\{ecuación\}]
```

Podemos comenzar a desarrollar una recursión n expandiéndonos a la siguiente forma:

```
\[\comenzar\{ecuación\}
b_{k}(t) =\sum_{l} P\ izquierda(x_{t+1},\lpuntos, x_{n},\pi_{t+1} =l\right)\ mediados\pi_{t} =k\right)\ derecha)
\ final\{ecuación\}]
```

De la propiedad de Markov, luego obtenemos:

```
\[\begin{ecuación}
b_{k}(t) =\sum_{l} P\ izquierda(x_{t+2},\ldots, x_{n}\mid\pi_{t+1} =l\right)\ derecha) * P\ izquierda(\pi_{t+1} =l\mid\pi_{t} =k\right)\ derecha) * P\ izquierda(x_{t+1}\mid\pi_{t+1} =k\right)\ derecha)
\ final\{ecuación\}\]
```

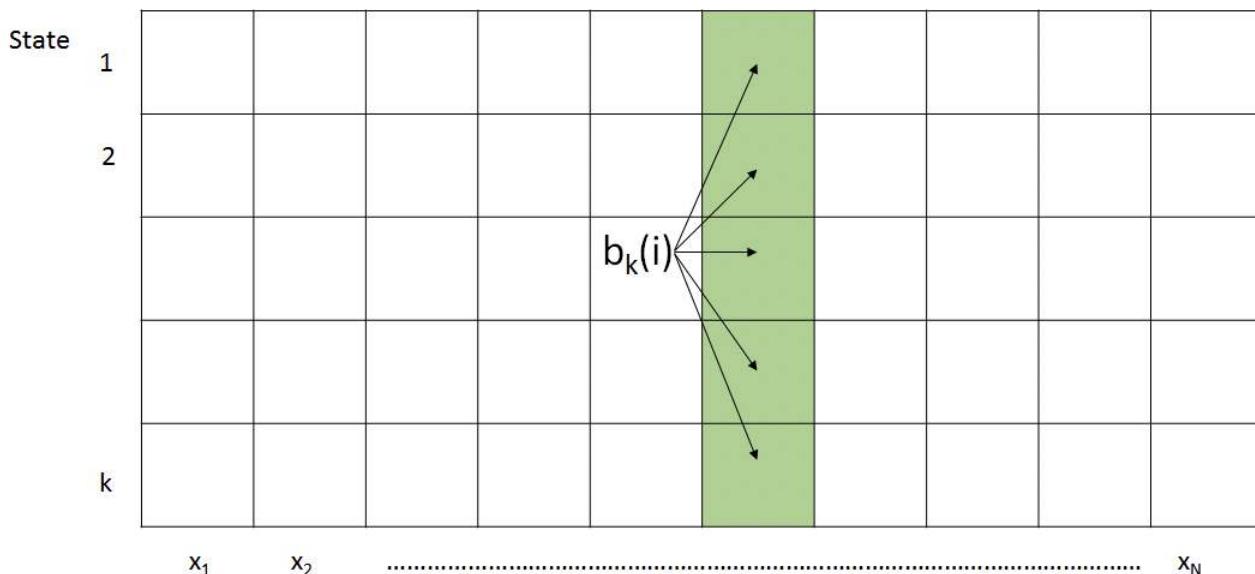
El primer término simplemente corresponde a  $b_l(t+1)$ . Expresar en términos de probabilidades de emisión y transición da la recursión final:

```
\[\begin{ecuación}
b_{k}(t) =\sum_{l} b_{l}(t+1) * a_{kl} * e_{l}\izquierda(x_{t+1}\right)\derecha)
\ final\{ecuación\}\]
```

La comparación de las recursiones hacia adelante y hacia atrás conduce a una visión interesante. Mientras que el algoritmo forward usa los resultados en  $t - 1$  para calcular el resultado para  $t$ , el algoritmo hacia atrás utiliza los resultados de  $t + 1$ , lo que lleva naturalmente a sus respectivos nombres. Otra diferencia significativa radica en las probabilidades de emisión; mientras que las emisiones para el algoritmo directo ocurren desde el estado actual y por lo tanto pueden excluirse de la suma, las emisiones para el algoritmo de retroceso ocurren en el tiempo  $t + 1$  y por lo tanto deben incluirse dentro de la suma.

Dadas sus similitudes, no es sorprendente que el algoritmo hacia atrás también se implemente utilizando una tabla de programación dinámica KxN. El algoritmo, como se representa en la Figura 8.3, comienza inicializando la columna más a la derecha de la tabla a la unidad. Procediendo de derecha a izquierda, cada columna se calcula tomando una suma ponderada de los valores de la columna a la derecha de acuerdo con la recursión descrita anteriormente. Después de calcular la columna situada más a la izquierda, se han calculado todas las probabilidades hacia atrás y el algoritmo termina. Debido a que hay entradas KN y cada entrada examina un total de K otras entradas, esto lleva a  $O(K^2 N)$  complejidad de tiempo y  $O(KN)$  espacio, límites idénticos a los del algoritmo de avance.

Así como  $P(X)$  se calculó sumando la columna más a la derecha de la tabla DP del algoritmo directo,  $P(X)$  también se puede calcular a partir de la suma de la columna más a la izquierda de la tabla DP del algoritmo hacia atrás. Por lo tanto, estos métodos son prácticamente intercambiables para este cálculo en particular.



- Input:  $x = x_1, \dots, x_N$
- Initialization:  $b_k(N) = a_{k0}$ , for all  $k$
- Iteration:  $b_k(i) = \sum_l e_l(x_{i+1}) a_{kl} b_l(i+1)$
- Termination:  $P(x) = \sum_l a_{0l} e_l(x_1) b_l(1)$

Figura 8.3: El algoritmo hacia atrás

#### ¿Sabías?

Tenga en cuenta que incluso cuando se ejecuta el algoritmo hacia atrás, se usan probabilidades de transición hacia adelante, es decir, si moverse en la dirección hacia atrás implica una transición desde el estado  $B \rightarrow A$ , se usa la probabilidad de transición desde el estado  $A \rightarrow B$ . Esto se debe a que moverse hacia atrás del estado  $B$  al estado  $A$  implica que el estado  $B$  sigue al estado  $A$  en nuestro orden normal, hacia adelante, llamando así a la misma probabilidad de transición.

#### El panorama general

¿Por qué tenemos que hacer cálculos tanto hacia adelante como hacia atrás para la decodificación posterior, mientras que los algoritmos que hemos discutido anteriormente requieren solo una dirección? La diferencia radica en el hecho de que la decodificación posterior busca producir probabilidades para los estados subyacentes de posiciones individuales en lugar de secuencias completas de posiciones. Al buscar encontrar el estado subyacente más probable de una posición dada, necesitamos tener en cuenta toda la secuencia en la que existe esa posición, tanto antes como después de ella, como corresponde a un enfoque bayesiano -y hacerlo en un algoritmo de programación dinámica, en el que calculamos recursivamente y terminamos con un maximizando la función, debemos acercarnos a nuestra posición de interés desde ambos lados.

Dado que podemos calcular tanto  $f_k(t)$  como  $b_k(t)$  en  $\theta(K^2 N)$  tiempo y  $\theta(KN)$  espacio para todos  $t = 1, n$ , podemos usar decodificación posterior para determinar el estado más probable  $\pi_t^*$  para  $t = 1, n$ . La expresión relevante viene dada por

```
\begin{equation}
\pi_t^* = \operatorname{argmax}_k P(\pi_t = k | x) = \frac{f_k(i) * b_k(i)}{P(x)}
\end{equation}
```

Con dos métodos (Viterbi y posterior) para decodificar, ¿cuál es más apropiado? Al intentar clasificar cada estado oculto, el método de decodificación Posterior es más informativo porque toma en cuenta todas las rutas posibles a la hora de determinar el estado más probable. En contraste, el método Viterbi sólo toma en cuenta una ruta, que puede terminar representando una mínima fracción de la probabilidad total. Al mismo tiempo, sin embargo, ¡la decodificación posterior puede dar una secuencia inválida de estados! Al seleccionar el estado de probabilidad máxima de cada posición de forma independiente, no estamos considerando cuán probables son las transiciones entre estos estados. Por ejemplo, los estados identificados en los puntos de tiempo  $t$  y  $t + 1$  podrían tener una probabilidad de transición cero entre ellos. Como resultado, seleccionar un método de decodificación depende en gran medida de la aplicación de interés.

### Preguntas frecuentes

P: ¿Qué implica cuando el algoritmo de Viterbi y la decodificación Posterior no están de acuerdo en el camino?

R: En cierto sentido, es simplemente un recordatorio de que nuestro modelo nos da para qué está seleccionando. Cuando buscamos el estado de probabilidad máxima de cada posición independiente y despreciamos las transiciones entre estos estados de probabilidad máxima, podemos obtener algo diferente a cuando buscamos encontrar la ruta total más probable. La biología es complicada; es importante pensar qué métrica es más relevante para la situación biológica en cuestión. En el contexto genómico, un desacuerdo podría ser el resultado de alguna biología 'funky'; empalme alternativo, por ejemplo. En algunos casos, el algoritmo de Viterbi estará cerca de la decodificación Posterior mientras que en algunos otros pueden estar en desacuerdo.

---

This page titled [8.2: Decodificación posterior](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [8.2: Posterior Decoding](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.3: Memoria de codificación en un HMM- Detección de islas CpG

Las islas CpG se definen como regiones dentro de un genoma que están enriquecidas con pares de nucleótidos C y G en la misma cadena. Por lo general, cuando este dinucleótido está presente dentro de un genoma, éste se metila, y cuando se produce la desaminación de la citosina, como lo hace con alguna frecuencia base, se convierte en una timina, otro nucleótido natural, y así no puede ser reconocido tan fácilmente por la célula como una mutación, provocando una mutación de C a T. Esta mayor frecuencia de mutación en las islas CpG agota las islas CpG a lo largo del tiempo evolutivo y las vuelve relativamente raras. Debido a que la metilación puede ocurrir en cualquiera de las cadenas, los CpG generalmente mutan en un TpG o un CpA. Sin embargo, cuando se sitúa dentro de un promotor activo, se suprime la metilación y los dinucleótidos CpG pueden persistir. De manera similar, los CPG en regiones importantes para la función celular se conservan debido a la presión evolutiva. Como resultado, la detección de islas CpG puede resaltar regiones promotoras, otras regiones transcripcionalmente activas o sitios de selección purificadora dentro de un genoma.

### ¿Sabías?

CpG significa [C] ytosina - [p] columna vertebral de hofosfato - [G] uanina. La 'p' implica que nos estamos refiriendo a la misma hebra de la doble hélice, en lugar de un par de bases G-C que ocurre a través de la hélice.

Dada su importancia biológica, las islas CpG son las principales candidatas para modelar. Inicialmente, se puede intentar identificar estas islas explorando el genoma para intervalos fijos ricos en GC. La eficacia de este enfoque se ve socavada por la selección de un tamaño de ventana apropiado; mientras que una ventana demasiado pequeña puede no capturar toda una isla CpG en particular, una ventana demasiado grande daría como resultado que faltaran muchas islas CpG más pequeñas pero genuinas. Examinar el genoma sobre una base por codón también conduce a dificultades porque los pares CpG no necesariamente codifican aminoácidos y, por lo tanto, pueden no estar dentro de un solo codón. En cambio, los HMM son mucho más adecuados para modelar este escenario porque, como veremos en breve en la sección sobre aprendizaje no supervisado, los HMM pueden adaptar sus parámetros subyacentes para maximizar su probabilidad.

No todos los HMM, sin embargo, son adecuados para esta tarea en particular. Un modelo HMM que solo considera las frecuencias de nucleótidos individuales de C y G no logrará capturar la naturaleza de las islas CpG. Considere uno de esos HMM con los dos siguientes estados ocultos:

- Estado '+' que representa islas CpG
- Estado '-': representando no islas

Cada uno de estos dos estados emite entonces bases A, C, G y T con cierta probabilidad. Aunque las islas CpG en este modelo pueden enriquecerse con C y G al aumentar sus respectivas probabilidades de emisión, este modelo no logrará capturar el hecho de que las C y G ocurren predominantemente en pares.

Debido a la propiedad de Markov que gobierna los HMM, la única información disponible en cada paso de tiempo debe estar contenida dentro del estado actual. Por lo tanto, para codificar la memoria dentro de una cadena de Markov, necesitamos aumentar el espacio estatal. Para ello, los estados individuales '+' y '-' pueden ser reemplazados por 4 estados '+' y 4 estados '-': A+, C+, G+, T+, A-, C-, G-, T- (Figura 8.4). Específicamente, hay 2 formas de modelar esto, y esta elección dará como resultado diferentes probabilidades de emisión:

- Un modelo sugiere que el estado A+, por ejemplo, implica que actualmente estamos en una isla CpG y el carácter anterior era un A. Las probabilidades de emisión aquí llevarán la mayor parte de la información y las transiciones serán bastante degeneradas.
- Otro modelo sugiere que el estado A+, por ejemplo, implica que actualmente estamos en una isla CpG y el carácter actual es un A. La probabilidad de emisión aquí será de 1 para A y 0 para todas las demás letras y las probabilidades de transición llevarán la mayor parte de la información en el modelo y las emisiones serán bastante degenerados. Vamos a asumir este modelo a partir de ahora.

### ¿Sabías?

El número de transiciones es el cuadrado del número de estados. Esto da una idea aproximada de cómo aumentar la escala de la “memoria” HMM (y por lo tanto los estados).

- La memoria de este sistema deriva del hecho de que cada estado sólo puede emitir un carácter y por lo tanto “recuerda” su carácter emitido. Además, la naturaleza dinucleotídica de las islas CpG se incorpora dentro de las matrices de transición. En particular, la frecuencia de transición de los estados C+ a G+ es significativamente mayor que de los estados C- a G-, lo que demuestra que estos pares ocurren con mayor frecuencia dentro de las islas.

### FAQ

P: Dado que cada estado emite solo un personaje, ¿podemos decir entonces que esto se reduce a una Cadena Markov en lugar de una HMM?

A: No. A pesar de que las emisiones indican la letra del estado oculto, no indican si el estado es una isla CpG o no: tanto un estado A- como uno A+ emiten solo el A observable.

### FAQ

P: ¿Cómo incorporamos nuestros conocimientos sobre el sistema mientras entrenamos modelos HMM, por ejemplo, algunas probabilidades de emisión de 0 en el caso de detección de isla CpG?

R: Podríamos forzar nuestro conocimiento sobre el modelo estableciendo algunos parámetros y dejando que otros varíen o podríamos dejar que el HMM se suelte en el modelo y dejar que descubra esas relaciones. De hecho, incluso hay métodos que simplifican el modelo al forzar que un subconjunto de parámetros sea 0 pero permitiendo que el HMM elija qué subconjunto.

Dado el marco anterior, podemos usar la decodificación posterior para analizar cada base dentro de un genoma y determinar si es muy probable que sea un constituyente de una isla CpG o no. Pero después de haber construido el modelo HMM expandido, ¿cómo podemos verificar que de hecho es mejor que el modelo de un solo nucleótido? Anteriormente demostramos que el algoritmo hacia adelante o hacia atrás se puede utilizar para calcular P(x) para un determinado

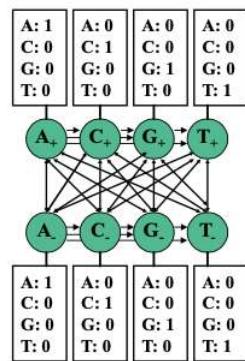


Figura 8.4: HMM para islas CpG

modelo. Si la probabilidad de nuestro conjunto de datos es mayor dado el segundo modelo que el primer modelo, lo más probable es que capture el comportamiento subyacente de manera más efectiva.

Sin embargo, existe un riesgo en complicar el modelo, que es el sobreajuste. Aumentar el número de parámetros para un HMM hace que el HMM sea más probable que sobreajuste los datos y sea menos preciso en la captura del comportamiento subyacente. Una solución común a esto en el aprendizaje automático es usar la regularización, que es esencialmente usar menos parámetros. En este caso, es posible reducir el número de parámetros a aprender al restringir que todas las probabilidades de transición +/- sean el mismo valor y todas las probabilidades de transición -/+ sean el mismo valor, ya que las transiciones de ida y vuelta de los estados + y - son lo que nos interesa modelar, y lo real bases donde ocurrió la transición no son tan importantes para nuestro modelo. Por lo

tanto, para este modelo restringido tenemos que aprender menos parámetros lo que conduce a un modelo más simple y puede ayudar a evitar el sobreajuste.

#### FAQ

P: ¿Hay otras formas de codificar la memoria para la detección de islas CpG? R: Otras ideas con las que se puede experimentar incluyen

- Emitir dinucleótidos y encontrar una manera de lidiar con el solapamiento.
- Agregar un estado especial que va de C a G.

---

This page titled [8.3: Memoria de codificación en un HMM- Detección de islas CpG](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [8.3: Encoding Memory in a HMM- Detection of CpG islands](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.4: Aprendizaje

Vimos cómo puntuar y decodificar una secuencia generada por HMM de dos maneras diferentes. Sin embargo, estos métodos suponían que ya conocíamos las probabilidades de emisión y transición. Si bien siempre somos libres de arriesgarnos a adivinar estos, a veces podemos querer usar un enfoque empírico más basado en datos para derivar estos parámetros. Afortunadamente, el marco HMM permite el aprendizaje de estas probabilidades cuando se proporciona un conjunto de datos de entrenamiento y una arquitectura de conjunto para el modelo.

Cuando se etiquetan los datos de entrenamiento, la estimación de las probabilidades es una forma de aprendizaje supervisado. Uno de esos casos ocurriría si se nos diera una secuencia de ADN de un millón de nucleótidos en la que todas las islas CpG hubieran sido anotadas experimentalmente y se les pidiera que la usáramos para estimar los parámetros de nuestro modelo.

En contraste, cuando los datos de entrenamiento no están etiquetados, el problema de estimación es una forma de aprendizaje no supervisado. Continuando con el ejemplo de isla CpG, esta situación ocurriría si la secuencia de ADN proporcionada no contenía ninguna anotación de isla y necesitábamos estimar los parámetros del modelo e identificar

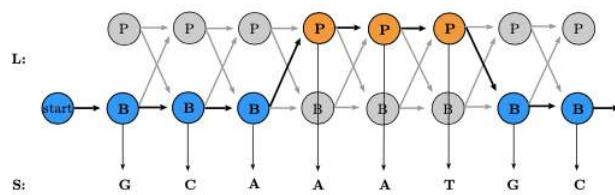


Figura 8.5: Aprendizaje supervisado de islas CpG

las islas.

### Aprendizaje supervisado

Cuando se proporcionan datos etiquetados, la idea de estimar los parámetros del modelo es sencilla. Supongamos que se le da una secuencia etiquetada  $x_1, \dots, x_N$  así como la verdadera secuencia de estado oculto  $\pi_1, \dots, \pi_N$ . Intuitivamente, se podría esperar que las probabilidades que maximizan la probabilidad de los datos sean las probabilidades reales que se observen dentro de los datos. Este es efectivamente el caso y se puede formalizar definiendo  $A_{kl}$  como el número de veces que el estado oculto  $k$  pasa a  $l$  y  $E_k(b)$  para que sea el número de veces que  $b$  se emite desde el estado oculto  $k$ . Los parámetros  $\theta$  que maximizan  $P(x|\theta)$  se obtienen simplemente contando de la siguiente manera:

```
\[
\begin{aligned}
&\text{\begin{alineado}} \\
&a_{\{k l\}} &=& \frac{\{A_{\{k l\}}\}}{\{\sum_{\{i\}} A_{\{k i\}}\}} \\
&e_{\{k\}}(b) &=& \frac{\{E_{\{k\}}(b)\}}{\{\sum_{\{c\}} E_{\{k\}}(c)\}} \\
&\text{\end{alineado}} \\
\end{aligned}
]
```

Un conjunto de entrenamiento de ejemplo se muestra en la Figura 8.5. En este ejemplo, es obvio que la probabilidad de transición de B a P es

$$\begin{aligned} (\text{begin ecuación}) \\ \frac{1}{3+1} = \frac{1}{4} \\ (\text{end ecuación}) \end{aligned}$$

(hay 3 transiciones B a B y transiciones de 1 B a P) y la probabilidad de emitir una G desde el estado B es

$$\begin{aligned} (\text{begin ecuación}) \\ \frac{2}{2+2+1} = \frac{2}{5} \\ (\text{end ecuación}) \end{aligned}$$

(hay 2 G emitidas desde el estado B, 2 C's y 1 A)

Observe, sin embargo, que en el ejemplo anterior la probabilidad de emisión del carácter T del estado B es 0 porque no se encontraron tales emisiones en el conjunto de entrenamiento. Una probabilidad cero, ya sea para la transición o emisión, es particularmente problemática porque conduce a una penalización logarítmica infinita. En realidad, sin embargo, la probabilidad cero puede simplemente haber surgido debido a un sobreajuste o un pequeño tamaño de muestra. Para rectificar este problema y mantener la flexibilidad dentro de nuestro modelo, podemos recopilar más datos sobre los que entrenar, reduciendo la posibilidad

de que la probabilidad cero se deba a un pequeño tamaño de muestra. Otra posibilidad es usar 'pseudorecuentos' en lugar de recuentos absolutos: agregar artificialmente algunos recuentos a nuestros datos de entrenamiento que creemos que representan con mayor precisión los parámetros reales y ayudan a contrarrestar los errores de tamaño de la muestra.

```
\begin{ecuación}
\begin{array}{l}
A_{k l}^* = A_{k l} + r_{k l} \\
E_k(b)^* = E_k(b) + r_k(b)
\end{array}
\end{ecuación}. \nonumber]
```

Los parámetros de pseudoconteo más grandes corresponden a una fuerte creencia previa sobre los parámetros, reflejada en el hecho de que estos pseudorecuentos, derivados de tus antecedentes, son comparativamente abrumadoras las observaciones, tus datos de entrenamiento. Asimismo, los parámetros de pseudoconteo pequeños ( $r \ll 1$ ) se utilizan con mayor frecuencia cuando nuestros antecedentes son relativamente débiles y pretendemos no abrumar los datos empíricos sino solo evitar probabilidades excesivamente duras de 0.

## Aprendizaje no supervisado

El aprendizaje no supervisado implica estimar parámetros basados en datos no etiquetados. Esto puede parecer imposible - ¿cómo podemos tomar datos de los que no sabemos nada y usarlos para "aprender"? - pero un enfoque iterativo puede arrojar resultados sorprendentemente buenos, y es la elección típica en estos casos. Esto puede pensarse vagamente como un algoritmo evolutivo: a partir de alguna elección inicial de parámetros, el algoritmo evalúa qué tan bien los parámetros explican o se relacionan con los datos, utiliza algún paso en esa evaluación para hacer mejoras en los parámetros, y luego evalúa los nuevos parámetros, produciendo incrementos mejoras en los parámetros en cada paso del mismo modo que la aptitud o falta de los mismos de un organismo particular en su entorno produce incrementos incrementales a lo largo del tiempo evolutivo ya que los alelos ventajosos se transmiten preferentemente.

Supongamos que tenemos algún tipo de creencia previa sobre cuál debería ser cada probabilidad de emisión y transición. Dados estos parámetros, podemos usar un método de decodificación para inferir los estados ocultos subyacentes a la secuencia de datos proporcionada. Usando este análisis particular de decodificación, podemos reestimar los recuentos y probabilidades de transición y emisión en un proceso similar al utilizado para el aprendizaje supervisado. Si repetimos este procedimiento hasta que la mejora en la probabilidad de los datos permanezca relativamente estable, la secuencia de datos debería conducir finalmente los parámetros a sus valores apropiados.

### FAQ

P: ¿Por qué funciona incluso el aprendizaje sin supervisión? ¿O es mágico?

R: El aprendizaje no supervisado funciona porque tenemos la secuencia (datos de entrada) y esto guía cada paso de la iteración; para pasar de una secuencia etiquetada a un conjunto de parámetros, los posteriores son guiados por la entrada y su anotación, mientras que para anotar los datos de entrada, los parámetros y la secuencia guían el procedimiento.

Para los HMM en particular, dos métodos principales de aprendizaje no supervisado son útiles.

### Maximización de expectativas mediante el entrenamiento de Viterbi

El primer método, el **entrenamiento Viterbi**, es relativamente sencillo pero no del todo riguroso. Después de elegir algunos parámetros iniciales del modelo de mejor estimación, procede de la siguiente manera:

**Paso E:** Realizar decodificación Viterbi para encontrar  $\pi^*$

**Paso M:** Calcular los nuevos parámetros  $A_{kl}^*, E_k(b)^*$  utilizando el formalismo de conteo simple en el aprendizaje supervisado (paso de Maximización)

**Iteración:** Repita los pasos E y M hasta que la probabilidad  $P(x|\theta)$  converja

Aunque el entrenamiento de Viterbi converge rápidamente, sus estimaciones de parámetros resultantes suelen ser inferiores a las del Algoritmo Baum-Welch. Este resultado se deriva del hecho de que el entrenamiento de Viterbi solo considera el camino oculto más probable en lugar de la colección de todos los caminos ocultos posibles.

### **Maximización de expectativas: El algoritmo Baum-Welch**

El enfoque más riguroso del aprendizaje no supervisado implica la aplicación de la Maximización de Expectativas a los HMM. En general, EM procede de la siguiente manera:

**Init:** Inicializar los parámetros a algún estado de mejor estimación

**Paso E:** Estimar la probabilidad esperada de estados ocultos dados los últimos parámetros y la secuencia observada (paso Expectativa)

**Paso M:** Elija nuevos parámetros de máxima verosimilitud usando la distribución de probabilidad de estados ocultos (paso de Maximización)

**Iteración:** Repita los pasos E y M hasta que converja la probabilidad de los datos dados los parámetros

El poder de EM radica en el hecho de que se garantiza que  $P(x|\theta)$  aumente con cada iteración del algoritmo. Por lo tanto, cuando esta probabilidad converge, se ha alcanzado un máximo local. Como resultado, si utilizamos una variedad de estados de inicialización, lo más probable es que podamos identificar el máximo global, es decir, los mejores parámetros  $\theta$ . El algoritmo Baum-Welch generaliza EM a HMM, en particular, utiliza los algoritmos de avance y retroceso para calcular  $P(x|\theta)$  y estimar  $A_{kl}$  y  $E_k(b)$ . El algoritmo procede de la siguiente manera:

**Inicialización 1.** Inicializar los parámetros a algún estado de mejor estimación

**Iteración 1.** Ejecute el algoritmo de avance

2. Ejecute el algoritmo hacia atrás

3. Calcular la nueva probabilidad logarítmica  $P(x|\theta)$

4. Calcular  $A_{kl}$  y  $E_k(b)$

5. Calcular  $a_{kl}$  y  $e_k(b)$  usando las fórmulas de pseudoconteo 6. Repita hasta que  $P(x|\theta)$  converja

Anteriormente, discutimos cómo calcular  $P(x|\theta)$  usando los resultados finales del algoritmo hacia adelante o hacia atrás. Pero, ¿cómo estimamos  $A_{kl}$  y  $E_k(b)$ ? Consideremos el número esperado de transiciones del estado  $k$  al estado  $l$  dado un conjunto actual de parámetros  $\theta$ . Podemos expresar esta expectativa como

$$A_{kl} = \sum_t P(\pi_t = k, \pi_{t+1} = l | x, \theta) = \sum_t \frac{P(\pi_t = k, \pi_{t+1} = l, x | \theta)}{P(x | \theta)}$$

La explotación de la propiedad de Markov y las definiciones de las probabilidades de emisión y transición conduce a la siguiente derivación:

```
\[ comenzar {alineado}
A_{k l} &= \sum_t \frac{P(\text{izquierda}(x_{-1}) \text{lpuntos } x_{-t}, \text{pi}_{-t} = k, \text{pi}_{-t+1} = l, x_{-t+1}) \text{lpuntos } x_{-N} \text{mediados} \theta \derecha)} {P(x \text{mediados} \theta)}
&= \sum_t \frac{P(\text{izquierda}(x_{-1}) \text{lpuntos } x_{-t}, \text{pi}_{-t} = k \derecha) * P(\text{izquierda}(\text{pi}_{-t+1} = l, x_{-t+1}) \text{lpuntos } x_{-N} \text{mediados} \text{pi}_{-t} \theta \derecha)} {P(x \text{mediados} \theta)}
&= \sum_t \frac{f_{-k}(t) * P(\text{izquierda}(\text{pi}_{-t+1} = l \text{mediados} \text{pi}_{-t} = k \derecha) * P(\text{izquierda}(x_{-t+1}) \text{mediados} \text{pi}_{-t+1} = l \derecha) * P(\text{izquierda}(x_{-t+2}) \text{lpuntos } x_{-N} \text{mediados} \text{pi}_{-t+1} = l, \theta \derecha))} {P(x \text{mediados} \theta)}
&= \sum_t f_{-k}(t) * a_{-k l} * e_{-l} * b_{-l} * P(x_{-t+1} \text{derecha}) * b_{-l(t+1)} * P(x \text{mid} \theta)
\] final {alineado}\]
```

Una derivación similar conduce a la siguiente expresión para  $E_k(b)$ :

$$E_k(b) = \sum_{x_i=b} \frac{f_k(t) * b_k(t)}{P(x | \theta)} \quad (8.4.1)$$

Por lo tanto, al ejecutar los algoritmos de avance y retroceso, tenemos toda la información necesaria para calcular  $P(x|\theta)$  y actualizar las probabilidades de emisión y transición durante cada iteración. Debido a que estas actualizaciones son operaciones de tiempo constante una vez que se han calculado  $P(x|\theta)$ ,  $f_k(t)$  y  $b_k(t)$ , la complejidad de tiempo total para esta versión del aprendizaje no supervisado es  $\theta(K^2 NS)$ , donde  $S$  es el número total de iteraciones.

#### FAQ

P: ¿Cómo codificas tus creencias previas al aprender con Baum-Welch?

R: Esas creencias previas están codificadas en las inicializaciones de los algoritmos hacia adelante y hacia atrás

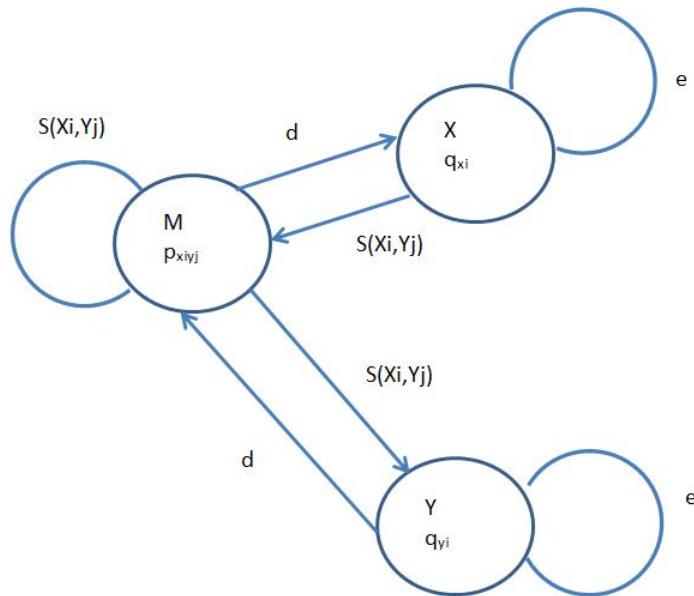


Figura 8.6: Modelo HMM para alineación con penalizaciones por hueco afín

This page titled [8.4: Aprendizaje](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [8.4: Learning](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín

Podemos usar HMM para alinear secuencias con penalizaciones por brecha afín. Recordemos que penalizaciones por brecha afín penaliza más abrir/ iniciar la brecha que extenderla, así la penalización de un hueco de longitud  $g$  es  $r(g) = -d - (g-1)*e$ , donde  $d$  es la penalización para abrir la brecha y  $e$  es la penalización para extender una brecha ya abierta.

Buscaremos alinear dos secuencias con la penalización por hueco afín. Se nos dan dos secuencias:  $X$  e  $Y$ , la matriz de puntuación  $S$  ( $S(x_i, y_j) =$  puntaje de coincidencia  $x_i$  con  $y_j$ ), penalización por apertura de brecha de  $d$  y penalización de extensión de hueco de  $e$ . Podemos mapear este problema en un problema HMM usando los siguientes estados, probabilidades de transición y probabilidades de emisión.

### Estados:

Hay tres estados que involucra:  $M$  (emparejando  $x_i$  con  $y_j$ ),  $X$  (alineando  $x_i$  con un hueco),  $Y$  (alineando  $y_j$  con un hueco). Además, junto a cada transición, hay una actualización de los índices  $i, j$ . Siempre que estemos en estado  $M$ ,  $(i, j) = (i, j) + (1,1)$ . En estado  $X$ ,  $(i, j) = (i, j) + (1,0)$ . En estado  $Y$ ,  $(i, j) = (i, j) + (0,1)$ .

### Probabilidades de transición:

Hay 7 probabilidades de transición a considerar como se muestra en la figura 6.  $P(\text{siguiente Estado} = M \mid \text{corriente} = M) = S(x_i, y_j)$

$P(\text{siguiente Estado} = X \mid \text{corriente} = M) = d$

$P(\text{siguiente Estado} = Y \mid \text{corriente} = M) = d$

$P(\text{siguiente Estado} = X \mid \text{corriente} = X) = e$

$P(\text{siguiente Estado} = M \mid \text{corriente} = X) = S(x_i, y_j)$

$P(\text{siguiente Estado} = Y \mid \text{corriente} = X) = e$

$P(\text{siguiente Estado} = M \mid \text{corriente} = Y) = S(x_i, y_j)$

También podemos guardar las probabilidades de transición en una matriz de transición  $A = [a_{ij}]$ , donde  $a_{ij} = P(\text{siguiente Estado} = j \mid \text{actual} = i) \text{ y } \sum_j a_{ij} = 1$

### Probabilidades de emisión:

Las probabilidades de emisión son:

Del estado  $M$ :  $p_{x_i y_j} = p(x_i \text{ alineado a } y_j)$

Del estado  $X$ :  $q_{x_i} = p(x_i \text{ alineado al hueco})$

Del estado  $Y$ :  $q_{y_i} = p(y_j \text{ alineado al hueco})$

Ejemplo:

$X = 'VLSPADK'$

$Y = 'HLAESK'$

La alineación generada por el modelo es:  $MMXXMYM$

Que corresponde a:

$X = 'VLSPAD K'$

$Y = 'HL\_ AESK'$

### ¿Sabías?

Para fines de clasificación, la decodificación posterior 'path' es más informativa que la ruta Viterbi ya que es una medida más refinada de la cual estados ocultos generaron  $x$ . Sin embargo, puede dar una secuencia no válida de estados, por ejemplo, cuando no todas las transiciones  $j \rightarrow k$  pueden ser posibles, podría tener state  $(i) = j$  y state  $(i+1) = k$

This page titled [8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- 8.5: Using HMMs to align sequences with affine gap penalties by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.6: Direcciones actuales de investigación, ¿qué hemos aprendido? , Bibliografía

- Los HMM han sido ampliamente utilizados en diversos campos de la biología computacional. Una de las primeras aplicaciones de este tipo fue en un algoritmo de búsqueda de genes conocido como GENSCAN escrito por Chris Burge y Samuel Karlin [1]. Debido a que la distribución geométrica de la longitud de los HMM no modela bien las regiones exónicas, Burge et al utilizaron una adaptación de HMM conocidos como modelos semi-Markov ocultos (HSMM). Estos tipos de modelos difieren en que cada vez que se alcanza un estado oculto, la duración de ese estado ( $d_j$ ) se elige de una distribución y el estado emite entonces exactamente  $d_j$  caracteres. La transición de este estado oculto al siguiente es entonces análoga al procedimiento HMM excepto que  $a_{kk} = 0$  para todos  $k$ , evitando así la auto-transición. Muchos de los mismos algoritmos que se desarrollaron previamente para HMM se pueden modificar para HSMM, aunque los detalles no se discutirán aquí, los algoritmos hacia adelante y hacia atrás se pueden modificar para que se ejecuten en el tiempo  $O(K^2 N^3)$ , donde  $N$  es el número de caracteres observados. Esta complejidad temporal supone que no hay límite superior en la duración de un estado, pero imponer tal límite reduce la complejidad a  $O(K^2 N D^2)$ , donde  $D$  es la duración máxima posible de un estado.

El diagrama de estado básico subyacente al modelo de Burge se representa en la Figura 8.7. El diagrama incluido solo enumera los estados en la cadena directa del ADN, pero en realidad también se incluye una imagen especular de estos estados para la cadena inversa, lo que resulta en un total de 27 estados ocultos. Como ilustra el diagrama, el modelo incorpora muchas de las principales unidades funcionales de genes, incluyendo exones, intrones, promotores, UTR y colas poli-A. Además, se utilizan tres estados intrónicos y exónicos diferentes para asegurar que la longitud total de todos los exones en un gen sea un múltiplo de tres. Similar al ejemplo de la isla CpG, este espacio de estado expandido habilitó la codificación de la memoria dentro del modelo.

- Recientemente se ha hecho un esfuerzo para hacer un enfoque basado en HMM para las búsquedas de homología, llamado HMMER, una alternativa viable a BLAST en términos de eficiencia computacional. A diferencia de la mayoría de los otros algoritmos de búsqueda de homología, HMMER, escrito por Sean Eddy, utiliza la certeza de sobrealineación promedio del algoritmo Forward, en lugar de solo informar la alineación de máxima verosimilitud (a la Viterbi); este enfoque suele ser mejor para detectar homologías más remotas, como tiempos de divergencia aumentar, pueden llegar a ser formas más viables de alinear secuencias, cada una de ellas individualmente no lo suficientemente fuerte como para diferenciarse del ruido sino que juntas dan evidencia de homología. Un desarrollo reciente particularmente emocionante es que HMMER ya está disponible como servidor web; se puede encontrar en <http://www.ebi.ac.uk/Tools/hmmer/>.
- Un tema interesante que puede explorarse también se refiere a la concordancia de las rutas de decodificación Viterbi y Posterior; no solo para la detección de islas CpG sino incluso para la detección del estado de cromatina. Uno puede mirar múltiples caminos por muestreo, haciendo preguntas como:
  - ¿Cuál es el camino máximo a posteriori vs viterbi? ¿Dónde se diferencian?
  - ¿Se pueden encontrar rutas completas pero máximamente disjuntas (de Viterbi)?

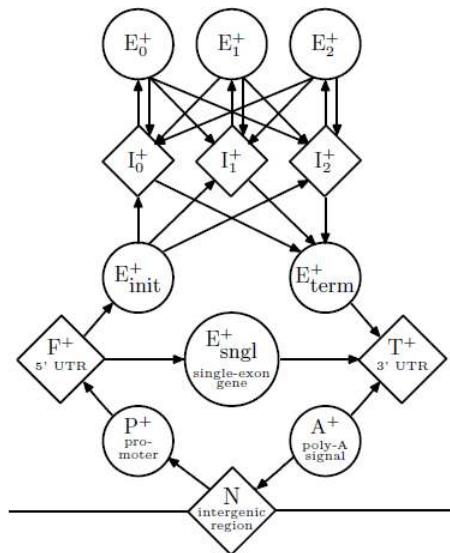


Figura 8.7: Diagrama de espacio de estado utilizado en GENSCAN

This page titled [8.6: Direcciones actuales de investigación, ¿qué hemos aprendido?](#), Bibliografía is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [8.6: Current Research Directions, What Have We Learned?, Bibliography](#) by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0.  
Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 8.9 ¿Qué hemos aprendido?

Usando el marco computacional básico proporcionado por Hidden Markov Models, hemos aprendido a inferir el conjunto más probable de estados ocultos subyacentes a una secuencia de caracteres observados. En particular, una combinación de los algoritmos hacia adelante y hacia atrás permitió una forma de esta inferencia, es decir, la decodificación posterior, en el tiempo O ( $KN^2$ ). También aprendimos cómo se puede usar el aprendizaje no supervisado o supervisado para identificar los mejores parámetros para un HMM cuando se proporciona un conjunto de datos sin etiquetar o etiquetado. La combinación de estos métodos de decodificación y estimación de parámetros permite la aplicación de HMM a una amplia variedad de problemas en biología computacional, de los cuales la isla CpG y la identificación génica forman un pequeño subconjunto. Dada la flexibilidad y el poder analítico que proporcionan los HMM, estos métodos jugarán un papel importante en la biología computacional en el futuro previsible.

---

8.9 ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [8.9 What Have We Learned?](#) has no license indicated.

## Bibliografía

- [1] Christopher B Burge y Samuel Karlin. Encontrar los genes en el ADN genómico. Dictamen Actual en Biología Estructural, 8 (3) :346 — 354, 1998.

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 9: Identificación Génica- Estructura Génica, Semi-Markov, CRFS

- 9.1: Introducción
- 9.2: Descripción general de los contenidos del capítulo
- 9.3: Genes eucariotas: una introducción
- 9.4: Supuestos para la identificación computacional de genes
- 9.5: Cadenas Ocultas de Markov
- 9.6: Campos aleatorios condicionales
- 9.7: Otros métodos
- 9.8: Conclusión, Bibliografía

#### Bibliografía

---

This page titled [9: Identificación Génica- Estructura Génica, Semi-Markov, CRFS](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 9.1: Introducción

Después de que un genoma ha sido secuenciado, un siguiente paso común es intentar inferir el potencial funcional del organismo o célula codificada a través de un análisis cuidadoso de esa secuencia. Esto toma principalmente la forma de identificar los genes codificantes de proteínas dentro de la secuencia ya que se cree que son las principales unidades de función dentro de los sistemas vivos; esto no quiere decir que sean las únicas unidades funcionales dentro de los genomas como cosas como los motivos reguladores y los ARN no codificantes también lo son elementos imperativos.

Esta anotación de las regiones codificadoras de proteínas es demasiado laboriosa para realizarla a mano, por lo que se automatiza en un proceso conocido como identificación génica computacional. Los algoritmos subyacentes a este proceso suelen basarse en Modelos Ocultos de Markov (HMM), un concepto discutido en capítulos anteriores para resolver problemas simples como saber si un casino está rodando un dado justo versus un dado cargado. Los genomas, sin embargo, son conjuntos de datos muy complicados, repletos de repeticiones largas, genes superpuestos (donde uno o más nucleótidos forman parte de dos o más genes distintos) y pseudogenes (regiones no transcritas que se ven muy similares a los genes) entre muchas otras ofuscaciones. Por lo tanto, los datos experimentales y evolutivos a menudo necesitan ser incluidos en los HMM para una mayor precisión anotacional, lo que puede resultar en una pérdida de escalabilidad o una dependencia de suposiciones incorrectas de independencia. Se han utilizado algoritmos alternativos para abordar los problemas de los HMM incluyendo aquellos basados en Campos Aleatorios Condicionales (CRF), que se basan en la creación de una distribución de los estados ocultos de la secuencia genómica en cuestión condicionada a datos conocidos. El uso de CRF no ha ido eliminando los HMM ya que ambos se utilizan con diversos grados de éxito en la práctica.<sup>1</sup>

---

<sup>1</sup> R. Guigo (1997). “Identificación computacional de genes: un problema abierto”. Computadoras Chem. Vol. 21. 165

This page titled [9.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **9.1: Introduction** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.2: Descripción general de los contenidos del capítulo

Este capítulo comenzará con una discusión sobre las complejidades del gen eucariota. Luego se describirá cómo se pueden usar los HMM como modelo para analizar genomas eucariotas en genes codificantes de proteínas y regiones que no lo son; esto incluirá una referencia a las fortalezas y debilidades de un enfoque HMM. Finalmente, se describirá como alternativa el uso de CRF para anotar regiones codificantes de proteínas.

This page titled [9.2: Descripción general de los contenidos del capítulo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [9.2: Overview of Chapter Contents](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.3: Genes eucariotas: una introducción

Dentro de los genomas eucariotas, solo una pequeña fracción del contenido de nucleótidos consiste realmente en genes codificadores de proteínas (en humanos, las regiones codificadoras de proteínas constituyen aproximadamente 1%-1.5% del genoma completo). El resto del ADN se clasifica como regiones intergénicas (Ver Figura 9.1) y contiene cosas como motivos reguladores, transposones, integrones y genes codificantes no proteicos.<sup>2</sup>

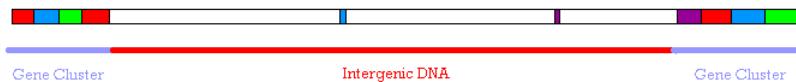
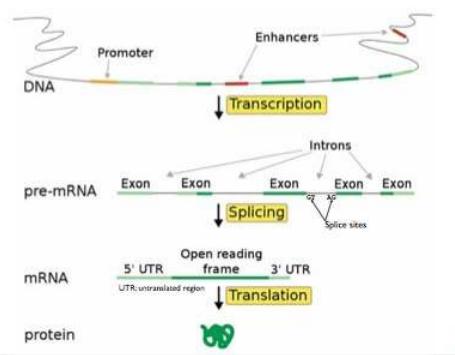


Figura 9.1: ADN intergénico

Además, de la pequeña fracción del ADN que se transcribe en ARNm, no todo se traduce en proteína. Ciertas regiones conocidas como intrones, se eliminan o “empalman” del ARNm precursor. Este ARNm ahora procesado, que contiene sólo “exones” y algunas otras modificaciones adicionales discutidas en capítulos anteriores, se traduce en proteína. (Ver Figura 9.2) El objetivo de la identificación computacional de genes no solo es seleccionar las pocas regiones del genoma eucariota completo que codifican proteínas sino también analizar esas regiones codificadoras de proteínas en identidades de exón o intrón para que se pueda conocer la secuencia de la proteína sintetizada.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 9.2: Empalme de intrón/exón

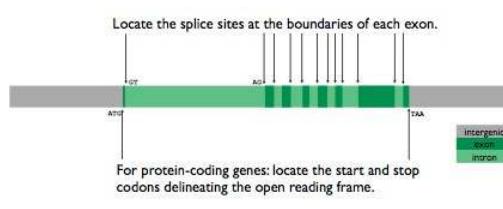
<sup>2</sup> “Región intergénica”. <http://en.Wikipedia.org/wiki/Intergenic> región

This page titled **9.3: Genes eucariotas: una introducción** is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **9.3: Eukaryotic Genes- An Introduction** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.4: Supuestos para la identificación computacional de genes

Los supuestos generales para la identificación computacional de genes son que los exones son delineados por una secuencia AG al inicio del exón y una secuencia de GT al final del exón. Para los genes que codifican proteínas, el codón de inicio (ATG) y los codones finales (TAA, TGA, TAG) delinean el marco de lectura abierto. (La mayoría de estas ideas se pueden ver en la Figura 9.3) Estos supuestos se incorporarán en HMM más complejos que se describen a continuación.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

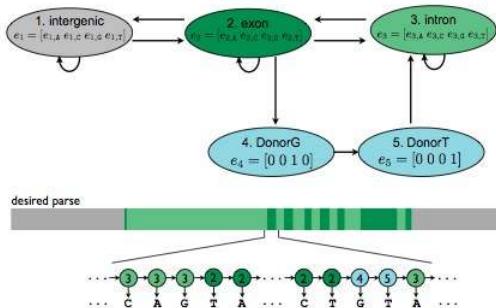
Figura 9.3: Delineación de Exones y Marcos de Lectura Abiertos

This page titled [9.4: Supuestos para la identificación computacional de genes](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [9.4: Assumptions for Computational Gene Identification](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.5: Cadenas Ocultas de Markov

Un juguete Hidden Markov Model es un enfoque generativo para modelar este comportamiento. Cada emisión del HMM es una base/letra de ADN. Los estados ocultos del modelo son intergénicos, exón, intrón. Mejorar este modelo implicaría incluir los estados ocultos DonOrg y Donort. Los estados DonOrg y Donort utilizan la información de que los exones son delineados por GT al final de la secuencia antes del inicio de un intrón. (Ver Figura 9.4 para la inclusión de DonOrg y DonORT en el modelo)

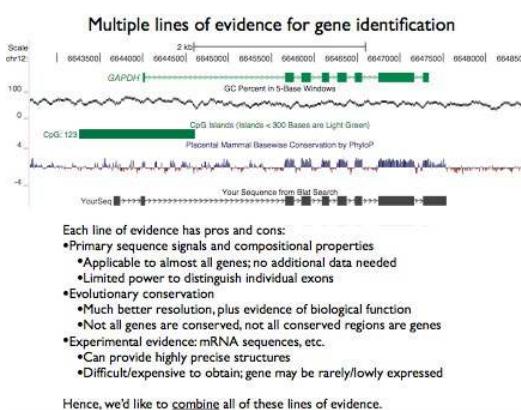


© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 9.4: Modelo oculto de Markov que utiliza la suposición de donante GT

La  $e$  en cada estado representa probabilidades de emisión y las flechas indican las probabilidades de transición.

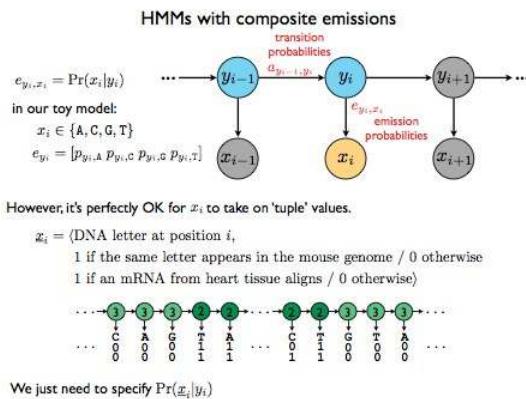
A parte de los supuestos iniciales, evidencia adicional como la conservación evolutiva y los datos de ARNm experimental pueden ayudar a crear un HMM para modelar mejor el comportamiento. (Ver Figura 9.5)



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 9.5: Múltiples líneas de evidencia para la identificación de genes

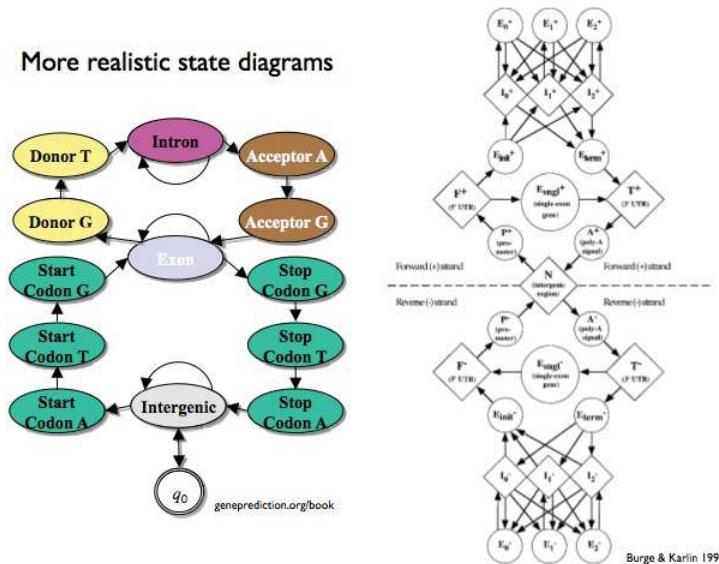
Combinando todas las líneas de evidencia discutidas anteriormente, podemos crear un HMM con emisiones compuestas en que cada valor emitido es una “tupla” de valores recolectados. (Ver Figura 9.6)



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 9.6: HMM con emisiones compuestas

Algunos supuestos de este modelo compuesto son que cada nueva “característica” de emisión es independiente del resto. Sin embargo, esto crea el problema de que con cada nueva característica, la tupla aumenta de longitud, y el número de estados del HMM aumenta exponencialmente, lo que lleva a una explosión combinatoria, lo que significa un pobre escalado. (En la Figura 9.7 se pueden encontrar ejemplos de HMM más complejos que pueden dar como resultado una mala escala)



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

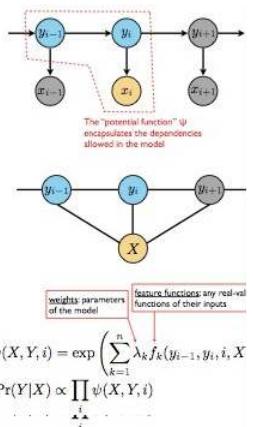
Figura 9.7: Diagrama de estado que considera la dirección de la traducción del ARN

This page titled [9.5: Cadenas Ocultas de Markov](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **9.5: Hidden Markov Chains** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.6: Campos aleatorios condicionales

Los campos aleatorios condicionales, CRF, son una alternativa a los HMM. Al ser un enfoque discriminativo, este tipo de modelos no toma en cuenta la distribución conjunta de todo, al igual que un HMM de mal escalado. Los estados ocultos en una CRF están condicionados a la secuencia de entrada. (Ver Figura 9.8)<sup>3</sup>



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 9.8: Campos aleatorios condicionales: un enfoque discriminativo condicionado a la secuencia de entrada

Una función característica es como una puntuación, devolviendo un número de valor real en función de sus entradas que refleja la evidencia de una etiqueta en una posición particular. (Ver Figura 9.9) La probabilidad condicional de la secuencia emitida es su puntuación dividida por la puntuación total del estado oculto. (Ver Figura 9.10)

$$f_1(y_{i-1}, y_i, i, X) = \begin{cases} 1 & \text{if } y_i = \text{exon} \text{ and position } i \text{ is conserved in mouse} \\ 0 & \text{otherwise} \end{cases}$$

$$f_2(y_{i-1}, y_i, i, X) = \begin{cases} 1 & \text{if } y_i = \text{exon} \text{ and position } i \text{ is conserved in rat} \\ 0 & \text{otherwise} \end{cases}$$

$$f_3(y_{i-1}, y_i, i, X) = \# \text{ of mRNA sequences aligning to position } i \text{ (if } y_i = \text{exon}; 0 \text{ otherwise)}$$

Figura 9.9: Ejemplos de funciones de características

$$\Pr(Y|X) = \frac{1}{Z(X)} \prod_i \psi(X, Y, i) \quad \text{where} \quad Z(X) = \sum_{Y'} \prod_i \psi(X, Y', i)$$

Figura 9.10: Puntuación de probabilidad condicional de una secuencia emitida

Cada función característica está ponderada, de modo que durante el entrenamiento, los pesos se pueden ajustar en consecuencia.

Las funciones características pueden incorporar grandes cantidades de evidencia sin la asunción de independencia de Naive Bayes, haciéndolas escalables y precisas. Sin embargo, el entrenamiento es mucho más difícil con los CRF que con los HMM.

<sup>3</sup> Campo Aleatorio Condicional. Wikipedia. <http://en.Wikipedia.org/wiki/Conditional campo aleatorio>

This page titled 9.6: Campos aleatorios condicionales is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Manolis Kellis et al. (MIT OpenCourseWare) via source content that was edited to the style and standards of the LibreTexts platform.

- 9.6: Conditional Random Fields by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.7: Otros métodos

Además de los HMM y CRF, existen otros métodos para la identificación computacional de genes. Los modelos semi-markov generan emisiones de longitud de secuencia variable, lo que significa que las transiciones no son del todo menos memoria en los estados ocultos.

Los modelos Max-min son adaptaciones de máquinas vectoriales de soporte. Estos métodos aún no se han aplicado a los genomas de mamíferos.<sup>4</sup>

<sup>4</sup> Para una mejor comprensión de SVM: <http://dspace.mit.edu/bitstream/hand...663/6-034Fall> - 2002/OcwWeb/Electrical-Engineering-and-Computer-Science/6-034Artificial-IntelligenceFall2002/Tools/detail/svmachine.htm

This page titled [9.7: Otros métodos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [9.7: Other Methods](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 9.8: Conclusión, Bibliografía

La identificación computacional de genes, ya que implica encontrar los elementos funcionales codificados dentro de un genoma, tiene mucho significado práctico y teórico para el avance de los campos bio- lógicos.

Los dos enfoques descritos anteriormente se resumen a continuación en la Figura 9.11:

A comparison	
HMM	CRF
$\psi(X, Y, i) = a_{y_{i-1}, y_i} \cdot e_{y_i, x_i}$	$\psi(X, Y, i) = \exp \left( \sum_{k=1}^n \lambda_k f_k(y_{i-1}, y_i, i, X) \right)$
$\Pr(X, Y) = \prod_i \psi(X, Y, i)$	
$\Pr(Y X) = \frac{\Pr(X, Y)}{\Pr(X)}$ <small>(Bayes' law)</small>	$\Pr(Y X) = \frac{1}{Z(X)} \prod_i \psi(X, Y, i)$
$\Pr(Y X) = \frac{1}{\Pr(X)} \prod_i \psi(X, Y, i)$	
$\Pr(X) = \sum_Y \prod_i \psi(X, Y, i)$ <small>↑</small>	$Z(X) = \sum_Y \prod_i \psi(X, Y, i)$
<p><b>Q: How do we compute this efficiently?</b>  <b>A: Forward algorithm. CRFs have a direct analog (Viterbi too)</b></p>	
$\lambda_1 = 1, \quad f_1(y_{i-1}, y_i, i, x) = \log(a_{y_{i-1}, y_i} \cdot e_{y_i, x}) \quad \Rightarrow \{HMM\} \subset \{CRF\}$	

Figura 9.11: Comparación de HMM y CRFs

### HMM

- modelo generativo
- genera aleatoriamente datos observables, generalmente con un estado oculto
- especifica una distribución de probabilidad conjunta
- $P(x, y) = P(x|y)P(y)$
- a veces difíciles de modelar dependencias correctamente
- los estados ocultos son las etiquetas para cada base de ADN/letra
- emisiones compuestas son una combinación de la base/letra de ADN que se emite con evidencia adicional

### CRF

- modelo discriminativo
- modelos dependencia de la variable no observada y de una variable observada x •  $P(y|x)$
- difícil de entrenar sin supervisión
- más efectivo para cuando el modelo no requiere distribución conjunta

En la práctica, la especificación génica resultante usando CONTRASTE, una implementación de CRF, es de aproximadamente 46.2% en su máximo. Esto se debe a que en biología, hay muchas excepciones al modelo estándar, como genes superpuestos, genes anidados y empalme alternativo. Tener modelos que incluyan todas esas excepciones a veces produce peores predicciones; esta es una compensación no trivial. No obstante, la tecnología está mejorando y dentro de los próximos cinco años, habrá más datos experimentales para impulsar el desarrollo de la identificación computacional de genes, lo que a su vez ayudará a generar una mejor comprensión de la sintaxis del ADN.

This page titled [9.8: Conclusión, Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [9.8: Conclusion, Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## Bibliografía

- 1.R. Guigo (1997). “Identificación computacional de genes: un problema abierto”.
2. “Región intergénica”. <http://en.Wikipedia.org/wiki/Intergenic> región
- 3.Campo Aleatorio Condicional. Wikipedia. [http://en.Wikipedia.org/wiki/Conditional\\_bitstream/hand.../svmachine.htm](http://en.Wikipedia.org/wiki/Conditional_bitstream/hand.../svmachine.htm) campo aleatorio 4.

---

Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 10: Plegamiento de ARN

- 10.1: Motivación y Propósito
- 10.2: Química del ARN
- 10.3: Origen y Funciones del ARN
- 10.4: Estructura del ARN
- 10.5: Problema de plegamiento de ARN y enfoques
- 10.6: Evolución del ARN
- 10.7: Aproximación probabilística al problema del plegamiento del ARN
- 10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía

---

This page titled [10: Plegamiento de ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 10.1: Motivación y Propósito

El ARN (ácido ribonucleico) como molécula se ha postulado como el origen de la vida. Aunque durante mucho tiempo se consideró nada más que un intermediario entre el código en el ADN y las proteínas funcionales, se ha demostrado que el ARN cumple muchas funciones diferentes, abarcando todo el ámbito de la genómica. Parte de la causa de su versatilidad son las muchas conformaciones posibles en las que se puede encontrar el ARN. Al estar conformado por una estructura más flexible que el ADN, el ARN exhibe interesantes y variadas estructuras que pueden informarnos sobre sus múltiples propósitos. Ciertas estructuras del ARN, por ejemplo, se prestan a actividades catalíticas mientras que otras sirven como ARNt, y ARNm que son tan importantes durante el proceso de convertir el código del ADN en proteínas. El objetivo de este capítulo es aprender métodos que puedan explicar, o incluso predecir la estructura secundaria del ARN con la esperanza de que arrojan luz sobre las muchas propiedades de esta molécula versátil.

Para lograr esto, primero analizamos el ARN desde una perspectiva biológica y explicamos los roles biológicos conocidos del ARN. Luego, estudiamos los diferentes métodos que existen para predecir la estructura del ARN. Existen dos enfoques principales para el problema del plegamiento del ARN: 1) predecir la estructura del ARN basada en la estabilidad termodinámica de la molécula, y buscar un óptimo termodinámico 2) modelos probabilísticos que intentan encontrar los estados de la molécula de ARN en un óptimo probabilístico.

Finalmente, podemos usar datos evolutivos para aumentar la confianza de nuestras predicciones por estos métodos.

---

This page titled [10.1: Motivación y Propósito](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **10.1: Motivation and Purpose** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.2: Química del ARN

El ARN consiste en un azúcar de 5 carbonos, ribosa, que se une a una base orgánica (ya sea adenina, uracilo, citosina o guanina). Hay dos diferencias bioquímicas entre ADN y ARN:

1. el azúcar de 5 carbonos no tiene grupo hidroxilo en la posición 5
2. la presencia de uracilo en el ARN que es la forma no metilada de timina en lugar de solo timina.

La presencia de ribosa en el ARN hace que su estructura sea más flexible que el ADN, permitiendo que la molécula de ARN se pliegue y haga enlaces dentro de sí mismo lo que hace que el ARN monocatenario sea más que el ADN monocatenario

---

This page titled [10.2: Química del ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [10.2: Chemistry of RNA](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.3: Origen y Funciones del ARN

La gente inicialmente creía que el ARN solo actuaba como un intermedio entre el código de ADN y la proteína, sin embargo, a principios de los 80, el descubrimiento de ARN catalíticos (ribozimas) amplió la perspectiva sobre lo que esta molécula realmente puede hacer en los seres vivos. Sidney Altman y Thomas Cech descubrieron la primera ribozima, la RNasa P que es capaz de escindir la cabeza del ARNt. Los intrones autoempalmantes (intrones del grupo I) fueron también una de las primeras ribozimas que fueron descubiertas. No necesitan ninguna proteína como catalizadores para empalmar. El ARN monocatenario o bicatenario también sirve como agente de almacenamiento y replicación de información en algunos virus.

La **hipótesis del mundo del ARN**, propuesta por Walter Gilbert en 1986, sugiere que el ARN fue el precursor de la vida moderna. Se basa en el hecho de que el ARN puede tener tanto almacenamiento de información como actividad catalítica al mismo tiempo, siendo ambas características fundamentales de un sistema vivo. En definitiva, la hipótesis de RNA World dice que, debido a que el ARN puede tener un papel catalítico en las células y hay evidencia de que el ARN puede autorreplicarse sin depender de otras moléculas, un RNA World es un precursor plausible del mundo actual basado en ADN y proteínas. Aunque hasta el día de hoy no hay ARN autorreplicante natural encontrado *in vivo*, moléculas de ARN autorreplicantes se han creado en laboratorio mediante selección artificial. Por ejemplo, se ha demostrado que una construcción químérica de una ribozima de ligasa natural con un dominio de unión a molde seleccionado *in vitro* es capaz de replicar al menos una vuelta de una hélice de ARN. Por ello, Gilbert propuso el ARN como un origen plausible de por vida. La teoría sugiere que a través de la evolución, el ARN ha pasado su papel de almacenamiento de información al ADN, una molécula más estable y una menos propensa a la mutación. Luego, el ARN asumió el papel de intermediario entre el ADN y las proteínas, lo que asumió parte del papel catalítico del ARN en la célula. Así, los científicos a veces se refieren al ARN como fósiles moleculares. A pesar de que el ARN ha perdido gran parte de su funcionalidad de almacenamiento de información frente al ADN y sus propiedades funcionales a las proteínas, el ARN sigue desempeñando un papel integral en los organismos vivos. Por ejemplo, la porción catalítica del ribosoma, es decir, la parte funcional principal del complejo ribosómico consiste en ARN. El ARN también tiene papeles reguladores en la célula, y básicamente sirve como un agente para que la célula sienta y reaccione al medio ambiente.

### Ribointerruptores

Los ARN reguladores tienen diferentes familias, y uno de los más importantes son los **ribointerruptores**. Los ribointerruptores están involucrados en diferentes niveles de regulación génica. En algunas bacterias, las regulaciones importantes las realizan familias de ARN simples. Un ejemplo es el termosensor en *Listeria*, un ribointerruptor que bloquea los ribosomas a baja temperatura (ya que los enlaces de hidrógeno son más estables). El ARN entonces forma una conformación semibicatenaria que no se une al ribosoma y apaga el ribosoma. A temperaturas más altas (37 °C), la doble cadena se abre y permite que el ribosoma se adhiera a cierta región en el ribointerruptor, haciendo posible una vez más la traducción. Otro famoso Riboswitch es el riboswitch adenina (y en general los ribointerruptores de purina), que regulan la síntesis de proteínas. Por ejemplo el ARNm *ydhl* que tiene un tallo terminador al final y lo bloquea de la traducción, pero cuando la concentración de Adenina en- se pliega en la célula, se une al ARNm y cambia su conformación de tal manera que el tallo terminador desaparece.

### MicroRNAs

Hay otros tipos de ARN como los **microARN**, una variante más moderna del ARN (relativamente). Su descubrimiento reveló una nueva capa no proteica de regulación génica (por ejemplo, los miARN EVF-2 y HOTAIR). El EVF-2 es interesante porque se transcribe a partir de un potenciador ultraconservado, y se separa de la cadena de transcripción formando una horquilla, y luego regresa al mismo potenciador (junto con una proteína Dlx-2) y regula su actividad. El ARN de HOTAIR induce cambios en el estado de la cromatina y regula la metilación de las Histonas, lo que a su vez silencia el cúmulo de HOX-D.

### Otros tipos de ARN

También podemos ver tipos de ARN no codificantes.

- Los **PIRNAs** son la clase más grande de moléculas pequeñas de ARN no codificantes en animales. Están involucrados principalmente en el silenciamiento de los transposones, pero probablemente tengan muchas funciones. También están involucrados en modificaciónes epigenéticas y silenciamiento génico postranscripcional.

- Los LncRNAs son transcritos largos producidos que operan funcionalmente como ARN y no se traducen en proteínas. Muchos estudios implican a los LncRNAs en los modos epigenéticos, tal vez actuando como un mecanismo de focalización o como un andamio molecular para las proteínas Polycomb. Es probable que los lncRNAs posean numerosas funciones, muchos son nucleares, muchos son citoplásmicos.

---

This page titled [10.3: Origen y Funciones del ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [10.3: Origin and Functions of RNA](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.4: Estructura del ARN

Hemos aprendido sobre diferentes funciones del ARN, y ya debería quedar claro cuán fundamental es el papel del ARN en los sistemas vivos. Porque es imposible entender cómo el ARN realmente hace todas estas actividades en la célula, sin saber cuál es su estructura, en esta parte vamos a investigar la estructura del ARN.

La estructura del ARN se puede estudiar en tres niveles diferentes:

1. Estructura primaria: la secuencia en la que se alinean las bases (U, A, C, G).
2. Estructura secundaria: el análisis 2-D de los enlaces [hidrógeno] entre diferentes partes del ARN. En otras palabras, donde el ARN se vuelve bicatenario, donde el ARN forma una horquilla o un bucle u otras formas similares.
3. Estructura terciaria: la estructura tridimensional completa del ARN, es decir, cómo se dobla la cuerda, dónde se retuerce y tal.

Como se mencionó anteriormente, la presencia de ribosa en el ARN le permite plegarse y crear doble hélice consigo mismo. La estructura primaria es bastante fácil de obtener a través de la secuenciación del ARN. Nos interesa principalmente entender la estructura secundaria del ARN: donde los bucles y los enlaces de hidrógeno forman y crean los atributos funcionales del ARN. Idealmente, nos gustaría estudiar la estructura terciaria porque este es el estado final del ARN, y lo que le da su verdadera funcionalidad. Sin embargo, la estructura terciaria es muy difícil de calcular y está más allá del alcance de esta conferencia.

A pesar de que estudiar la estructura secundaria puede ser complicado, hay algunas ideas simples que funcionan bastante bien en su predicción. A diferencia de las proteínas, en el ARN, la mayor parte de la energía libre estabilizadora para la molécula proviene de su estructura secundaria (más que terciaria en el caso de las proteínas). Los ARN inicialmente se pliegan en su estructura secundaria y luego forman su estructura terciaria, y por lo tanto hay hechos muy interesantes que podemos aprender sobre una determinada molécula de ARN con solo conocer su estructura secundaria.

Finalmente, otra gran propiedad de la estructura secundaria es que suele estar bien conservada en la evolución, lo que nos ayuda a mejorar las predicciones de la estructura secundaria y también a encontrar ARNc (ARN no codificantes) s. existen representaciones ampliamente utilizadas para la estructura secundaria del ARN:

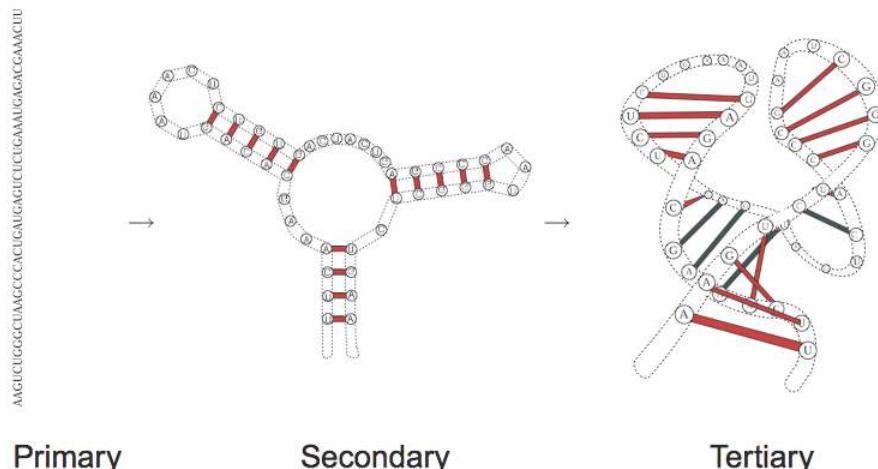


Figura 10.1: Representación gráfica de la jerarquía de la complejidad de la estructura del ARN © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

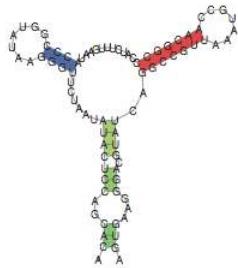


Figura 10.2: La representación típica de la estructura secundaria del ARN en los libros de texto. Muestra claramente la subestructura secundaria en ARN. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

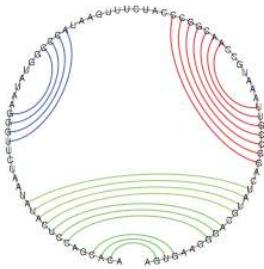


Figura 10.3: Dibujo gráfico donde la espina dorsal es un círculo y los emparejamientos de bases son los arcos dentro del círculo. Tenga en cuenta que la gráfica es planaria exterior, es decir, los arcos no se cruzan. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Formalmente: Una estructura secundaria es una gráfica etiquetada de vértices en  $n$  vértices con una matriz de adyacencia  $A = (a_{ij})$  cumpliendo:

- $a_{i, i+1} = 1$  para  $1 \leq i \leq n-1$  (columna vertebral continua)
- Para cada  $i$ ,  $1 \leq i \leq n$  hay como máximo uno  $a_{ij} = 1$  donde  $j \geq i + 1$  (una base solo forma un par entre sí en ese momento)
- Si  $a_{ij} = a_{kl} = 1$  y  $i < k < j$  then  $i < l < j$  (ignorar pseudo nudos)

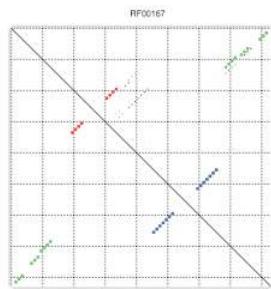


Figura 10.5: Una representación matricial, en la que se tiene un punto para cada par. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

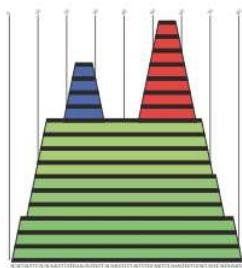


Figura 10.6: Parcela montañosa, en la que para parejas vas un paso arriba en la parcela y si no vas un paso a la derecha. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

This page titled [10.4: Estructura del ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [10.4: RNA Structure](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.5: Problema de plegamiento de ARN y enfoques

Por último, llegamos al punto en el que queremos estudiar la estructura del ARN. El objetivo aquí es predecir la estructura secundaria del ARN, dada su estructura primaria (o su secuencia). La buena noticia es que podemos encontrar la estructura óptima usando programación dinámica. Ahora para configurar nuestro marco de programación dinámica necesitaríamos un esquema de puntuación, el cual crearíamos utilizando la contribución de cada emparejamiento de bases a la estabilidad física de la molécula. Es decir, queremos crear una estructura con mínima energía libre, en nuestro modelo simple asignaríamos a cada par base un valor energético. 10.7

	A	C	G	U
A	+10	+10	+10	-2
C	+10	+10	-3	+10
G	+10	-3	+10	-1
U	-2	+10	-1	+10

Figura 10.7: Ejemplo de un esquema de puntuación para partidos de pares base. Tenga en cuenta que G-U puede formar un par de bamboleo en ARN.

La estructura óptima va a ser la que tenga un mínimo de energía libre y por convención la energía negativa se estabiliza, y la energía positiva no se estabiliza. Usando este marco, podemos usar programación dinámica (DP) para calcular la estructura óptima porque 1) este esquema de puntuación es aditivo 2) no permitimos pseudo nudos, lo que significa que podemos dividir el ARN en dos más pequeños que son independientes, y resolver el problema de estos ARN más pequeños.

Queremos encontrar una matriz DP  $E_{ij}$ , en la que calculemos la energía libre mínima para la subsecuencia  $i$  a  $j$ . El primer acercamiento a esto es el algoritmo de Nussinov.

### Algoritmo de Nussinov

La fórmula de recursión para este problema fue descrita por primera vez por Nussinov en 1978.

La intuición detrás de este algoritmo es la siguiente: dada una subsecuencia  $[i, j]$ , o no hay borde que se conecte a la  $i$ -ésima base (lo que significa que está desapareada) o hay algún borde que conecte la  $i$ -ésima base a la base  $k$  donde  $i < k \leq j$  (es decir, la  $i$ -ésima base está emparejada con la base  $k$ -ésima). En el caso de que la  $i$ -ésima base no esté apareada, la energía de la subsecuencia,  $E_{i,j}$ , simplemente se reduce a la energía de la subsecuencia de  $i + 1$  a  $j$ ,  $E_{i+1,j}$ . Este es el primer término de la relación de recursión de Nussinov. Sin embargo, si la  $i$ -ésima base se empareja con la base  $k$ -ésima, entonces  $E_{i,j}$  se reduce a la contribución de energía del emparejamiento  $i, k, \beta_{ik}$ , más la energía de las subsecuencias formadas dividiendo  $[i + 1, j]$  alrededor de  $k$ ,  $E_{i+1,k-1}$  y  $E_{k+1,j}$ . Al elegir la  $k$  que minimiza ese valor se obtiene el segundo término de la relación de recursión de Nussinov. La energía óptima de la subsecuencia, por lo tanto, es el mínimo de la energía de la subsecuencia cuando la  $i$ -ésima base se empareja con la base  $k$ -ésima óptima y cuando la  $i$ -ésima base no está apareada. Esto produce la relación global descrita en la figura 10.8.



$$E_{ij} = \min \left\{ E_{i+1,j}, \min_{k, \Pi_{ik}=1} \{ E_{i+1,k-1} + E_{k+1,j} + \beta_{ik} \} \right\}$$

- ▶  $E_{ij}$  ... Minimum energy of subsequence  $i \dots j$
- ▶  $\beta_{ij}$  ... Energy contribution of pair  $(i, j)$
- ▶  $\Pi_{ij}$  is 1 if bases  $i$  and  $j$  can pair and 0 otherwise.

Figura 10.8: La fórmula de recursión para el algoritmo de Nussinov, junto con una representación gráfica de cómo funciona.

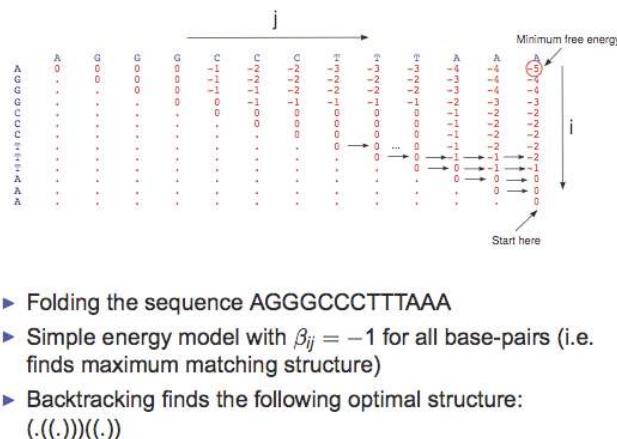
A partir de esta relación de recursión, podemos ver que la matriz DP contendrá entradas para todos  $i, j$  donde  $1 \leq i \leq n$  and  $1 \leq j \leq n$  and  $i \neq j$ . Es decir, la matriz será  $n \times n$  y sólo contendrá entradas en el triángulo superior derecho. La matriz se inicializa primero de tal manera que todos los valores en la diagonal son iguales a cero. Luego iteraremos sobre

$i = n - 1 \dots 1 \dots 1$  y  $j = i + 1 \dots n$  (abajo hacia arriba, de izquierda a derecha) y rellenamos cada entrada según la relación de recurrencia. La puntuación general es la puntuación de la subsecuencia  $[1, n]$ , que es la esquina superior derecha de la matriz. La Figura 10.9 ilustra este procedimiento.

Cuando calculamos la energía mínima libre, muchas veces nos interesa el pliegue correspondiente. Para recuperar el pliegue óptimo del algoritmo DP, se utiliza una matriz de rastreo para almacenar punteros desde cada entrada hasta su entrada principal. La Figura 10.10 describe el algoritmo de retroceso.

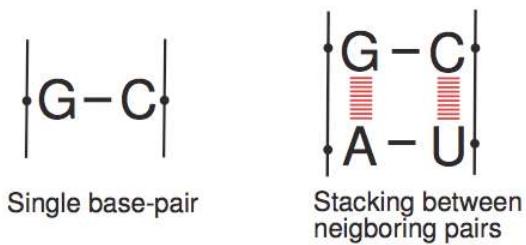
Este modelo es muy simplista y hay algunas limitaciones al mismo. El algoritmo de Nussinov, tal como se implementa ingenuamente, no toma en cuenta algunos de los aspectos limitantes del plegamiento del ARN. Lo más importante es que no considera las interacciones de apilamiento entre pares vecinos, un factor vital (incluso más que los enlaces de hidrógeno) en el plegamiento del ARN. Figura 10.11

Por lo tanto, es deseable integrar factores biofísicos en nuestra predicción. Una mejora, por ejemplo, es asignar energías a las caras gráficas (elementos estructurales en la figura 10.12), en lugar de pares de bases simples. La energía total de la estructura se convierte entonces en la suma de las energías de las subestructuras. Las energías de apilamiento se pueden calcular fundiendo oligonucleótidos experimentalmente.



© Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 10.9: La diagonal se inicializa a 0. Luego, la tabla se llena de abajo hacia arriba, de izquierda a derecha ac- cording a la relación de recurrencia. En este ejemplo, los emparejamientos de bases complementarios se puntúan como -1 y los emparejamientos no complementarios se puntúan como 0. La puntuación óptima para toda la secuencia se encuentra en la esquina superior derecha.



© Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 10.11: Apilamiento entre pares de bases vecinas en ARN. La estructura aromática plana de la base provoca interacciones cuánticas entre bases apiladas y cambia su estabilidad física.

## Algoritmo Zuker

Por lo tanto, utilizamos una variante que incluye energías de apilamiento para calcular la estructura del ARN. A esto se le llama el algoritmo Zuker. Al igual que Nussinovs, asume que la estructura óptima es la que tiene la energía libre de equilibrio más baja. Sin embargo, incluye las contribuciones de energía total de las diversas subestructuras que está parcialmente determinada por la energía de apilamiento. Algunos algoritmos modernos de plegamiento de ARN utilizan este algoritmo para predicciones de estructura de ARN.

En el algoritmo de Zuker, tenemos cuatro casos diferentes que tratar. La Figura 10.13 muestra un esquema gráfico de las etapas de descomposición. El procedimiento requiere cuatro matrices.  $F_{ij}$  contiene la energía libre de la estructura óptima global de la subsecuencia  $x_{ij}$ . La base recién agregada puede estar desemparejada o puede formar un par. Para

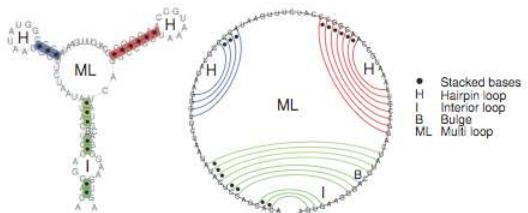
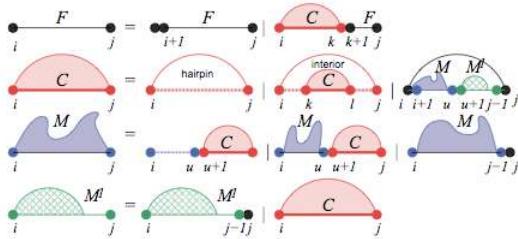


Figura 10.12: Varias subestructuras internas en un ARN plegado. Una horquilla consiste en un bucle terminal conectado a una región emparejada, un bucle interno es una región desapareada dentro de la región emparejada. Un Bulge es un caso especial de un bucle interior con un solo par erróneo. un Multi loop es un bucle que consiste en múltiples de estos componentes (en este ejemplo dos horquillas y una región emparejada, todas conectadas a un bucle). © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

en este último caso, se introduce la matriz auxiliar  $C_{ij}$ , que contiene la energía libre de la subestructura óptima de  $x_{ij}$  bajo la restricción de que  $i$  y  $j$  están emparejados. Esta estructura cerrada por un par de bases puede ser una horquilla, un bucle interior o un bucle múltiple.

El caso de horquilla es trivial porque no es necesaria una mayor descomposición. La caja de bucle interior también es simple porque vuelve a reducir al mismo paso de descomposición. El paso multi-loop es más complicado. La energía de un bucle múltiple depende del número de componentes, es decir, subestructuras que emanan del bucle. Para realizar un seguimiento implícito de este número, se necesitan dos matrices auxiliares adicionales.  $M_{ij}$  mantiene la energía libre de la estructura óptima de  $x_{ij}$  bajo la restricción de que  $x_{ij}$  es parte de un bucle múltiple con al menos un componente.  $M_{ij}^1$  contiene la energía libre de la estructura óptima de  $x_{ij}$  bajo la restricción de que  $x_{ij}$  es parte de un multi-loop y tiene exactamente un componente cerrado por par ( $i, k$ ) con  $i < k < j$ . La idea es descomponer un bucle múltiple en dos partes arbitrarias de las cuales el primero es un multi-loop con al menos un componente y el segundo un multi-loop con exactamente un componente y comenzando con un par de bases.

Estas dos partes correspondientes a  $M$  y  $M^1$  pueden descomponerse adicionalmente en subestructuras que ya conocemos, es decir, intervalos desapareados, subestructuras cerradas por un par de bases, o multi-bucle (más cortos). (Las recursiones también se resumen en 10.13).



$$F_{ij} = \min \left\{ F_{i+1,j}, \min_{i < k \leq j} C_{ik} + F_{k+1,j} \right\}$$

$$C_{ij} = \min \left\{ \mathcal{H}(i,j), \min_{i < k < l < j} C_{kl} + \mathcal{I}(i,j;k,l), \min_{i < u < j} M_{i+1,u} + M_{u+1,j-1}^1 + a \right\}$$

$$M_{ij} = \min \left\{ \min_{i < u < j} (u - i + 1)c + C_{u+1,j} + b, \min_{i < u < j} M_{i,u} + C_{u+1,j} + b, M_{i,j-1} + c \right\}$$

$$M_{ij}^1 = \min \left\{ M_{i,j-1}^1 + c, C_{ij} + b \right\},$$

Figura 10.13: F describe el caso desapareado, C se describe por una de las tres condiciones: horquilla, bucle interior, o una composición de estructuras, es decir, un bucle múltiple. M1 es un bucle múltiple con un solo componente, donde M podría tener múltiples de ellos. El | icono es notación para “o”. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

En realidad, sin embargo, a temperatura ambiente (o temperatura celular), el ARN no se encuentra realmente en un solo estado, sino que varía en un conjunto termodinámico de estructura. Los pares de bases pueden romper sus vínculos con bastante facilidad, y aunque podríamos encontrar un óptimo absoluto en términos de energía libre, podría darse el caso de que exista otra estructura subóptima que es muy diferente de lo que predijo e y que tiene un papel importante en la célula. Para solucionar el problema podemos calcular las probabilidades de pares de bases para obtener el conjunto de estructuras, y luego podemos tener una idea mucho mejor de cómo es probablemente la estructura del ARN. Para ello, utilizamos el factor Boltzman:

$$\text{Prob}(\mathcal{S}) = \frac{\exp(-\Delta G(\mathcal{S})/RT)}{Z}$$

Esto nos da la probabilidad de una estructura dada, en un sistema termodinámico. Necesitamos normalizar la temperatura usando la función de partición Z, que es la suma ponderada de todas las estructuras, en función de su factor de Boltzman:

$$Z = \sum_{\mathcal{S}} \exp(-\Delta G(\mathcal{S})/RT)$$

También podemos representar este conjunto gráficamente, usando una gráfica de puntos para visualizar las probabilidades del par base. Para calcular la probabilidad específica para un par base  $(i, j)$ , necesitamos calcular la función de partición, que viene dada por la siguiente fórmula:

$$p_{ij} = \frac{\widehat{Z}_i Z_{i+1,-1} \exp(-\beta_j/RT)}{Z}$$

Para calcular Z (la función de partición sobre toda la estructura), usamos la recursión similar al algoritmo de Nussinovs (conocido como algoritmo McCaskill). La función de partición interna se calcula usando la fórmula:

$$Z_{ij} = Z_{i+1,j} + \sum_{\substack{1 \leq k \leq j \\ n_k=1}} Z_{i+1,k-1} Z_{k+1,j} \exp(-\beta_{ik}/RT)$$

Con cada una de las adiciones correspondientes a una división diferente en nuestra secuencia como ilustra la siguiente figura. Obsérvese que la suma se multiplica a las funciones de energía ya que se expresa como exponencial.

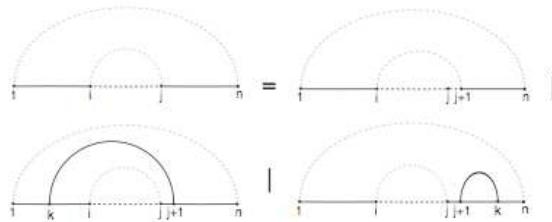


Del mismo modo, la función de partición externa se calcula con una misma idea usando la fórmula:

$$\widehat{Z}_{ij} = \widehat{Z}_{i,j+1} + \sum_{\substack{1 \leq k < i \\ \Pi_{k,j+1}=1}} \widehat{Z}_{k,j+1} \exp(-\beta_{k,j+1}/RT) Z_{k+1,i-1}$$

$$+ \sum_{\substack{j+2 \leq k \leq n \\ \Pi_{k,j+1}=1}} \widehat{Z}_{i,k} \exp(-\beta_{k,j+1}/RT) Z_{j+2,k-1}$$

correspondientes a diferentes divisiones en el área fuera de los pares de bases (i, j).



This page titled [10.5: Problema de plegamiento de ARN y enfoques](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **10.5: RNA Folding Problem and Approaches** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.6: Evolución del ARN

Es útil para entender la evolución de la estructura del ARN, ya que revela datos valiosos, y también puede darnos pistas para refinar nuestras predicciones de estructura. Cuando analizamos los ARN funcionalmente importantes a lo largo del tiempo, nos damos cuenta de que sus nucleótidos han cambiado en algunas partes, pero su estructura está bien conservada.

En el ARN hay muchas mutaciones compensatorias y mutaciones consistentes. En una mutación consistente, la estructura no cambia, por ejemplo, un par AU muta para formar un par G. En una mutación compensatoria en realidad hay dos mutaciones, una altera la estructura, pero la segunda mutación la restaura, por ejemplo un par AU cambia a una CU que no se empareja bien, pero a su vez la U muta a una G para restaurar un par CG. En un mundo ideal, si tenemos este conocimiento, esta es la clave para predecir la estructura del ARN, porque la evolución nunca miente. Podemos calcular el contenido de información mutua para dos ARN diferentes y compararlo. En otras palabras, se comparan las probabilidades de que dos estructuras de pares de bases estén de acuerdo aleatoriamente frente a si han evolucionado para ser conservar la estructura.

El contenido de información mutua se calcula a través de esta fórmula:

$$M_{ij} = \sum_{X,Y} f_{ij}(XY) \log \frac{f_{ij}(XY)}{f_i(X)f_j(Y)}$$

Si normalizamos estas probabilidades, y almacenamos el MI en bits, podemos trazarlo en un modelo 3D y rastrear las firmas evolutivas. De hecho, este fue el método para determinar la estructura de los ARN ribosómicos mucho antes de que fueran encontrados por cristalografía.

El verdadero problema es que no tenemos tanta información, así que lo que solemos hacer es combinar los métodos de predicción de plegamiento con información filogenética para obtener una predicción confiable. La forma más común de hacerlo es combinarlo con el algoritmo Zuker con algunas puntuaciones de covarianza. Por ejemplo, agregamos energía estabilizadora si tenemos una mutación compensatoria, y energía desestabilizadora si tenemos una mutación de un solo nucleótido.

---

This page titled [10.6: Evolución del ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [10.6: Evolution of RNA](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.7: Aproximación probabilística al problema del plegamiento del ARN

Secuencia codificante de ARN dentro del genoma Encontrar secuencias codificantes de ARN dentro del genoma es un problema muy difícil. Sin embargo hay formas de hacerlo. Una forma es combinar la información de estabilidad termodinámica, con una puntuación RNAold normalizada y luego podemos hacer una clasificación de Máquina de Vector de Soporte (SVM), y comparar la estabilidad termodinámica de la secuencia con algunas secuencias aleatorias del mismo contenido de GC y la misma longitud y ver cuántos estándares desviaciones es la estructura dada más estable que el valor esperado.

Podemos combinarlo con la medida evolutiva y ver si el ARN está más conservado o no. Esto nos da (con relativa precisión) una idea de si la secuencia genómica en realidad está codificando un ARN.

Hemos estudiado sólo la mitad de la historia. Si bien el enfoque termodinámico es una buena manera (y la clásica) de plegar los ARN, a alguna parte de la comunidad le gusta estudiarlo desde un aspecto diferente. Asumamos por ahora que no

### Secuencia codificante de ARN dentro del genoma

Encontrar secuencias codificantes de ARN dentro del genoma es un problema muy difícil. Sin embargo hay formas de hacerlo. Una forma es combinar la información de estabilidad termodinámica, con una puntuación de pliegue de ARN normalizada y luego podemos hacer una clasificación de Máquina de Vector de Soporte (SVM), y comparar la estabilidad termodinámica de la secuencia con algunas secuencias aleatorias del mismo contenido de GC y la misma longitud y ver cuántos estándares desviaciones es la estructura dada más estable que el valor esperado.

saber algo sobre la física del ARN o el factor Boltzman. En cambio, nos fijamos en el ARN como una cadena de letras para la que queremos encontrar la estructura más probable. Ya nos enteramos de los Modelos Ocultos de Markov en las conferencias anteriores. Son una buena manera de hacer predicciones sobre los estados ocultos de un sistema probabilístico. La pregunta es ¿podemos usar modelos Hidden Markov para el problema del plegamiento de ARN? La respuesta es sí.

Podemos combinarlo con la medida evolutiva y ver si el ARN está más conservado o no. Esto nos da (con relativa precisión) una idea de si la secuencia genómica en realidad está codificando un ARN.

Hemos estudiado sólo la mitad de la historia. Si bien el enfoque termodinámico es una buena manera (y la clásica) de plegar los ARN, a alguna parte de la comunidad le gusta estudiarlo desde un aspecto diferente.

Supongamos por ahora que no sabemos nada de la física del ARN o del factor Boltzman. En cambio, nos fijamos en el ARN como una cadena de letras para la que queremos encontrar la estructura más probable. Ya nos enteramos de los Modelos Ocultos de Markov en las conferencias anteriores. Son una buena manera de hacer predicciones sobre los estados ocultos de un sistema probabilístico. La pregunta es ¿podemos usar modelos Hidden Markov para el problema del plegamiento de ARN? La respuesta es sí.

Podemos representar la estructura del ARN como un conjunto de estados ocultos de puntos y corchetes (recordar la representación de paréntesis de puntos del ARN en la parte 3). Aquí hay una observación importante que hacer: las posiciones y los emparejamientos dentro del ARN no son independientes, por lo que no podemos simplemente tener un estado de corche-apertura sin ninguna consideración de los eventos que están sucediendo aguas abajo.

Por lo tanto, necesitamos extender el marco HMM para permitir correlaciones anidadas. Afortunadamente, ya existe el marco probabilístico para hacer frente a tal problema. Se le conoce como gramática estocástica libre de contexto (SCFG).

### Gramática libre de contexto en pocas palabras

Tienes:

- Conjunto finito de símbolos no terminales (estados) por ejemplo {A, B, C} y símbolos terminales, por ejemplo, {a, b, c}
- Conjunto finito de reglas de producción. e.g. {A → aB, B → AC, B → aa, → ab}
- Un inicial (inicio) no terminal

Quieres encontrar una manera de llegar de un estado a otro (o a una terminal).

$$A \rightarrow aB \rightarrow aAC \rightarrow aaaC \rightarrow aaaab$$

En un CFG estocástico, la única diferencia es que cada relación tiene una cierta probabilidad.e.g.:

$$P(B \rightarrow AC) = 0.25 P(B \rightarrow aa) = 0.75$$

La evaluación filogenética se combina fácilmente con scFg, ya que existen muchos modelos probabilísticos para datos filogenéticos. Los modelos probabilísticos no se discuten en detalle en esta conferencia pero la siguiente imagen básicamente da una analogía entre los modelos estocásticos y los métodos que hemos visto hasta ahora en la clase.

- Analogías al plegado termodinámico:
  - CYK  $\leftrightarrow$  Energía mínima libre (Nussinov/Zuker)
  - Algoritmo interior/exterior  $\leftrightarrow$  Funciones de partición (McCaskill)
- Analogías a los modelos de Hidden Markov:
  - CYK Mínimo  $\leftrightarrow$  Algoritmo de Viterbi
  - Algoritmo de interior/exterior  $\leftrightarrow$  Algoritmo
- Dado un SCFG parametrizado ( $\theta, \Omega$ ) y una secuencia x, el algoritmo de programación dinámica **Cocke-Younger-Kasami** (CYK) encuentra un árbol de análisis óptimo (probabilidad máxima) $\hat{\pi}$ :
 
$$\hat{\pi} = \text{ArgMaxProb}(\pi, x|\theta, \Omega)$$
- El algoritmo Inside, se utiliza para obtener la probabilidad total de la secuencia dado el modelo sumado sobre todos los árboles de análisis,

$$\text{Prob}(x|\Theta, \Omega) = \sum \text{Prob}(x, \pi|\Theta, \Omega) \quad (10.7.1)$$

## Aplicación de SCFG

- Predicción de la estructura secundaria de consenso: Pfold — First Phylos-SCFG

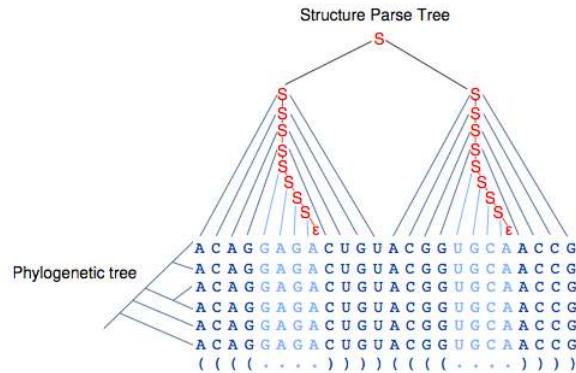


Figura 10.14: A) Secuencia única: Los símbolos terminales son bases o pares de bases, las probabilidades de emisión son frecuencias base en bucles y regiones pareadas B) FilosecFG: Los símbolos terminales son columnas de alineación simples o pareadas, Las probabilidades de emisión se calculan a partir del modelo filogenético y árbol usando Felsenstein AlgoritmoNosotros para tratar de entender mejor las interacciones ARN-ARN. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

- Identificación génica de ARN estructural: EvoFold

### 10.8

- Utiliza gramática Pfold
- Dos modelos competidores:
  - \* Modelo no estructural con todas las columnas tratadas como evolucionando independientemente
  - \* Modelo estructural con columnas dependientes e independientes
- Parametrización sofisticada

This page titled [10.7: Aproximación probabilística al problema del plegamiento del ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **10.7: Probabilistic Approach to the RNA Folding Problem** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía

Aún quedan muchos otros problemas que deben resolverse estudiando la estructura del ARN. En esta sección se perfilarán algunos de ellos.

### Otros problemas

Observe algunos de los problemas que se muestran gráficamente a continuación:

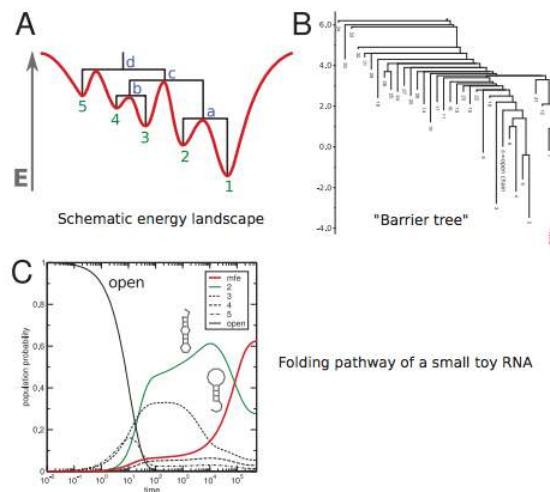


Figura 10.15: Podemos estudiar la cinética y las vías de plegamiento en mayor profundidad. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

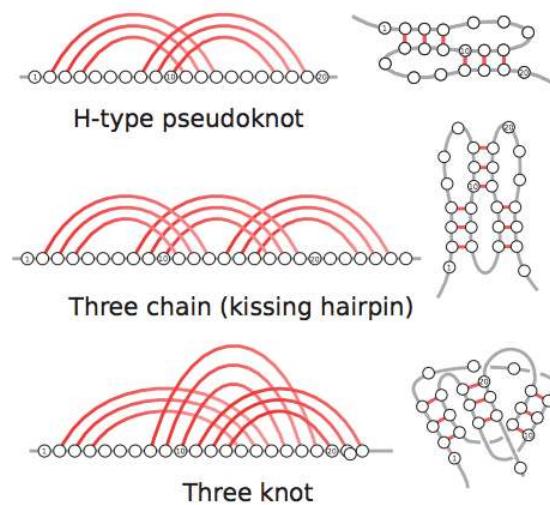


Figura 10.16: Podemos investigar pseudonudos. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

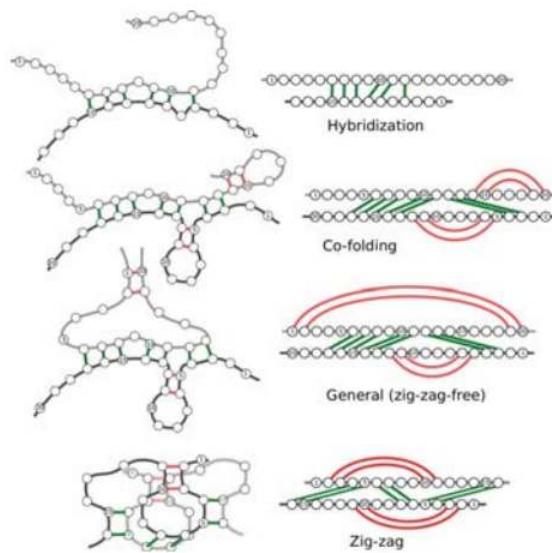


Figura 10.17: Podemos tratar de entender mejor las interacciones ARN-ARN. © Stefan Washietl. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

## Relevancia

Hay muchos ARN dentro de la célula aparte de los mRNAs, ARNr y ARNt. La pregunta es ¿cuál es la relevancia de todo este ARN no codificante? Algunos creen que es ruido resultado a través del experimento, algunos piensan que es solo ruido biológico que no tiene sentido en el organismo vivo. Por otro lado algunos creen que el ARN basura en realidad podría tener un papel importante como señales dentro de la célula y todo es realmente funcional, la verdad probablemente se encuentra en algún punto intermedio.

## Investigación actual

Hay regiones conservadas en el genoma que no codifican ninguna proteína, y ahora Stefan et al. las están investigando para ver si tienen estructuras lo suficientemente estables para formar ARN funcionales. Resulta que alrededor del 6% de estas regiones tienen señas de identidad de buena estructura de ARN, que sigue siendo 30000 elementos estructurales. El grupo ha anotado algunos de estos elementos, pero aún queda un largo camino por recorrer. Se han encontrado muchos miARN, snoRNAs y por supuesto muchos falsos positivos. ¡Pero hay resultados emocionantes que surgen en este tema! así que la nota final es, ¡es una muy buena zona para trabajar!

## Resumen y puntos clave

1. El espectro funcional de los ARN es prácticamente ilimitado
  - (a) Los ARN similares a las Ribozimas y Riboswitches contemporáneos podrían haber existido en un mundo de ARN. Algunos de ellos aún existen como fósiles vivos en las células actuales.
  - (b) Los ARN evolutivamente más jóvenes, incluidos los miARN y muchos ARNc largos, forman una capa reguladora no basada en proteínas.
2. La estructura del ARN es crítica para su función y se puede predecir computacionalmente
  - (a) Nussinov/Zuker: Estructura mínima de energía libre
  - (b) McCaskill: Función de partición y probabilidades de pares
  - (c) CYK/Interide-Outside: solución probabilística al problema usando SCFG
3. La información filogenética puede mejorar la predicción de estructuras
4. La biología computacional de los ARN es un campo activo de investigación con muchos problemas algorítmicos duros aún abiertos

## 10.10 Lectura adicional

- Visión general
  - Washietl S, Will S. et al. Análisis computacional de ARN no codificantes. Wiley Interdiscip Rev RNA. 2012; 10:1002/wrna.1134
- Función RNA: artículos de revisión de John Mattick
- Plegamiento de ARN de secuencia única
  - Nussinov R, Jacobson AB, algoritmo rápido para predecir la estructura secundaria del ARN monocatenario. Proc Natl Acad Sci U S A. 1980 Nov; 77: (11) 6309-13
  - Zuker M, Stiegler P Plegamiento óptimo por computadora de grandes secuencias de ARN utilizando termodinámica e información auxiliar. Nucleic Acids Res. 1981 Ene; 9: (1) 133-48
  - McCaskill JS La función de partición de equilibrio y las probabilidades de unión de pares de bases para la estructura secundaria de ARN. Biopolímeros. 1990; 29: (6-7) 1105-19
  - Dowell RD, Eddy SR, Evaluación de varias gramáticas estocásticas ligeras sin contexto para la predicción de estructura secundaria de ARN. BMC Bioinformática. 2004 jun; 5:71
  - Do CB, Woods DA, Batzoglou S, ContraFold: predicción de estructura secundaria de ARN sin modelos basados en la física. Bioinformática. 2006 Jul; 22: (14) e90-8
- Plegamiento de ARN consenso
  - Hofacker IL, Fekete M, Stadler PF, Predicción de estructura secundaria para secuencias de ARN alineadas. J Mol Biol. 2002 jun; 319: (5) 1059-66
  - Knudsen B, Hein J, predicción de estructura secundaria de ARN usando gramáticas estocásticas libres de contexto e historia evolutiva. Bioinformática. 1999 Jun; 15: (6) 446-54
- Búsqueda de genes de ARN
  - Pedersen JS, Bejerano G, Siepel A, Rosenblom K, Lindblad-Toh K, Lander ES, Kent J, Miller W, Haussler D Identificación y clasificación de estructuras secundarias de ARN conservadas en el genoma humano. PLoS Comput Biol. 2006 Abr; 2: (4) e33
  - Washietl S, Hofacker IL, Stadler PF, Predicción rápida y confiable de ARN no codificantes. Proc Natl Acad Sci U S A. 2005 Feb; 102: (7) 2454-9

## Bibliografía

1. [1] R Durbin. Análisis de Secuencia Biológica.
2. [2] W. Gilbert." origen de la vida: El mundo del rna". Naturaleza., 319 (6055) :618, 1986.
3. [3] Rachel Sealoff, 2012. Información extra extraída de la recitación 5 diapositivas.
4. [4] Z. Wang, M. Gestein y M. Snyder. RNA-seq: una herramienta revolucionaria para la transcriptómica. Nat Rev Genet., 10 (1) :57—63, 2009.
5. [5] Stefan Washietl, 2012. Todas las imágenes/fórmulas cortesía de las diapositivas de Stefan.
6. [6] R. Tejedor. Biología Molecular. 3<sup>a</sup> edición.

---

This page titled [10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [10.8: Advanced topics, Summary and key points, Further Reading, Bibliography](#) by Manolis Kellis et al. is licensed [CC BY-NC-SA 4.0](#).  
Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 11: Modificaciones de ARN

- 11.1: Introducción
- 11.2: Regulación Postranscripcional
- 11.3: ¿Qué hemos aprendido?

---

This page titled [11: Modificaciones de ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 11.1: Introducción

Muchas ideas en biología se basan en conocer los niveles de proteína en una célula. La abundancia de proteínas a menudo se extrae de los niveles de ARNm correspondientes. Esta extrapolación se realiza ya que es relativamente fácil medir los niveles de ARNm. Además, durante mucho tiempo, se pensó que toda la regulación de la expresión ocurrió antes de la formación del ARNm. Ahora, se sabe que la expresión sigue siendo regulada en la etapa de traducción. La Figura 1 muestra que los datos disponibles para la regulación postranscripcional son mínimos e ilustra un ejemplo de cómo los niveles de ARNm no son indicativos de la abundancia de proteínas.



Figura 11.1: El ARNm no siempre es un proxy apropiado para los niveles de proteína.

Existen muchos factores que pueden estar afectando la forma en que se traduce el ARNm, provocando que el nivel de ARNm no esté directamente relacionado con los niveles de proteína. Estos factores incluyen:

1. Tasas de elongación de la traducción
  - depende del sesgo de uso de codones, adaptación de ARNt y edición
2. Tasas de iniciación de la traducción
  - : depende de la frecuencia AUG, la presencia TOP, el tipo de iniciación (dependiente de CAP/IRE) y las estructuras secundarias
3. Tasas de terminación de la traducción
  - depende de la identidad del codón
4. Tasas de degradación del ARNm
  - : depende de la longitud de la cola de poliA, el tapado, la edición del ARNm y la
5. Tasas de degradación de proteínas
  - : depende de las secuencias PEST, la estabilidad de las proteínas, las regiones no estructuradas y la presencia de aminoácidos polares
6. Elementos reguladores cis y trans
  - depende de elementos ricos en UA, miARN, ARNc y proteínas de unión a ARN



Figura 11.2: Discrepancia entre los niveles de ARNm y la abundancia de proteínas.

This page titled [11.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [11.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 11.2: Regulación Postranscripcional

### Fundamentos de la traducción de proteínas

Para conocer los conceptos básicos de transcripción y traducción, consulte la Conferencia 1, secciones 4.3 - 4.5.



Figura 11.3: Las señales genéticas en las que se mapean aminoácidos para tres secuencias de nucleótidos específicas.

El código genético es casi universal.

#### FAQ

P: ¿Por qué el código genético es tan similar entre los organismos?

R: El material genómico no solo se transmite verticalmente (de los padres) sino también horizontalmente entre organismos. Esta interacción génica crea una presión evolutiva para un código genético universal.

#### FAQ

P: ¿Qué explica las ligeras diferencias en el código genético entre organismos?

R: La llegada evolutiva tardía/temprana de aminoácidos puede explicar las diferencias. Además, ciertas especies (por ejemplo, bacterias en respiraderos de aguas profundas) tienen más recursos para sintetizar aminoácidos específicos, por lo que favorecerán a los del código genético.

#### ¿Sabías?

La treonina y la alanina a menudo son intercambiadas accidentalmente por ARNt sintetasa porque se originaron a partir de un aminoácido.

### Traducción de medición

La eficiencia de la traducción se define como

$$T_{eff} = \frac{[\text{mRNA}]}{[\text{protein}]}$$

Nos interesa ver cuánto de nuestro ARNm se traduce en proteína, es decir, la eficiencia. Sin embargo, medir específicamente cuánto ARNm se convierte en proteína es una tarea difícil, una que requiere un poco de creatividad. Hay una variedad de formas de abordar este problema, pero cada una tiene sus propias caídas:

1. Medir los niveles de ARNm y proteínas directamente

Pitfall: No considera las tasas de síntesis y degradación. Este método mide los niveles de proteína para el ARNm 'viejo' ya que existe un retraso de tiempo de ARNm a proteína.

2. Usar drogas para inhibir la transcripción y la traducción. Dificultad: Las drogas tienen efectos secundarios que alteran la traducción

3. Fusión artificial de proteínas con etiquetas

Pitfall: Las etiquetas de proteínas pueden afectar la estabilidad de las proteínas

4. Marcador de pulso con nucleósidos o aminoácidos radiactivos (SILAC) \*\*en uso hoy\*\*

Dificultad: No ofrece información sobre los cambios dinámicos: es simplemente un snapshot de los niveles de ARNm y proteína resultantes después de X horas 193

Otra técnica común es usar el "perfil de ribosomas" para medir la traducción de proteínas en la resolución de subcodones. Esto se hace congelando ribosomas en el proceso de traducción y degradando las secuencias no protegidas por ribosomas. En este punto, las secuencias pueden ser reensambladas y la frecuencia con la que se traduce una región puede ser interpolada. La desventaja de

usar estas huellas de ribosomas, para ver qué regiones se están traduciendo, es que se pierden regiones entre ribosomas. Esta técnica requiere una RNA-seq en paralelo.

La pregunta sigue siendo, ¿por qué es ventajoso el perfilado de ribosomas? Esta técnica es un mejor enfoque para medir la abundancia de proteínas ya que:

1. Es una mejor medida de la abundancia de proteínas
2. Es independiente de la degradación proteica (en comparación con la relación abundancia de proteínas/ARNm)
3. Nos permite medir tasas de traducción específicas de codones

Usando perfiles de ribosomas, es posible ver qué codón se está decodificando: esto se hace mapeando las huellas de ribosomas y luego descifrando el codón de traducción basado en la longitud de la huella. Podemos verificar nuestra predicción mapeando perfiles de codones traducidos basados en la periodicidad (tres bases en un codón). La técnica se puede mejorar aún más mediante el uso de fármacos anti-traducción como harringtonina y ciclohexamida. La ciclohexamida bloquea la elongación y la Harringtonina inhibe la iniciación. El último puede ser utilizado para encontrar los puntos de partida (qué genes están a punto de traducirse). La Figura 4 muestra los efectos de los fármacos sobre los perfiles de ribosomas.



Figura 11.4: Representación de perfiles de ribosomas cuando se usan ciclohexamida (congelación por elongación) o Harringtonina (congelación de inicio).

Esta técnica tiene mucho más que ofrecer que simplemente cuantificar la traducción. El perfilado de ribosomas permite:

1. Predicción de isoformas alternativas (diferentes lugares donde puede comenzar la traducción) images/AltIsoforms.png



Figura 11.5: Perfil ribosómico cuando se usa harringtonina vs. sin fármaco. Los picos rojos muestran los diferentes lugares donde puede comenzar la iniciación de la traducción, representando las diferentes isoformas posibles.

2. Predicción de ORF no identificados (marcos de lectura abiertos)



Figura 11.2.1: Copiar y Pegar Subtitulado aquí. (Copyright; autor vía fuente)

Figura 11.6: Perfil ribosómico cuando se usa harringtonina vs. sin fármaco. Los picos rojos previamente no identificados ORF.

3. Comparación de la traducción a través de diferentes condiciones ambientales



Figura 11.7: Perfil ribosómico cuando durante condiciones ricas y condiciones de inanición. Estas imágenes muestran la dramática disminución en la traducción de proteínas durante la inanición. El perfil de ARNm no es indicativo de esto.

4. Comparando la traducción a través de las etapas

Así, vemos que el perfilado de ribosomas es una herramienta muy poderosa con mucho potencial para revelar información previamente esquiva sobre la traducción de un genoma.

## Evolución de codones

### Conceptos básicos

Algo que hay que dejar claro es que los codones no se utilizan con frecuencias iguales. De hecho, los codones que pueden considerarse óptimos difieren entre diferentes especies según la estabilidad del ARN, el sesgo de mutación específica de cadena, la eficacia transcripcional, la composición de GC, la hidropatía proteica y la eficiencia traduccional. Asimismo, los isoaceptores de

ARNt no se utilizan con frecuencias iguales dentro y entre especies. La motivación para la siguiente sección es determinar cómo podemos medir este sesgo de codones.

### Medidas del sesgo de codón

Existen algunos métodos para llevar a cabo esta tarea:

- a) Calcular la frecuencia de codones óptimos, que se define como codones “óptimos”/suma de codones “óptimos” y “no óptimos”. Las limitaciones de este método son que esto requiere conocer qué codón es reconocido por cada ARNt y asume que la abundancia de ARNt está altamente correlacionada con el número de copias del gen del ARNt.
- b) Calcular un índice de sesgo de codones. Esto mide la tasa de codones óptimos con respecto a los codones totales que codifican para ese mismo aminoácido. Sin embargo, en este caso el número de codones óptimos se normaliza con respecto al uso aleatorio esperado.  $CBI = (o_{opt} - e_{rand}) / (o_{tot} - e_{rand})$ . La limitación de este método es que requiere un conjunto de proteínas de referencia, como las proteínas ribosómicas altamente expresadas.
- c) Calcular un índice de adaptación de codones. Esto mide la adaptabilidad relativa o desviación del codón us- age de un gen hacia el uso de codones de un conjunto de proteínas de referencia, es decir, genes altamente expresados. Se define como la media geométrica de los valores de adaptabilidad relativa, medidos como pesos asociados a cada codón sobre la longitud de la secuencia génica (medidos en codones). Cada peso se calcula como la relación entre la frecuencia observada de un codón dado y la frecuencia de su aminoácido correspondiente. La limitación a este enfoque es que requiere la definición de un conjunto de proteínas de referencia, tal como lo hizo el último método.
- d) Calcular el número efectivo de codones. Esto mide el número total de codones diferentes utilizados en una secuencia, que mide el sesgo hacia el uso de un subconjunto más pequeño de codones, lejos del uso igual de codones sinónimos.  $N_c = 20$  si solo se usa un codón por aminoácido, y  $N_c = 61$  cuando todos los codones sinónimos posibles se usan por igual. Los pasos del proceso son calcular la homocigosidad para cada aminoácido estimada a partir de las frecuencias de codones al cuadrado, obtener el número efectivo de codones por aminoácido y calcular el número total de codones efectivos. Este método es ventajoso porque no requiere ningún conocimiento de emparejamiento ARNt-codón, y no requiere ningún conjunto de referencia. Sin embargo, está limitado en que no toma en cuenta el conjunto de ARNt.
- e) Calcular el índice de adaptación del ARNt. Supongamos que el número de copias del gen de ARNt tiene una alta correlación positiva con abundancia de ARNt dentro de la célula. Esto luego mide qué tan bien se adapta un gen al acervo de ARNt.

Es importante distinguir entre cuándo usar cada índice. La situación en la que un determinado índice es favorable está muy basada en el contexto, por lo que a menudo es preferible utilizar un índice por encima de todos los demás cuando la situación lo requiere. Al elegir cuidadosamente un índice, se puede descubrir información sobre la frecuencia con la que un codón se traduce en un aminoácido.

### Modificaciones de ARN

La historia se vuelve más complicada cuando consideramos modificaciones que pueden ocurrir al ARN. Por ejemplo, algunas modificaciones pueden expandir o restringir la capacidad de bamboleo del ARNt. Los ejemplos incluyen modificaciones de insosina y modificaciones de Xo5u. Estas modificaciones permiten a los ARNt decodificar un codón que antes no podían leer. Uno podría preguntarse por qué la modificación del ARN se seleccionó positivamente en el contexto de la evolución, y la razón es que esto permite aumentar la probabilidad de que exista un ARNt coincidente para decodificar un codón en un ambiente dado.

### Ejemplos de aplicaciones

Hay algunas aplicaciones naturales que resultan de nuestra comprensión de la evolución de codones.

- a) Optimización de codones para la expresión de proteínas heterólogas
- b) Predicción de regiones codificantes y no codificantes de un genoma
- c) Predicción de lectura de codones
- d) Comprensión de cómo se decodifican los genes: estudio de patrones de sesgo de uso de codones

## Regulación traslacional

Existen muchos medios conocidos de regulación a nivel postranscripcional. Estos incluyen la modulación de la disponibilidad de ARNt, cambios en el ARNm y elementos cis y trans-reguladores. Primero, la modulación de ARNt tiene un gran impacto. Cambios en los isoaceptores de ARNt, cambios en las modificaciones de ARNt y regulación en los niveles de aminoacilación de ARNt. Los cambios en el ARNm que afectan a la traducción incluyen cambios en la modificación del ARNm, la cola de poliA, el corte y empalme, el taponado y la localización del ARNm (importando y exportando desde el núcleo). Los elementos reguladores cis y trans incluyen interferencia de ARN (es decir, ARNip y miARN), eventos de cambio de marco y ribointerruptores. Además, ¡muchos elementos regulatorios aún están por descubrir!

---

This page titled [11.2: Regulación Postranscripcional](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **11.2: Post-Transcriptional Regulation** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 11.3: ¿Qué hemos aprendido?

Ojalá al final de este capítulo hayamos llegado a darnos cuenta de la importancia en la regulación transcripcional. Vemos que los niveles de ARNm no son 1:1 con niveles de proteína. Adicionalmente, vimos que el código genético no es universal, y lo que se considera pares ARNt-codón preferidos son dinámicos. Asimismo, las mutaciones sinónimas no son equivalentes entre especies. Hemos visto cuán poderosa es la técnica de perfilado de ribosomas, ya que nos permite medir la traducción con resolución de subcodones. A pesar de todo esto, es posible modelar la traducción y evoluciones de codones utilizando herramientas para ayudar a aumentar la eficiencia de la traducción/ plegamiento de proteínas en sistemas heterólogos, predecir regiones codificantes, comprender patrones de traducción específicos de tipo celular y comparar la traducción entre estados sanos y patológicos. Finalmente, al analizar la regulación traslacional, vemos cómo se afinan los niveles de proteínas, y vemos que hay muchas formas diferentes de lograr la regulación postranscripcional. Quizás podamos llegar a darnos cuenta de que hay más interconexión entre estas diferentes estrategias de regulación de lo que pensábamos originalmente.

This page titled 11.3: ¿Qué hemos aprendido? is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **11.3: What Have We Learned?** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 12: ARN intergénicos grandes no codificantes

12.1: Bibliografía

12.2: Introducción

12.3: ARN no codificantes de plantas a mamíferos

12.4: Tema práctico- RNaseQ

12.5: ARN largos no codificantes en la regulación epigenética

12.6: ARN intergergénicos no codificantes: ¿faltan lincs en células madre o cancerosas?

12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas?

---

This page titled [12: ARN intergénicos grandes no codificantes](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 12.1: Bibliografía

### Bibliografía

1. [1] R.P. Dilworth. Teorema de descomposición para conjuntos parcialmente ordenados. *Anal de Matemáticas*, 1950.
2. [2] Mitchell Guttman, Manuel Garber, Joshua Z Levin, Julie Donaghey, James Robinson, Xian Adiconis, Lin Fan, Magdalena J Koziol, Andreas Gnirke, Chad Nusbaum, y et al. La reconstrucción de ab initio de transcriptomas específicos de tipo celular en ratón revela la estructura multi-exónica conservada de lincrnas. *Nature Biotechnology*, 28 (5) :503—510, 2010.
- [3] C. Trapnell.

This page titled [12.1: Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.1: Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 12.2: Introducción

La epigenética es el estudio de cambios heredables en la expresión genética y fenotipo que no resultan de una secuencia de ADN. Cada célula, a pesar de tener una copia idéntica del genoma, es capaz de diferenciarse en un tipo especializado. Hay muchos dispositivos biológicos para lograrlos, incluida la metilación del ADN, la modificación de histonas y varios tipos de ARN.

La metilación del ADN es un código binario que efectivamente equivale a "encender" o "apagar" un gen. Sin embargo, muchas veces un gen podría necesitar ser más expresado en lugar de simplemente estar encendido. Para ello, las histonas tienen colas que están sujetas a modificación. La combinación única de estos dos elementos en un tramo de ADN puede pensarse como un código de barras para el tipo de célula. Aún más importante es el método de su preservación durante la replicación. En el caso de la metilación del ADN, se asigna una hebra apropiadamente metilada a cada célula madre o hija. Al dejar un rastro atrás, la célula es capaz de llenar los huecos y metilar adecuadamente a la otra célula.

Como intermediario entre las secuencias de ADN y las proteínas, el ARN es posiblemente el medio de regulación más versátil. Como tal, serán el foco de este capítulo.

### ¿Sabías?

Los tipos celulares se pueden determinar por modificación de histonas o metilación del ADN (un código binario, que se basa en un estado eucromático y heterocromático). Estas modificaciones de histonas pueden ser pensadas como un tipo de código de barras epigenético que permite escanear el ADN celular en busca de tipos. Los ARN no codificantes llamados ARN grandes no codificantes intergénicos (LincRNAs) están muy involucrados en este proceso.

Una historia rápida de ARN:

- 1975: Un laboratorio que prueba los niveles relativos de ARN y ADN en esperma de toro descubre el doble de ARN que ADN.
- 1987: Despues del desarrollo de la secuenciación automatizada, se encuentran por primera vez ARN raros no codificantes.
- 1988: Se ha demostrado que el ARN es importante para mantener las estructuras cromosómicas, a través de la arquitectura de
- 1990: Un gran número de experimentos comienzan a investigar
- Años 2000: Un estudio muestra que las histona-metiltransferasas dependen del ARN, ya que la ARNasa hace que las proteínas se deslocalicen.

La transcripción es un buen proxy de lo que está activo en la célula y lo que se convertirá en proteína. Las micromatrices condujeron al descubrimiento del doble de genes no codificantes que genes codificantes inicialmente; ahora sabemos que la proporción es incluso mucho mayor que esta.

This page titled [12.2: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.2: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 12.3: ARN no codificantes de plantas a mamíferos

Ciclo básico: ARN grande se corta en ARN pequeños (ARNIp) Uso de ARN por categoría:

Protistas: El ARN se usa como molde para empalmar el ADN (eliminación y corte y empalme del ADN dependiente del ARN)

ARNm y ADN en el núcleo: ADN troceado y recombinado basado en brechas en ARNm (“peculiar phenomena”)

Plantas: ARN polimerasa dependiente de ARN, donde la polimerasa toma molde de ARN y hace una copia del mismo, está disponible en plantas pero no en humanos, y puede producir ARN pequeños. Los mamíferos tienen como máximo un ejemplar. Muy diferente a la ARN polimerasa y ADN polimerasa en estructura. A partir de esto, sabemos que las plantas hacen metilación del ADN con ARN no codificante.

Moscas: usar ARN para un cambio de ARN; la regulación coordinada del gen hox requiere ARN no codificante. Mamíferos: Los ARN no codificantes pueden formar hélices triples, guiar proteínas hacia ellas; complejos modificadores de cromatina; involucrados en la línea germinal; guiar el comportamiento de los factores de transcripción.

Para el resto de esta charla, nos enfocamos específicamente en el LincRNA, que definiremos como ARN de más de 200 nucleótidos.

### RNAs largos no codificantes

Hay una serie de diferentes mecanismos y dispositivos biológicos por los cuales se produce la regulación epigenética. Uno de ellos son los ARN largos sin codificación que pueden considerarse como que cumplen una función de control de tráfico aéreo dentro de la célula.

Los ARN largos no codificantes comparten muchas características similares con los microARN. Están empalmados, contienen múltiples exones, están tapados y poli-adenuados. Sin embargo, no tienen marcos de lectura abiertos. Parecen genes codificadores de proteínas, pero no pueden.

Se clasifican mejor por su posición anatómica:

Antisentido: Estos están codificados en la cadena opuesta de un gen codificante de proteínas.

Intrónico: Totalmente contenido con un intrón de un gen codificante de proteínas.

Bidireccionales: Estos comparten el mismo promotor que un gen codificante de proteínas, pero están en el lado opuesto.

Intergénico: Estos no se superponen con ningún gen codificante de proteínas. Piense en ellos como sentados ciegamente a la intemperie. Son objetivos mucho más fáciles y serán el foco de este capítulo.

---

This page titled [12.3: ARN no codificantes de plantas a mamíferos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.3: Noncoding RNAs from Plants to Mammals](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

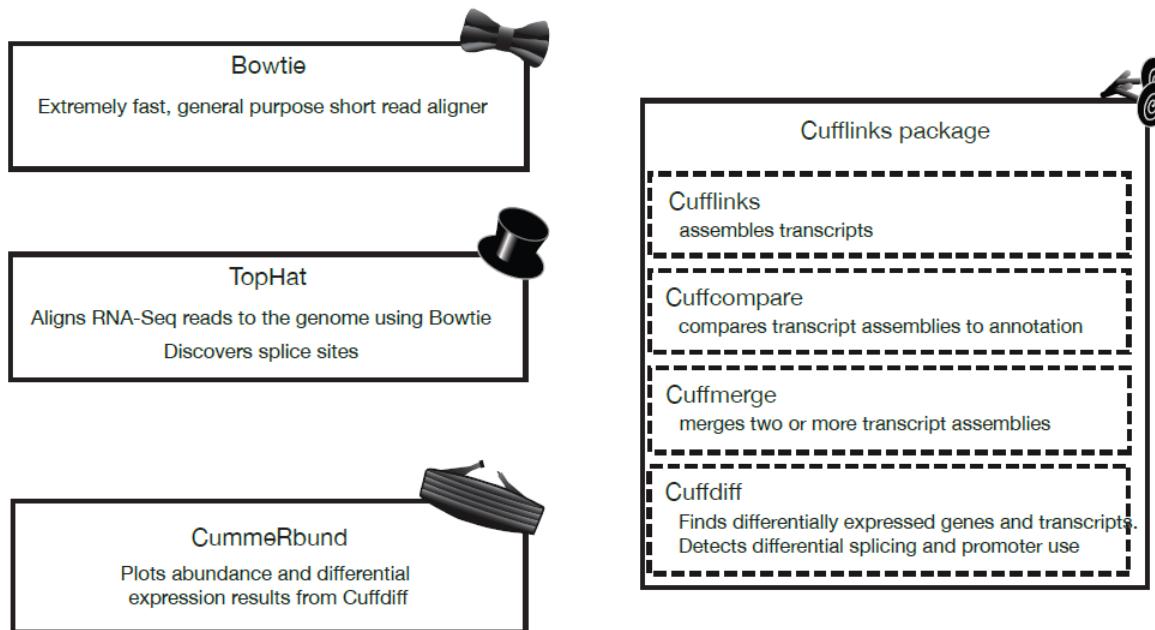
## 12.4: Tema práctico- RNaseQ

RNA-seq es un método que utiliza tecnología de secuenciación de próxima generación para secuenciar ADNc, lo que nos permite obtener información sobre el contenido del ARN. Los dos principales problemas que aborda ARN-seq son (1) descubrir nuevos genes como las isoformas de empalme de genes previamente descubiertos y (2) descubrir los niveles de expresión de genes y transcritos a partir de los datos de secuenciación. Además, RNA-seq también está comenzando a reemplazar muchas técnicas de secuenciación tradicionales, lo que permite a los laboratorios realizar experimentos de manera más eficiente.

### Cómo funciona

La máquina RNA-seq agarra un transcripto y lo rompe en diferentes fragmentos, donde los fragmentos se distribuyen normalmente. Con la velocidad con la que el RNA-seq puede secuenciar estos fragmentos de transcripción (o lecturas), hay un número abundante de lecturas que nos permiten extraer niveles de expresión. La idea básica detrás de este método se basa en el hecho de que cuanto más abundante sea una transcripción, más fragmentos secuenciaremos a partir de ella.

Las herramientas utilizadas para analizar los datos de RNA-seq se conocen colectivamente como las “Herramientas de Esmoquin”



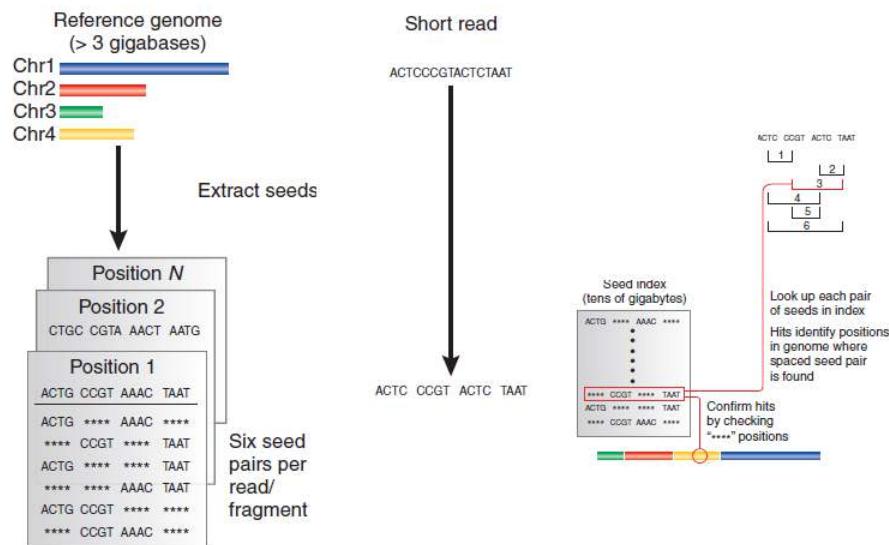
© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.1: Herramientas de esmoquin

### Alineación de lecturas de RNA-seq con genomas y transcriptomas

Dado que RNA-seq produce tantas lecturas, el algoritmo de alineación debe tener un tiempo de ejecución rápido, aproximadamente del orden de O (n). Existen dos estrategias principales para alinear lecturas cortas, las cuales requieren que ya tengamos las transcripciones.

1. Indización de semillas espaciadas

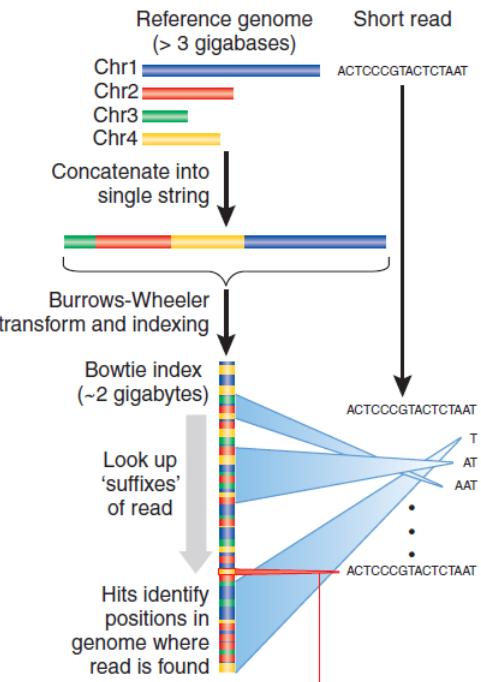


© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.2: Cómo funciona la indexación de semillas espaciadas

La indexación de semillas espaciadas implica tomar cada lectura y dividirla en fragmentos, o “semillas”. Tomamos cada combinación de dos fragmentos (“pares de semillas”) y los comparamos con un índice de semillas (que tomará decenas de gigabytes de espacio) para posibles aciertos. Compara las otras semillas con el índice para asegurarte de que tenemos un acierto.

## 2. Indexación de Madriñas-Wheeler



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.3: Cómo funciona la indexación de Burrows-Wheeler

La indexación de Burrows-Wheeler toma el genoma y lo revuelve de tal manera que puedes mirar el personaje leído a la vez y arrojar una gran parte del genoma como posibles posiciones de alineación muy rápidamente.

Un problema importante con estas dos estrategias de alineación de propósito general es que no tienen en cuenta grandes brechas en la alineación.

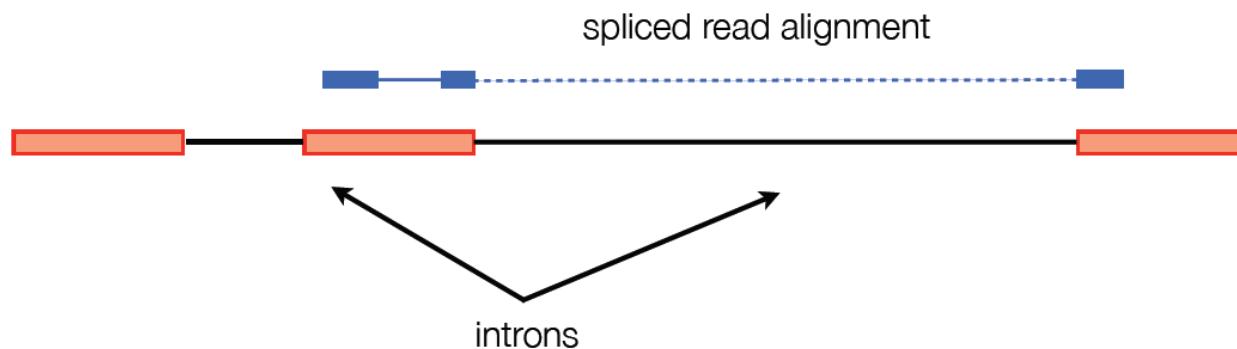


Figura 12.4: Un ejemplo de un hueco en alineación

Para sortear esto, TopHat rompe las lecturas en trozos más pequeños. Estas piezas están alineadas y las lecturas con piezas que se mapean muy separadas se marcan para posibles sitios de intrones. Las piezas que no pudieron alinearse se utilizan para confirmar los sitios de empalme. Luego, las lecturas se vuelven a unir para hacer alineaciones de lectura completas.

Existen dos estrategias para ensamblar transcripciones basadas en lecturas de RNA-seq.

#### 1. Enfoque guiado por genoma (utilizado en software como Gemelos)

La idea detrás de este enfoque es que no necesariamente sabemos si dos lecturas provienen de la misma transcripción, pero sabremos si provienen de diferentes transcripciones. El algoritmo es el siguiente: tomar las alineaciones y ponerlas en una gráfica. Agregue un borde de  $x \rightarrow y$  si  $x$  está a la izquierda de  $y$  en el genoma,  $x$  y  $y$  se superponen consistentemente, e y no está contenido en  $x$ . Entonces tenemos un borde de  $x \rightarrow y$  si pudieran provenir de la misma transcripción.

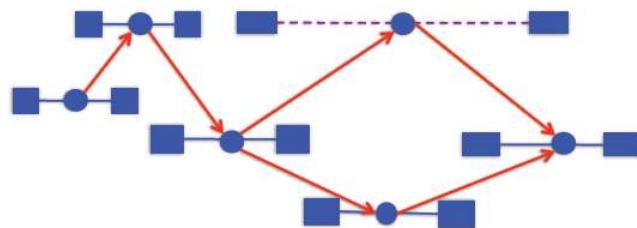


Figura 12.5: Un ejemplo de cómo usar la gráfica para encontrar transcripciones

Si cruzamos esta gráfica de izquierda a derecha, obtenemos una transcripción potencial. Aplicando el teorema de Dilworth para leer órdenes parciales, podemos ver que el tamaño de la anticadena más grande en la gráfica es el número mínimo de transcripciones necesarias para explicar la alineación. Una anticadena es un conjunto de alineaciones con la propiedad de que no hay dos compatibles (es decir, podrían surgir de la misma transcripción)

#### 2. Enfoque independiente del genoma (utilizado en software como trinity)

El enfoque independiente del genoma intenta juntar los transcritos directamente a partir de las lecturas usando métodos clásicos para el ensamblaje de lectura basado en superposición, similar a los métodos de ensamblaje del genoma.

## Cálculo de la expresión de genes y transcritos

Queremos contar el número de lecturas de cada transcripción para encontrar el nivel de expresión de la transcripción. Sin embargo, dado que dividimos las transcripciones en fragmentos del mismo tamaño, nos encontramos con el problema de que las transcripciones más largas producirán naturalmente más lecturas que una transcripción más corta. Para dar cuenta de esto, calculamos los niveles de expresión en FPKM, fragmentos por kilobase por millón de fragmentos mapeados.

Función de verosimilitud para un gen

Supongamos que secuenciamos una lectura particular, la llamamos F1.

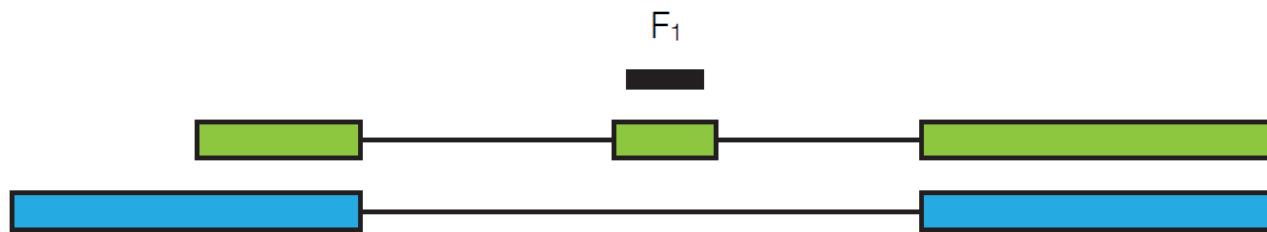


Figura 12.6: F1 204

Para obtener esta lectura en particular, necesitamos elegir la transcripción particular en la que se encuentra y luego tenemos que elegir esta lectura en particular de toda la transcripción. Si  $\gamma_{\text{green}}$  definimos que es la abundancia relativa de la transcripción verde, entonces tenemos

$$P(F_1 | \gamma_{\text{green}}) = \frac{\gamma_{\text{green}}}{l_{\text{green}}}$$

donde  $l_{\text{verde}}$  es la longitud de la transcripción verde. Ahora supongamos que miramos una lectura diferente, F2.

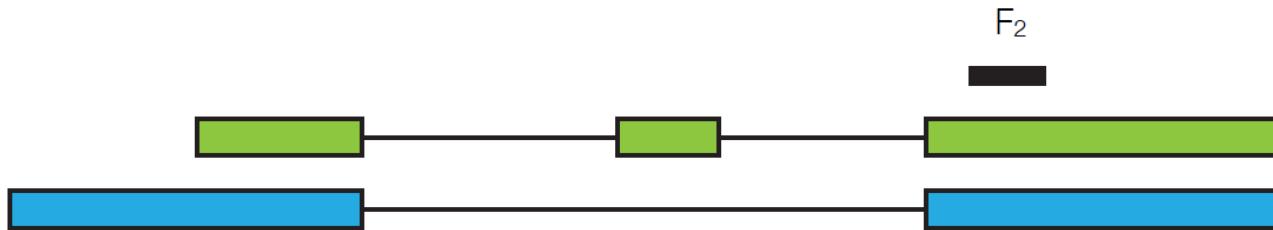


Figura 12.7: F2

Podría haber venido ya sea de la transcripción verde de la transcripción azul, entonces:

$$P(F_2 | \gamma) = \frac{\gamma_{\text{green}}}{l_{\text{green}}} + \frac{\gamma_{\text{blue}}}{l_{\text{blue}}}$$

Podemos ver que la probabilidad de obtener tanto F1 como F2 es solo el producto de las probabilidades individuales:

$$P(F | \gamma) = \frac{\gamma_{\text{green}}}{l_{\text{green}}} \cdot \left( \frac{\gamma_{\text{green}}}{l_{\text{green}}} + \frac{\gamma_{\text{blue}}}{l_{\text{blue}}} \right)$$

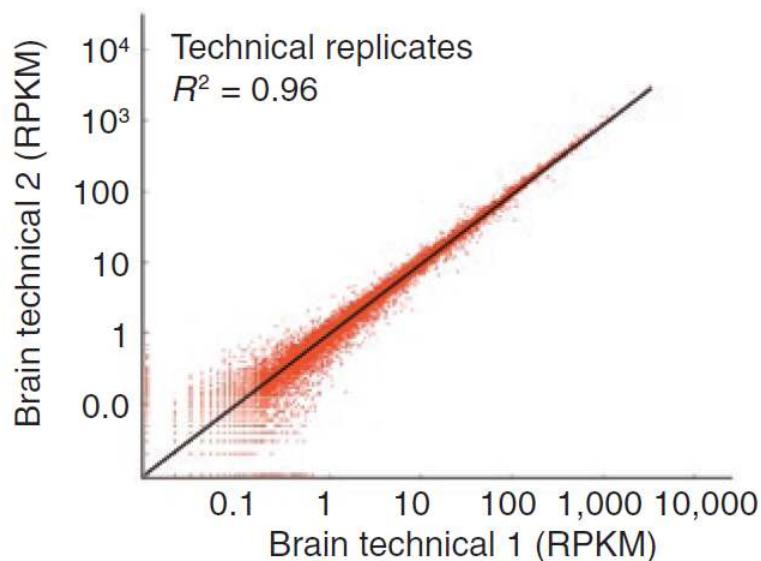
Definimos esto como nuestra función de verosimilitud,  $L(F|\gamma)$ . Dada una entrada de abundancias, obtenemos una probabilidad de cuán probable es nuestra secuencia de lecturas. Entonces, a partir de un conjunto de lecturas y transcripciones, podemos construir una función de verosimilitud y calcular los valores para gamma que maximizarán esta función. Gemelos logra esto usando escalada en colina o EM en la función de verosimilitud logarítmica.

## Análisis diferencial con RNA-seq

Supongamos que realizamos un análisis de RNA-seq para un gen bajo dos condiciones diferentes. ¿Cómo podemos saber si hay una diferencia significativa en los recuentos de fragmentos? Calculamos la expresión estimando el número esperado de fragmentos que provienen de cada transcripción. Para probar la significancia, necesitamos conocer la varianza de esa estimación. Modelamos la varianza como:

$$\text{Var}(\text{expresión}) = \text{Variabilidad técnica} + \text{Variabilidad biológica}$$

La variabilidad técnica, que es la variabilidad de la incertidumbre en las lecturas de mapeo, se puede modelar bien con una distribución de Poisson (ver figura a continuación). Sin embargo, el uso de Poisson para modelar la variabilidad biológica, o variabilidad entre repeticiones, da como resultado una sobredispersión.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.8: La variabilidad técnica sigue una distribución de Poisson

En el caso simple donde tenemos variabilidad entre réplicas, pero sin incertidumbre, podemos mezclar las distribuciones de Poisson de cada réplica en una nueva distribución para modelar la variabilidad biológica. Podemos tratar el parámetro lambda de la distribución de Poisson como una variable aleatoria que sigue una distribución gamma:

$$X \sim \text{Poisson}(\Gamma(r, p))$$

Los recuentos de este modelo siguen una distribución binomial negativa. Para determinar los parámetros para el binomio negativo para cada gen, podemos ajustar una función gamma a través de un diagrama de dispersión de la varianza de conteo promedio vs recuento entre réplicas.

En el caso simple donde hay incertidumbre de mapeo leído, pero no variabilidad biológica, necesitamos incluir la incertidumbre de mapeo en nuestra estimación de varianza. Dado que asignamos lecturas a transcripciones probabilísticamente, necesitamos calcular la varianza en esa asignación.

Los dos hilos de la investigación de análisis de expresión ARN-seq se enfocan en los problemas en estos dos casos simples. Uno de los hilos se enfoca en inferir la abundancia de isoformas individuales para aprender sobre el corte y empalme diferencial y el uso de promotores, mientras que el otro hilo se enfoca en modelar la variabilidad a través de réplicas para crear un análisis de expresión génica diferencial más robusto. Cuffdiff une estos dos hilos separados para estudiar el caso donde tenemos variabilidad biológica y ambigüedad de mapeo de lectura. Dado que la sobredispersión se puede modelar con una distribución binomial negativa y la

incertidumbre de mapeo se puede modelar con una distribución Beta, combinamos estas dos para modelar este caso con una distribución binomial beta negativa.

---

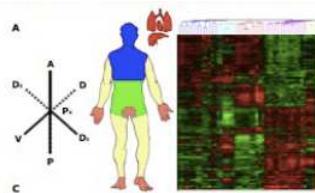
This page titled [12.4: Tema práctico- RNaseQ](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.4: Practical topic- RNAseq](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 12.5: ARN largos no codificantes en la regulación epigenética

Examinemos la piel humana como un ejemplo de ARN largos no codificantes que se utilizan en la regulación epigenética. La piel humana es enorme, de hecho es el órgano más grande en peso del cuerpo. Es intrincado, con características especializadas, y se está regenerando constantemente para reemplazar las viejas células muertas por otras nuevas. La piel debe ser controlada para que el cabello solo crezca en el dorso de la mano en lugar de en la palma de la mano. Además, estos límites no pueden cambiar y se mantienen desde su nacimiento.

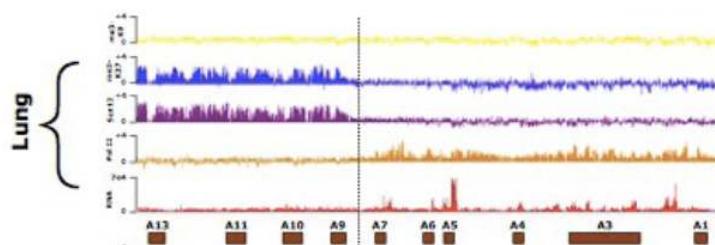
La piel en todas las partes del cuerpo está compuesta por una capa epitelial y una capa de tejido conectivo compuesta por células llamadas fibroblastos. Estos fibroblastos secretan señales de citocinas que controlan la capa externa, determinando propiedades como la presencia o ausencia de cabello. Los fibroblastos alrededor del cuerpo son idénticos excepto por el plegamiento epigenético específico que dicta qué tipo de piel se formará en una ubicación determinada. Con base en si la piel es distal o proximal, interior o exterior, posterior o anterior, un conjunto diferente de pliegues epigenéticos determinará el tipo de piel que se forma.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.9: Los fibroblastos humanos se especializan a través de la regulación epigenética para formar diferentes tipos de piel en función de su ubicación dentro del cuerpo. La investigación ha encontrado que el tipo de piel en las manos comparte una firma epigenética notablemente similar a la piel de los pies, que también se localiza distalmente.

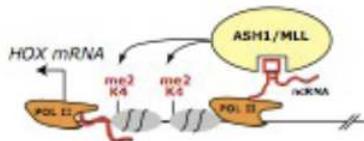
Se ha encontrado que genes HOX específicos delinean estos límites anatómicos durante el desarrollo. Con solo mirar el código genético HOX humano, se puede predecir dónde se ubicará una célula. Mediante el uso de ChIP- on-chip (microarrays de inmunoprecipitación de cromatina) se han encontrado dominios diamétricos de cromatina entre estos genes HOX. En la siguiente figura, podemos ver un claro límite entre los dominios de cromatina de un tipo celular localizado proximalmente y otro localizado distalmente. Este límite no sólo es preciso, sino que se mantiene a través de billones de células de la piel.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.10: Se analizan dos tipos de células cutáneas para determinar sus dominios de cromatina. Existe un claro límite entre el tipo de célula pulmonar que es proximal al cuerpo, y el tipo de célula pie que es distal al cuerpo.

Se ha investigado el ARN intergénico antisentido del transcripto HOTAIR o HOX como posible regulador de ARN que mantiene estos límites entre los dominios diamétricos en la cromatina. Cuando HOTAIR fue knockoutado en el locus HOXC, se planteó la hipótesis de que los dominios de la cromatina podrían deslizarse entre sí. Si bien se encontró que este HOTAIR no afectó directamente el límite epigenético, los investigadores sí encontraron evidencia de charla cruzada genómica basada en ARN. El gen HOTAIR afectó a un locus diferente llamado HOXD.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 12.11: Polycomb, una proteína que puede remodelar la cromatina para que pueda tener lugar el silenciamiento epigenético de genes, puede ser regulada por ARN no codificante como HOTAIR.

A través de un proceso de represión Polycomb dependiente de ARNnc, la secuencia HOTAIR puede controlar la regulación epigenética. Polycomb es una proteína que pone marcas de tope en las colas de las histonas para que puedan provocar pliegues específicos en el material genético. En sus propias histonas, son no dirigidas, por lo que es necesario que algún mecanismo dicte cómo se adhieren al genoma. Este proceso de descubrimiento ha llevado a un gran interés en el poder de los ARN no codificantes intergénicos largos para afectar la regulación epigenética.

---

This page titled [12.5: ARN largos no codificantes en la regulación epigenética](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.5: Long non-coding RNAs in Epigenetic Regulation](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 12.6: ARN integergénicos no codificantes: ¿faltan lincs en células madre o cancerosas?

### ejemplo: XIST

XIST fue uno de los primeros LINC RNA caracterizados. Participa directamente en la desactivación de uno de los cromosomas X femeninos durante el desarrollo embrionario. Se ha descrito como tener la capacidad de “arrugar un cromosoma entero”. Esto es importante porque la desactivación impide la sobreexpresión letal de los genes que se encuentran en el cromosoma X.

El ARN es importante para conseguir complejo policromado a cromosomas Los ARNc pueden activar genes aguas abajo en Cis, opuesto en trans; Xist hace lo mismo.

This page titled [12.6: ARN integergénicos no codificantes: ¿faltan lincs en células madre o cancerosas?](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **12.6: Integergenic Non-coding RNAs- missing lincs in Stem/Cancer cells?** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas?

¿Cómo encontraríamos los ARNnc? Tenemos alrededor de 20-30 ejemplos de ARNcn con evidencia de importancia, pero hay más por ahí. Los mapas de estado de la cromatina (de ENCODE, chip-seq) se pueden utilizar para encontrar unidades transcripcionales que no se superpongan a las proteínas. Podemos caminar a lo largo del mapa y buscar genes (mirar a ojo el mapa de la cromatina para encontrar los ARNcns). Casi el 90% de las veces que se encuentre una firma de este tipo, se transcribirá ARN a partir de ella. Podemos validar esto a través de Northern Blot

Al mirar un mapa de cromatina para encontrar NCRNAs, esencialmente estamos mirando a través del mapa con una ventana de un tamaño dado y viendo cuánta señal vs. ruido estamos recibiendo, en comparación con lo que podríamos esperar de una hipótesis de azar aleatorio. Como tanto las ventanas grandes como las pequeñas tienen beneficios, ambas deben usarse en cada sección del mapa. Las ventanas más grandes encapulan más información; las ventanas más pequeñas son más sensibles.

Después de encontrar regiones integénicas, encontramos regiones conservadas.

Comprobamos si las nuevas regiones están bajo presión selectiva; menos mutaciones en regiones conservadas. Si un nucleótido nunca tiene una mutación entre especies, está altamente conservado.

Los ARNs de LINC están más conservados que los intrones, pero menos conservados que los intrones que codifican proteínas, posiblemente debido a secuencias no conservadas en las regiones de bucle de los LINC RNA.

Encontrar cuáles son las funciones de los lincRNA: “Culpabilidad por asociación”: Podemos encontrar proteínas que se correlacionan con el lincRNA particular en términos de expresión; los LINC RNA probablemente se correlacionen con una vía particular. De esta manera, adquirimos un código de barras multidimensional para cada LincRNA (lo que es y con lo que no está relacionado). Podemos agrupar las firmas de lincRNA e identificar patrones comunes. Mucho tiene que ver con los genes del ciclo celular. (Este enfoque funciona 60-70% del tiempo)

Como la mayoría de los LINC RNA tienen más de 3000 bases, muchos contienen secuencias para marcos de lectura abiertos de 100 aminoácidos, simplemente por casualidad. Esto da como resultado muchos falsos negativos durante la detección.

Se ha encontrado que muchos LINC RNA tienden a vecinos de las regiones del desarrollo del genoma. También tienden a ser de baja expresión en comparación con los genes codificantes de proteínas.

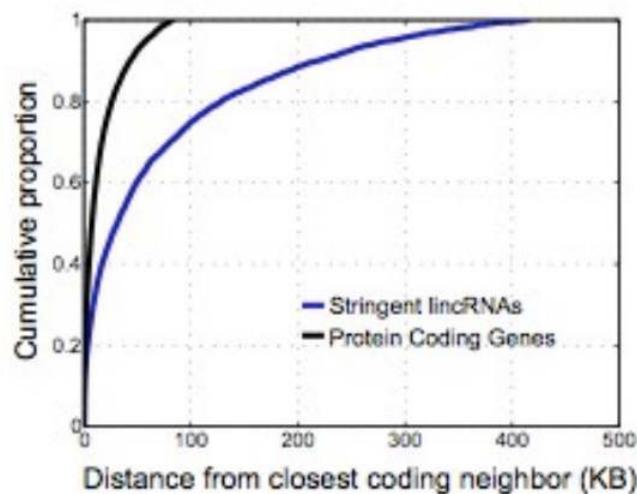


Figura 12.12: Reguladores del desarrollo vecinos de los LINC RNA

© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

### Ejemplo: p53

Validación independiente: se utilizan modelos animales, donde uno es un p53 de tipo silvestre, y uno es un knockout. Inducimos p53, luego preguntamos si los LINC RNA se encienden. 32 de los 39 LINC RNA encontrados asociados con p53 fueron inducidos

temporalmente al encender p53.

Un ARN en particular se sentó junto a un gen codificante de proteínas en la vía p53. Intentamos averiguar si p53 se unía al promotor y lo encendimos. Para ello, clonamos el promotor de lincRNA, y preguntamos ¿p53 lo enciende? Se realizó una prueba de detección de la proteína p53, para ver si se asoció con el LincRNA del promotor. Resultó que el lincRNA está directamente relacionado con p53 - p53 lo enciende. P53 también desactiva los genes; ciertos LINC RNA actúan como represor.

A partir de este ejemplo (y otros), empezamos a ver que los ARN suelen tener un compañero proteico

El ARN puede unir innumerables proteínas diferentes, permitiendo que las células tengan mucha diversidad. De esta manera es similar a la fosforilación. Los ARN se unen a importantes complejos de cromatina y se requieren para reprogramar las células de la piel en células madre.

---

This page titled [12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas?](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [12.7: Technologies- in the wet lab, how can we find these?](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 13: ARN pequeño

- 13.1: Introducción
- 13.2: Interferencia de ARN
- 13.3: Bibliografía

---

This page titled [13: ARN pequeño](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 13.1: Introducción

Los análisis a gran escala en la década de 1990 utilizando etiquetas de secuencia expresadas han estimado un total de 35.000 a 100,000 genes codificados por el genoma humano. Sin embargo, la secuenciación completa del genoma humano ha revelado sorprendentemente que es probable que el número de genes que codifican proteínas sea ~20,000 — 25,000 [12]. Si bien esto representa < 2% of the total genome sequence, whole genome and transcriptome sequencing and tiling resolution genomic microarrays suggests that over > 90% del genoma todavía se transcribe activamente [8], en gran parte como ARN no codificantes de proteínas (ARNnc). Aunque la especulación inicial ha sido que estos son ruido transcripcional no funcional inherente a la maquinaria de transcripción, ha habido evidencia creciente que sugiere el importante papel que estos ARNnc juegan en los procesos celulares y la manifestación/progresión de enfermedades. Por lo tanto, estos hallazgos desafiaron la visión canónica del ARN que sirve solo como el intermedio entre el ADN y la proteína.

### Clasificaciones de ARNcr

El creciente enfoque en los últimos años en los últimos años junto con los avances en las tecnologías de secuenciación (es decir, Roche 454, Illumina/Solexa y SoLiD; consulte [16] para obtener más detalles sobre estos métodos) ha llevado a una explosión en la identificación de diversos grupos de ARNnc. Aunque aún no ha habido una nomenclatura consistente, los ARNnc pueden agruparse en dos clases principales en función del tamaño del transcrito: los ARNnc pequeños (<200 nucleótidos) y los ARNnc largos (LncRNAs) ( $\geq 200$  nucleótidos) (Cuadro 13.1) [6, 8, 13, 20, 24]. Entre estos, el papel de los ARNnc pequeños, el microARN (miARN) y el ARN interferente pequeño (ARNip) en el silenciamiento del ARN han sido los más documentados en la historia reciente. Como tal, gran parte de la discusión en el resto de este capítulo se centrará en los roles de estos pequeños ARNnc. Pero primero, describiremos brevemente el otro conjunto diverso de ARNcr.

Cuadro 13.1: Clasificaciones de ARNcr (basadas en [6, 8, 13, 20, 24])

Nombre	Abreviatura	Función
ARN de transferencia de ARN ribosómico ARN nucleolar pequeño ARN específico de cuerpo cajal pequeño ARN nuclear pequeño ARN guía	<i>RNAs de limpieza</i> ARNr ARNt ARNp ARNp (~60-220 nt) ARNsi ARNs (~60-300 nt) ARNg	traducción traducción modificación de ARNr modificación de empalme empalme de ARN edición de ARN
MicroARN ARN interferente pequeño ARN que interactúa con Piwi ARN de iniciación de la transcripción diminuto ARN corto asociado al promotor ARN antisentido del sitio de inicio de la transcripción ARN corto asociado a termini ARN corto asociado a terminales antisentido ARN derivado de retrotransposón ARN derivado de 3'UTR X-ncRNA RNA pequeño asociado a NF90 RNA inusualmente pequeño RNA de bóveda RNA Y humano	<i>ARNcr pequeños (&lt;200 nt)</i> miARN (~19-24 nt) ARNip (~21-22 nt) PIRNA (~26-31 nt) TirNA (~17-18 nt) PASR (~22-200 nt) TsSa-ARN (~20-90 nt) TASR AtasR RE-RNA UARna X-NCrna SnAr VtRNA Hy ARN	Silenciamiento de ARN Silenciamiento de ARN Silenciamiento de transposones, regulación epigenética ¿Regulación transcripcional? desconocido ¿Mantenimiento transcripcional? no claro
ARNcr intergénico grande Regiones ultraconservadas transcritas Pseudogenes Transcritos cadena arriba del promotor ARN que contiene una repetición telomérica Repetición GAA-que contiene ARN Potenciador ARN ARNc intrónico largo ARN antisentido ARN largo asociado al promotor ARN de intrón escindido estable ARN de intrón escindido largo No inducido por estrés largo transcripciones de codificación	<i>ARNcr largos (&gt;200 nt)</i> T-UCR ninguno PROMPT TERRA GRC-RNA ERna ninguno ARNa PALR ninguno LSINCT	Regulación epigenética ¿Regulación de miARN? ¿Regulación de miARN? ¿Activación transcripcional? heterocromatina telomérica principal- tenance no claro

## NcRNA pequeño

Durante las últimas décadas, ha habido una serie de especies pequeñas de ARN no codificantes bien estudiadas. Todas estas especies están involucradas en la traducción del ARN (ARN de transferencia (ARNt)) o en la modificación y procesamiento del ARN (ARN nucleolar pequeño (SNORNA) y ARN nuclear pequeño (ARNsn)). En particular, los SNORNA (agrupados en dos clases amplias: C/D Box y H/ACA Box, involucrados en la metilación y pseudouridilación, respectivamente) se localizan en el núcleo y participa en el procesamiento y modificación del ARNr. Otro grupo de ARNnc pequeños son los ARNsn que interactúan con otras proteínas y entre sí para formar empalmes para el corte y empalme de ARN. Sorprendentemente, estos ARNsn son

modificados (metilación y pseudouridilación) por otro conjunto de ARNnc pequeños, los ARN pequeños específicos del cuerpo Cajal (sCARNA), que son similares al ARNnoP (en secuencia, estructura y función) y se localizan en el cuerpo Cajal en el núcleo. Sin embargo, en otra clase de ARNnc pequeños, se ha demostrado que los ARN guía (ARNg) predominantemente en los tripanosomátidos están involucrados en la edición de ARN. También se han propuesto recientemente muchas otras clases (véase el Cuadro 13.1) aunque sus roles funcionales aún están por determinar. Quizás los ARNnc más estudiados en los últimos años son los microRNAs (miRNAs), involucrados en el silenciamiento génico y responsables de la regulación de más del 60% de genes codificadores de proteínas [6]. Dado el extenso trabajo que se ha centrado en la iARN y la amplia gama de aplicaciones basadas en ARNi que han surgido en los últimos años, la siguiente sección (Interferencia de ARN) estará enteramente dedicada a este tema.

## NcRNA largo

Los ARNcr largos (lncRNAs) constituyen la porción más grande de los ARNcr [6]. Sin embargo, el énfasis puesto en el estudio del ARNnc largo solo se ha realizado en los últimos años. Como resultado, la terminología para esta familia de ARNnc aún se encuentra en su infancia y a menudo es inconsistente en la literatura. Esto también se complica en parte por los casos en los que algunos lncRNAs también pueden servir como transcritos para la generación de ARN cortos. A la luz de estas confusiones, como se discutió en el capítulo anterior, los ARNnc se han definido arbitrariamente como ARNnc con un tamaño mayor a 200 nts (basado en el corte en los protocolos de purificación de ARN) y pueden clasificarse ampliamente en: sentido, antisentido, bidireccional, intrónico o intergénico [19]. Por ejemplo, una clase particular de ARNnc llamada ncRNA intergénico largo (lincRNA) se encuentra exclusivamente en la región intergénica y posee modificaciones de cromatina indicativas de transcripción activa (por ejemplo, H3K4me3 en el sitio de inicio de la transcripción y H3K36me3 en toda la región del gen) [8].

A pesar del reciente aumento del interés en los lncRNAs, el descubrimiento de los primeros lncRNAs (XIST y H19), basado en la búsqueda de bibliotecas de cDNA, se remonta a las décadas de 1980 y 1990 antes del descubrimiento de miRNAs [3, 4]. Estudios posteriores demostraron la asociación de lncRNAs con proteínas del grupo polycomb, sugiriendo roles potenciales de lncRNAs en el silenciamiento/activación de genes epigenéticos [19]. Recientemente se encontró que otro lncRNA, HOXA Antisense Intergenic RNA (HOTAIR), está altamente sobreexpresado en tumores de mama metastásicos [11]. La asociación de HOTAIR con el complejo polycomb nuevamente respalda un papel unificado potencial de los lncRNAs en la remodelación de la cromatina/regulación epigenética (ya sea en forma cis-reguladora (XIST y H19), o trans-reguladora (por ejemplo, HOTAIR)) y etiología de enfermedades.

Estudios recientes también han identificado HULC y pseudogen (transcripción que se asemeja a genes reales pero contiene mutaciones que impiden su traducción en proteínas funcionales) PTENP1 que puede funcionar como un señuelo en la unión a miARN para reducir la efectividad general de los miARN [18, 25]. Otros roles potenciales de los lncRNAs aún no se han explorado. Sin embargo, es cada vez más claro que los lncRNAs tienen menos probabilidades de ser el resultado del ruido transcripcional, sino que pueden desempeñar un papel crítico en el control de los procesos celulares.

---

This page titled [13.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [13.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 13.2: Interferencia de ARN

La interferencia de ARN ha sido uno de los descubrimientos más significativos y emocionantes de la historia reciente. El impacto de este descubrimiento es enorme con aplicaciones que van desde estudios de derribo y pérdida de función hasta la generación de mejores modelos animales con caída condicional de gen (s) deseado (s) a gran escala, cribas basadas en ARNi para ayudar al descubrimiento de fármacos.

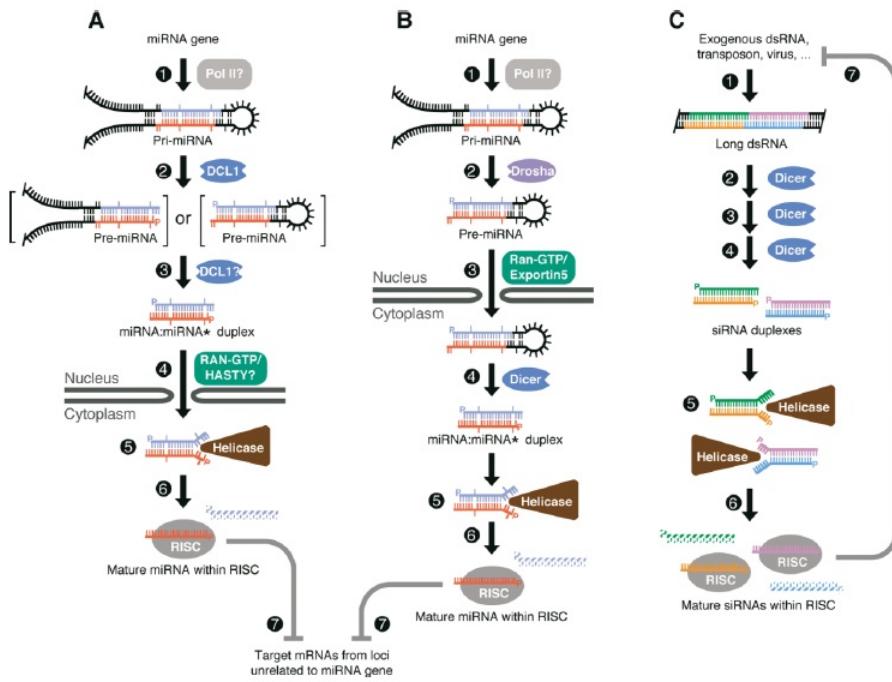
### Historia del descubrimiento

El descubrimiento del fenómeno de silenciamiento génico se remonta a la década de 1990 con Napoli y Jorgensen demostrando la regulación a la baja de la chalcona sintasa tras la introducción del transgén exógeno en plantas [17]. Posteriormente se observó una supresión similar en otros sistemas [10, 22]. En otro conjunto de trabajos no relacionados en ese momento, Lee et al. identificaron en una pantalla genética que lin-4 endógena expresaba un producto no codificante de proteínas que es complementario al gen lin-14 y controlaban el momento del desarrollo larval (del primer al segundo estado larval) en *C. elegans* [15]. Ahora lo conocemos como el primer miARN que se descubre. En 2000, se descubrió otro miARN, let-7, en el mismo organismo y se encontró que estaba involucrado en la promoción de la transición larval tardía a adulta [21]. El trabajo seminal de Mello y Fire en 1998 (por el cual fue galardonado con el Premio Nobel en 2006) demostró que la introducción del ARNb exógeno en *C. elegans* silenció específicamente genes mediante interferencia de ARN, explicando el fenómeno de supresión previa observado en plantas [7]. Estudios posteriores encontraron la conversión de ARNb en ARNip en la vía de ARNi. En 2001, el término miARN y el vínculo entre miARN y ARNi se describió en tres artículos en *Science* [23]. Con esto, nos hemos dado cuenta de que la maquinaria reguladora génica estaba compuesta predominantemente por dos clases de ARN pequeños, con miARN involucrado en la regulación de genes endógenos y ARNip involucrado en defensa en respuesta a ácidos nucleicos virales, transposones y transgenes [5]. Trabajos posteriores revelaron efectores aguas abajo: Dicers (para la escisión de especies precursoras) y proteínas Argonauta (parte del complejo silenciador inducido por ARN para realizar los efectos silenciadores reales), completando nuestra comprensión actual de las vías de silenciamiento del ARN. Los detalles del mecanismo y las diferencias entre las especies se discuten más adelante.

### Vías de biogénesis

Hay un tema común involucrado tanto para el silenciamiento mediado por ARNip como para el miARN. En la biogénesis tanto de ARNip como de miARN, los precursores bicatenarios se escinden por una RNasa en fragmentos cortos de ~22 nt. Una de las cadenas (la cadena guía) se carga en una proteína Argonauta, un componente central del complejo de ribonucleoprotien más grande RISC que facilita el reconocimiento y silenciamiento del ARN diana. El mecanismo de silenciamiento es la escisión del ARNm diana o la represión de la traducción.

A parte de este tema común, las proteínas involucradas en estos procesos difieren entre especies y existen etapas adicionales en el procesamiento del miARN previo a su maduración e incorporación al RISC (Figura 13.1). Para la biogénesis del ARNip, los precursores son los dsRNAs, muchas veces de fuentes exógenas como virus o transposones. Sin embargo, estudios recientes también han encontrado ARNip endógenos [9]. Independientemente de la fuente, estos ARNb son procesados por la endonucleasa RNasa III, Dicer, en ARNs de ~22 nt. Esta escisión catalizada por RNasa III deja los 5'fosfatos característicos y los voladizos 3' de 2 nt [2]. Cabe destacar que diferentes especies han evolucionado con diferente número de parálogos. Esto se vuelve importante ya que, para ser discutido más adelante, la vía de biogénesis de miARN también utiliza Dicer para el procesamiento de precursores de miARN (más específicamente pre-miARN). Para especies como *D. melanogaster*, hay dos proteínas Dicer distintas y como resultado normalmente hay un procesamiento preferencial de los precursores (por ejemplo, Dicer-1 para escisión de miARN y Dicer-2 para escisión de ARNip) [5]. En contraste, los mamíferos y nematodos solo tienen una única proteína Dicer y como tal ambas vías de biogénesis convergen a la misma etapa de procesamiento [5]. En etapas posteriores de la ruta de biogénesis de ARNip, una de las cadenas en el dúplex de ARNip se carga en RISC para silenciar los ARN diana (Figura 13.1C).



Cortesía de Elsevier, Inc., <http://www.sciencedirect.com>. Usado con permiso. Fuente: Bartel, David P. "MicroARN: Genómica, Biogénesis, Mecanismo y Función". Célula 116, núm. 2 (2004): 281-97.

Figura 13.1: Vías de biogénesis de ARNip y miARN. (A) Biogénesis de miARN vegetal (B) Biogénesis de miARN animal (C) Biogénesis de ARNip animal. Adoptada de Bartel, 2004 (ref [2]). Copyright © 2004 Cell Press.

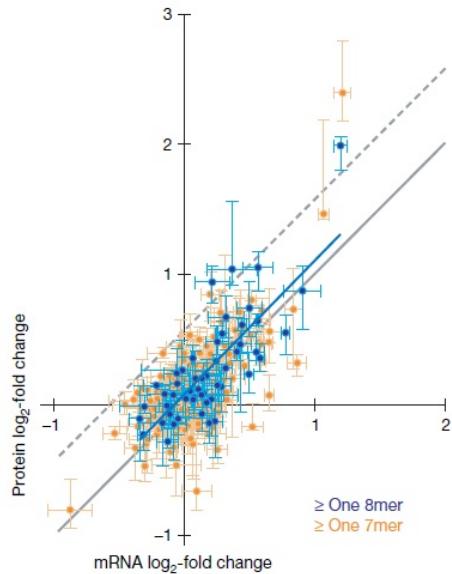
En la vía de biogénesis de miARN, la mayoría de los precursores son transcritos pol II de las regiones intrónicas, algunas de las cuales codifican múltiples miARN en grupos. Estos precursores, en forma de estructura tallo-bucle, se llaman pri-miARN. Los pri-miRNAs se escinden primero en el núcleo por una endonucleasa RNasa III (Drosha en animales y Dcl1 en plantas) en intermedios de tallo bucle de ~60-70 nt, denominados pre-miRNAs [2]. En los animales, el pre-miARN es luego exportado al citoplasma por Exportin-5. Esto es seguido por la escisión del intermedio pre-miARN por Dicer para eliminar el tallo-asa. Una de las cadenas en el dúplex de miARN maduro resultante se carga en RISC, similar a la descrita para la biogénesis de ARNip Figura 13.1B. Curiosamente, en las plantas, el pri-miARN se procesa en miARN maduro a través de dos escisiones por la misma enzima, Dcl1, en el núcleo antes de su exportación al citoplasma para su carga (Figura 13.1A).

### Funciones y mecanismo de silenciamiento

La visión clásica de la función de miARN basada en los primeros descubrimientos de miARN ha sido análoga a un cambio binario mediante el cual miARN reprime la traducción de algunas dianas de ARNm clave para iniciar una transición de desarrollo. Sin embargo, estudios posteriores han ampliado enormemente esta definición. En las plantas, la mayoría de los miARN se unen a la región codificante del ARNm con complementariedad casi perfecta. Por otro lado, los miARN animales se unen con complementariedad parcial (excepto por una región semilla, residuos 2-8) a las regiones 3' UTR del ARNm. Como tal, hay potencialmente cientos de dianas por un solo miARN en animales en lugar de solo unos pocos [1]. Además, en los mamíferos, solo una parte de las dianas predichas están involucradas en el desarrollo, y el resto se predice que cubrirá una amplia gama de procesos moleculares y biológicos [2]. Por último, el silenciamiento de miARN actúa a través de la represión de la traducción y la escisión del ARNm (y también la desestabilización como se analiza a continuación) (como se muestra por ejemplo por Bartel y sus compañeros de trabajo en la escisión dirigida por miR-96 de HOXB6 [26]). En conjunto, la visión moderna de la función de miARN ha sido que el miARN amortigua la expresión de muchas dianas de ARNm para optimizar la expresión, reforzar la identidad celular y agudizar las transiciones.

El mecanismo para el cual miARN media el silenciamiento del ARNm diana sigue siendo un área de investigación activa. Como se discutió anteriormente, el silenciamiento del ARN puede tomar la forma de escisión, desestabilización (que conduce a la posterior degradación del ARNm) o represión de la traducción. En las plantas, se ha encontrado que el modo predominante de silenciamiento

del ARN es a través de la escisión catalizada por Argonautas. Sin embargo, la contribución de estos diferentes modos de silenciamiento ha sido menos clara en los animales. Análisis globales recientes del grupo Bartel en colaboración con Gygi e Ingolia y Weissman arrojan luz sobre esta cuestión. En un estudio de 2008, los grupos Bartel y Gygi examinaron los cambios globales en el nivel de proteínas mediante espectrometría de masas después de la introducción o deleción de miARN [1]. Sus resultados revelaron la represión de cientos de genes por miARN individuales y, lo que es más importante, la desestabilización del ARNm representa la mayoría de las dianas altamente reprimidas (Figura 13.2).



Cortesía de Macmillan Publishers Limited. Usado con permiso. Fuente: Baek, Daehyun, et al. "El impacto de los microARN en la producción de proteínas". *Naturaleza* 455, núm. 7209 (2008): 64-71.

Figura 13.2: Cambios en proteínas y ARNm tras la pérdida de miR-223, a partir de mensajes con al menos un sitio 3'UTR de 8 meros (azul) o al menos un 7-mero (naranja). Adoptada de Baek et al., 2008 (ref [1]). Copyright © 2008 Macmillan Publishers Limited.

Esto está respaldado además por un estudio posterior utilizando tanto RNA-seq como un nuevo perfil de ribosomas demostrado por primera vez por Ingolia y Weissman 2009 que permite interrogar las actividades de traducción global con resolución de subcodones [14]. Los resultados mostraron que la desestabilización del ARNm diana es el mecanismo predominante a través del cual el miARN reduce la producción de proteínas.

---

This page titled [13.2: Interferencia de ARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [13.2: RNA Interference](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 13.3: Bibliografía

### Bibliografía

- [1] Daehyun Baek, Judit Villén, Chanseok Shin, Fernando D Camargo, Steven P Gygi y David P Bartel. El impacto de los microARN en la producción de proteínas. *Nature*, 455 (7209) :64—71, septiembre de 2008.
- [2] David P Bartel. MicroARN: genómica, biogénesis, mecanismo y función. *Cell*, 116 (2) :281—97, enero de 2004.
- [3] M S Bartolomei, S Zemel y S M Tilghman. Impresión parental del gen H19 de ratón. *Naturaleza*, 351 (6322) :153—5, mayo de 1991.
- [4] C J Marrón, A Ballabio, J L Rupert, R G Lafreniere, M Grompe, R Tonlorenzi, y H F Willard. Un gen de la región del centro de inactivación X humano se expresa exclusivamente a partir del cromosoma X inactivo. *Nature*, 349 (6304) :38—44, enero de 1991.
- [5] Richard W Carthew y Erik J Sontheimer. Orígenes y Mecanismos de miRNAs y ARNip. *Cell*, 136 (4) :642—55, febrero de 2009.
- [6] Manel Esteller. RNAs no codificantes en enfermedades humanas. *Nature Reviews Genetics*, 12 (12) :861—874, Novembre de 2011.
- [7] A Fuego, S Xu, M K Montgomery, S A Kostas, S E Conductor, y C C Mello. Potente y específica interferencia genética por ARN bicatenario en *Caenorhabditis elegans*. *Nature*, 391 (6669) :806—11, febrero de 1998.
- [8] Ewan a Gibb, Carolyn J Brown y Wan L Lam. El papel funcional del ARN largo no codificante en carcinomas humanos. *Cáncer molecular*, 10 (1) :38, enero de 2011.
- [9] Daniel E Golden, Vincent R Gerbasi y Erik J Sontheimer. Un trabajo interno para los ARNip. *Molecular cell*, 31 (3) :309—12, agosto de 2008.
- [10] S Guo y K J Kemphues. par-1, un gen requerido para establecer polaridad en embriones de *C. elegans*, codifica una supuesta quinasa Ser/Thr que se distribuye asimétricamente. *Cell*, 81 (4) :611—20, mayo de 1995.
- [11] Rajnish A Gupta, Nilay Shah, Kevin C Wang, Jeewon Kim, Hugo M Horlings, David J Wong, Miao-Chih Tsai, Tiffany Hung, Pedram Argani, John L Rinn, Yulei Wang, Pius Brzoska, Benjamin Kong, Rui Li, Robert B West, Marc J van de Vijver, Saraswati Sukumar y Howard Y Chang. El ARN largo no codificante HOTAIR reprograma el estado de la cromatina para promover la metástasis del cáncer. *Naturaleza*, 464 (7291) :1071—6, abril de 2010.
- [12] Masahira Hattori. Finalización de la secuencia eucromática del genoma humano. *Nature*, 431 (7011) :931—45, octubre de 2004.
- [13] Christopher L Holley y Veli K Topkara. Una introducción a los ARN pequeños no codificantes: miARN y SNORNA. *Medicamentos y Terapia Cardiovascular*, 25 (2) :151—159, 2011.
- [14] Nicholas T Ingolia, Sina Ghaemmaghami, John R S Newman y Jonathan S Weissman. Análisis de todo el genoma in vivo de la traducción con resolución de nucleótidos mediante perfiles de ribosomas. *Science (New York, N.Y.)*, 324 (5924) :218—23, abril de 2009.
- [15] R C Lee, R L Feinbaum y V Ambros. El gen heterocrónico lin-4 de *C. elegans* codifica ARN pequeños con complementariedad antisentido a lin-14. *Cell*, 75 (5) :843—54, diciembre de 1993.
- [16] Michael L Metzker. Tecnologías de secuenciación - la próxima generación. *Nature Reviews Genetics*, 11 (1) :31—46, enero de 2010.
- [17] C. Napoli, C. Lemieux y R. Jorgensen. La introducción de un gen químico de chalcona sintasa en petunia da como resultado la cosupresión reversible de genes homólogos en trans. *La célula vegetal*, 2 (4) :279—289, abril de 1990.
- [18] Laura Poliseno, Leonardo Salmena, Jiangwen Zhang, Brett Carver, William J Haveman y Pier Paolo Pandolfi. Una función independiente de la codificación de los ARNm de genes y pseudogenes regula la biología tumoral. *Nature*, 465 (7301) :1033—8, junio de 2010.

- [19] Chris P Ponting, Peter L Oliver y Wolf Reik. Evolución y funciones de los ARN largos no codificantes. *Cell*, 136 (4) :629—41, febrero de 2009.
- [20] J. R. Prensner y A. M. Chinnaiyan. La emergencia de los lncRNAs en la biología del cáncer. *Descubrimiento del cáncer*, 1 (5) :391—407, octubre de 2011.
- [21] B J Reinhart, F J Slack, M Basson, A E Pasquinelli, J C Bettinger, A E Rougvie, H R Horvitz y G Ruvkun. El ARN let-7 de 21 nucleótidos regula el tiempo de desarrollo en *Caenorhabditis elegans*. *Nature*, 403 (6772) :901—6, febrero de 2000.
- [22] N Romano y G Macino. Quelling: inactivación transitoria de la expresión génica en *Neurospora crassa* por transformación con secuencias homólogas. *Microbiología molecular*, 6 (22) :3343—53, noviembre de 1992.
- [23] G Ruvkun. Biología molecular. Destellos de un diminuto mundo de ARN. *Ciencia*, 294 (5543) :797—9, octubre de 2001.
- [24] Ryan J Taft, Ken C Pang, Timothy R Mercer, Marcel Dinger y John S Mattick. ARN no codificantes: reguladores de la enfermedad. *The Journal of pathology*, 220 (2) :126—39, enero de 2010.
- [25] Jiayi Wang, Xiangfan Liu, Huacheng Wu, Peihua Ni, Zhidong Gu, Yongxia Qiao, Ning Chen, Fenyong Sun y Qishi Fan. CREB regula positivamente la expresión de HULC de ARN largo no codificante a través de la interacción con microARN-372 en cáncer de hígado. *Investigación de ácidos nucleicos*, 38 (16) :5366—83, septiembre de 2010.
- [26] Soraya Yekta, I-Hung Shih y David P Bartel. Escisión dirigida por microARN del ARNm de HOXB8. *Ciencia*, 304 (5670) :594—6, abril de 2004.

---

This page titled [13.3: Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [13.3: Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 14: Secuenciación de ARNm para análisis de expresión y descubrimiento de transcritos

[14.1: Introducción](#)

[14.2: Microarrays de expresión](#)

[14.3: La biología de la secuenciación de ARNm](#)

[14.4: Mapeo de Lectura - Alineación Espaciada de](#)

[14.5: Reconstrucción](#)

[14.6: Cuantificación](#)

---

This page titled [14: Secuenciación de ARNm para análisis de expresión y descubrimiento de transcritos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 14.1: Introducción

El propósito de la secuenciación de ARNm (RNA-seq) es medir los niveles de transcritos de ARNm para cada gen en una célula dada. La secuenciación de ARNm fue una tarea desalentadora, y requiere aproximadamente 40 millones de lecturas alineadas para medir con precisión los transcritos de mRNA. Esto no fue posible hasta 2009, cuando las tecnologías de secuenciación de generación se volvieron más avanzadas y eficientes.

En este capítulo, exploraremos las diferentes técnicas para usar datos de secuenciación de ARNm para ayudar en el descubrimiento de genes y transcritos, así como en el análisis de expresión.

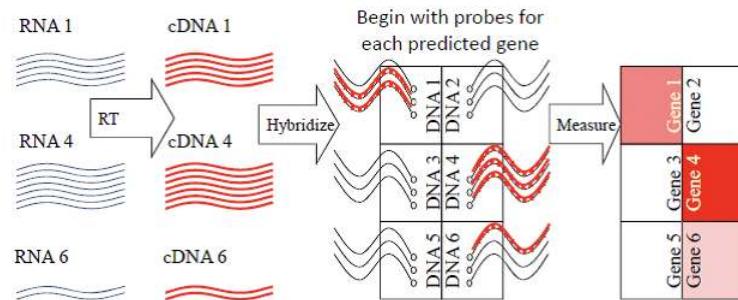
---

This page titled [14.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **14.1: Introduction** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 14.2: Microarrays de expresión

Antes del desarrollo de la tecnología de secuenciación de ARNm, los niveles de ARNm se midieron usando microarrays de expresión. Estas micromatrices funcionan insertando una sonda de ADN en un portaobjetos y midiendo los niveles de transcritos que se someten a hibridación complementaria con el ADN, proceso que podría analizar la expresión de gen a gen (Figura 1).



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 14.1: Figura 1: Proceso de micromatriz de expresión

Sin embargo, esta tecnología tiene varias limitaciones: no puede distinguir las isoformas de ARNm, no puede analizar en la secuencia, ni a nivel digital, solo puede medir transcritos conocidos y las mediciones de expresión se vuelven menos confiables para niveles de transcritos altamente saturados.

This page titled [14.2: Microarrays de expresión](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [14.2: Expression Microarrays](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 14.3: La biología de la secuenciación de ARNm

El primer paso en la secuenciación del ARNm es lisar las células de interés. Esto crea una masa de proteínas, nucleótidos y otras moléculas que luego se filtran para que solo queden moléculas de ARN (o específicamente ARNm). Los transcritos resultantes se fragmentan en lecturas de 200-1000 pares de bases de largo y se someten a una reacción de transcripción inversa para construir una biblioteca de ADN específica de cadena. Finalmente, ambos extremos de estos fragmentos de ADN se secuencian. Después de establecer estas lecturas secuenciadas, la parte computacional de RNA-seq se puede dividir en tres partes: mapeo de lectura, reconstrucción y cuantificación.

This page titled [14.3: La biología de la secuenciación de ARNm](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **14.3: The Biology of mRNA Sequencing** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 14.4: Mapeo de Lectura - Alineación Espaciada de

La idea detrás del mapeo de lectura es alinear las lecturas secuenciadas con un genoma de referencia. Los algoritmos de alineación de secuencias discutidos en capítulos anteriores no funcionarán para este caso debido a la escala del problema. El objetivo es alinear millones de lecturas con el genoma y tomaría demasiado tiempo si cada una estuviera alineada individualmente. En su lugar, presentaremos el enfoque de Alineación de Semillas Espaciadas. Este proceso comienza usando el genoma de referencia para crear una tabla hash de 8 meros, que no tienen que ser contiguos. Las posiciones de estas semillas espaciadas almacenadas se mapean a la tabla hash. Usando estos 8-meros espaciados, cada lectura se compara con cada posición posible en el genoma de referencia y se califica en función del número de coincidencias de pares de bases (Figura 2).

Con mayor precisión, para cada posición, es posible calcular la puntuación usando la ecuación  $q_{MS} = -10 \log_{10} (1 - P(i|G, q))$ , donde  $P(i|G, q)$  representa la probabilidad de que la lectura,  $q$ , se mapee a la posición  $i$  del genoma de referencia  $G$ . Más detalles sobre la obtención de esta puntuación se pueden encontrar en la Figura 13.2.

Es posible ajustar los parámetros de este método para alterar la sensibilidad, velocidad y memoria

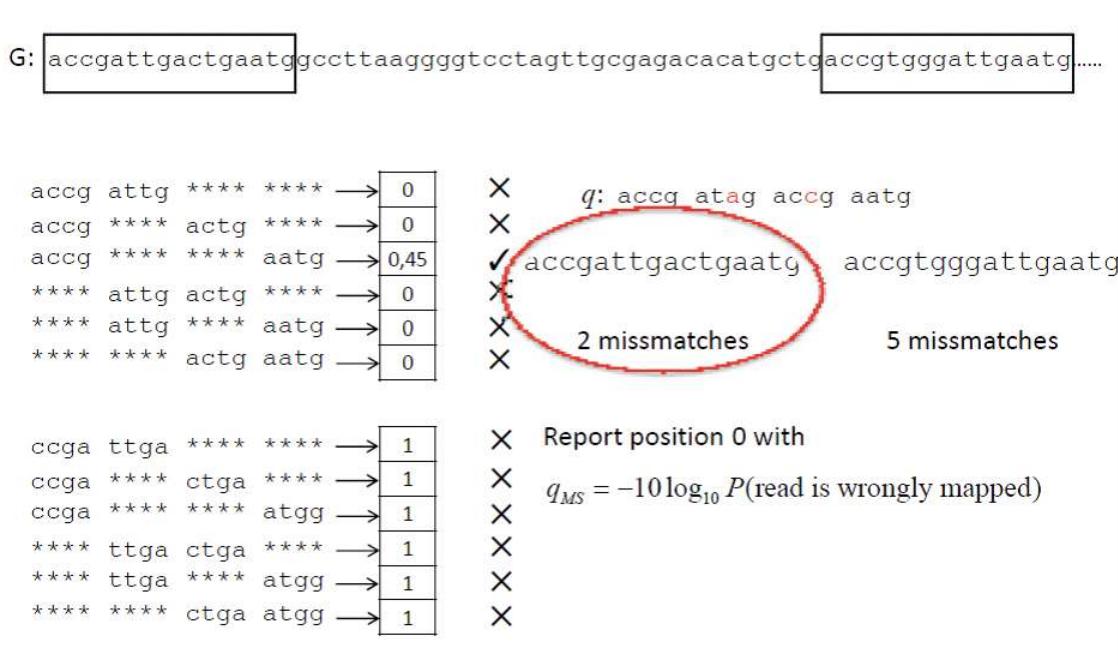


Figura 14.2: Método de k-mer espaciado para mapear lecturas al genoma de referencia

del algoritmo. El uso de semillas k-mer más pequeñas permite una coincidencia de pares de bases menos precisa (mayor sensibilidad), pero requiere que se intenten más coincidencias. Las semillas más pequeñas ocupan menos memoria, mientras que las semillas más grandes corren más rápido.

Existen métodos distintos al descrito anteriormente para realizar esta alineación. El más popular de los cuales es el enfoque Burrows-Wheeler. La transformación Burrows-Wheeler es un algoritmo aún más eficiente para mapear lecturas y se discutirá en un capítulo posterior. Es capaz de acelerar el proceso de búsqueda de coincidencias en el genoma grande reordenando el genoma en una permutación muy específica. Esto permite que las lecturas se emparejen únicamente en función de la longitud de la lectura y no del genoma. Como una mejor tecnología de secuenciación permite longitudes de lectura más grandes, se necesitarán desarrollar más algoritmos para manejar el procesamiento adicional.

A diferencia de Chip-seq, una tecnología similar, RNA-seq es más compleja. Esto se debe a que el mapeador de lectura necesita preocuparse por pequeños exones intercalados entre intrones grandes y poder encontrar ambos lados de un exón. Esta complejidad se puede superar mediante el uso de la técnica de emparejamiento de semillas espaciadas anteriormente mencionada, y detectando cuando dos k-meros de la misma lectura están separados por una larga distancia. Esto señalaría un posible intrón y puede ser fijo extendiendo luego los k-meros para llenar huecos (métodos SNO). Otro método es basar el alineamiento en lecturas contiguas, las cuales se fragmentan aún más en regiones de 20-30 pb. Estas regiones se remapean y las posiciones con dos o más alineaciones

diferentes se marcan como uniones de empalme. Los alineadores de exón primero son más rápidos que los métodos anteriores, pero tienen un costo: no logran diferenciar los psuedogenes, los genes presplicados y los genes transpuestos.

---

This page titled [14.4: Mapeo de Lectura - Alineación Espaciada de](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [14.4: Read Mapping - Spaced Seed Alignment](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 14.5: Reconstrucción

La reconstrucción de lecturas es un problema en gran medida estadístico. El objetivo es determinar una puntuación para cada ventana de tamaño fijo en el genoma. Esta puntuación representa la probabilidad de ver el número observado de lecturas dado el tamaño de la ventana. En otras palabras, ¿es poco probable el número de lecturas en una ventana particular dado el genoma? El número esperado de lecturas por ventana se deriva de una distribución uniforme basada en el número total de lecturas (Figura 3). Esta partitura es modelada por una distribución de Poisson.

Sin embargo, este puntaje debe dar cuenta del problema de múltiples hipótesis de prueba, debido a los aproximadamente 150 millones de bases esperadas. Una opción para hacer frente a esto es la corrección Bonferroni, donde el nominal

Figura 14.3: Recuadro 1: ¿Cómo calculamos los QM?

What does  $q_{MS} = -10 \log_{10} P(\text{read is wrongly mapped})$  mean?

Lets compute the probability the read originated at genome position i

$q$ : accg at~~a~~g acc~~c~~ aatg

$q_s$ : 30 40 25 30 30 20 10 20 40 30 20 30 40 40 30 25

$q_s[k] = -10 \log_{10} P(\text{sequencing error at base k})$ , the PHRED score. Equivalently:

$$P(\text{sequencing error at base k}) = 10^{-\frac{q_s}{10}}$$

So the probability that a read originates from a given genome position i is:

$$P(q | G, i) = \prod_{j \text{ match}} P(q_j \text{ good call}) \prod_{j \text{ mismatch}} P(q_j \text{ bad call}) \approx \prod_{j \text{ mismatch}} P(q_j \text{ bad call})$$

In our example

$$P(q | G, 0) = [(1 - 10^{-3})^6 (1 - 10^{-4})^4 (1 - 10^{-2.5})^2 (1 - 10^{-2})^2] [10^{-1} 10^{-2}] = [0.97]^6 * [0.001] \approx 0.001$$

What does  $q_{MS} = -10 \log_{10} P(\text{read is wrongly mapped})$  mean?

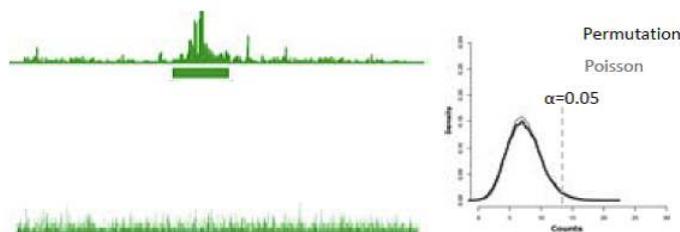
$$P(q | G, i) = \prod_{j \text{ match}} P(q_j \text{ good call}) \prod_{j \text{ mismatch}} P(q_j \text{ bad call}) \approx \prod_{j \text{ mismatch}} P(q_j \text{ bad call})$$

But what we need is the posterior probability, the probability that the region starting at i was sequenced *given* that we observed the read q:

$$P(i | G, q) = \frac{P(q | G, i)P(i | G)}{P(q | G)} = \frac{P(q | G, i)P(i | G)}{\sum_j P(q | G, j)}$$

Fortunately, there are efficient ways to approximate this probability (see Li, H *genome Research* 2008, for example)

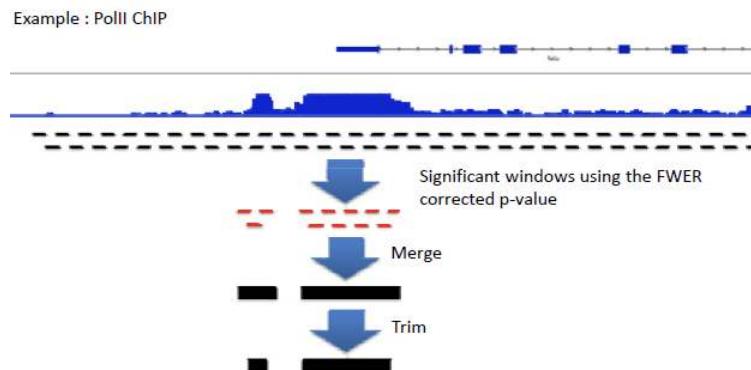
$$q_{MS} = -10 \log_{10}(1 - P(i | G, q))$$



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 14.4: Figura 3: La reconstrucción funciona determinando, para una ventana particular, la probabilidad de observar ese número de lecturas (arriba a la izquierda) dada la distribución uniforme del total de lecturas (abajo a la izquierda). Esta probabilidad sigue la distribución de Poisson.

valor  $p = n * \text{valor } p$ . Este método conduce a una baja sensibilidad, debido a su naturaleza muy conservadora. Otra opción es permutar las lecturas observadas en el genoma, y encontrar el número máximo de lecturas observadas en una sola base. Esto permite un modelo de distribución de conteo máximo, pero el proceso es muy lento. La distribución de escaneo acelera este proceso al computar una forma cerrada para la distribución de conteo máximo para dar cuenta de la dependencia de ventanas superpuestas (Figura 4). La probabilidad de observar  $k$  lecturas en una ventana de tamaño  $w$  en un genoma de tamaño  $L$  dado un total de  $N$  lecturas puede aproximarse por [el portaobjetos no está claro].



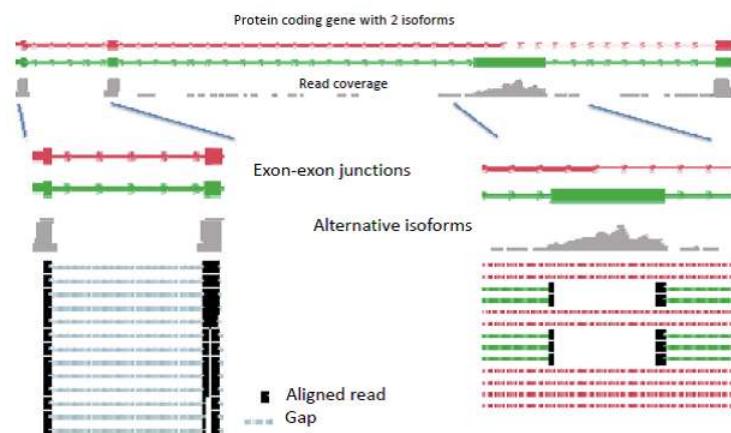
© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 14.5: Figura 4: Proceso de reconstrucción del genoma a partir de lecturas, utilizando la distribución de exploración

Elegir un tamaño de ventana también es una decisión importante, ya que los genes existen en diferentes niveles de expresión y abarcan diferentes órdenes de magnitud. Las ventanas pequeñas son mejores para detectar regiones puntuadas, mientras que las ventanas más grandes pueden detectar intervalos más largos de mejora moderada. En la mayoría de los casos, se utilizan ventanas de diferentes tamaños para captar señales de tamaño variable.

La reconstrucción de la transcripción puede verse como un problema de segmentación, con varios desafíos. Como se mencionó anteriormente, los genes se expresan en diferentes niveles, en varios órdenes de magnitud. Además, las lecturas utilizadas para la reconstrucción se obtienen de ARNm tanto maduro como inmaduro, este último aún conteniendo intrones. Finalmente, muchos genes tienen múltiples isoformas, y la naturaleza corta de las lecturas dificulta la diferenciación entre estos diferentes transcriptos. Una herramienta computacional llamada Escritura utiliza el conocimiento a priori de conectividad de fragmentos para detectar transcripciones.

Las isoformas alternativas solo se pueden detectar a través de lecturas de unión de exones, que contienen los extremos de un exón. Las lecturas más largas tienen una mayor probabilidad de abarcar estos cruces (Figura 5). La Escritura trabaja modelando las lecturas usando la estructura gráfica, donde las bases están conectadas a bases vecinas, así como empalmar vecinos. Este proceso difiere de la técnica del gráfico de cadenas, porque se enfoca en el genoma completo, y no mapea secuencias superpuestas directamente. Al deslizar la ventana, las Escrituras pueden saltar a través de uniones de empalme pero aún así examinar isoformas alternativas. A partir de esta gráfica de conectividad orientada, el programa identifica segmentos a través de la gráfica y busca segmentos significativos (Recuadro 2).



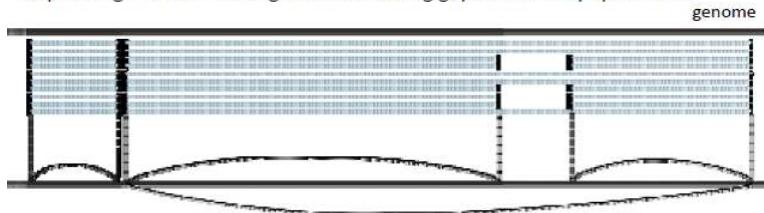
© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 14.6: Figura 5: Las isoformas alternativas presentan un desafío para la reconstrucción, que debe depender de lecturas que abarcan la unión del exón

El ensamblaje de transcripciones directas es otro método de reconstrucción (a diferencia de los métodos guiados por el genoma como la Escritura). Los métodos de ensamblaje de transcritos son capaces de reconstruir transcritos a partir de organismos sin una secuencia de referencia, mientras que los enfoques guiados por genomas son ideales para anotar genomas de alta calidad y expandir el catálogo de transcritos expresados. Los enfoques híbridos se utilizan para transcritos o transcriptomas de menor calidad que han sido sometidos a reordenamientos importantes, como los de las células cancerosas. Las herramientas populares de ensamblaje de transcripciones incluyen Oasis, Trans-Abyss y Trinity. Otro software popular guiado por genoma es Cufflinks. Independientemente de la metodología o el tipo de software, cualquier experimento de secuenciación que produzca más cobertura genómica experimentará una mejor reconstrucción del transcríto.

Figura 14.7: Recuadro 2: El método de las Escrituras

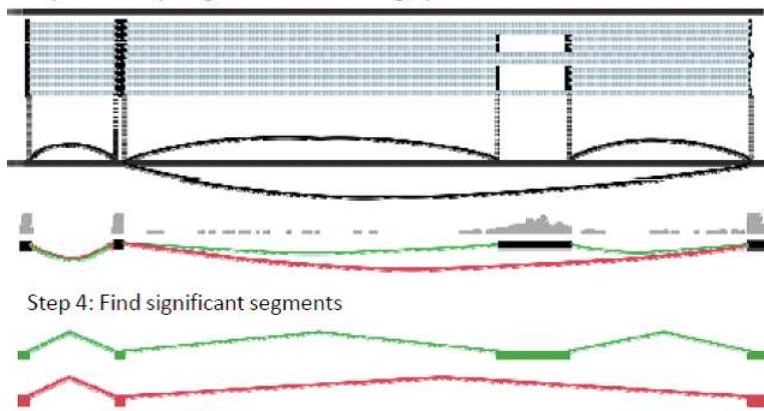
Step 1: Align Reads to the genome allowing gaps flanked by splice sites



Step 2: Build an oriented connectivity graph using every spliced alignment and orienting edges using the flanking splicing motifs

The “connectivity graph” connects all bases that are directly connected within the transcriptome

Step 3: Identify “segments” across the graph



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

This page titled [14.5: Reconstrucción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [14.5: Reconstruction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 14.6: Cuantificación

El objetivo de la etapa de cuantificación es puntuar regiones en el genoma en función del número de lecturas. Recordemos que cada transcripción está fragmentada en muchas lecturas más pequeñas. Por lo tanto, es insuficiente simplemente contar el número de lecturas por región, ya que este valor estaría influenciado por (1) las tasas de expresión y (2) la longitud de la transcripción. Cuanto mayor sea la tasa de expresión de una transcripción, más lecturas tendremos para ello. De igual manera, cuanto más larga sea una transcripción, más lecturas tendremos. Este problema se puede resolver normalizando el número de lecturas por la longitud de la transcripción y el número total de lecturas en el experimento. Esto proporciona el valor RPGM, o lecturas por kilobase de secuencia exónica por millón de lecturas mapeadas.

Este método es robusto para genes con una sola isoforma. Sin embargo, existe la posibilidad de solapamiento entre variantes conflictivas de una transcripción. Cuando están involucradas múltiples variantes de transcripción, este problema se conoce como análisis de expresión diferencial. Existen algunos métodos diferentes para manejar esta complejidad. El modelo de intersección de exones puntúa solo los exones constituyentes. El modelo de unión de exones simplemente puntúa basándose en una transcripción fusionada, pero puede sesgarse fácilmente en función de las proporciones relativas de cada isoforma. Un modelo más completo es el modelo de expresión de transcripción, que asigna lecturas únicas a diferentes isoformas.

---

This page titled [14.6: Cuantificación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [14.6: Quantification](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 15: Regulación Génica I - Agrupación de Expresión Génica

- 15.1: Introducción
- 15.2: Métodos para medir la expresión génica
- 15.3: Algoritmos de Clustering
- 15.4: Direcciones actuales de investigación
- 15.5: Lectura adicional
- 15.6: Recursos
- 15.7: Qué hemos aprendido, Bibliografía

---

This page titled [15: Regulación Génica I - Agrupación de Expresión Génica](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

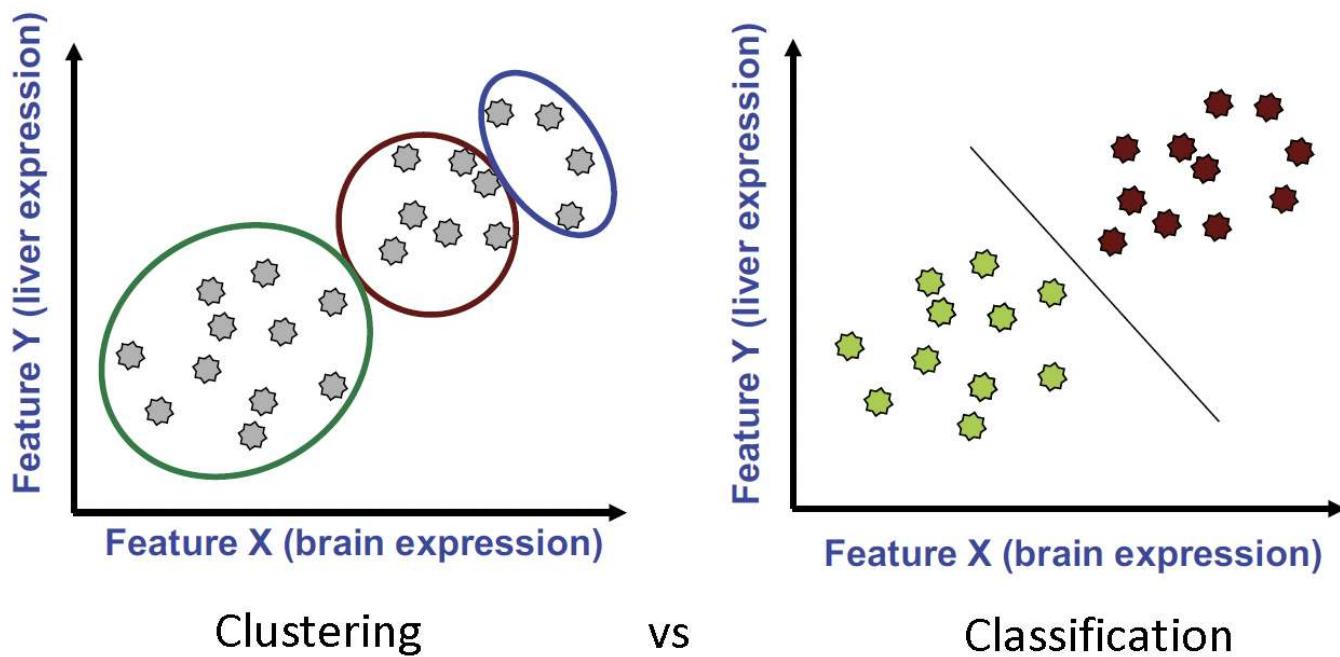
## 15.1: Introducción

En este capítulo, consideraremos el problema de discernir similitudes o patrones dentro de grandes conjuntos de datos. Encontrar la estructura en dichos conjuntos de datos nos permite sacar conclusiones sobre el proceso, así como la estructura subyacente a las observaciones. Abordamos este problema a través de la aplicación de técnicas de clustering. El siguiente capítulo se centrará en las técnicas de clasificación.

### Agrupación vs Clasificación

Una distinción importante que debe hacerse desde el principio es la diferencia entre clasificación y agrupamiento. Clasificación es el problema de identificar a cuál de un conjunto de categorías (subpoblaciones) pertenece una nueva observación, a partir de un conjunto de entrenamiento de datos que contienen observaciones o instancias cuya categoría miembro se conoce. El conjunto de entrenamiento se utiliza para aprender reglas que asignarán etiquetas con precisión a nuevas observaciones. La dificultad es encontrar las características más importantes (selección de características).

En la terminología del aprendizaje automático, la clasificación se considera una instancia de aprendizaje supervisado, es decir, aprendizaje donde se dispone de un conjunto de formación de observaciones correctamente identificadas. El procedimiento no supervisado correspondiente se conoce como clustering o cluster analysis, e implica agrupar los datos en categorías basadas en alguna medida de similitud inherente, como la distancia entre instancias, consideradas como vectores en un espacio vectorial multidimensional. La dificultad es identificar la estructura de los datos. La Figura 15.1 ilustra la diferencia entre agrupamiento y clasificación.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 15.1: Agrupación comparada con la clasificación. En la agrupación agrupamos las observaciones en clústeres en función de lo cerca que están entre sí. En clasificación queremos una regla que asigne etiquetas con precisión a nuevos puntos.

### Aplicaciones

La agrupación se desarrolló originalmente dentro del campo de la inteligencia artificial. Poder agrupar objetos similares, con implicaciones plenas de generalidad implícitas, es de hecho un atributo bastante deseable para una inteligencia artificial, y uno que los humanos realizan rutinariamente a lo largo de la vida. A medida que el desarrollo de algoritmos de clustering avanzaba a buen ritmo, rápidamente quedó claro que no había ninguna barrera intrínseca involucrada en la aplicación de estos algoritmos a

conjuntos de datos cada vez más grandes. Esta realización condujo a la rápida introducción del agrupamiento en la biología computacional y otros campos que se ocupan de grandes conjuntos de datos.

La agrupación en clústeres tiene muchas aplicaciones para la biología computacional. Por ejemplo, consideremos los perfiles de expresión de muchos genes tomados en diversas etapas de desarrollo. La agrupación puede mostrar que ciertos conjuntos de genes se alinean (es decir, muestran los mismos niveles de expresión) en varias etapas. Esto puede indicar que este conjunto de genes tiene expresión o regulación común y podemos usar esto para inferir una función similar. Además, si encontramos un gen no caracterizado en dicho conjunto de genes, podemos razonar que el gen no caracterizado también tiene una función similar a través de la culpa por asociación.

Las marcas de cromatina y los motivos reguladores se pueden usar para predecir relaciones lógicas entre reguladores y genes diana de manera similar. Este tipo de análisis permite la construcción de modelos que permiten predecir la expresión génica. Estos modelos se pueden utilizar para modificar las propiedades reguladoras de un gen en particular, predecir cómo surgió un estado de enfermedad o ayudar a dirigir genes a órganos particulares basados en circuitos reguladores en las células del órgano relevante.

La biología computacional trata con conjuntos de datos cada vez más grandes y de acceso abierto. Un ejemplo de ello es el proyecto ENCODE [2]. Lanzado en 2003, el objetivo de ENCODE es construir una lista completa de elementos funcionales en el genoma humano, incluyendo elementos que actúan a nivel de proteína y ARN, y elementos reguladores que controlan las células y circunstancias en las que un gen está activo. Los datos de CODIFICAR ahora están disponibles libre e inmediatamente para todo el genoma humano: <http://genome.ucsc.edu/ENCODE/>. Utilizando todos estos datos, es posible hacer predicciones funcionales sobre los genes mediante el uso de clustering.

---

This page titled [15.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

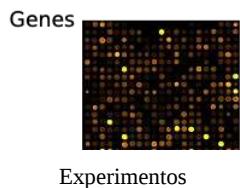
- [15.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.2: Métodos para medir la expresión génica

La forma más intuitiva de investigar un determinado fenotipo es medir los niveles de expresión de proteínas funcionales presentes en un momento dado en la célula. Sin embargo, medir la concentración de proteínas puede ser difícil, debido a sus diferentes ubicaciones, modificaciones y contextos en los que se encuentran, así como por la incompletitud del proteoma. Sin embargo, los niveles de expresión de ARNm son más fáciles de medir y a menudo son una buena aproximación. Al medir el ARNm, analizamos la regulación a nivel de transcripción, sin las complicaciones agregadas de la regulación traduccional y la degradación activa de proteínas, lo que simplifica el análisis a costa de perder información. En este capítulo, consideraremos dos técnicas para generar datos de expresión génica: microarrays y RNA-seq.

### Microarrays

Las micromatrices permiten el análisis de los niveles de expresión de miles de genes preseleccionados en un experimento. El principio básico detrás de las micromatrices es la hibridación de fragmentos de ADN complementarios. Para comenzar, segmentos cortos de ADN, conocidos como sondas, se unen a una superficie sólida, comúnmente conocida como chip génico. Luego, la población de ARN de interés, que ha sido tomada de una célula, se transcribe de forma inversa a ADNc (ADN complementario) vía transcriptasa inversa, que sintetiza ADN a partir de ARN usando la cola poli-A como cebador. Para secuencias intergénicas que no tienen cola poli-A, se puede ligar un cebador estándar a los extremos del ARNm. El ADN resultante tiene más complementariedad con el ADN en el portaobjetos que el ARN. El ADNc se lava luego sobre el chip y la hibridación resultante desencadena la fluorescencia de las sondas. Esto se puede detectar para determinar la abundancia relativa del ARNm en la diana, como se ilustra en la figura 15.2.



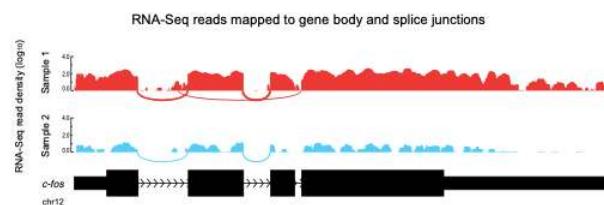
© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 15.2: Los valores de expresión génica de experimentos de micromatrices se pueden representar como mapas de calor para visualizar el resultado del análisis de datos.

Actualmente se utilizan dos tipos básicos de microarrays. Los chips génicos de Affymetrix tienen una mancha por cada gen y tienen sondas más largas del orden de 100 nucleótidos. Por otro lado, los conjuntos de oligonucleótidos manchados tejan genes y tienen sondas más cortas alrededor de las decenas de bases.

Existen numerosas fuentes de error en los métodos actuales y los métodos futuros buscan eliminar pasos en el proceso. Por ejemplo, la transcriptasa inversa puede introducir desapareamientos, que debilitan la interacción con la sonda correcta o causan hibridación cruzada, o unión a múltiples sondas. Una solución a esto ha sido usar múltiples sondas por gen, ya que la hibridación cruzada será diferente para cada gen. Aún así, la transcripción inversa es necesaria debido a la estructura secundaria del ARN. La estabilidad estructural del ADN hace que sea menos probable que se doble y no se hibride con la sonda. La siguiente generación de tecnologías, como RNA-seq, secuencian el ARN a medida que sale de la célula, sondeando esencialmente cada base del genoma.

## RNA-seq



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 15.3: lecturas de ARN-seq mapeando a un gen (c-fos) y sus uniones de corte y empalme. Las densidades a lo largo del exón representan densidades de lectura mapeadas a exones (en log10), los arcos corresponden a lecturas de unión, donde el ancho del arco se dibuja en proporción al número de lecturas en esa unión. El gen está regulado a la baja en la Muestra 2 en comparación con la Muestra 1.

RNA-seq, también conocida como secuenciación de escopeta de transcriptoma completo, intenta realizar la misma función que las micromatrices de ADN se han utilizado en el pasado, pero con mayor resolución. En particular, las micromatrices de ADN utilizan sondas específicas, y la creación de estas sondas depende necesariamente del conocimiento previo del genoma y del tamaño de la matriz que se está produciendo. ARN-seq elimina estas limitaciones simplemente secuenciando todo el ADNc producido en experimentos de micromatrices. Esto es posible gracias a la tecnología de secuenciación de próxima generación. La técnica ha sido rápidamente adoptada en estudios de enfermedades como el cáncer [4]. Los datos de RNA-seq se analizan luego agrupando de la misma manera que normalmente se analizarían los datos de las micromatrices.

## Matrices de Expresión Génica

Las micromatrices y ARN-seq se utilizan frecuentemente para comparar los perfiles de expresión génica de las células en diversas condiciones. La cantidad de datos generados a partir de estos experimentos es enorme. Las micromatrices pueden analizar miles de genes, y ARN-seq puede, en principio, analizar cada gen que se expresa activamente. El nivel de expresión de cada uno de esos genes se mide a través de una variedad de condiciones, incluyendo cursos de tiempo, etapas de desarrollo, fenotipos, sano vs. enfermo y otros factores.

Para entender lo que transmite el mapa de calor de una matriz de expresión génica (Figura 15.4), primero tenemos que entender lo que nos dice la matriz de datos de expresión. Mediante el uso de microarrays y RNA-seq, podemos obtener el nivel de expresión génica en forma cuantitativa en un experimento. Si tenemos múltiples experimentos, podemos construir una matriz de valores (Figura 15.5) que representa un valor logarítmico de (T/R), donde T es el nivel de expresión génica en la muestra de prueba y R es el nivel de expresión génica en la muestra de referencia.

La matriz de expresión eliminada debido a restricciones de derechos de autor.

Figura 15.4: Transformación de la Figura 4 en un mapa de calor

Si visualizamos la matriz como un mapa de calor, entonces obtenemos la siguiente nueva matriz coloreada:

	Exp 1	Exp 2	Exp 3	Exp 4	Exp 5	Exp 6
Gene 1	-1.2	-2.1	-3	-1.5	1.8	2.9
Gene 2	2.7	0.2	-1.1	1.6	-2.2	-1.7
Gene 3	-2.5	1.5	-0.1	-1.1	-1	0.1
Gene 4	2.9	2.6	2.5	-2.3	-0.1	-2.3
Gene 5	0.1		2.6	2.2	2.7	-2.1
Gene 6	-2.9	-1.9	-2.4	-0.1	-1.9	2.9

Figura 15.5: Nivel de expresión génica en comparación del valor logarítmico con la muestra

Estas matrices pueden agruparse jerárquicamente mostrando la relación entre pares de genes, pares de pares, etc., creando un dendrograma en el que se pueden ordenar las filas y columnas usando algoritmos óptimos de ordenación de hojas.

Imagen en el dominio público. Esta gráfica se generó utilizando el programa Cluster from Michael Eisen, que está disponible en [Rana.lbl.gov/EisenSoftware.htm](http://Rana.lbl.gov/EisenSoftware.htm), con datos extraídos de la base de datos Stembase de datos de expresión génica.

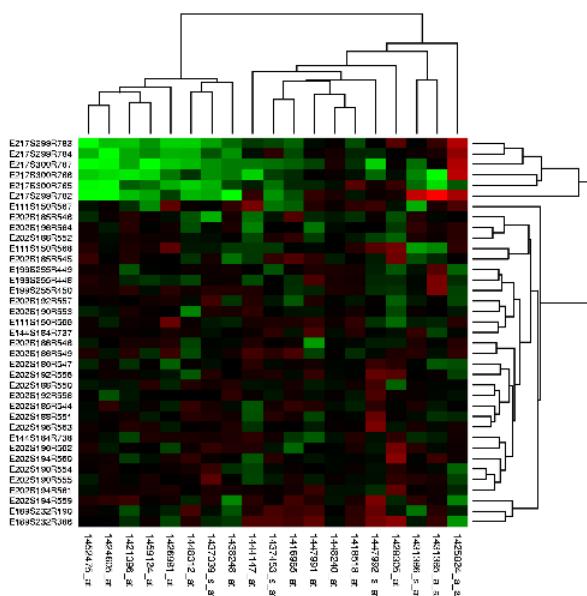
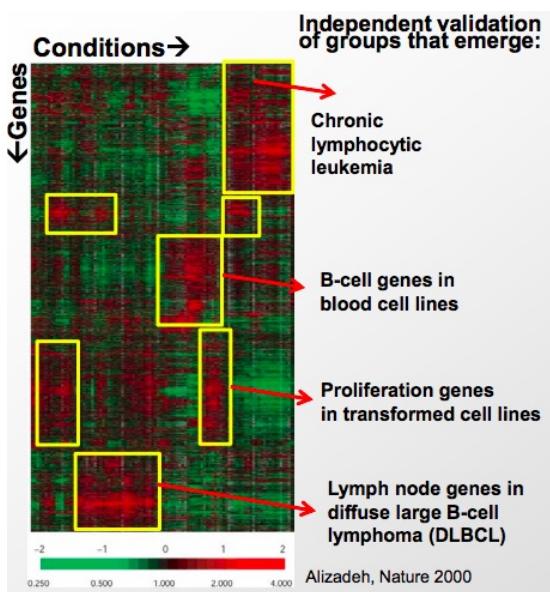


Figura 15.6: Matriz de muestra de valores de expresión génica, representada como un mapa de calor y con conglomerados jerárquicos. [1]

Al revelar la estructura oculta de un largo segmento del genoma, obtenemos una gran comprensión de lo que hace un fragmento de gen, y posteriormente entendemos más sobre la causa raíz de una enfermedad desconocida.



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Alizadeh, Ash A., Michael B. Eisen, et al. "Distintos tipos de linfoma difuso de células B grandes identificados por perfiles de expresión génica". *Naturaleza* 403, núm. 6769 (2000): 503-11.

Figura 15.7: Uso de matriz de expresión génica para inferir más sobre una enfermedad y un segmento génico

Este poder predictivo y analítico se incrementa debido a la capacidad de biclustering de los datos; es decir, agrupamiento a lo largo de ambas dimensiones de la matriz. La matriz permite comparar perfiles de expresión de genes, así como comparar la similitud de diferentes padecimientos como enfermedades. Un reto, sin embargo, es la maldición de la dimensionalidad. A medida que aumenta el espacio de los datos, disminuye el agrupamiento de los puntos. A veces, los datos se pueden reducir a espacios dimensionales más bajos para encontrar estructura en los datos usando agrupamiento para inferir qué puntos pertenecen juntos en función de la proximidad.

Interpretar los datos también puede ser un reto, ya que puede haber otros fenómenos biológicos en juego. Por ejemplo, los exones codificantes de proteínas tienen mayor intensidad, debido a que los intrones se degradan rápidamente. Al mismo tiempo, no todos los intrones son basura y puede haber ambigüedades en el empalme alternativo. También hay mecanismos celulares que degradan los transcritos aberrantes a través de la decadencia mediada sin sentido.

---

This page titled [15.2: Métodos para medir la expresión génica](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **15.2: Methods for Measuring Gene Expression** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.3: Algoritmos de Clustering

Para analizar los datos de expresión génica, es común realizar análisis de agrupamiento. Existen dos tipos de algoritmos de agrupamiento: particionamiento y aglomerativo. La agrupación particional divide los objetos en clústeres no superpuestos para que cada objeto de datos esté en un subconjunto. Alternativamente, los métodos de agrupamiento aglomerativo producen un conjunto de clústeres anidados organizados como una jerarquía que representa estructuras de niveles de detalle más amplios a más finos.

### Agrupación K-Means

El algoritmo k-means agrupa n objetos en función de sus atributos en k particiones. Este es un ejemplo de partición, donde cada punto se asigna a exactamente un clúster de tal manera que se minimiza la suma de distancias desde cada punto hasta su centro etiquetado correspondientemente. La motivación subyacente a este proceso es hacer los clusters más compactos posibles, generalmente en términos de una métrica de distancia euclidiana.

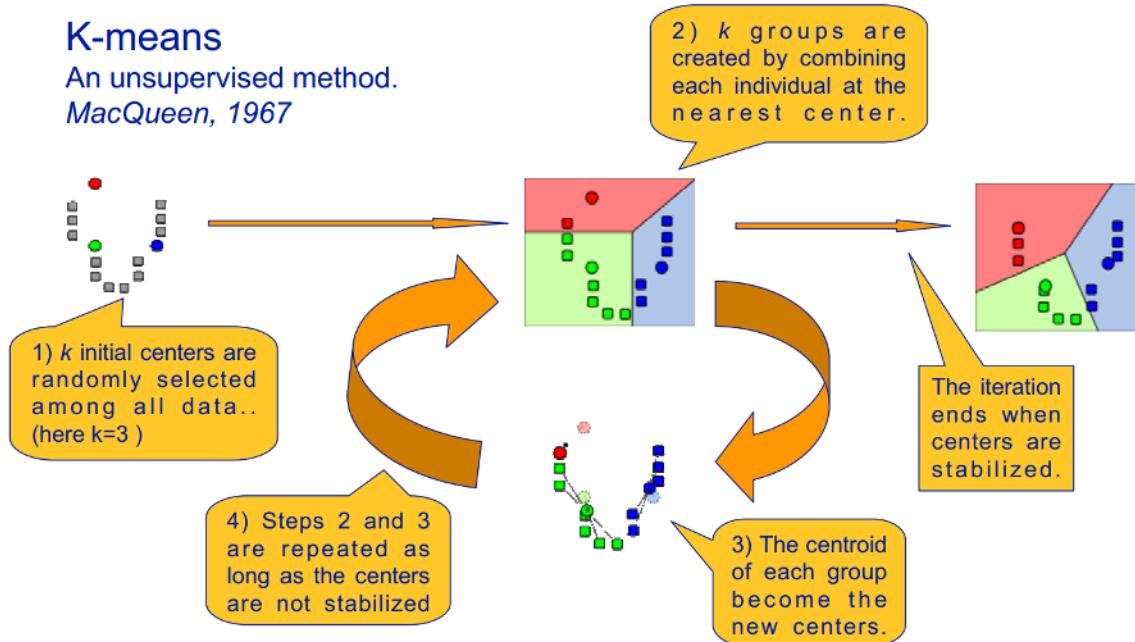


Figura 15.8: El algoritmo de agrupación de k-medias © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

El algoritmo k-means, como se ilustra en la figura 15.8, se implementa de la siguiente manera:

1. Supongamos un número fijo de clústeres, k
2. **Inicialización:** Inicializar aleatoriamente las k medias  $\mu_k$  asociadas a los clústeres y asignar cada punto de datos  $x_i$  al clúster más cercano, donde la distancia entre  $x_i$  y  $\mu_k$  viene dada por  $d_{i,k} = (x_i - \mu_k)^2$ .
3. **Iteración:** Recalcular el centroide del clúster dados los puntos que se le asignan:  $\mu_k(n+1) = \sum_{x_i \in k} \frac{x_i}{|x_k|}$  donde  $x_k$  es el número de puntos con etiqueta k. Reasignar puntos de datos a los k nuevos centroides por la métrica de distancia dada. Los nuevos centros se calculan efectivamente para ser el promedio de los puntos asignados a cada cluster.
4. **Terminación:** Iterar hasta la convergencia o hasta que se haya alcanzado un número de iteraciones especificado por el usuario. Tenga en cuenta que la iteración puede quedar atrapada en algún óptimo local.

Existen varios métodos para elegir k: simplemente mirar los datos para identificar clústeres potenciales o intentar iterativamente valores para n, mientras penaliza la complejidad del modelo. Siempre podemos hacer mejores clústeres aumentando k, pero en algún momento comenzamos a sobreajustar los datos.

También podemos pensar en k-means como tratar de minimizar un criterio de costo asociado con el tamaño de cada clúster, donde el costo aumenta a medida que los clústeres se vuelven menos compactos. Sin embargo, algunos puntos pueden estar casi a medio camino entre dos centros, lo que no encaja bien con el agrupamiento binario que pertenece a k-means.

## Agrupación difusa de K-medias

En la **agrupación difusa**, cada punto tiene una probabilidad de pertenecer a cada clúster, en lugar de pertenecer completamente a un solo clúster. Fuzzy k-means trata específicamente de lidiar con el problema donde los puntos están algo entre centros o de otra manera ambiguos reemplazando la distancia con la probabilidad, que por supuesto podría ser alguna función de la distancia, como tener probabilidad relativa a la inversa de la distancia. La k-media difusa usa un centroide ponderado basado en esas probabilidades. Los procesos de inicialización, iteración y terminación son los mismos que los utilizados en k-medias. Los conglomerados resultantes se analizan mejor como distribuciones probabilísticas en lugar de una asignación dura de etiquetas. Uno debería darse cuenta de que k-means es un caso especial de k-medias difusas cuando la función de probabilidad utilizada es simplemente 1 si el punto de datos está más cerca de un centroide y 0 en caso contrario.

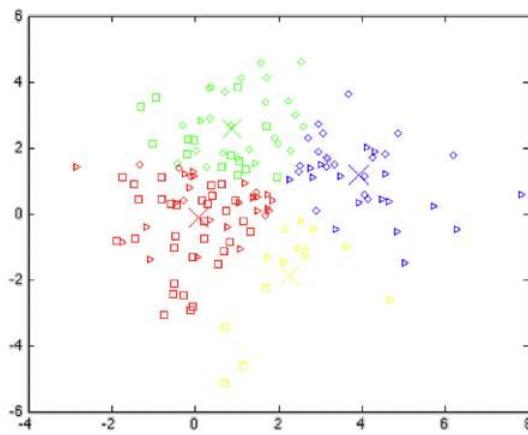


Figura 15.9: Ejemplos de asignaciones finales de conglomerados de k-medias difusas usando  $k= 4$  con centroides, clústeres correctos y clústeres asignados más probables marcados como cruces, formas de puntos y colores respectivamente. Tenga en cuenta que el conjunto de datos original no es gaussiano.

El algoritmo difuso k-means es el siguiente:

1. Asumir un número fijo de clústeres  $k$
2. Inicialización: Inicializar aleatoriamente las  $k$  medias  $\mu_k$  asociadas a los clústeres y calcular la probabilidad de que cada punto de datos  $x_i$  sea miembro de un clúster dado  $k$ ,  $P$  (el punto  $x_i$  tiene la etiqueta  $k|x_i, k$ ).
3. Iteración: Recalcular el centroide del clúster como el centroide ponderado dadas las probabilidades de pertenencia a todos los puntos de datos  $x_i$ :

$$\mu_k(n+1) = \frac{\sum_{x_i \in k} x_i \times P(\mu_k | x_i)^b}{\sum_{x_i \in k} P(\mu_k | x_i)^b}$$

Y recalcular las membresías actualizadas  $P(\mu_k | x_i)$  (hay diferentes formas de definir membresía, aquí hay solo un ejemplo):

$$P(\mu_k | x_i) = \left( \sum_{j=1}^k \left( \frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{b-1}} \right)^{-1}$$

4. Terminación: Iterar hasta que la matriz de membresía converja o hasta que se haya alcanzado un número de iteraciones especificado por el usuario (la iteración puede quedar atrapada en algunos máximos o mínimos locales)

El  $b$  aquí es el exponente de ponderación que controla los pesos relativos que se colocan en cada partición, o el grado de borrosidad. Cuando  $b > 1$ , las particiones que minimizan la función de error cuadrado son cada vez más duras (no borrosas), mientras que como  $b > \infty$  todas las membresías se acercan a 1, que es el estado más difuso. No hay  $k$  evidencia teórica de cómo elegir un  $b$  óptimo, mientras que los valores empíricos útiles se encuentran entre [1, 30], y en la mayoría de los estudios,  $1.5 \leq b \leq 3.0$  funcionó bien.

## K-Means como modelo generativo

Un **modelo generativo** es un modelo para generar aleatoriamente valores de datos observables, dados algunos parámetros ocultos. Mientras que un modelo generativo es un modelo de probabilidad de todas las variables, un modelo discriminativo proporciona un modelo condicional solo de la(s) variable(s) objetivo(s) usando las variables observadas.

Para hacer de k-medias un modelo generativo, ahora lo miramos de manera probabilística, donde asumimos que los puntos de datos en el clúster  $k$  se generan usando una distribución gaussiana con la media en el centro del clúster y una varianza de 1, lo que da

$$P(x_i | \mu_k) = \frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{(x_i - \mu_k)^2}{2} \right\}. \quad (15.3.1)$$

Esto da una representación estocástica de los datos, como se muestra en la figura 15.10. Ahora esto se convierte en un problema de máxima verosimilitud, que, mostraremos a continuación, es exactamente equivalente al algoritmo original de k-means mencionado anteriormente.

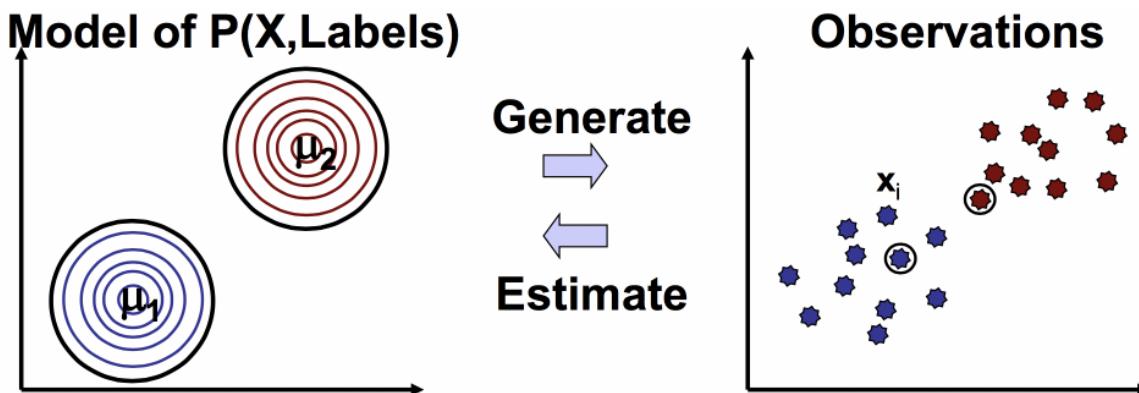


Figura 15.10: K-medias como modelo generativo. Las muestras se extrajeron de distribuciones normales. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

En el paso de generación, queremos encontrar una partición más probable, o asignación de etiqueta, para cada  $x_i$  dada la media  $\mu_k$ . Con la suposición de que cada punto se dibuja de forma independiente, podríamos buscar la etiqueta de máxima verosimilitud para cada punto por separado:

$$\arg \max_k P(x_i | \mu_k) = \arg \max_k \frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{(x_i - \mu_k)^2}{2} \right\} = \arg \min_k (x_i - \mu_k)^2$$

Esto es totalmente equivalente a encontrar el centro de clúster más cercano en el algoritmo original de k-means.

En el paso Estimación, buscamos la estimación de máxima verosimilitud de la media del clúster  $\mu_k$ , dadas las particiones (etiquetas):

$$\begin{aligned} \arg \max_{\mu} \left\{ \log \prod_i P(x_i | \mu) \right\} &= \arg \max_{\mu} \sum_i \left\{ -\frac{1}{2}(x_i - \mu)^2 + \log \left( \frac{1}{\sqrt{2\pi}} \right) \right\} \\ &= \arg \min_{\mu} \sum_i (x_i - \mu)^2 \end{aligned}$$

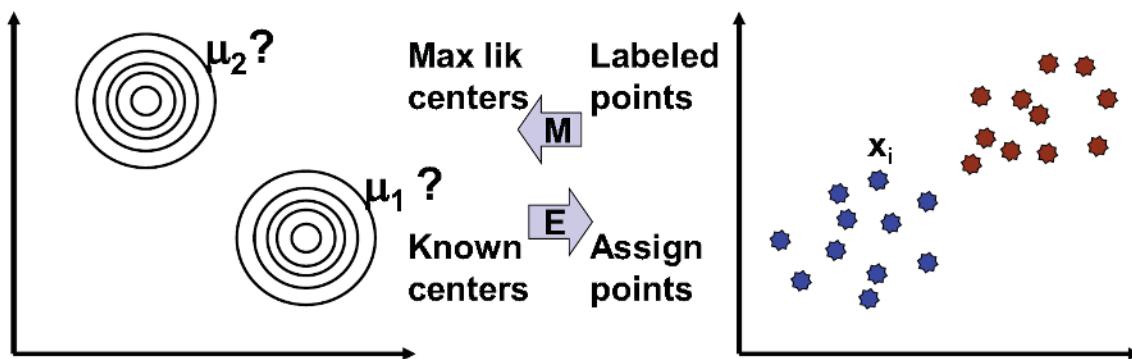
Tenga en cuenta que la solución de este problema es exactamente el centroide de la  $x_i$ , que es el mismo procedimiento que el algoritmo k-means original.

Desafortunadamente, dado que k-medias asume independencia entre los ejes, la covarianza y la varianza no se contabilizan usando k-medias, por lo que modelos como las distribuciones oblongas no son posibles. Sin embargo, este problema puede resolverse cuando se generaliza este problema en un problema de maximización de expectativas.

## Maximización de expectativas

K-medias se puede ver como un ejemplo de EM (**algoritmos de maximización de expectativas**), como se muestra en la figura 15.11 donde la expectativa consiste en la estimación de etiquetas ocultas, Q, y la maximización de la probabilidad esperada ocurre dados los datos y Q. Asignando a cada punto la etiqueta del centro más cercano corresponde a la E paso de estimar la etiqueta más probable dado el parámetro anterior. Después, utilizando los datos producidos en el paso E como observación, mover el centroide al promedio de las etiquetas asignadas a ese centro corresponde al paso M de maximizar la probabilidad del centro dadas las etiquetas. Este caso es análogo al aprendizaje de Viterbi. Se puede hacer una comparación similar para k-medias difusas, que es análoga a Baum-Welch de los HMM. La Figura 15.12 compara el agrupamiento, HMM y el descubrimiento de motivos con respecto al algoritmo de minimización de expectativas.

Cabe señalar que utilizando el marco EM, el enfoque de k medias puede generalizarse a racimos de forma oblonga y tamaños variables. Con k medias, los puntos de datos siempre se asignan al centro de clúster más cercano. Al introducir una matriz de covarianza en la función de probabilidad gaussiana, podemos permitir clústeres de diferentes tamaños. Al establecer la varianza para que sea diferente a lo largo de diferentes ejes, incluso podemos crear distribuciones oblongas.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 15.11: K-medias como algoritmo de maximización de expectativas (EM).

Update rule	Update assignments (E step) → Estimate hidden labels	Algorithm implementing E step in each of the three settings			Update model parameters (M step) → max likelihood
		Expression clustering	HMM learning	Motif discovery	
The hidden label is:		Cluster labels	State path $\pi$	Motif positions	
Pick a best	Assign each point to best label	K-means: Assign each point to nearest cluster	Viterbi training: label sequence with best path	Greedy: Find best motif match in each sequence	Average of those points assigned to label
Average all	Assign each point to all labels, probabilistically	Fuzzy K-means: Assign to all clusters, weighted by proximity	Baum-Welch training: label sequence w all paths (posterior decoding)	MEME: Use all positions as a motif occurrence weighed by motif match score	Average of all points, weighted by membership
Sample one	Pick one label at random, based on their relative probability	N/A: Assign to a random cluster, sample by proximity	N/A: Sample a single label for each position, according to posterior prob.	Gibbs sampling: Use one position for the motif, by sampling from the match scores	Average of those points assigned to label(a sample)

Figura 15.12: Comparación de agrupamiento, HMM y descubrimiento de motivos con respecto al algoritmo de minimización de expectativas (EM).

EM está garantizado para converger y garantizado para encontrar la mejor respuesta posible, al menos desde un punto de vista algorítmico. El problema notable con esta solución es que la existencia de máximos locales de densidad de probabilidad puede impedir que el algoritmo converja al máximo global. Un enfoque que puede evitar esta complicación es intentar múltiples inicializaciones para determinar mejor el panorama de probabilidades.

## Las limitaciones del algoritmo K-Means

El algoritmo k-means tiene algunas limitaciones que son importantes tener en cuenta a la hora de usarlo y antes de elegirlo. En primer lugar, requiere de una métrica. Por ejemplo, no podemos usar el algoritmo k-means en un conjunto de palabras ya que no tendríamos ninguna métrica.

La segunda limitación principal del algoritmo k-means es su sensibilidad al ruido. Una forma de tratar de reducir el ruido es ejecutar un análisis de componentes principales de antemano. Otra forma es ponderar cada variable para dar menos peso a las variables afectadas por el ruido significativo: los pesos se calcularán dinámicamente en cada iteración del algoritmo K-medias [3].

La tercera limitación es que la elección de los centros iniciales puede influir en los resultados. Existen heurísticas para seleccionar los centros de clúster iniciales, pero ninguno de ellos es perfecto.

Por último, necesitamos conocer a priori el número de clases. Como hemos visto, hay formas de sortear este problema, esencialmente ejecutando varias veces el algoritmo variando k o usando la regla general  $(k \approx \sqrt{n/2})$  si nos falta en el lado computacional. [es.wikipedia.org/wiki/Determining\\_the\\_number\\_of\\_clusters\\_in\\_a\\_data\\_set](https://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set) resume bien las diferentes técnicas para seleccionar el número de clústeres. La agrupación jerárquica proporciona un enfoque práctico para elegir el número de clústeres.

## Clustering jerárquico

Si bien la agrupación discutida hasta ahora a menudo proporciona información valiosa sobre la naturaleza de varios datos, generalmente pasan por alto un componente esencial de los datos biológicos, a saber, la idea de que la similitud podría existir en múltiples niveles. Para ser más precisos, la similitud es una propiedad intrínsecamente jerárquica, y este aspecto no se aborda en los algoritmos de agrupamiento discutidos hasta ahora. La agrupación jerárquica aborda específicamente esto de una manera muy

simple, y es quizás el algoritmo más utilizado para los datos de expresión. Como se ilustra en la figura 15.13, se implementa de la siguiente manera:

1. Inicialización: Inicializar una lista que contenga cada punto como un clúster independiente.
2. Iteración: Cree un nuevo clúster que contenga los dos clústeres más cercanos de la lista. Agregar este nuevo clúster a la lista y eliminar los dos grupos constitutivos de la lista.

Un beneficio clave de usar clústeres jerárquicos y hacer un seguimiento de los tiempos en los que fusionamos ciertos clústeres es que podemos crear una estructura de árbol que detalla los momentos en que nos unimos a cada clúster, como se puede ver en la figura 15.13. Así, para obtener una serie de clusters que se ajuste a tu problema, simplemente cortas a un nivel de corte de tu elección como en la figura 15.13 y eso te da el número de racimos correspondientes a ese nivel de corte. No obstante, tenga en cuenta que un escollo potencial con este enfoque es que en ciertos niveles de corte, los elementos que están bastante cerca en el espacio (como e y b en la figura 15.13), podrían no estar en el mismo clúster.

Por supuesto, se requiere un método para determinar distancias entre clústeres. La métrica particular utilizada varía según el contexto, pero (como puede verse en la figura 15.14) algunas implementaciones comunes incluyen la máxima,

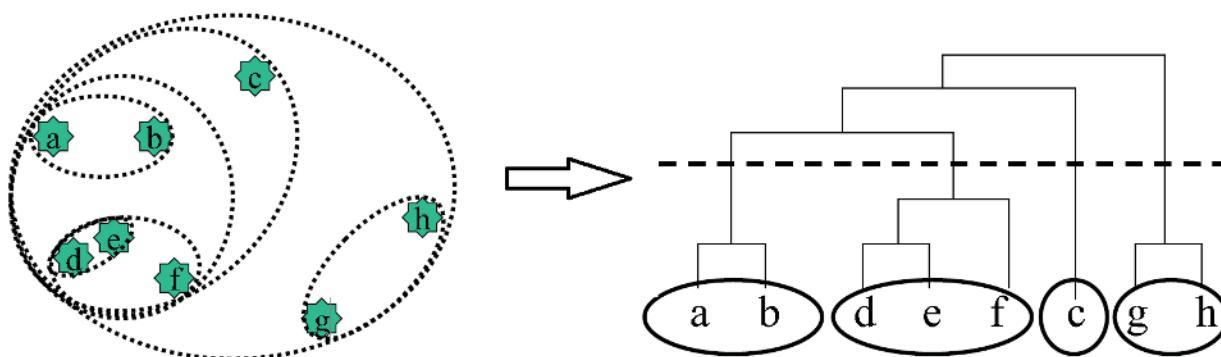


Figura 15.13: Clustering jerárquico © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

las distancias mínimas y medias entre los conglomerados constituyentes, y la distancia entre los centroides de los clústeres.

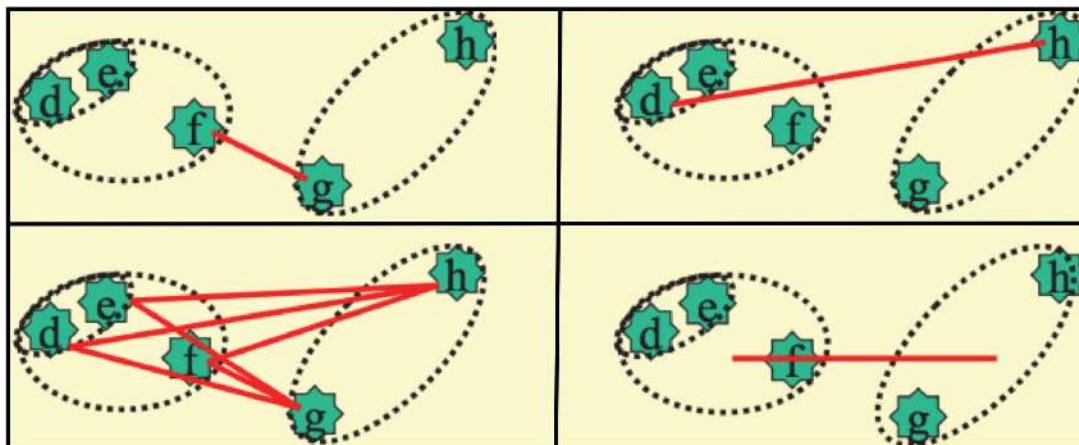


Figura 15.14: Métricas de Distancia para Clustering Jerárquico. En sentido horario desde arriba a la izquierda: mínima, máxima, distancia media y distancia centroide. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Señaló que a la hora de elegir los clusters más cercanos, calcular todas las distancias por pares consume mucho tiempo y espacio, por lo que se necesita un mejor esquema. Una forma posible de hacer esto es: 1) definir algunos cuadros delimitadores que dividen el espacio de entidades en varios subespacios 2) calcular distancias por pares dentro de cada cuadro 3) desplazar el límite de las cajas en diferentes direcciones y recalcular distancias por pares 4) elegir el par más cercano en función de los resultados en todas las iteraciones.

## Evaluar el desempeño del clúster

La validez de un agrupamiento particular se puede evaluar de varias maneras diferentes. La sobrerepresentación de un grupo conocido de genes en un clúster, o, más generalmente, la correlación entre el agrupamiento y las asociaciones biológicas confirmadas, es un buen indicador de validez y significación. Sin embargo, si aún no se dispone de datos biológicos, existen formas de evaluar la validez utilizando estadísticas. Por ejemplo, los clústeres robustos aparecerán a partir de la agrupación incluso cuando solo se utilicen subconjuntos del total de datos disponibles para generar clústeres. Además, la significancia estadística de un agrupamiento se puede determinar calculando la probabilidad de que una distribución particular se haya obtenido aleatoriamente para cada clúster. Este cálculo utiliza variaciones en la distribución hipergeométrica. Como se puede ver en la figura 15.15, podemos hacer esto calculando la probabilidad de que tengamos más de  $r^+$  cuando seleccionamos  $k$  elementos de un total de  $N$  elementos. [http://en.Wikipedia.org/wiki/Cluster...tering\\_results](http://en.Wikipedia.org/wiki/Cluster...tering_results) da varias fórmulas para evaluar la calidad de la agrupación.

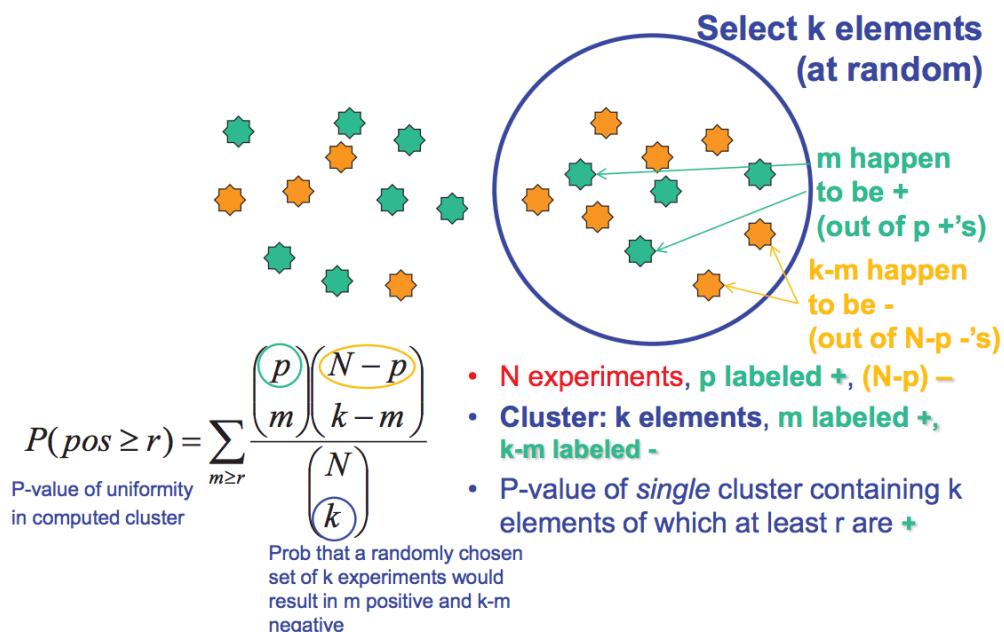


Figura 15.15: Cálculo de probabilidad de que tengas más de  $r^+$  en un clúster seleccionado aleatoriamente. © fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

This page titled [15.3: Algoritmos de Clustering](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [15.3: Clustering Algorithms](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.4: Direcciones actuales de investigación

Los problemas más significativos asociados con la agrupación ahora están asociados con el escalado de algoritmos existentes limpiamente con dos atributos: tamaño y dimensionalidad. Para hacer frente a conjuntos de datos cada vez más grandes, se han desarrollado algoritmos como la agrupación de dosel, en los que los conjuntos de datos se agrupan de manera general de una manera destinada a preprocessar los datos, después de lo cual se aplican algoritmos de agrupamiento estándar (por ejemplo, k-medias) para subdividir los diversos clústeres. El aumento de la dimensionalidad es un problema mucho más frustrante, e intentar remediarlo generalmente implica un proceso de dos etapas en el que los subespacios relevantes apropiados se identifican primero mediante transformaciones apropiadas en el espacio original y luego se someten a algoritmos de agrupamiento estándar.

---

This page titled [15.4: Direcciones actuales de investigación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [15.4: Current Research Directions](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.5: Lectura adicional

- Trevor Hastie, Robert Tibshirani y Jerome Friedman. Los elementos del aprendizaje estadístico: minería de datos, inferencia y predicción. Segunda Edición, febrero de 2009. Encontrado en línea en [www-stat.stanford.edu/~tibs/ElemStatLearn/download.html](http://www-stat.stanford.edu/~tibs/ElemStatLearn/download.html)
- Recetas numéricas: El arte de la computación científica (3a ed.). Nueva York: Cambridge University Press.
- McLachlan, G.J. y Basford, K.E. (1988) "Modelos de mezcla: inferencia y aplicaciones al agrupamiento", Marcel Dekker.
- Bezdek, J. C., Ehrlich, R., Completo, W. (1984). FCM: El algoritmo de agrupamiento difuso c-means. Computadoras y Geociencias, 10 (2), 191-203.
- [NLP.Stanford.edu/IR-Libro/html...stering-1.html](http://NLP.Stanford.edu/IR-Libro/html...stering-1.html)
- [compbio.uthsc.edu/microarray/lecture1.html](http://compbio.uthsc.edu/microarray/lecture1.html)

This page titled [15.5: Lectura adicional](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [15.5: Further Reading](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.6: Recursos

- Cluster 3.0: software de agrupamiento de código abierto que implementa los métodos de agrupamiento más utilizados para el análisis de datos de expresión génica.
- MATLAB: K-means clustering: <http://www.mathworks.com/help/stats/kmeans.html>; Fuzzy C- significa agrupación: <http://www.mathworks.com/help/fuzzy/fcm.html>; Clustering jerárquico: <http://www.mathworks.com/help/stats/linkage.html>
- Orange es una suite de software de minería de datos gratuita (consulte el módulo ORNGClustering para secuencias de comandos en Python): <http://bonsai.hgc.jp/~mdehoon/software.htm>
- R (ver Análisis de conglomerados y modelos de mezcla finita)
- CLÚSTER SAS

This page titled [15.6: Recursos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [15.6: Resources](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 15.7: Qué hemos aprendido, Bibliografía

Para resumir, en este capítulo hemos visto que:

- En clustering, identificamos estructura en datos no etiquetados. Por ejemplo, podríamos usar clustering para identificar grupos de genes que muestran perfiles de expresión similares.
  - — Algoritmos de agrupación de particiones, construyen clústeres no superpuestos de manera que cada elemento se asigna exactamente a un clúster. Ejemplo: k-means
  - — Los algoritmos de agrupamiento aglomerativo construyen un conjunto jerárquico de clústeres anidados, lo que indica la relación entre clústeres. Ejemplo: agrupación jerárquica
  - — Mediante el uso de algoritmos de agrupamiento, podemos revelar la estructura oculta de una matriz de expresión génica, lo que nos da pistas valiosas para comprender el mecanismo de enfermedades complicadas y categorizar diferentes enfermedades
- En la clasificación, dividimos los datos en etiquetas conocidas. Por ejemplo, podríamos construir un clasificador para dividir un conjunto de muestras tumorales en aquellas que probablemente respondan a un medicamento dado y aquellas que es poco probable que respondan a un medicamento dado en función de sus perfiles de expresión génica. Nos centraremos en la clasificación en el próximo capítulo.

### Bibliografía

[1] [es.wikipedia.org/wiki/Archivo:HeatMap.png](https://es.wikipedia.org/wiki/Archivo:HeatMap.png).

[2] <http://genome.ucsc.edu/ENCODE/>.

[3] J.Z. Huang, M.K. Ng, Hongqiang Rong y Zichen Li. Ponderación variable automatizada en clústeres tipo k-medias. Análisis de patrones e inteligencia artificial, IEEE Transactions on, 27 (5) :657 —668, mayo de 2005.

[4] Christopher A. Maher, Chandan Kumar-Sinha, Xuhong Cao, Shanker Kalyana-Sundaram, Bo Han, Xiao- jun Jing, Lee Sam, Terrence Barrette, Nallasivam Palanisamy y Arul M. Chinnaiyan. Secuenciación de transcriptomas para detectar fusiones génicas en cáncer. Naturaleza, 458 (7234) :97—101, 05 de mar de 2009.

This page titled [15.7: Qué hemos aprendido, Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [15.7: What Have We Learned, Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 16: Regulación Génica II - Clasificación

[16.1: Introducción](#)

[16.2: Clasificación—Técnicas Bayesianas](#)

[16.3: Máquinas vectoriales de soporte de clasificación](#)

[16.4: Clasificación Tumoral con SVMs](#)

[16.5: Aprendizaje Semi-Supervisado](#)

[16.6: Lectura adicional, Recursos, Bibliografía](#)

---

This page titled [16: Regulación Génica II - Clasificación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 16.1: Introducción

En el capítulo anterior analizamos el clustering, que proporciona una herramienta para analizar datos sin ningún conocimiento previo de la estructura subyacente. Como mencionamos antes, este es un ejemplo de aprendizaje “no supervisado”. Este capítulo trata sobre el aprendizaje supervisado, en el que podemos utilizar datos preclasificados para construir un modelo mediante el cual clasificar más puntos de datos. De esta manera, utilizaremos la estructura existente y conocida para desarrollar reglas para identificar y agrupar más información.

Hay dos formas de hacer clasificación. Las dos formas son análogas a las dos formas en las que realizamos el descubrimiento de motivos: HMM, que es un modelo generativo que nos permite describir realmente la probabilidad de que una designación particular sea válida, y CRF, que es un método discriminativo que permite distinguir entre objetos en un contexto. Existe una dicotomía entre enfoques generativos y discriminativos. Utilizaremos un enfoque bayesiano para clasificar proteínas mitocondriales, y SVM para clasificar muestras tumorales.

En esta conferencia veremos dos nuevos algoritmos: un clasificador generativo, Nave Bayes, y un clasificador discriminativo, Máquinas de vectores de soporte (SVM). Discutiremos las aplicaciones biológicas de cada uno de estos modelos, específicamente en el uso de clasificadores Nave Bayes para predecir proteínas mitocondriales en todo el genoma y el uso de SVM para la clasificación del cáncer basado en el monitoreo de la expresión génica por microarrays de ADN. También se discutirán las características sobresalientes de ambas técnicas y las advertencias del uso de cada técnica.

Al igual que con la agrupación, la clasificación (y más generalmente el aprendizaje supervisado) surgió de los esfuerzos en Inteligencia Artificial y Machine Learning. Además, gran parte de la infraestructura motivadora para la clasificación ya había sido desarrollada por teóricos de la probabilidad antes de la llegada de la IA o la ML.

---

This page titled [16.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [16.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 16.2: Clasificación—Técnicas Bayesanas

Considerar el problema de identificar proteínas mitocondriales. Si observamos el genoma humano, ¿cómo determinamos qué proteínas están involucradas en los procesos mitocondriales, o más generalmente qué proteínas están dirigidas a las mitocondrias?

<sup>1</sup> Esto es particularmente útil porque si conocemos las proteínas mitocondriales, podemos estudiar cómo estas proteínas median los procesos de la enfermedad y las funciones metabólicas. El método de clasificación que veremos considera 7 características para todas las proteínas humanas:

1. señal de focalización
2. dominios proteicos
3. coexpresión
4. espectrometría de masas
5. homología de secuencia
6. inducción
7. motivos

Nuestro enfoque general será determinar cómo se distribuyen estas características tanto para proteínas mitocondriales como no mitocondriales. Entonces, dada una nueva proteína, podemos aplicar análisis probabilísticos a estas siete características para decidir en qué clase cae más probablemente.

### Funciones individuales y regla de Bayes

Al principio, centrémonos en una característica. Primero debemos asumir que existe una distribución dependiente de clase para las entidades. Primero debemos derivar esta distribución a partir de datos reales. Lo segundo que necesitamos es la posibilidad a priori de dibujar una muestra de clase particular antes de mirar los datos. La posibilidad de obtener una clase en particular es simplemente el tamaño relativo de la clase. Una vez que tenemos estas probabilidades, podemos usar la regla de Bayes para obtener la probabilidad de que una muestra esté en una clase particular dados los datos (esto se llama el posterior). Tenemos probabilidades generativas hacia adelante y usamos las reglas de Bayes para realizar la inferencia hacia atrás. Tenga en cuenta que no basta con considerar la probabilidad de que la entidad se haya extraído de cada distribución dependiente de clase, porque si supiéramos a priori que una clase (digamos la clase A) es mucho más común que la otra, entonces debería tomar evidencia abrumadora de que la entidad se extrajo de la distribución de la clase B para nosotros para creer que la característica era efectivamente de la clase B. La forma correcta de encontrar lo que necesitamos con base tanto en evidencia como en conocimiento previo es usar la Regla de Bayes:

$$P(\text{Class} \mid \text{feature}) = \left( \frac{P(\text{ feature} \mid \text{ Class }) P(\text{ Class })}{P(\text{ feature })} \right)$$

- Posterior:  $P(\text{Class}|\text{Feature})$
- Previo:  $P(\text{Clase})$
- Probabilidad:  $P(\text{Característica}|\text{Clase})$

Esta fórmula nos da exactamente la conexión que necesitamos para convertir las probabilidades de características conocidas en probabilidades de clase para nuestro algoritmo de clasificación. Nos permite integrar tanto la probabilidad que derivamos de nuestras observaciones como nuestro conocimiento previo sobre lo común que es algo. En el caso del ADNmt, por ejemplo, podemos estimar que el ADN mitocondrial constituye algo así como 1500/21000 (es decir, menos del 10%) del genoma humano. Por lo tanto, aplicando la regla de Bayes, nuestro clasificador solo debe clasificar un gen como mitocondrial si existe una probabilidad muy fuerte basada en las características observadas, ya que la probabilidad previa de que algún gen sea mitocondrial es tan baja.

Con esta regla, ahora podemos formar una regla de máxima verosimilitud para predecir una clase de objetos basada en una entidad observada. Queremos elegir la clase que tenga la probabilidad más alta dada la característica observada, por lo que elegiremos Clase1 en lugar de Clase2 si:

$$\left( \frac{P(\text{ feature } | \text{ Class 1})P(\text{ Class 1})}{P(\text{ feature })} \right) > \left( \frac{P(\text{ feature } | \text{ Class 2})P(\text{ Class 2})}{P(\text{ feature })} \right)$$

Observe que P (entidad) aparece en ambos lados, por lo que podemos cancelarlo por completo, y simplemente elegir la clase con el valor más alto de P (Feature|Class) P (Class).

Otra forma de ver esto es como una función discriminante: Al reorganizar las fórmulas anteriores y tomar el logaritmo, debemos seleccionar Class1 en lugar de Clase2 precisamente cuando

$$\log \left( \frac{P(X | \text{Class1})P(\text{ Class 1})}{P(X | \text{ Class 2})P(\text{ Class 2})} \right) > 0$$

En este caso el uso de logaritmos aporta distintas ventajas:

1. Estabilidad numérica
2. Matemáticas más fáciles (es más fácil agregar los términos expandidos que multiplicarlos)
3. Aumentan monótonamente los discriminadores.

Esta función discriminante no capta las penalizaciones asociadas a la clasificación errónea (es decir, es una clasificación más perjudicial que otra). En este caso, esencialmente estamos minimizando el número de clasificaciones erróneas que hacemos en general, pero no asignando sanciones a clasificaciones erróneas individuales. A partir de ejemplos discutidos en clase y en el conjunto de problemas -si estamos tratando de clasificar a un paciente como que tiene cáncer o no, se podría argumentar que es mucho más dañino clasificar erróneamente a un paciente como sano si tiene cáncer que clasificar erróneamente a un paciente como que tiene cáncer si está sano. En el primer caso, el paciente no será atendido y tendría más probabilidades de morir, mientras que el segundo error implica dolor emocional pero no mayor probabilidad de pérdida de vidas. Para formalizar la penalización de clasificación errónea definimos algo llamado una función de pérdida, Lkf, que asigna una pérdida a la clasificación errónea de un objeto como clase j cuando la clase verdadera es la clase k (un ejemplo específico de una función de pérdida se vio en el conjunto de problemas 2).

## Recopilación de datos

El precedente nos dice cómo manejar las predicciones si ya conocemos las probabilidades exactas correspondientes a cada clase. Si queremos clasificar las proteínas mitocondriales en función de la característica X, aún necesitamos formas de determinar las probabilidades P (mito), P (no mito), P (x|mito) y P (x|no mito). Para ello, necesitamos un conjunto de entrenamiento: un conjunto de datos que ya están clasificados que nuestro algoritmo puede utilizar para aprender las distribuciones correspondientes a cada clase. Un **conjunto de entrenamiento de alta calidad** (uno que es a la vez grande e imparcial) es la parte más importante de cualquier clasificador. Una pregunta importante en este punto es, ¿cuántos datos necesitamos sobre genes conocidos para construir un buen clasificador para genes desconocidos? Esta es una pregunta dura cuya respuesta no se conoce del todo. Sin embargo, hay algunos métodos simples que pueden darnos una buena estimación: cuando tenemos un conjunto fijo de datos de entrenamiento, podemos mantener un conjunto de holdout que no usamos para nuestro algoritmo, y en su lugar usar esos puntos de datos (conocidos) para probar la precisión de nuestro algoritmo cuando intentamos clasificarlos. Al probar diferentes tamaños de entrenamiento versus conjunto de holdout, podemos verificar la curva de precisión de nuestro algoritmo. En términos generales, tenemos suficientes datos de entrenamiento cuando vemos que la curva de precisión se aplana a medida que aumentamos la cantidad de datos de entrenamiento (esto indica que es probable que los datos adicionales den solo una ligera mejora marginal). El conjunto de retención también se llama el conjunto de prueba, porque nos permite probar la potencia de generalización de nuestro clasificador.

Supongamos que ya hemos recopilado nuestros datos de capacitación, sin embargo, ¿cómo debemos modelar P (X|Class)? Hay muchas posibilidades. Una es usar el mismo enfoque que hicimos con la agrupación en la última conferencia y modelar la característica como gaussiana luego podemos seguir el principio de máxima verosimilitud para encontrar el mejor centro y varianza. La utilizada en el estudio mitocondrial es una estimación de densidad simple: para cada característica, dividir el rango de posibilidades en un conjunto de bins (digamos, cinco bins por entidad). Luego usamos los datos dados para estimar la probabilidad de que una entidad caiga en cada bin para una clase dada. El principio detrás de esto es nuevamente la máxima verosimilitud, pero para una distribución multinomial más que a una gaussiana. Podemos optar por discretizar una distribución continua, ya que estimar una distribución continua puede ser compleja.

Hay un problema con esta estrategia: ¿y si uno de los contenedores tiene cero muestras en ella? Una probabilidad de cero anulará todo lo demás en nuestras fórmulas, de modo que en lugar de pensar que este bin es simplemente improbable, nuestro clasificador va a creer que es imposible. Hay muchas soluciones posibles, pero la que se toma aquí es aplicar la Corrección de Laplace: agregar una pequeña cantidad (digamos, un elemento) a cada bin, para dibujar estimaciones de probabilidad ligeramente hacia uniformes y dar cuenta del hecho de que (en la mayoría de los casos) ninguno de los bins es realmente imposible. Otra forma de evitar tener que aplicar la corrección es elegir bins que no sean demasiado pequeños para que los bins no tengan cero muestras en ellos en la práctica. Si tienes muchos muchos puntos, puedes tener más bins, pero corres el riesgo de sobreajustar tus datos de entrenamiento.

## Estimación de Priors

Ahora tenemos un método para aproximar la distribución de entidades para una clase dada, pero aún necesitamos conocer la probabilidad relativa de las clases mismas. Hay tres enfoques generales:

1. Estimar los antecedentes contando la frecuencia relativa de cada clase en los datos de entrenamiento. Esto es propenso al sesgo, sin embargo, ya que los datos disponibles a menudo están sesgados de manera desproporcionada hacia clases menos comunes (ya que a menudo son objeto de estudios especiales). Si tenemos una muestra de alta calidad (representativa) para nuestros datos de capacitación, sin embargo, esto funciona muy bien.
2. Estimación a partir del conocimiento experto: puede haber estimaciones previas obtenidas por otros métodos independientes de nuestros datos de entrenamiento, que luego podemos usar como una primera aproximación en nuestras propias predicciones. Es decir, podrías preguntar a los expertos cuál es el porcentaje de proteínas mitocondriales.
3. Supongamos que todas las clases son igualmente probables que normalmente haríamos esto si no tenemos ninguna información sobre las frecuencias verdaderas. Esto es efectivamente lo que hacemos cuando usamos el principio de máxima verosimilitud: nuestro algoritmo de agrupamiento estaba esencialmente usando el análisis bayesiano bajo el supuesto de que todos los antecedentes son iguales. Esto es en realidad una suposición fuerte, pero cuando no tienes otros datos, esto es lo mejor que puedes hacer.

Para clasificar el ADN mitocondrial utilizamos el método (2), ya que ya se conocían algunas estimaciones sobre las proporciones de ADNmt. Pero hay una complicación: hay más de 1 características.

## Múltiples características y Naive Bayes

Al clasificar el ADN mitocondrial, estábamos viendo 7 características y no solo una. Para usar los métodos anteriores con múltiples características, necesitaríamos no solo un bin para cada rango de características individual, sino uno para cada combinación de características si observamos dos entidades con cinco rangos cada una, eso ya es 25 bins. Las siete características nos dan casi 80,000 contenedores y podemos esperar que la mayoría de esos contenedores estén vacíos simplemente porque no tenemos suficientes datos de capacitación para llenarlos todos. Esto causaría problemas porque los ceros provocan infinitos cambios en las probabilidades de estar en una clase. Claramente este enfoque no escalará bien ya que agregamos más características, por lo que necesitamos estimar las probabilidades combinadas de una mejor manera.

La solución que usaremos es asumir que las entidades son independientes, es decir, que una vez que conocemos la clase, la distribución de probabilidad de cualquier entidad no se ve afectada por los valores de las otras entidades. Esta es la Asunción Nave Bayes, y casi siempre es falsa, pero a menudo se usa de todos modos por las razones combinadas de que es muy fácil de manipular matemáticamente y muchas veces se acerca lo suficientemente a la verdad que da una aproximación razonable. (Tenga en cuenta que esta suposición no dice que todas las entidades sean independientes: si miramos el modelo general, puede haber conexiones fuertes entre diferentes entidades, pero la suposición dice que esas conexiones están divididas por las diferentes clases, y que dentro de cada clase individual no hay más dependencias.) Además, si sabes que algunas características están acopladas, podrías aprender la distribución conjunta en solo algunos pares de las entidades.

Una vez que asumimos la independencia, la probabilidad de características combinadas es simplemente el producto de las probabilidades individuales asociadas a cada característica. Así que ahora tenemos:

$$P(f_1, f_2, K, f_N | Class) = P(f_1 | Class)P(f_2 | Class)KP(f_N | Class)$$

Donde  $f_1$  representa la entidad 1. Del mismo modo, la función discriminante se puede cambiar a la multiplicación de las probabilidades previas:

$$G(f_1, f_2, K, f_N) = \log \left( \frac{\prod P(f_i | \text{Class 1}) P(\text{Class 1})}{\prod P(f_i | \text{Class 2}) P(\text{Class 2})} \right)$$

## Prueba de un clasificador

Un clasificador siempre debe probarse sobre datos no contenidos en su conjunto de entrenamiento. Podemos imaginar en el peor de los casos un algoritmo que acaba de memorizar sus datos de entrenamiento y se comportó aleatoriamente en cualquier otra cosa un clasificador que hiciera esto funcionaría perfectamente en sus datos de entrenamiento, pero que no indica nada sobre su rendimiento real en nuevas entradas. Es por ello que es importante utilizar una prueba, o holdout, establecida como se mencionó anteriormente. Sin embargo, una tasa de error simple no encapsula todas las posibles consecuencias de un error. Para un clasificador binario simple (un objeto está en o no en una sola clase de destino), existen los siguientes para los tipos de errores:

1. Verdadero positivo (TP)
2. Verdadero negativo (TN)
3. Falso positivo (FP)
4. Falso negativo (FN)

La frecuencia de estos errores se puede encapsular en métricas de rendimiento de un clasificador que se definen como,

1. Sensibilidad ¿qué fracción de objetos que están en una clase se etiquetan correctamente como esa clase? Es decir, ¿qué fracción tiene verdaderos resultados positivos? Alta sensibilidad significa que es muy probable que los elementos de una clase sean etiquetados como esa clase. Baja sensibilidad significa que hay demasiados falsos negativos.
2. Especificidad ¿qué fracción de objetos que no están en una clase están correctamente etiquetados como no estar en esa clase? Es decir, ¿qué fracción tiene verdaderos resultados negativos? Alta especificidad significa que los elementos etiquetados como pertenecientes a una clase son muy propensos a pertenecer realmente a ella. Baja especificidad significa que hay demasiados falsos positivos.

En la mayoría de los algoritmos existe un **compromiso entre sensibilidad y especificidad**. Por ejemplo, podemos alcanzar una sensibilidad del 100% etiquetando todo como perteneciente a la clase objetivo, pero tendremos una especificidad del 0%, por lo que esto no es útil. Generalmente, la mayoría de los algoritmos tienen algún límite de probabilidad que utilizan para decidir si etiquetar un objeto como perteneciente a una clase (por ejemplo, nuestra función discriminante anterior). Elevar ese umbral aumenta la especificidad pero disminuye la sensibilidad, y disminuir el umbral hace lo contrario. El algoritmo MAESTRO para clasificar proteínas mitocondriales (descrito en esta conferencia) alcanza 99% de especificidad y 71% de sensibilidad.

## Clasificación de Proteína Mitocondrial MAESTRO

Encuentran una distribución dependiente de clase para cada característica al crear varios bins y evaluar la porción de proteínas mitocondriales y no mitocondriales en cada bin. Esto permite evaluar la **utilidad de cada característica** en la clasificación. Terminas con un montón de clasificadores de fuerza media, pero cuando los combinás entre sí, ojalá termines con un clasificador más fuerte. Calvo et al. [1] buscaron construir predicciones de alta calidad de proteínas humanas localizadas en la mitocondria mediante la generación e integración de conjuntos de datos que proporcionen pistas complementarias sobre la localización mitocondrial. Específicamente, para cada producto genético humano p, asignan una puntuación  $s_i(p)$ , usando cada uno de los siguientes siete conjuntos de datos a escala de genoma, puntuación de señal dirigida, puntaje de dominio de proteína, puntaje de motivo cis, puntaje de homología de levadura, puntaje de ascendencia, puntaje de coexpresión y puntaje de inducción (detalles de cada uno de los significado y contenido de cada uno de estos conjuntos de datos se pueden encontrar en el manuscrito). Cada una de estas puntuaciones  $s_1 - S_7$  se puede utilizar individualmente como un predictor débil de localización mitocondrial en todo el genoma. El desempeño de cada método se evaluó utilizando grandes conjuntos de entrenamiento curados estándar de oro - 654 proteínas mitocondriales T<sub>mito</sub> mantenidas por la base de datos MiToP2 y 2,847 proteínas no mitocondriales T<sub>mito</sub> anotadas para localizarse en otros compartimentos celulares. Para mejorar la precisión de la predicción, los autores integraron estos ocho enfoques utilizando un clasificador Bayes de nave que se implementó como un programa llamado MAESTRO. Así podemos tomar varios clasificadores débiles, y combinarlos para obtener un clasificador más fuerte.

Cuando se aplicó MAESTRO a través del proteoma humano, se predijeron 1451 proteínas como proteínas mitocondriales y se hicieron 450 nuevas predicciones de proteínas. Como se mencionó en el apartado anterior El algoritmo MAESTRO logra una

especificidad del 99% y una sensibilidad del 71% para la clasificación de las proteínas mitocondriales, lo que sugiere que incluso con el supuesto de independencia característica, las técnicas de clasificación de Nave Bayes pueden resultar extremadamente poderosas para grandes dimensiones (i.e. genome-wide) clasificación a escala.

---

<sup>1</sup> Las mitocondrias son la maquinaria productora de energía de la célula. Muy temprano en la vida, las mitocondrias fueron engullidas por el predecesor de los eucariotas modernos, y ahora, tenemos diferentes compartimentos en nuestras células. Entonces la mitoconria tiene su propio genoma, pero está muy agotada de su propio genoma ancestral; solo quedan unos 11 genes. Pero hay cientos de genes que hacen que las mitocondrias funcionen, y estas proteínas son codificadas por genes transcritos en el núcleo, para luego transportarse a las mitocondrias. Entonces el objetivo es averiguar qué proteínas codificadas en el genoma están dirigidas a las mitocondrias. Esto es importante porque existen muchas enfermedades asociadas a la mitoconria, como el envejecimiento.

---

This page titled 16.2: Clasificación—Técnicas Bayesianas is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [16.2: Classification--Bayesian Techniques](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 16.3: Máquinas vectoriales de soporte de clasificación

En la sección anterior se analizó el uso de modelos probabilísticos (o generativos) para la clasificación, esta sección analiza el uso de técnicas discriminativas en esencia, ¿podemos ejecutar nuestros datos a través de una función para determinar su estructura? Tales técnicas discriminativas evitan el costo inherente que implica los modelos generativos que podrían requerir más información de la que realmente es necesaria.

Las técnicas de máquina vectorial de soporte implican esencialmente dibujar un vector que es perpendicular a la línea (hiperplano) que separa los datos de entrenamiento. El enfoque es que observamos los datos de entrenamiento para obtener un hiperplano de separación para que dos clases de datos se encuentren en diferentes lados del hiperplano. Hay, en general, muchos hiperplanos que pueden separar los datos, por lo que queremos dibujar el hiperplano que más separa los datos -deseamos elegir la línea que maximice la distancia desde el hiperplano a cualquier punto de datos. En otras palabras, el SVM es un clasificador de margen máximo. Se puede pensar que el hiperplano está rodeado con márgenes de igual tamaño a cada lado de la línea, sin puntos de datos dentro del margen en ninguno de los lados. Queremos trazar la línea que nos permita trazar el margen más grande. Tenga en cuenta que una vez que se determine la línea de separación y el margen, algunos puntos de datos estarán justo en el límite del margen. Estos son los puntos de datos que nos evitan ampliar más el margen, y así determinar la línea/margen. Tales puntos se llaman los vectores de soporte. Si agregamos nuevos puntos de datos fuera del margen o eliminamos puntos que no son vectores de soporte, no cambiaremos el margen máximo que podemos lograr con cualquier hiperplano.

Supongamos que el vector perpendicular al hiperplano es  $w$ , y que el hiperplano pasa por el punto  $(\frac{b}{|w|})$ . Entonces un punto  $x$  se clasifica como estar en la clase positiva si  $w \cdot x$  es mayor que  $b$ , y negativo en caso contrario. Se puede demostrar que el  $w$  óptimo, es decir, el hiperplano que logra el margen máximo, puede escribirse realmente como una combinación lineal de los vectores de datos  $\alpha_i * x_i$ . Entonces, para clasificar un nuevo punto de datos  $x$ , necesitamos tomar el producto punto de  $w$  con  $x$  para llegar a un escalar. Observe que este escalar,  $\alpha_i * (x_i * x)$  solo depende del producto punto entre  $x$  y los vectores de entrenamiento  $x_i$ s. Además, se puede demostrar que encontrar el hiperplano de margen máximo para un conjunto de puntos (de entrenamiento) equivale a maximizar un programa lineal donde la función objetiva solo depende del producto punto de los puntos de entrenamiento entre sí. Esto es bueno porque nos dice que la complejidad de resolver ese programa lineal es independiente de la dimensión de los puntos de datos. Si precalculamos los productos de puntos por pares de los vectores de entrenamiento, entonces no importa cuál es la dimensionalidad de los datos con respecto al tiempo de ejecución de resolver el programa lineal.

### Núcleos

Vemos que los SVM dependen únicamente del producto punto de los vectores. Entonces, si llamamos a nuestra transformación  $\phi(v)$ , para dos vectores solo nos importa el valor de  $\phi(v_1) \cdot \phi(v_2)$ . El truco para usar kernels es darnos cuenta de que para ciertas transformaciones  $\phi$ , existe una función  $K(v_1, v_2)$ , tal que:

$$K(v_1, v_2) = \phi(v_1) \cdot \phi(v_2)$$

En la relación anterior, el lado derecho es el producto punto de vectores con dimensión muy alta, pero el lado izquierdo es función de dos vectores con dimensión inferior. En nuestro ejemplo anterior de mapeo  $x \rightarrow (x, y = x^2)$ , obtenemos

$$K(x_1, x_2) = (x_1 x_2^2) \cdot (x_2, x_2^2) = x_1 x_2 + (x_1 x_2)^2$$

Ahora no aplicamos realmente la transformación  $\phi$ , podemos hacer todos nuestros cálculos en el espacio dimensional inferior, pero obtenemos todo el poder de usar una dimensión superior.

Los núcleos de ejemplo son los siguientes:

1. Núcleo lineal:  $K(v_1, v_2) = v_1 \cdot v_2$  que representa el mapeo trivial de  $\phi(x) = x$
2. Núcleo polinómico:  $K(v_1, v_2) = (1 + v_1 \cdot v_2)^n$  el cual se utilizó en el ejemplo anterior con  $n = 2$ .
3. Núcleo de base radial:  $K(v_1, v_2) = \exp(-\beta|v_1 - v_2|^2)$  Esta transformación es en realidad de un punto  $v_1$  a una función (que puede pensarse como un punto en el espacio Hilbert) en un espacio infinito-dimensional. Entonces, lo que realmente estábamos haciendo es transformar nuestro conjunto de entrenamiento en funciones, y combinar el para obtener un límite de decisión. Las funciones son gaussianas centradas en los puntos de entrada.
4. Núcleo sigmoide:  $K(v_1, v_2) = \tanh[\beta(v_1^T v_2 + r)]$  Los núcleos sigmoides han sido populares para su uso en SVM debido a su origen en redes neuronales (por ejemplo, las funciones del núcleo sigmoide son equivalentes a redes neuronales perceptrón

de dos niveles). Se ha señalado en trabajos anteriores (Vapnik 1995) que la matriz del núcleo puede no ser positiva semidefinida para ciertos valores de los parámetros  $\mu$  y  $r$ . Sin embargo, el núcleo sigmoide se ha utilizado en aplicaciones prácticas [2].

Aquí hay un ejemplo específico de una función del kernel. Considere las dos clases de datos unidimensionales:

$\{-5, -4, -3, 3, 4, 5\}$  y  $\{-2, -1, 0, 1, 2\}$

Estos datos claramente no son separables linealmente, y el mejor límite de separación que podemos encontrar podría ser  $x > -2.5$ . Ahora considere aplicar la transformación. Los datos ahora se pueden escribir como nuevos pares,

$\{-5, -4, -3, 3, 4, 5\} \rightarrow \{(-5, 25), (-4, 16), (-3, 9), (3, 9), (4, 16), (5, 25)\}$

y

$\{-2, -1, 0, 1, 2\} \rightarrow \{(-2, -4), (-1, 1), (0, 0), (1, 1), (2, 4)\}$

Estos datos son separables por la regla  $y > 6.5$ , y en general entre más dimensiones transformamos los datos, más separables se vuelven.

Una forma alternativa de pensar sobre este problema es transformar el clasificador de nuevo en el espacio original de baja dimensión. En este ejemplo en particular, obtendríamos la regla  $x^2 < 6.5$ , que bisecionaría la recta numérica en dos puntos. En general, a mayor dimensionalidad del espacio en el que nos transformamos, más complicado es un clasificador que obtenemos cuando nos transformamos de nuevo al espacio original.

Una de las advertencias de transformar los datos de entrada usando un kernel es el riesgo de sobreajustar (o sobreclasificar) los datos. De manera más general, el SVM puede generar tantas dimensiones de vectores de características que no generaliza bien a otros datos. Para evitar el sobreajuste, la validación cruzada se utiliza normalmente para evaluar el ajuste proporcionado por cada conjunto de parámetros probado durante el proceso de búsqueda de cuadrícula o patrón. En el kernel de base radial, puede aumentar esencialmente el valor de  $\beta$  hasta que cada punto esté dentro de su propia región de clasificación (con lo que se derrotó el proceso de clasificación por completo). Los SVM generalmente evitan este problema de sobreajuste debido a que maximizan los márgenes entre puntos de datos.

Cuando se utilizan conjuntos de entrenamiento difíciles de separar, los SVM pueden incorporar un parámetro de costo  $C$ , para permitir cierta flexibilidad en la separación de las categorías. Este parámetro controla la compensación entre permitir errores de entrenamiento y forzar márgenes rígidos. De esta manera, puede crear un margen blando que permita algunas clasificaciones erróneas. Al aumentar el valor de  $C$  se incrementa el costo de clasificar erróneamente los puntos y se fuerza la creación de un modelo más preciso que puede no generalizarse bien.

¿Podemos usar cualquier función como nuestro kernel? La respuesta a esto es proporcionada por Mercers Condition que nos proporciona un criterio analítico para elegir un kernel aceptable. Mercers Condition establece que un kernel  $K(x, y)$  es un kernel válido si y solo si se mantiene lo siguiente Para cualquier  $g(x)$  tal que  $\int g(x)^2 dx$  sea finito, tenemos:

$$\iint K(x, y)g(x)g(y)dxdy \geq 0 [3]$$

En total, hemos definido discriminadores SVM y mostrado cómo realizar la clasificación con funciones de mapeo de kernel adecuadas que permiten realizar cálculos en menor dimensión mientras se está para capturar toda la información disponible en dimensiones más altas. En la siguiente sección se describe la aplicación de las SVM a la clasificación de tumores para el diagnóstico de cáncer.

---

This page titled [16.3: Máquinas vectoriales de soporte de clasificación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [16.3: Classification Support Vector Machines](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 16.4: Clasificación Tumoral con SVMs

Un enfoque genérico para clasificar dos tipos de leucemias agudas leucemia mieloide aguda (LMA) y leucemia linfoide aguda (LLA) fue presentado por Golub et al. [4]. Este enfoque se centró en abordar de manera efectiva tres cuestiones principales:

1. Si había genes cuyo patrón de expresión a predecir estaba fuertemente correlacionado con la distinción de clase (es decir, se pueden distinguir LLA y LMA)
2. Cómo usar una colección de muestras conocidas para crear un “predictor de clase” capaz de asignar una nueva muestra a una de dos clases
3. Cómo probar la validez de sus predictores de clase

Abordaron (1) utilizando una técnica de “análisis de vecindad” para establecer si las correlaciones observadas eran más fuertes de lo que se esperaría por casualidad. Este análisis mostró que aproximadamente 1100 genes estaban más altamente correlacionados con la distinción de clase AML-ALL de lo que se esperaría por casualidad. Para abordar (2) desarrollaron un procedimiento que utiliza un subconjunto fijo de “genes informativos” (elegidos en base a su correlación con la distinción de clase de AML y ALL) y hace una predicción basada en el nivel de expresión de estos genes en una nueva muestra. Cada gen informativo emite un “voto ponderado” para una de las clases, con el peso de cada voto dependiente del nivel de expresión en la nueva muestra y el grado de correlación de esos genes con la distinción de clase. Se suman los votos para determinar la clase ganadora. Abordar (3) y probar efectivamente su predictor primero probando mediante validación cruzada en el conjunto de datos inicial y luego evaluando su precisión en un conjunto independiente de muestras. Con base en sus pruebas, pudieron identificar 36 de las 38 muestras (¡que formaban parte de su set de entrenamiento!) y las 36 predicciones fueron clínicamente correctas. En el conjunto de pruebas independientes 29 de 34 muestras se predijeron fuertemente con una precisión del 100% y 5 no se predijeron.

Un enfoque SVM para este mismo problema de clasificación fue implementado por Mukherjee et al. [5]. La salida de SVM clásica es una designación de clase binaria. En esta aplicación particular es particularmente importante poder rechazar puntos para los que el clasificador no tiene la suficiente confianza. Por lo tanto, los autores introdujeron un intervalo de confianza en la salida del SVM que permite el rechazo de puntos con valores de confianza bajos. Como en el caso de Golub et al. [4] fue importante para los autores inferir qué genes son importantes para la clasificación. El SVM se entrenó en las 38 muestras del conjunto de entrenamiento y se probó en las 34 muestras en el conjunto de prueba independiente (exactamente en el caso de Golub et al.). Los resultados de los autores se resumen en la siguiente tabla (donde  $|d|$  corresponde al punto de corte para rechazo).

Esto da como resultado una mejora significativa con respecto a las técnicas reportadas anteriormente, lo que sugiere que las SVM juegan un papel importante en la clasificación de grandes conjuntos de datos (como los generados por experimentos de microarrays de ADN).

---

This page titled [16.4: Clasificación Tumoral con SVMs](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **16.4: Tumor Classification with SVMs** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 16.5: Aprendizaje Semi-Supervisado

En algunos escenarios tenemos un conjunto de datos con solo unos pocos puntos de datos etiquetados, una gran cantidad de puntos de datos sin etiquetar y estructura inherente a los datos. Este tipo de escenarios tanto de clustering como de clasificación no funcionan bien y se requiere un enfoque híbrido. Este enfoque semi-supervisado podría implicar el agrupamiento de datos primero seguido de la clasificación de los clusters generados.

---

This page titled [16.5: Aprendizaje Semi-Supervisado](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [16.5: Semi-Supervised Learning](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 16.6: Lectura adicional, Recursos, Bibliografía

### Lectura adicional

- Richard O. Duda, Peter E. Hart, David G. Stork (2001) Clasificación de patrones (2da edición), Wiley, Nueva York
- Consulte el capítulo anterior para más libros y artículos.

### Recursos

- Caja de herramientas de reconocimiento estadístico de patrones para Matlab.
- Consulte el capítulo anterior para más herramientas

### Bibliografía

[1] Calvo, S., Jain, M., Xie, X., Sheth, S.A., Chang, B., Goldberger, O.A., Spinaz- zola, A., Zeviani, M., Carr, S.A., y Mootha, V.K. (2006). Identificación sistemática de genes de enfermedades mitocondriales humanas a través de la genómica integradora. *Nat. Genet.* 38, 576582.

[2] Scholokopf, B., et al., 1997. Comparación de máquinas de vectores de soporte con núcleos gaussianos con clasificadores de función de base radial. *Transacciones IEEE en Procesamiento de Señal*.

[3] Christopher J.C. Burges. Un tutorial sobre máquinas vectoriales de soporte para reconocimiento de patrones. *Minería de Datos y Descubrimiento de Conocimiento*, 2:121 —167, 1998.

[4] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, y C. D. Bloomfield. Clasificación molecular del cáncer: descubrimiento de clases y predicción de clases mediante monitoreo de expresión génica. *Ciencia*, 286:531 —537, 1999.

[5] S. Mukherjee, P. Tamayo, D. Slonim, A. Verri, T. Golub, J. P. Mesirov y T. Poggio. Admite la clasificación de máquina vectorial de datos de microarrays. Informe técnico, AI Memo 1677, Instituto Tecnológico de Massachusetts, 1998.

Genes	Rechaza	Errores	Nivel de confianza	d
7129	3	0	93%	0.1
40	0	0	93%	0.1
5	3	0	92%	0.1

This page titled [16.6: Lectura adicional, Recursos, Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [16.6: Further Reading, Resources, Bibliography](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 17: Motivos Regulatorios, Muestreo de Gibbs y EM

- 17.1: Representación de Motivos y Contenido de Información
- 17.2: Introducción a los motivos reguladores y la regulación génica
- 17.3: Maximización de expectativas
- 17.4: Muestreo de Gibbs- Muestra de distribución conjunta ( $M, Z_{ij}$ )
- 17.5: Descubrimiento del motivo de novo
- 17.6: Posiblemente cosas en desuso por debajo-
- 17.7: Comparando diferentes métodos
- 17.8: OOPS, ZOOPS, MTC
- 17.9: Ampliación del Enfoque EM

---

This page titled [17: Motivos Regulatorios, Muestreo de Gibbs y EM](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 17.1: Representación de Motivos y Contenido de Información

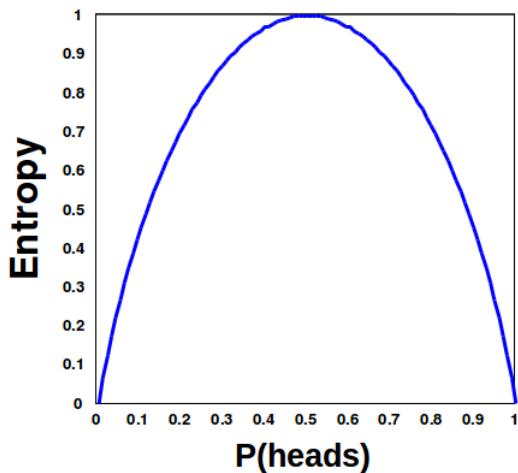
En lugar de una Matriz de Perfil, también podemos representar Motivos usando la teoría de la información. En la teoría de la información, la información sobre un determinado evento se comunica a través de un mensaje. La cantidad de información transportada por un mensaje se mide en bits. Podemos determinar los bits de información transportados por un mensaje observando la distribución de probabilidad del evento descrito en el mensaje. Básicamente, si no sabemos nada sobre el resultado del evento, el mensaje contendrá muchos bits. No obstante, si estamos bastante seguros de cómo va a llevarse a cabo el evento, y el mensaje sólo confirma nuestras sospechas, el mensaje lleva muy pocos bits de información. Por ejemplo, La sentencia “Un se levantará mañana” no es muy sorprendente, por lo que la información de esa sentencia si es bastante baja.. No obstante, la sentencia “Un no se levantará mañana” es muy sorprendente y tiene un alto contenido de información. Podemos calcular la cantidad específica de información en un mensaje dado con la ecuación:  $-\log p$ .

La entropía de Shannon es una medida de la cantidad esperada de información contenida en un mensaje. En otras palabras, es la información contenida por un mensaje de cada evento que posiblemente pueda ocurrir ponderada por cada probabilidad de eventos. La entropía de Shannon viene dada por la ecuación:

$$H(X) = - \sum_i p_i \log_2 p_i$$

La entropía es máxima cuando todos los eventos tienen la misma probabilidad de ocurrir. Esto se debe a que la Entropía nos dice la cantidad esperada de información que aprenderemos. Si cada uno incluso tiene las mismas posibilidades de ocurrir sabemos lo menos posible sobre el evento, por lo que se maximiza la cantidad esperada de información que aprenderemos. Por ejemplo, un volteo de moneda tiene una entropía máxima solo cuando la moneda es justa. Si la moneda no es justa, entonces sabemos más sobre el evento de la volteo de moneda, y el mensaje esperado del resultado del flip de moneda contendrá menos información.

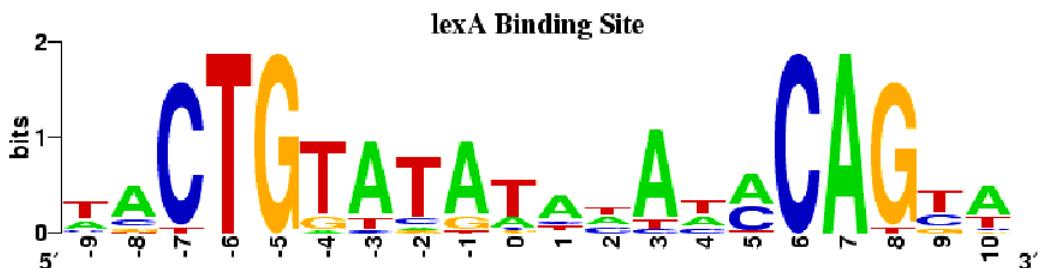
### Example: Coin Toss



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.11: La entropía se maximiza cuando tanto la cabeza como la cola tienen la misma probabilidad de ocurrir

Podemos modelar un motivo por la cantidad de información que tenemos de cada posición después de aplicar Gibbs Sampling o EM. En la siguiente figura, la altura de cada letra representa el número de bits de información que hemos aprendido sobre esa base. Las pilas más altas corresponden a una mayor certeza sobre cuál es la base en esa posición del motivo, mientras que las pilas inferiores corresponden a un mayor grado de incertidumbre. Con cuatro codones para elegir, la Entropía Shannon de cada posición es de 2 bits. Otra forma de observar esta figura es que la altura de una letra es proporcional a la frecuencia de la base en esa posición.



## 17.2: Introducción a los motivos reguladores y la regulación génica

Ya hemos explorado las áreas de programación dinámica, alineación de secuencias, clasificación y modelado de secuencias, modelos ocultos de Markov y maximización de expectativas. En el siguiente capítulo, veremos cómo estas técnicas también son útiles para identificar nuevos motivos y dilucidar sus funciones.

### El código regulatorio: Factores y Motivos de Transcripción

Los motivos son cortos (6-8 bases de largo), patrones recurrentes que tienen funciones biológicas bien definidas. Los motivos incluyen patrones de ADN en regiones potenciadoras o motivos promotores, así como motivos en secuencias de ARN tales como señales de corte y empalme. Como hemos comentado, la actividad genética está regulada en respuesta a las variaciones ambientales. Los motivos son responsables de reclutar Factores de Transcripción, o proteínas reguladoras, para el gen diana apropiado. Los motivos también pueden ser reconocidos por microARN, que se unen a motivos dados a través de la complementariedad; nucleosomas, que reconocen motivos basados en su contenido de GC; y otros ARN, que utilizan una combinación de secuencia y estructura de ADN. Una vez unidos, pueden activar o reprimir la expresión del gen asociado.

Los factores de transcripción (TFs) pueden utilizar varios mecanismos para controlar la expresión génica, incluyendo la acetilación y desacetilación de proteínas histonas, el reclutamiento de moléculas de cofactor en el complejo TF-ADN y la estabilización o alteración de las interfaces ARN-ADN durante la transcripción. A menudo regulan un grupo de genes que están involucrados en procesos celulares similares. Por lo tanto, es probable que los genes que contienen el mismo motivo en sus regiones aguas arriba estén relacionados en sus funciones. De hecho, muchos motivos reguladores se identifican analizando las regiones aguas arriba de genes que se sabe que tienen funciones similares.

Los motivos se han vuelto extremadamente útiles para definir redes reguladoras genéticas y descifrar las funciones de los genes individuales. Con nuestras habilidades computacionales actuales, el descubrimiento y análisis de motivos reguladores ha progresado considerablemente y se mantiene a la vanguardia de los estudios genómicos.

### Desafíos del descubrimiento de motivos

Antes de poder adentrarnos en algoritmos para el descubrimiento de motivos, primero debemos entender las características de los motivos, especialmente aquellos que hacen que los motivos sean algo difíciles de encontrar. Como se mencionó anteriormente, los motivos son generalmente muy cortos, generalmente de solo 6-8 pares de bases de largo. Adicionalmente, los motivos pueden ser degenerados, donde solo los nucleótidos en ciertas ubicaciones dentro del motivo afectan la función del motivo. Esta degeneración surge porque los factores de transcripción son libres de interactuar con sus motivos correspondientes de maneras más complejas que una simple relación de complementariedad. Como se ve en 17.1, muchas proteínas interactúan con el motivo no abriendo el ADN para verificar la complementariedad de bases, sino explorando los espacios, o surcos, entre las dos cadenas principales de fosfato de azúcar. Dependiendo de la estructura física del factor de transcripción, la proteína solo puede ser sensible a la diferencia entre purinas y pirimidinas o bases débiles y fuertes, a diferencia de identificar pares de bases específicos. La topología del factor de transcripción puede incluso hacerla tal que ciertos nucleótidos no interactúen en absoluto, permitiendo que esas bases actúen como comodines.

Este tema de degeneración dentro de un motivo plantea un problema desafiante. Si solo estuviéramos buscando un k-mer fijo, simplemente podríamos buscar el k-mer en todas las secuencias que estamos viendo usando alineación local

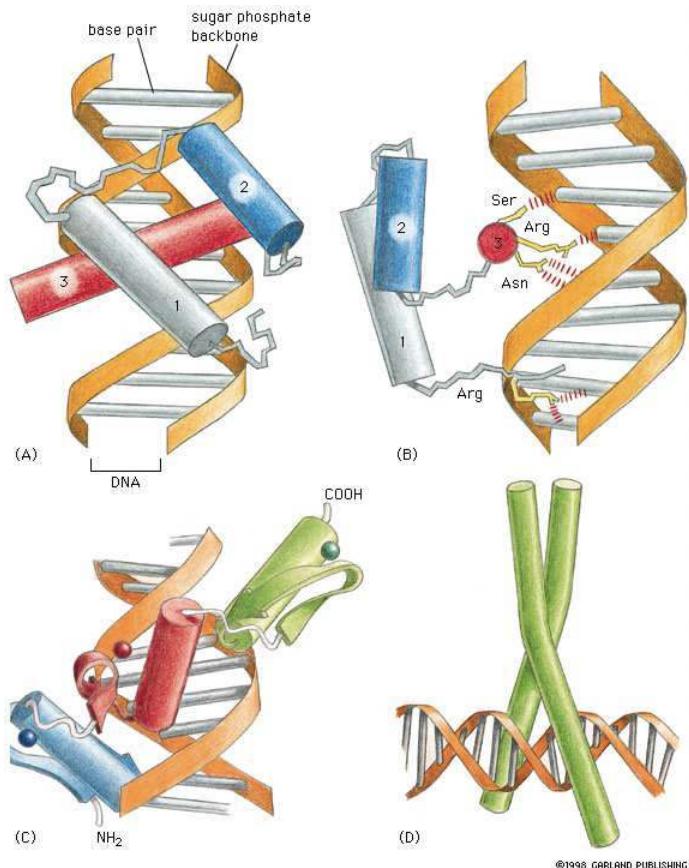


Figura 17.1: Factores de transcripción que se unen al ADN en un sitio de motivo

© Garland Publishing. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

herramientas. Sin embargo, el motivo puede variar de una secuencia a otra. Debido a esto, se dice que una cadena de nucleótidos que se sabe que es un motivo regulador es una instancia de un motivo porque representa una de las posibles muchas combinaciones diferentes de nucleótidos que cumplen la función del motivo.

En nuestros enfoques, hacemos dos suposiciones sobre los datos. Primero, suponemos que no hay correlaciones por pares entre bases, es decir, que cada base es independiente de cada otra base. Si bien tales correlaciones existen en la vida real, considerarlas en nuestro análisis conduciría a un crecimiento exponencial del espacio de parámetros que se está considerando, y consecuentemente correríamos el riesgo de sobreajustar nuestros datos. La segunda suposición que hacemos es que todos los motivos tienen longitudes fijas; de hecho, esta aproximación simplifica enormemente el problema. Sin embargo, incluso con estos dos supuestos, el hallazgo de motivos sigue siendo un problema muy desafiante. El tamaño relativamente pequeño de los motivos, junto con su gran variedad, hace bastante difícil localizarlos. Además, la ubicación de un motivo en relación con el gen correspondiente está lejos de ser fija; el motivo puede estar aguas arriba o aguas abajo, y la distancia entre el gen y el motivo también varía. De hecho, a veces el motivo está lejos de 10k a 10M pares de bases del gen.

### Los motivos resumen la especificidad de la secuencia de TF

Debido a que las instancias de motivos exhiben gran variedad, generalmente usamos una Matriz de Peso de Posición (PWM) para caracterizar el motivo. Esta matriz da la frecuencia de cada base en cada ubicación del motivo. La siguiente figura muestra un ejemplo de PWM, donde  $p_{ck}$  corresponde a la frecuencia de la base c en la posición k dentro del motivo, con  $p_{c0}$  denotando la distribución de bases en regiones no motivadas.

Ahora definimos el problema del hallazgo de motivos de manera más rigurosa. Suponemos que se nos da un conjunto de genes coregulados y funcionalmente relacionados. Muchos motivos fueron descubiertos previamente haciendo huella

## sequence positions

	1	2	3	4	5	6	7	8
A	0.1	0.3	0.1	0.2	0.2	0.4	0.3	0.1
C	0.5	0.2	0.1	0.1	0.6	0.1	0.2	0.7
G	0.2	0.2	0.6	0.5	0.1	0.2	0.2	0.1
T	0.2	0.3	0.2	0.2	0.1	0.3	0.3	0.1

Figura 17.2: Ejemplo de Matriz de Perfil

experimentos, que aislan secuencias unidas por factores de transcripción específicos, y por lo tanto es más probable que correspondan a motivos. Existen varios métodos computacionales que se pueden utilizar para localizar motivos:

1. Realice una alineación local a través del conjunto de secuencias y explore las alineaciones que resultaron en una puntuación de alineación muy alta.
2. Modele las regiones promotoras usando un modelo oculto de Markov y luego use un modelo generativo para encontrar secuencias no aleatorias.
3. Reducir el espacio de búsqueda aplicando conocimientos previos sobre cómo deberían verse los motivos.
4. Búsqueda de bloques conservados entre diferentes secuencias.
5. Examine la frecuencia de kmers a través de regiones altamente propensas a contener un motivo.
6. Utilice métodos probabilísticos, como EM, Gibbs Sampling o un algoritmo codicioso

El método 5, utilizando frecuencias relativas de kmer para descubrir motivos, presenta algunos desafíos a considerar. Por ejemplo, podría haber muchas palabras comunes que ocurren en estas regiones que de hecho no son motivos regulatorios sino diferentes conjuntos de instrucciones. Además, dada una lista de palabras que podrían ser un motivo, no es seguro que el motivo más probable sea la palabra más común; por ejemplo, mientras que los motivos generalmente están sobrerepresentados en las regiones promotoras, los factores de transcripción pueden ser incapaces de unirse si hay un exceso de motivos presentes. Una posible solución a este problema podría ser encontrar kmers con frecuencia relativa máxima en las regiones promotoras en comparación con las regiones de fondo. Esta estrategia se realiza comúnmente como una etapa de post-procesamiento para reducir el número de motivos posibles.

En la siguiente sección, hablaremos más sobre estos algoritmos probabilísticos así como los métodos para usar la frecuencia kmer para el descubrimiento de motivos. También volveremos a la idea de usar kmers para encontrar motivos en el contexto del uso de la conservación evolutiva para el descubrimiento de motivos.

---

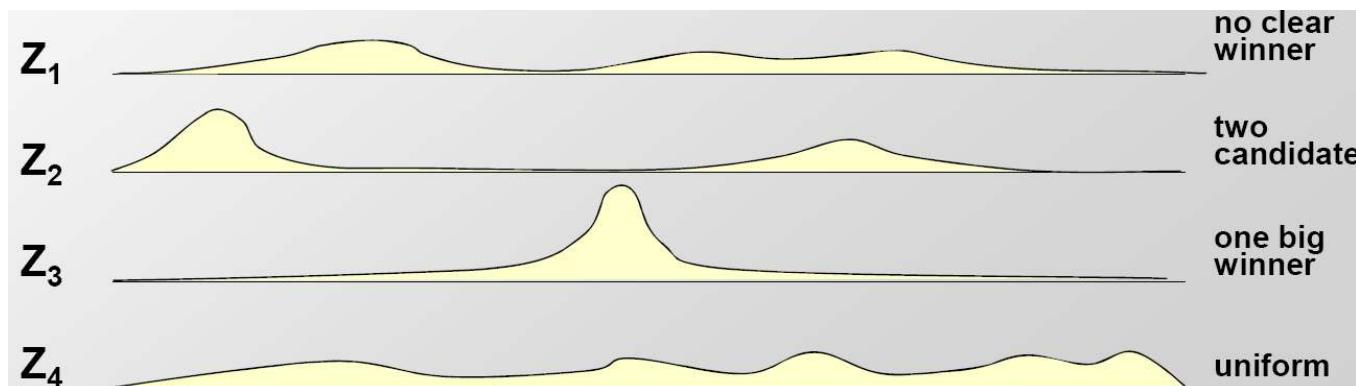
This page titled [17.2: Introducción a los motivos reguladores y la regulación génica](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.2: Introduction to regulatory motifs and gene regulation](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.3: Maximización de expectativas

### La idea clave detrás de EM

Se nos da un conjunto de secuencias con el supuesto de que los motivos se enriquecen en ellas. La tarea es encontrar el motivo común en esas secuencias. La idea clave detrás de los siguientes algoritmos probabilísticos es que si se nos dieran posiciones iniciales de motivo en cada secuencia, encontrar el motivo PWM sería trivial; de manera similar, si se nos diera el PWM para un motivo particular, sería fácil encontrar las posiciones iniciales en las secuencias de entrada. Sea  $Z$  la matriz en la que  $Z_{ij}$  corresponde a la probabilidad de que una instancia de motivo comience en la posición  $j$  en la secuencia  $i$  (en la Figura 17.8 se muestra un gráfico de las distribuciones de probabilidad resumidas en  $Z$ ). Por lo tanto, estos algoritmos se basan en un enfoque iterativo básico: dada una longitud de motivo  $L$  y una matriz inicial  $Z$ , podemos usar las posiciones iniciales para estimar el motivo  $y$ , y a su vez, usar el motivo resultante para reestimar las posiciones iniciales, iterando sobre estos dos pasos hasta la convergencia en un motivo.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.3: Ejemplos de la matriz  $Z$  calculada

### El paso E: Estimación de $Z_{ij}$ a partir del PWM



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.4: Selección de la ubicación del motivo: el algoritmo codicioso siempre escogerá la ubicación más probable para el motivo. El algoritmo EM tomará un promedio mientras que Gibbs Sampling utilizará realmente la distribución de probabilidad dada por  $Z$  para muestrear un motivo en cada paso

**Paso 1: Inicialización** El primer paso en EM es generar una matriz de ponderación de probabilidad inicial (PWM). El PWM describe la frecuencia de cada nucleótido en cada ubicación del motivo. En 17.5, hay un ejemplo de un PWM. En este ejemplo, asumimos que el motivo tiene ocho bases de largo.

Si se le da un conjunto de secuencias alineadas y la ubicación de motivos sospechosos dentro de ellas, entonces encontrar el PWM se logra calculando la frecuencia de cada base en cada posición del motivo sospechoso. Podemos inicializar el PWM eligiendo ubicaciones de inicio al azar.

Nos referimos al PWM como  $p_{ck}$ , donde  $p_{ck}$  es la probabilidad de que la base  $c$  ocurra en la posición  $k$  del motivo. Nota: si hay 0 probabilidad, generalmente es una buena idea insertar pseudo- recuentos en tus probabilidades. El PWM también se

llama la matriz de perfil. Además del PWM, también mantenemos una distribución de fondo  $p_{ck}$ ,  $k=0$ , una distribución de las bases no en el motivo.

**Paso 2: Expectativa** En el paso de expectativa, generamos un vector  $Z_{ij}$  que contiene la probabilidad de que el motivo comience en la posición  $j$  en la secuencia  $i$ . En EM, el vector  $Z$  nos da una forma de clasificar todos los nucleótidos en las secuencias y decirnos si son parte del motivo o no. Podemos calcular  $Z_{ij}$  usando la Regla de Bayes. Esto simplifica a:

$$Z_{ij}^t = \frac{\Pr^t(X_i | Z_{ij}) \Pr^t(Z_{ij} = 1)}{\sum_{k=1}^{L-W+1} \Pr^t(X_i | Z_{ik} = 1) \Pr^t(Z_{ik} = 1)}$$

sequence positions								
	1	2	3	4	5	6	7	8
A	0.1	0.3	0.1	0.2	0.2	0.4	0.3	0.1
C	0.5	0.2	0.1	0.1	0.6	0.1	0.2	0.7
G	0.2	0.2	0.6	0.5	0.1	0.2	0.2	0.1
T	0.2	0.3	0.2	0.2	0.1	0.3	0.3	0.1

Figura 17.5: Matriz de peso de posición de muestra

$$\Pr(X_i | Z_{ij} = 1, p) = \underbrace{\prod_{k=1}^{j-1} p_{c_k, 0}}_{\text{before motif}} \underbrace{\prod_{k=j}^{j+W-1} p_{c_k, k-j+1}}_{\text{motif}} \underbrace{\prod_{k=j+W}^L p_{c_k, 0}}_{\text{after motif}}$$

donde  $\Pr^t(X_i | Z_{ij} = 1) = \Pr(X_i | Z_{ij} = 1, p)$  se define como

Esta es la probabilidad de secuencia  $i$  dado que el motivo comienza en la posición  $j$ . El primer y último producto corresponden a la probabilidad de que las secuencias que preceden y siguen al motivo candidato provengan de alguna distribución de probabilidad de fondo mientras que el producto medio corresponde a la probabilidad que la instancia de motivo candidato vino de una distribución de probabilidad de motivo. En esta ecuación, asumimos que la secuencia tiene longitud  $L$  y el motivo tiene longitud  $W$ .

**paso: Encontrar el motivo de máxima verosimilitud desde las posiciones iniciales  $Z_{ij}$**

**Paso 3: Maximización** Una vez que hemos calculado  $Z_t$ , podemos usar los resultados para actualizar tanto el PWM como la distribución de probabilidad de fondo. Podemos actualizar el PWM usando la siguiente ecuación

$$p_{c,k}^{(t+1)} = \frac{n_{c,k} + d_{c,k}}{\sum_b (n_{b,k} + d_{b,k})}$$

pseudo-counts

$$n_{c,k} = \begin{cases} \sum_i \sum_{\{j | X_{i,j+k-1}=c\}} Z_{ij} & k > 0 \text{ motif} \\ n_c - \sum_{j=1}^W n_{c,j} & k = 0 \end{cases}$$

**total # of c's  
in data set**

Figura17.3.1: Copiar y Pegar Subtítulo aquí. (Copyright; autor vía fuente)

Paso 4: Repita los pasos 2 y 3 hasta la convergencia.

Una forma posible de probar si la matriz de perfil ha convergido es medir cuánto cambia cada elemento en el PWM después de la maximización de pasos. Si el cambio está por debajo de un umbral elegido, entonces podemos terminar el algoritmo. EM es un algoritmo determinista y depende completamente de los puntos de partida iniciales porque utiliza un promedio sobre la distribución de probabilidad completa. Por lo tanto, es recomendable volver a ejecutar el algoritmo con diferentes posiciones iniciales para intentar reducir la posibilidad de converger sobre un máximo local que no sea el máximo global y para tener una buena idea del espacio de solución.

---

This page titled [17.3: Maximización de expectativas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.3: Expectation maximization](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.4: Muestreo de Gibbs- Muestra de distribución conjunta (M, Z<sub>ij</sub>)

### Muestreo de posiciones de motivo basadas en el vector Z

El muestreo de Gibbs es similar al EM excepto que es un proceso estocástico, mientras que EM es determinista. En el paso de expectativa, solo consideramos nucleótidos dentro de la ventana de motivos en el muestreo de Gibbs. En el paso de maximización, tomamos muestras de Z<sub>ij</sub> y usamos el resultado para actualizar el PWM en lugar de promediar sobre todos los valores como en EM.

**Paso 1: Inicialización** Al igual que con EM, generas tu PWM inicial con un muestreo aleatorio de posiciones iniciales iniciales. La principal diferencia radica en el paso Maximización. Durante la EM, el algoritmo crea el motivo de secuencia considerando todos los posibles puntos de partida del motivo. Durante Gibbs, el algoritmo escoge un único punto de partida del motivo con la probabilidad de los puntos de partida Z.

**Paso 2: Elimina** una secuencia, X<sub>i</sub>, de tu conjunto de secuencias. Cambiará la ubicación inicial de para esta secuencia en particular.

**Paso 3: Actualizar** Usando el conjunto restante de secuencias, actualice el PWM contando con qué frecuencia ocurre cada base en cada posición, agregando pseudorecuentos según sea necesario.

**Paso 4: Muestra** Utilizando el PWM recién actualizado, computar la puntuación de cada punto de partida en la secuencia X<sub>i</sub>. Para generar cada puntaje, Z<sub>ij</sub>, se utiliza la siguiente fórmula:

$$A_{ij} = \frac{\prod_{k=j}^{j+W-1} p_{ek}, k - j + 1}{\prod_{k=j}^{j+W-1} p_{ck}, 0}$$

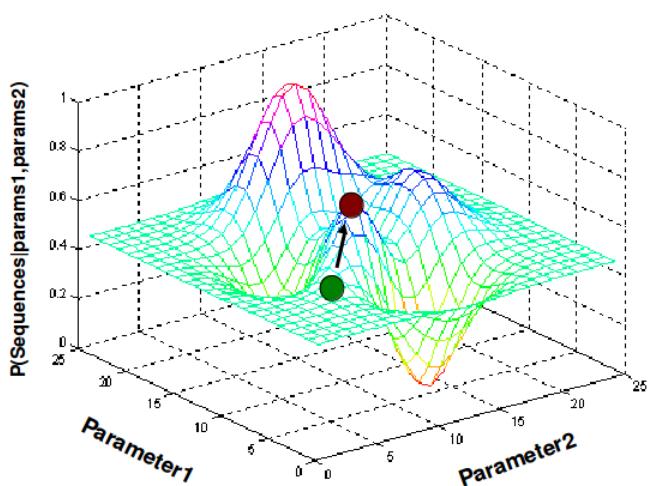
Esta es simplemente la probabilidad de que la secuencia se generó usando el motivo PWM dividida por la probabilidad de que la secuencia se generó usando el PWM de fondo.

Seleccione una nueva posición inicial para X<sub>i</sub> eligiendo aleatoriamente una posición en función de su Z<sub>ij</sub>.

**Paso 5:** Iterar Loop de nuevo al Paso 2 e iterar el algoritmo hasta la convergencia.

### Más probabilidades de encontrar el máximo global, fácil de implementar

Debido a que Gibbs actualiza su motivo de secuencia durante la Maximización basándose en una sola muestra del Motivo en lugar de cada muestra ponderada por sus puntuaciones, Gibbs es menos dependiente del PWM inicial. Es mucho más probable que EM se atasque en un máximo local que Gibbs debido a este hecho. No obstante, esto no quiere decir que Gibbs siempre devolverá el máximo global. Gibbs debe ejecutarse varias veces para asegurarse de haber encontrado el máximo global y no el máximo local. Dos implementaciones populares de Gibbs Sampling aplicadas a este problema son AlignAce y BioProspector. Un Sampler Gibbs más general se puede encontrar en el programa WinBugs. Tanto AlignAce como BioProspector utilizan el algoritmo antes mencionado para varias elecciones de valores iniciales y luego reportan motivos comunes. El muestreo de Gibbs es más fácil de implementar que el E-M, y en teoría, converge rápidamente y es menos probable que se atasque en un óptimo local. Sin embargo, la búsqueda es menos sistemática.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.6: Muestreo de Gibbs

This page titled [17.4: Muestreo de Gibbs- Muestra de distribución conjunta \( \$M, Z\_{ij}\$ \)](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.4: Gibbs Sampling- Sample from joint \( \$M,Z\_{ij}\$ \) distribution](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.5: Descubrimiento del motivo de novo

Como se discutió al inicio de este capítulo, el problema central para la búsqueda de motivos es definir los criterios para qué es un motivo válido y dónde se encuentran. Dado que la mayoría de los motivos están vinculados a importantes funciones biológicas, uno podría someter al organismo a una variedad de condiciones con la esperanza de desencadenar estas funciones biológicas. Luego se podrían buscar genes expresados diferencialmente, y luego usar esos genes como base para los cuales los genes están funcionalmente relacionados y, por lo tanto, es probable que sean controlados por la misma instancia de motivo. Sin embargo, esta técnica no solo se basa en el conocimiento previo de interesantes funciones biológicas para sondear, sino que también está sujeta a sesgos en el procedimiento experimental. Alternativamente, se podría usar Chip-seq para buscar motivos, pero este método se basa no solo en tener un Factor de Transcripción conocido de interés, sino que también requiere desarrollar anticuerpos para reconocer dicho Factor de Transcripción, lo que puede ser costoso y llevar mucho tiempo.

Idealmente uno sería capaz de descubrir motivos de novo, o sin depender de un conjunto de genes o Factor de Transcripción ya conocido. Si bien esto parece un problema difícil, de hecho se puede lograr aprovechando la conservación en todo el genoma. Debido a que las funciones biológicas generalmente se conservan entre especies y tienen distintas firmas evolutivas, se pueden alinear secuencias de especies cercanas y buscar específicamente en regiones conservadas (también conocidas como Isla de Conservación) para aumentar la tasa de búsqueda de motivos funcionales.

### Descubrimiento de motivos mediante la conservación de todo el genoma

Las islas de conservación a menudo se superponen a motivos conocidos, por lo que hacer exploraciones de todo el genoma a través de regiones conservadas evolutivas puede ayudarnos a descubrir motivos, de novo. Sin embargo, no todas las regiones conservadas serán motivos; por ejemplo, los nucleótidos que rodean a los motivos también pueden conservarse aunque ellos mismos no sean parte de un motivo. Distinguir motivos de regiones conservadas de fondo se puede hacer buscando enriquecimientos que seleccionen más específicamente para kmers involucrados en motivos reguladores. Por ejemplo, se pueden encontrar motivos reguladores mediante la búsqueda de secuencias conservadas enriquecidas en regiones intergénicas aguas arriba de los genes en comparación con regiones de control tales como secuencias codificantes, ya que se esperaría que los motivos se enriquecieran en o alrededor de promotores de genes. También se puede ampliar este modelo para encontrar motivos degenerados: podemos buscar la conservación de motivos más pequeños, no degenerados separados por un hueco de longitud variable, como se muestra en la siguiente figura. También podemos extender este motivo a través de una búsqueda codiciosa para acercarnos a encontrar el motivo local de máxima verosimilitud. Finalmente, la evolución de los motivos también puede revelar qué motivos están degenerados; dado que un motivo particular es más probable que se degenera si a menudo es reemplazado por otro motivo a lo largo de la evolución, la agrupación de motivos puede revelar qué kmers probablemente corresponderán al mismo motivo.

De hecho, la estrategia tiene su relevancia biológica. En 2003, el profesor Kellis argumentó que debe haber cierta presión selectiva para hacer que una secuencia particular se produzca en lugares específicos. Su tesis doctoral sobre el tema se puede encontrar en la siguiente ubicación:



Figura 17.7: Uso de semillas de motivos para encontrar motivos degenerados

### Validación de motivos descubiertos con conjuntos de datos funcionales

Estos motivos predichos pueden ser validados con conjuntos de datos funcionales. Los motivos predichos con al menos una de las siguientes características tienen más probabilidades de ser motivos reales: -enriquecimiento en genes co-regulados. Esto se puede extender aún más a grupos génicos más grandes; por ejemplo, se ha encontrado que los motivos están enriquecidos en genes expresados en tejidos específicos -solapamiento con experimentos de unión a TF -enriquecimiento en genes de los mismos sesgos posicionales complejos con respecto al sitio de inicio de la transcripción (TSS): los motivos están enriquecidos en los genes TSS - cadena arriba vs corriente abajo de los genes, sesgos positonales inter- vs. intra-génicos: los motivos generalmente están agotados en secuencias codificantes -similitud con motivos de factores de transcripción conocidos: algunos, pero no todos, los motivos descubiertos pueden coincidir con motivos conocidos (sin embargo, no todos los motivos están conservados y los motivos conocidos pueden no ser exactamente correcto)

This page titled [17.5: Descubrimiento del motivo de novo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.5: De novo motif discovery](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.6: Posiblemente cosas en desuso por debajo-

### Codiosos

Si bien el algoritmo codicioso no se usa mucho en la práctica, es importante saber cómo funciona y principalmente sus ventajas y desventajas en comparación con el muestreo EM y Gibbs. El algoritmo Greedy funciona igual que el muestreo de Gibbs excepto por una diferencia principal en el Paso 4. En lugar de elegir aleatoriamente seleccionar una nueva ubicación de inicio, siempre elige la ubicación de inicio con la mayor probabilidad.

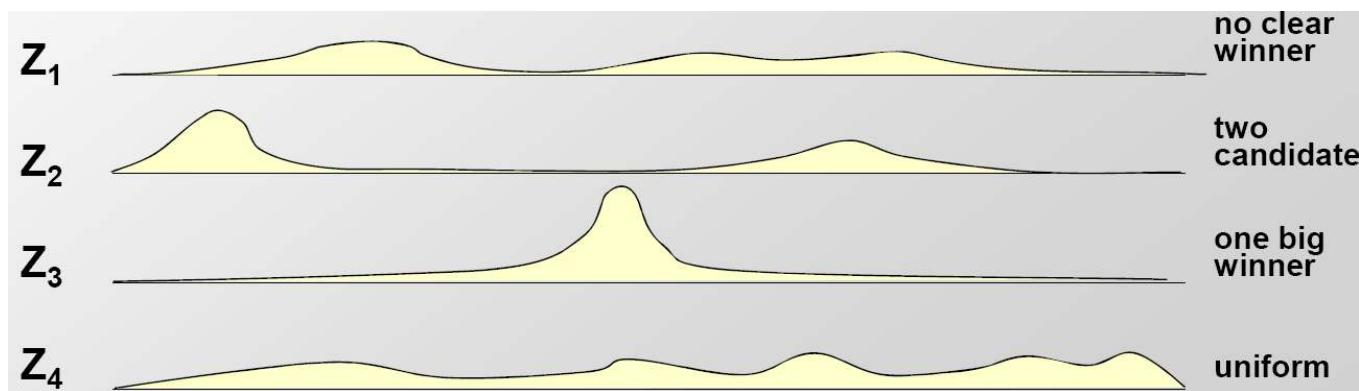
Esto hace que el algoritmo Greedy sea ligeramente más rápido que el muestreo de Gibbs pero reduce considerablemente sus posibilidades de encontrar un máximo global. En los casos en que la distribución de probabilidad de ubicación inicial se distribuye de manera bastante uniforme, el algoritmo codicioso ignora los pesos de cualquier otra posición inicial que no sea la más probable.

This page titled [17.6: Posiblemente cosas en desuso por debajo-](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **17.6: Possibly deprecated stuff below-** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.7: Comparando diferentes métodos

La principal diferencia entre Gibbs, EM y el algoritmo Greedy radica en su paso de maximización después de calcular su matriz Z. Ejemplos de la matriz Z se representan gráficamente a continuación. Esta matriz Z se utiliza entonces para volver a calcular la matriz de perfil original hasta la convergencia. Algunos ejemplos de esta matriz están representados gráficamente por 17.8



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.8: Ejemplos de la matriz Z calculada a través de EM, muestreo de Gibbs y el algoritmo codicioso

Intuitivamente, el algoritmo codicioso siempre escogerá la ubicación más probable para el motivo. El algoritmo EM tomará un promedio de todos los valores, mientras que Gibbs Sampling utilizará realmente la distribución de probabilidad dada por Z para muestrear un motivo en un paso.



© fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 17.9: Selección de la ubicación del motivo: el algoritmo codicioso siempre escogerá la ubicación más probable para el motivo. El algoritmo EM tomará un promedio mientras que Gibbs Sampling utilizará realmente la distribución de probabilidad dada por Z para muestrear un motivo en cada paso

This page titled [17.7: Comparando diferentes métodos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.7: Comparing different Methods](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.8: OOPS, ZOOPS, MTC

Los diferentes tipos de modelo de secuencia hacen suposiciones diferentes sobre cómo y dónde aparecen las ocurrencias de motivos en el conjunto de datos. El tipo de modelo más simple es OOPS (OOPS) ya que asume que hay exactamente una ocurrencia por secuencia del motivo en el conjunto de datos. Este es el caso que hemos analizado en la sección de muestreo de Gibbs. Este tipo de modelo fue introducido por Lawrence & Reilly (1990) [2], cuando describen por primera vez una generalización de OOPS, llamada ZOOPS (Zero-o-Uno-Sucece-per-Sequence), que asume cero o una ocurrencia de motivo por secuencia de conjunto de datos. Finalmente, los modelos TCM (mezcla de dos componentes) asumen que hay cero o más ocurrencias no superpuestas del motivo en cada secuencia del conjunto de datos, como describen Baily & Elkan (1994). [1] Cada uno de estos tipos de modelo de secuencia consta de dos componentes, que modelan, respectivamente, el motivo y no-posiciones del motivo (fondo) en secuencias. Un motivo es modelado por una secuencia de variables aleatorias discretas cuyos parámetros dan las probabilidades de que cada una de las diferentes letras (4 en el caso del ADN, 20 en el caso de las proteínas) ocurran en cada una de las diferentes posiciones en una ocurrencia del motivo. Las posiciones de fondo en la secuencia se modelan mediante una única variable aleatoria discreta.

This page titled [17.8: OOPS, ZOOPS, MTC](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.8: OOPS,ZOOPS,TCM](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 17.9: Ampliación del Enfoque EM

### Modelo ZOOPS

El enfoque presentado antes (OOPS) se basa en la suposición de que cada secuencia se caracteriza por un solo motivo (por ejemplo, hay exactamente una ocurrencia de motivo en una secuencia dada). El modelo ZOOPS toma en consideración la posibilidad de secuencias que no contengan motivos.

En este caso deja que yo sea una secuencia que no contenga un motivo. Esta información extra se agrega a nuestro modelo anterior usando otro parámetro  $\lambda$  para denotar la probabilidad previa de que cualquier posición en una secuencia sea el inicio de un motivo. A continuación, la probabilidad de que toda la secuencia contenga un motivo es  $\lambda = (L - W + 1) * \lambda$

#### El E-Step

El paso E del modelo ZOOPS calcula el valor esperado de la información faltante, la probabilidad de que una ocurrencia de motivo comience en la posición  $j$  de la secuencia  $X_i$ . A continuación se dan las fórmulas utilizadas para los tres tipos de modelo.

$$Z_{ij}^t = \frac{\Pr^{(t)}(X_i | Z_{ij} = 1) \lambda^{(t)}}{\Pr^{(t)}(X_i | Q_i = 0)(1 - \lambda^{(t)}) + \sum_{k=1}^{L-W+1} \Pr^{(t)}(X_i | Z_{ik} = 1) \lambda^{(t)}}$$

donde  $\lambda^{(t)}$  es la probabilidad de que la secuencia  $i$  tenga un motivo,  $\Pr^{(t)}(X_i | Q_i = 0)$  es la probabilidad de que  $x_i$  se genere a partir de una secuencia  $i$  que no contiene un motivo

#### El M-Step

El paso M de EM en MEME reestima los valores para  $\lambda$  usando las fórmulas anteriores. La matemática sigue siendo la misma que para OOPS, solo actualizamos los valores para  $\lambda$  y  $\gamma$

$$\lambda^{(t+1)} = \frac{\gamma^{(t+1)}}{(L - W + 1)} = \frac{1}{n(L - W + 1)} \sum_{i=1}^n \sum_{j=1}^m Z_{i,j}^{(t)}$$

El modelo anterior toma en consideración secuencias que no tienen ningún motivo. El reto es también tomar en consideración la situación en la que hay más de un motivo por secuencia. Esto se puede lograr con el modelo más general TCM. El TCM (modelo de mezcla de dos componentes) se basa en la suposición de que puede haber cero, una o incluso dos ocurrencias de motivo por secuencia.



Figura 17.10: Secuencias con cero, uno o dos motivos.

### Encontrar múltiples motivos

Todos los modelos de secuencia anteriores modelan secuencias que contienen un solo motivo (observe que el modelo TCM puede describir secuencias con múltiples ocurrencias del mismo motivo). Para encontrar múltiples motivos diferentes, no superpuestos en un solo conjunto de datos, se incorpora información sobre los motivos ya descubiertos en el modelo actual para evitar redescubrir el mismo motivo. Los tres tipos de modelos de secuencia asumen que las ocurrencias de motivos son igualmente probables en cada

posición  $j$  en las secuencias  $x_i$ . Esto se traduce en una distribución de probabilidad previa uniforme sobre las variables de datos faltantes  $Z_{ij}$ . Se tuvo que usar un nuevo prior en cada  $Z_{ij}$  durante la etapa E que toma en cuenta la probabilidad de que una nueva ocurrencia de motivo Ancho-W comenzando en la posición  $X_{ij}$  pudiera superponerse a las ocurrencias de los motivos previamente encontrados. Para ayudar a calcular el nuevo previo en  $Z_{ij}$  introducimos variables  $V_{ij}$  donde  $V_{ij} = 1$  si una ocurrencia de motivo Ancho-W podría comenzar en la posición  $j$  en la secuencia  $X_i$  sin solapar una ocurrencia de un motivo encontrado en una pasada previa. De lo contrario  $V_{ij} = 0$ .

$$V_{ij} = \begin{cases} 1, & \text{no previous motifs in } [X_{i,j}, \dots, X_{i,j+w-1}] \\ 0, & \text{otherwise} \end{cases}$$

---

This page titled [17.9: Ampliación del Enfoque EM](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [17.9: Extension of the EM Approach](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 18: Genómica Regulatoria

18.1: Introducción a la Genómica Regulatoria

18.2: Descubrimiento de Motivos De Novo

18.3: Predecir objetivos regulares

18.4: Genes y dianas de microARN

---

This page titled [18: Genómica Regulatoria](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 18.1: Introducción a la Genómica Regulatoria

Cada célula tiene el mismo ADN, pero todas tienen diferentes patrones de expresión debido a la regulación temporal y espacial de los genes. La genómica reguladora explica estos complejos patrones de expresión génica. Los reguladores que estaremos discutiendo son:

- Factor de Transcripción (TF) - Regula la transcripción de ADN a ARNm. Los TFs son proteínas que se unen al ADN antes de la transcripción y aumentan o disminuyen la transcripción. Podemos determinar la especificidad de un TF a través de métodos experimentales utilizando proteínas o anticuerpos. Podemos encontrar los genes por su similitud para conocer los TFs.
- Micro ARN (miARN) - Regula la traducción del ARNm a Proteínas. Los miARN son moléculas de ARN que se unen al ARNm después de la transcripción y pueden reducir la traducción. Podemos determinar la especificidad de un miARN a través de métodos experimentales, como la clonación, o métodos computacionales, usando conservación y estructura.

### Problemas Abiertos

Tanto los TFs como los miARN son reguladores y los podemos encontrar a través de métodos experimentales y computacionales. Discutiremos algunos de estos métodos computacionales, específicamente el uso de firmas evolutivas. Estos reguladores se unen a patrones específicos, llamados motivos. Podemos predecir los motivos a los que se unirá un regulador utilizando métodos experimentales y computacionales. Discutiremos la identificación de miARN a través de firmas evolutivas y estructurales y la identificación de ambos TFs y miARN a través del descubrimiento comparativo de novo, que teóricamente puede encontrar todos los motivos. Dado un motivo, es difícil encontrar el regulador que se une a él.

Un objetivo es un lugar donde un factor se une. Hay muchos motivos de secuencia, sin embargo muchos no se unirán; solo un subconjunto serán dianas. Las dianas para un regulador específico se pueden determinar mediante métodos experimentales. En la Conferencia 11, se discutieron métodos para encontrar un motivo dado un objetivo. También discutiremos la búsqueda de dianas dado un motivo.

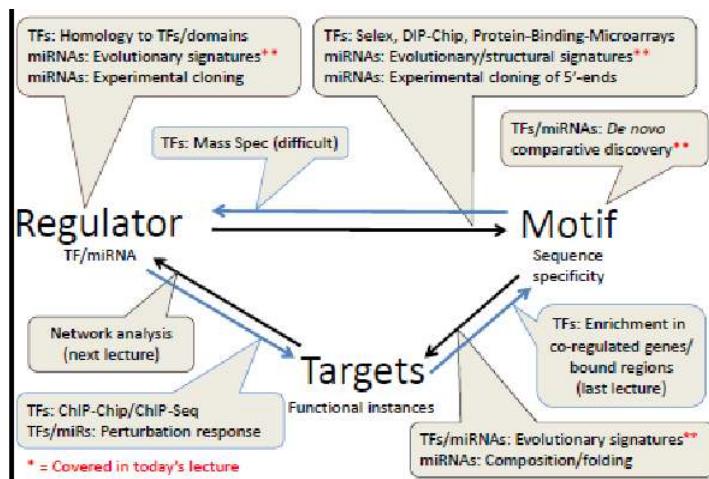


Figura 18.1: Desafíos en la Genómica Regulatoria

This page titled [18.1: Introducción a la Genómica Regulatoria](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [18.1: Introduction to Regulatory Genomics](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 18.2: Descubrimiento de Motivos De Novo

### Descubrimiento de Motivos

Los factores de transcripción influyen en la expresión de genes diana como activadores o represores al unirse al ADN cerca de genes. Esta unión está guiada por la especificidad de la secuencia de TF. Cuanto más cerca esté el ADN de la preferencia de base, más probable es que el factor se una. Estos motivos se pueden encontrar tanto computacional como experimentalmente. Existen tres enfoques principales para descubrir estos motivos.

- **Corregulación** - En la Conferencia 11, discutimos un tipo de corregulación de descubrimiento de motivos mediante la búsqueda de secuencias que probablemente tengan el motivo unido. Luego podemos usar enfoques enumerativos o métodos de alineación para encontrar estos motivos en las regiones aguas arriba. Podemos aplicar técnicas similares a datos experimentales donde se sabe dónde se une el motivo.
- **Factor Centric** - También hay métodos centrados en factores para descubrir motivos. Estos son en su mayoría métodos experimentales que requieren una proteína o anticuerpo. Los ejemplos incluyen SELEX, DIP-chip y PBMs. Todos estos métodos son in vitro.
- **Evolutivo** - En lugar de centrarse en un solo factor, los métodos evolutivos se enfocan en todos los factores. Podemos comenzar por mirar un solo factor y determinar qué propiedades podemos explotar. Hay ciertas secuencias que se conservan preferentemente (islas de conservación). Sin embargo, estos no siempre son motivos y en cambio pueden deberse a la conservación casual o no de motivos. Luego podemos observar muchas regiones, encontrar motivos más conservados y determinar cuáles están más conservados en general. Al probar la conservación en muchas regiones a través de muchos genomas, aumentamos el poder. Estos motivos tienen ciertas firmas evolutivas que nos ayudan a identificarlos: los motivos están más conservados en regiones intergénicas que en regiones codificantes, es más probable que los motivos estén aguas arriba de un gen que aguas abajo. Este es un método para tomar un motivo conocido y probar si se conserva.

Ahora queremos encontrar todo lo que esté más conservado de lo esperado. Esto se puede hacer usando un enfoque de escalada en colina. Comenzamos por enumerar las semillas del motivo, que suelen estar en forma de 3-gap-3. Luego, cada una de estas semillas es puntuada y clasificada usando una relación de conservación corregida por composición y pequeños recuentos. Estas semillas se expanden luego para llenar bases no especificadas alrededor de la semilla usando escalada en colinas. A través de estos métodos, es posible llegar a semillas iguales, o muy similares de diferentes maneras. Así, nuestro paso final consiste en agrupar las semillas usando similitud de secuencia para eliminar redundancia.

Un método final que podemos utilizar es registrar la frecuencia con la que una secuencia es reemplazada por otra en evolución. Esto produce racimos de k-meros que corresponden a un solo motivo.

### Validación de motivos descubiertos

Hay muchas formas en las que podemos validar motivos descubiertos. En primer lugar, esperamos que coincidan con motivos reales, lo que ocurre significativamente más a menudo que con motivos aleatorios. Sin embargo, esto no es un acuerdo perfecto, posiblemente debido a que muchos motivos conocidos no se conservan y que los motivos conocidos están sesgados y pueden haber perdido motivos reales. Sesgo posicional. Sesgado hacia el TSS,

Los motivos también tienen enriquecimientos funcionales. Si un TF específico se expresa en un tejido, entonces esperamos que la región aguas arriba tenga el motivo de ese factor. Esto también revela módulos de motivos cooperantes. También vemos que la mayoría de los motivos se evitan en genes expresados ubicuamente, de manera que no se encienden y apagan aleatoriamente.

### Resumen

Hay desventajas en todos estos enfoques. Tanto los enfoques TF como los centrados en la región no son integrales y están sesgados. Los enfoques centrados en TF requieren un factor de transcripción o anticuerpo, requieren mucho tiempo y dinero, y también tienen desafíos computacionales. El descubrimiento de novo utilizando la conservación es imparcial, pero no puede hacer coincidir los motivos con los factores y requiere múltiples genomas.

This page titled [18.2: Descubrimiento de Motivos De Novo](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **18.2: De Novo Motif Discovery** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 18.3: Predecir objetivos regulares

### Identificación de instancias de motivo

Una vez descubiertos los motivos potenciales, el siguiente paso es descubrir qué coincidencias de motivos son reales. Esto se puede hacer por métodos tanto experimentales como computacionales.

- Experimental - Las instancias se pueden identificar experimentalmente usando los métodos Chip-chip y Chip-DEQ. Ambos son métodos *in vivo*. Esto se hace mediante la reticulación de las células. El ADN se divide primero en secciones. Después se agrega la proteína y su anticuerpo o proteína etiquetada, que se une a diversas secuencias. Estas secuencias unidas ahora se extraen y el entrecruzamiento se invierte. Esto nos permite determinar en qué parte del genoma se unió el factor. Esto tiene una alta tasa de falsos positivos debido a que hay muchos casos en los que un factor se une, pero no es funcional. Este es un método experimental muy popular, pero está limitado por la disponibilidad de anticuerpos, que son difíciles de obtener por muchos factores.
- Computacional- Enfoques de cómputos. También hay muchos enfoques computacionales para identificar instancias. Los enfoques de genoma único utilizan agrupamiento de motivos. Buscan muchos partidos para aumentar el poder y son capaces de encontrar regiones regulatorias (CRM). Sin embargo, pierden instancias de motivos que ocurren solos y requieren de un conjunto de factores específicos que actúan juntos. Los enfoques multigenómicos, conocidos como huellas filogénicas, enfrentan muchos desafíos. Comienzan alineando muchas secuencias, pero incluso en motivos funcionales, las secuencias pueden moverse, mutar o faltar. El enfoque adoptado por Kheradpour maneja esto al no requerir una conservación perfecta (mediante el uso de una puntuación de longitud de rama) y al no requerir una alineación exacta (buscando dentro de una ventana).

Las puntuaciones de longitud de rama (BLS) se calculan tomando una coincidencia de motivo y buscándolo en otras especies. Luego, se encuentra el subárbol más pequeño que contiene todas las especies con un motivo coincidente. El porcentaje del árbol total es el BLS. El cálculo del BLS de esta manera permite mutaciones permitidas por degeneración de motivos, desalighment y movimiento dentro de una ventana, y motivos faltantes en árboles de especies densas.

Este BLS se traduce luego en un puntaje de confianza. Esto nos permite evaluar la probabilidad de una puntuación dada y dar cuenta de las diferencias en la composición del motivo y la longitud. Calculamos esta puntuación de confianza contando todas las instancias de motivos y motivos de control en cada BLS. Entonces queremos ver qué fracción de las instancias de motivo parecen ser reales. El puntaje de confianza es entonces señal/ (señal+ruido). Los motivos de control utilizados en este cálculo se producen produciendo 100 barajados del motivo original, y filtrando los resultados requiriendo que coincidan con el genoma con +/- 20% del motivo original. Estos son luego ordenados en función de su similitud con motivos conocidos y agrupados. A lo sumo se toma un motivo de cada racimo, en orden creciente de similitud, para producir nuestros motivos de control.

### Validación de objetivos

Similar al descubrimiento de motivos, podemos validar dianas al ver dónde caen en el genoma. La confianza selecciona para instancias de motivos TF en promotores y motivos de miARN en UTR 3', que es lo que esperamos. Los TFs pueden ocurrir en cualquiera de las cadenas, mientras que el miARN debe caer en una sola cadena. Así, aunque no hay preferencia por los TF, los miARN se encuentran preferentemente en la cadena positiva.

Otro método de validación de objetivos es computando enriquecimientos. Esto requiere tener un conjunto de regiones de fondo y primer plano. Estos podrían ser un promotor de genes co-regulados frente a todos los genes o regiones unidos por un factor frente a otras regiones intergénicas. El enriquecimiento se calcula tomando la fracción de instancias de motivo dentro del primer plano vs la fracción de bases en primer plano. La composición y el nivel de conservación se corrigen con motivos control. Estas fracciones se pueden hacer más conservadoras usando un intervalo de confianza binomial.

Las dianas se pueden validar comparando con instancias experimentales encontradas usando CHIP-seq. Esto muestra que las instancias conservadas del motivo CTCF están altamente enriquecidas en sitios ChIP-seq. El aumento de la confianza también aumenta el enriquecimiento. Usando esto, se verifican muchas instancias de motivos. Chip-seq no siempre encuentra motivos funcionales, por lo que estos resultados se pueden verificar aún más comparando con regiones unidas conservadas. Esto encuentra que el enriquecimiento en las intersecciones es dramáticamente mayor. Esto muestra dónde son vinculantes factores que tienen un

efecto que vale la pena conservar en la evolución. Estos dos enfoques son complementarios y son aún más efectivos cuando se usan juntos.

---

This page titled [18.3: Predecir objetivos regulares](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [18.3: Predicting Regular Targets](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 18.4: Genes y dianas de microARN

### Descubrimiento de genes de miARN

Los miARN son reguladores postranscripcionales que se unen a los ARNm para silenciar un gen. Son un regulador sumamente importante en el desarrollo. Estos se forman cuando se transcribe un gen de miARN a partir del genoma. La hebra resultante forma una horquilla en algún momento. Este es procesado, recortado y exportado al cito-plasma. Después, otra proteína recorta la horquilla y una mitad se incorpora a un complejo RISK. Al hacer esto, es capaz de decirle al complejo RISK dónde unirse, lo que determina qué gen está apagado. La segunda hebra generalmente se descarta. Es un problema computacional determinar qué hebra es cuál. El problema computacional aquí es cómo encontrar los genes que corresponden a estos miARN.

El primer problema es encontrar horquillas. Simplemente plegar el genoma produce aproximadamente 760,000 horquillas, pero solo hay de 60 a 200 miARN verdaderos. Por lo tanto, necesitamos métodos que ayuden a mejorar la especificidad. Se pueden considerar características estructurales, incluyendo energía de plegado, bucles (número, simetría), longitud y simetría de horquilla, subestructuras y emparejamientos, sin embargo, esto solo aumenta la especificidad en un factor de 40. Así, la estructura por sí sola no puede predecir miARN. También se pueden considerar las firmas evolutivas. Los miARN muestran propiedades de conservación características. Las horquillas consisten en un bucle, dos brazos y regiones flanqueantes. En la mayoría de los ARN, el bucle es el más bien conservado debido a que se utiliza en la unión. En miARN, sin embargo, los brazos están más conservados porque determinan dónde se unirá el complejo RISK. Esto aumenta la especificidad en un factor de 300. Tanto estas características estructurales como las propiedades de conservación se pueden combinar para predecir mejor los miARN potenciales.

Estas características se combinan mediante aprendizaje automático, específicamente bosques aleatorios. Esto produce muchos clasificadores débiles (árboles de decisión) en subconjuntos de positivos y negativos. Cada árbol vota entonces la clasificación final de un miARN dado. El uso de esta técnica nos permite alcanzar la sensibilidad deseada (incrementada en 4,500 veces).

### Validación de miARN descubiertos

Los miARN descubiertos pueden validarse comparando con miARN conocidos. Un ejemplo dado en clase muestra que el 81% de los miARN descubiertos ya eran conocidos por existir, lo que demuestra que estos métodos funcionan bien. Los miARN putativos aún no se han probado, sin embargo esto puede ser difícil de hacer ya que las pruebas se realizan mediante clonación.

La especificidad de región es otro método para validar miARN. En el fondo, las horquillas se distribuyen de manera bastante uniforme entre intrones, exones, regiones intergénicas y repeticiones y transposones. El aumento de la confianza en las predicciones hace que casi todos los miARN caigan en intrones y regiones intergénicas, como se esperaba. Estas predicciones también coinciden con las lecturas de secuenciación.

Esto también produjo algunas propiedades genómicas típicas de los miARN. Tienen preferencia por la hebra transcrita. Esto les permite incorporarse al intrón del gen real, y por lo tanto no requieren una transcripción separada. También se agrupan con miARN conocidos y predichos. Esto indica que están en la misma familia y tienen un orgán común.

### Identificación del extremo 5' del miARN

Las primeras siete bases determinan dónde se une un miARN, por lo que es importante saber exactamente dónde ocurre el cleavage. Si este punto de caída está equivocado incluso por dos bases, se predecirá que el miARN se unirá a un gen completamente diferente. Estos puntos de cleavage se pueden descubrir computacionalmente mediante la búsqueda de 7-meros altamente conservados que podrían ser objetivos. Estos 7-meros también se correlacionan con la falta de anti-dianas en genes expresados ubicuamente. Usando estas características, características estructurales y características conservacionales, es posible adoptar un enfoque de aprendizaje automático (SVM) para predecir el sitio de cleavage. Algunos miARN no tienen una sola posición de puntuación alta, y estos también muestran un procesamiento impreciso en la célula. Si la secuencia estelar está altamente puntuada, entonces tiende a expresarse más en la célula también.

### Motivos funcionales en regiones codificantes

Cada tipo de motivo tiene firmas distintas. El ADN es simétrico de cadena, el ARN es específico de cadena e invariante de marco, y la proteína es específica de cadena y sesgada al marco. Esta invarianza de marco se puede utilizar como firma. Cada cuadro puede

ser evaluado por separado. Los motivos debidos a sesgos de uso de di-codones se conservan en un solo desplazamiento de marco, mientras que los motivos debido a la regulación del nivel de ARN se conservan en los tres desplazamientos de marco. Esto permite distinguir presiones superpuestas.

---

This page titled [18.4: Genes y dianas de microARN](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [18.4: MicroRNA Genes and Targets](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19: Epigenómica

Esta página se ha generado automáticamente porque un usuario ha creado una subpágina de esta página.

---

19: Epigenómica is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## CHAPTER OVERVIEW

### 19: Epigenómica/Estados de cromatina

- 19.1: Introducción
- 19.2: Información Epigenética en Nucleosomas
- 19.3: Ensayos Epigenómicos
- 19.4: Procesamiento primario de datos de ChIP
- 19.5: Anotar el genoma usando firmas de cromatina
- 19.6: Direcciones actuales de investigación
- 19.7: Lectura adicional, herramientas y técnicas
- 19.8: ¿Qué hemos aprendido? , Bibliografía

---

This page titled [19: Epigenómica/Estados de cromatina](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 19.1: Introducción

El cuerpo humano contiene aproximadamente 210 tipos de células diferentes, pero cada tipo de célula comparte la misma secuencia genómica. A pesar de tener el mismo código genético, las células no solo se desarrollan en tipos distintos a partir de esta misma secuencia, sino que también mantienen el mismo tipo celular a lo largo del tiempo y a través de divisiones. Esta información sobre el tipo de célula y el estado de la célula se denomina información epigenómica. El epigenoma (“epi” significa arriba en griego, por lo que epigenoma significa por encima del genoma) es el conjunto de modificaciones químicas o marcas que influyen en la expresión génica y se transfieren a través de divisiones celulares y, en algunos casos limitados, a través de generaciones de organismos.

Como se muestra en la Figura 19.1, la información epigenómica en una célula se codifica de diversas maneras. Por ejemplo, la metilación del ADN (por ejemplo, en dinucleótidos CpG) puede alterar la expresión génica. De manera similar, el posicionamiento de los nucleosomas (unidad de empaquetamiento de ADN) determina a qué partes del ADN son accesibles para que los factores de transcripción se unan a y otras enzimas. Casi dos décadas de trabajo han revelado cientos de modificaciones postraduccionales de colas de histonas. Dado que un número extremadamente grande de estados de modificación de histonas son posibles para cualquier cola de histona dada, se ha propuesto la “hipótesis del código de histonas”. Esta hipótesis establece que combinaciones particulares de modificaciones de histonas codifican información. Aunque es una hipótesis controvertida, ha guiado el campo de la epigenética. El núcleo de la epigenética es comprender cómo se establecen las modificaciones químicas a la cromatina (ya sean metilación del ADN, modificaciones de histonas o arquitectura de cromatina) y cómo la célula “interpreta” esta información para establecer y mantener estados de expresión génica.

En este capítulo exploraremos las técnicas experimentales y computacionales utilizadas para descubrir estados de cromatina dentro de un tipo de célula. Aprenderemos cómo se puede usar la inmunoprecipitación de cromatina para inferir las regiones del genoma unidas por una proteína o interés, y se puede usar un algoritmo común (la transformada de Burrows-Wheeler) para mapear rápidamente grandes números de lecturas de secuenciación cortas a un genoma de referencia. A partir de esto se abstracta un nivel y utilizamos un modelo oculto de Markov (HMM) para segmentar el genoma en regiones que comparten estados de cromatina similares. Cerraremos mostrando cómo estos mapas integrales de estados de cromatina pueden compararse entre tipos de células y pueden usarse para proporcionar información sobre cómo se establecen y mantienen los estados celulares y el impacto de la variación genética en la expresión génica.

---

This page titled [19.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [19.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.2: Información Epigenética en Nucleosomas

Con el fin de colocar dos metros de ADN en un núcleo celular de 5-20  $\mu\text{m}$  de diámetro y disponer el ADN para facilitar el acceso a la maquinaria transcripcional, el ADN se empaqueta en cromatina. Los nucleosomas forman la unidad de este empaque. Un nucleosoma está compuesto por ADN de aproximadamente 150-200 pb de largo envuelto alrededor de un octámero que consiste en dos copias de cada una de las proteínas histonas H2A, H2B, H3 y H4 (y ocasionalmente un ligador histona H1 o H5). Si bien la estructura e importancia del empaquetamiento de nivel superior de los nucleosomas es menos conocida, la disposición de nivel inferior y la modificación de los nucleosomas es muy importante para la regulación transcripcional y el desarrollo de diferentes tipos de células. Las proteínas histonas H3 y H4 son las proteínas más conservadas en el dominio eucariota de la vida.

Los nucleosomas codifican la información epigenética de dos formas principales: accesibilidad a la cromatina y modificaciones de histonas.

Primero, las posiciones de los nucleosomas en el ADN determinan qué partes del ADN son accesibles. Los nucleosomas a menudo se posicionan en los promotores de genes inactivos. Para iniciar la transcripción de un gen, los factores de transcripción (TFs) y el complejo de ARN polimerasa tienen que unirse a su promotor. Por lo tanto, cuando un gen se vuelve activo, los nucleosomas localizados en su promotor a menudo se eliminan del promotor para permitir que la ARN polimerasa inicie la transcripción. Por lo tanto, el posicionamiento del nucleosoma en el ADN es estable, pero mutable. Esta propiedad de estabilidad y mutabilidad es un requisito previo para cualquier forma de información epigenética porque las células necesitan mantener la identidad de un tipo celular en particular, sin embargo, ser capaces de cambiar su estado epigenético para responder a las circunstancias ambientales.

La accesibilidad a la cromatina también puede modularse mediante ARN transcripto (específicamente, “ARN potenciador” o ERna) que flota alrededor del núcleo. En particular, Mousavi et al. encontraron en 2013 que ERNAs, que están trangrafiadas en regiones potenciadoras extrágénicas, mejoran la ocupación de ARN pol II (que está limitada por la velocidad por la accesibilidad de la cromatina) y el despliegue de otra maquinaria transcripcional, lo que lleva a una mayor expresión de genes diana distales [9].

En segundo lugar, las histonas contienen colas no estructuradas que sobresalen de los dominios del núcleo globular que comprenden el octámero del nucleosoma. Estas colas pueden sufrir modificaciones postraduccionales como metilación, acetilación y fosforilación, cada una de las cuales afecta la expresión génica. Algunas proteínas involucradas en la regulación transcripcional se unen específicamente a modificaciones de histonas particulares o combinaciones de modificaciones, y reclutan aún más factores de transcripción que potencian o reprimen la expresión de genes cercanos. Así, la “hipótesis del código de histonas” postula que diferentes combinaciones de modificaciones de histonas en loci genómicos específicos codifican la función bio- lógica a través de la regulación transcripcional diferencial. En este modelo, las modificaciones de histonas son análogas a diferentes lectores que marcan secciones de un libro con notas post-it de diferentes colores; las modificaciones de histonas permiten que el mismo genoma sea interpretado (es decir, transcripto) de manera diferente en diferentes momentos y en diferentes tejidos. Hay más de 100 modificaciones distintas de histonas que se han encontrado experimentalmente. Seis de las modificaciones histonas más bien caracterizadas, junto con los anchos de firma típicos de sus apariciones en el genoma y sus supuestos elementos reguladores asociados, se enumeran en la Tabla 19.1. Tenga en cuenta que todas estas modificaciones están en lisinas en H3 y H4. Las modificaciones de H3 y H4 están más bien caracterizadas porque H3 y H4 son las histonas más conservadas (haciendo que las modificaciones de esas histonas tengan más probabilidades de conservar la función reguladora) y porque existen buenos anticuerpos para todas las modificaciones comúnmente observadas de esas histonas.

Las modificaciones de histonas son tan comúnmente referenciadas que se ha desarrollado una taquigrafía para identificarlas. Esta taquigrafía consiste en el nombre de la proteína histona, el residuo de aminoácido en su cola que ha sido modificado y el tipo de modificación realizada a este residuo. Para ilustrar, el cuarto residuo del extremo N de la histona H3, lisina, a menudo se metila en los promotores de genes activos. Esta modificación se describe como H3K4me3 (si se metila tres veces). La primera parte de la

taquigrafía corresponde a la proteína histona, en este caso H3; K4 corresponde al 4º residuo desde el final, en este caso una lisina, y me3 corresponde a la modificación real, la adición de 3 grupos metilo en este caso.

Modificación de histonas	Firma	Elemento regulador asociado
H3K4me1	(amplio) focal	promotores activos/potenciadores
H3K4me3	(amplio) focal	promotores activos/potenciadores
H3K9me3	ancho	regiones reprimidas
H3K27ac	focal	promotores activos/potenciadores
H3K27me3	ancho	regiones reprimidas
H3K36me3	ancho	regiones transcritas

Cuadro 19.1: Seis de las modificaciones de histonas más bien caracterizadas junto con sus anchos de firma típicos y supuestos elementos reguladores asociados. “Focal” indica que cada instancia de la modificación de histonas tiene una firma relativamente estrecha en el genoma (ancho de pico < 5kb) mientras que “ancho” indica firmas amplias.

Un ejemplo de modificaciones epigenéticas que influyen en la función biológica se ve a menudo en las regiones potenciadoras del genoma. A menudo, estas regiones potenciadoras están lejos de los genes y promotores que regulan. El potenciador es capaz de entrar en contacto con un promotor específico por modificación de histonas (acetilación y metilación). Esto hace que el ADN se pliegue sobre sí mismo para poner en contacto el promotor, potenciador y factores de transcripción reclutados, activando el promotor previamente reprimido. Este sistema puede ser muy dinámico de tal manera que a menos de un minuto después de la modificación de la histona la célula mostrará signos de influencia epigenética, mientras que otras modificaciones (principalmente las durante el desarrollo) se mostrarán de manera más lenta. Este es también un ejemplo de cómo ciertos tipos de modificaciones de las histonas pueden ayudarnos a predecir regiones potenciadoras.

Es posible que más de una modificación de histona esté presente en un locus genómico dado, y las modificaciones de histonas pueden actuar de manera cooperativa y competitiva. Incluso es posible que las dos copias de una proteína histona dada dentro del mismo nucleosoma tengan diferentes modificaciones (aunque generalmente los “escritores” de modificación de histonas se localizarán juntos, creando así la misma modificación en ambas copias dentro del nucleosoma). Por lo tanto, es necesario tomar en cuenta simultáneamente todas las modificaciones de histonas en una región genómica para llamar con precisión el estado de cromatina de esa región. Como se describe en la Sección, con la finalización del Proyecto de Hoja de Ruta Epigenoma en 2015, se puede utilizar un robusto modelo oculto de Markov (con modificaciones de histonas como emisiones y estados de cromatina como estados ocultos) para hacerlo.

### ¿Sabías?

Los organismos más simples que tienen modificaciones epigenéticas son las levaduras. La levadura es un organismo unicelular; por lo tanto, las modificaciones epigenéticas no son responsables de la diferenciación celular. A medida que los organismos se vuelven más complejos tienden a tener más modificaciones epigenéticas.

## Herencia Epigenética

El grado en que las características epigenéticas/epigenómicas son heredables es poco conocida y, por lo tanto, es objeto de mucho debate e investigación en curso. En los organismos que se reproducen sexualmente, la mayoría de las modificaciones epigenéticas se pierden durante la meiosis y/o en la fecundación, pero algunas modificaciones a veces se mantienen. Adicionalmente, existen sesgos en las formas en que las marcas epigenéticas paternas versus maternas se eliminan o remodelan durante este proceso. En particular, la metilación del ADN materno a menudo se retiene en la fecundación, mientras que el ADN paterno casi siempre está completamente desmetilado. Además, por razones desconocidas, algunos elementos genómicos, como los satélites centroméricos, tienen más probabilidades de evadir el reinicio epigenético. En los casos en que el borrado epigenómico no ocurre completamente en la meiosis y la fertilización, puede ocurrir la herencia epigenética transgeneracional. Ver en general [4].

Otro mecanismo que probablemente se regirá por la herencia epigenética es el fenómeno de la impronta parental. En la impronta parental, ciertos genes autosómicos se expresan si y sólo si se heredan de la madre de un individuo, y otros genes autosómicos se expresan si y sólo si se heredan del padre de un individuo. Ejemplos son el gen Igf2 en ratones (solo se expresa si se hereda del padre) y el gen H19 en ratones (solo se expresa si se hereda de la madre). No hay cambios en la secuencia de ADN de estos genes, pero se observan grupos metilo adicionales en ciertos nucleótidos dentro de la copia inactivada del gen. Los mecanismos y causalidad de esta impronta son poco entendidos.

---

This page titled [19.2: Información Epigenética en Nucleosomas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [19.2: Epigenetic Information in Nucleosomes](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.3: Ensayos Epigenómicos

### ChIP: un método para determinar dónde se unen las proteínas al ADN o dónde se modifican las histonas

Dada la importancia de la información epigenómica en biología, se han realizado grandes esfuerzos para estudiar señales que cuantifiquen esta información. Un método común para la medición de marcas epigenómicas se llama inmunoprecipitación de cromatina (ChIP). La tecnología ChIP produce fragmentos de ADN cuya ubicación en el genoma denota las posiciones de una modificación histona particular o factor de transcripción. Los procedimientos de ChIP se describen como sigue y se representan en la Figura 19.2:

1. Las células se exponen a un agente reticulante como el formaldehído, lo que hace que se formen enlaces covalentes entre el ADN y sus proteínas unidas (por ejemplo, histonas con modificaciones específicas).
2. El ADN genómico se aísla del núcleo celular.
3. El ADN aislado es cizallado por sonicación o enzimas.
4. Los anticuerpos se cultivan para reconocer una proteína específica, como las involucradas en la modificación de histonas. Los anticuerpos se cultivan exponiendo las proteínas de interés a mamíferos, como cabras o ratas, cuya respuesta inmune provoca entonces la producción de los anticuerpos deseados.
5. Se añaden anticuerpos a la solución para inmunoprecipitar y purificar los complejos.
6. Se invierte la reticulación entre la proteína y el ADN y se purifican los fragmentos de ADN específicos de las marcas epigenéticas.

Después de un experimento ChIP, tenemos secuencias cortas de ADN que corresponden a lugares donde las histonas estaban unidas al ADN. Para identificar la ubicación de estos fragmentos de ADN en el genoma, se pueden hibridarlos con segmentos de ADN conocidos en una matriz o chip génico y visualizarlos con marcas fluorescentes; este método se conoce como Chip-chip. Alternativamente, se puede hacer una secuenciación masiva paralela de próxima generación de estos fragmentos; esto se conoce como CHIP-seq. Este último enfoque, Chip-seq, es un enfoque más nuevo que se usa con mucha más frecuencia. Se prefiere porque tiene un rango dinámico de detección más amplio y evita problemas como la hibridación cruzada en Chip-chip.

Cada etiqueta de secuencia tiene 30 pares de bases de largo. Estas etiquetas se mapean a posiciones únicas en el genoma de referencia de 3 mil millones de bases. El número de lecturas depende de la profundidad de secuenciación, pero normalmente hay del orden de 10 millones de lecturas mapeadas para cada experimento de Chip-seq.

Existe una tubería bastante estándar utilizada para inferir el enriquecimiento de la proteína de interés en cada sitio del genoma dado un conjunto de lecturas de secuenciación cortas de un experimento de ChIP-seq. Primero, los fragmentos de ADN deben mapearse al ADN (llamado mapeo de lectura). A continuación, debemos determinar qué regiones del genoma tienen un enriquecimiento estadísticamente significativo de la proteína de interés (llamada llamada pico). Después de estos pasos de preprocesamiento, podemos construir diferentes modelos supervisados y no supervisados para estudiar los estados de la cromatina y su relación con la función biológica. Nos fijamos en cada uno de estos pasos a su vez.

### Secuenciación de bisulfito: un método para determinar dónde está metilado el ADN

La metilación del ADN fue la primera modificación epigenómica que se descubrió y es un importante regulador transcripcional, ya que la metilación de residuos de citosina en dinucleótidos CpG da como resultado “silenciamiento” o represión de la transcripción. La secuenciación de bisulfito es un método mediante el cual el ADN se trata con bisulfito antes de la secuenciación, permitiendo la determinación precisa de los nucleótidos en los que el ADN había sido metilado. El tratamiento con bisulfito convierte los residuos de citosina no metilados en uracilo, pero no afecta a las cytosinas metiladas. Así, el ADN genómico puede secuenciarse con o sin tratamiento con bisulfito, y las secuencias pueden compararse, y los sitios en los que la citosina no se ha convertido en uracilo en el ADN tratado (o, equivocadamente, sitios en los que hay diferencia generada por bisulfito entre las secuencias tratadas y no tratadas) son sitios en los que la citosina fue metilada. Este análisis supone la conversión completa de residuos de citosina no metilados a uracilo, por lo que la conversión incompleta puede dar como resultado falsos positivos (es decir, nucleótidos identificados como metilados pero que de hecho no estaban metilados) [11].

This page titled [19.3: Ensayos Epigenómicos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [19.3: Epigenomic Assays](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.4: Procesamiento primario de datos de ChIP

### Leer mapeo

El problema del mapeo de lectura busca asignar una lectura dada a la mejor ubicación coincidente en el genoma de referencia. Dada la gran cantidad de lecturas y el tamaño del genoma humano, un requisito común de todos los algoritmos de mapeo de lectura es que sean eficientes tanto en el espacio como en el tiempo. Además, deben permitir desapareamientos debido a errores de secuenciación y SNP.

A partir de conferencias anteriores, conocemos varias formas de realizar mapeo de lecturas: alineación de secuencias ( $O(mn)$  tiempo) y enfoques basados en hash como BLAST, por ejemplo. También existen otros enfoques: coincidencia lineal de cadenas de tiempo ( $O(m+n)$  tiempo) y árboles de sufijos y matrices de sufijos ( $O(m)$  tiempo). Sin embargo, un problema con todas estas técnicas es que tienen un gran requerimiento de memoria (a menudo  $O(mn)$ ). En cambio, se utilizan técnicas de vanguardia basadas en la transformación Burrows-Wheeler [1]. Estos corren en  $O(m)$  tiempo y requieren solo  $O(n)$  espacio.

La transformación de Burrows-Wheeler surgió originalmente de la necesidad de comprimir información. Toma una cadena larga y la reorganiza de una manera que tiene letras repetitivas adyacentes. Esta cadena se puede comprimir porque, por ejemplo, en lugar de escribir 100 A's la computadora ahora solo puede indicar que hay 100 A's en fila. La transformación Burrows-Wheeler también tiene algunas otras propiedades especiales que explotaremos para buscar en tiempo sublineal.

La transformación Burrows-Wheeler crea una cadena transformada única que es más corta que la cadena original. También se puede revertir fácilmente para generar la cadena original, por lo que no se pierde información. La cadena transformada está en orden ordenado, lo que permite una búsqueda fácil. Los detalles de la transformación Burrows-Wheeler se describen a continuación y se ilustran en la Figura 19.3.

Primero, producimos una transformación a partir de una cadena original mediante los siguientes pasos. En particular, producimos una transformación del genoma de referencia.

1. Para un genoma de referencia dado, agregue un carácter especial al principio y al final de la cadena (por ejemplo, “BANANA” se convierte en ^BANANA@). Luego genere todas las rotaciones de esta cadena (por ejemplo, una de esas rotaciones sería NANA@^BA).
2. Ordena las rotaciones lexicográficamente —es decir, en orden alfabetico— con caracteres especiales ordenados por último.
3. Conservar únicamente la última columna de la lista ordenada de rotaciones. Esta columna contiene la cadena transformada

Una vez que se ha calculado una transformada de Burrows-Wheeler, es posible revertir la transformación para calcular la cadena original. Esto se puede hacer con el procedimiento en la Figura???. Brevemente, la transformación inversa funciona de la siguiente manera: dada la cadena transformada, ordena los caracteres de cadena en orden alfabetico; esto da la primera columna de la transformación. Combina la última columna con la primera para obtener pares de caracteres de las rotaciones originales. Ordenar los pares y repetir.

Mediante el uso de punteros de clasificación en lugar de cadenas completas, es posible generar esta transformación del genoma de referencia utilizando un espacio que es lineal en su tamaño. Además, incluso con un número muy grande de lecturas, sólo es necesario hacer la transformada en una dirección hacia adelante. Después de contar las lecturas en el espacio transformado, entonces solo es necesario hacer la transformación inversa una vez para mapear los recuentos a las coordenadas del genoma.

En particular, a partir de la transformación de Burrows-Wheeler observamos que todas las ocurrencias del mismo sufijo están efectivamente una al lado de la otra en lugar de dispersas por todo el genoma. Además, la iésima ocurrencia de un carácter en la primera columna corresponde a la iésima ocurrencia en la última columna. La búsqueda de subcadenas usando la transformación también es fácil. Supongamos que estamos buscando la subcadena “ANA” en la cadena dada. Entonces el problema de la búsqueda se reduce a buscar un prefijo “ANA” entre todos los sufijos ordenados posibles (generados por rotaciones). La última letra de la subcadena (“A”) se busca primero en las primeras letras de las rotaciones ordenadas. Entonces, se consideran las rotaciones de una letra de estos partidos; se buscan las dos últimas letras de la subcadena (“NA”) entre las dos primeras letras de estas rotaciones de una letra. Este proceso se puede continuar con sufijos de longitud crecientes para encontrar la subcadena como prefijo de una rotación. Específicamente, cada lectura es buscada y se encuentra como prefijo de una rotación del genoma de referencia; esto da la

posición de la lectura en el genoma. Al hacer una transformación inversa, es posible encontrar las coordenadas genómicas de las lecturas mapeadas.

Tenga en cuenta que esta idea no es más rápida en teoría que el hash, pero puede ser más rápida en la práctica porque usa una huella de memoria más pequeña.

## Métricas de control de calidad

Al igual que con todos los datos experimentales, los métodos ChIP contienen sesgos y su producción puede ser de calidad variada. En consecuencia, antes de procesar los datos, es necesario controlar para estos sesgos, determinar qué lecturas en los datos alcanzan cierto nivel de calidad, y establecer umbrales objetivo sobre la calidad del conjunto de datos en su conjunto. En esta sección describiremos estos problemas de control de calidad y métricas asociadas a ellos.

### QC1: Uso de ADN de entrada como control

Primero, las lecturas dadas por ChIP no están dispersas uniformemente en el genoma. Por ejemplo, las regiones accesibles del genoma pueden fragmentarse más fácilmente, lo que lleva a una fragmentación no uniforme. Para controlar este sesgo, podemos ejecutar el experimento ChIP en la misma porción de ADN sin usar un anticuerpo. Esto produce ADN de entrada, que luego se puede fragmentar y mapear para dar una pista de señal que se puede considerar como un fondo, es decir, lecturas que esperaríamos por casualidad. (En efecto, incluso en el fondo no vemos uniformidad.) Adicionalmente, tenemos una pista de señal para el experimento verdadero, que proviene del ADN cromo-inmunoprecipitado. Se muestra en la Figura 19.4

### QC2: Umbral de puntuación de calidad de secuenciación de nivel de lectura

Al secuenciar ADN, cada par de bases se asocia con una puntuación de calidad. Por lo tanto, las lecturas dadas por ChIP- seq contienen puntuaciones de calidad en el nivel de pares base, donde las puntuaciones de menor calidad implican una mayor probabilidad de mapeos erróneos. Podemos usar fácilmente esta información en un paso de preprocessamiento simplemente rechazando cualquier lectura cuyo puntaje de calidad promedio caiga por debajo de algún umbral (por ejemplo, solo use lecturas donde Q, el puntaje de calidad promedio, sea mayor que 10). Se muestra en la Figura 19.5

### QC3: Fracción de lecturas cortas mapeadas

Cada lectura que pase la métrica de calidad anterior puede mapearse exactamente a una ubicación en el genoma, a múltiples ubicaciones o a ninguna ubicación en absoluto. Cuando se lee el mapa a múltiples ubicaciones, hay una serie de enfoques para manejar esto:

- Un enfoque conservador: No asignamos las lecturas a ningún lugar porque somos muy inciertos. Con: podemos perder señal
- Un enfoque probabilístico: Asignamos fraccionalmente las lecturas a todas las ubicaciones. Con: puede agregar artefactos (picos irreales)
- Un enfoque de muestreo: Solo seleccionamos una ubicación al azar para una lectura. Lo más probable es que, a través de muchas lecturas, las asignemos de manera uniforme. Con: puede agregar artefactos (picos irreales)
- Un enfoque EM: Podemos mapear lecturas basadas en la densidad de lecturas inequívocas. Es decir, muchas lecturas únicas que mapean a una región dan una alta probabilidad previa de que una lectura mapee a esa región. Nota: debemos hacer la suposición de que las densidades son constantes dentro de cada región
- Un enfoque de extremos emparejados: Debido a que secuenciamos ambos extremos de un fragmento de ADN, si conocemos el mapeo de la lectura desde un extremo, podemos determinar el mapeo de la lectura en el otro extremo aunque sea ambiguo.

De cualquier manera, probablemente habrá lecturas que no mapeen al genoma. Una métrica de control de calidad estaría considerando la fracción de lecturas de ese mapa; podemos establecer un objetivo del 50%, por ejemplo. De igual manera, puede haber regiones a las que no se lee mapa. Esto puede deberse a una falta de cobertura de ensamblaje o a demasiadas lecturas mapeadas a la región; tratamos las regiones no mapeables como datos faltantes.

### QC4: Análisis de correlación cruzada

Un control de calidad adicional que es el análisis de correlación cruzada. Si se emplean lecturas de un solo extremo, la proteína de unión a ADN generará un pico de lecturas que mapean el desplazamiento de la cadena directa a una distancia aproximadamente igual a la longitud del fragmento de ADN desde un pico de lecturas que mapean a la cadena inversa. Un patrón similar se genera a partir de lecturas finales emparejadas, en las que los extremos de lectura caen en dos grupos con un desplazamiento dado, un

extremo de lectura mapeará a la hebra hacia adelante y el otro a la hebra inversa. La longitud promedio del fragmento se puede inferir calculando la correlación entre el número de lecturas que mapean a la cadena directa y el número de lecturas que se mapean a la cadena inversa como una función de la distancia entre las lecturas directa e inversa. La correlación alcanzará su pico en la longitud media del fragmento.

El análisis de correlación cruzada también proporciona información sobre la calidad del conjunto de datos ChIP-seq. El ADN de entrada no debe contener ningún pico real, pero a menudo muestra una fuerte correlación cruzada a una distancia igual a la longitud de lectura. Esto ocurre porque algunas lecturas mapean de manera única entre regiones que no se pueden mapear. Si una lectura puede mapear de manera única en la posición  $x$  entre dos regiones no mapeables en la cadena directa, entonces una lectura también puede mapear únicamente a la hebra inversa en la posición  $x + r - 1$ , donde  $r$  es la longitud de lectura. Las lecturas de ese mapa de esta manera generan la fuerte correlación cruzada a distancia igual a la longitud de lectura en el ADN de entrada. Si un experimento de ChIP-seq no tuvo éxito y no enriqueció significativamente para la proteína de interés, entonces un gran componente de las lecturas será similar a la entrada no enriquecida, lo que producirá un pico en la correlación cruzada a la longitud de lectura. Por lo tanto, la fuerza de la correlación cruzada a la longitud de lectura en relación con la fuerza a la longitud del fragmento se puede utilizar para evaluar la calidad del conjunto de datos de ChIP-seq. Las bibliotecas de ChIP-seq aceptables deben tener una correlación cruzada a la longitud del fragmento al menos tan alta como a la longitud de lectura, y cuanto mayor sea la relación entre la correlación cruzada de longitud de fragmento y la correlación cruzada de longitud de lectura, mejor.

#### QC5: Complejidad de Biblioteca

Como métrica final de control de calidad, podemos considerar la complejidad de la biblioteca, o la fracción de lecturas que no son redundantes. En una región con señal, podríamos esperar que las lecturas provengan de todas las posiciones de esa región; sin embargo, a veces vemos que solo un pequeño número de posiciones en una región tienen lecturas mapeadas a ellas. Esto puede ser el resultado de un artefacto de amplificación en el que una sola lectura amplifica mucho más de lo que debería. En consecuencia, consideraremos la fracción no redundante de una biblioteca:

$$\text{NRF} = \frac{\text{No. of distinct unique-mapping reads}}{\text{No. of unique mapping reads}}$$

Este valor mide la complejidad de la biblioteca. Los valores bajos indican baja complejidad, lo que puede ocurrir, por ejemplo, cuando no hay suficiente ADN o un fragmento de ADN está sobresecuenciado. Cuando se trabaja con al menos 10 millones de lecturas mapeadas de forma única, normalmente establecemos un objetivo de al menos 0.8 para la NRF.

#### Llamadas pico y selección

Después de alinear las lecturas, se pueden generar pistas de señal como se muestra en la Figura 19.6. Estos datos se pueden ordenar en un histograma largo que abarca la longitud del genoma, lo que corresponde al número de lecturas (o grado de fluorescencia en el caso de Chip-chip) que se encuentran en cada posición del genoma. Más lecturas (o fluorescencia) sugieren una presencia más fuerte del marcador epigenético de interés en esta ubicación particular.

En particular, para generar estas pistas de señal transformamos los recuentos leídos en una señal de intensidad normalizada. Primero, podemos usar el análisis de correlación cruzada de cadenas para estimar la distribución de longitud de fragmento  $f$ . Como ahora conocemos  $f$ , así como la longitud de cada lectura, podemos extender cada lectura (típicamente solo 36 pb) desde la dirección 5' a 3' para que su longitud sea igual a la longitud promedio del fragmento. Entonces, en lugar de simplemente sumar la intensidad de cada base en las lecturas originales, podemos sumar la intensidad de cada base en las lecturas extendidas de ambas hebras. En otras palabras, a pesar de que solo secuenciamos una pequeña lectura, podemos usar información sobre un segmento completo del cual esa lectura forma parte. Podemos hacer esta misma operación sobre los datos de control. Esto produce pistas de señal tanto para el experimento verdadero como para el control, como se muestra en la Figura 19.7.

Para procesar los datos, primero estamos interesados en usar estas pistas de señales para descubrir regiones (es decir, intervalos discretos) de enriquecimiento. Este es el objetivo del pico de llamadas. Hay muchos programas que realizan llamadas pico con diferentes enfoques. Por ejemplo, MACS utiliza una distribución local de Poisson como modelo estadístico, mientras que PeakSeq utiliza un modelo binomial condicional.

Una forma de modelar la distribución de conteo de lecturas es con una distribución de Poisson. Podemos estimar el recuento esperado de lecturas,  $\lambda_{\text{local}}$  a partir de los datos de control. Entonces,

$$\Pr(\text{ count } = x) = \frac{\lambda_{\text{local}}^x e^{-\lambda_{\text{local}}}}{x!}$$

Así, el valor p de Poisson para un recuento leído x viene dado por  $\Pr(\text{count} \geq x)$ . Especificamos un valor p umbral (por ejemplo, 0.00001) por debajo del cual las regiones genómicas se consideran picos.

Podemos transformar este valor p en una tasa empírica de falsos descubrimientos, o eFDR, intercambiando los datos del experimento ChIP (true) con las pistas de ADN de entrada (control). Esto produciría las ubicaciones en el genoma donde la señal de fondo es mayor que la señal de ChIP. Para cada valor p, podemos encontrar tanto a partir de los datos de ChIP como de los datos de control. Entonces, para cada valor p, el eFDR es simplemente el número de picos de control dividido por el número de picos de ChIP. Con esto, entonces podemos elegir a qué picos llamar en función de un umbral de eFDR.

Un problema importante que surge es que no se puede usar un único eFDR universal o umbral de valor p. Los umbrales ideales dependen de una variedad de factores, incluyendo el ChIP, la profundidad de secuenciación y la ubicuidad del factor objetivo. Además, pequeños cambios en el umbral de eFDR pueden producir cambios muy grandes en los picos que se descubren. Una medida alternativa es la tasa de descubrimiento irreproducible, o IDR, y esta medida evita estos problemas específicos de FDR.

### Tasa de Descubrimiento Irreducible (IDR)

Un inconveniente importante del uso de métodos estadísticos tradicionales para evaluar la significación de los picos de Chip-seq es que los enfoques basados en el valor de FDR y p hacen suposiciones particulares con respecto a la relación entre enriquecimiento y significación. Evaluar la importancia de los picos de ChIP usando IDR en lugar de un valor p o FDR es ventajoso porque nos permite aprovechar la información presente en réplicas biológicas para llamar picos sin establecer un umbral de significancia. Los enfoques basados en IDR se basan en la idea de que es probable que la señal real sea reproducible entre réplicas, mientras que el ruido no debe ser reproducible. El uso de IDR para llamar a picos significativos devuelve picos que satisfacen un umbral dado de significancia. Para determinar qué picos son significativos a través de IDR, los picos en cada réplica biológica se clasifican en función de su enriquecimiento en orden descendente. Los picos de N superiores en cada réplica se comparan entre sí, y el IDR para una réplica dada es la fracción de picos presentes en los picos de N superiores en el replicar que no están presentes en las otras réplicas (es decir, la fracción de picos que no son reproducibles entre réplicas). Para desarrollar más intuición matemática, la siguiente subsección (totalmente opcional) introducirá rigurosamente el concepto del IDR.

### Derivación matemática del IDR

Dado que el IDR utiliza rangos, esto significa que las distribuciones marginales son uniformes, y la información se codifica principalmente en las distribuciones conjuntas de los rangos a través de réplicas biológicas. Específicamente, cuando las distribuciones marginales son uniformes, podemos modelar las distribuciones conjuntas a través de un modelo de cópula. En pocas palabras, una cópula es una distribución de probabilidad multivariada en la que la probabilidad marginal de cada variable es uniforme. **El Teorema de Skar** afirma que existe al menos una función cópula que nos permite expresar la articulación en términos de la dependencia de las distribuciones marginales.

$$F_k(x_1, x_2, \dots, x_k) = C_x(F_{X_1}(x_1), \dots, F_{X_k}(x_k))$$

Donde  $C_x$  es la función cópula y la  $F(x)$  es la distribución acumulativa para una variable x. Dada esta información, podemos establecer una distribución de Bernoulli  $K_i \sim \text{Bern}(\pi_i)$  que denota si el i-ésimo pico es del conjunto consistente o del conjunto espurio. Podemos derivar  $z_1 = (z_{1,1}, z_{1,2})$  si  $K_i = 1$  o  $z_0 = (z_{0,1}, z_{0,2})$  si  $K_i = 0$  (donde  $z_{0,i}$  significa que es del conjunto espurio en replicado biológico i). Usando esto, podemos modelar los modelos  $z_{1,1}$  y  $z_{0,1}$  de la siguiente manera:

```
\left[ \begin{array}{c}
z_{i,1} \\
z_{i,2}
\end{array} \right] \mid K_i = k \sim N \left[ \begin{array}{c}
\mu_k \\
\mu_k
\end{array} \right], \rho \sigma_k^2 \left( \begin{array}{cc}
1 & \rho \\
\rho & 1
\end{array} \right)
```

```
\rho_k \sigma_k^2 & \sigma_k^2
\end{array}\derecha)\derecha\nonumber]
```

Podemos utilizar dos modelos diferentes para modelar si proviene del conjunto espurio (denotado por 0), o del conjunto real (1). Si el conjunto real, tenemos  $\mu_1 > 0$  y  $0 < \rho_1 < 1$ , donde como en el conjunto nulo tenemos  $\mu_0 = 0$ , y  $\sigma_0^2 = 1$ . Podemos modelar una variable  $u_{i,1}$  y  $u_{i,2}$  con las siguientes fórmulas:

$$u_{i,1} = G(z_{i,1}) = \pi_1 \Phi\left(\frac{z_{i,1} - \mu_1}{\sigma_1}\right) + \pi_0 \Phi(z_{i,1})$$

$$u_{i,2} = G(z_{i,2}) = \pi_1 \Phi\left(\frac{z_{i,2} - \mu_1}{\sigma_1}\right) + \pi_0 \Phi(z_{i,2})$$

Donde  $\Phi$  es la función de distribución acumulativa normal. Entonces, dejemos que las  $x_{i,1} = F^{-1}(u_{i,1})$  y  $x_{i,2} = F^{-1}(u_{i,2})$ ,  $F_1$  y  $F_2$  observadas sean las distribuciones marginales de las dos coordenadas. Así, para una señal  $i$ , tenemos:

$$= \pi_0 h_0(G^{-1}(F_1(x_{i,1})), G^{-1}(F_2(x_{i,2}))) + \pi_1 h_1(G^{-1}(F_1(x_{i,1})), G^{-1}(F_2(x_{i,2})))$$

Podemos expresar  $h_0$  y  $h_1$  con las siguientes distribuciones normales, similares a las  $z_1$  y  $z_2$  que se definieron anteriormente:

```

\begin{aligned}
& h_0 \sim N(\left( \begin{array}{l} 0 \\
0 \\
\end{array} \right) \left( \begin{array}{ll} 1 & 0 \\
0 & 1 \end{array} \right) \left( \begin{array}{c} \mu_1 \\
\mu_2 \end{array} \right), \left( \begin{array}{cc} \sigma_1^2 & \sigma_{12}^2 \\
\sigma_{12}^2 & \sigma_2^2 \end{array} \right))
\end{aligned}

```

podemos inferir los parámetros  $\theta = (\mu_1, \rho_1, \sigma_1, \pi_0)$ , usando un algoritmo EM, donde la inferencia se basa en  $P(K_i = 1 | (x_{i,1}, x_{i,2}); \hat{\theta})$ . Así, podemos definir la tasa de descubrimiento irreproducible local como:

$$\text{idr}(x_{i,1}, x_{i,2}) = P(K_i = 0 \mid (x_{i,1}, x_{i,2}); \hat{\theta})$$

Entonces para controlar el IDR en algún nivel \alpha, podemos clasificar  $(x_{i,1}, x_{i,2})$  por sus valores IDR. Entonces podemos seleccionar  $(x_{(i),1}, x_{(i),2})$ ,  $i = 1, l$ , donde

$$I = \operatorname{argmax}_i \frac{1}{i} \sum_{j=1}^i idr_j \leq \alpha$$

La IDR es análoga a un control de FDR en este modelo de mezcla de cópula. Esta subsección resume la información proporcionada en esta conferencia: [www.biostat.wisc.edu/~kendzi... AT877/SK\\_2.pdf](http://www.biostat.wisc.edu/~kendzi... AT877/SK_2.pdf). El artículo original, junto con una formulación aún más detallada de IDR, se puede encontrar en Li et al. [10].

## **Ventajas y casos de uso del IDR**

El análisis de IDR se puede realizar con N creciente, hasta que se alcanza la IDR deseada (por ejemplo, N se incrementa hasta IDR=0.05, lo que significa que 5% de los picos de N superiores no son reproducibles). Tenga en cuenta que N puede ser diferente

para diferentes réplicas del mismo experimento, ya que algunas réplicas pueden ser más reproducibles que otras debido a artefactos técnicos o biológicos.

IDR también es superior a enfoques más simples para usar la reproducibilidad entre experimentos para definir la significación. Un enfoque podría ser tomar la unión de todos los picos en ambas réplicas como significativa, sin embargo; este método aceptará tanto picos reales como el ruido en cada conjunto de datos. Otro enfoque es tomar la intersección de picos en ambas réplicas, es decir, solo contar picos presentes en ambos conjuntos de datos como significativos. Si bien este método eliminará de manera muy efectiva los picos espurios, es probable que pierda muchos picos genuinos. Se puede pensar que la IDR combina ambos enfoques, ya que acepta todos los picos, independientemente de si son reproducibles, siempre y cuando los picos tengan suficiente enriquecimiento para caer dentro del segmento de los datos con una tasa de irreproducibilidad global por encima de un umbral dado. Otra ventaja de la IDR es que aún se puede realizar aunque no se disponga de réplicas biológicas, lo que a menudo puede ser el caso de los experimentos de ChIP realizados en tipos de células raras. Las réplicas de PSUDO se pueden generar a partir de un único conjunto de datos asignando aleatoriamente la mitad de las lecturas a una pseudo-réplica y la mitad a otra pseudo-réplica.

### Interpretación de marcas de cromatina

Ahora pasamos a técnicas para interpretar las marcas de cromatina. Hay muchas formas de analizar las marcas epigenómicas, como agregar señales de cromatina (por ejemplo, H3K4me3) en tipos de características conocidas (por ejemplo, promotores de genes con niveles de expresión altos o bajos) y realizar métodos de aprendizaje automático supervisados o no supervisados para derivar características epigenómicas que predicen diferentes tipos de elementos genómicos como promotores, potenciadores o grandes ARN intergénicos no codificantes. En particular, en esta conferencia, examinamos en detalle el análisis de las marcas de cromatina tal como se hace en [7].

---

This page titled [19.4: Procesamiento primario de datos de ChIP](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [19.4: Primary data processing of ChIP data](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.5: Anotar el genoma usando firmas de cromatina

La hipótesis del código histona sugiere que las interacciones cromatina-ADN están guiadas por modificaciones combinatorias de sus tonos. Estas modificaciones combinatorias, cuando se toman juntas, pueden determinar en parte cómo una región de ADN es interpretada por la célula (es decir, como un dominio de unión al factor de transcripción, un sitio de corte y empalme, una región potenciadora, un gen expresado activamente, un gen reprimido o una región no funcional). Nos interesa interpretar este “código” (es decir, determinar a partir de marcas de histonas en una región si la región es un sitio de inicio de la transcripción, potenciador, promotor, etc.). Con una comprensión de las marcas de histonas combinatorias, podemos anotar el genoma en regiones funcionales y predecir nuevos potenciadores, promotores, genes, etc. El desafío es que hay docenas de marcas y exhiben efectos combinatorios complejos.

Dicho de otra manera, el ADN puede tomar una serie de estados (ocultos) (codificantes, no codificantes, etc.). Cada uno de estos estados emite una combinación específica de modificaciones epigenéticas (H3K4me3, H3K36me3, etc) que la célula reconoce. Queremos poder predecir estos estados ocultos y biológicamente relevantes a partir de modificaciones epigenéticas observadas.

En esta sección, exploramos una técnica para interpretar el “código” y su aplicación a un conjunto de datos específico [7], que midió 41 marcas de cromatina en todo el genoma humano.

### Datos

Los datos para este análisis consistieron en 41 marcas de cromatina incluyendo acetilaciones, metilaciones, H2AZ, CTCF y PolLi en células T CD4. Primero, el genoma se dividió en bins no superpuestos de 200 pb en los que se determinó la ausencia binaria o presencia de cada una de las 41 marcas de cromatina. Estos datos se procesaron mediante binarización de datos, en la que a cada marca en cada intervalo se le asigna un valor de 0 o 1 dependiendo de si el enriquecimiento de la señal de la marca en ese intervalo excede un umbral. Específicamente, sea  $C_{ij}$  el número de lecturas detectadas por Chip-seq para la marca  $i$ , mapeando al bin  $j$  de 200 pb. Sea  $\lambda_i$  el número promedio de lecturas que mapean a un bin para la marca  $i$ . Se determina que la marca  $i$  está presente en el bin  $j$  si  $P(X > C_{ij})$  es menor que el umbral aceptado de  $10^{-4}$  donde  $X$  es una variable aleatoria de Poisson con media  $\lambda_i$  y ausente de otra manera. El umbral es definido por el usuario, similar a un valor  $p$  de Poisson. En palabras de orden, el enriquecimiento de lectura para un bin específico tiene que ser significativamente mayor que un proceso aleatorio de poner lecturas en bins. Un ejemplo de estados de cromatina alrededor del gen CAPZA2 en el cromosoma 7 se muestra en la Figura 19.8. Entonces de esta manera, para cada marca  $i$ , podemos etiquetar cada bin  $j$  con un 1 si la marca está presente y un 0 si no lo está. Mirando los datos como un todo, podemos pensarlo como una matriz binaria grande, donde cada fila corresponde a una marca y cada columna corresponde a un bin (que es simplemente una región de 200bp del genoma).

Los datos adicionales utilizados para el análisis incluyeron datos de ontología génica, datos de SNP, datos de expresión y otros.

### HMMs para anotación de estado de cromatina

Nuestro objetivo es identificar combinaciones biológicamente significativas y espacialmente coherentes de marcas de cromatina. Recuerde que dividimos el genoma en bloques de 200bp, así que por espacialmente coherente queremos decir que si tenemos un elemento genómico que es mayor que 200bps, esperamos que la combinación de marcas de cromatina sea consistente en cada bin de 200bp en la región. Llamaremos a estas combinaciones biológicamente significativas y espacialmente coherentes de marcas de cromatina **estados de cromatina**. En conferencias anteriores, hemos visto HMM aplicados a la anotación genómica para genes e islas CpG. Nos gustaría aplicar las mismas ideas a esta situación, pero en este caso, no conocemos los estados ocultos a priori (por ejemplo, región insular CpG o no), nos gustaría aprenderlos de novo. Este modelo puede capturar tanto el orden funcional de diferentes estados (por ejemplo, desde el promotor hasta las regiones transcritas) como la propagación de ciertos dominios de cromatina a través de los genomas. Para resumir, queremos aprender un HMM donde los estados ocultos del HMM son estados de cromatina.

Como aprendimos anteriormente, incluso si no conocemos las probabilidades de emisión y las probabilidades de transición de un HMM, podemos usar el algoritmo de entrenamiento Baum-Welch para aprender los valores de máxima verosimilitud para esos parámetros. En nuestro caso, tenemos una dificultad añadida, ¡ni siquiera sabemos cuántos estados de cromatina existen! En las siguientes subsecciones, ampliaremos cómo se modelan los datos y cómo podemos elegir el número de estados para el HMM.

#### Emisión de un Vector

En los HMM de conferencias anteriores, cada estado emitía ya sea un solo nucleótido o una sola cadena de nucleótidos a la vez. En el HMM para este problema, cada estado emite una combinación de marcas epigenéticas. Cada combinación se puede representar como un vector *n-dimensional* donde n es el número de marcas de cromatina que se están analizando (n = 41 para nuestros datos). Por ejemplo, suponiendo que tenga cuatro posibles modificaciones epigenéticas: H3K4me3, H2BK5ac, Methyl-C y Methyl-A, una secuencia que contiene H3K4me3 y Methyl-C podría presentarse como el vector (1, 0, 1, 0). Uno podría imaginar muchas distribuciones de probabilidad diferentes en vectores n binarios y, por simplicidad, asumimos que las marcas son independientes y se modelan como variables aleatorias de Bernoulli. Entonces estamos asumiendo que las marcas son independientes dado el estado oculto del HMM (tenga en cuenta que esto no es lo mismo que asumir que las marcas son independientes).

Si hay n marcas de entrada, cada estado k tiene un vector ( $p_{k1}, \dots, p_{kn}$ ) de probabilidades de observar marcas 1 a n. Dado que la probabilidad se modela como un conjunto de variables aleatorias independientes de Bernoulli, la probabilidad de observar un conjunto de marcas dado que estamos en el estado oculto k es igual al producto de las probabilidades de observar marcas individuales. Por ejemplo si n = 4, las marcas observadas en bin j fueron (1, 0, 1, 0) y estábamos en estado k, entonces la probabilidad de que esos datos sean  $p_{k1}(1-p_{k2})p_{k3}(1-p_{k4})$ .

Las probabilidades de emisión aprendidas para los datos se muestran en la Figura 19.9.

### Probabilidades de transición

Recordemos que las probabilidades de transición representan la frecuencia de transición de un estado oculto a otro estado oculto. En este caso, nuestros estados ocultos son estados de cromatina. La matriz de transición para nuestros datos se muestra en la Figura 19.10. Como se ve en la figura, la matriz es escasa, lo que indica que solo algunas de las posibles transiciones ocurren realmente. La matriz de transición revela las relaciones espaciales entre estados vecinos. Los bloques de estados en la matriz revelan subgrupos de estados y a partir de estos bloques de nivel superior, podemos ver transiciones entre estos metaestados.

### Elegir el número de estados a modelar

Como ocurre con la mayoría de los algoritmos de aprendizaje automático, aumentar la complejidad del modelo (por ejemplo, el número de estados ocultos) permitirá que se ajuste mejor a los datos de entrenamiento. Sin embargo, los datos de entrenamiento son sólo una muestra limitada de la población real. A medida que agregamos más complejidad, en algún momento estamos ajustando patrones en los datos de entrenamiento que solo existen por muestreo limitado, para que el modelo no se generalice a la población verdadera. Esto se llama **sobreajuste** de datos de entrenamiento; debemos dejar de agregar complejidad al modelo antes de que se ajuste al ruido en los datos de entrenamiento.

El **Criterio de Información Bayesiana (BIC)** es una técnica común para optimizar la complejidad de un modelo que equilibra el ajuste aumentado a los datos con la complejidad del modelo. Usando BIC, podemos visualizar la potencia creciente del HMM en función del número de estados. Generalmente, se elegirá un valor para  $k$  (el número de estados) tal que la adición de más estados tenga relativamente poco beneficio en términos de ganancia de potencia predictiva. Sin embargo, existe una compensación entre la complejidad del modelo y la interpretabilidad del modelo que BIC no puede ayudar con. Es probable que el modelo óptimo según BIC tenga más estados que un modelo ideal porque estamos dispuestos a intercambiar algún poder predictivo por un modelo con menos estados que pueda interpretarse biológicamente. El genoma humano es tan grande y las marcas de cromatina tan complejas que las diferencias estadísticamente significativas son fáciles de encontrar, sin embargo, muchas de estas diferencias no son biológicamente significativas.

Para resolver este problema, comenzamos con un modelo con más estados ocultos de los que creemos necesarios y podamos estados ocultos siempre y cuando todos los estados de interés en el modelo más grande sean capturados adecuadamente. El algoritmo Baum-Welch (y EM en general) es sensible a las condiciones iniciales, por lo que intentamos varias inicializaciones aleatorias en nuestro aprendizaje. Por cada número de estados ocultos de 2 a 80, generamos tres inicializaciones aleatorias de los parámetros y entrenamos el modelo usando Baum-Welch. El mejor modelo según BIC tuvo 79 estados y luego los estados se eliminaron iterativamente de este conjunto de 79 estados.

Como mencionamos anteriormente, Baum-Welch es sensible a los parámetros iniciales, por lo que cuando podamos estados, usamos una inicialización anidada en lugar de una inicialización aleatoria para el modelo podado. Específicamente, los estados fueron removidos con avidez del modelo BIC-Optimal 79 estados. El estado a eliminar fue el estado que tal que todos los estados de los 237 modelos inicializados aleatoriamente fueron bien capturados. Al eliminar un estado, se eliminarían las probabilidades de

emisión y cualquier estado que pasara al estado eliminado tendría esa probabilidad de transición redistribuida uniformemente a los estados restantes. Esto se utilizó como la inicialización al entrenamiento de Baum-Welch. El número de estados para que un modelo analice se puede seleccionar eligiendo el modelo entrenado a partir de dicha inicialización anidada con el menor número de estados que capta suavemente todos los estados ofreciendo distintas interpretaciones biológicas. El modelo final resultante tuvo 51 estados.

También podemos verificar el ajuste del modelo observando cómo los datos violan los supuestos del modelo. Dado el estado oculto, el HMM asume que cada marca es independiente. Podemos probar qué tan bien se ajustan los datos a esta suposición trazando la dependencia entre marcas. Esto puede revelar estados que encajan bien y aquellos que no. En particular, los estados repetitivos revelan un caso donde el modelo no encaja bien. A medida que agregamos más estados, el modelo es más capaz de ajustarse a los datos y, por lo tanto, ajustarse a las dependencias. Al monitorear el ajuste en estados individuales que nos interesan, podemos controlar la complejidad del modelo.

## Resultados

Este modelo multivariado HMM resultó en un conjunto de 51 estados de cromatina biológicamente relevantes. Sin embargo, no hubo relación uno a uno entre cada estado y clases conocidas de elementos genómicos (por ejemplo, intrones, exones, promotores, potenciadores, etc.) En cambio, múltiples estados de cromatina a menudo se asociaron con un elemento genómico. Cada estado de cromatina codificaba información biológica específica relevante sobre su elemento genómico asociado. Por ejemplo, tres estados de cromatina diferentes se asociaron con el sitio de inicio de la transcripción (TSS), pero uno se asoció con TSS de genes altamente expresados, mientras que los otros dos se asociaron con TSS de genes de expresión media y baja respectivamente. Dicho uso de marcadores epigenéticos mejoró enormemente la anotación genómica, particularmente cuando se combina con señales evolutivas discutidas en conferencias anteriores. Los 51 estados de cromatina se pueden dividir en cinco grandes grupos. Las propiedades de estos grupos se describen de la siguiente manera y se ilustran adicionalmente en 19.11:

### 1. Estados Asociados a los Promotores (1-11):

Todos estos estados de cromatina tuvieron un alto enriquecimiento para las regiones promotoras. 40-89% de cada estado estuvo dentro de 2 kb de un TSS RefSeq en comparación con 2.7% de ancho del genoma. Todos estos estados tuvieron una alta frecuencia de H3K4me3, enriquecimientos significativos para sitios hipersensibles a la DNase I, islas CpG, motivos evolutivamente servidos y factores de transcripción unidos. Sin embargo, estos estados diferían en los niveles de marcas asociadas como H3K79me2/3, H4K20me1, acetilaciones, etc. Estos estados también disminuyeron en su enriquecimiento funcional basado en Ontología Génica (GO). Por ejemplo, los genes asociados con la activación de células T se enriquecieron en el estado 8 mientras que los genes asociados con el desarrollo embrionario se enriquecieron en el estado 1. Adicionalmente, entre estos estados promotores hubo distintos enriquecimientos posicionales. Los estados 1-3 alcanzaron su pico tanto aguas arriba como aguas abajo del TSS; los estados 4-7 se concentraron justo sobre el SST mientras que los estados 8-11 alcanzaron un pico entre 400 pb y 1200 pb aguas abajo del SST. Esto sugiere que las marcas de cromatina pueden reclutar factores de iniciación y que el acto de transcripción puede reforzar estas marcas. El enriquecimiento funcional distinto también sugiere que las marcas codifican una historia de activación.

### 2. Estados Asociados a la Transcripción (12-28):

Este fue el segundo grupo más grande de estados de cromatina e incluyó 17 estados asociados a la transcripción. Hay 70-95% contenido en regiones transcritas anotadas en comparación con 36% para el resto del genoma. Estos estados no se asociaron predominantemente con una sola marca sino que se definieron por una combinación de siete marcas: H3K79me3, H3K79me2, H3K79me1, H3K27me1, H2BK5me1, H4K20me1 y H3K36me3. Estos estados tienen subgrupos asociados con ubicaciones 5' proximales o 5' distales. Algunos de estos estados se asociaron con exones empalmados, sitios de inicio de la transcripción o sitios finales. De interés, el estado 28, que se caracterizó por alta frecuencia para H3K9me3, H4K20me3 y H3K36me3, mostró un alto enriquecimiento en genes de dedos de zinc. Esta combinación específica de marcas se reportó previamente como regiones marcadoras de unión a KAP1, un co-represor específico de dedos de zinc.

### 3. Estados intergénicos activos (29-39):

Estos estados se asociaron con varias clases de regiones potenciadoras candidatas y regiones aislantes y se asociaron con frecuencias más altas para H3K4me1, H2AZ, varias marcas de acetilación pero frecuencias más bajas de marcas de

metilación. Además, las marcas de cromatina podrían usarse para distinguir los potenciadores activos de los menos activos. Estas regiones generalmente estaban alejadas de los promotores y estaban fuera de los genes transcritos. Curiosamente, varios estados intergénicos activos mostraron un enriquecimiento significativo para SNP de la enfermedad, o polimorfismo de un solo nucleótido en el estudio de asociación de todo el genoma (GWAS). Por ejemplo, se encontró que un SNP (rs12619285) asociado a los niveles plasmáticos de recuento de eosinófilos en enfermedades inflamatorias se localizó en el estado de cromatina 33, el cual fue enriquecido por impactos de GWAS. En contraste, la región circundante de este SNP se asignó a otros estados de cromatina sin asociación significativa de GWAS. Esto puede arrojar luz sobre la posible importancia funcional de los SNP de la enfermedad en función de sus distintos estados de cromatina.

#### 4. Estados Reprimidos a Gran Escala (40-45):

Estos estados marcaron regiones reprimidas y heterocromáticas a gran escala, representando 64% del genoma. H3K27me3 y H3K9me3 fueron las dos marcas más frecuentemente detectadas en este grupo.

#### 5. Estados Repetitivos (46-51):

Estos estados mostraron enriquecimientos fuertes y distintos para elementos repetitivos específicos. Por ejemplo, el estado 46 tuvo una firma de secuencia fuerte de repeticiones de baja complejidad como (CA) n, (TG) n y (CATG) n. Los estados 48-51 mostraron frecuencias aparentemente altas para muchas modificaciones pero también enriquecimiento en lecturas de control de anticuerpos no específicos. El modelo también pudo capturar artefactos resultantes de la falta de cobertura para copias adicionales de elementos repetidos.

Dado que muchos de los estados de cromatina fueron descritos por múltiples marcas, se cuantificó la contribución de cada marca a un estado. Se probaron diferentes subconjuntos de marcas de cromatina para evaluar su potencial para distinguir entre estados de cromatina. En general, se encontró que subconjuntos crecientes de marcas convergen a un estado preciso de cromatina cuando las marcas se eligieron con avidez.

El poder predictivo de los estados de cromatina para el descubrimiento de elementos funcionales superó consistentemente a las predicciones basadas en marcas individuales. Tal modelo no supervisado usando combinación de marcas epigenómicas e información genómica espacial realizada así como muchos modelos supervisados en anotación genómica. Se demostró que este modelo HMM basado en estados de cromatina fue capaz de revelar promotores previamente no anotados y regiones transcritas que fueron apoyadas por evidencia experimental independiente. Cuando se analizaron las marcas de cromatina en todo el genoma, algunas de las propiedades observadas fueron estados enriquecidos satélite (47-51) enriquecidos en centrómero, el estado enriquecido con dedos de zinc (estado 28) enriquecido en el cromosoma 19 etc. Así, dicha anotación genómica basada en estados de cromatina puede ayudar a interpretar mejor datos biológicos y potencialmente descubrir nuevas clases de elementos funcionales en el genoma.

### Múltiples tipos de celdas

Todo el trabajo anterior se realizó en un solo tipo de célula (células T CD4+). Dado que los marcadores epigenómicos varían con el tiempo, según los tipos de células y las circunstancias ambientales, es importante considerar la dinámica de los estados de cromatina en diferentes tipos de células y condiciones experimentales. El proyecto ENCODE [3] en el Grupo Brad Bernstein Chromatin ha medido 9 marcas diferentes de cromatina en nueve líneas celulares humanas. En este caso, queremos aprender un solo conjunto de marcas de cromatina para todos los datos. Hay dos enfoques para este problema: la concatenación y el apilamiento. Para la concatenación, podríamos combinar todas las 9 líneas celulares como si fueran una sola línea celular. Al concatenar las diferentes líneas celulares, nos aseguramos de que se aprenda un conjunto común de definiciones de estado. Podemos hacer esto aquí porque las marcas perfiladas eran las mismas en cada experimento. Sin embargo, si perfilamos diferentes marcas para diferentes líneas celulares, necesitamos usar otro enfoque. Alternativamente, podemos alinear las 9 líneas celulares y tratar todas las marcas como un supervector. Esto nos permite aprender estados de actividad específicos de líneas celulares, por ejemplo, podría haber un estado para potenciadores específicos de ES (en ese estado habría marcas potenciadoras en ES, pero no marcas en otros tipos de células). Desafortunadamente, esto aumenta en gran medida la dimensión de los vectores emitidos por el HMM, lo que se traduce en un aumento en la complejidad del modelo necesaria para ajustarse adecuadamente a los datos.

Supongamos que teníamos múltiples tipos de células donde perfilamos diferentes marcas y queríamos concatenarlas. Un enfoque es aprender modelos independientes y luego combinarlos. Podríamos encontrar estados correspondientes al emparejar vectores de

emisión que son similares o al emparejar estados que aparecen en los mismos lugares del genoma. Un segundo enfoque es tratar las marcas faltantes como datos faltantes. El marco EM permite puntos de datos no especificados, por lo que siempre y cuando se observen relaciones por pares entre marcas en algún tipo de celda, podemos usar EM. Por último, podemos predecir las marcas de cromatina faltantes con base en las marcas observadas usando máxima verosimilitud como en el algoritmo de Viterbi. Este es un enfoque menos poderoso si el objetivo final es el aprendizaje del estado de cromatina porque solo estamos mirando el estado más probable en lugar de promediar sobre todas las posibilidades como en el segundo enfoque.

En el caso de 9 marcas en 9 líneas celulares humanas, se concatenaron las líneas celulares y se aprendió un modelo con 15 estados [8]. Cada tipo de célula se analizó para el enriquecimiento de clase. Se demostró que algunos estados de cromatina, como los que codifican promotores activos, fueron altamente estables en todos los tipos de células. Otros estados, como los que codifican potenciadores fuertes, estaban altamente enriquecidos de una manera específica de tipo celular, lo que sugiere su papel en la expresión génica específica de tejido. Finalmente, se demostró que había correlación significativa entre las marcas epigenéticas en los potenciadores y las marcas epigenéticas en los genes que regulan, aunque estas pueden estar a miles de pares de bases de distancia. Tal modelo de estado de cromatina ha demostrado ser útil para emparejar potenciadores con sus respectivos genes, un problema que en gran parte no ha sido resuelto en la biología moderna. Por lo tanto, los estados de cromatina proporcionan un medio para estudiar la naturaleza dinámica de la cromatina en muchos tipos de células. En particular, podemos ver la actividad de una región particular del genoma a partir de la anotación de la cromatina. También nos permite resumir información importante contenida en 2.4 mil millones de lecturas en tan solo 15 estados de cromatina.

Una publicación de 2015 Nature del Epigenome Roadmap Project ha demostrado que produjo una referencia incomparable para las firmas de epigenómica humana en más de cien tejidos diferentes [2]. En su análisis, hacen uso de varios de los conceptos que hemos discutido en profundidad en este capítulo, como un modelo ChromHMM de 15 estados o 18 estados para anotar el epigenoma. El entrenamiento sobre 111 conjuntos de datos permitió una mayor robustez a los modelos HMM discutidos anteriormente. El proyecto Roadmap exploró muchas direcciones interesantes en su artículo, y se recomienda encarecidamente a los lectores interesados que lean esta publicación. Conclusiones interesantes incluyen que los estados asociados a H3K4-me1 son las marcas de cromatina más específicas de tejido, y que los promotores bivalentes y los estados reprimidos también fueron las anotaciones más variables en diferentes tipos de tejido. Para los potenciadores, el proyecto Roadmap encontró que una cantidad significativa de SNP relacionados con la enfermedad están asociados con regiones potenciadoras anotadas. La exploración activa de esta conexión está en curso en el Grupo de Biología Computacional del MIT.

---

This page titled [19.5: Anotar el genoma usando firmas de cromatina](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [19.5: Annotating the Genome Using Chromatin Signatures](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.6: Direcciones actuales de investigación

Varios proyectos eorts de producción de datos a gran escala como ENCODE, MODENcode y Epigenome Roadmap están actualmente en curso y por lo tanto existen varias oportunidades para analizar computacionalmente estos nuevos datos. Los datos epigenómicos también se están utilizando para estudiar cómo el comportamiento puede alterar su genoma. Se están realizando estudios que analizan la dieta y el ejercicio y sus efectos sobre la susceptibilidad a enfermedades.

Otra área interesante de investigación es el análisis de los cambios epigenéticos en la enfermedad. La investigación actual en el Grupo de Biología Computacional del MIT está analizando el vínculo entre los estados de cromatina y la enfermedad de Alzheimer. A continuación se presenta una selección de artículos sobre la vinculación epigenética-enfermedad.

This page titled [19.6: Direcciones actuales de investigación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **19.6: Current Research Directions** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.7: Lectura adicional, herramientas y técnicas

Hay varios trabajos interesantes que están analizando los estados de la cromatina y la epigenética en general. A continuación se enumeran varias URL para comenzar su exploración:

1. [www.nature.com/nmeth/journal/... meth.1673.html](http://www.nature.com/nmeth/journal/... meth.1673.html)
2. [www.nature.com/nature/journal/... ture09906.html](http://www.nature.com/nature/journal/... ture09906.html)
3. [www.nature.com/nbt/journal/v2... /nbt.1662.html](http://www.nature.com/nbt/journal/v2... /nbt.1662.html)
4. [http://www.nytimes.com/2012/09/09/opinion/epigenetics-and-disease.html?\\_r=1](http://www.nytimes.com/2012/09/09/opinion/epigenetics-and-disease.html?_r=1)
5. [www.nature.com/doifinder/10.1038/nature14248](http://www.nature.com/doifinder/10.1038/nature14248)

Se trata de algunas publicaciones seleccionadas que tratan sobre la epigenética y la enfermedad.

1. [www.nature.com/nature/journal/... ture02625.html](http://www.nature.com/nature/journal/... ture02625.html)
2. <http://www.sciencedirect.com/science/article/pii/S11124712003725>
3. [www.nature.com/nbt/journal/v2... f/nbt.1685.pdf](http://www.nature.com/nbt/journal/v2... f/nbt.1685.pdf)

### Herramientas y Técnicas

ChromHMM es el HMM descrito en el texto. Está disponible para su descarga gratuita con instrucciones y ejemplos en: <http://compbio.mit.edu/ChromHMM/>.

Segway es otro método para analizar múltiples pistas de datos de genómica funcional. Utiliza una red bayesiana dinámica (los HMM son un tipo particular de red bayesiana dinámica) que le permite analizar todo el genoma a una resolución de 1 pb. El inconveniente es que es mucho más lento que ChromHMM. Está disponible de forma gratuita para su descarga aquí: <http://noble.gs.washington.edu/proj/segway/>.

---

This page titled [19.7: Lectura adicional, herramientas y técnicas](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

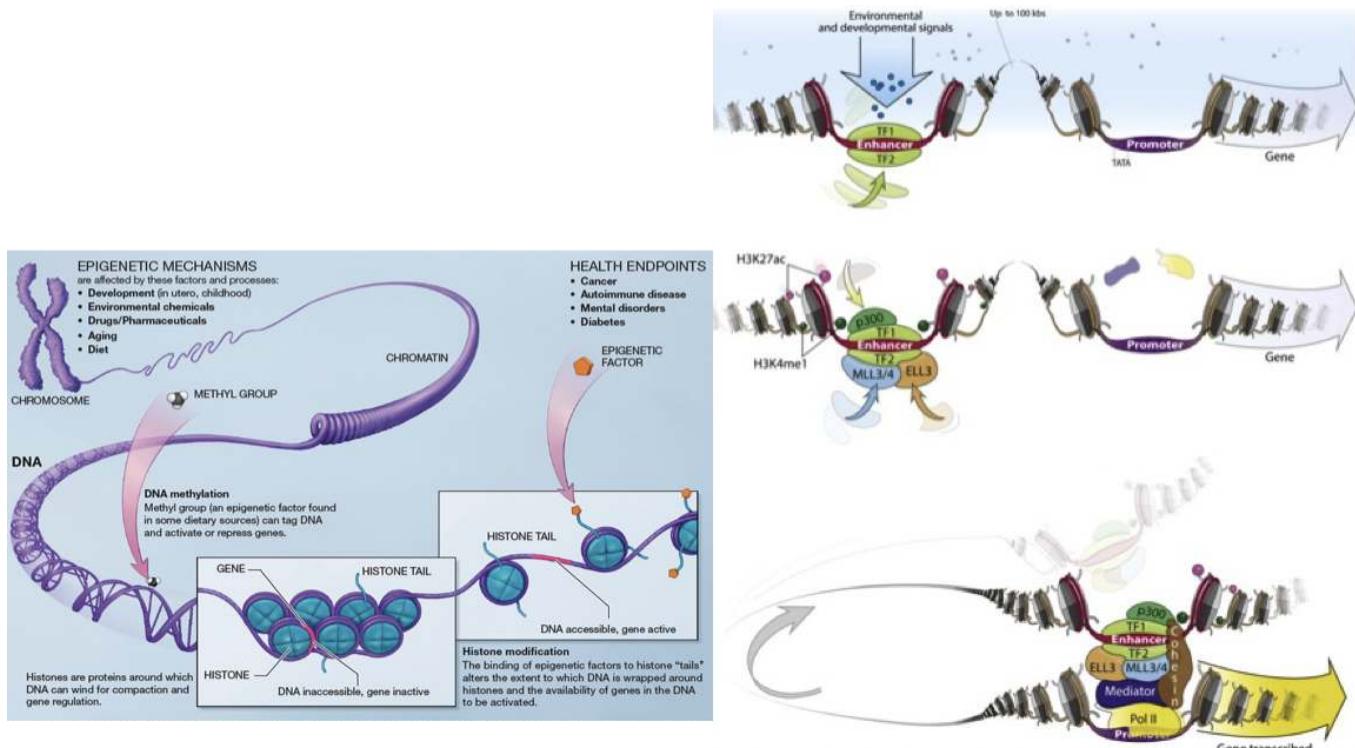
- [19.7: Further Reading, Tools and Techniques](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 19.8: ¿Qué hemos aprendido? , Bibliografía

En esta conferencia aprendimos cómo las marcas de cromatina pueden ser utilizadas para inferir estados biológicamente relevantes. El análisis en [7] presenta un método sofisticado para aplicar técnicas previamente aprendidas como los HMM a un problema complejo. La conferencia también presentó la poderosa transformación Burrows-Wheeler que ha permitido el mapeo de lectura eficiente.

### Bibliografía

- [1] Langmead B, Trapnell C, Pop M y Salzberg S. Ultrafast, alineamiento memory-eficiente de secuencias cortas de ADN con el genoma humano. *Biología del Genoma*, 10 (3), 2009.
- [2] Roadmap Epigenomics Consortium, Kundaje A, Meuleman W, et al. Análisis integrador de 111 epigenomas humanos de referencia. *Naturaleza*, 518 (7539) :317—330, 2015.
- [3] El Consorcio del Proyecto ENCODE. Una enciclopedia integrada de elementos de ADN en el genoma humano. *Naturaleza*, 489 (7414) :57—74, 2012.
- [4] Escuché E y Martienssen RA. Herencia epigenética transgeneracional: Mitos y mecanismos. *Cell*, 157 (1) :95—109, 2014.
- [5] Mardis ER. Chip-seq: bienvenido a la nueva frontera. *Nature Methods*, 4 (8) :614—614, 2007.
- [6] Herz H-M, Hu D y Shilatifard A. Mal funcionamiento del potenciador en cáncer. *Molecular Cell*, 53 (6) :859—866, 2014.
- [7] Ernst J y Kellis M. Descubrimiento y caracterización de estados de cromatina para la anotación sistemática del genoma humano. *Nature Biotechnology*, 28:817 —825, 2010.
- [8] Ernst J, Kheradpour P, Mikkelsen TS, et al. Mapeo y análisis de la dinámica del estado de la cromatina en nueve tipos celulares humanos. *Naturaleza*, 473 (7345) :43—49, 2011.
- [9] Mousavi K, Zare H, dell'Orso S, Grontved L, et al. ERnas promueven la transcripción estableciendo accesibilidad a la cromatina en loci genómicos definidos. *Molecular Cell*, 51 (5) :606—17, 2013.
- [10] Qunhua Li, James B. Brown, Haiyan Huang y Peter J. Bickel. Medición de la reproducibilidad de experimentos de alto rendimiento. *Los anales de la estadística aplicada*, 5 (3) :1752—1779, 2011.
- [11] Li Y y Tollefsbol TO. Detección de metilación del ADN: Análisis de secuenciación genómica con bisulfito. *Métodos Biología Molecular*, 791:11 —21, 2011.



Cortesía de Institutos Nacionales de Salud. Imagen en el dominio público (izquierda).

Cortesía de Elsevier, Inc. Usado con permiso. Fuente: Herz, Hans-Martin, Deqing Hu, et al. "Mal Mal Mejorador en Cáncer". *Molecular Cell* 53, núm. 6 (2014): 859-66 (derecha).

Figura 19.1: A. Existe una amplia diversidad de modificaciones en el epigenoma. Algunas regiones del ADN se enrollan de forma compacta alrededor de las histonas, haciendo que el ADN sea inaccesible y los genes inactivos. Otras regiones tienen ADN más accesible y, por lo tanto, genes activos. Los factores epigenéticos pueden unirse a las colas de estas histonas para modificar estas propiedades. B. Las modificaciones de histonas proporcionan información sobre qué tipos de proteínas están unidas al ADN y cuál es la función de la región. En este ejemplo, las modificaciones de histonas permiten que una región potenciadora (potencialmente a más de 100 bases de kilo de distancia) interactúe con la región promotora. [6]

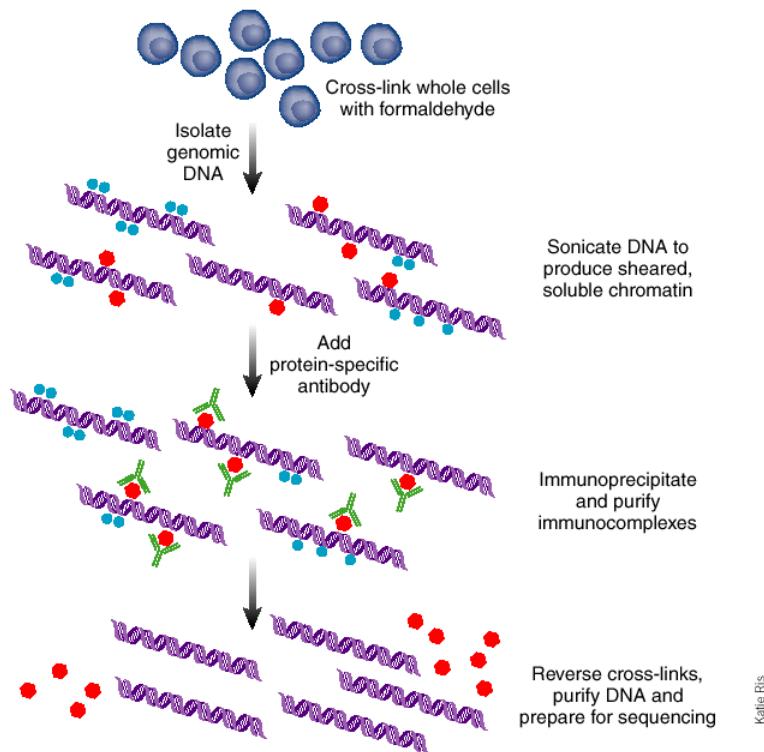


Figura 19.2: El método de inmunoprecipitación de cromatina [5]. Los pasos en esta figura corresponden a los seis pasos del procedimiento.

Input				All Rotations				Sorted List of Rotations				Output Last Column			
Add 1	Sort 1	Add 2	Sort 2	Add 3	Sort 3	Add 4	Sort 4	Add 5	Sort 5	Add 6	Sort 6	Add 7	Sort 7	Add 8	Sort 8
B	A	BA	AN	BAN	ANA	BANA	ANAN	BANAN	ANANA	BANANA	ANANAB	BANANAB	ANANABA	BANANABA	ANANAB^A
N	A	NA	AN	NAN	ANA	NANA	ANA@	NANA@	ANA@^	NANA@^	ANA@^B	NANA@^B	ANA@^BA	NANA@^BA	NANA@^BAN
N	A	NA	A@	NA@	A@^	NA@^	A@^B	NA@^B	A@^BA	NA@^BA	A@^BAN	NA@^BAN	A@^BANA	NA@^BANA	NA@^BANAN
^	B	^B	BA	^BA	BAN	^BAN	BANA	^BANA	BANAN	^BANAN	BANANA	^BANANA	BANANA	^BANANA	BANANA@^B
A	N	AN	NA	ANA	NAN	ANAN	NANA	ANANA	NANA@	ANANA@	NANA@^	ANANA@^	NANA@^B	ANANA@^B	ANANA@^BAN
A	N	AN	NA	ANA	NA@	ANA@	NA@^	ANA@^	NA@^B	ANA@^B	NA@^BA	ANA@^BA	NA@^BAN	ANA@^BAN	ANA@^BANAN
@	^	^@^	^B	^@B	^BA	^@BA	^BAN	^@BAN	^BANA	^@BANA	^BANAN	^@BANAN	^BANANA	^@BANANA	^BANANA@^B
A	@	@	A@	@^	@^B	@^B	A@^B	@^B	A@^BA	@^BA	A@^BAN	@^BANA	A@^BANA	@^BANAN	A@^BANAN

Figura 19.3: (Arriba) En la transformación hacia adelante Madrows-Wheeler se generan y clasifican las rotaciones. La última columna de la lista ordenada (en negrilla) consiste en la cadena transformada. (Abajo) En la transformación inversa de Burrows-Wheeler se ordena la cadena transformada, y se generan dos columnas: una que consiste en la cadena original y la otra compuesta por la ordenada. Éstas forman de manera eficaz dos columnas a partir de las rotaciones en la transformación hacia adelante. Este proceso se repite hasta que se generan las rotaciones completas.

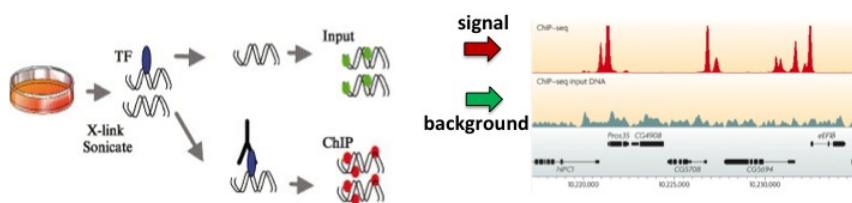


Figura 19.4: Para usar ADN de entrada como control, se puede ejecutar el experimento ChIP como normal mientras simultáneamente se ejecuta el mismo experimento (con el mismo ADN) sin un anticuerpo. Esto genera una señal de fondo para la cual podemos corregir.

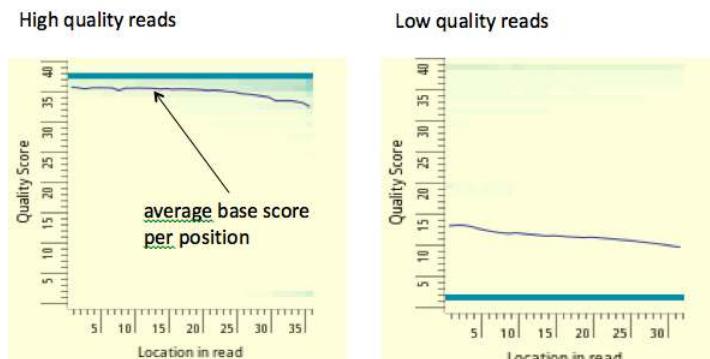
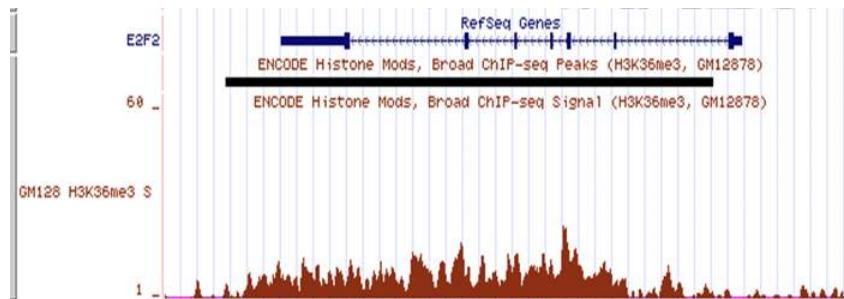


Figura 19.5: En la figura anterior cada columna se encuentra un histograma codificado por colores que codifica la fracción de todas las lecturas mapeadas que tienen puntuación base Q (eje y) en cada posición (eje x). Un puntaje promedio bajo por base implica una mayor probabilidad de errores de mapeo. Normalmente rechazamos lecturas cuya puntuación promedio Q es menor a 10.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 19.6: Una pista de señal de muestra. Aquí, la señal roja se deriva del número de lecturas que mapearon al genoma en cada posición para un experimento ChIP-seq con la diana H3K36me3. La señal da un nivel de enriquecimiento de la marca

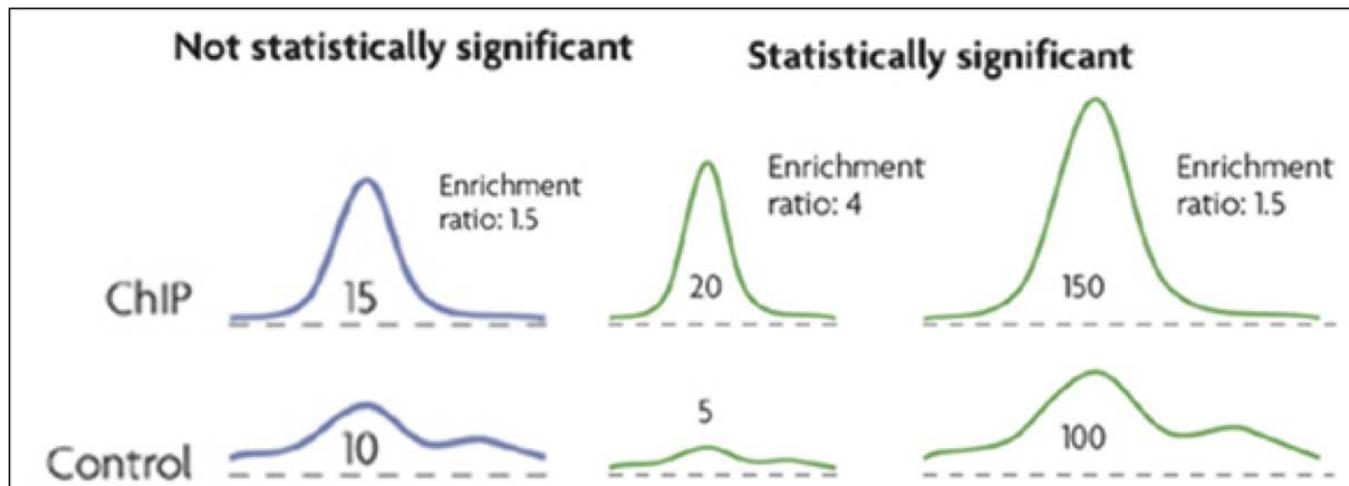
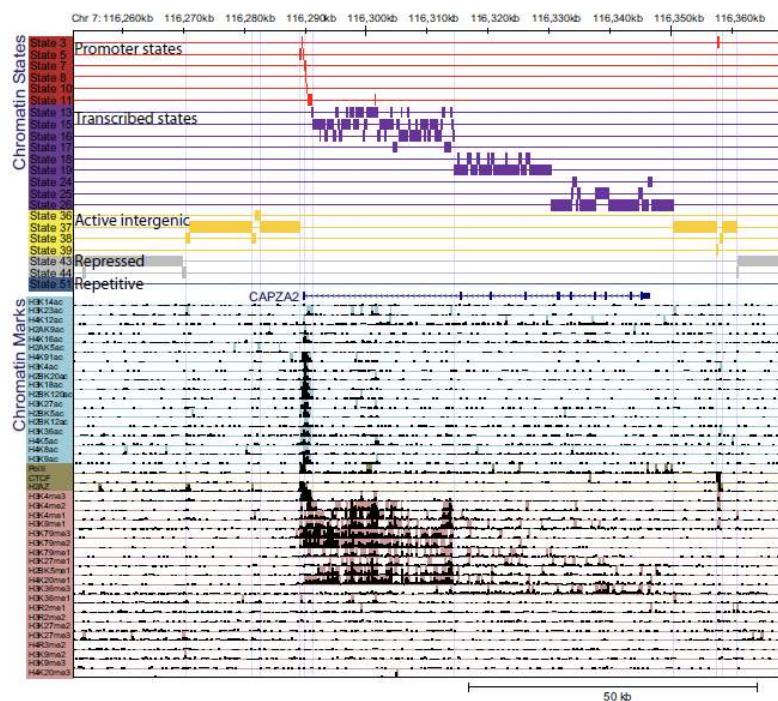


Figura 19.7: Pistas de señal de muestra tanto para el experimento verdadero como para el fondo (control). Se considera que las regiones tienen enriquecimiento estadísticamente significativo cuando los valores verdaderos de la señal del experimento están muy por encima de los valores de la señal de fondo.



Cortesía de Macmillan Publishers Limited. Usado con permiso. Fuente: Ernst, Jason y Manolis Kellis. "Descubrimiento y Caracterización de los Estados Cromatinos para la Anotación Sistemática del Genoma Humano". *Nature Biotechnology* 28, núm. 8 (2010): 817-25.

Figura 19.8: Ejemplo de los datos y la anotación del modelo HMM. La sección inferior muestra el número bruto de lecturas mapeadas al genoma. La sección superior muestra la anotación del modelo HMM.

Figura 19.9: Probabilidades de emisión para el modelo final con 51 estados. La celda correspondiente a la marca  $i$  y al estado  $k$  representa la probabilidad de que la marca  $i$  se observe en el estado  $k$ .

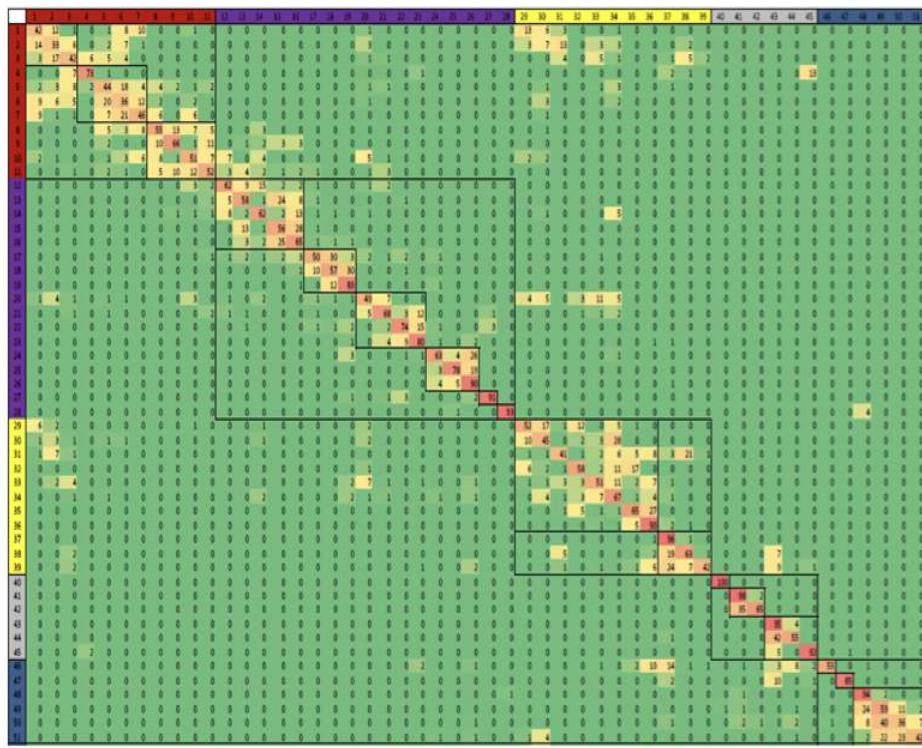


Figura 19.10: Probabilidades de transición para el modelo final con 51 estados. La probabilidad de transición aumenta de verde a rojo. Las relaciones espaciales entre estados de cromatina vecinos y distintos subgrupos de estados se revelan agrupando la matriz de transición. Notablemente, la matriz es escasa, por lo que indica que la mayoría no son posibles.

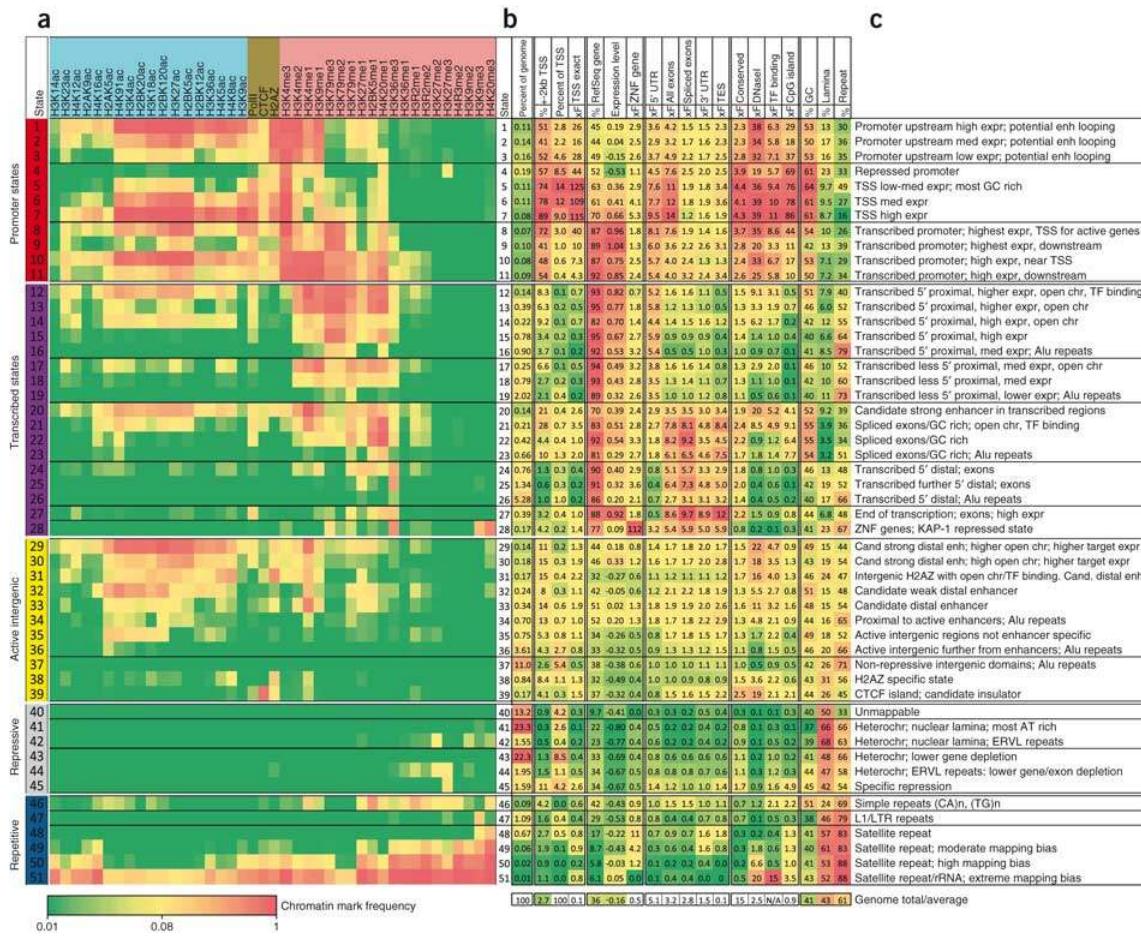


Figura 19.11: Definición del estado de cromatina e interpretación funcional. [7] a. Combinaciones de marcas de cromatina asociadas a cada estado. Cada fila muestra la combinación específica de marcas asociadas a cada estado de cromatina y las frecuencias entre 0 y 1 con las que ocurren en escala de colores. Estos corresponden a los parámetros de probabilidad del HMM aprendidos a través del genoma durante el entrenamiento del modelo. b. Enriquecimientos genómicos y funcionales de estados de cromatina, incluido el enriquecimiento en veces en la parte dierente del genoma (por ejemplo, regiones transcritas, TSS, RefSeq 5 o 3end del gen, etc.), además plegar el enriquecimiento para elementos conservados evolutivamente, sitios hipersensibles a la ADNasa I, islas CpG, etc. Todos los enriquecimientos se basan en las asignaciones de probabilidad posteriores. c. Breve descripción de la función e interpretación del estado biológico (chr, cromatina; enh, potenciador).

This page titled 19.8: ¿Qué hemos aprendido? , Bibliografía is shared under a CC BY-NC-SA 4.0 license and was authored, remixed, and/or curated by Manolis Kellis et al. (MIT OpenCourseWare) via source content that was edited to the style and standards of the LibreTexts platform.

- **19.8: What Have We Learned?, Bibliography** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source:

<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>

## CHAPTER OVERVIEW

### 20: Redes I- Inferencia, Estructura, Métodos Espectrales

20.1: Introducción

20.2: Medidas de Centralidad de Red

20.3: Revisión de álgebra lineal

20.4: Análisis de componentes principales dispersos

20.5: Comunidades y Módulos de Red

20.6: Núcleo de Difusión en Red

20.7: Redes neuronales

20.8: Temas abiertos y desafíos

20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía

20.10: ¿Qué hemos aprendido?

Bibliografía

---

This page titled [20: Redes I- Inferencia, Estructura, Métodos Espectrales](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 20.1: Introducción

La biología molecular y celular describen un sistema enormemente diverso de componentes que interactúan que es capaz de producir fenómenos intrincados y complejos. Las interacciones dentro del proteoma describen el metabolismo celular, las cascadas de señalización y la respuesta al ambiente. Las redes son una herramienta valiosa para ayudar a representar, comprender y analizar las complejas interacciones entre los componentes biológicos. Los sistemas vivos pueden verse como una composición de múltiples capas que codifican cada una información sobre el sistema. Algunas capas importantes son:

1. Genoma: Incluye ADN codificante y no codificante. Los genes definidos por ADN codificante se utilizan para construir ARN, y los elementos reguladores de la CIS regulan la expresión de estos genes.
2. Epigenoma: Definido por la configuración de la cromatina. La estructura de la cromatina se basa en la forma en que las histonas organizan el ADN. El ADN se divide en regiones libres de nucleosomas y nucleosomas, formando su forma final e influyendo en la expresión génica.<sup>1</sup>
3. Los ARN del transcriptoma (por ejemplo, ARNm, miARN, ncRNA, piRNA) se transcriben a partir del ADN. Tienen funciones reguladoras y fabrican proteínas.
4. Proteoma Compuesto por proteínas. Esto incluye factores de transcripción, proteínas de señalización y enzimas metabólicas.

Cada capa consiste en una red de interacciones. Por ejemplo, los ARNm y los miARN interactúan para regular la producción de proteínas. Las capas también pueden interactuar entre sí, formando una red entre redes. Por ejemplo, un ARN largo no codificante llamado Xist produce cambios epigenómicos en el cromosoma X para lograr una compensación de dosis a través de la inactivación de X.

### Presentamos Redes Biológicas

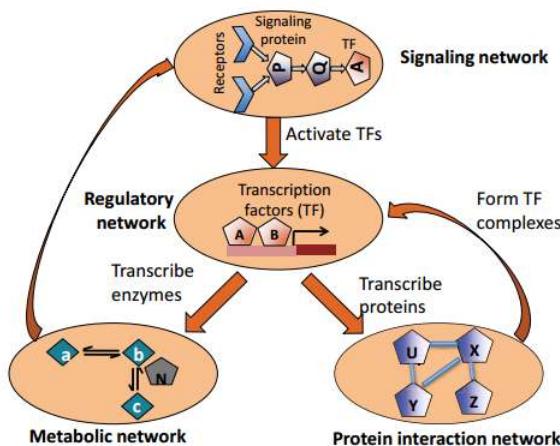
Cinco tipos ejemplares de redes biológicas: **Red**

**Reguladora** — conjunto de interacciones regulatorias en un organismo.

- Los nodos representan reguladores (por ejemplo, factores de transcripción) y dianas asociadas.
- Los bordes representan la interacción regulatoria, dirigida desde el factor regulatorio hasta su objetivo. Se firman según el efecto positivo o negativo y se ponderan según la fuerza de la reacción.

**Red Metabólica** — conecta los procesos metabólicos. Hay cierta flexibilidad en la representación, pero un ejemplo es una gráfica que muestra productos metabólicos compartidos entre enzimas.

- Los nodos representan enzimas.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 20.1: Interacciones entre redes biológicas.

- Los bordes representan reacciones reguladoras, y se ponderan de acuerdo con la fuerza de la reacción. Los bordes no están dirigidos.

### Red de Señalización — representa rutas de señales biológicas.

- Los nodos representan proteínas llamadas receptores de señalización.
- Los bordes representan señales biológicas transmitidas y recibidas, dirigidas del transmisor al receptor. Los bordes están dirigidos y no ponderados.

### Red de proteínas: muestra interacciones físicas entre proteínas.

- Los nodos representan proteínas individuales.
- Los bordes representan interacciones físicas entre pares de proteínas. Estos bordes son no dirigidos y no ponderados.

**Red de coexpresión:** describe las funciones de coexpresión entre genes. Muy general; representa redes de interacción funcionales más que físicas, a diferencia de los otros tipos de redes. Potente herramienta en el análisis computacional de datos biológicos.

- Los nodos representan genes individuales.
- Los bordes representan relaciones de coexpresión. Estos bordes son no dirigidos y no ponderados.

Hoy, nos centraremos exclusivamente en las redes regulatorias. Las redes regulatorias controlan la expresión génica específica del contexto y, por lo tanto, tienen un gran control sobre el desarrollo. Vale la pena estudiarlos porque son propensos al mal funcionamiento y están asociados con enfermedades.

## Interacciones entre redes biológicas

Las redes biológicas individuales (es decir, capas) pueden considerarse nodos en una red más grande que representa todo el sistema biológico. Podemos, por ejemplo, tener una red de señalización detectando el entorno que gobierna la expresión de los factores de transcripción. En este ejemplo, la red mostraría que los TFs gobiernan la expresión de las proteínas, las proteínas pueden desempeñar papeles como enzimas en las vías metabólicas, y así sucesivamente.

Las rutas generales de intercambio de información entre estas redes se muestran en la figura 21.1a.

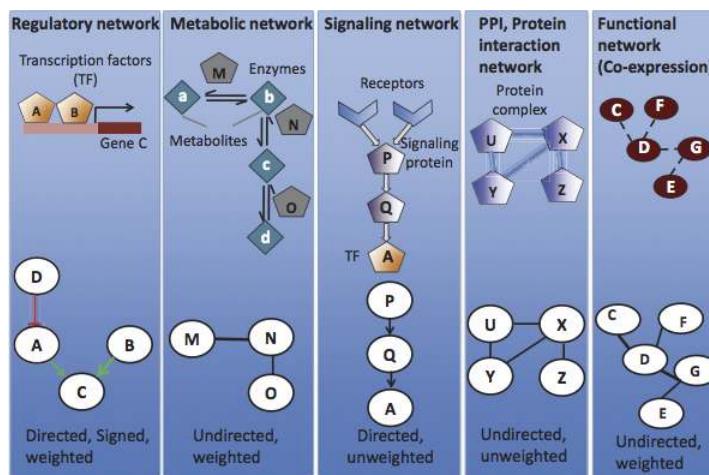


Figura 20.2: Representación de diferentes tipos de redes.

fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte [ocw.mit.edu/help/faq-fiar-use/](http://ocw.mit.edu/help/faq-fiar-use/).

## Representación de la red

En la figura 20.2 se muestran varias de estas redes y sus visualizaciones como gráficas. Sin embargo, ¿cómo decidimos que estas redes particulares representan los modelos biológicos subyacentes? Dado un gran conjunto de datos biológicos, ¿cómo podemos entender las dependencias entre objetos biológicos y cuál es la mejor manera de modelar estas dependencias? A continuación, presentamos varios enfoques para la representación de la red. En la práctica, ningún modelo es perfecto. La elección del modelo debe equilibrar el conocimiento biológico y la computabilidad para un análisis razonablemente eficiente.

Las redes se describen típicamente como gráficas. Las gráficas están compuestas por 1. nodos, que representan objetos; y 2. bordes, que representan conexiones o interacciones entre nodos. Hay tres formas principales de pensar sobre las redes biológicas como gráficas.

**Redes probabilísticas** — también conocidas como modelos gráficos. Modelan una distribución de probabilidad entre nodos.

- Modelado de la distribución conjunta de probabilidad de variables mediante gráficas.
- Algunos ejemplos son Bayesian Networks (dirigido), Markov Random Fields (Undirected). Más sobre las redes bayesianas en los capítulos posteriores.

**Redes Físicas** — En este esquema solemos pensar en los nodos como interactuando físicamente entre sí y los bordes capturan esa interacción.

- Los bordes representan la interacción física entre los nodos. • Ejemplo: redes regulatorias físicas.

**Relevancia Red** — Modele la correlación entre nodos. • Los pesos de borde representan similitudes de nodos.

- Ejemplo: redes regulatorias funcionales.

### Redes como Gráficas

Los informáticos consideran subtipos de gráficos, cada uno con diferentes propiedades para sus bordes y nodos.

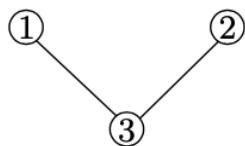


Figura 20.3: Una red simple en 3 nodos. La matriz de adyacencia de esta gráfica se da en la ecuación (21.1).

- **Gráfica ponderada:** Los bordes tienen un peso asociado. Los pesos son generalmente positivos. Cuando todos los pesos son 1, entonces lo llamamos una **gráfica no ponderada**.
- **Gráficas dirigidas:** Los bordes poseen direccionalidad. Por ejemplo  $A! B$  no es lo mismo que  $A B$ . Cuando los bordes no tienen dirección, lo llamamos una **gráfica no dirigida**.
- **Multígrafos (pseudografías):** Cuando permitimos que más de un borde vaya entre dos nodos (más de dos si está dirigido) entonces lo llamamos multígrafo. Esto puede ser útil para modelar múltiples interacciones entre dos nodos cada uno con pesos diferentes por ejemplo.
- **Gráfica simple:** Todos los bordes no están dirigidos y no ponderados. Se prohíben múltiples aristas entre nodos y bordes propios.

### Representación Matriz de Gráficas

**Matriz de adyacencia** Una forma de representar una red es usar la llamada matriz de adyacencia. La matriz de adyacencia de una red con  $n$  nodos es una  $n \times n$  matriz  $A$  donde  $A_{ij}$  es igual a uno si hay un borde entre los nodos  $i$  y  $j$ , y 0 en caso contrario. Por ejemplo, la matriz de adyacencia del gráfico representado en la figura 21.6b viene dada por:

```

\begin{array}{ll}
A = & \left[ \begin{array}{lll}
0 & 0 & 1 \\
0 & 0 & 1 \\
1 & 1 & 0
\end{array} \right]
\end{array}
  
```

Si la red está ponderada (es decir, si cada uno de los bordes de la red tiene un peso asociado), la definición de la matriz de adyacencia se modifica para que  $A_{ij}$  mantenga el peso del borde entre  $i$  y  $j$  si el borde existe, y cero en caso contrario.

Otra conveniencia que viene con la representación de la matriz de adyacencia es que cuando tenemos una matriz binaria (gráfica no ponderada) entonces la suma de la fila  $i$  nos da el grado de nodo  $i$ . En una gráfica no dirigida, el **grado** de un nodo es el número de aristas que tiene. Ya que cada entrada en la fila nos dice si el nodo  $i$  está conectado a otro nodo, al sumar todos estos valores sabemos a cuántos nodos está conectado el nodo  $i$ , así obtenemos el grado.

<sup>1</sup> Más en la conferencia de epigenética.

---

This page titled [20.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [20.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

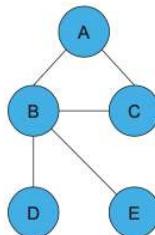
## 20.2: Medidas de Centralidad de Red

Se discutió en el capítulo anterior cómo podemos tomar una red biológica y modelarla matemáticamente. Ahora como visualizamos estas gráficas y tratamos de entenderlas necesitamos alguna medida por la importancia de un nodo/borde a las características estructurales del sistema. Hay muchas maneras de medir la importancia (lo que denominamos centralidad) de un nodo. En este capítulo exploraremos estas ideas e investigaremos su significación.

### Centralidad de Titulación

La primera idea sobre la centralidad es medir la importancia por el grado de un nodo. Esta es probablemente una de las medidas de centralidad más intuitivas ya que es muy fácil de visualizar y razonar. Cuantos más bordes tengas conectados contigo, más importante para la red eres.

Exploraremos un ejemplo sencillo y veamos cómo se trata de encontrar estas centralidades. Tenemos la siguiente gráfica



Y nuestro objetivo es encontrar la centralidad de grado de cada nodo en la gráfica. Para proceder, primero escribimos la matriz de adyacencia para esta gráfica. El orden de los bordes es A, B, C, D, E

```
\[A=\left[\begin{array}{lllll} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]
```

Anteriormente discutimos cómo encontrar el grado para un nodo dado una matriz de adyacencia. Sumamos a lo largo de cada fila de la matriz de adyacencia.

```
\[D=\left[\begin{array}{l} 1 \\ 4 \\ 3 \\ 1 \\ 1 \end{array}\right]
```

Ahora D es un vector con el grado de cada nodo. Este vector nos da unas medidas de centralidad relativa para los nodos de esta red. Podemos observar que el nodo B tiene el grado más alto de centralidad.

Aunque esta métrica nos da mucha perspicacia, tiene sus limitaciones. Imagine una situación en la que haya un nodo que conecte dos partes de la red entre sí. El nodo tendrá un grado de 2, pero es mucho más importante que eso.

### Centralidad entre

La centralidad entre medias nos da otra forma de pensar sobre la importancia en una red. Mide el número de caminos más cortos en la gráfica que pasan por el nodo dividido por el número total de caminos más cortos. En otras palabras, esta métrica calcula todas

las rutas más cortas entre cada par de nodos y ve cuál es el porcentaje de que pasa por el nodo k, ese porcentaje nos da la centralidad para el nodo k.

- Nodos con centralidad de alto entremezclamiento controlan el flujo de información en una red.
- El entreborde se define de manera similar.

## Cercanía Centralidad

Para definir adecuadamente la cercanía necesitamos definir el término **distancia**. La distancia entre dos nodos es la ruta más corta entre ellos. La distancia de un nodo es la suma de distancias entre ese nodo y todos los demás nodos. Y la cercanía de un nodo es la inversa de su distancia. En otras palabras, es el inverso normalizado de la suma de distancias topológicas en la gráfica.

El nodo más central es el nodo que propaga la información más rápido a través de la red.

La descripción de la centralidad de la cercanía la hace similar a la centralidad del grado. ¿El más alto grado de centralidad es siempre la más alta centralidad de cercanía? No. Piense en el ejemplo donde un nodo conecta dos componentes, ese nodo tiene una centralidad de bajo grado pero una centralidad de alta cercanía.

## Centralidad Eigenvector

La centralidad de los vectores propios extiende el concepto de grado. Lo mejor para pensarla es el promedio de las centralidades de sus vecinos de la red. El vector de centralidades puede escribirse como:

$$\mathbf{x} = \frac{1}{\lambda} \mathbf{Ax}$$

donde A es la matriz de adyacencia. La solución a la ecuación anterior va a ser el **vector propio** correspondiente al **componente principal** (el valor propio más grande).

La siguiente sección incluye una revisión de conceptos de álgebra lineal incluyendo valores propios y vectores propios.

---

This page titled [20.2: Medidas de Centralidad de Red](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [20.2: Network Centrality Measures](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.3: Revisión de álgebra lineal

Nuestro objetivo de esta sección es recordarte algunos conceptos que aprendiste en tu clase de álgebra lineal. Esto no pretende ser un recorrido detallado. Si quieres aprender más sobre alguno de los siguientes conceptos, te recomiendo recoger un libro de álgebra lineal y leer de esa sección. Pero esto servirá como un buen recordatorio y señalar conceptos que son importantes para nosotros en este capítulo.

### vectores propios

Dada una matriz cuadrada  $A, (m \times m)$ , el autovector  $v$  es la solución a la siguiente ecuación.

$$Av = \lambda v$$

En otras palabras, si multiplicamos la matriz por ese vector, solo cambiamos nuestra posición paralela al vector (recuperamos una versión escalada del vector  $v$ ).

Y  $\lambda$  (cuánto se escala el vector  $v$ ) se llama **el valor propio**.

Entonces, ¿cuántos valores propios hay como máximo? Demos los primeros pasos para resolver esta ecuación.

$$Av = \lambda v \Rightarrow (A - \lambda I)v = 0$$

que tiene soluciones distintas de cero cuando  $|A - \lambda I| = 0$ . Esa es una ecuación de orden  $m$ -ésimo en  $\lambda$  que puede tener como máximo  $m$  soluciones distintas. Recuerda que esas soluciones pueden ser complejas, aunque  $A$  sea real.

### Descomposición vectorial

Dado que los vectores propios forman el conjunto de todas las bases, representan completamente el espacio de columna. Dado eso, podemos descomponer cualquier vector arbitrario  $x$  en una combinación de vectores propios.

$$x = \sum_i c_i v_i$$

Así, cuando multiplicamos un vector con una matriz  $A$ , podemos reescribirlo en términos de los vectores propios.

```
\begin{array}{c}
A x = A \cdot (c_1 v_1 + c_2 v_2 + \dots + c_m v_m) \\
A x = c_1 A v_1 + c_2 A v_2 + \dots + c_m A v_m \\
A x = c_1 \lambda_1 v_1 + c_2 \lambda_2 v_2 + \dots + c_m \lambda_m v_m
\end{array}
```

Entonces la acción de  $A$  en  $x$  está determinado por los valores propios de  $\lambda$  y los vectores propios. Y podemos observar que los valores propios pequeños tienen un efecto grande en la multiplicación.

#### ¿Sabías?

- Para matrices simétricas, los vectores propios para valores propios distintos son ortogonales.
- Todos los valores propios de una matriz simétrica real son reales.
- Todos los valores propios de una matriz semidefinita positiva no son negativos.

### Descomposición Diagonal

También conocido como Eigen Descomposición. Sea  $S$  una  $m \times m$  matriz cuadrada con  $m$  vectores propios linealmente independientes (una matriz no defectuosa).

Luego, existe una descomposición (teorema de digitalización matricial)

$$S = U \Lambda U^{-1}$$

Donde las columnas de  $U$  son los vectores propios de  $S$ .  $\Lambda$  es una matriz diagonal con valores propios en su diagonal.

## Descomposición del Valor Singular

A menudo, la descomposición de valores singulares (SVD) se utiliza para el caso más general de factorizar una matriz  $m \times n$  no cuadrada:

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T \quad (20.3.1)$$

donde  $\mathbf{U}$  es una  $m \times m$  matriz que representa vectores propios ortogonales de  $\mathbf{A}\mathbf{A}^T$ ,  $\mathbf{V}$  es una  $n \times n$  matriz que representa vectores propios ortogonales de  $\mathbf{A}^T\mathbf{A}$  y  $\Sigma$  es una  $m \times n$  matriz que representa raíces cuadradas de los valores propios de  $\mathbf{A}^T\mathbf{A}$  (llamados valores singulares de  $\mathbf{A}$ ):

$$\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r), \sigma_i = \sqrt{\lambda_i} \quad (20.3.2)$$

El SVD de cualquier matriz dada se puede calcular con un solo comando en Matlab y no cubriremos los detalles técnicos de computarlo. Tenga en cuenta que la matriz “diagonal” resultante  $\Sigma$  puede no ser de rango completo, es decir, puede tener diagonales cero, y el número máximo de valores singulares distintos de cero es  $\min(m, n)$ .

Por ejemplo, vamos

```
\[A=\left[\begin{array}{cc}
```

```
1 & -1\\
```

```
1 & 0\\
```

```
1 & 0\\
```

```
\end{array}\right]\nonumber]
```

así  $m=3$ ,  $n=2$ . Su SVD es

```
\left[\begin{array}{ccc}
```

$$0 & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\$$

$$\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & \frac{1}{\sqrt{3}} \\$$

$$\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{3}} \\\end{array}\right] \text{izquierda} \left[\begin{array}{cc}
$$1 & 0 \\$$

$$0 & \sqrt{3} \\$$

$$0 & 0 \\$$

```
\end{array}\right] \text{derecha} \left[\begin{array}{cc}
```

$$\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\$$

$$\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\$$

$$0 & 0 \\$$

```
\end{array}\right]\nonumber]
```$$

Típicamente, los valores singulares están dispuestos en orden decreciente.

La SVD es ampliamente utilizada en técnicas estadísticas, de análisis numérico y de procesamiento de imágenes. Una aplicación típica de SVD es la aproximación óptima de rango bajo de una matriz. Por ejemplo, si tenemos una gran matriz de datos, por ejemplo 1000 por 500, y nos gustaría aproximarla con una matriz de rango inferior sin mucha pérdida de información, formulada como el siguiente problema de optimización:

Encuentra  $\mathbf{A}_k$  de rango  $k$  tal que  $\mathbf{A}_k = \min_{\mathbf{X}: \text{rank}(\mathbf{X})=k} \|\mathbf{A} - \mathbf{X}\|_F$

donde el subíndice F denota la norma Frobenius  $\|\mathbf{A}\|_F = \sqrt{\sum_i \sum_j |a_{ij}|^2}$ . Por lo general,  $k$  es mucho menor que  $r$ . La solución a este problema es la SVD de  $\mathbf{X} \mathbf{U} \Sigma \mathbf{V}^T$ , con los valores singulares de  $r-k$  más pequeños en  $\Sigma$  conjunto a cero:

$$\mathbf{A}_k = \mathbf{U} \text{diag}(\sigma_1, \dots, \sigma_k, 0) \mathbf{V}^T$$

Tal aproximación se puede demostrar que tiene un error de  $\|\mathbf{A} - \mathbf{A}_k\|_F = \sigma_{k+1}$ . Esto también se conoce como el teorema de Eckart-Young.

Una aplicación común de SVD al análisis de red es usar la distribución de valores singulares de la matriz de adyacencia para evaluar si nuestra red parece una matriz aleatoria. Debido a que la distribución de los valores singulares (ley de semicírculo

Wigner) y la del valor propio más grande de una matriz (distribución Tracy-Widom) se han derivado teóricamente, es posible derivar la distribución de valores propios (valores singulares en SVD) de una red observada (matriz), y calcular un p-valor para cada uno de los valores propios. Entonces solo necesitamos mirar los valores propios significativos (valores singulares) y sus correspondientes vectores propios (vectores singulares) para examinar estructuras significativas en la red. La siguiente figura muestra la distribución de valores singulares de un conjunto unitario gaussiano aleatorio (GUE, ver este enlace de Wikipedia para definición y propiedades [en.wikipedia.org/wiki/Random\\_matrix](https://en.wikipedia.org/wiki/Random_matrix)) matriz, que forman un semicírculo de acuerdo con la ley de semicírculo de Wigner (Figura 20.4).

Un ejemplo del uso de SVD para inferir patrones estructurales en una matriz o red se muestra en la Figura 20.5. El panel superior izquierdo muestra una estructura (roja) añadida a una matriz aleatoria (fondo azul en el mapa de calor), que abarca la primera fila y las tres primeras columnas. SVD detecta esto mediante la identificación de un gran valor singular (circular en rojo en la distribución de valores singulares) y cargas de fila grandes correspondientes ( $U_1$ ) así como tres cargas de columnas grandes ( $V_1$ ). A medida que se agregan más estructuras a la red (paneles superior derecha e inferior), se pueden descubrir usando SVD observando los siguientes valores singulares más grandes y las cargas de fila/columna correspondientes, etc.

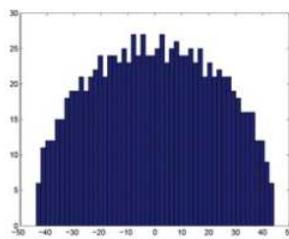


Figura 20.4: Ley de semicírculo Wigner

---

This page titled [20.3: Revisión de álgebra lineal](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [20.3: Linear Algebra Review](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.4: Análisis de componentes principales dispersos

### Limitaciones del Análisis de Componentes Principales

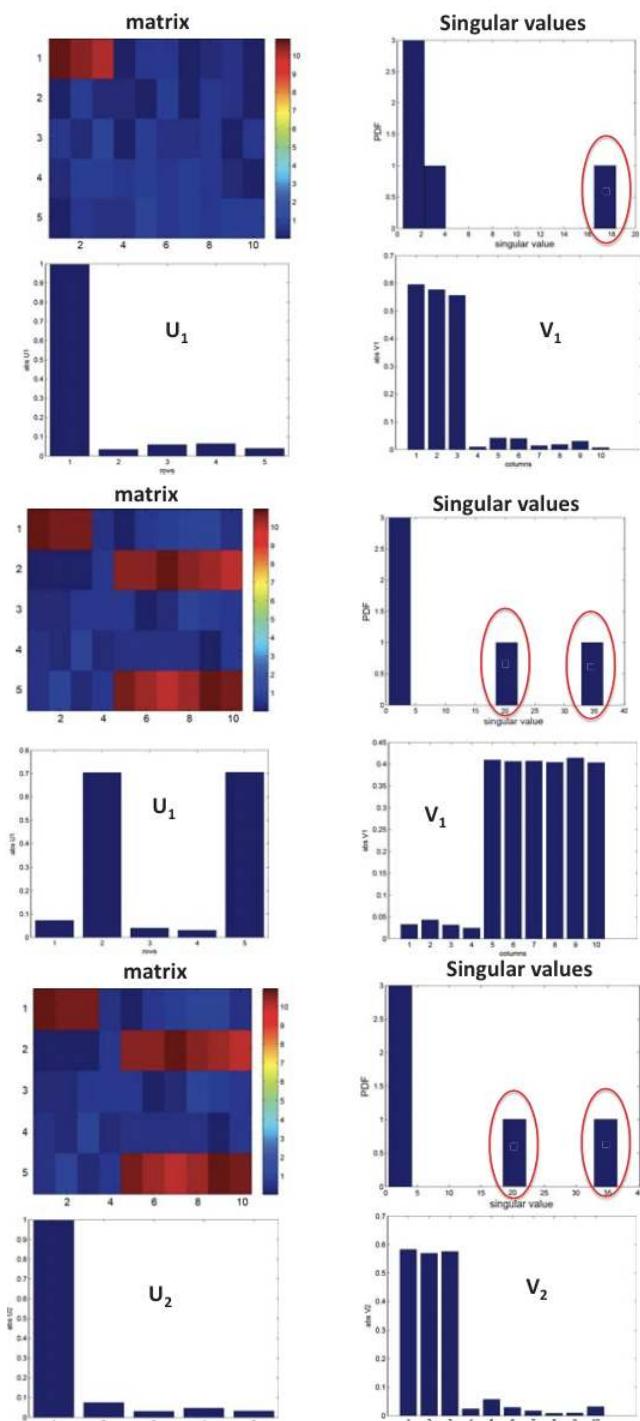
Al analizar datos de expresión génica basados en micromatrices, a menudo estamos tratando con matrices de datos de dimensiones  $m \times n$  donde  $m$  es el número de matrices y  $n$  es el número de genes. Por lo general  $n$  está en el orden de miles y  $m$  está en el orden de cientos. Nos gustaría identificar las características más importantes (genes) que mejor explican la variación de expresión, o patrones, en el conjunto de datos. Esto se puede hacer realizando PCA en la matriz de expresión:

$$\mathbf{E} = \mathbf{UDV}^T$$

Esto es en esencia un SVD de la matriz de expresión  $\mathbf{E}$  que gira y escala el espacio de características para que los vectores de expresión de cada gen en el nuevo sistema de coordenadas ortogonales estén lo más descorrelacionados posible, donde  $\mathbf{E}$  es la matriz de expresión  $m$  por  $n$ ,  $\mathbf{U}$  es la matriz  $m$  por  $m$  de vectores singulares izquierdos (es decir, principal componentes), o “genes propios”,  $\mathbf{V}$  es la matriz  $n$  por  $n$  de vectores singulares derechos, o “matrices propias”, y  $\mathbf{D}$  es una matriz diagonal de valores singulares, o “expresiones propias” de genes propios. Esto se ilustra en la Figura 20.6.

En PCA, cada componente principal (eigen-gen, una columna de  $\mathbf{U}$ ) es una combinación lineal de  $n$  variables (genes), que corresponde a un vector de carga (columna de  $\mathbf{V}$ ) donde las cargas son coecientes correspondientes a variables en la combinación lineal.

Sin embargo, una aplicación directa de PCA a matrices de expresión o cualquier matriz de datos grandes puede ser problemática porque los componentes principales (genes propios) son combinaciones lineales de todas las  $n$  variables (genes), lo cual es difícil de interpretar en términos de relevancia funcional. En la práctica nos gustaría utilizar una combinación del menor número posible de genes para explicar los patrones de expresión, lo que se puede lograr mediante una versión dispersa de PCA.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 20.5: Inferencia estructural usando SVD

## PCA disperso

El PCA disperso (SPCA) modifica el PCA para limitar los componentes principales (PC) a tener cargas escasas, reduciendo así el número de variables utilizadas explícitamente (genes en datos de microarrays, etc.) y facilitando la interpretación. Esto se hace formulando PCA como un problema de optimización de tipo regresión lineal e imponiendo restricciones de dispersión.

Un problema de regresión lineal toma un conjunto de variables de entrada  $\mathbf{x} = (1, x_1, \dots, x_p)$  y las variables de respuesta  $\mathbf{y} = \mathbf{x}\beta + \epsilon$  donde  $\beta$  es un vector de fila de coeficientes de regresión  $(\beta_0, \beta_1, \dots, \beta_p)^T$  y  $\epsilon$  es el error. El modelo de regresión para N observaciones se puede escribir en forma de matriz:

```
\left[ \begin{array}{c}
y_{1} \\
y_{2} \\
\vdots \\
y_{N}
\end{array} \right] = \left[ \begin{array}{cccccc}
1 & x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\
1 & x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_{N,1} & x_{N,2} & \cdots & x_{N,p}
\end{array} \right] \left[ \begin{array}{c}
\beta_0 \\
\beta_1 \\
\vdots \\
\beta_p
\end{array} \right] + \epsilon
```

El objetivo del problema de regresión lineal es estimar los coeficientes  $\beta$ . Hay varias formas de hacerlo, y los métodos más utilizados incluyen el método de mínimos cuadrados, el método Lasso y el método de la red elástica.

El método de **mínimos cuadrados** minimiza la suma residual del error cuadrado:

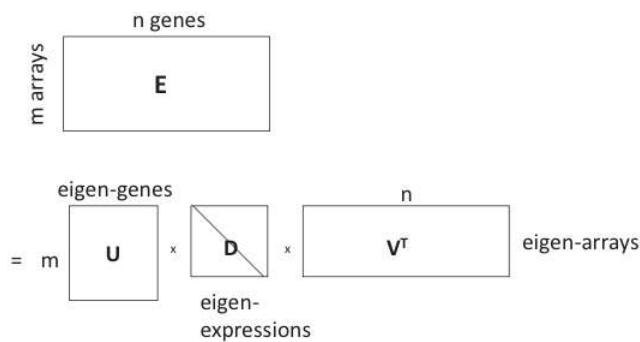
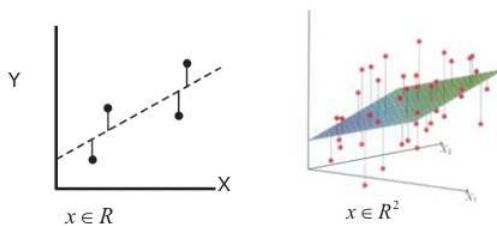


Figura 20.6: Descomposición del gen propio usando PCA

$$\hat{\beta} = \operatorname{argmin}_{\beta} \{RSS(\beta) | D\} \quad (20.4.1)$$

donde  $RSS(\beta) \equiv \sum_{i=1}^N (y_i - X_i \beta)^2$  ( $X_i$  es la i-ésima instancia de las variables de entrada  $x$ ). Esto se ilustra en la Figura 20.7 para los casos 2-D y 3-D, donde se produce una línea de regresión o un hiperplano.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 20.7: Solución de mínimos cuadrados de regresión lineal. Izquierda: 2-D caso, derecha: 3-D caso

El método de **Lazo** no solo minimiza la suma de errores residuales sino que al mismo tiempo minimiza una penalización de Lazo, que es proporcional a la norma L-1 del vector coffieciente  $\beta$ :

$$\hat{\beta} = \operatorname{argmin}_{\beta} \{ RSS(\beta) + L_1(\beta) | D \} \quad (20.4.2)$$

donde  $L_1(\beta) = \lambda \sum_{j=1}^p |\beta_j|$ ,  $\lambda \geq 0$ . La penalización ideal para PCA dispersa es la norma L<sub>0</sub> que penaliza a  $j=1$  cada elemento distinto de cero por 1, mientras que los elementos cero son penalizados por 0. Sin embargo, la función de penalización L<sub>0</sub> no es convexa y la mejor solución para explorar el espacio exponencial (número de combinaciones posibles de elementos distintos de cero) es NP-duro. La norma L<sub>1</sub> proporciona una aproximación convexa a la norma L<sub>0</sub>. El modelo de regresión Lasso en esencia reduce continuamente los coecientes hacia cero tanto como sea posible, produciendo un modelo escaso. Selecciona automáticamente para el conjunto más pequeño de variables que explican variaciones en los datos. Sin embargo, el método Lasso sues del problema de que si existe un grupo de variables altamente correlacionadas tiende a seleccionar solo una de estas variables. Además, Lasso selecciona como máximo N variables, es decir, el número de variables seleccionadas está limitado por el tamaño de la muestra.

El método **Elastic Net** elimina la limitación de selección de grupo del método Lasso agregando una restricción de arista:

$$\hat{\beta} = \operatorname{argmin}_{\beta} \{ RSS(\beta) + L_1(\beta) + L_2(\beta) | D \} \quad (20.4.3)$$

donde  $L_2(\beta) = \lambda_2 \sum_{j=1}^p |\beta_j|^2$ ,  $\lambda_2 \geq 0$ . En la solución de red elástica, se seleccionará un grupo de variables altamente correlacionadas una vez que se incluya una de ellas.

Todos los términos de penalización añadidos anteriores surgen del marco teórico de regularización. Nos saltamos las matemáticas detrás de la técnica y señalamos una explicación concisa en línea y un tutorial de regularización en [http://scikit-learn.org/stable/modul...ear\\_model.html](http://scikit-learn.org/stable/modul...ear_model.html).

El PCA puede ser reconstruido en un marco de regresión viendo cada PC como una combinación lineal de las variables p. Sus cargas pueden ser recuperadas por PC retrocediendo sobre las variables p (Figura 20.8). Dejar  $\mathbf{x} = \mathbf{UDV}^T$ .  $\forall i$ , denotar  $\mathbf{Y}_i = \mathbf{U}_i \mathbf{D}_{ii}$ , entonces  $\mathbf{Y}_i$  es el i-ésimo componente principal de X. Nosotros declaramos sin probar el siguiente teorema que confirma la corrección de la reconstrucción:

**Teorema 20.4.1.**  $\forall \lambda > 0$ , suppose  $\hat{\beta}_{\text{ridge}}$  es la estimación de cresta dada por

$$\hat{\beta}_{\text{ridge}} = \operatorname{argmin}_{\beta} |Y_i - X_i \beta|^2 + \lambda |\beta|^2$$

$$\text{and let } \hat{\mathbf{v}} = \frac{\hat{\beta}_{\text{ridge}}}{|\hat{\beta}_{\text{ridge}}|}, \text{then } \hat{\mathbf{v}} = \mathbf{V}_i$$

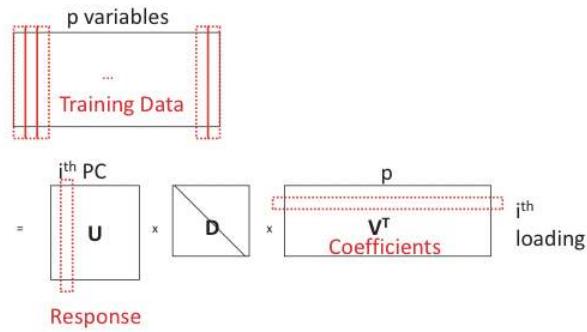


Figura 20.8: PCA en un marco de regresión

Tenga en cuenta que la penalización de cresta no penaliza a los coecientes sino que asegura la reconstrucción de los PCs. Tal problema de regresión no puede servir como alternativa al PCA ingenuo ya que usa exactamente sus resultados U en el modelo, pero puede modificarse agregando la penalización de Lazo al problema de regresión para penalizar por los valores absolutos de los coeficientes:

$$\hat{\beta}_{\text{ridge}} = \underset{\beta}{\operatorname{argmin}} |Y_i - X_i \beta|^2 + \lambda |\beta|^2 + \lambda_1 |\beta| \quad (20.4.4)$$

donde  $\mathbf{X} = \mathbf{UDV}^T$  y  $\forall i, Y_i = U_i D_{ii}$  es el i-ésimo componente principal de X. Los resultantes  $\hat{\beta}$  cuando se escalan por su norma son exactamente a lo que apunta SPCA: cargas escasas:

$$\hat{V}_i = \frac{\hat{\beta}}{|\hat{\beta}|} \approx V_i \quad (20.4.5)$$

$X\hat{V}_i \approx Y_i$  siendo el i-ésimo componente principal disperso.

Aquí damos un conjunto de datos de ejemplo simulado y comparamos la recuperación de factores ocultos usando PCA y SPCA. Tenemos 10 variables para las cuales generar puntos de datos:  $X = (X_1, \dots, X_{10})$ , y se utiliza un modelo de 3 factores ocultos  $V_1$ ,  $V_2$  y  $V_3$  para generar los datos:

```
\begin{array}{l}
V_1 \sim N(0,290) \\
V_2 \sim N(0,300) \\
V_3 \sim -0.3 V_1 + 0.925 V_2 + e, e \sim N(0,1) \\
X_i = V_1 + e_i, e_i \sim N(0,1), i=1,2,3,4 \\
X_i = V_2 + e_i, e_i \sim N(0,1), i=5,6,7,8 \\
X_i = V_3 + e_i, e_i \sim N(0,1), i=9,10 \\
\end{array}
```

De estos datos se espera que dos estructuras significativas surjan de un modelo de PCA disperso, cada una regida por factores ocultos  $V_1$  y  $V_2$  respectivamente ( $V_3$  es meramente una mezcla lineal de los dos). En efecto, como se muestra en la Figura 20.9, al limitar el número de variables utilizadas, SPCA recupera correctamente los PC explicando los efectos de  $V_1$  y  $V_2$  mientras que el PCA no distingue bien entre la mezcla de factores ocultos.

|                          | PCA    |        |        | SPCA ( $\lambda = 0$ ) |      |
|--------------------------|--------|--------|--------|------------------------|------|
|                          | PC1    | PC2    | PC3    | PC1                    | PC2  |
| $X_1$                    | 0.116  | -0.478 | -0.087 | 0.0                    | 0.5  |
| $X_2$                    | 0.116  | -0.478 | -0.087 | 0.0                    | 0.5  |
| $X_3$                    | 0.116  | -0.478 | -0.087 | 0.0                    | 0.5  |
| $X_4$                    | 0.116  | -0.478 | -0.087 | 0.0                    | 0.5  |
| $X_5$                    | -0.395 | -0.145 | 0.270  | 0.5                    | 0.0  |
| $X_6$                    | -0.395 | -0.145 | 0.270  | 0.5                    | 0.0  |
| $X_7$                    | -0.395 | -0.145 | 0.270  | 0.5                    | 0.0  |
| $X_8$                    | -0.395 | -0.145 | 0.270  | 0.5                    | 0.0  |
| $X_9$                    | -0.401 | 0.010  | -0.582 | 0.0                    | 0.0  |
| $X_{10}$                 | -0.401 | 0.010  | -0.582 | 0.0                    | 0.0  |
| Adjusted<br>Variance (%) | 60.0   | 39.6   | 0.08   | 40.9                   | 39.5 |

Figura 20.9: PCA vs SPCA

This page titled [20.4: Análisis de componentes principales dispersos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **20.4: Sparse Principal Component Analysis** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.5: Comunidades y Módulos de Red

¿Es posible usar redes para inferir las etiquetas de nodos no etiquetados, o datos? Suponiendo que algunos de los datos están etiquetados en una red, podemos usar la idea de que las redes capturan información relacional a través de una metodología de “Culpabilidad por asociación”. En pocas palabras, podemos mirar a los etiquetados “amigos” de un nodo en una red para inferir la etiqueta de un nuevo nodo. A pesar de que la forma de razonamiento “Culpabilidad por asociación” es una falacia lógica e insuficiente en entornos judiciales legales, a menudo es útil predecir etiquetas (por ejemplo, funciones génicas) para nodos en una red mirando las etiquetas de los vecinos de un nodo. Esencialmente, es probable que un nodo conectado a muchos nodos con la misma etiqueta también tenga esa etiqueta. En términos de redes biológicas donde los nodos representan genes, y los bordes representan interacciones (regulación, coexpresión, interacciones proteína-proteína etc., ver Figura 20.11), es posible predecir la función de un gen no anotado en base a las funciones de los genes a los que está conectado el gen consulta. Es fácil ver que podemos aplicar esto inmediatamente en un algoritmo iterativo, donde comenzamos con un conjunto de nodos etiquetados y nodos no etiquetados, y actualizamos iterativamente los atributos relacionales y luego reinferimos etiquetas de nodos. Iteramos hasta que todos los nodos estén etiquetados. Esto se conoce como el algoritmo de clasificación iterativa.

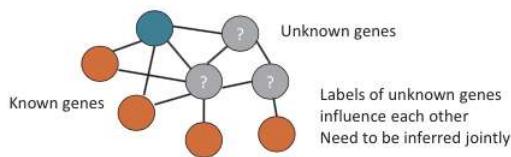


Figura 20.10: El algoritmo de clasificación iterativa

“Culpabilidad por asociación” implica una noción de asociación. La definición de asociación que consideramos implícitamente anteriormente es una definición sencilla donde consideramos todos los nodos conectados directamente a un nodo en particular. ¿Podemos dar una mejor definición de asociación? Considerando esta pregunta, llegamos naturalmente a la idea de comunidades, o módulos, en gráficas. El término comunidad intenta capturar la noción de región en una gráfica con nodos densamente conectados, vinculados a otras regiones de la gráfica con un número escaso de aristas. Las gráficas como estas, con subgrafías densamente conectadas, a menudo se denominan modulares. Obsérvese que no hay consenso sobre la definición exacta de comunidades. Para su uso práctico, la definición de comunidades debe estar motivada biológicamente e informada por conocimientos previos sobre el sistema que se modela. En biología, las redes reguladoras suelen ser modulares, con genes en cada subgrafía densamente conectada que comparten funciones y corregulación similares. Sin embargo, se han desarrollado amplias categorías de comunidades basadas en características topológicas diferentes. Se pueden dividir aproximadamente en 4 categorías: comunidades centradas en nodos, centradas en grupos, centradas en redes y centradas en jerarquías. Aquí examinamos un criterio de uso común para cada uno de los tres primeros tipos y recorremos brevemente algunos algoritmos conocidos que detectan estas comunidades.

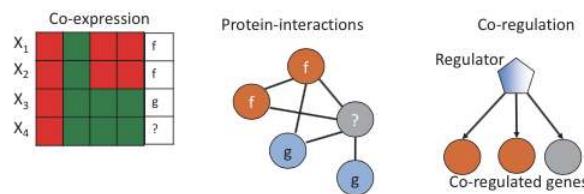


Figura 20.11: Predicción de la función génica usando asociaciones

### Comunidades centradas en nodos

Los criterios de comunidad centrados en *nodos* generalmente requieren que *cada nodo* de un grupo satisfaga ciertas propiedades. Una definición de comunidad centrada en nodos de uso frecuente es la **camarilla**, que es una subgrafía máxima completa en la que todos los nodos son adyacentes entre sí. La Figura 20.12 muestra un ejemplo de una camarilla (nodos 5,6,7 y 8) en una red.

Encontrar exactamente la camarilla máxima en una red es NP-hard, por lo que es muy costoso desde el punto de vista computacional implementar un algoritmo sencillo para la búsqueda de camarillas. La heurística se usa a menudo para limitar la complejidad del tiempo negociando una cierta fracción de precisión. Una heurística de uso común para el hallazgo de camarilla máxima se basa en la observación de que en una camarilla de tamaño k, cada nodo mantiene un grado de al menos k-1. Por lo tanto, podemos aplicar el siguiente procedimiento de poda:

- Muestrear una subred de la red dada y encontrar una camarilla en la subred usando una eficiente

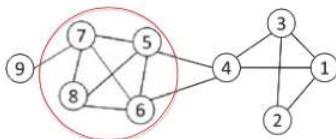


Figura 20.12: Una red de ejemplo que contiene una camarilla de 4

(por ejemplo, codicioso) enfoque

- Supongamos que la camarilla identificada tiene tamaño  $k$ , para encontrar una camarilla más grande, se eliminan todos los nodos con grado menor o igual a  $k-1$
- Repita hasta que la red sea lo suficientemente pequeña

En la práctica se podarán muchos nodos como redes sociales y muchas formas de redes biológicas siguen una ley de poder distribución de grados de nodo que da como resultado un gran número de nodos con grados bajos.

Tome la red en la Figura 20.12 para un ejemplo de tal procedimiento de búsqueda de camarilla. Supongamos que muestreamos una subred con nodos numerados del 1 al 9 y encontramos una camarilla  $\{1, 2, 3\}$  de tamaño 3. Para encontrar una camarilla con tamaño mayor a 3, eliminamos iterativamente los nodos al con grado  $\leq 2$ , es decir, los nodos  $\{2, 9\}$ ,  $\{1, 3\}$  y 4 se eliminarán secuencialmente. Esto nos deja con la camarilla de 4  $\{5, 6, 7, 8\}$ .

## Comunidades centradas en grupos

Los criterios de comunidad centrados en el grupo consideran las conexiones *dentro de un grupo* como un todo, y el grupo tiene que satisfacer ciertas propiedades sin hacer zoom en el nivel de nodo, por ejemplo, la densidad de borde del grupo debe exceder un umbral dado. Llamamos a un subgrafo  $G_s \left( V_s, E_s \right)$  una **cuasi-camarilla** densa si

$$\frac{2|E_s|}{|V_s||V_s - 1|} \geq \gamma \quad (20.5.1)$$

donde el denominador es el número máximo de aristas en la red. Con tal definición, se puede adoptar una estrategia similar a la heurística que discutimos para encontrar camarillas máximas:

- Muestree una subred y encuentre una camarilla γ quasi-densa máxima (por ejemplo, de tamaño  $|V_s|$ )
- Eliminar nodos con grado menor que el grado promedio ( $< |V_s| \gamma \leq \frac{2|E_s|}{|V_s|-1}$ )
- Repita hasta que la red sea lo suficientemente pequeña

## Comunidades centradas en la red

Las definiciones centradas en la red buscan dividir *toda la red* en varios conjuntos disjuntos. Existen varios enfoques para tal objetivo, como se enumeran a continuación:

- Algoritmo de agrupamiento de Markov [6]: El Algoritmo de Clustering de Markov (MCL) funciona haciendo una caminata de corrimiento en la gráfica y observando la distribución en estado estacionario de esta caminata. Esta distribución en estado estacionario permite agrupar la gráfica en subgrafías densamente conectadas.
- Algoritmo Girvan-Newman [2]: El algoritmo Girvan-Newman utiliza el número de rutas más cortas que pasan por un nodo para calcular la *esencialidad* de un borde que luego se puede usar para agrupar la red.
- Algoritmo de partición espectral

En esta sección veremos en detalle el algoritmo de particionamiento espectral. Referimos al lector a las referencias [2, 6] para una descripción de los otros algoritmos.

El algoritmo de partición espectral se basa en una cierta forma de representar una red usando una matriz. Antes de presentar el algoritmo introducimos una descripción importante de una red: su matriz laplaciana.

**Matriz laplaciana** Para el algoritmo de clustering que presentaremos más adelante en esta sección, necesitaremos contar el número de bordes entre los dos grupos diferentes en una partición de la red. Por ejemplo, en la Figura 21.6a, el número de aristas entre los dos grupos es de 1. La *matriz laplaciana* que vamos a introducir ahora es útil para representar esta cantidad algebraicamente. La matriz laplaciana  $L$  de una red en  $n$  nodos es una  $n \times n$  matriz  $L$  que es muy similar a la matriz de adyacencia  $A$  excepto por los cambios de signo y para los elementos diagonales. Mientras que los elementos diagonales de la matriz de adyacencia son siempre iguales a cero (ya que no tenemos auto-bucle), los elementos diagonales de la matriz Laplaciana mantienen el *grado* de cada nodo (donde el grado de un nodo se define como el número de aristas que inciden en él). También los elementos fuera de la diagonal de la matriz laplaciana se establecen en -1 en presencia de un borde, y cero en caso contrario. En otras palabras, tenemos:

$$L_{i,j} = \begin{cases} \text{degree}(i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and there is an edge between } i \text{ and } j \\ 0 & \text{if } i \neq j \text{ and there is no edge between } i \text{ and } j \end{cases} \quad (20.5.2)$$

Por ejemplo, la matriz laplaciana de la gráfica de la figura 21.6b viene dada por (enfatizamos los elementos diagonales en negrita):

```
\[L=\left[\begin{array}{ccc}
1 & 0 & -1 \\
0 & 1 & -1 \\
-1 & -1 & 2
\end{array}\right]\nonumber]
```

**Algunas propiedades de la matriz laplaciana** La matriz laplaciana de cualquier red disfruta de algunas propiedades agradables que serán importantes más adelante cuando veamos el algoritmo de clustering. Revisamos brevemente estos aquí.

La matriz laplaciana  $L$  es siempre **simétrica**, es decir,  $L_{i,j} = L_{j,i}$  para cualquier  $i, j$ . Una consecuencia importante de esta observación es que todos los valores propios de  $L$  son reales (es decir, no tienen parte imaginaria compleja). De hecho incluso se puede demostrar que los valores propios de  $L$  son todos no negativos.<sup>2</sup> La propiedad final que mencionamos sobre  $L$  es que todas las filas y columnas de  $L$  suman a cero (esto es fácil de verificar usando la definición de  $L$ ). Esto significa que el valor propio más pequeño de  $L$  es siempre igual a cero, y el vector propio correspondiente es  $s = (1, 1, \dots, 1)$ .

**Contando el número de aristas entre grupos usando la matriz Laplaciana** Usando la matriz Laplaciana ahora podemos contar fácilmente el número de aristas que separan dos partes disjuntas de la gráfica usando operaciones simples de matriz. En efecto, supongamos que particionamos nuestra gráfica en dos grupos, y que definimos un vector  $s$  de tamaño  $n$  que nos dice a qué grupo pertenece cada nodo  $i$ :

```
\[s_i = \left[\begin{array}{l}
1 \text{ if node } i \text{ está en el grupo 1} \\
-1 \text{ if node } i \text{ está en el grupo 2}
\end{array}\right]\nonumber]
```

Entonces se puede demostrar fácilmente que el número total de aristas entre el grupo 1 y el grupo 2 viene dado por la cantidad  $\frac{1}{4}s^T L s$  donde  $L$  es el Laplaciano de la red.

Para ver por qué este es el caso, primero calculemos el producto matriz-vector  $Ls$ . En particular, fijemos un nodo que digo en el grupo 1 (es decir,  $s_i = +1$ ) y veamos el componente  $i$ -ésimo del producto matriz-vector  $Ls$ . Por definición del producto matriz-vector tenemos:

```
\[(Ls)_i = \sum_{j=1}^n L_{i,j} s_j = L_{i,i} s_i + \sum_{j \text{ in group 1}} L_{i,j} s_j + \sum_{j \text{ in group 2}} L_{i,j} s_j
```

Podemos descomponer esta suma en tres summands de la siguiente manera:

$$(Ls)_i = \sum_{j=1}^n L_{i,j} s_j = L_{i,i} s_i + \sum_{j \text{ in group 1}} L_{i,j} s_j + \sum_{j \text{ in group 2}} L_{i,j} s_j$$

Usando la definición de la matriz Laplaciana vemos fácilmente que el primer término corresponde al grado de  $i$ , es decir, el número de aristas incidentes a  $i$ ; el segundo término es igual al negativo del número de aristas que conectan  $i$  a algún otro nodo del grupo 1, y el tercer término es igual al número de aristas conectando  $i$  a algún nodo ingroup 2. De ahí que tengamos:

$(Ls)_i = \text{grado}(i) (\# \text{ bordes de } i \text{ al grupo 1}) + (\# \text{ bordes de } i \text{ al grupo 2})$

Ahora como cualquier borde de  $i$  debe ir al grupo 1 o al grupo 2 tenemos

$\text{grado}(i) = (\# \text{ bordes de } i \text{ al grupo 1}) + (\# \text{ bordes de } i \text{ al grupo 2}).$

Así combinando las dos ecuaciones anteriores obtenemos:

$$(Ls)_i = 2 \times (\# \text{ edges from } i \text{ to group 2})$$

Ahora para obtener el número total de aristas entre el grupo 1 y el grupo 2, simplemente sumamos la cantidad sobre todos los nodos  $i$  del grupo 1:

Misplaced '#'

También podemos mirar nodos en el grupo 2 para calcular la misma cantidad y tenemos:

Misplaced '#'

Ahora promediando las dos ecuaciones anteriores obtenemos el resultado deseado:

Misplaced '#'

```
\begin{aligned}
&= \frac{1}{4} \sum_i (Ls)_i \\
&= \frac{1}{4} s^T L s
\end{aligned}
```

donde  $s^T$  es el vector de fila obtenido al transponer el vector de columna  $s$ .

**El algoritmo de agrupamiento espectral** Ahora veremos cómo se puede utilizar la vista de álgebra lineal de las redes dada en la sección anterior para producir una partición “buena” de la gráfica. En cualquier buena partición de una gráfica, el número de aristas entre los dos grupos debe ser relativamente pequeño en comparación con el número de aristas dentro de cada grupo. Así, una forma de abordar el problema es buscar una partición para que el número de aristas entre los dos grupos sea mínimo. Utilizando las herramientas introducidas en la sección anterior, este problema es así equivalente a encontrar un vector  $s \in \{-1, +1\}^n$  tomando solo valores -1 o +1 tal que  $\frac{1}{4}s^T L s$  sea mínimo, donde  $L$  es la matriz Laplaciana de la gráfica. En otras palabras, queremos resolver el problema de minimización:

$$\underset{s \in \{-1, +1\}^n}{\text{minimize}} \frac{1}{4} s^T L s$$

Si  $s^*$  es la solución óptima, entonces la partición óptima es asignar el nodo  $i$  al grupo 1 si  $s_i = +1$  o bien al grupo 2.

Esta formulación parece tener sentido pero desafortunadamente hay una pequeña falla: la solución a este problema siempre terminará siendo  $s = (+1, \dots, +1)$  lo que corresponde a poner todos los nodos de la red en el grupo 1, ¡y ningún nodo en el grupo 2! El número de aristas entre el grupo 1 y el grupo 2 es entonces simplemente cero y, de hecho, ¡es mínimo!

Para obtener una partición significativa tenemos que considerar así particiones de la gráfica que no son triviales. Recordemos que la matriz laplaciana  $L$  es siempre simétrica, y así admite una descomposición propia:

$$L = U \Sigma U^T = \sum_{i=1}^n \lambda_i u_i u_i^T$$

donde  $\sigma$  es una matriz diagonal que contiene los valores propios no negativos  $\lambda_1, \dots, \lambda_n$  de  $L$  y  $U$  es la matriz de vectores propios y satisface  $U^T = U^{-1}$ .

El costo de una partición  $s \in \{-1, +1\}^n$  viene dado por

$$\frac{1}{4} s^T L s = \frac{1}{4} s^T U \Sigma U^T s = \frac{1}{4} \sum_{i=1}^n \lambda_i \alpha_i^2$$

donde  $\alpha = U^T s$  dan la descomposición de  $s$  como una combinación lineal de los vectores propios de  $L: s = \sum_{i=1}^n \alpha_i u_i$ .

Recordemos también que  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . Así, una forma de hacer que la cantidad anterior sea lo más pequeña posible (sin escoger la partición trivial) es concentrar todo el peso en 2 que es el valor propio distinto de cero más pequeño de L. Para lograr esto simplemente elegimos s de manera que  $\alpha_2 = 1$  y  $\alpha_k = 0$  para todos  $k \neq 2$ . En otras palabras, esto corresponde a tomar s para que sea igual a  $u_2$ , el segundo vector propio de L. Dado que en general el autovector  $u_2$  no es de valor entero (es decir, los componentes de  $u_2$  pueden ser diferentes a -1 o +1), tenemos que convertir primero

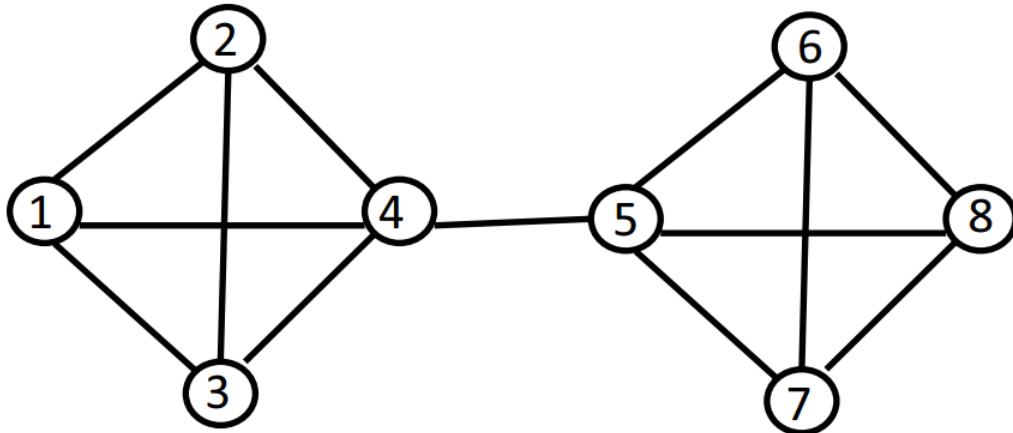


Figura 20.13: Una red con 8 nodos.

el vector  $u_2$  en un vector de  $+1$  o  $-1$ . Una manera simple de hacer esto es solo mirar los signos de los componentes de  $u_2$  en lugar de los valores mismos. Nuestra partición está así dada por:

```
\[s=\ nombreoperador {signo}\ izquierda (u_{2}\ derecha) =\ izquierda\ {\begin {array} {ll}
1 &\text{if}\ left(u_{2}\ derecha)_i \geq 0 \\
-1 &\text{if}\ left(u_{2}\ derecha)_i < 0
\end {array}} derecha.\nonumber\]
```

Para recapitular, el algoritmo de agrupamiento espectral funciona de la siguiente manera:

## Algoritmo de partición spectral

- Entrada: una red
  - Salida: una partición de la red donde cada nodo se asigna ya sea al grupo 1 o al grupo 2 para que el número de bordes entre los dos grupos sea pequeño

1. Calcular la matriz Laplaciana L de la gráfica dada por:

$\backslash [L_{-} \{i, j\}] = \backslash left \{ \backslash begin \{array\} \{ll\}$

\operatorname{degree} (i) \& \text{if} i=j \\\

-1 &\text{if} \neq \text{y hay un borde entre} \text{y} \&

0 & \text{if} { i } \neq j \text{ \text{y no hay borde entre} } { i } \text{ \text{y} } { j }

\end{array}\right.\nonumber\\

2. Calcular el autovector  $\mathbf{u}_2$  para el segundo valor propio más pequeño de  $\mathbf{L}$ .

3. Salida de la siguiente partición: Asignar el nodo  $i$  al grupo 1 si  $(\mathbf{u}_i \cdot \mathbf{e}) > 0$ , de lo contrario asignar el nodo  $i$  al grupo 2.

A continuación damos un ejemplo donde aplicamos el algoritmo de clustering espectral a una red con 8 nodos.

**Ejemplo** Ilustramos aquí el algoritmo de particionamiento descrito anteriormente en una red simple de 8 nodos dado en la figura 21.7. La matriz de adyacencia y la matriz Laplaciana de esta gráfica se dan a continuación:

```
\[A=\left[\begin{array}{lllllll}
0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 & 1 & 0
\end{array}\right]
```

Usando el comando `eig` de Matlab podemos calcular la descomposición propia  $L = U\Sigma U^T$  de la matriz Laplaciana y obtenemos:

```

\begin{array}{rrrrrrr}
0.3536 & -0.3825 & 0.2714 & -0.1628 & -0.7783 & 0.0495 & -0.0064 & -0.1426 \\
0.3536 & -0.3825 & 0.5580 & -0.1628 & 0.6066 & 0.0495 & -0.0064 & -0.1426 \\
0.3536 & -0.3825 & -0.4495 & 0.6251 & 0.0930 & 0.0495 & -0.3231 & -0.1426 \\
0.3536 & -0.2470 & -0.3799 & -0.2995 & 0.0786 & -0.1485 & 0.3358 & 0.6626 \\
0.3536 & \mathbf{0.2470} & -0.3799 & -0.2995 & 0.0786 & -0.1485 & 0.3358 & -0.6626 \\
0.3536 & \mathbf{0.3825} & 0.3514 & 0.5572 & -0.0727 & -0.3466 & 0.3860 & 0.1426 \\
0.3536 & \mathbf{0.3825} & 0.0284 & -0.2577 & -0.0059 & -0.3466 & -0.7266 & 0.1426 \\
0.3536 & \mathbf{0.3825} & 0.0000 & 0.0000 & 0.0000 & 0.8416 & -0.0000 & 0.1426 \\
\end{array}
\end{aligned}
\nonumber

```

Hemos resaltado en negrita el segundo valor propio más pequeño de  $L$  y el autovector asociado. Para agrupar la red observamos el signo de los componentes de este vector propio. Vemos que los primeros 4 componentes son negativos, y los últimos 4 componentes son positivos. Así, agruparemos los nodos 1 a 4 juntos en el mismo grupo, y los nodos 5 a 8 en otro grupo. Esto parece un buen agrupamiento y de hecho esta es la agrupación “natural” que se considera a primera vista de la gráfica.

## ¿Sabías?

El problema matemático que formulamos como motivación para el algoritmo de agrupamiento espectral es encontrar una partición de la gráfica en dos grupos con un número mínimo de bordes entre los dos grupos. El algoritmo de particionamiento espectral que presentamos no siempre da una solución óptima a este problema pero suele funcionar bien en la práctica.

En realidad resulta que el problema tal y como lo formulamos se puede resolver exactamente usando un algoritmo eficiente. El problema a veces se llama el problema de corte mínimo ya que estamos buscando cortar un número mínimo de bordes de la gráfica para desconectarla (los bordes que cortamos son los que se encuentran entre el grupo 1 y el grupo 2). El problema de corte mínimo se puede resolver en tiempo polinomial en general, y remitimos al lector a la entrada de Wikipedia sobre corte mínimo [9] para más información. El problema sin embargo con las particiones de corte mínimo es que generalmente conducen a particiones de la gráfica que no están equilibradas (por ejemplo, un grupo tiene solo 1 nodo, y los nodos restantes están todos en el otro grupo). En general, uno quisiera imponer restricciones adicionales a los clústeres (por ejemplo, límites inferiores o superiores en el tamaño de los clústeres, etc.) para obtener clústeres más realistas. Con tales limitaciones, el problema se vuelve más difícil, y remitimos al lector a la entrada de Wikipedia sobre Particionamiento gráfico [8] para más detalles.

### Preguntas frecuentes

P: ¿Cómo dividir la gráfica en más de dos grupos?

R: En esta sección solo nos fijamos en el problema de particionar la gráfica en dos clústeres. ¿Y si queremos agrupar la gráfica en más de dos clústeres? Hay varias extensiones posibles del algoritmo que se presentan aquí para manejar  $k$  clústeres en lugar de solo dos. La idea principal es mirar los  $k$  vectores propios para los  $k$  valores propios distintos de cero más pequeños del Laplaciano, y luego aplicar el algoritmo de clustering k-means apropiadamente. Refirimos al lector al tutorial [7] para más información.

<sup>2</sup> Una forma de ver esto es notar que  $L$  es diagonalmente dominante y los elementos diagonales son estrictamente positivos (para más detalles el lector puede buscar “diagonalmente dominante” y “teorema del círculo Gershgorin” en Internet).

This page titled [20.5: Comunidades y Módulos de Red](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **20.5: Network Communities and Modules** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.6: Núcleo de Difusión en Red

Anteriormente, definimos una métrica de distancia entre dos nodos como la ruta más corta ponderada. Esta métrica de distancia simple es suficiente para muchos propósitos, pero notablemente no utiliza ninguna información sobre la estructura general de la gráfica. Muchas veces, definir la distancia en función del número de caminos posibles entre dos nodos, ponderados por la verosimilitud o probabilidad de tomar tales caminos, da una mejor representación del sistema real que estamos modelando. Exploramos métricas de distancia alternativas en esta sección.

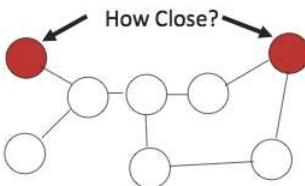


Figura 20.14: ¿Qué otras métricas de distancia son útiles?

Las matrices de kernel de difusión ayudan a capturar la estructura de red global de los gráficos, informando una definición más compleja de distancia.

Que  $A$  sea nuestra matriz de adyacencia regular.  $D$  es la matriz diagonal de grados. Podemos definir  $L$ , la matriz laplaciana, de la siguiente manera:

$$L = D - A$$

Luego definimos un núcleo de difusión  $K$  como

$$K = \exp(-\beta L)$$

Dónde  $\beta$  está el parámetro de difusión. Tenga en cuenta que estamos tomando una matriz exponencial y no una exponencial por elementos, que se basa en la expansión de la serie Taylor de la siguiente manera:

$$= \sum_{k=0}^{\infty} \frac{1}{k} (-\beta L)^k$$

Entonces, ¿qué representa la matriz  $K$ ? Hay múltiples formas de interpretar  $K$ , enumeraremos las más relevantes para nosotros a continuación:

**Caminatas aleatorias** — Una forma de interpretar  $K$  como el resultado de una caminata aleatoria. Supongamos que tenemos una gráfica y en el nodo de interés, tenemos una distribución de probabilidad sobre los bordes que representa esa probabilidad de que nos movemos a lo largo de ese borde. Al igual que la siguiente figura:

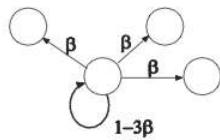


Figura 20.15: Ilustración de una caminata aleatoria

$\beta$  es la probabilidad de transición a lo largo de un borde específico. Y también hay una probabilidad de que no nos movemos (representado aquí como un bucle self). Tenga en cuenta que para que la distribución de probabilidad sea válida debe sumar hasta 1.

Si tenemos la configuración anterior, entonces  $K_{ij}$  es igual a la probabilidad de que la caminata que comenzó en  $i$  esté en  $j$  después de pasos de tiempo infinitos. Para derivar ese resultado, podemos escribir nuestra gráfica como modelo de Markov y tomar el límite comot  $\rightarrow \infty$

**Proceso estocástico** — Otra forma en que podemos interpretar el núcleo de difusión es a través de un proceso estocástico.

- para cada nodo  $i$ , considere una variable aleatoria  $Z_i(t)$

- dejar que  $Z_i(t)$  sea la media cero con alguna varianza definida.
- la covarianza para  $Z_i(t)$  y  $Z_j(t)$  es cero (independientes entre sí).
- cada variable envía una fracción a los vecinos

```
\begin{array}{c}
Z_{i(t+1)} = Z_{i(t)} + \beta \sum_{j \neq i} (Z_{j(t)} - Z_{i(t)}) \\
Z(t+1) = (I - \beta L) Z(t) \\
Z(t) = (I - \beta L)^t Z(0) \\
\end{array}
```

dejar que el operador de evolución temporal  $T(t)$  sea

$$T(t) = (I - \beta L)^t$$

entonces la covarianza es igual a

$$\text{Cov}_{ij}(t) = \sigma^2 T_{ij}(2t)$$

Entonces a medida que  $\Delta t \rightarrow 0$  tomamos conseguimos

$$\text{Cov}(t) = \sigma^2 \exp(-2\beta t L)$$

This page titled [20.6: Núcleo de Difusión en Red](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [20.6: Network Diffusion Kernels](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.7: Redes neuronales

Las redes neuronales salieron modelando el cerebro y el sistema nervioso en un intento de lograr un aprendizaje similar al del cerebro. Son muy paralelos y al aprender conceptos simples podemos lograr comportamientos muy complejos. En relevancia para este libro, también han demostrado ser muy buenos modelos biológicos (no es de extrañar dar de dónde surgieron).

### Redes de alimentación directa

En una red neuronal mapeamos la entrada a la salida pasando por estados ocultos que son parametrizados por aprendizaje.

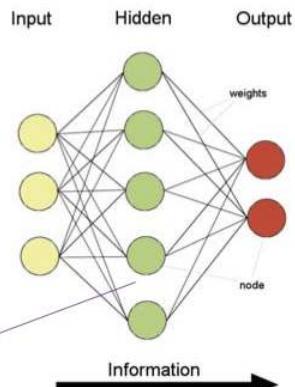


Figura 20.16: Ilustración de una red neuronal

- El flujo de información es unidireccional
- Los datos se presentan a la capa de entrada
- Pasado a capa oculta
- Pasado a capa de salida
- La información se distribuye
- El procesamiento de la información es paralelo

### Back-propagación

La retropropagación es uno de los resultados más influyentes para entrenar redes neuronales y permitirnos lidiar fácilmente con redes multicapa.

- Requiere conjunto de entrenamiento (pares de entrada-salida)
- Comienza con pequeños pesos aleatorios
- Error se usa para ajustar pesos (aprendizaje supervisado)

Básicamente realiza descenso de gradiente en el paisaje de error tratando de minimizar el error. Así, la retropropagación puede ser lenta.

### Aprendizaje Profundo

El aprendizaje profundo es una colección de técnicas estadísticas de aprendizaje automático utilizadas para aprender jerarquías de características. A menudo se basa en redes neuronales artificiales. Las redes neuronales profundas tienen más de una capa oculta. Cada capa sucesiva en una red neuronal utiliza entidades en la capa anterior para aprender entidades más complejas. Uno de los objetivos (relevantes) de los métodos de aprendizaje profundo es realizar la extracción jerárquica de características. Esto hace que el aprendizaje profundo sea un enfoque atractivo para modelar procesos generativos jerárquicos como se encuentran comúnmente en la biología de sistemas.

#### Ejemplo: DeepBind (Alipanahi et al. 2015)

DeepBind [1] es una herramienta de aprendizaje automático desarrollada por Alipanahi et al. para predecir las especificidades de secuencia de proteínas de unión a ADN y ARN utilizando métodos basados en el aprendizaje profundo.

Los autores señalan tres diferencias encontradas cuando se entrenaen modelos de secuencia de especificidades sobre los grandes volúmenes de datos de secuencia producidos por tecnologías modernas de alto rendimiento: (a) los datos vienen en formas cualitativamente diferentes, incluyendo microarrays de unión a proteínas, ensayos RNACompete, ChIP- seq y HT -SELEX, (b) la cantidad de datos es muy grande (los experimentos típicos miden de diez a cien mil secuencias y (c) cada tecnología de adquisición de datos tiene sus propios formatos y perfil de error y por lo tanto se necesita un algoritmo que sea robusto a estos efectos no deseados.

El método DeepBind es capaz de resolver estas diferencias mediante (a) implementación paralela en una unidad de procesamiento gráfico, (b) tolerar un grado moderado de ruido y datos de entrenamiento mal clasificados y (c) entrenar el modelo predictivo de manera automática evitando la necesidad de afinación manual. Las siguientes figuras ilustran aspectos del pipeline Deep Bind.

Para abordar la preocupación por el sobreajuste, los autores utilizaron varios regularizadores, incluyendo deserción, decaimiento de peso y parada temprana.

### Deserción: Prevención de Sobre-Ajuste

La deserción escolar [5] es una técnica para abordar el problema del sobreajuste en los datos de entrenamiento en el contexto de grandes redes. Debido a la multiplicación de gradientes en el cálculo de la regla de cadena, se coadaptan pesos unitarios ocultos, lo que puede conducir a un sobreajuste. Una forma de evitar la co-adaptación de pesos unitarios ocultos es simplemente soltar unidades (aleatoriamente). Una consecuencia beneficiosa de la caída de unidades es que las redes neuronales más grandes son más intensivas computacionalmente para entrenar.

No obstante, este enfoque toma un poco más de tiempo con respecto a la formación. Además, afinar el tamaño del paso es un desafío. Los autores proporcionan un Apéndice, en el que (en la parte (A)) proporcionan una útil “Guía práctica para formar redes de deserción escolar”. Señalan que los valores típicos para el parámetro dropout  $p$  (que

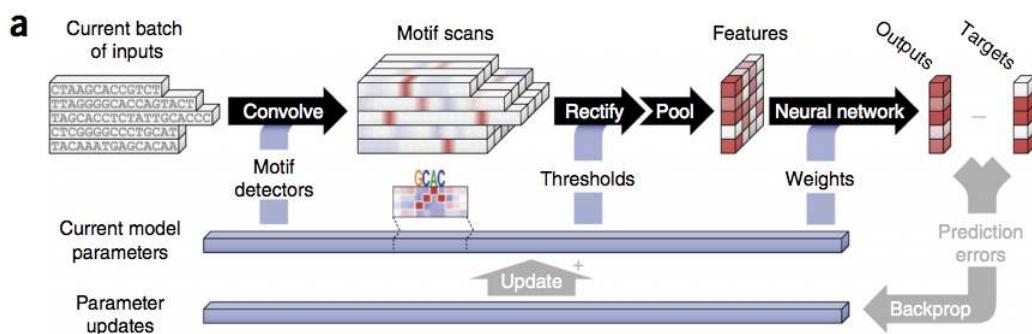


Figura 20.17: Un diagrama de flujo del procedimiento DeepBind (tomado del documento DeepBind). Cinco secuencias están siendo procesadas en paralelo por el modelo. El modelo convoluciona las secuencias (podemos pensar en el modelo deepbind como un filtro que explora las secuencias), las rectifica y las agrupa para producir un vector de características que luego pasa a través de una red neuronal profunda. La salida de la red profunda se compara con la salida deseada y el error se retropropaga a través de la tubería.

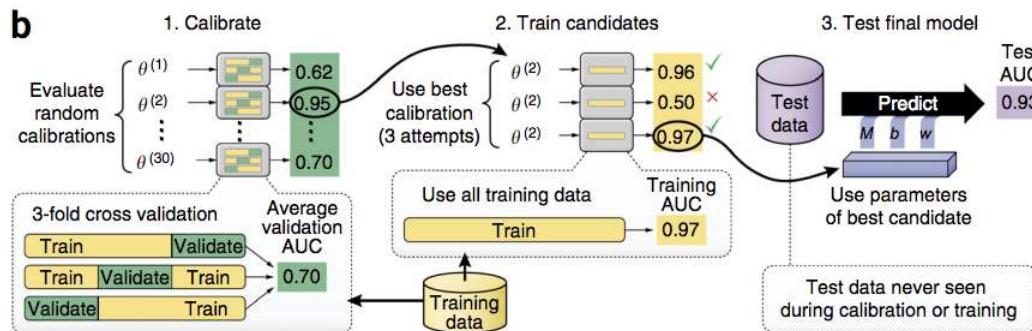


Figura 20.18: Ilustración del procedimiento de calibración, entrenamiento y pruebas utilizado por el método DeepBind (tomado del artículo DeepBind).

Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Alipanahi, Babak, Andrew Delong, et al. “Predecir las especificidades de secuencia de Proteínas de unión a ADN y ARN por Deep Learning”. *Biotecnología de la naturaleza* (2015)

determina la probabilidad de que se caiga un nodo) están entre 0.5 y 0.8 para las capas ocultas y 0.8 para las capas de entrada.

---

This page titled [20.7: Redes neuronales](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [20.7: Neural Networks](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.8: Temas abiertos y desafíos

Algunos de los retos con respecto a los temas tratados anteriormente son

- Validación ¿Cómo sabemos que la estructura de la red es correcta?
- ¿Cómo sabemos si la función de red es correcta?
- Medir y modelar la expresión de proteínas
- Comprender la evolución de las redes reguladoras
- En su mayoría, es intratable calcular distribuciones conjuntas por lo que nos enfocamos en distribuciones marginales.
- A menudo tenemos un número muy grande de reguladores u objetivos haciendo que algunos de los problemas requieran una suposición simplificadora para poder hacerlo tratable.

---

This page titled [20.8: Temas abiertos y desafíos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **20.8: Open Issues and Challenges** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía

Para conocer más sobre los temas tratados en este capítulo, puede buscar los siguientes términos clave.

- Modelos gráficos probabilísticos
- Finalización de la red
- Factorización de matriz no negativa
- Alineación de Redes
- Integración de Redes

---

This page titled 20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **20.9: Further Reading, What Have We Learned?, Bibliography** by Manolis Kellis et al. is licensed CC BY-NC-SA 4.0. Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 20.10: ¿Qué hemos aprendido?

- Las redes vienen en varios tipos y se pueden representar en vistas probabilísticas y algebraicas
- Diferentes medidas de centralidad miden la importancia de los nodos/bordes a partir de diferentes aspectos
- PCA y SVD son útiles para descubrir patrones estructurales en la red mediante la realización de la descomposición de la matriz
- La PCA dispersa mejora la PCA al seleccionar algunas variables más representativas en los datos y recupera con mayor precisión la estructura de la comunidad
- Las comunidades de red tienen una variedad de definiciones, cada una de las cuales tiene algoritmos específicos diseñados para la detección de comunidades
- Las redes neuronales y las redes de aprendizaje profundo son máquinas de aprendizaje supervisado que capturan patrones complejos en datos.

20.10: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [20.10: What Have We Learned?](#) has no license indicated.

## Bibliografía

- [1] B. Alipanahi, A. Delong, M.T. Weirauch, y B.J. Frey. Predecir las especificidades de secuencia de ADN y proteínas de unión a ARN mediante aprendizaje profundo. *Nature Biotechnology*, 33:831 —838, 2015.
- [2] M. Girvan y M.E.J. Newman. Estructura comunitaria en redes sociales y biológicas. *Actas de la Academia Nacional de Ciencias*, 99 (12) :7821—7826, 2002.
- [3] O. Hein, M. Schwind y W. K. onig. Redes libres de escala: El impacto de la distribución de grados de cola de grasa en los procesos de difusión y comunicación. *Wirtschaftsinformatik*, 48 (4) :267—275, 2006.
- [4] T.I. Lee, N.J. Rinaldi, F. Robert, D.T. Odom, Z. Bar-Joseph, G.K. Gerber, N.M. Hannett, C.T. Harbison, C.M. Thompson, I. Simon, et al. Redes reguladoras transcripcionales en *saccharomyces cerevisiae*. *Señalización científica*, 298 (5594) :799, 2002.
- [5] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever y R. Salakhutdinov. Abandono: Una forma sencilla de evitar que las redes neuronales se sobreajusten. *Revista de Investigación en Aprendizaje Automático*, 15:1929 —1958, 2014.
- [6] S.M. van Dongen. Agrupación gráfica por simulación de flujo. Tesis de doctorado, Universidad de Utrecht, Las tierras abisales, 2000.
- [7] U. Von Luxburg. Un tutorial sobre agrupamiento espectral. *Estadística y computación*, 17 (4) :395—416, 2007.
- [8] Wikipedia. Partición gráfica. [es.wikipedia.org/wiki/Graph\\_Partitionado](https://es.wikipedia.org/wiki/Graph_Partitionado), 2012.
- [9] Wikipedia. Corte mínimo. [es.wikipedia.org/wiki/Minimum\\_cut](https://es.wikipedia.org/wiki/Minimum_cut), 2012.

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 21: Redes Regulatorias- Inferencia, Análisis, Aplicación

[21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación](#)

[21.2: Inferencia de estructura](#)

[21.3: Visión general de la tarea de aprendizaje PGM](#)

[21.4: Aplicación de Redes](#)

[21.5: Propiedades Estructurales de Redes](#)

[21.6: Clustering de redes, Bibliografía](#)

[Bibliografía](#)

---

This page titled [21: Redes Regulatorias- Inferencia, Análisis, Aplicación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación

Los sistemas vivos están compuestos por múltiples capas que codifican información sobre el sistema. Las capas primarias son:

1. Epigenoma: Definido por la configuración de la cromatina. La estructura de la cromatina se basa en la forma en que las histonas organizan el ADN. El ADN se divide en regiones libres de nucleosomas y nucleosomas, formando su forma final e influyendo en la expresión génica.
2. Genoma: Incluye ADN codificante y no codificante. Los genes definidos por ADN codificante se utilizan para construir ARN, y los elementos reguladores de la CIS regulan la expresión de estos genes.
3. Los ARN del transcriptoma (por ejemplo, ARNm, miARN, ARNc, ARNpi) se transcriben a partir del ADN. Tienen funciones reguladoras y fabrican proteínas.
4. Proteoma Compuesto por proteínas. Esto incluye factores de transcripción, proteínas de señalización y enzimas metabólicas.

Las interacciones entre estos componentes son todas diferentes, pero entenderlas puede poner partes particulares del sistema en el contexto del todo. Para descubrir relaciones e interacciones dentro y entre capas, podemos usar redes.

### Presentamos Redes Biológicas

Las redes biológicas están compuestas de la siguiente manera: **Red**

**Reguladora** — conjunto de interacciones reguladoras en un organismo.

- Los nodos son reguladores (por ejemplo, factores de transcripción) y dianas asociadas.
- Los bordes corresponden a la interacción regulatoria, dirigida desde el factor regulatorio hasta su objetivo. Se firman según el efecto positivo o negativo y se ponderan según la fuerza de la reacción.

**Metabolic Net** — conecta los procesos metabólicos. Hay cierta flexibilidad en la representación, pero un ejemplo es una gráfica que muestra productos metabólicos compartidos entre enzimas.

- Los nodos son enzimas.
- Los bordes corresponden a reacciones reguladoras, y se ponderan de acuerdo con la fuerza de la reacción.

**Red de Señalización** — representa rutas de señales biológicas.

- Los nodos son proteínas llamadas receptores de señalización.
- Los bordes se transmiten y reciben señales biológicas, dirigidas del transmisor al receptor.

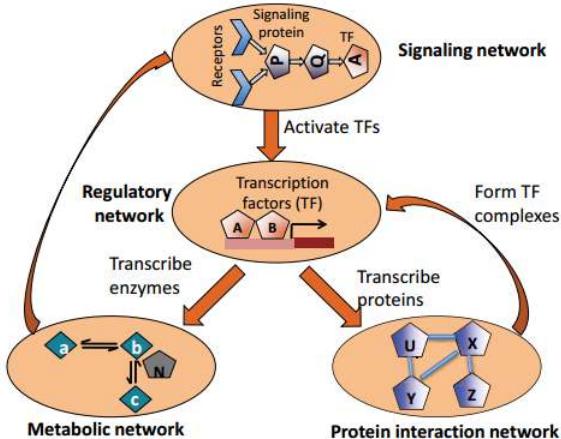
**Protein Net:** muestra interacciones físicas entre proteínas.

- Los nodos son proteínas individuales.
- Los bordes son interacciones físicas entre proteínas.

**Red de coexpresión:** describe las funciones de coexpresión entre genes. Muy general; representa redes de interacción funcionales más que físicas, a diferencia de los otros tipos de redes. Potente herramienta en el análisis computacional de datos biológicos.

- Los nodos son genes individuales.
- Los bordes son relaciones de coexpresión.

Hoy, nos centraremos exclusivamente en las redes regulatorias. Las redes reguladoras controlan la expresión génica específica del contexto y, por lo tanto, tienen un gran control sobre el desarrollo. Vale la pena estudiarlos porque son propensos al mal funcionamiento y a causar enfermedades.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

a) Interacciones entre redes biológicas.

## Interacciones entre redes biológicas

Las redes biológicas individuales (es decir, capas) pueden considerarse nodos en una red más grande que representa todo el sistema biológico. Podemos, por ejemplo, tener una red de señalización detectando el entorno que gobierna la expresión de los factores de transcripción. En este ejemplo, la red mostraría que los TFs gobiernan la expresión de proteínas, las proteínas pueden desempeñar papeles como enzimas en las vías metabólicas, y así sucesivamente.

Las rutas generales de intercambio de información entre estas redes se muestran en la figura 21.4.

## Estudiando Redes Regulatorias

En general, las redes se utilizan para representar dependencias entre variables. Las dependencias estructurales se pueden representar por la presencia de un borde entre nodos; como tal, los nodos no conectados son condicionalmente independientes. Probabilísticamente, a los bordes se les puede asignar un “peso” que represente la fuerza o la probabilidad de la interacción. Las redes también pueden verse como matrices, permitiendo operaciones matemáticas. Estos marcos proporcionan una manera efectiva de representar y estudiar sistemas biológicos.

Estas redes son particularmente interesantes de estudiar porque el mal funcionamiento puede tener un gran efecto. Muchas enfermedades son causadas por las reconexiones de las redes regulatorias. Controlan la expresión específica del contexto en el desarrollo. Debido a esto, se pueden usar en biología de sistemas para predecir el desarrollo, el estado celular, el estado del sistema y más. Además, encapsulan gran parte de la diferencia evolutiva entre organismos que son genéticamente similares.

Para describir las redes regulatorias, hay varias preguntas desafiantes que responder.

**Identificación de elementos** ¿Cuáles son los elementos de una red? En la última conferencia se identificaron elementos constitutivos de redes regulatorias. Estos incluyen motivos aguas arriba y sus factores asociados.

**Análisis de estructura de red** ¿Cómo se conectan los elementos de una red? Dada una red, el análisis de estructura consiste en el examen y caracterización de propiedades importantes. Se puede hacer redes biológicas pero no se restringe a ellas.

**Inferencia de red** ¿Cómo interactúan los reguladores y activan los genes? Esta es la tarea de identificar los bordes de los genes y caracterizar sus acciones.

**Aplicaciones de red** ¿Qué podemos hacer con las redes una vez que las tenemos? Las aplicaciones incluyen la función de predicción de genes reguladores y la predicción de los niveles de expresión de genes regulados.

<sup>1</sup> Más en la conferencia de epigenética.

This page titled [21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [21.1: Regulatory Networks- Inference, Analysis, Application](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 21.2: Inferencia de estructura

### Preguntas Clave en la Inferencia de Estructuras

**¿Cómo elegir modelos de red?** Existen varios modelos para representar redes, un problema clave es elegir entre ellos en función de los datos y la dinámica predicha.

**¿Cómo elegir los métodos de aprendizaje?** Existen dos métodos amplios para las redes de aprendizaje. Los métodos no supervisados intentan inferir relaciones para puntos de datos no alabeados y se describirán en secciones por venir. Los métodos supervisados toman un subconjunto de bordes de red que se sabe que son regulatorios y aprenden un clasificador para predecir otros nuevos.<sup>2</sup>

**¿Cómo incorporar datos?** Se puede usar una variedad de fuentes de datos para aprender y construir redes, incluidos Motivos, ensayos de unión a ChIP y expresión. Las fuentes de datos están siempre en expansión; la expansión de la disponibilidad de datos está en el corazón de la revolución actual en el análisis de redes biológicas.

### Representaciones matemáticas abstractas para redes

Piense en una red como una función, una caja negra. Las redes regulatorias, por ejemplo, toman expresiones de entrada de los reguladores y escupen la expresión de salida de los objetivos. Los modelos difieren en elegir la naturaleza de las funciones y asignar significado a nodos y bordes.

**Red booleana** Este modelo discretiza los niveles de expresión de los nodos y las interacciones. Las funciones representadas por bordes son puertas lógicas.

**Modelo de Ecuación Diferencial** Estos modelos capturan dinámicas de red. Los cambios en la tasa de expresión son función de los niveles de expresión y las tasas de cambio de los reguladores. Para estos puede ser muy difícil estimar parámetros. ¿Dónde se encuentran los datos para sistemas fuera de equilibrio?

**Modelo gráfico probabilístico** Estos sistemas modelan las redes como una distribución conjunta de probabilidad sobre variables aleatorias. Los bordes representan dependencias condicionales. Los modelos gráficos probabilísticos (PGM) están enfocados en la conferencia.

### Modelos gráficos probabilísticos

Los modelos gráficos probabilísticos (PGM) son entrenables y capaces de lidar con el ruido y por lo tanto son una buena técnica gráfica Bayesian Network Directed.<sup>3</sup> En las PGM, los nodos pueden ser factores de transcripción o genes y son modelados por variables aleatorias. Si conoce la distribución conjunta sobre estas variables aleatorias, puede construir la red como un PGM. Dado que esta estructura gráfica es una representación compacta de la red, podemos trabajar con ella fácilmente y realizar tareas de aprendizaje. Los ejemplos de PGMS incluyen:

**Red Bayesiana** Técnica gráfica dirigida. Cada nodo es padre o hijo. Los padres determinan completamente el estado de los hijos pero sus estados pueden no estar disponibles para el experimentador. La estructura de la red describe la distribución total de probabilidad conjunta de la red como producto de distribuciones individuales para los nodos. Al dividir la red en potenciales locales, la complejidad computacional se reduce drásticamente.

**Red Bayesiana Dinámica** Técnica gráfica dirigida. Las redes bayesianas estáticas no permiten dependencias cíclicas pero podemos intentar modelarlas con redes bayesianas permitiendo dependencias arbitrarias entre nodos en diferentes puntos de tiempo. Por lo tanto, se permiten dependencias cíclicas a medida que la red avanza a través del tiempo y la probabilidad conjunta de la red en sí misma puede describirse como una articulación en todo momento.

**Markov Campo Aleatorio** Técnica gráfica no dirigida. Modela potenciales en términos de camarillas. Permite modelar gráficos generales incluyendo cílicos con mayor orden que las dependencias por pares.

**Gráfico Factor** Técnica gráfica no dirigida. Las gráficas factoriales introducen nodos “factoriales” especificando potenciales de interacción a lo largo de los bordes. Los nodos factoriales también se pueden introducir para modelar potenciales de orden superior que por pares.

Es más fácil aprender redes para modelos bayesianos. Los campos aleatorios de Markov y los gráficos factoriales requieren la determinación de una función de partición complicada. Para codificar la estructura de la red, solo es necesario asignar variables aleatorias a TFs y genes y luego modelar la distribución de probabilidad conjunta.

Redes bayesianas proporcionan representaciones compactas de JPD

La principal fortaleza de las redes bayesianas proviene de la simplicidad de su descomposición en padres e hijos. Debido a que las redes están dirigidas, la distribución de probabilidad conjunta completa se descompone en un producto de distribuciones condicionales, una por cada nodo de la red.<sup>4</sup>

### Inferencia de red a partir de datos de expresión

Utilizando datos de expresión y conocimiento previo, el objetivo de la inferencia de la red es producir un gráfico de red. Las gráficas serán no dirigidas o dirigidas. Las redes regulatorias, por ejemplo, a menudo serán dirigidas mientras que las redes de expresión, por ejemplo, no estarán dirigidas.

---

<sup>2</sup> Los métodos supervisados no se abordarán hoy.

<sup>3</sup> Estos son los modelos de Dr. Roys de elección para tratar con redes biológicas.

This page titled [21.2: Inferencia de estructura](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [21.2: Structure Inference](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 21.3: Visión general de la tarea de aprendizaje PGM

Tenemos que aprender parámetros a partir de los datos que tenemos. Una vez que tenemos un conjunto de parámetros, tenemos que usar parametrizaciones para aprender estructura. Nos enfocaremos en enfoques basados en puntaje para la construcción de redes, definiendo una puntuación para ser optimizada como métrica para la construcción de redes.

### Aprendizaje de parámetros para redes bayesianas

**Máxima verosimilitud** Elige parámetros para maximizar la probabilidad de los datos disponibles dado el modelo.

En máxima verosimilitud, calcule la verosimilitud de los datos como puntuaciones de cada variable aleatoria dados los padres y tenga en cuenta que las puntuaciones se dependen de la elección de un modelo, las puntuaciones se maximizarán de diferentes maneras. Para la distribución gaussiana es posible simplemente calcular parámetros optimizando la puntuación. Para elecciones de modelos más complicadas puede ser necesario hacer descenso de gradiente.

**Estimación de parámetros bayesianos** Se trata de *theta* sí misma como una variable aleatoria y elige los parámetros maximizando la probabilidad posterior. Estos métodos requieren una estructura fija y buscan elegir parámetros internos maximizando la puntuación.

### Aprendizaje de la estructura

Podemos calcular mejor las parametrizaciones de conjeturas de redes estructuradas. ¿Cómo encontramos las estructuras por sí mismas?

El aprendizaje de la estructura procede comparando la probabilidad de parametrizaciones de ML a través de diferentes estructuras gráficas y con el fin de buscar aquellas estructuras que obtengan una puntuación óptima de ML.

Un marco bayesiano puede incorporar probabilidades previas sobre las estructuras gráficas si se le da alguna razón para creer a priori que algunas estructuras son más probables que otras.

Para realizar la búsqueda en el aprendizaje de estructuras, inevitablemente tendremos que usar un enfoque codicioso porque el espacio de estructuras es demasiado grande para enumerarlo. Dichos métodos procederán mediante una búsqueda incremental análoga a la optimización de descenso de gradiente para encontrar parametrizaciones ML.

Se considera y evalúa un conjunto de gráficas de acuerdo con la puntuación ML. Dado que el optima local puede existir, es bueno sembrar búsquedas gráficas desde múltiples puntos de partida.

Además de ser incapaces de capturar dependencias cíclicas como se mencionó anteriormente, las redes bayesianas tienen ciertas otras limitaciones.

**Enlaces indirectos** Dado que las redes bayesianas simplemente miran las dependencias estadísticas entre nodos, es fácil que se les engañe para que pongan bordes donde de hecho solo están presentes las relaciones indirectas.

**Interacciones descuidadas** Especialmente cuando las puntuaciones estructurales se optimizan localmente, es posible que se pierdan por completo interacciones biológicas significativas. Los genes coexpresados pueden no compartir reguladores adecuados.

Los métodos bayesianos de **velocidad lenta** discutidos hasta ahora son demasiado lentos para trabajar eficazmente los datos del genoma completo.

### Excluyendo Enlaces Indirectos

**¿Cómo eliminar los enlaces indirectos?** Los enfoques teóricos de la información se pueden utilizar para eliminar enlaces extraños mediante la poda de estructuras de red para eliminar información redundante. Se describen dos métodos.

**ARACNE** Por cada triplete de bordes, se calcula una puntuación de información mutua y el algoritmo ARACNE excluye los bordes con la menor información sujeta a ciertos umbrales por encima de los cuales se mantienen los bordes mínimos.

**MRNET** Maximiza la dependencia entre reguladores y objetivos al tiempo que minimiza la cantidad de información redundante compartida entre reguladores al eliminar bordes correspondientes a reguladores con baja varianza.

Como alternativa, es posible simplemente mirar motivos reguladores y eliminar bordes de regulación no predichos por motivos comunes.

## Programas Regulatorios de Aprendizaje para Módulos

**¿Cómo corregir omisiones para genes corregulados?** Al aprender parámetros para modelos reguladores en lugar de genes individuales, es posible explotar la tendencia de los genes coexpresados a regularse de manera similar. Similar al método de usar motivos regulatorios para podar bordes redundantes, al modelar módulos a la vez, reducimos los recuentos de borde de red al tiempo que aumentamos el volumen de datos para trabajar.

Con extensiones, también es posible modelar dependencias cíclicas. Las redes de módulos permiten la revisitación de agrupamiento donde los genes se reasignan a clústeres en función de qué tan bien son pronosticados por un programa regulatorio para un módulo.

Sin embargo, los módulos no pueden acomodar la membresía del módulo para compartir genes. Dividir y conquistar para acelerar el aprendizaje

**¿Cómo acelerar el aprendizaje?** El Dr. Roy ha desarrollado un método para dividir el gran problema de aprendizaje en tareas más pequeñas utilizando una técnica de dividir y conquistar para gráficos no dirigidos. Al comenzar con clústeres es posible inferir redes reguladoras para grupos individuales y luego cruzar bordes, reasignar genes e iterar.

## Conclusiones en la Inferencia de Red

Las redes regulatorias son importantes pero difíciles de construir en general. Al explotar la modularidad, a menudo es posible encontrar estructuras confiables para gráficas y subgrafías.<sup>5</sup>

Muchas extensiones están en el horizonte para las redes regulatorias. Estos incluyen inferir bordes causales a partir de correlaciones de expresión, aprender a compartir genes entre clústeres y otros.

---

<sup>4</sup> Las redes bayesianas se parametrizan  $\theta$  según nuestra elección específica de modelo de red. Con diferentes opciones de variables aleatorias, tendremos diferentes opciones para parametrizaciones,  $\theta$  y por lo tanto diferentes tareas de aprendizaje:

Las variables aleatorias **discretas** sugieren simples  $\theta$  correspondientes a elecciones de parámetros para una distribución multinomial.

Las variables aleatorias **continuas** pueden modelarse con  $\theta$  correspondientes a medias y covarianzas de gaussianos u otra distribución continua.

<sup>5</sup> El Dr. Roy señala que hay muchos algoritmos disponibles para ejecutar inferencia de red de módulos con diversas distribuciones. Hay disponibles paquetes de redes neuronales y paquetes bayesianos entre otros.

---

This page titled [21.3: Visión general de la tarea de aprendizaje PGM](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [21.3: Overview of the PGM Learning Task](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source:  
<https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 21.4: Aplicación de Redes

Utilizando árboles de regresión lineal y regresión, intentaremos predecir la expresión a partir de redes. Mediante la clasificación colectiva y el etiquetado de relajación, intentaremos asignar función a elementos desconocidos de la red.

Nos gustaría utilizar redes para:

1. predecir la expresión de genes a partir de reguladores.

En la predicción de la expresión, el objetivo es parametrizar una relación dando niveles de expresión génica a partir de los niveles de expresión del regulador. Se puede resolver de diversas maneras incluyendo regresión y se relaciona con el problema de encontrar redes funcionales.

2. predecir funciones para genes desconocidos.

### Descripción general de los modelos funcionales

Un modelo de predicción es un gaussiano condicional: un modelo simple entrenado por regresión lineal. Un modelo de predicción más complejo es un árbol de regresión entrenado por regresión no lineal.

#### Modelos gaussianos condicionales

Los modelos gaussianos condicionales predicen sobre un espacio continuo y son entrenados por una regresión lineal simple para maximizar la probabilidad de datos. Predicen dianas cuyos niveles de expresión son medios de gaussianos sobre reguladores.

El aprendizaje gaussiano condicional toma una red estructurada y dirigida con objetivos y factores de transcripción reguladores. Puede estimar los parámetros gaussianos,  $\mu$ , a partir de los datos encontrando parámetros que maximizan la probabilidad; después de una derivada, el enfoque ML se reduce a resolver una ecuación lineal.

A partir de una red reguladora funcional derivada de múltiples fuentes de datos<sup>6</sup>, el Dr. Roy entrenó un modelo gaussiano para la predicción utilizando datos de expresión de curso de tiempo y lo probó en un conjunto de pruebas de suspensión. En comparaciones con predicciones por un modelo entrenado a partir de una red aleatoria, se encontró que la red predijo sustancialmente mejor que al azar.

El modelo lineal utilizado hace una fuerte suposición sobre la linealidad de la interacción. Probablemente esta no sea una suposición muy precisa pero parece funcionar hasta cierto punto con el conjunto de datos probado.

#### Modelos de árbol de regresión

Los modelos de árbol de regresión permiten al modelador utilizar una distribución multimodal incorporando dependencias no lineales entre el regulador y la expresión del gen diana. La estructura final de un árbol de regresión describe la gramática de expresión en términos de una serie de elecciones realizadas en los nodos del árbol de regresión. Debido a que los objetivos pueden compartir programas regulatorios, se pueden incorporar nociones de motivos recurrentes. Los árboles de regresión son modelos ricos pero difíciles de aprender. Árboles de regresión en la predicción de expresión

En la práctica, la predicción se abre paso por un árbol de regresión dados los niveles de expresión del regulador. Al llegar a los nodos foliares del árbol de regresión, se realiza una predicción para la expresión génica.

#### Predicción funcional para nodos no anotados

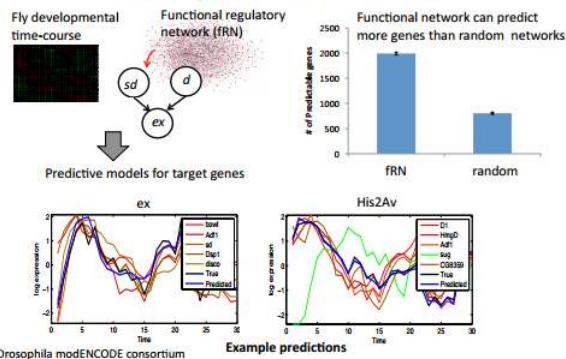
Dada una red con un conjunto incompleto de etiquetas, el objetivo de la anotación de funciones es predecir etiquetas para genes desconocidos. Utilizaremos métodos que caen dentro de la amplia categoría de culpabilidad por asociación. Si no sabemos nada de un nodo sino que sus vecinos están involucrados en una función, asigne esa función al nodo desconocido.

La asociación puede incluir cualquier noción de relación de red discutida anteriormente, tal como coexpresión, interacciones proteína-proteína y corregulación. Muchos métodos funcionan, dos serán discutidos: clasificación colectiva y clasificación de relajación; ambos funcionan para redes regulatorias codificadas como gráficas no dirigidas.

#### Clasificación Colectiva

Ver la predicción funcional como un problema de clasificación: Dado un nodo, ¿cuál es su clase reguladora?.

## Predicting expression of targets in the fly developmental time course



a) Desarrollo de moscas.

fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Para utilizar la estructura gráfica en el problema de predicción, capturamos propiedades de la vecindad de un gen en atributo relacional. Dado que todos los puntos están conectados en una red, los puntos de datos ya no están distribuidos inde- penalmente - el problema de predicción se vuelve sustancialmente más difícil que un problema de clasificación estándar.

La clasificación iterativa es un método sencillo con el que resolver el problema de clasificación. Comenzando con una suposición inicial para genes no etiquetados, infiere etiquetas iterativamente, permitiendo que las etiquetas cambiadas influyan en las predicciones de etiquetas de nodo de una manera similar al muestreo de gibbs<sup>7</sup>

El etiquetado de relajación es otro enfoque originalmente desarrollado para trazar redes terroristas. El modelo utiliza un puntaje de sospecha donde los nodos son etiquetados con una desconfianza de acuerdo con la desconfianza de sus vecinos. El método se llama etiquetado de relajación porque gradualmente se asienta en una solución de acuerdo con un parámetro de aprendizaje. Es otra instancia de aprendizaje iterativo donde a los genes se les asignan probabilidades de tener una función dada.

### Redes reglamentarias para la predicción de funciones

Para pares de nodos, calcule una similitud regulatoria —la cantidad de interacción— igual al tamaño de la intersección de sus reguladores dividido por el tamaño de su unión. Al tener esta similitud de interacción en forma de gráfica no dirigida sobre objetivos de red, se pueden utilizar clústeres derivados de una red en la clasificación funcional final.

El modelo tiene éxito en la predicción del desarrollo del disco invaginal y del sistema neural. La línea azul de la Fig. 21.2a muestra la puntuación de cada gen que predice su participación en el desarrollo del sistema neural.

La coexpresión y corregulación se puede utilizar lado a lado para aumentar el conjunto de genes conocidos por participar en el desarrollo del sistema neural.

<sup>6</sup> fuentes de datos incluyeron cromatina, unión física, expresión, motivo

<sup>7</sup> ver la conferencia anterior de Manolis describiendo el descubrimiento de motivos

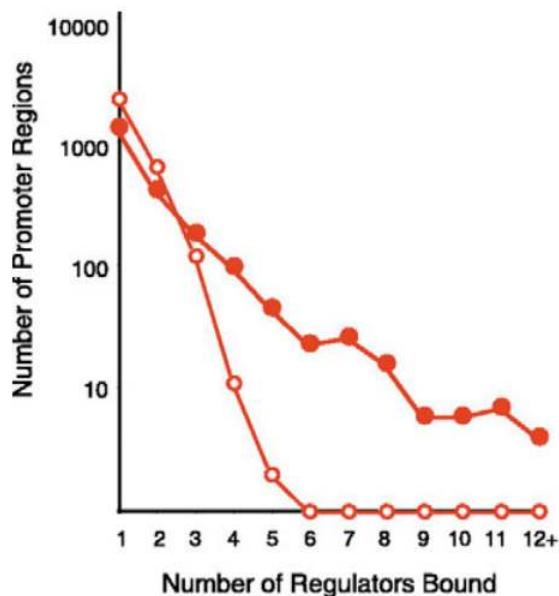
This page titled [21.4: Aplicación de Redes](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [21.4: Application of Networks](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 21.5: Propiedades Estructurales de Redes

Gran parte del trabajo temprano en redes fue realizado por científicos fuera de la biología. Los físicos miraron internet y redes sociales y describieron sus propiedades. Los biólogos observaron que las mismas propiedades también estaban presentes en las redes biológicas y nació el campo de las redes biológicas. En esta sección analizamos algunas de estas propiedades estructurales compartidas por las diferentes redes biológicas, así como las redes que surgen en otras disciplinas también.

### Distribución de grados



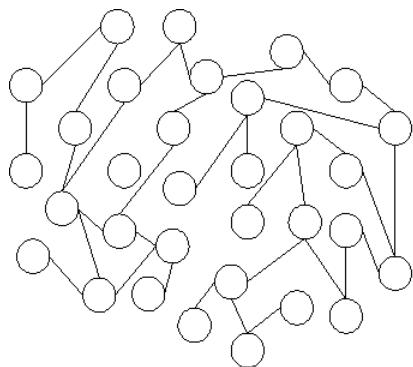
fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte [ocw.mit.edu/help/faq-fair-use/](http://ocw.mit.edu/help/faq-fair-use/).

Figura 21.3: Los símbolos sólidos dan la distribución en grado de genes en la red reguladora de *S. cerevisiae* (el grado de entrada de un gen es el número de factores de transcripción que se unen al promotor de este gen). Los símbolos abiertos dan la distribución en grados en la red aleatoria comparable. Figura tomada de [4].

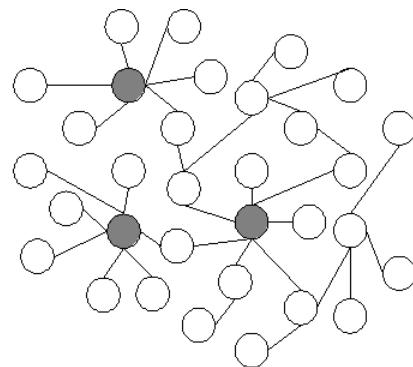
En una red, el grado de un nodo es el número de vecinos que tiene, es decir, el número de nodos a los que está conectado por un borde. La distribución de grados de la red da el número de nodos que tienen grado  $d$  por cada valor posible de  $d = 1, 2, 3, \dots$ . Por ejemplo, la figura 21.3 da la distribución de grados de la red reguladora del gen de *S. cerevisiae*. Se observó que la distribución de grados de las redes biológicas sigue una ley de potencia, es decir, el número de nodos en la red que tienen grado  $d$  es aproximadamente  $cd$  donde  $c$  es una constante de normalización y  $d$  es un coeficiente positivo. En tales redes, la mayoría de los nodos tienen un pequeño número de conexiones, excepto algunos nodos que tienen una conectividad muy alta.

Esta propiedad —de la distribución del grado de derecho de poder— se observó en realidad en muchas redes diferentes a través de diferentes disciplinas (por ejemplo, redes sociales, la World Wide Web, etc.) e indica que esas redes no son “aleatorias”: de hecho, las redes aleatorias (construidas a partir del modelo Erdős-Renyi) tienen un grado que sigue una distribución de Poisson donde casi todos los nodos tienen aproximadamente el mismo grado y los nodos con grado mayor o menor son muy raros [6] (ver figura 21.4).

Las redes que siguen una distribución de grado de derecho de poder se conocen como **redes libres de escala**. Los pocos nodos en una red libre de escala que tienen un grado muy grande se llaman *hubs* y tienen interpretaciones muy importantes. Por ejemplo, en las redes reguladoras de genes, los hubs representan factores de transcripción que regulan un número muy grande de genes. Las redes libres de escala tienen la propiedad de ser altamente resilientes a fallas de nodos “aleatorios”, sin embargo son muy vulnerables a fallas coordinadas (es decir, la red falla si falla uno de los nodos hub, consulte [1] para más información).



(a) Random network

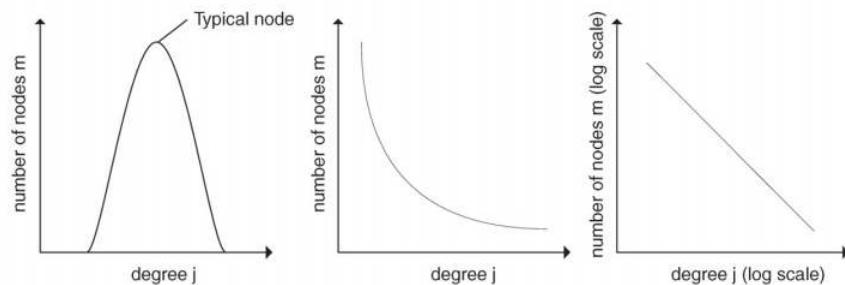


(b) Scale-free network

Carlos Castillo. Algunos derechos reservados. Licencia: CC BY-SA. Este contenido está excluido de nuestra Licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

a) Gráfica libre de escalas vs. una gráfica aleatoria (figura tomada de [10]).

a) Gráfica libre de escalas vs. una gráfica aleatoria (figura tomada de [10]).



b) Distribución de grados de la red libre de escala frente a la red aleatoria (cifra tomada de [3]).

Vieweg Verlag. Todos los derechos reservados. Este contenido está excluido de nuestro Creativo Licencia Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Hein, Oliver, et al. "Redes sin escala". *Wirtschaftsinformatik* 48, núm. 4 (2006): 267-75.

Figura 21.4: Redes Erd OS-Renyi sin escala frente a aleatorias

En una red regulatoria, se pueden identificar cuatro niveles de nodos:

1. Influyentes nodos reguladores maestros en la parte superior. Se trata de hubs que cada uno controla indirectamente muchos objetivos.
2. Reguladores de cuello de botella. Los nodos en el medio son importantes porque tienen un número máximo de objetivos directos.
3. Los reguladores en la parte inferior tienden a tener menos objetivos pero, sin embargo, ¡a menudo son biológicamente esenciales!
4. Objetivos.

## Motivos de red

Los motivos de red son subgrafías de la red que ocurren significativamente más que al azar. Algunos tendrán interesantes propiedades funcionales y presumiblemente son de interés biológico.

La Figura 21.5 muestra motivos reguladores de la red reguladora de levaduras. Los bucles de retroalimentación permiten el control de los niveles del regulador y los bucles de avance permiten la aceleración de los tiempos de respuesta entre otras cosas

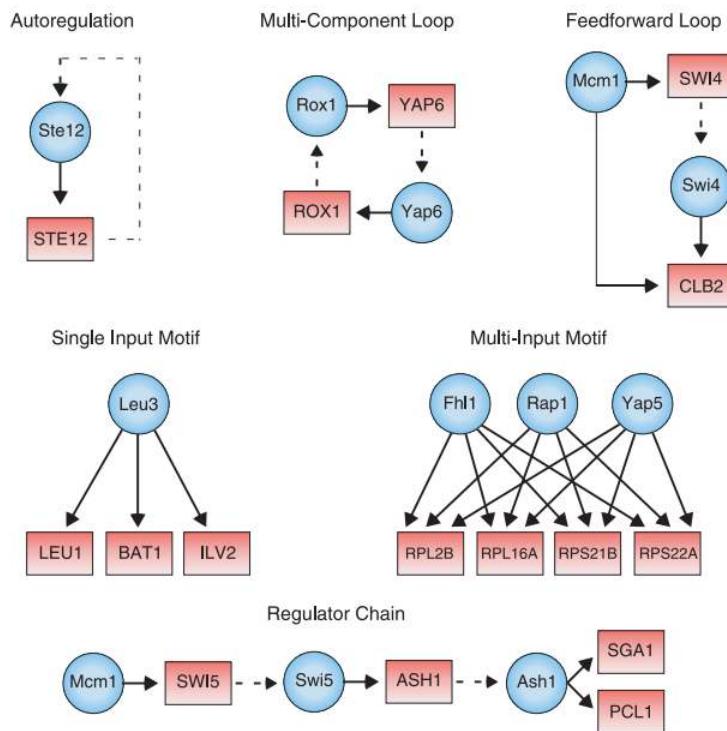


Figura 21.5: Motivos de red en redes reguladoras: Búcles de alimentación directa involucrados en la aceleración de la respuesta del gen diana. Los reguladores están representados por círculos azules y los promotores génicos están representados por rectángulos rojos (figura tomada de [4])

This page titled [21.5: Propiedades Estructurales de Redes](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **21.5: Structural Properties of Networks** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 21.6: Clustering de redes, Bibliografía

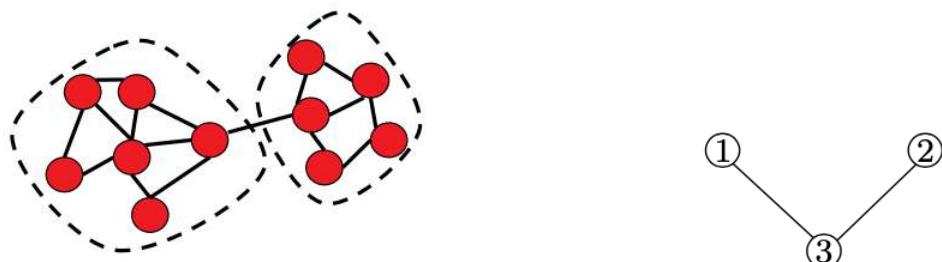
Un problema importante en el análisis de redes es poder **agrupar** o **modular la red** para identificar subgrafos que están densamente conectados (ver por ejemplo, la figura 21.6a). En el contexto de las redes de interacción génica, estos conglomerados podrían corresponder a genes que están involucrados en funciones similares y que están co-regulados.

Existen varios algoritmos conocidos para lograr esta tarea. Estos algoritmos suelen denominarse *algoritmos de particionamiento gráfico* ya que partitionan el gráfico en módulos separados. Algunos de los algoritmos conocidos incluyen:

- Algoritmo de agrupamiento de Markov [5]: El Algoritmo de Clustering de Markov (MCL) funciona haciendo una caminata aleatoria en la gráfica y observando la distribución en estado estacionario de esta caminata. Esta distribución en estado estacionario permite agrupar la gráfica en subgrafías densamente conectadas.
- Algoritmo Girvan-Newman [2]: El algoritmo Girvan-Newman utiliza el número de rutas más cortas que pasan por un nodo para calcular la *esencialidad* de un borde que luego se puede usar para agrupar la red.
- Algoritmo de partición espectral

En esta sección veremos en detalle el algoritmo de particionamiento espectral. Referimos al lector a las referencias [2, 5] para una descripción de los otros algoritmos.

El algoritmo de partición espectral se basa en una cierta forma de representar una red usando una matriz. Antes de presentar el algoritmo revisaremos así cómo representar una red usando una matriz, y cómo extraer información sobre la red usando operaciones matriciales.



fuente desconocida. Todos los derechos reservados. Este contenido es

excluido de nuestra licencia Creative Commons. Para más  
información, véase <http://ocw.mit.edu/help/faq-fair-use/>.

a) Una partición de una red en dos grupos.

b) Una red simple en 3 nodos. La  
adyacencia  
matriz de esta gráfica se da en la ecuación  
(21.1).

Figura 21.6

### Vista algebraica a las redes

**Matriz de adyacencia** Una forma de representar una red es usar la llamada matriz de adyacencia. La matriz de adyacencia de una red con  $n$  nodos es una matriz  $\times N \times N$  donde  $A_{i,j}$  es igual a uno si hay un borde entre los nodos  $i$  y  $j$ , y 0 en caso contrario. Por ejemplo, la matriz de adyacencia del gráfico representado en la figura 21.6b viene dada por:

```
\[A=\left[\begin{array}{lll} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{array}\right]\nonumber]
```

Si la red está ponderada (es decir, si los bordes de la red tienen cada uno un peso asociado), la definición de la matriz de adyacencia se modifica para que  $A_{i,j}$  mantenga el peso del borde entre  $i$  y  $j$  si el borde existe, y cero en caso contrario.

**Matriz laplaciana** Para el algoritmo de clustering que presentaremos más adelante en esta sección, necesitaremos contar el número de bordes entre los dos grupos diferentes en una partición de la red. Por ejemplo, en la Figura 21.6a, el número de aristas entre los

dos grupos es de 1. La *matriz laplaciana* que vamos a introducir ahora es útil para representar esta cantidad algebraicamente. La matriz laplaciana L de una red en n nodos es una matriz  $n \times n$  L que es muy similar a la matriz de adyacencia A excepto por los cambios de signo y para los elementos diagonales. Mientras que los elementos diagonales de la matriz de adyacencia son siempre iguales a cero (ya que no tenemos auto-bucle), los elementos diagonales de la matriz Laplaciana mantienen el *grado* de cada nodo (donde el grado de un nodo se define como el número de aristas que inciden en él). También los elementos fuera de la diagonal de la matriz laplaciana se establecen en 1 en presencia de un borde, y cero en caso contrario. En otras palabras, tenemos:

```
\[L_{i,j} = \begin{cases} \text{degree}(i) & \text{if } i=j \\ -1 & \text{if } i \neq j \text{ and there is an edge between } i \text{ and } j \\ 0 & \text{if } i \neq j \text{ and there is no edge between } i \text{ and } j \end{cases}\]
```

Por ejemplo, la matriz laplaciana de la gráfica de la figura 21.6b viene dada por (enfatizamos los elementos diagonales en negrita):

```
\[L = \begin{array}{ccc} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{array}\]
```

**Algunas propiedades de la matriz laplaciana** La matriz laplaciana de cualquier red disfruta de algunas propiedades agradables que serán importantes más adelante cuando veamos el algoritmo de clustering. Revisamos brevemente estos aquí.

La matriz laplaciana L es siempre **simétrica**, es decir,  $L_{i,j} = L_{j,i}$  para cualquier  $i, j$ . Una consecuencia importante de esta observación es que todos los valores propios de L son reales (es decir, no tienen parte imaginaria compleja). De hecho incluso se puede demostrar que los valores propios de L son todos no negativos<sup>8</sup> La propiedad final que mencionamos sobre L es que todas las filas y columnas de L suman a cero (esto es fácil de verificar usando la definición de L). Esto significa que el valor propio más pequeño de L es siempre igual a cero, y el vector propio correspondiente es  $s = (1, 1, \dots, 1)$ .

**Contando el número de aristas entre grupos usando la matriz Laplaciana** Usando la matriz laplaciana ahora podemos contar fácilmente el número de aristas que separan dos partes disjuntas de la gráfica usando operaciones simples de matriz. En efecto, supongamos que particionamos nuestra gráfica en dos grupos, y que definimos un vector s de tamaño n que nos dice a qué grupo pertenece cada nodo i:

```
\[s_i = \begin{cases} 1 & \text{if node } i \text{ is in group 1} \\ -1 & \text{if node } i \text{ is in group 2} \end{cases}\]
```

Entonces se puede demostrar fácilmente que el número total de aristas entre el grupo 1 y el grupo 2 viene dado por la cantidad  $\frac{1}{4}s^T L s$  donde L es el laplaciano de la red.

Para ver por qué este es el caso, primero calculemos el producto matriz-vector Ls. En particular, fijemos un nodo que digo en el grupo 1 (es decir,  $s_i = +1$ ) y veamos el componente iésimo del producto matriz-vector Ls. Por definición del producto matriz-vector tenemos:

$$(Ls)_i = \sum_{j=1}^n L_{i,j} s_j$$

Podemos descomponer esta suma en tres summands de la siguiente manera:

$$(Ls)_i = \sum_{j=1}^n L_{i,j} s_j = L_{i,i} s_i + \sum_{j \text{ in group 1}} L_{i,j} s_j + \sum_{j \text{ in group 2}} L_{i,j} s_j$$

Usando la definición de la matriz Laplaciana vemos fácilmente que el primer término corresponde al grado de  $i$ , es decir, el número de aristas incidentes a  $i$ ; el segundo término es igual al negativo del número de aristas que conectan  $i$  a algún otro nodo del grupo 1,

y el tercer término es igual al número de aristas *que conectan i a algún nodo ingroup 2*. De ahí que tengamos:

$$(Ls)_i = \text{grado}(i) (\# \text{ bordes de } i \text{ al grupo 1}) + (\# \text{ bordes de } i \text{ al grupo 2})$$

Ahora como cualquier borde de *i* debe ir al grupo 1 o al grupo 2 tenemos:

$$\text{grado}(i) = (\# \text{ bordes de } i \text{ al grupo 1}) + (\# \text{ bordes de } i \text{ al grupo 2}).$$

Así combinando las dos ecuaciones anteriores obtenemos:

$$(Ls)_i = 2 (\# \text{ bordes de } i \text{ al grupo 2}).$$

Ahora para obtener el número total de aristas entre el grupo 1 y el grupo 2, simplemente sumamos la cantidad anterior sobre todos los nodos *i* del grupo 1:

Misplaced '#'

También podemos mirar nodos en el grupo 2 para calcular la misma cantidad y tenemos:

Misplaced '#'

Ahora promediando las dos ecuaciones anteriores obtenemos el resultado deseado:

```
\begin{aligned}
&\text{\# bordes entre el grupo 1 y el grupo 2} = \frac{1}{4} \sum_{i \in \text{en el grupo 1}} (Ls)_i - \frac{1}{4} \sum_{i \in \text{en el grupo 2}} (Ls)_i \\
&\begin{array}{|l}
\hline
= \frac{1}{4} \sum_{i \in \text{L s}} (Ls)_i \\
= \frac{1}{4} s^T L s
\end{array}
\end{aligned}
```

donde *sT* es el vector de fila obtenido al transponer el vector de columna *s*.

## El algoritmo de agrupamiento espectral

Ahora veremos cómo se puede utilizar la vista de álgebra lineal de las redes dada en la sección anterior para producir una “buena” partición de la gráfica. En cualquier buena partición de una gráfica, el número de aristas entre los dos grupos debe ser relativamente pequeño en comparación con el número de aristas dentro de cada grupo. Así, una forma de abordar el problema es buscar una partición para que el número de aristas entre los dos grupos sea mínimo. Utilizando las herramientas introducidas en la sección anterior, este problema es así equivalente a encontrar un vector  $s \in \{-1, +1\}^n$  tomando solo valores 1 o +1 tal que  $\frac{1}{4}s^T L s$  sea mínimo, donde *L* es la matriz Laplaciana de la gráfica. En otras palabras, queremos resolver el problema de minimización:

$$\underset{s \in \{-1, +1\}^n}{\text{minimize}} \frac{1}{4} s^T L s$$

Si  $s^*$  es la solución óptima, entonces la partición óptima es asignar el nodo *i* al grupo 1 si  $s_i = +1$  o bien al grupo 2.

Esta formulación parece tener sentido pero desafortunadamente hay una pequeña falla: la solución a este problema siempre terminará siendo  $s = (+1, \dots, +1)$  lo que corresponde a poner todos los nodos de la red en el grupo 1, ¡y ningún nodo en el grupo 2! El número de aristas entre el grupo 1 y el grupo 2 es entonces simplemente cero y, de hecho, ¡es mínimo!

Para obtener una partición significativa tenemos que considerar así particiones de la gráfica que no son triviales. Recordemos que la matriz laplaciana *L* es siempre simétrica, y así admite una descomposición propia:

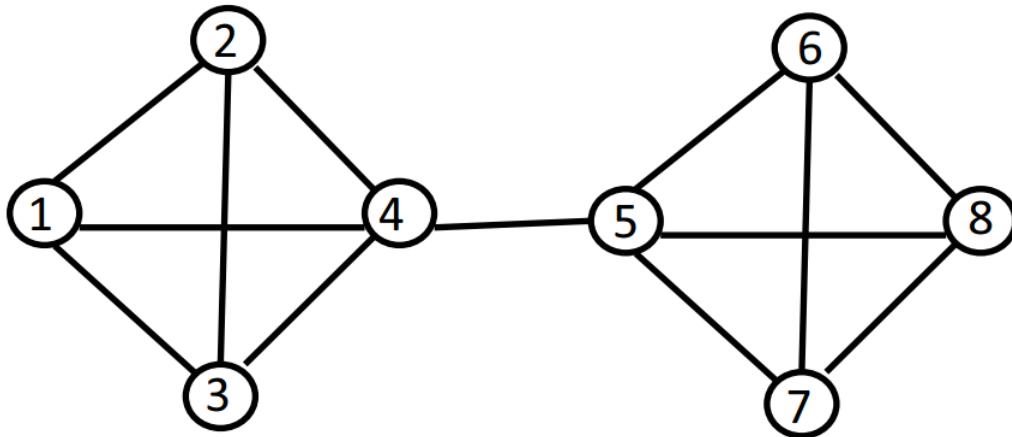
$$L = U \Sigma U^T = \sum_{i=1}^n \lambda_i u_i u_i^T$$

donde  $\sigma$  es una matriz diagonal que contiene los valores eigen no negativos  $\lambda_1, \dots, \lambda_n$  de *L* y *U* es la matriz de vectores propios y satisface  $U^T = U^{-1}$ .

El costo de una partición  $s \in \{-1, +1\}^n$  viene dado por

$$\frac{1}{4} s^T L s = \frac{1}{4} s^T U \Sigma U^T s = \frac{1}{4} \sum_{i=1}^n \lambda_i \alpha_i^2$$

donde  $\alpha = U^T s$  dan la descomposición de  $s$  como una combinación lineal de los vectores propios de  $L$ :  $s = \sum_{i=1}^n \alpha_i u_i$ .



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 21.7: Una red con 8 nodos

Recordemos también eso  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . Así, una forma de hacer que la cantidad anterior sea lo más pequeña posible (sin escoger la partición trivial) es concentrar todo el peso sobre el  $\lambda_2$  que se encuentra el valor propio más pequeño distinto de cero de  $L$ . Para lograr esto simplemente elegimos  $s$  de manera que  $\alpha_2 = 1$  y  $\alpha_k = 0$  para todos  $k \neq 2$ . En otras palabras, esto corresponde a tomar  $s$  para que sea igual a  $u_2$  el segundo vector propio de  $L$ . Dado que en general el autovector  $u_2$  no tiene valor entero (es decir, los componentes de  $u_2$  pueden ser diferentes a 1 o -1), tenemos que convertir primero el vector  $u_2$  en un vector de +1 o -1's. Una forma sencilla de hacer esto es solo mirar los signos de los componentes de  $u_2$  en lugar de los valores mismos. Nuestra partición está así dada por:

```
\[s=\operatorname{signo}\left(u_{\{2\}}\right)=\left(\begin{array}{ll}
1 & \text{if } \left(u_{\{2\}}\right)_i \geq 0 \\
-1 & \text{if } \left(u_{\{2\}}\right)_i < 0
\end{array}\right)\]
```

Para recapitular, el algoritmo de agrupamiento espectral funciona de la siguiente manera:

#### Algoritmo de partición espectral

- Entrada: una red
- Salida: una partición de la red donde cada nodo está asignado ya sea al grupo 1 o al grupo 2 para que el número de aristas entre los dos grupos sea pequeño

1. Calcular la matriz Laplaciana  $L$  de la gráfica dada por:

```
\[L_{i,j}=\left(\begin{array}{ll}
\operatorname{degree}(i) & \text{if } i=j \\
-1 & \text{if } i \neq j \text{ y hay un borde entre } i \text{ y } j \\
0 & \text{if } i \neq j \text{ y no hay borde entre } i \text{ y } j
\end{array}\right)\]
```

2. Calcular el autovector  $u_2$  para el segundo valor propio más pequeño de  $L$ .

3. Salida de la siguiente partición: Asignar nodo  $i$  al grupo 1 if  $(u_2)_i \geq 0$ , de lo contrario asignar el nodo  $i$  al grupo 2.

A continuación damos un ejemplo donde aplicamos el algoritmo de clustering espectral a una red con 8 nodos.

**Ejemplo** Ilustramos aquí el algoritmo de particionamiento descrito anteriormente en una red simple de 8 nodos dado en la figura 21.7. La matriz de adyacencia y la matriz Laplaciana de esta gráfica se dan a continuación:

```
\[A=\left[\begin{array}{llllllll}
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
amp; 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
0 & 1 \ 0 & 0 & 0 & 1 & amp; 1 & 0 \\
\end{array}\right]\quad L=\left[\begin{array}{rrrrrrr}
3 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\
1 & 3 & 0 & 0 & -1 & 0 & 0 & 0 \\
& 0 \\
-1 & y & -1 & 4 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
amp; 3 & -1 \\
0 & 0 & 0 & 0 & -1 & 1 & -1 & 3 \\
\end{array}\right]\nonumber]
```

Usando el comando eig de Matlab podemos calcular la descomposición propia  $L = U\Sigma U^T$  de la matriz Laplaciana y obtenemos:

```
\[U=\left[\begin{array}{rrrrrrr}
0.3536 & -0.3825 & 0.2714 & -0.1628 & -0.7783 & 0.0495 & -0.0064 & -0.1426 \\
0.3536 & -0.3825 & 0.5580 & -0.1628 & 0.6066 & 0.0495 & -0.0064 & -0.1426 \\
0.3536 & -0.3825 & ; & -0.4495 & 0.6251 & 0.0930 & 0.0495 & -0.3231 & -0.1426 \\
0.3536 & -0.2470 & -0.3799 & -0.2995 & 0.0786 & -0.1485 & 0.3358 & 0.6626 \\
0.3536 & \mathbf{0.2470} & -0.3799 & -0.2995 & 0.0786 & -0.1485 & 0.3358 & -0.6626 \\
0.3536 & \mathbf{0.3825} & 0.3514 & 0.5572 & -0.0727 & -0.3466 & 0.3860 & 0.1426 \\
0.3536 & \mathbf{0.3825} & 0.0284 & -0.2577 & -0.0059 & -0.3466 & -0.7218 & 0.1426 \\
0.3536 & ; \mathbf{0.3825} & 0.0000 & 0.0000 & 0.0000 & 0.8416 & -0.0000 & 0.1426 \\
\end{array}\right]\nonumber]\\[1ex]
\[\Sigma=\left[\begin{array}{llllllll}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}\right]\nonumber]
```

Hemos resaltado en negrita el segundo valor propio más pequeño de  $L$  y el autovector asociado. Para agrupar la red observamos el signo de los componentes de este vector propio. Vemos que los primeros 4 componentes son negativos, y los últimos 4 componentes son positivos. Así, agruparemos los nodos 1 a 4 juntos en el mismo grupo, y los nodos 5 a 8 en otro grupo. Esto parece un buen agrupamiento y de hecho esta es la agrupación “natural” que se considera a primera vista de la gráfica.

### ¿Sabías?

El problema matemático que formulamos como motivación para el algoritmo de agrupamiento espectral es encontrar una partición de la gráfica en dos grupos con un número mínimo de bordes entre los dos grupos. El algoritmo de particionamiento espectral que presentamos no siempre da una solución óptima a este problema pero suele funcionar bien en la práctica.

En realidad resulta que el problema tal y como lo formulamos se puede resolver exactamente usando un algoritmo eficiente. El problema a veces se llama el problema de corte mínimo ya que estamos buscando cortar un número mínimo de bordes de la gráfica para desconectarla (los bordes que cortamos son los que se encuentran entre el grupo 1 y el grupo 2). El problema de corte mínimo se puede resolver en tiempo polinomial en general, y remitimos al lector a la entrada de Wikipedia sobre corte mínimo [9] para más información. El problema sin embargo con las particiones de corte mínimo es que generalmente conducen a particiones de la gráfica que no están equilibradas (por ejemplo, un grupo tiene solo 1 nodo, y los nodos restantes están todos en el otro grupo). En general, uno quisiera imponer restricciones adicionales a los clústeres (por ejemplo, límites inferiores o superiores en el tamaño de los clústeres, etc.) para obtener clústeres más realistas. Con tales limitaciones, el problema se vuelve más difícil, y remitimos al lector a la entrada de Wikipedia sobre Particionamiento gráfico [8] para más detalles.

### Preguntas frecuentes

P: ¿Cómo dividir la gráfica en más de dos grupos?

R: En esta sección solo nos fijamos en el problema de particionar la gráfica en dos clústeres. ¿Y si queremos agrupar la gráfica en más de dos clústeres? Hay varias extensiones posibles del algoritmo que se presentan aquí para manejar  $k$  clústeres en lugar de solo dos. La idea principal es mirar los  $k$  vectores propios para los  $k$  valores propios distintos de cero más pequeños del Laplaciano, y luego aplicar el algoritmo de clustering k-means apropiadamente. Refirimos al lector al tutorial [7] para más información.

<sup>8</sup> Una forma de ver esto es notar que  $L$  es diagonalmente dominante y los elementos diagonales son estrictamente positivos (para más detalles el lector puede buscar “diagonalmente dominante” y “teorema del círculo Gershgorin” en Internet).

This page titled [21.6: Clustering de redes, Bibliografía](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **21.6: Network clustering, Bibliography** by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## Bibliografía

- 
- [1] R. Albert. Redes libres de escala en biología celular. Revista de ciencia celular, 118 (21) :4947—4957, 2005.
  - [2] M. Girvan y M.E.J. Newman. Estructura comunitaria en redes sociales y biológicas. Actas de la Academia Nacional de Ciencias, 99 (12) :7821—7826, 2002.
  - [3] O. Hein, M. Schwind y W. K. onig. Redes libres de escala: El impacto de la distribución de grados de cola de grasa en los procesos de difusión y comunicación. Wirtschaftsinformatik, 48 (4) :267—275, 2006.
  - [4] T.I. Lee, N.J. Rinaldi, F. Robert, D.T. Odom, Z. Bar-Joseph, G.K. Gerber, N.M. Hannett, C.T. Harbi- hijo, C.M. Thompson, I. Simon, et al. Redes reguladoras transcripcionales en *saccharomyces cerevisiae*. Señalización científica, 298 (5594) :799, 2002.
  - [5] S.M. van Dongen. Agrupación gráfica por simulación de flujo. Tesis doctoral, Universidad de Utrecht, Las tierras abisales, 2000.
  - [6] M. Vidal, M.E. Cusick, y A.L. Barabasi. Redes interactómicas y enfermedades humanas. Cell, 144 (6) :986— 998, 2011.
  - [7] U. Von Luxburg. Un tutorial sobre agrupamiento espectral. Estadística y computación, 17 (4) :395—416, 2007.
  - [8] Wikipedia. Partición gráfica, 2012.
  - [9] Wikipedia. Corte mínimo, 2012.
  - [10] Wikipedia. Red sin escalas, 2012.
- 

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 22: Interacciones de cromatina

- 22.1: Introducción
- 22.2: Terminología relevante
- 22.3: Métodos moleculares para estudiar la organización del genoma nuclear
- 22.4: Mapeo de interacciones genoma-lámina nuclear (LADs)
- 22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear
- 22.6: Arquitectura de la Organización del Genoma
- 22.7: Comprensión mecanicista de la arquitectura del genoma
- 22.8: Direcciones actuales de investigación

---

This page titled [22: Interacciones de cromatina](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 22.1: Introducción

En los últimos años, se han descubierto muchos mecanismos sutiles y previamente ignorados para la regulación genética fina. Aparte de la regulación directa por proteínas, estos mecanismos incluyen la participación de regiones codificantes no proteicas del genoma, factores epigenómicos como modificaciones de histonas y diversos cambios de ARN. La organización espacial de la cromatina dentro del núcleo, los complejos modificadores de la cromatina y sus consecuencias funcionales también se han convertido en un área de interés. En este capítulo, profundizaremos en el estudio de las estructuras de cromatina 3D, comenzando por el estado del arte en este campo, la terminología más relevante y los métodos actuales. Especialmente nos centraremos en el estudio de las regiones de ADN localizadas por regiones periféricas del núcleo (así en estrecho contacto con la lámina nuclear). Finalmente, discutiremos los métodos computacionales involucrados en el estudio de la organización del genoma nuclear.

### Lo que ya se sabe

El ADN se compacta localmente en nucleosomas, envolviéndose alrededor de octámeros de histonas. Cada nucleosoma comprende aproximadamente 147 bps empaquetados en 1.67 vueltas superhelicoidales a la izquierda. El ADN se compacta globalmente como cromosomas (durante la división celular y la mitosis). Los cromosomas se han teñido con colores diferentes, y se ha demostrado que algunos cromosomas tienen preferencias radiales dentro del núcleo celular, incluso cuando la célula no está experimentando división activa y los cromosomas no están condensados. Es decir, algunos cromosomas prefieren permanecer cerca del centro del núcleo mientras que otros tienden hacia la periferia. Estos se conocen como territorios cromosómicos (CT). Los territorios de los cromosomas homólogos generalmente no se encuentran cerca unos de otros. También se sabe que existe una 'arquitectura' nuclear global que es observable, conservada incluso entre diferentes tipos de células.

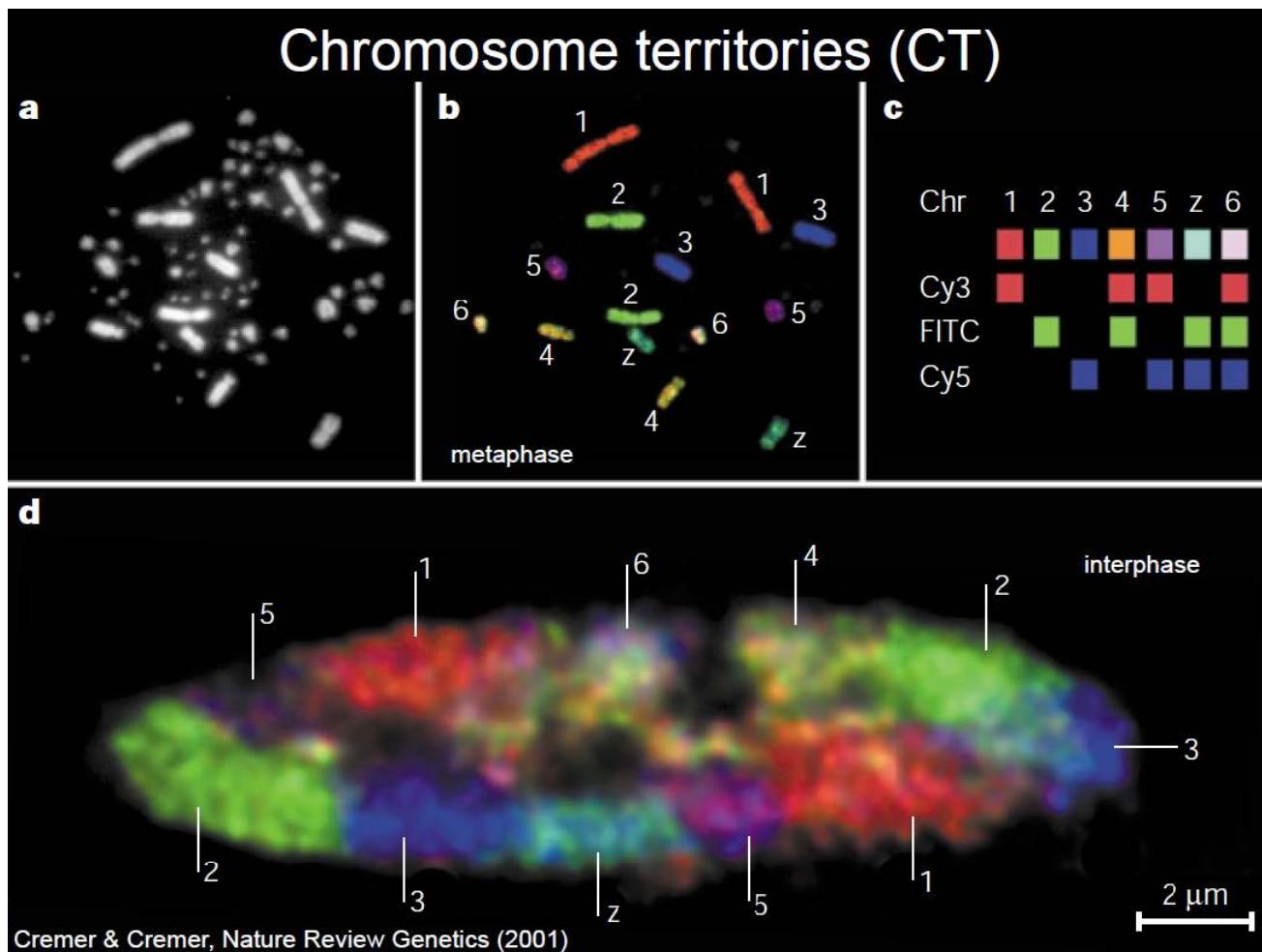


Figura 22.1: Territorios cromosómicos

Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Cremer, Thomas y Christoph Cremer. "Territorios cromosómicos, arquitectura nuclear y Regulación Génica en Células de Mamíferos". *Nature Opiniones Genéticas* 2, no.4 (2001): 292-301.

## Lo que no sabemos

Si bien la organización local (empaquetamiento de nucleosomas) y la organización global (condensación cromosómica) del ADN se entienden de alguna manera, las estructuras intermedias del ADN aún no están bien caracterizadas; muchos estados especulados solo se han observado *in vitro*. También se desconoce en gran medida el posicionamiento de las regiones genómicas en el núcleo a nivel subcromosómico, por ejemplo, la conformación 3D específica de una determinada región cromosómica que contiene varios genes.

Si bien se sabe que los cromosomas conservan cierta arquitectura general durante todo el ciclo celular, se desconoce cómo se mantiene esa ubicación y cómo los cromosomas diferentes continúan interactuando a lo largo de todo el ciclo celular.

Juntos, aunque sí entendemos ciertas partes de la función de los cromosomas, no tenemos una comprensión mecanicista completa de este proceso.

## ¿Por qué lo estudiamos?

En general, nos interesa comprender las características funcionales de las regiones genómicas y los mecanismos moleculares codificados en su interior, lo que podría tener implicaciones en enfermedades humanas. Particularmente, se ha demostrado que es probable que los genes que están codificados en regiones espacialmente vecinas estén corregulados. Además, el ADN empacado

dentro del núcleo es el equivalente a envolver 20 km de hilo de 20  $\mu\text{m}$  de espesor en algo del tamaño de una pelota de tenis, que alcanzaría de Kendall Square a Harvard y ¡retrocedería más de 6 veces y media! ¿No es esto increíble??

---

This page titled [22.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.2: Terminología relevante

### Dominios asociados a la lámina (LADs)

Los dominios asociados a la lámina (LADs) son las porciones de la cromatina que interactúan con la lámina nuclear. El mapeo de las interacciones de la cromatina y la lámina nuclear proporciona una visión hacia el mapeo del plegamiento cromático. Si bien no se sabe mucho sobre los LADs, se sabe que estas regiones están estrechamente relacionadas tanto con la alta expresión génica como con la baja densidad génica, una combinación interesante. Además, los LADs están asociados con CTCF, promotores e islas CPG a lo largo de sus fronteras.

### Histonas

Las histonas son proteínas altamente alcalinas que se encuentran eucariotas que comprenden el núcleo de los nucleosomas, empaquetando y ordenando el ADN nuclear. Una forma de octámero por 2 copias de las histonas centrales H2A, H2B, H3 y H4 forma el nucleosoma, que actúa como un carrete para que el ADN se enrolle alrededor.

### Cromatina

La cromatina es una forma compleja por ADN, proteínas y ARN que genera la arquitectura global del ADN en núcleos eucariotas. Sus principales funciones incluyen el empaquetamiento del ADN, el refuerzo de la macromolécula de ADN para permitir la mitosis, prevenir el daño del ADN y regular la expresión génica y la replicación del ADN. La mayoría de los mecanismos subyacentes a la formación y regulación de la estructura de la cromatina son en gran parte desconocidos; sin embargo, durante la división celular, la cromatina se organiza por medio de cromosomas.

### Territorios cromosómicos (TC)

Los cromosomas no se distribuyen aleatoriamente por todo el núcleo. Los cromosomas ocupan regiones específicas del núcleo. Estas regiones se llaman territorios cromosómicos.

### Principios de plegado bruto

This page titled [22.2: Terminología relevante](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.2: Relevant terminology](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.3: Métodos moleculares para estudiar la organización del genoma nuclear

Existen dos tipos principales de métodos para investigar la estructura tridimensional de la cromatina en el núcleo.

- El primer conjunto de métodos, ChIP y DaMid, son métodos que miden las interacciones ADN-'Landmark'. Es decir, miden interacciones de loci genómicos con hitos nucleares relativamente fijos, y solo se identificarán regiones del genoma que entren en contacto con la lámina nuclear.

El segundo conjunto de métodos, los métodos basados en 3C, son aquellos que miden las interacciones ADN-ADN. Pueden identificarse dos regiones cualesquiera de ADN que interactúen, independientemente de que estén cerca del interior o de la periferia del núcleo.

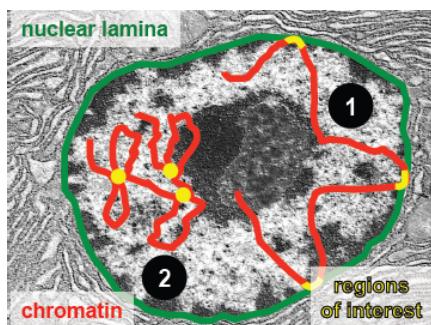


Figura 22.2:1) ChIP y DaMid solo identifican regiones que han entrado en contacto cercano con la lámina nuclear. 2) Los métodos basados en 3C identifican todas las interacciones ADN-ADN, independientemente de si están en la periferia del núcleo o no

### Métodos para medir las interacciones ADN-Lamina Nuclear

Los siguientes métodos, ChIP y DaMid, examinan regiones del ADN que específicamente entran en contacto con la lámina nuclear.

#### ChIP: Cromatina Immuno Precipitación

ChIP es un método para detectar regiones de ADN que están unidas a proteínas de interés. Las proteínas unidas con ADN se reticulan en su lugar con formaldehído. Los complejos proteína-ADN se extraen mediante cromatografía de anidad, principalmente utilizando anticuerpos específicos que se dirigen a la proteína de interés. Luego se desasocian los complejos recuperados, se rompen las reticulaciones y se fragmenta y analiza el ADN que se unió a las proteínas. Los fragmentos de ADN se pueden analizar usando secuenciación (ChIP-seq) o micromatrizes (Chip-chip). Sin embargo, un gran desafío asociado con las diversas técnicas de ChIP es que puede ser difícil obtener un anticuerpo de alta anidad. Para estudiar la estructura 3D del ADN dentro del núcleo, se puede usar Chip-seq con anticuerpos que encaran las proteínas de la lámina.

#### DAMID: Identificación de ADN adenina metiltransferasa

La DAMID se usa para mapear los sitios de unión de las proteínas de unión a cromatina. En el método DaMid, la ADN adenina metiltransferasa (Dam) de *E. coli* se fusiona a la proteína laminB1 (la enzima Dam cuelga del extremo de la proteína y, por lo tanto, está en las proximidades para interacciones). En *E. coli*, la enzima Dam metila la adenina en la secuencia GATC; los genomas bacterianos contienen proteínas con funciones como Dam para proteger su propio ADN de la digestión por enzimas de restricción, o como parte de sus sistemas de reparación de ADN. Como este proceso no ocurre naturalmente en eucariotas, las adeninas metiladas en una región pueden atribuirse así a una interacción con la proteína fusionada con Dam, lo que implica que esa región particular entró en contacto cercano con la lámina nuclear. Como control, la Presa sin fundir se puede expresar en niveles bajos. Esto da como resultado una distribución dispersa de adenina metilada para la cual la posición precisa de las adeninas metiladas se puede utilizar para inferir la variación en la accesibilidad del ADN. Las adeninas metiladas se determinan mediante ensayos de PCR de disulfuro u otra técnica de PCR sensible a las metilaciones en el ADN molde. En uno de esos ensayos, el genoma puede ser digerido por DpNi, que sólo corta secuencias GATC metiladas. Las secuencias adaptadoras se ligan entonces a los extremos de estas piezas digeridas, y la PCR se ejecuta usando cebadores que coinciden con los adaptadores. Solo se amplifican las regiones que ocurren entre las posiciones proximales del GATC. La medición final es el log de la relación entre la asociación de lámina de prueba y control: los valores positivos se asocian preferentemente a la lámina, y por lo tanto se identifican como LAD. Una ventaja

de usar DaMid sobre ChIP es que DaMid no requiere un anticuerpo específico que puede ser difícil de encontrar. Sin embargo, una desventaja del uso de DaMid es que la proteína de fusión debe elaborarse y expresarse.

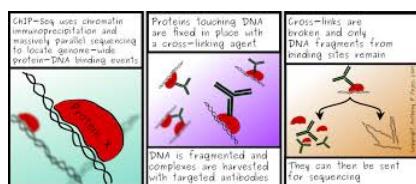


Figura 22.3: Método de medición ChIP

Cortesía de Anthony P. Fejes. Usado con permiso.

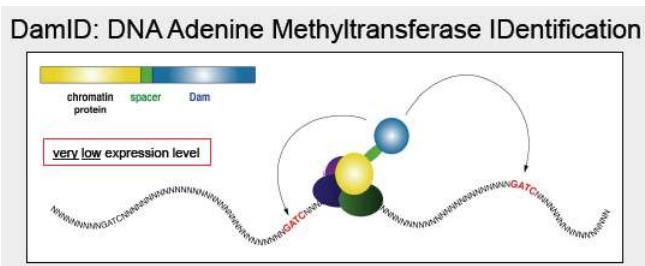


Figura 22.4: Método de medición de DAMID

Cortesía de Bas van Steense. Usado con permiso.

### FAQ

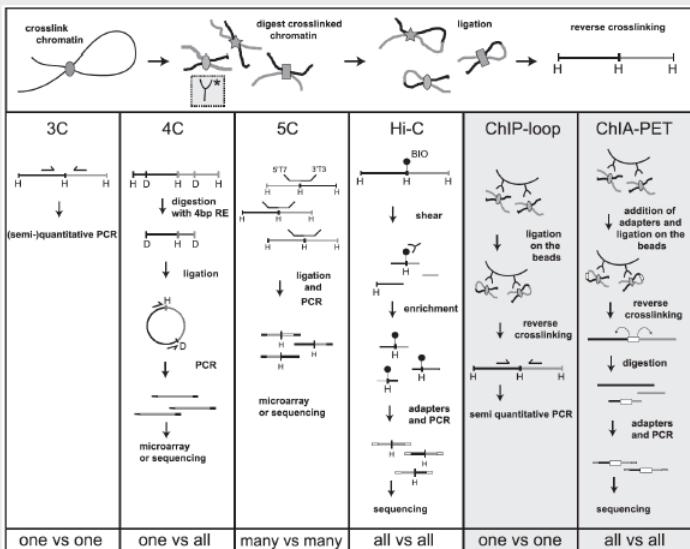
P: ¿Qué tan cerca tiene que llegar el ADN a la DAMID para ser metilada?

R: No tiene que unirse directamente a la lámina, pero sí tiene que acercarse bastante. El DAMID tiene un rango de aproximadamente 1.5kb.

### Medición de contactos ADN-ADN

Todos los siguientes métodos se basan en la Captura de Conformación Cromosómica (3C) con ciertas modificaciones.

## Chromosome Conformation Capturing (3C) based methods



de Wit & de Laat, Genes & Dev. (2012)

Figura 22.5: Métodos basados en 3C para identificar interacciones de cromatina

Prensa de Laboratorio Cold Spring Harbor. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: de Wit, Elzo y Wouter de Laat. "Una década de Tecnologías 3C: Insights into Nuclear Organization".

Genes & Desarrollo 26, núm. 1 (2012): 11-24.

### 3C

La captura de conformación cromosómica (3C) es un método que detecta qué loci genómicos están cerca de otros loci dentro del núcleo. Similar al método ChIP, se utiliza un agente de reticulación para congelar proteínas unidas al ADN en su lugar y formar complejos proteína-ADN. El ADN puede entonces ser digerido por una enzima de restricción después de permitir que la proteína unida se disocie. Por lo general, se usa una enzima con un sitio de reconocimiento de 6 bps de largo que deja extremos pegajosos, como HindIII. Los fragmentos generados son inducidos entonces a autoligarse (Usando concentraciones muy bajas de ADN para evitar la ligación del fragmento con otro fragmento aleatorio). El resultado es un conjunto de fragmentos lineales de ADN, conocidos como la biblioteca 3C, que pueden analizarse mediante PCR diseñando cebadores específicamente para la interacción de interés. 3C puede describirse como un método 'uno contra uno', ya que los cebadores utilizados son específicamente diana para amplificar el producto de la interacción entre 2 regiones de interés.

### Captura de conformación de cromatina circularizada (4C)

Los métodos 4C pueden describirse como 'uno contra todos' porque para una sola región de interés, podemos examinar todas sus interacciones con todas las demás regiones del genoma. 4C funciona de manera similar a 3C siendo la diferencia principal la enzima de restricción utilizada. En 4C, se emplea un cortador común para generar más y más fragmentos más pequeños. Estos fragmentos se ligan luego de nuevo. Algunos fragmentos más pequeños pueden ser excluidos, pero el resultado es un fragmento circularizado de ADN. Se pueden diseñar cebadores para amplificar el fragmento 'desconocido' de ADN de manera que se identifiquen todas las interacciones con la región de interés.

### Captura de conformación cromosómica con copia de carbono (5C)

5C es un método 'muchos vs muchos' y permite la identificación de interacciones entre muchas regiones de interés y muchas otras regiones, también de interés, para ser analizadas a la vez. 5C funciona de manera similar a 3C. Sin embargo, después de obtener la biblioteca 3C, se realiza la amplificación mediada por ligación múltiple (LMA). La LMA es un método en el que se amplifican múltiples dianas. La biblioteca 5C resultante puede analizarse en una micromatriz o secuenciación de alto rendimiento.

## Hi-C

Hi-C puede describirse como un método 'todo contra todos' porque identifica todas las interacciones de la cromatina. Hi-C funciona marcando todos los fragmentos de ADN con biotina antes de la ligadura, lo que marca todas las uniones de ligadura. Luego se usan perlas magnéticas para purificar las uniones marcadas con biotina. Esta biblioteca Hi-C puede entonces ser alimentada en la secuenciación de próxima generación.

## Chip-loop

Chip-loop se puede describir como un método 'uno contra uno', ya que similar al 3C, solo se puede identificar una interacción entre dos regiones de interés. Chip-loop es un híbrido entre ChIP y los métodos 3C. Los complejos ADN-proteína se reticulan primero y se digieren. Entonces, como en ChIP, la proteína de interés y el ADN unido a ella son arrastrados hacia abajo usando un anticuerpo. El protocolo luego procede como en 3C: los extremos libres de los fragmentos se ligan, la reticulación se invierte y la secuenciación puede proceder usando cebadores diseñados específicamente para una interacción 'uno contra uno'.

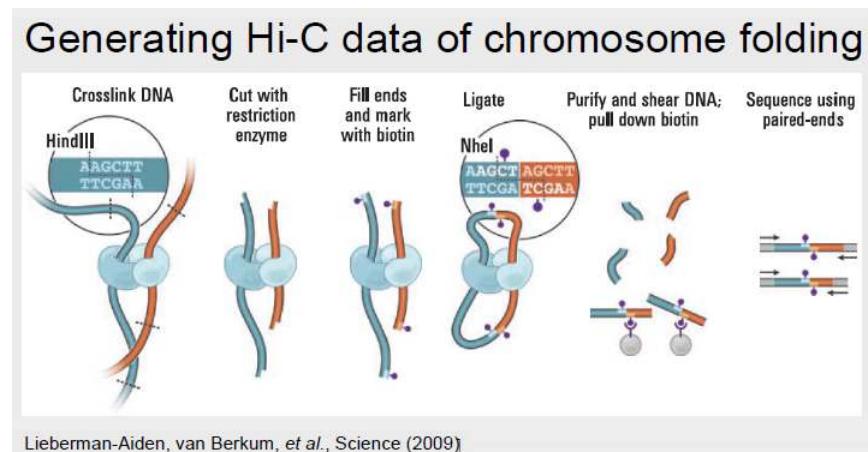
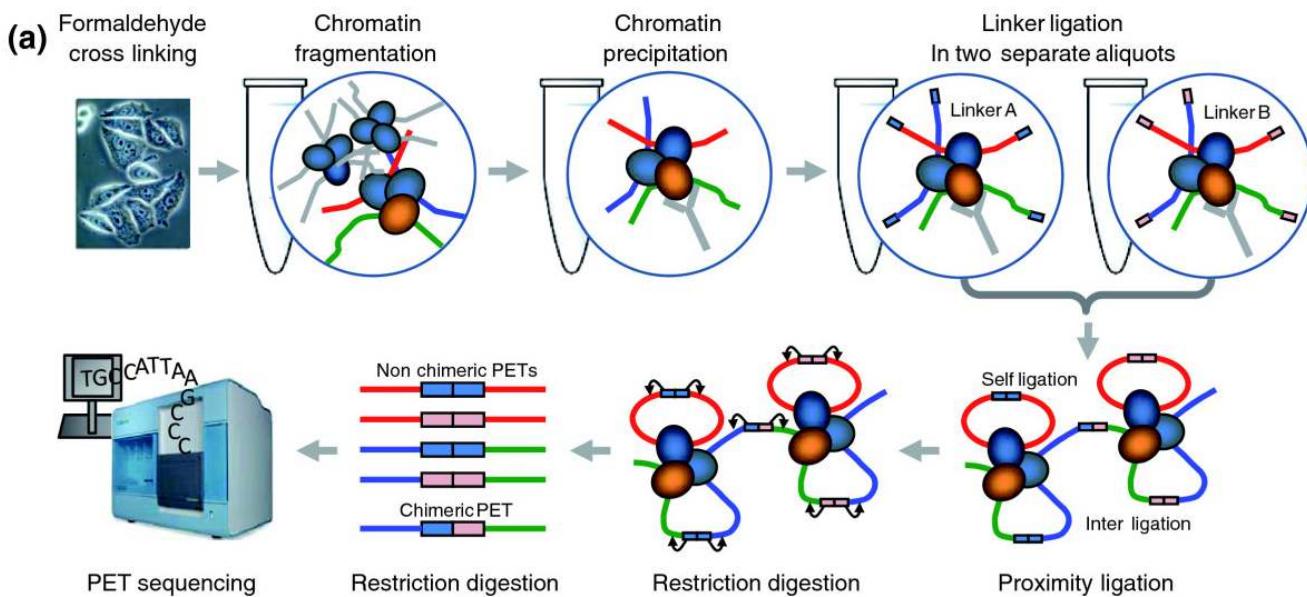


Figura 22.6: Método para generar datos Hi-C

## Chia-pet

Análisis de Intercepción de Cromatina por Secuenciación de Etiquetas de Extremo Pareado, o Chia-PET, combina las metodologías ChIP y 3C para determinar las interacciones de cromatina de largo alcance en todo el genoma. Se puede describir como un método 'todo vs todos', ya que aunque se debe identificar una sola proteína de interés, se identificará cualquier interacción. En Chia-PET, los complejos ADN-proteína están reticulados, como en los métodos previamente discutidos. Sin embargo, la sonicación se usa para romper la cromatina y reducir las interacciones no específicas. Al igual que en el protocolo ChIP, se usa un anticuerpo para tirar hacia abajo regiones de ADN unidas a una proteína de interés. Dos enzimas oligonucleótidosdiferentes se ligan entonces a los extremos libres del ADN. Ambos enzimas tienen sitios de corte MMeI. Los enzimas se ligan entonces entre sí para que los extremos libres se conecten, después de lo cual los fragmentos se digieren con MmeI. MmeI corta 20 nt aguas abajo de su secuencia de reconocimiento, por lo que el resultado de la digestión es el enzima bordeado por la secuencia de interés en ambos lados. Esta es una estructura de 'etiqueta-enlazador-etiqueta', y los fragmentos se conocen como PET. Las PET pueden secuenciarse y mapearse de nuevo al genoma para determinar regiones de ADN que interactúan.

Figure 1.



Li et al. Genome Biology 2010 11:R22 doi:10.1186/gb-2010-11-2-r22

Figura 22.7: Protocolo Chia-PET

This page titled [22.3: Métodos moleculares para estudiar la organización del genoma nuclear](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.3: Molecular Methods for Studying Nuclear Genome Organization](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.4: Mapeo de interacciones genoma-lámina nuclear (LADs)

En esta sección, presentaremos cómo se utilizaron los métodos DAMID y Hi-C para mapear dominios asociados a láminas en el genoma.

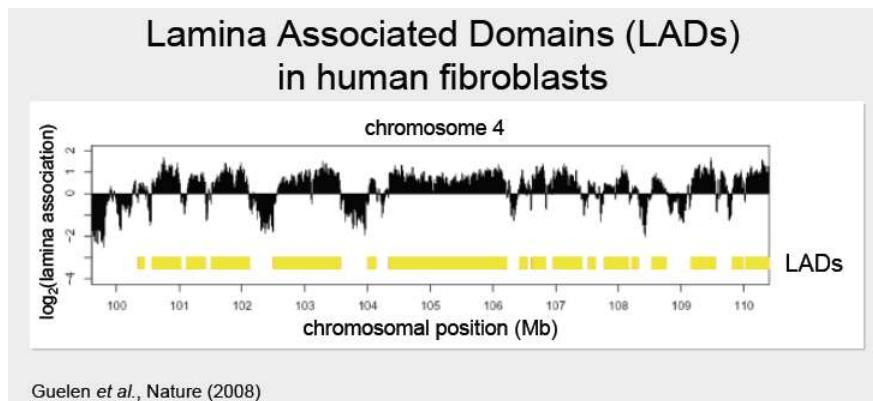
### Interpretación de datos de DAMID

Se utilizó el método DAMID (descrito en la sección 3) para identificar regiones de ADN que interactuaban con la proteína de lámina en la lámina nuclear.

#### ¿Sabías?

Los experimentos de DAMID normalmente se ejecutan durante 24 horas y la metilación es irreversible. Los resultados también son el promedio sobre millones de células. Por lo tanto, el DAMID no es adecuado para el posicionamiento del genoma relacionado con el tiempo exacto, aunque los estudios unicelulares pronto podrán hacer abordar este tema!

Los resultados del experimento de DAMID se graficaron como  $\log_2 \frac{\text{Dam fusionprotein}}{\text{Dam only}}$  proteína, tal como se hizo en la siguiente figura (picos negros). Para el experimento de fusión LaminB1, las regiones positivas (subrayadas en amarillo en la figura siguiente) indican regiones que se asocian preferentemente con la lámina nuclear. Estas regiones positivas son desmultadas como Dominios Asociados a Lámina, o LADs. Aproximadamente 1300 LAD fueron descubiertos en humanos.



Guelen et al., Nature (2008)

Figura 22.8: Dominios Asociados a Lámina (LADs)  
fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative  
Licencia Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

fibroblastos. Eran sorprendentemente grandes, oscilando entre aproximadamente 0.1Mb - 10Mb, con un tamaño mediano de 5Mb.

#### ¿Sabías?

Los experimentos de DAMID normalmente se ejecutan durante 24 horas y la metilación es irreversible. Los resultados también son el promedio sobre millones de células. Por lo tanto, el DAMID no es adecuado para el posicionamiento del genoma relacionado con el tiempo exacto, aunque los estudios unicelulares pronto podrán hacer abordar este tema!

#### FAQ

P: ¿En esta representación es significativo un valor de 0?

R: No, definitivamente hay un punto donde no sabemos dónde está el 0 real. En cambio, podemos intentar hacer una buena estimación de dónde debe estar el valor 0, para ver la preferencia relativa (el interior vs exterior del núcleo)

Después de identificar los LAD, podemos alinear sus límites para descubrir diversas características interesantes como densidades génicas conocidas o niveles de expresión génica a los datos para construir nuestro modelo de LAD. Los experimentos han demostrado que los LAD se caracterizan por bajos niveles de densidad génica y expresión génica. Se notó que los límites de la

LAD están muy definidos. Al alinear las posiciones de inicio de muchos LAD, se descubrió que los bordes están particularmente marcados por islas CpG, promotores que apuntan hacia afuera y sitios de unión a CTCF.

#### FAQ

P: ¿Por qué sitios de unión a CTCF? ¿Qué tienen de importante ellos?

R: ¡Esa es la pregunta! Quizás ayuden a mantener a los LADs. Quizás podrían impedir que los LADs se 'extender'.

#### FAQ

P: ¿Cómo se relaciona la organización con la expresión policlonal?

R: Ciertamente algo sucede; sin embargo, la represión policlonal funciona a menor escala que la LAD. Ocurre fuera de los LADs, como mecanismo de represión adicional

### Interpretación de datos Hi-C

Se recolectaron datos Hi-C y la lectura se mapeó de nuevo al genoma. Los recuentos de lecturas se compilaron en una matriz  $O$  (que se muestra a continuación para el cromosoma 14) donde el elemento  $O_{i,j}$  indica el número de lecturas correspondientes a una interacción entre las posiciones  $i$  y  $j$ . Una diagonal fuerte está claramente presente, e indica que las regiones que están cercanas entre sí en 1D también son probables para interactuar en 3D. Los errores en la interpretación de los datos Hi-C pueden ocurrir cuando se violan los supuestos de la técnica: por ejemplo, la suposición de que el genoma de referencia es correcto, lo que puede no ser cierto en el caso de una célula cancerosa. La matriz fue entonces

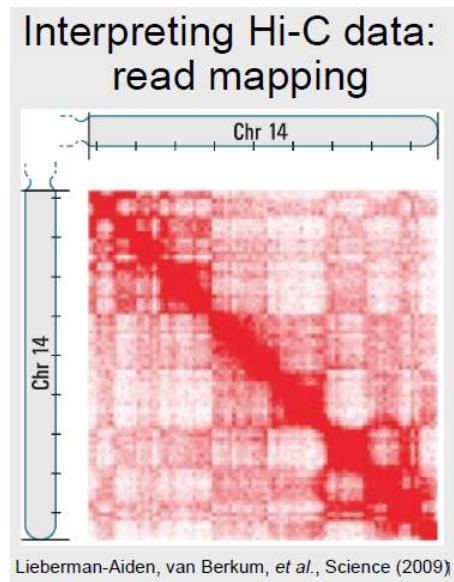


Figura 22.9: Matriz que representa el recuento de lecturas Hi-C

Asociación Americana para el Avance de la Ciencia. Todos los derechos reservados. Este contenido está excluido de nuestro Creativo

Licencia Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Lieberman-Aiden, Erez, et al. "El mapeo integral de interacciones de largo alcance revela principios plegables de el Genoma Humano". Ciencia 326, núm. 5950 (2009): 289-93.

normalizado para tener en cuenta la distancia genética entre dos regiones, y una matriz que indica qué interacciones están Enriquecidas o agotadas en los datos. Para comparar los datos en la matriz, que es bidimensional, con conjuntos de datos genómicos, que son unidimensionales, se debe utilizar el Análisis de Componentes Principales (PCA). Después del PCA, es posible la caracterización funcional de los datos. Hi-C identificó dos tipos globales de regiones:

- Tipo A, que se caracteriza por cromatina abierta, riqueza génica y marcas activas de cromatina.

- Tipo B, que se caracteriza por la cromatina cerrada, y es pobre en genes.

Ambos tipos de regiones son principalmente autointeraccionantes y las interacciones entre los dos tipos son infrecuentes. Hi-C también confirmó la presencia de territorios cromosómicos, ya que hubo muchas más interacciones intracromosómicas que intercromosómicas.

---

This page titled [22.4: Mapeo de interacciones genoma-lámina nuclear \(LADs\)](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.4: Mapping Genome-Nuclear Lamina Interactions \(LADs\)](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear

### Fuentes de sesgo

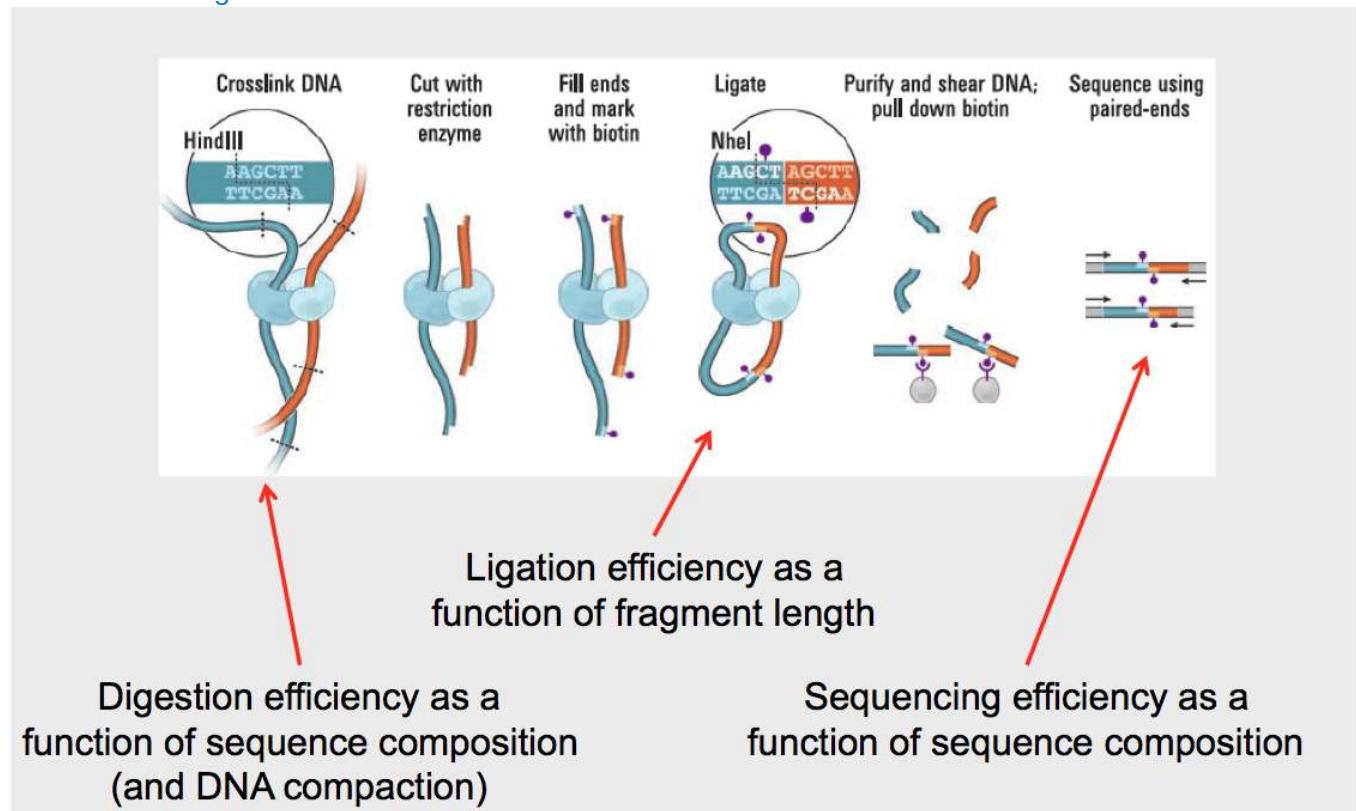


Figura 22.10: Imagen que representa fuentes de sesgo

Asociación Americana para el Avance de la Ciencia. Todos los derechos reservados. Este contenido está excluido de nuestra Licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Lieberman-Aiden, Erez, et al. “El mapeo integral de interacciones de largo alcance revela el plegado Principios del Genoma Humano”. Ciencia 326, núm. 5950 (2009): 289-93.

Los tres pasos que potencialmente podrían introducir sesgos incluyen: Digestión, Ligación y Secuenciación. La eficiencia de la digestión es una función de las enzimas de restricción utilizadas y, por lo tanto, algunas regiones del genoma podrían ser menos propensas a ser digeridas ya que su distribución del sitio de reconocimiento particular podría ser realmente escasa. Además, algunas regiones podrían enriquecerse en el sitio de reconocimiento y con ello quedarán sobrerepresentadas en los resultados. Una solución para esto es usar muchas enzimas de restricción diferentes y comparar los resultados. La eficiencia de ligadura es una función de las longitudes de los fragmentos. Dependiendo de cómo las enzimas de restricción corten la secuencia, algunos extremos pueden tener más o menos probabilidades de ligarse entre sí. Finalmente, la eficiencia de secuenciación es una función de la composición de la secuencia. Algunas cadenas de ADN serán más difíciles de secuenciar, basadas en la riqueza de GC y la presencia de repeticiones, lo que introducirá sesgo.

### Corrección de sesgo

Para minimizar el sesgo de ligadura, se eliminan los productos de ligadura no específicos. Dado que los productos de ligación no específicos típicamente tienen sitios de restricción lejanos, introducen fragmentos mucho más grandes. Además, la influencia del tamaño del fragmento en la eficiencia de ligación ( $F_{\text{lig}}(a_{\text{len}}, b_{\text{len}})$ ), la influencia del contenido de G/C en la amplificación y secuenciación ( $F_{\text{gc}}(a_{\text{gc}}, b_{\text{gc}})$ ), y la influencia de la singularidad de la secuencia en la mapabilidad ( $M(a) * M(b)$ ) todos pueden ser contabilizados y corregidos con la ecuación:

$P(X_{a,b}) = P_{\text{anterior}} * F_{\text{len}}(a_{\text{len}}, b_{\text{len}}) * F_{\text{gc}}(a_{\text{gc}}, b_{\text{gc}}) * M(a) * M(b)$  Alternativamente, las fuentes de sesgo pueden ser menos explícitamente representadas por la siguiente ecuación:

$$O_{i,jj} = B_i * B_j * T_{i,j}$$

donde la suma de todas las probabilidades de contacto relativas  $T_{i,j}$  para cada bin es igual a 1. Los sesgos sólo se asumen como multiplicativos. Esto se resuelve mediante equilibrio de matriz, o ajuste proporcional mediante un algoritmo de corrección iterativa.

## Modelado 3D de datos basados en 3C

El modelado 3D puede revelar muchos principios generales de la organización del genoma. Los modelos actuales se generan usando una combinación de interacciones inter-locus y distancias espaciales conocidas entre hitos nucleares. Sin embargo, queda mucha incertidumbre en los modelos 3D actuales debido a que los datos se recogen de millones de celdas. Los problemas prácticos que afectan al modelado 3D se deben a la gran cantidad de datos necesarios para construir modelos y a las diferentes dinámicas entre una célula individual y una población, que conducen a modelos inestables. El modelado de próxima generación tiende hacia el uso de genómica unicelular.

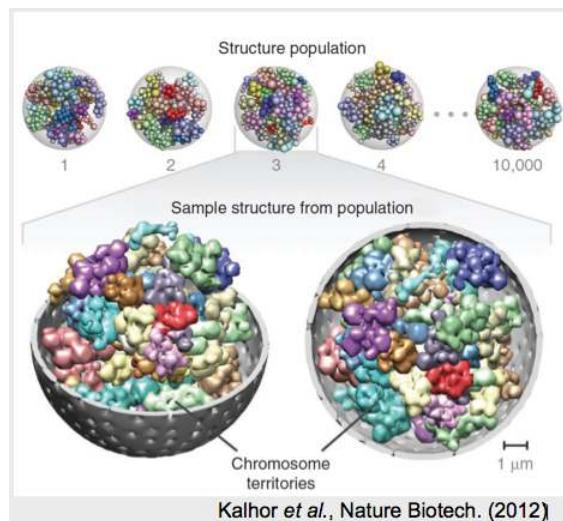


Figura 22.11: Territorios cromosómicos en 3D

This page titled [22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.5: Computational Methods for Studying Nuclear Genome Organization](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#).  
Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.6: Arquitectura de la Organización del Genoma

### Múltiples tipos de células influyen en la determinación de la arquitectura

Las células madre embrionarias (ESC), las células progenitoras neurales (NPC) y los astrocitos (AC) son todos tipos de células isogénicas (todas comienzan como células madre embrionarias). Las células madre embrionarias se dividen constantemente y son completamente indiferenciadas; generan las células progenitoras neurales, que aún se dividen pero menos, y solo se diferencian a mitad de camino. Las células progenitoras neurales generan entonces los astrocitos completamente diferenciados. Se descubrió que durante este proceso de diferenciación, algunas áreas pasaron de ser Dominios Asociados a Lámina a ser dominios interiores. En las células madre embrionarias, hay muy poca transcripción. Sin embargo, la transcripción sube a medida que las células se diferencian cada vez más. Esto coincide con la localización de los dominios de estar asociados principalmente con la lámina (y por lo tanto no expresados) a ser localizados en el interior. Aunque cada uno de estos tipos de células tiene propiedades muy diferentes, un mapa DaMID muestra un alto nivel de similitud entre las tres células isogénicas, así como una célula de fibroblastos independiente. Se emplearon Modelos Ocultos de Markov para identificar los Dominios Asociados a Lámina entre las células. Se encontró una arquitectura cromosómica central con aproximadamente 70% del cromosoma constitutivo (CLAD/ CilAD) y 30% del cromosoma facultativo (FlaD).

### Comparación entre especies de asociaciones de láminas

Para determinar las asociaciones de láminas entre especies, se utilizó un ratón y un ser humano. Se construyó una alineación amplia del genoma entre un ratón y un ser humano. Para cada región genómica en el ratón, la mejor región recíproca se emparejó en el ser humano. Luego se volvió a mapear el genoma humano, y se utilizó para reconstruir un genoma de ratón. Los datos de DAMID se proyectaron en este mapa y hubo 83% de concordancia entre los dos genomas (91% para las regiones constitutivas; 67% para las regiones facultativas).

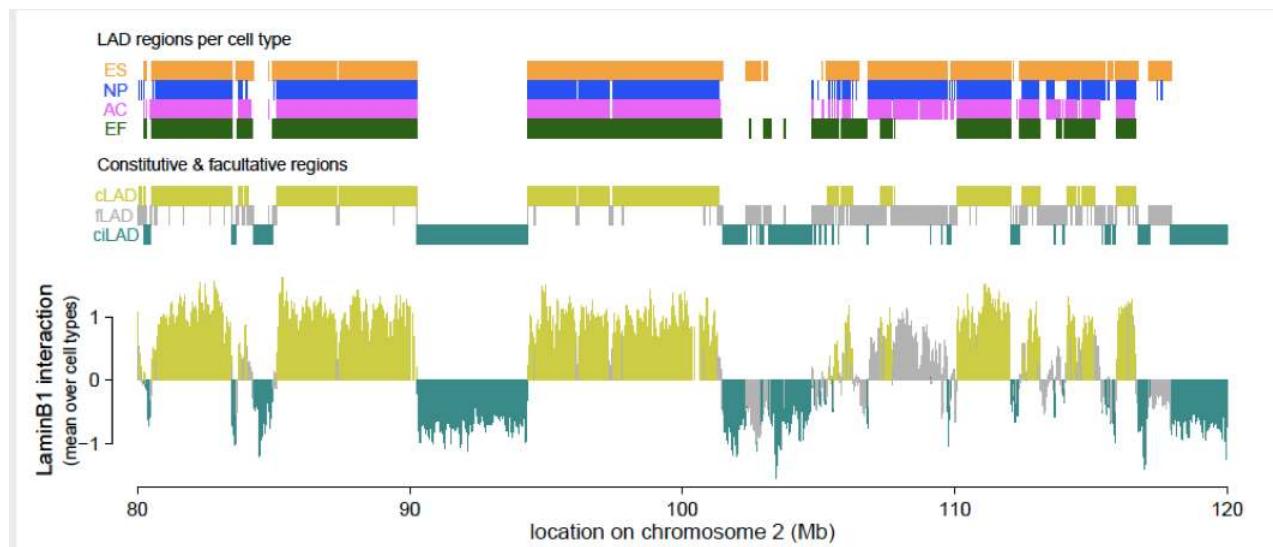


Figura 22.12: Una arquitectura cromosómica central es evidente. Alrededor del 70% de las regiones son constitutivas (CLAD/ CilAD) y el 30% de las regiones son facultativas (FLaD). Prensa de Laboratorio Cold Spring Harbor. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>. Fuente: Meuleman, Wouter, et al. "Las interacciones constitutivas de lámina-genoma Nuclear están altamente conservadas y asociadas con una secuencia rica en A/T". *Genome Research* 23, núm. 2 (2013): 270-80.

### Regla de contenido A-T

Se ha encontrado que el contenido de A-T es un fuerte predictor para la asociación de láminas dentro de la arquitectura del núcleo. El soporte adicional para esta predicción es que la estructura LAD que conforma la arquitectura del núcleo es similar a una estructura isocórica (una gran región uniforme de ADN).

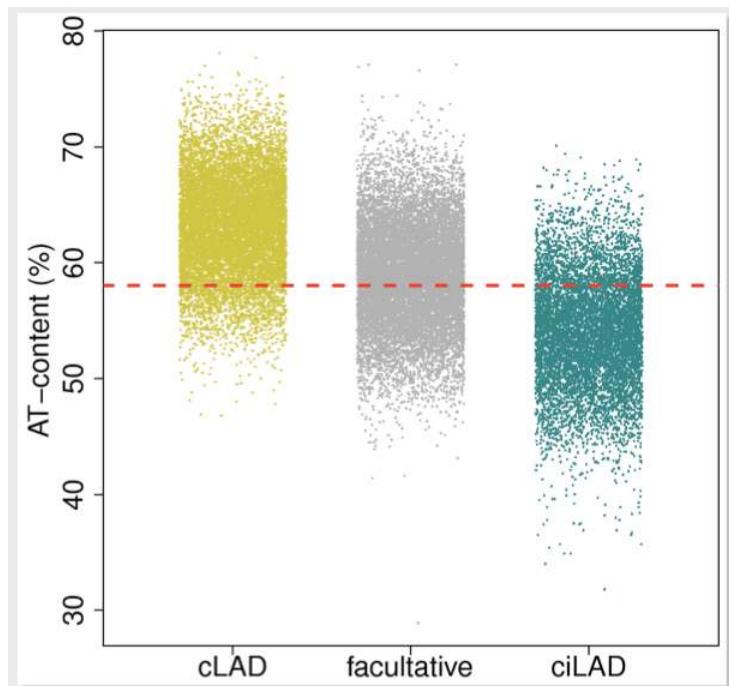


Figura 22.13: Las regiones AT son indicadores para regiones constitutivas

Prensa de Laboratorio Cold Spring Harbor. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Meuleman, Wouter, et al. “Las interacciones constitutivas de lámina nucleares-genoma están altamente conservadas y Asociado con una Secuencia rica en A/T”. *Genoma Reserach* 23, núm. 2 (2013): 270-80.

---

This page titled [22.6: Arquitectura de la Organización del Genoma](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.6: Architecture of Genome Organization](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.7: Comprensión mecanicista de la arquitectura del genoma

La organización de los cromosomas, particularmente en relación espacial con otras partes del cromosoma, no se entiende bien durante el proceso mitótico. Se cree que la conformación de las células se ajusta a dos estados diferentes. Las conformaciones altamente compartimentadas y específicas de tipo celular están casi completamente limitadas a la interfase. Durante la transición a la metafase, los cromosomas entran en un locus y un estado de plegamiento independiente del tejido.

Durante el proceso mitótico, aproximadamente 30% de los LAD se colocan a lo largo de la periferia celular. Este posicionamiento, sin embargo, refleja el contacto proteína-lámina a intervalos intermitentes, sin embargo, las células están restringidas a la periferia de las células. Durante la división mitótica, este posicionamiento laminar es heredado estocásticamente por las células infantiles.

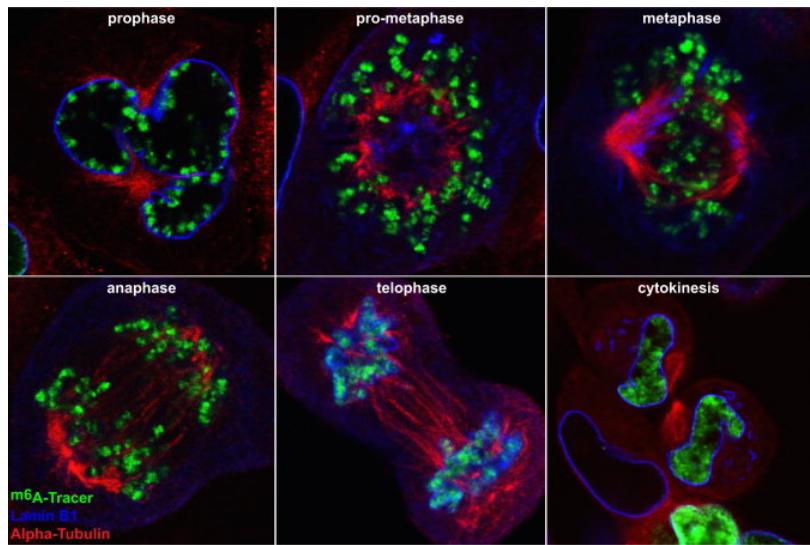


Figura 22.14: LADs a través del ciclo celular (único) (Kind et al, Cell 2013)

Cortesía de Elsevier, Incorporar, Usado con permiso.

Fuente: Kind, Jop, et al. "Dinámica unicelular del Genoma nuclear

Interacciones Lámina." *Célula* 153, núm. 1 (2013): 178-92

### Modelado

El modelado tridimensional será cada vez más importante en la comprensión de las interacciones cromosómicas. Las técnicas actuales han modelado el genoma de levadura y el locus de la globina (Duan et al. Nature (2010), Bau et al. Naturaleza SMB (2011)). A partir de modelar studeis ha quedado claro que no podemos generar una relación directa entre la probabilidad de contacto y la distancia espacial (es decir, probabilidad de contacto! = distancia espacial).

Modelar, sin embargo, es un problema inverso, es más difícil ir en un sentido que en otro. Específicamente, es más fácil pasar de la estructura proteica a un mapa de contacto proteico que viceversa. De igual manera, la estructura cromosómica es un problema duro, aunque tengamos un mapeo de contacto.

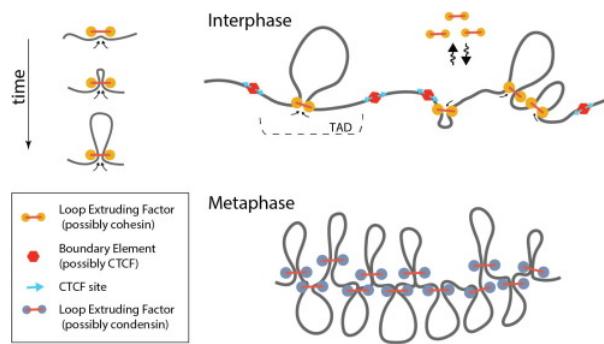


Figura 22.15: Extrusión en bucle como mecanismo de orientación cromosómica

This page titled [22.7: Comprensión mecánica de la arquitectura del genoma](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.7: Mechanistic Understanding of Genome Architecture](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 22.8: Direcciones actuales de investigación

### LADs

- (a) 30% del genoma es variable entre tipos celulares, cómo podemos diferenciar estas diferencias
- b) ¿Cómo interactúan la lámina y los LAD? ¿Hay alguna atracción entre los LADs y estos dominios, o se basa en la repulsión del interior?
- (c) Por qué y cómo se reprimen los genes a lo largo de la periferia de los LAD.

### TAD y otros compartimentos:

- a) ¿Cuál es la base biológica de los compartimentos? ¿Existe algún componente multifacético de los compartimentos?
- b) ¿Cómo funcionan los cohesinos? ¿Son suficientes los pares de cohesina-extrusión para explicar todos los dominios?
- (c) ¿Los bucles mejorador-promotor están confinados a dominios específicos? ¿Son estos componentes dinámicos?/¿Son bucles arquitectónicos mediados por CTCF?

### Otros/Misceláneos:

- a) ¿Cómo relacionamos los diferentes componentes cromosómicos (es decir, LAD, TAD, dominio polycomb, orígenes de réplica, modificaciones de histonas, expresión génica)?
- b) Bases evolutivas de la arquitectura genómica: ¿hubo una presión evolutiva y cuándo surgieron los principios del plegamiento?
- (c) ¿En los cambios cromosómicos ocurren primero las localizaciones o cambios en la expresión?

### ¿Sabías?

¡Esta pregunta ha sido abordada (parcialmente)! Al investigar células que pasan por múltiples rondas de diferenciación, se ha observado que algunas regiones se localizarán a la lámina en la primera diferenciación pero ¡no quedarán reprimidas hasta la segunda diferenciación!

### Hipótesis de guardia corporal

La hipótesis del guardaespaldas fue propuesta en 1975 por Hsu TC. Sugiere que el ADN inactivo se localiza en la periferia del núcleo para que pueda 'proteger' las regiones importantes y activas del ADN de peligros extraños como virus o radicales libres. Los intentos de probar la hipótesis introduciendo daño artificial en el ADN han producido resultados circunstanciales, y la pregunta permanece abierta. Experimentos unicelulares

Se sabe que las células conservan su organización original después de la mitosis, como lo demuestran los experimentos de tinción cromosómica. Sin embargo, experimentos recientes han demostrado que puede haber una gran diferencia en la organización entre las células madre e hija. Ciertas propiedades globales, como los territorios cromosómicos, se conservan, pero la organización en un detalle más fino puede diferir mucho. La experimentación unicelular es una técnica emergente que puede ser capaz de abordar esta pregunta abierta.

### FAQ

P: ¿Alguien ha intentado incrementar la expresión de un gen en medio de un LAD? ¿Qué pasó?

R: No está claro si existe un ejemplo específico de esto, sin embargo se han realizado varios estudios relacionados. Los investigadores han intentado 'atar' una región de ADN a la lámina nuclear para ver si se desactiva espontáneamente. Sin embargo, los resultados no fueron concluyentes ya que en la mitad de los casos la región se volvería inactiva y en la otra mitad ¡no lo haría! Hasta el momento este tipo de manipulaciones no han dado mucho, pero se encontró que si un segmento de ADN desprovisto de proteínas se digirió y se mezclaba con proteínas de lámina altamente purificadas, los fragmentos unidos revelan un patrón muy similar al de los LAD. Esto nos dice que la lámina se une directamente al ADN. Sin embargo, esto parece variar entre especies.

This page titled [22.8: Direcciones actuales de investigación](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [22.8: Current Research Directions](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## CHAPTER OVERVIEW

### 23: Introducción al Modelado Metabólico en Estado Estable

[23.1: Introducción](#)

[23.2: Construcción de modelos](#)

[23.3: Análisis de Flujo Metabólico](#)

[23.4: Aplicaciones](#)

[23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía](#)

[23.6: Herramientas y Techiniques](#)

[Bibliografía](#)

---

This page titled [23: Introducción al Modelado Metabólico en Estado Estable](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## 23.1: Introducción

El modelado metabólico nos permite utilizar modelos matemáticos para representar sistemas biológicos complejos. Esta conferencia discute el papel de modelar el estado estacionario de los sistemas biológicos en la comprensión de las capacidades metabólicas de los organismos. También discutimos brevemente qué tan bien los modelos de estado estacionario son capaces de replicar experimentos in vitro.

### ¿Qué es Metabolismo?

Según Matthews y van Holde, el metabolismo es la totalidad de todas las reacciones químicas que ocurren en la materia viva. Esto incluye reacciones catabólicas, que son reacciones que conducen a la descomposición de moléculas en componentes más pequeños, y reacciones anabólicas, que son responsables de la creación de moléculas más complejas (por ejemplo, proteínas, lípidos, carbohidratos y ácidos nucleicos) a partir de componentes más pequeños. Estas reacciones son responsables de la liberación de energía de los enlaces químicos y del almacenamiento de esta energía. Las reacciones metabólicas también son responsables de la transducción y transmisión de información (por ejemplo, a través de la generación de GMPc como mensajero secundario o ARNm como sustrato para la traducción de proteínas).

### ¿Por qué Modelo Metabolismo

Una aplicación importante del modelado metabólico es en la predicción de los efectos de los medicamentos. Un tema importante de modelación es el organismo *Mycobacterium tuberculosis* [15]. La interrupción de las vías de síntesis de ácido micólico de este organismo puede ayudar a controlar la infección por TB. El modelado computacional nos brinda una plataforma para identificar las mejores dianas farmacológicas en este sistema. Los estudios de inactivación génica en *Escherichia coli* han permitido a los científicos determinar qué genes y combinaciones de genes afectan el crecimiento de este importante organismo modelo [6]. Tanto los acuerdos como los desacuerdos entre modelos y datos experimentales pueden ayudarnos a evaluar nuestro conocimiento de los sistemas biológicos y ayudarnos a mejorar nuestras predicciones sobre las capacidades metabólicas. En la próxima conferencia, aprenderemos la importancia de incorporar datos de expresión en modelos metabólicos. Además, una variedad de procesos de enfermedades infecciosas implican cambios metabólicos a nivel microbiano.

---

This page titled [23.1: Introducción](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [23.1: Introduction](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 23.2: Construcción de modelos

Un objetivo general del modelado metabólico es la capacidad de tomar una representación esquemática de una ruta y cambiarla en una fórmula matemática que modela la ruta. Por ejemplo, convertir la siguiente vía en un modelo matemático sería de increíble utilidad.

### Reacciones Químicas

En los modelos metabólicos, nos preocupa modelar reacciones químicas que son catalizadas por enzimas. Las enzimas actúan actuando sobre un estado de transición del complejo enzima-sustrato que disminuye la energía de activación de una reacción química. El diagrama de la diapositiva 5 de la página 1 de las diapositivas de la conferencia demuestra este fenómeno. Una ecuación de velocidad típica (que describe la conversión de los sustratos S de la reacción enzimática en sus productos P) puede ser descrita por una ley de tasa de Michaelis-Menten:

$$\frac{V}{V_{\max}} = \frac{[S]}{K_m + [S]}$$

En esta ecuación, V es la tasa de la ecuación en función de la concentración de sustrato [S]. Es claro que los parámetros  $K_m$  y  $V_{\max}$  son necesarios para caracterizar la ecuación.

La inclusión de múltiples sustratos, productos y relaciones regulatorias aumenta rápidamente el número de parámetros necesarios para caracterizar tales ecuaciones. Las figuras de las diapositivas 1, 2 y 3 de la página 2 de las notas de la conferencia demuestran la complejidad de las vías bioquímicas. El modelado cinético rápidamente se vuelve inviable: los parámetros necesarios son difíciles de medir y también varían entre organismos [10]. Así, nos interesa un método de modelado que nos permita utilizar un pequeño número de parámetros determinados con precisión. Para ello, recordamos la maquinaria básica de la estequiometría de la química general. Considera la ecuación química  $A+2B \rightarrow 3C$ , que dice que una unidad de reactivo A se combina con 2 unidades de reactivo B para formar 3 unidades de reactivo C. La velocidad de formación del compuesto X viene dada por la derivada temporal de [X]. Obsérvese que C se forma tres veces más rápido que A. Por lo tanto, debido a la estequiometría de la reacción, vemos que la velocidad de reacción (o flujo de reacción) viene dada por

$$\text{flux} = \frac{d[A]}{dt} = \frac{1}{2} \frac{d[B]}{dt} = \frac{1}{3} \frac{d[C]}{dt}$$

Esto será útil en las secciones subsiguientes. Ahora debemos exponer los supuestos simplificadores que hacen que nuestro modelo sea manejable.

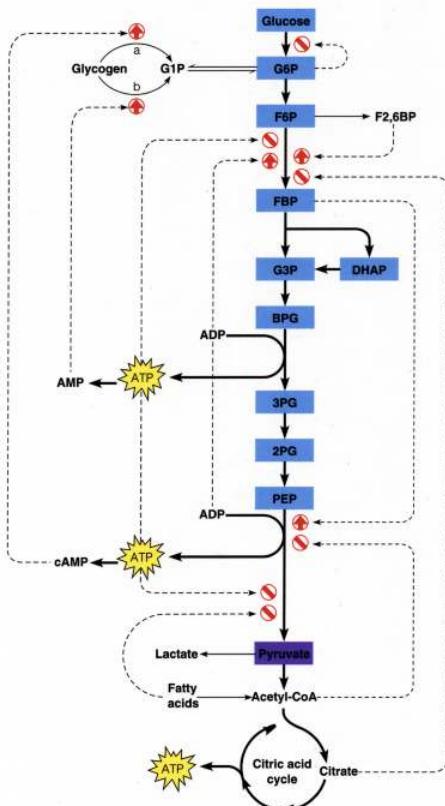


Figura 23.1: Proceso que conduce e incluye el ciclo del ácido cítrico. Pearson. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>. Fuente: Matthews, C.K., et al. "Bioquímica". (2000).

## Asunción de Estado Constante

La suposición del estado estacionario supone que no hay acumulación de ningún metabolito en el sistema. Esto nos permite representar las reacciones completamente en términos de su química (es decir, las relaciones estequiométricas entre los componentes de la reacción enzimática). Tenga en cuenta que esto no implica la ausencia de flujo a través de ninguna reacción dada. Más bien, el estado estacionario implica en realidad dos supuestos que son críticos para simplificar el modelado metabólico. La primera es que las concentraciones internas de metabolitos son constantes, y la segunda es que los flujos, es decir, los flujos de entrada y salida, también son constantes.

Una analogía es una serie de cascadas que aportan agua a las piscinas. A medida que el agua cae de una alberca a otra, los niveles de agua no cambian a pesar de que el agua sigue fluyendo (ver página 2 diapositiva 5). Este marco nos impide ser obstaculizados por la cinética transitoria demasiado complicada que puede resultar de las perturbaciones del sistema. Dado que generalmente nos interesan las capacidades metabólicas a largo plazo (funciones en una escala superior a milisegundos o segundos), la dinámica del estado estacionario puede darnos toda la información que necesitamos.

La suposición de estado estacionario hace que la capacidad de generalizar entre especies y reutilizar vías conservadas en modelos sea mucho más factible. Las estequiométrias de reacción a menudo se conservan en todas las especies, ya que solo implican la conservación de la masa. La biología de la catálisis enzimática, y los parámetros que la caracterizan, no se conservan de manera similar. Estos incluyen parámetros dependientes de especies como la energía de activación de una reacción, la anidad del sustrato de una enzima y las constantes de velocidad para diversas reacciones. Sin embargo, ninguno de estos es requerido para el modelado en estado estacionario.

También es de interés señalar que, dado que las constantes de tiempo para las reacciones metabólicas suelen ser del orden de los milisegundos, la mayoría de las tecnologías de medición utilizadas hoy en día no son capaces de capturar estas dinámicas extremadamente rápidas. Este es el caso de mediciones basadas en espectrometría de masas metabolómica, por ejemplo. En este

método, las cantidades de todos los metabolitos internos en un sistema se miden en un momento dado, pero las mediciones se pueden tomar en el mejor de los casos cada hora. En la mayoría de las circunstancias, todo lo que se mide es el estado estacionario.

## Reconstrucción de vías metabólicas

Existen varias bases de datos que pueden proporcionar la información necesaria para reconstruir vías metabólicas in silico. Estas bases de datos permiten acceder a la estequiométría de reacción usando números de Enzyme Commission. Las estequiométrías de reacción son las mismas en todos los organismos que utilizan una enzima dada. Entre las bases de datos de interés se encuentran ExPASY [5], MetaCyc [16] y KEGG [14]. Estas bases de datos a menudo contienen vías organizadas por función que se pueden descargar en formato SBML, lo que hace que la reconstrucción de vías sea muy fácil para vías bien caracterizadas.

---

This page titled [23.2: Construcción de modelos](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [23.2: Model Building](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 23.3: Análisis de Flujo Metabólico

El análisis de flujo metabólico (MFA) es una forma de calcular la distribución de los flujos de reacción que es posible en una red metabólica dada en estado estacionario. Podemos imponer restricciones a ciertos flujos con el fin de limitar el espacio descrito por la distribución de posibles flujos. En esta sección, desarrollaremos una formulación matemática para MFA. Una vez más, este análisis es independiente de la biología particular del sistema; más bien, sólo dependerá de las estequiométrias (universales) de las reacciones en cuestión.

### Representación matemática

Considerar un sistema con  $m$  metabolitos y  $n$  reacciones. Sea  $x_i$  la concentración de sustrato  $i$ , de manera que la velocidad de cambio de la concentración de sustrato esté dada por la derivada temporal de  $x_i$ . Sea  $x$  el vector de columna (con  $m$  componentes) con elementos  $x_i$ . Por simplicidad, consideramos un sistema con  $m = 4$  metabolitos A, B, C y D. Este sistema consistirá en muchas reacciones entre estos metabolitos, resultando en un complicado equilibrio entre estos compuestos.

Una vez más, considere la simple reacción  $A + 2B \rightarrow 3C$ . Podemos representar esta reacción en forma de vector como  $(-1 \ -2 \ 3 \ 0)$ . Obsérvese que los dos primeros metabolitos (A y B) tienen signos negativos, ya que se consumen en la reacción. Además, los elementos del vector están determinados por la estequiometría de la reacción, como en la Sección 2.1. Repetimos este procedimiento para cada reacción en el sistema. Estos vectores se convierten en las columnas de la matriz estequiométrica  $S$ . Si el sistema tiene  $m$  metabolitos y  $n$  reacciones,  $S$  será una matriz  $m \times n$ . Por lo tanto, si definimos  $v$  como el vector de columna de  $n$ -componentes de flujos en cada reacción, el vector  $Sv$  describe la velocidad de cambio de la concentración de cada metabolito. Matemáticamente, esto se puede representar como la ecuación fundamental del análisis del flujo metabólico:

$$\frac{dx}{dt} = Sv$$

La matriz  $S$  es una estructura de datos extraordinariamente poderosa que puede representar una variedad de escenarios posibles en sistemas biológicos. Por ejemplo, si dos columnas  $c$  y  $d$  de  $S$  tienen la propiedad de que  $c = d$ , las columnas representan una reacción reversible. Además, si una columna tiene la propiedad de que solo un componente es distinto de cero, representa en reacción de intercambio, en la que hay un flujo hacia (o desde) un sumidero (o fuente) supuestamente infinito, dependiendo del signo del componente distinto de cero.

Ahora imponemos la suposición de estado estacionario, que dice que el tamaño izquierdo de la ecuación anterior es idénticamente cero. Por lo tanto, necesitamos encontrar vectores  $v$  que satisfagan el criterio  $Sv = 0$ . Las soluciones a esta ecuación determinarán flujos factibles para este sistema.

### Espacio nulo de $S$

El espacio de flujo factible de las reacciones en el sistema modelo se define por el espacio nulo de  $S$ , como se vio anteriormente. Recordemos del álgebra lineal elemental que el espacio nulo de una matriz es un espacio vectorial; es decir, dados dos vectores  $y$  y  $z$  en el espacio nulo, el vector  $ay + bz$  (para los números reales  $a, b$ ) también está en el espacio nulo. Dado que el espacio nulo es un espacio vectorial, existe una base  $b_i$ , un conjunto de vectores que es linealmente independiente y abarca el espacio nulo. La base tiene la propiedad de que para cualquier flujo  $v$  en el espacio nulo de  $S$ , existen números reales  $\alpha_i$  tales que

$$v = \sum_i \alpha_i b_i$$

¿Cómo encontramos una base para el espacio nulo de una matriz? Una herramienta útil es la descomposición de valores singulares (SVD) [4]. La descomposición del valor singular de una matriz  $S$  se define como una representación  $S = U\Lambda V^*$ , donde  $U$  es una matriz unitaria de tamaño  $m$ ,  $V$  es una matriz unitaria de tamaño  $n$ , y  $\Lambda$  es una matriz diagonal  $m \times n$ , con los valores singulares (necesariamente positivos) de  $S$  en orden descendente. (Recordemos que una matriz unitaria es una matriz con columnas y filas ortonormales, es decir,  $U^*U = UU^* = I$  la matriz de identidad). Se puede demostrar que cualquier matriz tiene una SVD. Tenga en cuenta que el SVD se puede reorganizar en la ecuación  $Sv = \sigma u$ , donde  $u$  y  $v$  son columnas de las matrices  $U$  y  $V$  y  $\sigma$  es un valor singular. Por lo tanto, si  $\sigma = 0$ ,  $v$  pertenece al espacio nulo de  $S$ . Efectivamente, las columnas de  $V$  que corresponden a los valores singulares cero forman una base ortonormal para el espacio nulo de  $S$ . De esta manera, la SVD nos permite caracterizar completamente los posibles flujos para el sistema.

## Restricción del espacio de flujo

La primera restricción mencionada anteriormente es que todos los vectores de flujo de estado estacionario deben estar en el espacio nulo. También los flujos negativos no son termodinámicamente posibles. Por lo tanto, una restricción fundamental es que todos los flujos deben ser positivos. (Dentro de este marco representamos reacciones reversibles como reacciones separadas en la matriz estequiométrica S que tienen dos flujos unidireccionales).

Estas dos restricciones clave forman un sistema que puede resolverse mediante análisis convexo. La región de solución puede describirse mediante un conjunto único de Caminos Extremos. En esta región, los vectores de flujo en estado estacionario v pueden describirse como una combinación lineal positiva de estas vías extremas. Los Caminos Extremos, representados en la diapositiva 25 como vectores  $b_i$ , circunscriben un cono de flujo convexo. Cada dimensión es una velocidad para alguna reacción. En el portaobjetos 25, la dimensión z representa la velocidad de reacción para  $v_3$ . Podemos reconocer que en cualquier momento, el organismo está viviendo en un punto en el cono de flujo, es decir, está demostrando una distribución de flujo particular. Cada punto en el cono de flujo puede ser descrito por un posible vector de flujo de estado estacionario, mientras que los puntos fuera del cono no pueden.

Un problema es que el cono de flujo sale al infinito, mientras que los flujos infinitos no son físicamente posibles. Por lo tanto, una restricción adicional es tapar el cono de flujo al determinar los flujos máximos de cualquiera de nuestras reacciones (estos valores corresponden a nuestros parámetros  $V_{max}$ ). Dado que muchas reacciones metabólicas son interiores a la célula, no hay necesidad de establecer un límite para cada flujo. Estos topes se pueden determinar experimentalmente midiendo flujos máximos, o calculados usando herramientas matemáticas como reglas de difusividad.

También podemos agregar flujos de entrada y salida que representan el transporte dentro y fuera de nuestras celdas ( $V_{in}$  y  $V_{out}$ ). Estos suelen ser mucho más fáciles de medir que los flujos internos y, por lo tanto, pueden servir para ayudarnos a generar un espacio de flujo más relevante biológicamente. Un ejemplo de algoritmo para resolver este problema es el algoritmo simplex [1]. Las diapositivas 24-27 demuestran cómo las restricciones en los flujos cambian la geometría del cono de flujo. En realidad, estamos lidiando con problemas en espacios de dimensiones superiores.

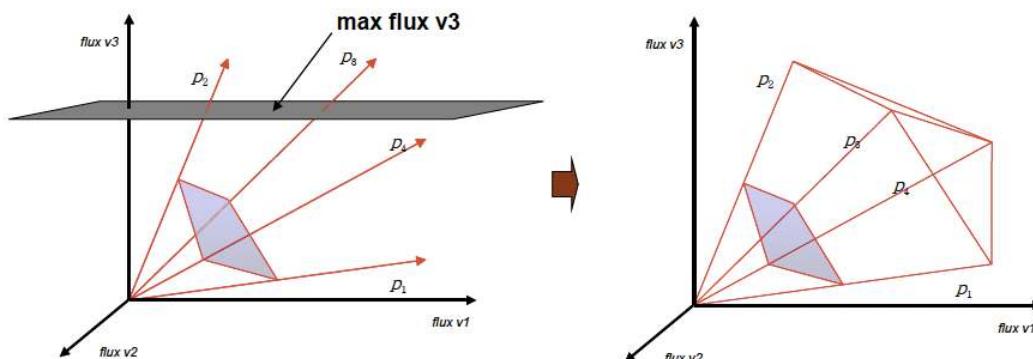


Figura 23.2: Agregar restricciones a vías extremas.

## Programación Lineal

La programación lineal es una solución genérica que es capaz de resolver problemas de optimización dadas las restricciones lineales. Estos se pueden representar en algunas formas diferentes.

Forma canónica:

- Maximizar:  $c^T x$
- Sujeto a:  $Ax \leq b$

Forma estándar:

- Maximizar  $\sum c_i * x_i$
- Sujeto a:  $a_{ij}X_i \leq b_i$  forall  $i, j$
- Restricción no negatividad:  $X_i \geq 0$

Una introducción concisa y clara a la Programación Lineal está disponible aquí: [www.purplemath.com/modules/linprog.htm](http://www.purplemath.com/modules/linprog.htm) Las restricciones descritas a lo largo de la sección 3 nos dan el problema de programación lineal descrito en la conferencia. La programación lineal puede considerarse una primera aproximación y es un problema clásico en la optimización. Para tratar de reducir nuestro flujo factible, asumimos que existe una función de aptitud que es una combinación lineal de cualquier número de flujos en el sistema. La programación lineal (o optimización lineal) implica maximizar o minimizar una función lineal sobre un poliedro convexo especificado por restricciones lineales y de no negatividad.

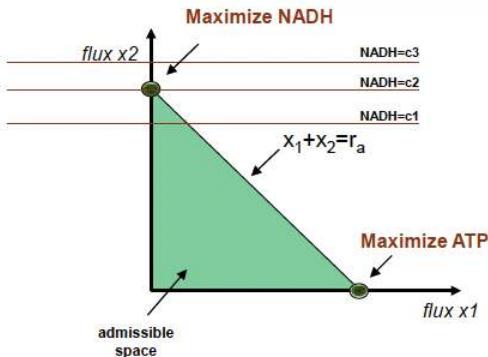


Figura 23.3: Maximización de dos funciones con programación lineal.

Resolvemos este problema identificando la distribución de flujo que maximiza una función objetiva:

El punto clave en la programación lineal es que nuestras soluciones se encuentran en los límites del espacio de flujo permisible y pueden estar en puntos, bordes o ambos. Sin embargo, por definición, una solución óptima (si existe) estará en un punto del espacio de flujo permisible. Este concepto se demuestra en la diapositiva 30. En ese portaobjetos, A es la matriz estequiométrica, x es el vector de flujos y b es un vector de flujos máximos permisibles.

Los programas lineales, cuando se resuelven a mano, generalmente se realizan por el método Simplex. El método simplex configura el problema en una matriz y realiza una serie de pivotes, basados en las variables básicas de la declaración del problema. En el peor de los casos, sin embargo, esto puede correr en tiempo exponencial. Por suerte, si hay una computadora disponible, otros dos algoritmos están disponibles. El algoritmo elíptico y los métodos de Punto Interior son capaces de resolver cualquier programa lineal en tiempo polinomial. Es interesante señalar, que muchos problemas aparentemente difíciles pueden modelarse como programas lineales y resolverse eficientemente (o tan eficientemente como una solución genérica puede resolver un problema específico).

En microbios como *E. coli*, esta función objetiva suele ser una combinación de flujos que contribuyen a la biomasa, como se ve en el portaobjetos 31. Sin embargo, esta función no necesita ser completamente biológicamente significativa.

Por ejemplo, podríamos simular la maximización de micolatos en *M. tuberculosis*, aunque esto no ocurra biológicamente. Nos daría predicciones significativas sobre qué perturbaciones podrían realizarse *in vitro* que perturbarían la síntesis de micolados incluso en ausencia de la maximización de la producción de esos metabolitos. El análisis de balance de flujo (FBA) fue pionero por el grupo Palsson en la UCSD y desde entonces se ha aplicado a *E. coli*, *M. tuberculosis*, y el glóbulo rojo humano [?].

---

This page titled [23.3: Análisis de Flujo Metabólico](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

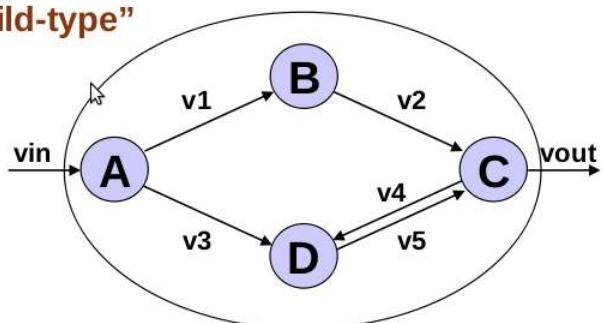
- [23.3: Metabolic Flux Analysis](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 23.4: Aplicaciones

### Análisis de detección de Silico

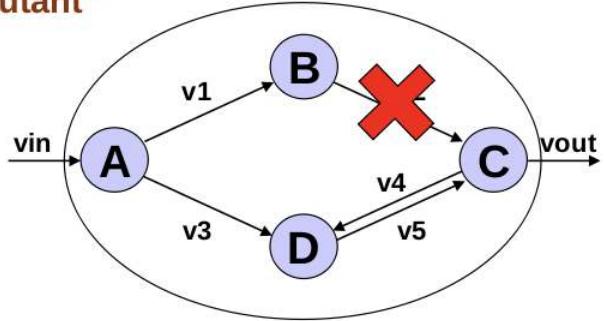
Con la disponibilidad de una herramienta tan poderosa como FBA, surgen más preguntas de forma natural. Por ejemplo, ¿somos capaces de predecir el fenotipo knockout génico en función de sus efectos simulados sobre el metabolismo? Además, ¿por qué trataríamos de hacer esto, a pesar de que existen otros métodos, como el mapa de interacción de proteínas conectivo? Dicho análisis es realmente necesario, ya que otros métodos no toman en consideración directa el flujo metabólico u otras condiciones metabólicas específicas.

#### “wild-type”



|   | v1 | v2 | v3 | v4 | v5 | vin | vout |
|---|----|----|----|----|----|-----|------|
| A | -1 | 0  | -1 | 0  | 0  | 1   | 0    |
| B | 1  | -1 | 0  | 0  | 0  | 0   | 0    |
| C | 0  | 1  | 0  | -1 | 1  | 0   | -1   |
| D | 0  | 0  | 1  | 1  | -1 | 0   | 0    |

#### “mutant”



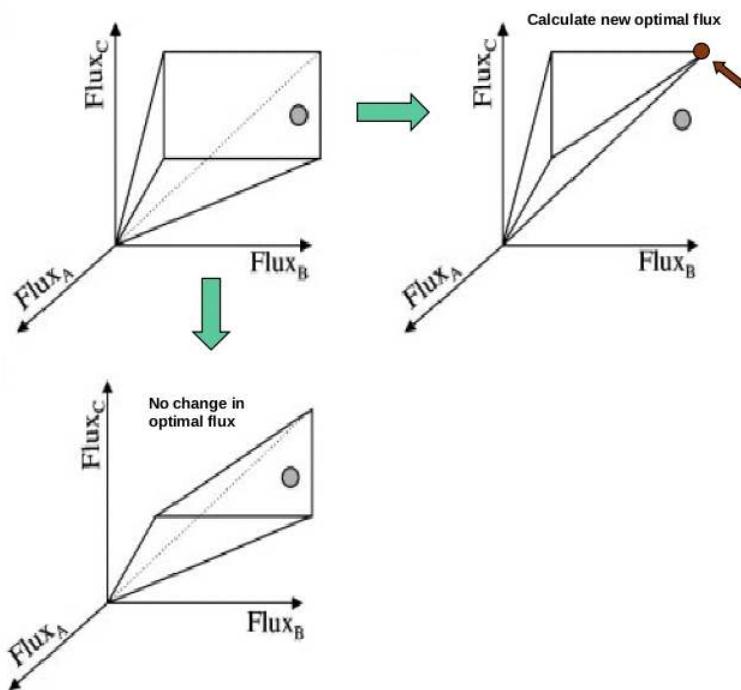
|   | v1 | v2 | v3 | v4 | v5 | vin | vout |
|---|----|----|----|----|----|-----|------|
| A | -1 |    | -1 | 0  | 0  | 1   | 0    |
| B | 1  |    | 0  | 0  | 0  | 0   | 0    |
| C | 0  |    | 0  | -1 | 1  | 0   | -1   |
| D | 0  |    | 1  | 1  | -1 | 0   | 0    |

fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 23.4: Eliminar una reacción es lo mismo que eliminar un gen de la matriz estequiométrica.

La noqueación de un gen en un experimento se modela simplemente eliminando una de las columnas (reacciones) de la matriz estequiométrica. (Una pregunta durante la clase aclaró que un solo gen puede noquear múltiples columnas/reacciones). De este modo, estas mutaciones knockout limitarán aún más el espacio de solución factible al eliminar los flujos y sus vías extremas relacionadas. Si el flujo óptimo original estaba fuera del nuevo espacio, entonces se crea un nuevo flujo óptimo. Así, el análisis de FBA producirá diferentes soluciones. La solución es una tasa de crecimiento máxima, la cual puede ser confirmada o desprobada experimentalmente. La tasa de crecimiento en la nueva solución proporciona una medida del fenotipo knockout. Si estos knockouts de genes son de hecho letales, entonces la solución óptima será una tasa de crecimiento de cero.

Estudios de Edwards, Palsson (1990) exploran el uso de la predicción de fenotipos knockout para predecir cambios metabólicos en respuesta a la eliminación de enzimas en *E. coli*, un procariota [? ]. En otras palabras, se construyó un modelo metabólico in silico de *E. coli* para simular mutaciones que afectan las vías de glucólisis, fosfato de pentosa, TCA y transporte de electrones (436 metabolitos y 719 reacciones incluidas). Para cada condición específica, se comparó el crecimiento óptimo de mutantes con los no mutantes. Luego se compararon los resultados in vivo e in silico, con 86% de acuerdo. Los errores en el modelo indican un modelo subdesarrollado (falta de conocimiento). Los autores discuten 7 errores no modelados por FBA, incluyendo mutantes que inhiben la síntesis estable de ARN y producen intermedios tóxicos.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 23.5: La restricción del espacio de solución factible puede crear un nuevo flujo óptimo.

### Predicciones del modelo de flujo cuantitativo en si

¿Pueden los modelos predecir cuantitativamente los flujos, la tasa de crecimiento? Demostramos la capacidad de FBA para dar predicciones cuantitativas sobre la tasa de crecimiento y los flujos de reacción dadas las diferentes condiciones ambientales. Más específicamente, la predicción se refiere a flujos medibles externamente en función de las tasas de absorción controladas y las condiciones ambientales. Dado que FBA maximiza una función objetivo, dando como resultado un valor específico para esta función, en teoría deberíamos poder extraer información cuantitativa del modelo.

Un ejemplo temprano de Edwards, Ibarra y Palsson (1991), predijo la tasa de crecimiento de *E. coli* en cultivo dado un rango de tasas fijas de absorción de oxígeno y dos fuentes de carbono (acetato y succinato), que podrían controlar en un reactor discontinuo [6]. Supusieron que las células de *E. coli* ajustan su metabolismo para maximizar el crecimiento (usando una función objetivo de crecimiento) bajo condiciones ambientales dadas y utilizaron FBA para modelar las vías metabólicas en la bacteria. La entrada a este modelo en particular es el acetato y el oxígeno, el cual está etiquetado como VIN.

Las tasas de absorción controladas fijaron los valores de los flujos de entrada de oxígeno y acetato/succinato en la red, pero los otros flujos se calcularon para maximizar el valor del objetivo de crecimiento.

La tasa de crecimiento aún se trata como la solución al análisis de FBA. En suma, la tasa de crecimiento óptima se predice en función de las restricciones de absorción de oxígeno versus acetato y oxígeno versus succinato. El modelo básico es una línea predictiva y puede confirmarse experimentalmente en un biorreactor midiendo la captación y el crecimiento de reactores discontinuos (nota: la captación experimental no fue restringida, solo se midió).

Este modelo de Palsson fue la primera buena prueba de principio in silico modelo. Las predicciones cuantitativas de la tasa de crecimiento de los autores bajo las condiciones dierentes coincidieron muy estrechamente con las tasas de crecimiento observadas experimentalmente, lo que implica que *E. coli* sí tiene una red metabólica que está diseñada para maximizar el crecimiento. Tuvo buenas tasas verdaderas positivas y verdaderas negativas. La concordancia entre las predicciones y los resultados experimentales es muy impresionante para un modelo que no incluye ninguna información cinética, solo estequiométría. El profesor Galagan advirtió, sin embargo, que a menudo es difícil saber qué es el buen acuerdo, porque desconocemos la importancia del tamaño de los

residuos. El organismo se cultivó sobre una serie de nutrientes diferentes. Por lo tanto, los investigadores pudieron predecir el crecimiento específico de la condición. Ten en cuenta que esto funcionó, ya que solo ciertos genes son necesarios para algunos nutrientes, como fbp para la gluconeogénesis. Por lo tanto, noquear fbp solo será letal cuando no haya glucosa en el ambiente, condición específica que resultó en una solución de crecimiento al ser analizada por FBA.

## Modelado de estado cuasi estacionario (QSSM)

Ahora podemos describir cómo usar FBA para predecir cambios dependientes del tiempo en las tasas de crecimiento y las concentraciones de metabolitos usando modelos cuasi de estado estacionario. En el ejemplo anterior se utilizó FBA para hacer predicciones cuantitativas de crecimiento bajo condiciones ambientales específicas (predicciones puntuales). Ahora, después de los flujos de crecimiento y captación, pasamos a otra suposición y tipo de modelo.

¿Podemos usar un modelo de metabolismo en estado estacionario para predecir los cambios dependientes del tiempo en la célula o los entornos? Tenemos que hacer una serie de supuestos cuasi de estado estacionario (QSSA):

1. El metabolismo se ajusta a los cambios ambientales/celulares más rápidamente que los cambios en sí
2. Las concentraciones celulares y ambientales son dinámicas, pero el metabolismo opera con la condición de que la concentración sea estática en cada punto temporal (modelo de estado estacionario).

¿Es posible utilizar QSSM para predecir la dinámica metabólica a lo largo del tiempo? Por ejemplo, si se toma menos acetato por célula a medida que crece el cultivo, entonces la tasa de crecimiento debe disminuir. Pero ahora, se aplican los supuestos de QSSA. Es decir, en efecto, en cualquier momento dado, el organismo se encuentra en estado estacionario.

¿Qué valores obtiene uno como solución al problema de FBA? Hay flujos de la tasa de crecimiento. Estamos predicando tasa y flujos (solución) donde VIN/OUT incluyó. Hasta ahora suponíamos que la entrada y salida son infinitos sumideros y fuentes. Para modelar la dinámica de sustrato/crecimiento, el análisis se realiza de manera un poco diferente al análisis de flujo cuantitativo previo. Primero dividimos el tiempo en cortes  $t$ . En cada punto de tiempo  $t$ , usamos FBA para predecir la captación de sustrato celular ( $S_u$ ) y el crecimiento ( $g$ ) durante el intervalo  $t$ . El QSSA significa que estas predicciones son constantes sobre  $t$ . Luego, integramos para obtener la biomasa ( $B$ ) y la concentración de sustrato ( $S_c$ ) en el siguiente punto de tiempo  $t + t$ . Por lo tanto, el nuevo VIN se calcula cada vez en función de los puntos  $t$  de tiempo intermedio. Así podemos predecir la tasa de crecimiento y la captación de glucosa y acetato (nutrientes disponibles en el ambiente). El análisis de cuatro pasos es:

1. La concentración en el tiempo viene dada por la concentración de sustrato de la última etapa más cualquier sustrato adicional proporcionado al cultivo celular por un flujo de entrada, tal como en un lote alimentado.
2. La concentración de sustrato se escala por tiempo y biomasa ( $X$ ) para determinar la disponibilidad de sustrato para las células. Esto puede exceder la tasa máxima de captación de las células o ser menor que ese número.
3. Utilice el modelo de balance de flujo para evaluar la tasa de absorción real del sustrato, que puede ser mayor o menor que el sustrato disponible según lo determinado por la etapa 2.
4. La concentración para el siguiente paso de tiempo se calcula integrando las ecuaciones diferenciales estándar:

$$\frac{d}{dt} \frac{dB}{dt} = g B \quad \text{fila derecha } B = B_{t=0} e^{gt} \quad \text{nonúmero}$$

$$\frac{dS_c}{dt} = -S_u B \rightarrow S_c = S_{c_0} \frac{X}{g} (e^{gt} - 1)$$

El trabajo adicional de Varma et al. (1994) especifica la tasa de captación de glucosa a priori [17]. Las simulaciones del modelo funcionan para predecir cambios dependientes del tiempo en el crecimiento, la absorción de oxígeno y la secreción de acetato. Este modelo inverso traza las tasas de absorción versus el crecimiento, mientras que aún logra resultados comparables *in vivo* e *in silico*. Los investigadores utilizaron modelos cuasi de estado estacionario para predecir los perfiles dependientes del tiempo de crecimiento celular y concentraciones de metabolitos en cultivos discontinuos de *E. coli* que tenían un suministro inicial limitado de glucosa (izquierda) o un suministro continuo lento de glucosa (diagrama derecho). Un gran ajuste es evidente.

Los diagramas anteriores muestran los resultados de las predicciones del modelo (líneas continuas) y lo comparan con los resultados experimentales (puntos individuales). Así, en *E. coli*, las predicciones cuasiestables son impresionantemente precisas incluso con un modelo que no tiene en cuenta ningún cambio en los niveles de expresión enzimática a lo largo del tiempo. Sin

embargo, este modelo no sería adecuado para describir comportamientos que se sabe que implican regulación génica. Por ejemplo, si las células hubieran sido cultivadas en medio de media glucosa/mitad lactosa, el modelo no habría podido predecir el cambio en el consumo de una fuente de carbono a otra. (Esto ocurre experimentalmente cuando *E. coli* activa vías alternas de utilización de carbono solo en ausencia de glucosa).

### Regulación vía lógica booleana

Hay una serie de niveles de regulación a través de los cuales se controla el flujo metabólico en los niveles metabolitos, transcripcionales, traduccionales, postraduccionales. Los errores asociados a FBA pueden explicarse por la incorporación de información reguladora de genes en los modelos. Una forma de hacerlo es la lógica booleana. La siguiente tabla describe si los genes para enzimas asociadas están activados o apagados en presencia de ciertos nutrientes (un ejemplo de incorporación de las preferencias de *E. coli* mencionadas anteriormente):

|                            |                             |
|----------------------------|-----------------------------|
| ON<br>sin glucosa (0)      | EN<br>acetato presente (1)  |
| EN<br>glucosa presente (1) | OFF<br>acetato presente (1) |

Por lo tanto, se puede pensar que el siguiente paso a dar es incorporar este hecho a los modelos. Por ejemplo, si tenemos glucosa en el ambiente, los genes relacionados con el procesamiento de acetato están apagados y por lo tanto ausentes de la matriz S que ahora se vuelve dinámica como resultado de la incorporación de la regulación en nuestro modelo. Al final, nuestro modelo no es cuantitativo. La regulación básica describe entonces que si una enzima procesadora de nutrientes está encendida, la otra está apagada. Básicamente se trata de un montón de lógica booleana, basada en la presencia de enzimas, metabolitos, genes, etc. Estas suposiciones de estilo booleano se utilizan luego en cada pequeño cambio en el tiempo dt para evaluar la tasa de crecimiento, los flujos y tales variables. Entonces, dados los flujos predichos, el VIN, el VOUT y los estados del sistema, se puede usar la lógica para apagar y encender genes, efectivamente una S por t. Podemos comenzar a armar todos los análisis anteriores y llegar a un enfoque general en el modelado metabólico. Podemos decir que si la glucólisis está encendida, entonces la gluconeogénesis debe estar apagada.

El primer intento de incluir la regulación en un modelo de FBA fue publicado por Covert, Schilling y Palsson en 1901 [7]. Los investigadores incorporaron un conjunto de eventos reguladores transcripcionales conocidos en su análisis de una red reguladora metabólica al aproximar la regulación génica como un proceso booleano. Se dijo que se producía o no una reacción dependiendo de la presencia tanto de la enzima como del sustrato (s): si la enzima que cataliza la reacción (E) no se expresa o un sustrato (A) no está disponible, el flujo de reacción será cero:

$$rxn = SI(A) \text{ Y } (E)$$

La lógica booleana similar determinó si las enzimas se expresaban o no, dependiendo de los genes expresados actualmente y las condiciones ambientales actuales. Por ejemplo, la transcripción de la enzima (E) ocurre solo si el gen apropiado (G) está disponible para la transcripción y no está presente ningún represor (B):

$$trans = SI(G) \text{ Y NO(B)}$$

Los autores utilizaron estos principios para diseñar una red booleana que introduce el estado actual de todos los genes relevantes (encendido o apagado) y el estado actual de todos los metabolitos (presentes o no presentes), y emite un vector binario que contiene el nuevo estado de cada uno de estos genes y metabolitos. Las reglas de la red booleana se construyeron a partir de eventos regulatorios celulares determinados experimentalmente. El tratamiento de las reacciones y las concentraciones de enzimas/metabolitos como variables binarias no permite el análisis cuantitativo, pero este método puede predecir cambios cualitativos en los flujos metabólicos cuando se fusionan con FBA. Siempre que una enzima está ausente, la columna correspondiente se retira de la matriz de reacción de FBA, como se describió anteriormente para la predicción del fenotipo knockout. Esto lleva a un proceso iterativo:

1. Dados los estados iniciales de todos los genes y metabolitos, calcular los nuevos estados usando la red booleana;

2. realizar FBA con columnas apropiadas eliminadas de la matriz, con base en los estados de las enzimas, para determinar las nuevas concentraciones de metabolitos;
3. repetir el cálculo de la red booleana con las nuevas concentraciones de metabolitos; etc. El modelo anterior no es cuantitativo, sino más bien una simulación pura de activar y desactivar genes en cualquier momento en particular instante.

En algunas reacciones metabólicas, existen reglas sobre permitir que el organismo cambie las fuentes de carbono (C1, C2).

Una aplicación de este método del estudio de Covert et al. [7] fue simular el desplazamiento diáuxico, un cambio de metabolizar una fuente de carbono preferida a otra fuente de carbono cuando la fuente preferida no está disponible. El proceso modelado incluye dos productos génicos, una proteína reguladora RpC1, que detecta (es activada por) el Carbono 1, y una proteína de transporte Tc2, que transporta el Carbono 2. Si RpC1 es activado por el Carbono 1, Tc2 no se transcribirá, ya que la célula utiliza preferentemente el Carbono 1 como fuente de carbono. Si el Carbono 1 no está disponible, la célula cambiará a vías metabólicas basadas en Carbono 2 y activará la expresión de Tc2.

Los booleanos pueden representar esta información:

$$\text{RpC1} = \text{IF}(\text{Carbon1}) \quad \text{Tc2} = \text{SI NO}(\text{RPC1})$$

Covert et al. encontraron que este enfoque dio predicciones sobre el metabolismo que coincidieron con los resultados del cambio diáuxico inducido experimentalmente. Este desplazamiento diáuxico está bien modelado por el análisis in silico ver figura anterior. En el segmento A, C1 se agota como nutriente y hay crecimiento. En el segmento B, no hay crecimiento ya que C1 se ha agotado y aún no se elaboran las enzimas procesadoras C2, ya que los genes no se han activado (o están en proceso), de ahí el retraso de la cantidad constante de biomasa. En el segmento C, las enzimas para C2 se encendieron y la biomasa aumenta a medida que el crecimiento continúa con una nueva fuente de nutrientes. Por lo tanto, si no hay C1, C2 se agota. A medida que C1 se agota, el organismo desplaza la actividad metabólica a través de la regulación genética y comienza a tomar C2. Regulación predice diauxie, el uso de C1 antes de C2. Sin regulación, el sistema crecería tanto en C1 como en C2 juntos para obtener un máximo de biomasa.

Hasta el momento hemos discutido el uso de este enfoque combinado de la red FBA-Booleana para modelar la regulación a nivel transcripcional/traduccional, y también funcionará para otros tipos de regulación. La principal limitación es para formas lentas de regulación, ya que este método supone que los pasos regulatorios se completan dentro de un solo intervalo de tiempo (porque el cálculo booleano se realiza en cada paso de tiempo de FBA y no toma en cuenta estados previos del sistema). Esto está bien para cualquier forma de regulación que actúe al menos tan rápido como la transcripción/traducción. Por ejemplo, la fosforilación de enzimas (un proceso de activación enzimática) es muy rápida y puede modelarse incluyendo la presencia de una enzima fosforilasa en la red booleana.

Sin embargo, la regulación que ocurre en escalas de tiempo más largas, como el secuestro de ARNm, no es tomada en cuenta por este modelo. Este enfoque también tiene un problema fundamental en el sentido de que no permite introducir mediciones experimentales reales de los niveles de expresión génica en momentos relevantes.

No necesitamos nuestras simulaciones para predecir artificialmente si ciertos genes están activados o apagados. Los datos de expresión de microarrays nos permiten determinar qué genes se están expresando, y esta información se puede incorporar a nuestros modelos.

## Acoplamiento de la expresión génica con el metabolismo

En la práctica, no necesitamos modelar artificialmente los niveles de genes, podemos medirlos. Como se discutió anteriormente, es posible medir los niveles de expresión de todos los ARNm en una muestra dada. Dado que los datos de expresión de ARNm se correlacionan con los datos de expresión de proteínas, sería extremadamente útil incorporarlos en el FBA. Por lo general, los datos de experimentos de microarrays se agrupan, y se plantea la hipótesis de que los genes desconocidos tienen una función similar a la función de aquellos genes conocidos con los que se agrupan. Este análisis puede ser defectuoso, sin embargo, ya que los genes con acciones similares pueden no siempre agruparse. La incorporación de datos de expresión de microarrays en FBA podría permitir un método alternativo de interpretación de los datos. Aquí surge una pregunta, ¿cuál es la relación entre el nivel génico y el flujo a través de una reacción?

Si la reacción  $A \rightarrow B$  es catalizada por una enzima. Si hay mucha A presente, el aumento de la expresión del gen para la enzima provoca un aumento de la velocidad de reacción. De lo contrario, aumentar el nivel de expresión génica no aumentará la velocidad de reacción. Sin embargo, la concentración enzimática puede tratarse como una restricción sobre el flujo máximo posible, dado que el sustrato también tiene un límite fisiológico razonable.

El siguiente paso, entonces, es relacionar el nivel de expresión de ARNm con la concentración de enzima. Esto es más difícil, ya que las células tienen una serie de mecanismos reguladores para controlar las concentraciones de proteínas independientemente de las concentraciones de ARNm. Por ejemplo, las proteínas traducidas pueden requerir una etapa de activación adicional (por ejemplo, fosforilación), cada molécula de ARNm puede traducirse en un número variable de proteínas antes de que se degrada (por ejemplo, mediante ARN antisentido), la tasa de traducción del ARNm a la proteína puede ser más lenta que los intervalos de tiempo considerados. En cada etapa de FBA, y la tasa de degradación de proteínas también puede ser lenta. A pesar de estas complicaciones, los niveles de expresión de ARNm de los experimentos de micromatrices generalmente se toman como límites superiores en las posibles concentraciones de enzimas en cada punto de tiempo medido. Dada la relación anterior entre la concentración enzimática y el flujo, esto significa que los niveles de expresión de ARNm también son límites superiores en los flujos máximos posibles a través de las reacciones catalizadas por sus proteínas codificadas. La validez de este supuesto aún se está debatiendo, pero ya se ha desempeñado bien en los análisis de FBA y es consistente con evidencia reciente de que las células sí controlan los niveles de enzimas metabólicas principalmente ajustando los niveles de ARNm. (En 1907, el profesor Galagan discutió un estudio de Zaslaver et al. (1904) que encontró que los genes requeridos en una ruta de biosíntesis de aminoácidos se transcriben secuencialmente según sea necesario [2]). Esta es una suposición particularmente útil para incluir datos de expresión de micromatrices en FBA, ya que FBA hace uso de valores de flujo máximo para limitar el cono de equilibrio de flujo.

Colijn et al. abordan la cuestión de la integración algorítmica de datos de expresión y redes metabólicas [3]. Aplican FBA para modelar el flujo máximo a través de cada reacción en una red metabólica. Por ejemplo, si se dispone de datos de micromatrices de un organismo que crece en glucosa y de un organismo que crece en acetato, es probable que se observen diferencias reguladoras significativas entre los dos conjuntos de datos. Vmax nos dice cuál es el máximo que podemos alcanzar. Microarray detecta el nivel de transcripciones, y da un límite superior de Vmax.

Además de predecir vías metabólicas bajo diferentes condiciones ambientales, los experimentos de FBA y microarrays se pueden combinar para predecir el estado de un sistema metabólico bajo diferentes tratamientos farmacológicos. Por ejemplo, varios medicamentos contra la tuberculosis se dirigen a la biosíntesis de ácido micológico. El ácido micológico es un constituyente principal de la pared celular. En un artículo de 1904 de Boshode et al., los investigadores probaron 75 fármacos, combinaciones de fármacos y condiciones de crecimiento para

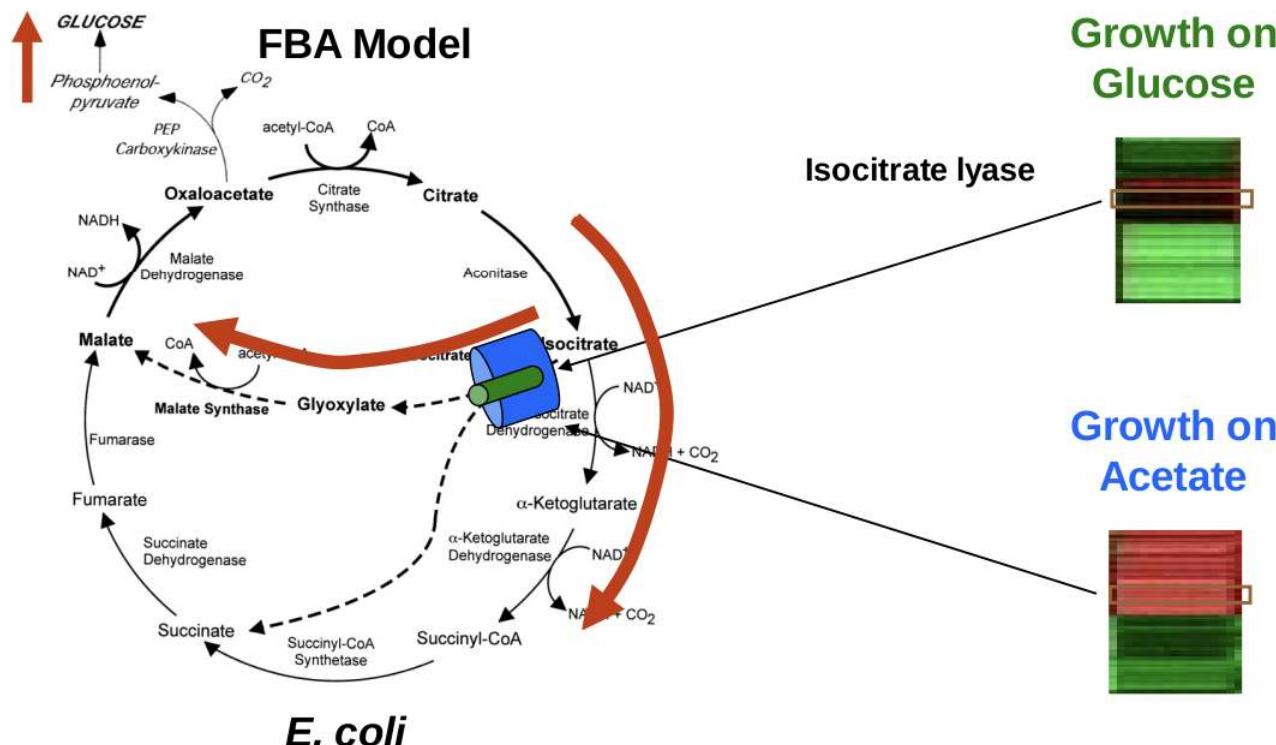


Figura 23.6: Modelo de Coljin et. al [3]

Cortesía de los autores. Licencia: CC BY.

Fuente: Coljin, Caroline, et al. "Interpretación de datos de expresión con modelos de flujo metabólico: predicción de Mycobacterium

Tubercluosis Producción de ácido micológico." *PLoS Biología Computacional* 5, núm. 8 (2009): e1000489

ver qué efecto tuvieron los diferentes tratamientos sobre la síntesis de ácido micónico [9]. En 1905, Raman et al. publicaron un modelo FBA de biosíntesis de ácido micónico, consistente en 197 metabolitos y 219 reacciones [13].

El flujo básico de la predicción fue tomar un valor de expresión control y un valor de expresión de tratamiento para un conjunto particular de genes, luego alimentar esta información al FBA y medir el efecto final sobre el tratamiento sobre la producción de ácido micónico. Para examinar los inhibidores y potenciadores predichos, examinaron la significación, que examina si el efecto se debe al ruido, y la especificidad, que examina si el efecto se debe al ácido micónico o a la supresión/mejora global del metabolismo. Los resultados fueron bastante aleatorios. Varios inhibidores conocidos del ácido micónico fueron identificados por el FBA. También se encontraron resultados interesantes entre fármacos no conocidos específicamente para inhibir la síntesis de ácido micónico. Se predijeron 4 nuevos inhibidores y 2 nuevos potenciadores de la síntesis de ácido micónico. Un fármaco en particular, Triclosán, parece ser un potenciador según el modelo FBA, mientras que actualmente se le conoce como inhibidor. Sería interesante estudiar más a fondo esta droga en particular. Las pruebas experimentales y la validación están actualmente en curso.

La agrupación también puede ser ineficaz para identificar la función de diversos tratamientos. Los inhibidores predichos y los potenciadores predichos de la síntesis de ácido micónico no se agrupan entre sí. Además, no se requiere un conjunto de entrenamiento etiquetado para la clasificación algorítmica basada en la FBA, mientras que es necesario para los algoritmos de agrupamiento supervisados.

### Predicción de la fuente de nutrientes

Ahora, tenemos la idea de predecir la fuente de nutrientes que un organismo puede estar usando en un ambiente, observando los datos de expresión y buscando la expresión génica asociada al procesamiento de nutrientes. Esto es más fácil, ya que no podemos entrar al medio ambiente y medir todos los niveles químicos, pero podemos obtener datos de expresión con bastante facilidad. Es decir, tratamos de predecir una fuente de nutrientes a través de predicciones del estado metabólico a partir de datos de expresión,

con base en el supuesto de que es probable que los organismos ajusten el estado metabólico a los nutrientes disponibles. Luego, los nutrientes pueden clasificarse según qué tan bien coinciden con los estados metabólicos.

Al revés también podría funcionar. ¿Puedo predecir un nutriente dado un estado? Tales predicciones podrían ser útiles para determinar los requerimientos nutrimetales de un organismo con un entorno natural desconocido, o para determinar cómo un organismo cambia su ambiente. (La TB, por ejemplo, es capaz de vivir dentro del ambiente de un fagolisosoma macrófago, presumiblemente alterando las condiciones ambientales en el

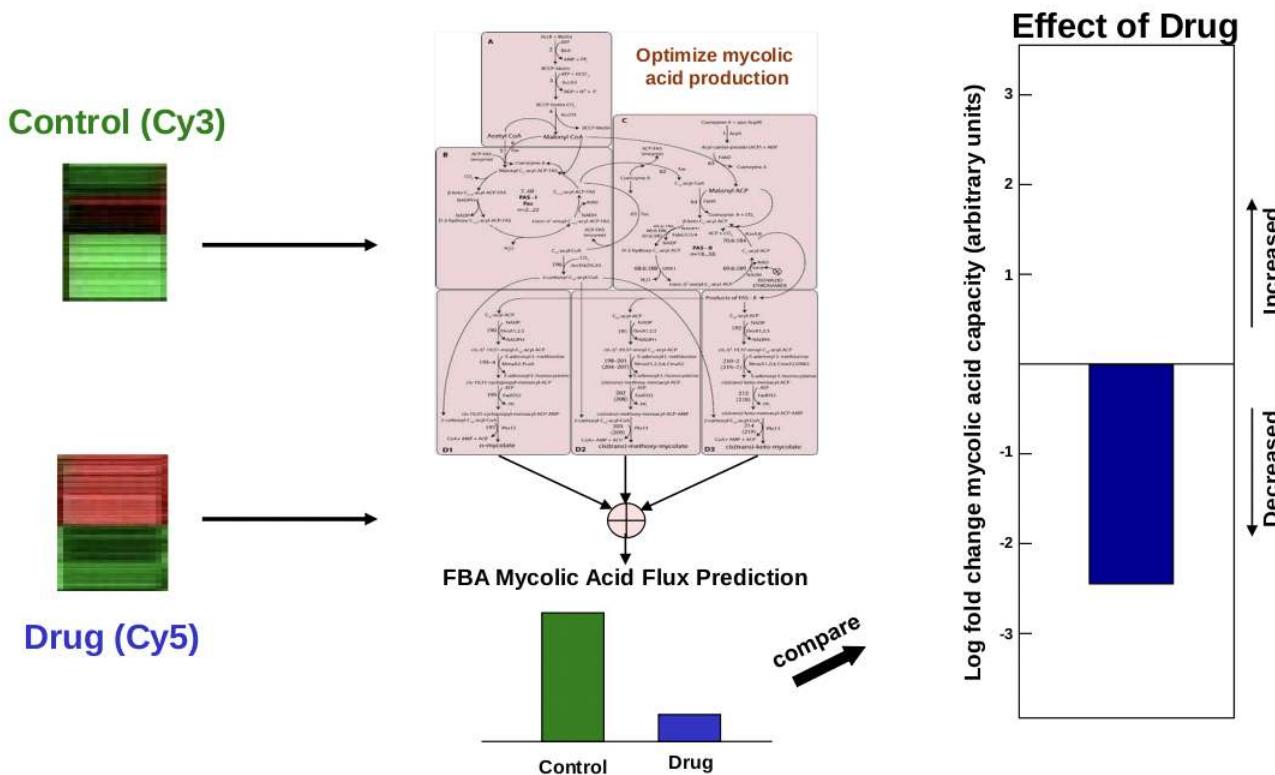


Figura 23.7: Flujo básico en la predicción del estado de un sistema metabólico bajo tratamientos farmacológicos varing fagolisosoma y previendo su maduración.)

Cortesía de los autores. Licencia: CC BY.

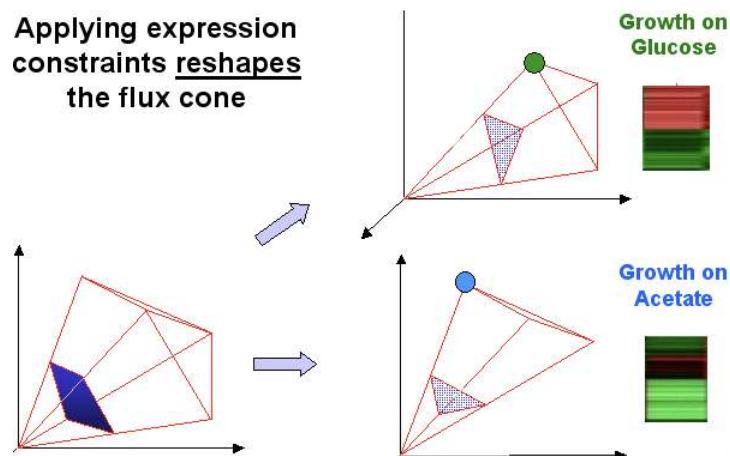
Fuente: Coljin, Caroline, et al. “Interpretación de datos de expresión con modelos de flujo metabólico: predicción

Producción de ácido micólico por *Mycobacterium Tuberculosis*.” *PLoS Biología Computacional* 5, núm. 8 (2009): e1000489

Podemos usar FBA para definir un espacio de posibles estados metabólicos y elegir uno. Los pasos básicos son:

- Comience con el cono de flujo máximo (representando el mejor crecimiento con todos los nutrientes disponibles en el ambiente). Encuentre un flujo óptimo para cada nutriente.
- Aplicar conjunto de datos de expresión (aún sin conocer nutriente). Esto le permitirá limitar la forma del cono y averiguar el nutriente, que se representa como uno con la distancia más cercana a la solución óptima.

### Applying expression constraints reshapes the flux cone

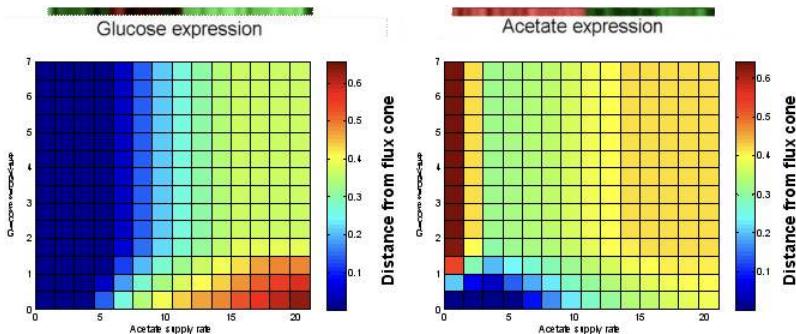


fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 23.8: La aplicación del conjunto de datos de expresión permite restringir la forma del cono.

En la Figura 8, puede ver que el primer cono tiene una serie de óptimos, por lo que se desconoce el nutriente real. Sin embargo, después de aplicar los datos de expresión, el cono es remodelado. Tiene solo un óptimo, el cual se encuentra aún en un espacio factible y con ello debe ser ese nutriente que estás buscando.

Como antes, los niveles de expresión medidos proporcionan restricciones en los flujos de reacción, alterando la forma del cono de equilibrio de flujo (ahora el cono de equilibrio de flujo restringido por expresión). FBA se puede usar para determinar el conjunto óptimo de flujos que maximizan el crecimiento dentro de estas restricciones de expresión, y este conjunto de flujos se puede comparar con patrones de crecimiento óptimos determinados experimentalmente bajo cada condición ambiental de interés. La diferencia entre el estado calculado del organismo y el estado óptimo bajo cada condición es una medida de cuán subóptimo sería el estado metabólico actual del organismo si de hecho estuviera creciendo bajo esa condición.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 23.9: Resultados del experimento de predicción de fuentes nutritenciales.

Los datos de expresión del crecimiento y el metabolismo se pueden aplicar para predecir la fuente de carbono que se está utilizando. Por ejemplo, considere el producto nutritivo de *E. coli*. Podemos simular este sistema para glucosa versus acetato. El color indica la distancia desde el cono de flujo restringido hasta la solución de flujo óptimo para ese combo de nutrientes (mismo procedimiento descrito anteriormente). Entonces, se pueden clasificar múltiples nutrientes, priorizados de acuerdo con los datos de expresión. Datos inéditos de Desmond Lun y Aaron Brandes proporcionan un ejemplo de este enfoque.

Utilizaron FBA para predecir en qué fuente nutritiva estaban creciendo los cultivos de *E. coli*, con base en datos de expresión génica. Compararon los flujos óptimos conocidos (el punto óptimo en el espacio de flujo) para cada condición nutritiva con los valores de flujo óptimos permitidos dentro del cono de equilibrio de flujo restringido por expresión. Aquellas condiciones nutritivas

con flujos óptimos que permanecieron dentro (o más cercanos a) el cono de expresión restringida fueron las posibilidades más probables para el ambiente real del cultivo.

Los resultados del experimento se muestran en la Figura 9, donde cada cuadrado en las matrices de resultados se colorea con base en la distancia entre los flujos óptimos para esa condición nutritiva y los flujos óptimos calculados con base en los datos de expresión. Los valores rojos indican grandes distancias desde el cono de flujo con restricción de expresión y los valores azules indican distancias cortas desde el cono. En los experimentos de glucosa-acetato, por ejemplo, los resultados del experimento de la izquierda indican que las condiciones bajas de acetato son las más probables (y la glucosa fue el nutriente en el cultivo) y los resultados del experimento de la derecha indican que las condiciones bajas en glucosa/acetato medio son las más probables (y el acetato fue el nutriente en el cultivo). Cuando se consideraron 6 nutrientes posibles, el modelo siempre predijo el correcto, y cuando se consideraron 18 nutrientes posibles, el correcto siempre fue una de las 4 mejores predicciones de clasificación. Estos resultados sugieren que es posible utilizar datos de expresión y modelos FBA para predecir condiciones ambientales a partir de información sobre el estado metabólico de un organismo.

Esto es importante porque la TB utiliza ácidos grasos en macrófagos en los sistemas inmunitarios. No sabemos cuáles son exactamente los que se utilizan. Podemos averiguar lo que la TB ve en su entorno como fuente de alimento y factor de proliferación analizando qué genes de procesamiento de nutrientes relacionados se encienden en las fases de crecimiento y similares. De esta manera podemos averiguar los nutrientes que necesita para crecer, permitiendo una forma potencial de matarlo al no suministrarle dichos nutrientes o noqueando esos genes en particular.

Es más fácil obtener datos de expresión para ver la actividad del flujo que ver qué se está agotando en el ambiente analizando la química en un nivel tan pequeño. Además, es posible que no podamos cultivar algunas bacterias en el laboratorio, pero podemos resolver el problema obteniendo los datos de expresión de las bacterias que crecen en un ambiente natural y luego ver qué está usando para crecer. Entonces, podemos agregarlo al medio de laboratorio para cultivar las bacterias con éxito.

---

This page titled [23.4: Aplicaciones](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [23.4: Applications](#) by [Manolis Kellis et al.](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://ocw.mit.edu/courses/6-047-computational-biology-fall-2015/>.

## 23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía

- Becker, S. A. y B. O. Palsson (1908). Las redes metabólicas específicas del contexto son consistentes con los experimentos. PLoS Biología Computacional 4 (5): e1000082.

— Si la expresión génica es inferior a algún umbral, apague el gen en el modelo.

- Shlomi, T., M. N. Cabili, et al. (1908). Predicción basada en red del metabolismo específico de tejido humano.

Nat Biotech 26 (9): 1003-1010.

- Problema de optimización anidada.

- Primero, FBA estándar

- Segundo, maximizar el número de enzimas cuya actividad de flujo predicha es *consistente con su nivel de expresión medido*

---

23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

- 23.5: Further Reading, Tools and Techniques, Bibliography has no license indicated.

## 23.6: Herramientas y Techniques

- Kegg
- BioCyc
- Explorador de vías ([pahtwayexplorer.genome.tugraz.at](http://pathwayexplorer.genome.tugraz.at))
- Grupo Palssons en la UCSD ([gcrg.ucsd.edu/](http://gcrg.ucsd.edu/))
- [www.systems-biology.org](http://www.systems-biology.org)
- Base de datos de biomodels ([www.ebi.ac.uk/biomodels/](http://www.ebi.ac.uk/biomodels/))
- Base de datos modelo JWS ([jjj.biochem.sun.ac.za/database/index.html](http://jjj.biochem.sun.ac.za/database/index.html))

---

23.6: Herramientas y Techniques is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [23.6: Tools and Techniques](#) has no license indicated.

## Bibliografía

[1]

[2] Zaslaver A, Mayo AE, Rosenberg R, Bashkin P, Sberro H, Tsalyuk M, Surette MG y Alon U. Programa de transcripción Just-in-Time en vías metabólicas. *Nat. Gen.*, 36:486 —491, 2004. 379

[3] Caroline Coljin. Interpretación de datos de expresión con modelos de flujo metabólico: Predicción de la producción de ácido micólico de *Mycobacterium tuberculosis*. *PLoS Biología Computacional*, 5 (8), ago 2009.

[4] Price N. D., Reed J. L., Papin J.A, Famili I. y Palsson B.O. Análisis de capacidades metabólicas usando descomposición de valores singulares de matrices de vías extremas. *Biophys J.*, 84 (2) :794—804, feb 2003.

[5] Gasteiger E., Gattiker A., Hoogland C. y Divanyi I., Appel R.D., y Bairoch A. Expasy: El servidor proteómico para el conocimiento y análisis de proteínas en profundidad. *Ácidos nucleicos Res.*, 31 (13) :3784—3788.

[6] J.S. Edwards, R. U. Ibarra, y B.O. Palsson. Las predicciones in silico de las capacidades metabólicas de *e coli* son consistentes con datos experimentales. *Nat Biotechnology*, 19:125 —130, 2001.

[7] Covert M et al. Regulación de la expresión génica en modelos de equilibrio de flujo del metabolismo. *Revista de Biología Teórica*, 213:73 —88, nov 2001.

[8] J. Forster, I. Famili, B.O. Palsson, y J. Nielsen. Evaluación a gran escala de delecciones de genes in silico en *saccharomyces cerevisiae*. *OMICS*, 7 (2) :193—202, 2003. PMID: 14506848.

[9] Boshoff H.I., Myers T.G., Copp B.R., McNeil M.R., Wilson M.A., y Bary C.E. La respuesta transcripcional de *Mycobacterium tuberculosis* a los inhibidores del metabolismo: nuevos conocimientos sobre los mecanismos de acción de los fármacos. *J Biol Chem*, 279:40174 —40184, sep de 2004.

[10] Holmberg. Sobre la identificabilidad práctica de modelos de crecimiento microbiano incorporando no linealidades de tipo michaelis-menten. *Biociencias matemáticas*, 62 (1) :23—43, 1982.

[11] Edwards J.S. y Palsson B.O. volumen 97, páginas 5528—5533. Actas de la Academia Nacional de Ciencias de los Estados Unidos de América, mayo de 2000. PMC25862.

[12] Edwards J.S., Covert M., y Palsson B. Modelado metabólico de microbios: el enfoque del balance de flujo. *Microbiología Ambiental*, 4 (3) :133—140, 2002.

[13] Raman Karthik, Preethi Rajagopalan y Nagasuma Chandra. Análisis de balance de flujo de la vía del ácido micólico: Dianas para fármacos antituberculosos. *PLoS Biología Computacional*, 1, Oct 2005.

[14] Kanehisa M., Goto S., Kawashima S., y Nakaya. De la genómica a la genómica química: nuevos desarrollos en kegg. *Ácidos nucleicos Res.*, 34, 2006.

[15] Jamshidi N. y Palsson B. Investigando las capacidades metabólicas de *Mycobacterium tuberculosis h37rv* usando la cepa in silico inj661 y proponiendo dianas de fármacos alternativos. *BMC Systems Biology*, 26, 2007.

[16] Caspi R., Foerster H., Fulcher C.A., Kaipa P., Krummenacker M., Latendresse M., Paley S., Rhee S.Y., Shearer A.G., Tissier C., Walk T.C. ZhangP., y Karp P. La base de datos metacícos de vías metabólicas y enzimas y la colección biocítica de bases de datos de trayectorias/genoma. *Ácidos nucleicos Res*, 36 (Suppl), 2008.

[17] A. Varma y B. O. Palsson. Los modelos estequiométricos de equilibrio de flujo predicen cuantitativamente el crecimiento y la secreción de subproductos metabólicos en *Escherichia coli* de tipo silvestre w3110. *Microbiología Aplicada y Ambiental*, 60:3724 —3731, Oct 1994.

---

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 24: El Proyecto Encode- Experimentación Sistemática y Genómica Integrativa

24.1: Introducción

24.2: Técnicas Experimentales

24.3: Técnicas Computacionales

24.4: Direcciones actuales de investigación

24.5: Lectura adicional, Herramientas y técnicas, Bibliografía

24.6: Herramientas y Técnicas

Bibliografía

Sección 7: ¿Qué hemos aprendido?

---

24: El Proyecto Encode- Experimentación Sistemática y Genómica Integrativa is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 24.1: Introducción

El genoma humano fue secuenciado en 2003, un paso importante en la comprensión del plan de vida. Sin embargo, antes de que esta información pueda ser completamente utilizada, se debe determinar la ubicación, identidad y función de todos los genes que codifican proteínas y no codificantes de proteínas. Además, el genoma humano tiene muchos otros elementos funcionales, que van desde promotores, secuencias reguladoras y otros factores que determinan la estructura de la cromatina. Estos también deben estar determinados para comprender a fondo el genoma humano.

El proyecto ENCODE (Encyclopedia of DNA Elements) tiene como objetivo resolver estos problemas delineando todos los elementos funcionales del genoma humano. Para lograr este objetivo, se formó un consorcio para orientar el proyecto. El consorcio tuvo como objetivo avanzar y desarrollar tecnologías para anotar el genoma humano con mayor precisión, integridad y costo-efectividad, junto con más estandarización. También tuvieron como objetivo desarrollar una serie de técnicas computacionales para analizar y analizar los datos obtenidos.

Para lograr este objetivo, se puso en marcha un proyecto piloto. El proyecto piloto ENCODE tuvo como objetivo estudiar 1% del genoma humano en profundidad, aproximadamente de 2003 a 2007. De 2007 a 2012, el proyecto ENCODE se incrementó para anotar todo el genoma. Finalmente, a partir de 2012, el proyecto ENCODE apunta a mayores incrementos en todas las dimensiones: secuenciación más profunda, más ensayos, más factores de transcripción, etc.

En este capítulo se describirán algunas de las técnicas experimentales y computacionales utilizadas en el proyecto ENCODE.

---

24.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 24.1: Introduction has no license indicated.

## 24.2: Técnicas Experimentales

El proyecto ENCODE utilizó una amplia gama de técnicas experimentales, que van desde RNA-seq, Cage-seq, Exon Arrays, Maine-seq, Chromatin Chip-seq, DNase-seq y muchas más.

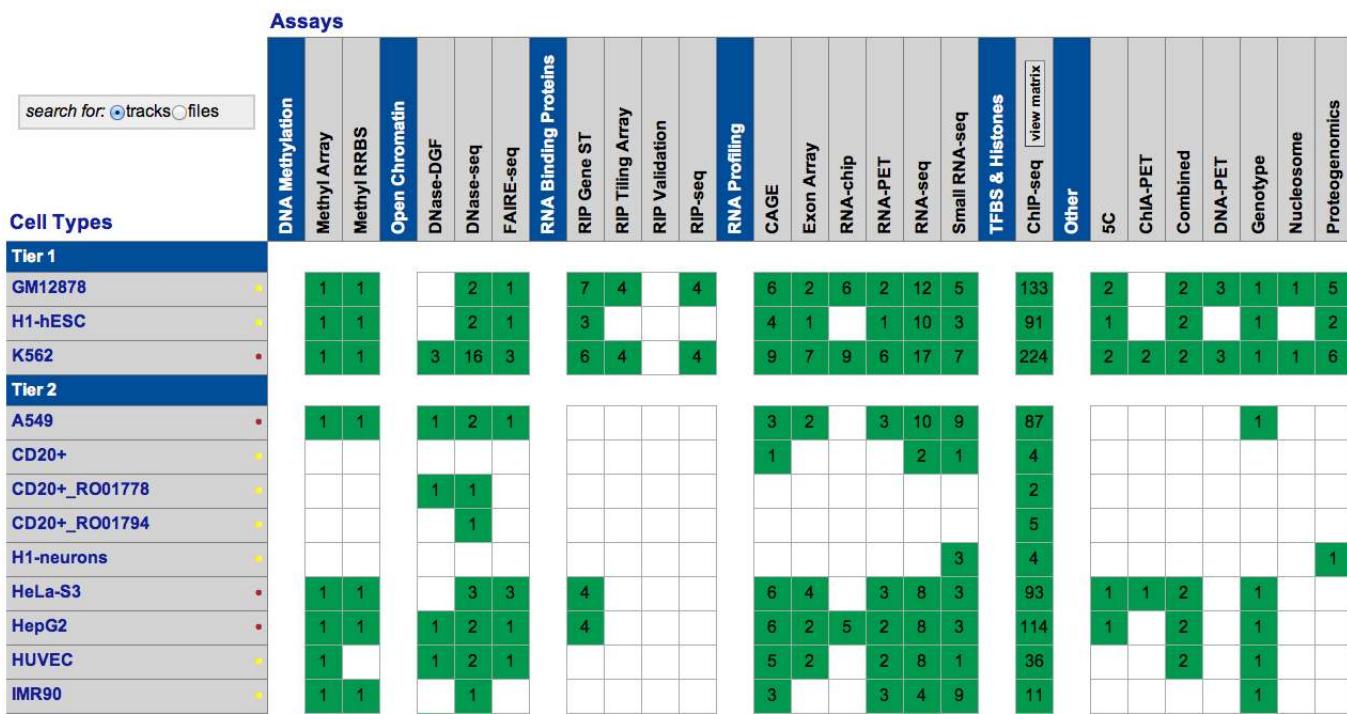


Figura 24.1: Instantánea de la matriz experimental del proyecto ENCODE.

Proyecto ENCODE. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Una de las técnicas más importantes utilizadas fue Chip-seq (inmunoprecipitación de cromatina seguida de secuenciación). El primer paso en un experimento ChIP es apuntar a fragmentos de ADN asociados con una proteína específica. Esto se hace mediante el uso de un anti-cuerpo que se dirige a la proteína específica y se utiliza para inmunoprecipitar el complejo ADN-proteína. El paso final es ensayar el ADN. Esto determinará las secuencias unidas a las proteínas.

Chip-seq tiene varias ventajas sobre las técnicas anteriores (por ejemplo, Chip-chip). Por ejemplo, Chip-seq tiene resolución de un solo nucleótido y su alineabilidad aumenta con la longitud de lectura. Sin embargo, Chip-seq tiene varias desventajas. Los errores de secuenciación tienden a aumentar sustancialmente cerca del final de las lecturas. Además, con un bajo número de lecturas, la sensibilidad y especificidad tienden a disminuir al detectar regiones enriquecidas. Ambos problemas surgen al procesar los datos y muchas de las técnicas computacionales buscan rectificarlo.

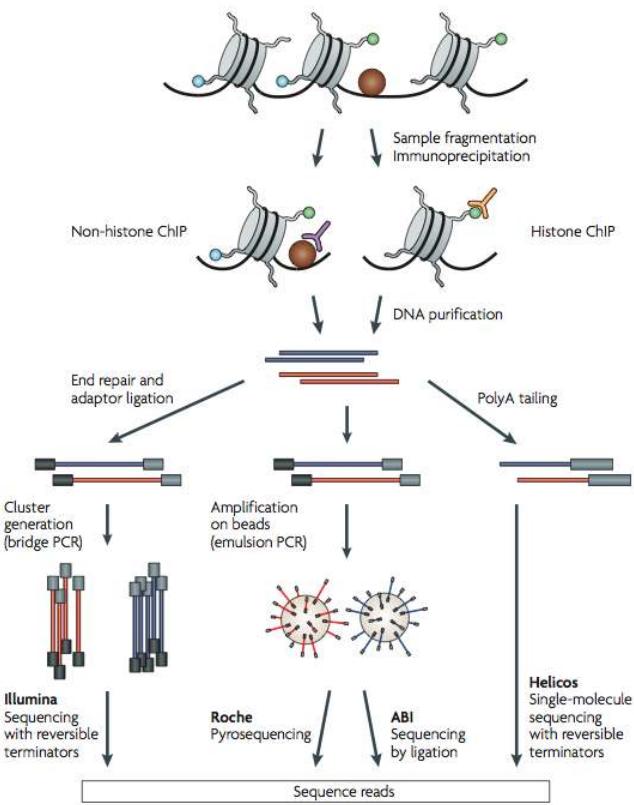


Figura 24.2: Visión general de Chip-seq [3]

Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Park, Peter J. "Chip-seq: Ventajas y Desafíos de una Tecnología de Madurado". *Nature Reviews Genetics* 10, núm. 10 (2009): 669-80

---

24.2: Técnicas Experimentales is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 24.2: Experimental Techniques has no license indicated.

## 24.3: Técnicas Computacionales

Esta sección se centrará en las técnicas de procesamiento de datos brutos del proyecto ENCODE. Antes de que se puedan analizar los datos ENCODE (por ejemplo, para el descubrimiento de motivos, análisis de coasociación, agregación de señales sobre elementos, etc.), los datos sin procesar deben procesarse.

Incluso antes de que se procesen los datos, se aplica algún control de calidad. El control de calidad es necesario por varias razones. Incluso sin anti-cuerpos, las lecturas no están uniformemente dispersas. Las razones biológicas incluyen fragmentación no uniforme del genoma, regiones de cromatina abiertas que se fragmentan más fácilmente y secuencias repetitivas sobrecolapsadas en genomas ensamblados. El proyecto ENCODE corrigió estos sesgos de varias maneras. Se eliminaron porciones del ADN antes de la etapa ChIP, eliminando grandes porciones de datos no deseados. También se realizaron experimentos de control sin el uso de anticuerpos. Finalmente, se utilizaron lecturas de secuencias de ADN de entrada de fragmentos como fondo.

Debido al ruido inherente en el proceso Chip-seq, algunas lecturas serán de menor calidad. Usando una métrica de calidad de lectura, se desecharon lecturas por debajo de un umbral.

Las lecturas más cortas (y en menor medida, lecturas más largas) pueden mapear exactamente a una ubicación (mapeo único), múltiples ubicaciones (mapeo repetitivo) o ninguna ubicación en absoluto (no mapeable) en el genoma. Hay muchas formas potenciales de lidar con el mapeo repetitivo, que van desde la difusión probabilística de la lectura hasta el uso de un enfoque EM. Sin embargo, dado que el proyecto ENCODE pretende ser lo más correcto posible, no asigna lecturas repetitivas a ninguna ubicación.

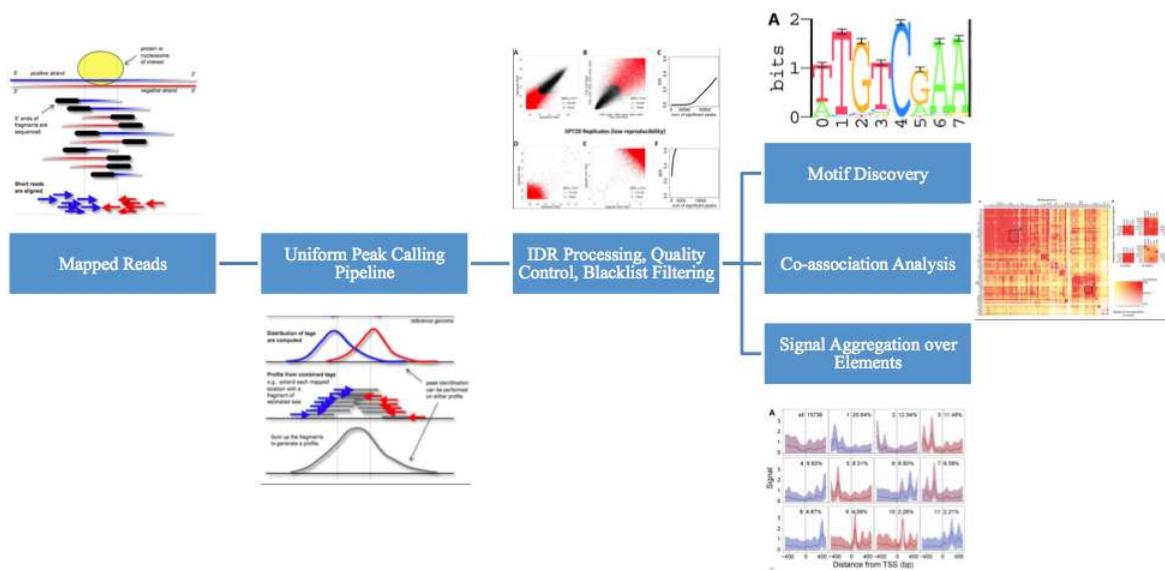


Figura 24.3: Codificar tubería de procesamiento uniforme

Proyecto ENCODE. Todos los derechos reservados. Este contenido está exculeded de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Si una muestra no contiene ADN suficiente y/o si está sobresecuenciada, simplemente secuenciará repetidamente duplicados de PCR de un conjunto restringido de fragmentos de ADN distintos. Esto se conoce como una biblioteca de baja complejidad y no es deseable. Para resolver este problema, se crea un histograma con el número de duplicados y se desechan muestras con una fracción no redundante (NRF) baja.

Chip-seq secuencias aleatorias de un extremo de cada fragmento, por lo que para determinar qué lecturas provienen de qué segmento, típicamente se usa el análisis de correlación cruzada de cadenas [Fig. 04]. Para lograr esto, se calculan las señales de hebra directa e inversa. Entonces, se desplazan secuencialmente el uno hacia el otro. En cada paso, se calcula la correlación. En el desplazamiento de longitud del fragmento  $f$ , los picos de correlación.  $f$  es la longitud a la que se fragmenta el ADN de ChIP.

Mediante análisis adicionales, podemos determinar que debemos tener una correlación cruzada absoluta alta a la longitud del fragmento, y una correlación cruzada de longitud de fragmento alta en relación con la correlación cruzada de longitud de lectura. La RSC (Correlación Relativa de Strand) debe ser mayor a 1.

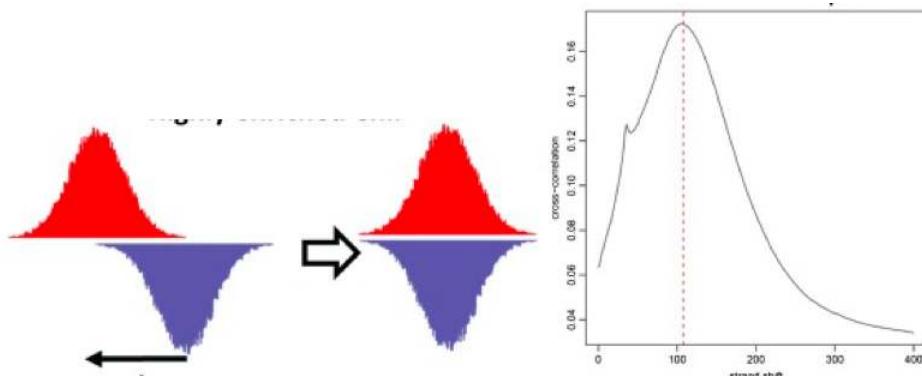


Figura 24.4: Cambio de señales de hebra hacia adelante y hacia atrás, gráfica de correlación cruzada

$$RSC = \frac{CC_{\text{fragment}} - \min(CC)}{CC_{\text{readlength}} - \min(CC)} \quad (24.3.1)$$

Una vez aplicado el control de calidad, los datos se procesan posteriormente para determinar las áreas reales de enriquecimiento. Para lograr esto, el proyecto ENCODE utilizó una versión modificada de peak calling. Existen muchos algoritmos de llamada de picos existentes, pero el proyecto ENCODE utilizó MACS y PeakSeq, ya que son deterministas. Sin embargo, no es posible establecer un valor  $p$  uniforme o una constante de tasa de descubrimiento falso (FDR). El FDR y el valor  $p$  dependen de ChIP y profundidad de secuenciación de entrada, la ubicuidad de unión del factor, y es altamente inestable. Además, diferentes herramientas requieren diferentes valores.

El proyecto ENCODE utiliza réplicas (del mismo experimento) y combina los datos para encontrar resultados más significativos. Las soluciones simples tienen grandes problemas: tomar la unión de los picos mantiene la basura de ambos, la intersección es demasiado estricta y arroja buenos picos, y tomar la suma de los datos no explota la independencia de los conjuntos de datos. En cambio, el proyecto ENCODE utiliza la tasa de descubrimiento independiente (IDR). La idea clave es que los verdaderos picos estarán altamente clasificados en ambas réplicas. Así, para encontrar picos significativos, los picos se consideran en orden de rango, hasta que los rangos ya no se correlacionan.

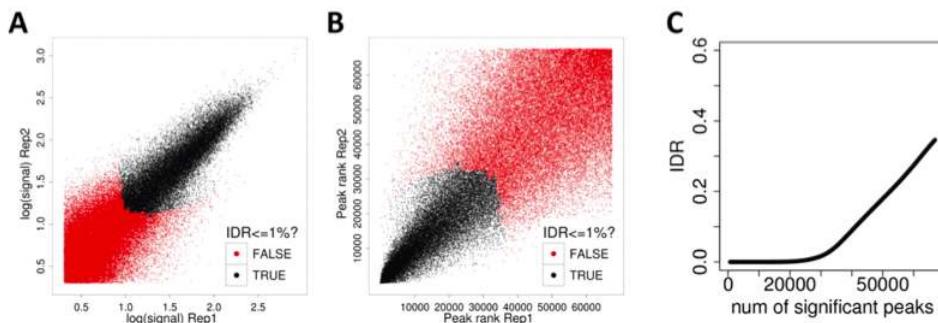


Figura 24.5: IDR para evaluar la reproducibilidad de conjuntos de datos Chip-seq. Los gráficos de dispersión muestran puntuaciones de señal de picos que se superponen en cada par replicado. (A, B) resultados para replicado de alta calidad. (C) IDR estimado para umbrales de rango variables. [1]

El corte podría ser diferente para las dos réplicas y los picos reales incluidos pueden diferir entre réplicas. Se modela como un modelo de mezcla gaussiana, que se puede ajustar a través de un algoritmo similar a EM. El uso de IDR conduce a una mayor consistencia entre las personas que llaman pico. Esto se debe a que la FDR solo se basa en el enriquecimiento sobre la entrada, IDR explota se replica. Además, usando métodos de muestreo, si solo hay una réplica, la tubería IDR aún se puede usar con pseudo-réplicas.

24.3: Técnicas Computacionales is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 24.3: Computational Techniques has no license indicated.

## 24.4: Direcciones actuales de investigación

El proyecto ENCODE sigue en curso. Utilizando técnicas de saturación, creemos que solo hemos descubierto un máximo del 50% de los elementos. Es probable que este número sea menor debido a tipos de células inaccesibles y otros factores. Además, varios tipos de células son extremadamente raros y de difícil acceso, por lo que secuenciar datos de estos tipos de células es otro desafío.

En las fronteras computacionales, el proyecto ENCODE ha producido una enorme cantidad de datos brutos. Similar a cómo la secuencia completa del genoma humano desató una serie de proyectos computacionales, los datos ENCODE pueden ser utilizados para una variedad de proyectos computacionales.

---

24.4: Direcciones actuales de investigación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [24.4: Current Research Directions](#) has no license indicated.

## 24.5: Lectura adicional, Herramientas y técnicas, Bibliografía

El sitio Nature con ENCODE papers está disponible en <http://www.nature.com/encode/>.

El portal oficial ENCODE es <http://encodeproject.org/ENCODE/>.

Para navegar por CODIR datos, visite <http://encodeproject.org/cgi-bin/hgHubConnect>.

Las herramientas de procesamiento de datos para ENCODIR datos están disponibles en <http://encodeproject.org/ENCODE/analysis.html>.

---

24.5: Lectura adicional, Herramientas y técnicas, Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 24.5: Further Reading, Tools and Techniques, Bibliography has no license indicated.

## 24.6: Herramientas y Técnicas

ENCODE minería de datos, [http://genome.ucsc.edu/cgi-bin/hgTab...up=regulation & hgta\\_track=wgenCodeHudsonAlphachipSeq](http://genome.ucsc.edu/cgi-bin/hgTab...up=regulation & hgta_track=wgenCodeHudsonAlphachipSeq)

ENCODE visualización de datos, [http://genome.ucsc.edu/cgi-bin/hgTra...erUser=submit & hgs\\_OtherUserName=kate&hgs\\_OtherUserSessionName=encoDeportalSession](http://genome.ucsc.edu/cgi-bin/hgTra...erUser=submit & hgs_OtherUserName=kate&hgs_OtherUserSessionName=encoDeportalSession)

Software y recursos para la rnalización de datos ENCODE, <http://genome.ucsc.edu/ENCODE/analysisTools.html>

Herramientas de software utilizadas para crear el recurso ENCODE, <http://genome.ucsc.edu/ENCODE/encodeTools.html>

---

24.6: Herramientas y Técnicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 24.6: Tools and Techniques has no license indicated.

## Bibliografía

[1] S G Landt, G K Marinov, A Kundaje, P Kheradpour, F Pauli, S Batzoglou, B E Bernstein, P Bickel, J B Marrón, P Cayting, Y Chen, G DeLvo, C Epstein, K I Fisher-Aloro, G Euskirchen, M Gerstein, J Gertz, A J Hartemink, M M Hoffman, V Iyer, Y L Jung, S Karmakar, M Kellis, P V Kharchenko, Q Li, T Liu, X S Liu, L Ma, A Milosavljevic, R M Myers, P J Parque, M J Pazin, M D Perry, D Raha, T E Reddy, J Rozowsky, N Shores, A Sidow, M Slattery, J A Stamatoyannopoulos, M Y Tolstorukov, K P Blanco, S Xi, P J Farnham, J D Lieb, B J Wold, y M Snyder. Lineamientos y prácticas CHIP-seq de los consorcios ENCODE y MODENCO. *Genome Research*, 22 (9) :1813—1831, 2012.

[2] Philippe Lefran ord cois, Ghia M Euskirchen, Raymond K Auerbach, Joel Rozowsky, Theodore Gibson, Christopher M Yellman, Mark Gerstein y Michael Snyder. Levadura ecente Chip-seq usando secuenciación de ADN multiplex de lectura corta. *BMC Genómica*, 10 (1) :37, 2009.

[3] Parque P J. Chip-seq: ventajas y retos de una tecnología de maduración. Opiniones sobre la naturaleza. *Genética*, 10 (10) :669 —680, octubre de 2009.

---

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## Sección 7: ¿Qué hemos aprendido?

Este capítulo proporciona una visión general del proyecto ENCODE que tiene como objetivo anotar todo el genoma humano. Recoge secuencias de ADN utilizando diversas técnicas experimentales como Chip-seq, RNA-seq y Cage-seq. Después de que se hayan obtenido los datos, es necesario procesarlos antes de intentar el análisis. Los datos pasan por una serie de pasos; control de calidad, llamadas pico, procesamiento IDR y filtrado de listas negras. Una vez que se ha asegurado la precisión de los datos, se pueden realizar otros análisis en forma de descubrimiento de motivos, análisis de coasociación y agregación de señales sobre elementos.

---

Sección 7: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 7: What Have We Learned?](#) has no license indicated.

## CHAPTER OVERVIEW

### 25: Biología Sintética

[25.1: Introducción a la Biología Sintética](#)

[25.2: Direcciones actuales de investigación](#)

[25.3: Herramientas y Técnicas](#)

[25.4: ¿Qué hemos aprendido? , Bibliografía](#)

[Bibliografía](#)

---

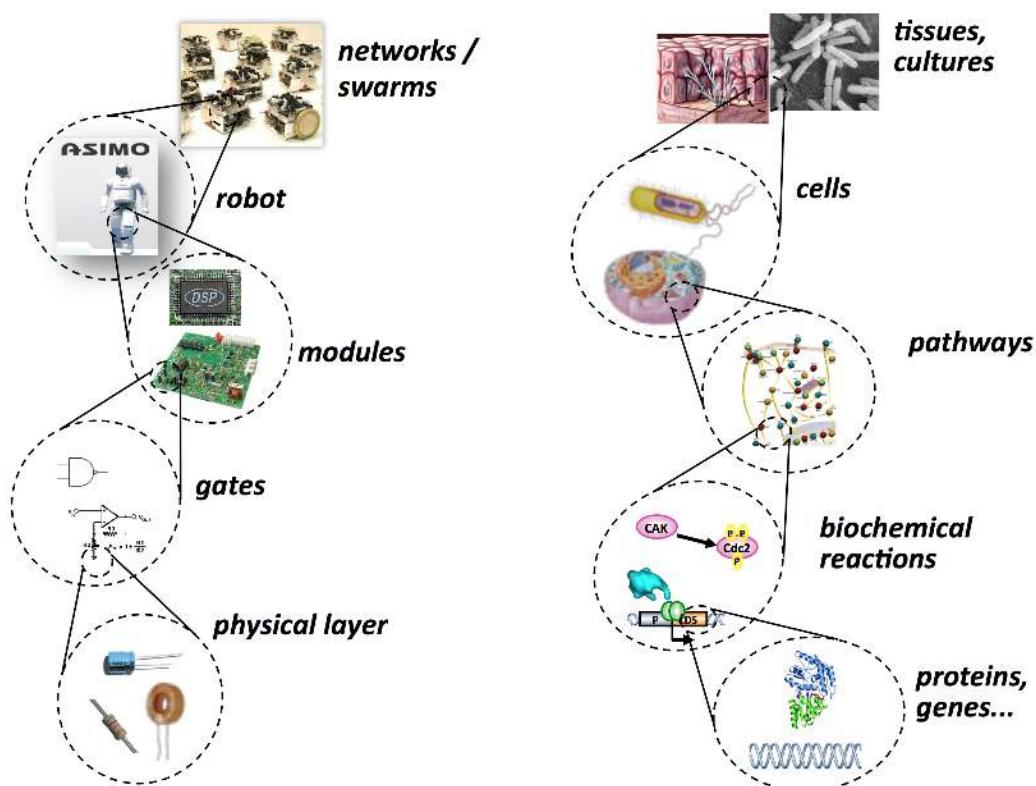
25: Biología Sintética is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 25.1: Introducción a la Biología Sintética

Una célula es como robot en el sentido de que necesita poder percibirla el entorno y el estado interno, realizar cálculos y emitir juicios, y completar una tarea o función. La disciplina emergente de la biología sintética tiene como objetivo hacer que el control de entidades biológicas como las células y las proteínas sea similar al diseño de un robot. La biología sintética combina tecnología, ciencia e ingeniería para construir dispositivos y sistemas biológicos con fines útiles, incluyendo soluciones a problemas mundiales en salud, energía, medio ambiente y seguridad.

La biología sintética involucra todos los niveles de biología, desde el ADN hasta los tejidos. El biólogo sintético tiene como objetivo crear capas de abstracción biológica como las de las computadoras digitales para crear circuitos y programas biológicos de manera eficiente. Uno de los principales objetivos de la biología sintética es el desarrollo de un conjunto estándar y bien definido de herramientas para la construcción de sistemas biológicos que permita que el nivel de abstracción disponible para los ingenieros eléctricos que construyen circuitos complejos esté disponible para los biólogos sintéticos.

La biología sintética es un campo relativamente nuevo. El tamaño y complejidad de los circuitos genéticos sintéticos ha sido hasta ahora pequeño, del orden de seis a once promotores. Los circuitos genéticos sintéticos siguen siendo pequeños en tamaño total ( $10^3$  -  $10^5$  pares de bases) en comparación con el tamaño del genoma típico en un mamífero u otro animal ( $10^5$  -  $10^7$  pares de bases) también.



Cortesía de EMBO y Nature Publishing Group. Usado con permiso.

Fuente: Andrianantoandro, Ernesto, et al. "Biología Sintética: Nueva Ingeniería

Reglas para una Disciplina Emergente". *Biología de Sistemas Moleculares* 2, núm. 1 (2006).

Figura 26.1: Las capas de abstracción en robótica comparadas con las de biología (crédito a Ron Weiss).

Uno de los primeros hitos en la biología sintética ocurrió en el año 2000 con el represor. El represor [2] es una red reguladora genética sintética que actúa como un sistema oscilador eléctrico con períodos de tiempo fijos. Se expresó una proteína verde fluorescente dentro de *E. coli* y la fluorescencia se midió a lo largo del tiempo. Se configuraron tres genes en un bucle de retroalimentación para que cada gen reprimiera al siguiente gen en el bucle y fuera reprimido por el gen anterior.

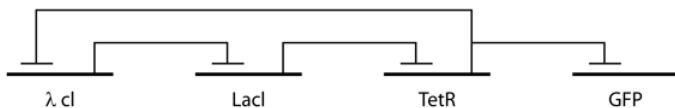
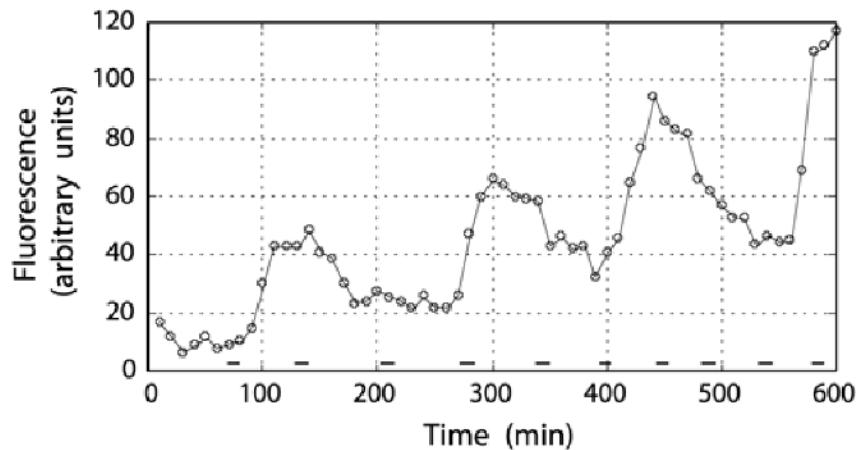


Figura 26.2: La red reguladora genética represorista.



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Elowitz, Michael B. y Stanislas Leibler. “Un osciladorio sintético

Red de Reguladores Transcripcionales” *Nature* 403, núm. 6767 (2000): 335-8.

Figura 26.3: Fluorescencia de una sola célula con el circuito represor durante un periodo de 10 horas.

El represor logró producir fluctuaciones periódicas en la fluorescencia. Sirvió como uno de los primeros triunfos en biología sintética. Otros logros en la última década incluyen el control programado de la población bacteriana, la formación programada de patrones, la comunicación celular artificial en levaduras, la creación de puertas lógicas mediante la complementación química con factores de transcripción y la síntesis completa, clonación y ensamblaje de un genoma bacteriano.

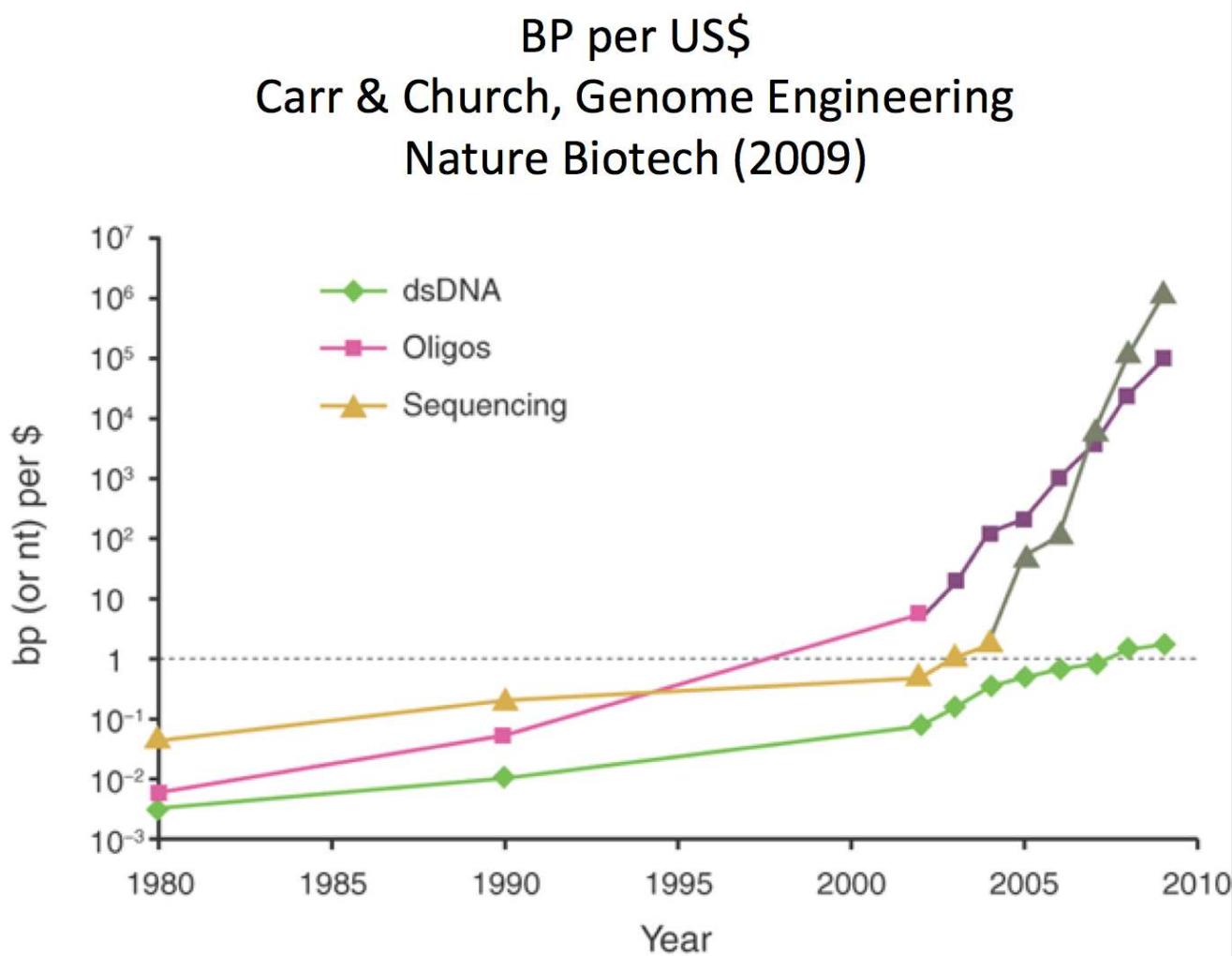
[25.1: Introducción a la Biología Sintética](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [25.1: Introduction to Synthetic Biology](#) has no license indicated.

## 25.2: Direcciones actuales de investigación

This page is a draft and is under active development.

La funcionalidad de codificación en el ADN es una forma en que los biólogos sintéticos programan células A medida que el precio de la secuenciación y síntesis del ADN sigue disminuyendo, la codificación de cadenas de ADN se ha vuelto más factible. De hecho, el número de pares de bases que se pueden sintetizar por US\$ ha aumentado exponencialmente, similar a la Ley de Moore.



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Carr, Peter A. y George M. Church. "Ingeniería del Genoma".

*Nature Biotechnology* 27, núm. 12 (2009): 1151-62

Figura 26.4: Costo de sintetizar un par base versus dólar estadounidense

Esto ha hecho que el proceso de diseñar, construir y probar circuitos biológicos sea mucho más rápido y económico. Una de las principales áreas de investigación en biología sintética es la creación de síntesis rápida y automatizada de moléculas de ADN y la creación de células con la secuencia de ADN deseada. El objetivo de crear un sistema de este tipo es acelerar el diseño y depuración de hacer un sistema biológico para que los sistemas biológicos sintéticos puedan ser prototipados y probados en un proceso rápido e iterativo.

La biología sintética también tiene como objetivo desarrollar componentes biológicos abstractos que tengan un comportamiento estándar y bien definido como una pieza que un ingeniero eléctrico podría ordenar de un catálogo. Para ello, en 2003 se creó el Registro de Partes Biológicas Estándar ([partsregistry.org](http://partsregistry.org)) [4] y actualmente contiene más de 7000 piezas disponibles para los usuarios. La parte de investigación de la creación de dicho registro incluye la clasificación y descripción de las partes biológicas. El objetivo es encontrar piezas que tengan características deseables como:

Los reguladores de **ortogonalidad** no deben interferir entre sí. Deben ser independientes.

**Composabilidad** Los reguladores se pueden fusionar para dar función compuesta.

Los reguladores de **conectividad** se pueden encadenar para permitir cascadas y retroalimentación.

**Homogeneidad** Los reguladores deben obedecer una física muy similar. Esto permite previsibilidad y eficiencia.

La biología sintética aún se está desarrollando, y la investigación aún puede ser realizada por personas con pocos antecedentes en el campo. La Fundación Internacional de Máquinas Genéticamente Modificadas (iGEM) ([igem.org](http://igem.org)) [3] creó el concurso iGEM donde estudiantes de pregrado y preparatoria compiten para diseñar y construir sistemas biológicos que operan dentro de células vivas. A los equipos estudiantiles se les entrega un kit de partes biológicas a principios del verano y trabajan en sus propias instituciones para crear un sistema biológico. Algunos proyectos interesantes incluyen:

**Biodetector de arsénico** El objetivo fue desarrollar un biosensor bacteriano que responda a un rango de concentraciones de arsénico y produzca un cambio en el pH que pueda calibrarse en relación con la concentración de arsénico. El objetivo del equipo era ayudar a muchos países subdesarrollados, en particular Bangladesh, a detectar la contaminación por arsénico en el agua. El dispositivo propuesto se pretendía ser más económico, portátil y más fácil de usar en comparación con otros detectores.

**BactoBlood** El equipo de UC Berkeley trabajó para desarrollar un sustituto de glóbulos rojos rentable construido a partir de bacterias *E. coli* diseñadas. El sistema está diseñado para transportar oxígeno de manera segura en el torrente sanguíneo sin inducir sepsis, y para ser almacenado por períodos prolongados en estado líofilizado.

**E. Chromi** El proyecto del equipo Cambridge se esforzó por facilitar el diseño y construcción de biosensores. Diseñaron y caracterizaron dos tipos de piezas - Afinadores de Sensibilidad y Generadores de Color - *E. coli* diseñados para producir diferentes pigmentos en respuesta a diferentes concentraciones de un inductor. La disponibilidad de estas piezas revolucionó el camino del futuro diseño de biosensores.

---

25.2: Direcciones actuales de investigación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 25.2: Current Research Directions has no license indicated.

## 25.3: Herramientas y Técnicas

This page is a draft and is under active development.

La biología sintética combina muchos campos, y las técnicas utilizadas no son particulares de la biología sintética. Al igual que el proceso de resolver otros problemas de ingeniería, el proceso de creación de un sistema biológico útil tiene fases de diseño, construcción, prueba y mejora. Una vez que se crea un diseño o declaración de las propiedades deseadas de un sistema biológico, el problema se convierte en encontrar los componentes biológicos adecuados para construir dicho sistema.

BioCompiler [1] es una herramienta desarrollada para permitir la programación de circuitos biológicos utilizando un lenguaje de programación de alto nivel. Uno puede escribir programas en un lenguaje similar al LISP y compilar su programa en un circuito biológico. BioCompiler utiliza un proceso similar al de un compilador para un lenguaje de programación. Utiliza un programa escrito por humanos como descripción de alto nivel del circuito genético, luego genera una descripción formal del programa. A partir de ahí, busca piezas abstractas de la red reguladora genética que se pueden combinar para crear el circuito genético y pasa por su biblioteca de partes de ADN para encontrar secuencias adecuadas que coincidan con la funcionalidad de las piezas abstractas de la red reguladora genética. Luego se pueden generar instrucciones de ensamblaje para crear células con la red reguladora genética adecuada.

Figura 26.5: Un ejemplo de un programa BioCompiler y el proceso de actualizarlo (crédito a Ron Weiss)

Las piezas biológicas estándar de BioRick (biobricks.org) son otra herramienta utilizada en biología sintética. Similar a las partes del Registro de Partes Biológicas Estándar, las partes biológicas estándar de BioRick son secuencias de ADN de estructura y función definidas. Cada parte de BioRick es una secuencia de ADN que se mantiene unida en un plásmido circular. En cada extremo del BioRick contiene una secuencia conocida y bien definida con enzimas de restricción que pueden abrir el plásmido en posiciones conocidas. Esto permite la creación de piezas más grandes de BioRick encadenando otras más pequeñas. Algunos competidores en la competencia iGEM utilizaron sistemas BioRick para desarrollar una línea de *E. coli* que producía aromas como el plátano o la menta.

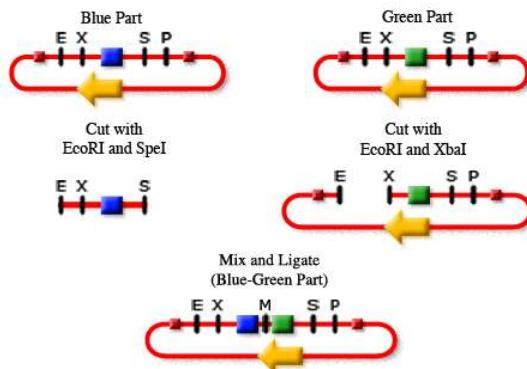


Figura 26.6: Un ejemplo de combinación de BioRick Pieces tomado de 2006.igem.org/wiki/index.php/Standard\_Assembly

25.3: Herramientas y Técnicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 25.3: Tools and Techniques has no license indicated.

## 25.4: ¿Qué hemos aprendido? , Bibliografía

La biología sintética es una disciplina emergente que tiene como objetivo crear sistemas biológicos útiles para resolver problemas en energía, medicina, medio ambiente y muchos más campos. Los biólogos sintéticos intentan utilizar la abstracción para permitirles construir sistemas más complejos a partir de sistemas más simples de manera similar a cómo un ingeniero de software o un ingeniero eléctrico haría un programa de cómputos o un circuito complejo. El Registro de Partes Biológicas Estándar y BioRick estándar de piezas biológicas tienen como objetivo caracterizar y estandarizar piezas biológicas tal como lo haría un transistor o puerta lógica para permitir la abstracción. Herramientas como BioCompiler permiten a las personas describir un circuito genético utilizando un lenguaje de alto nivel y construir realmente un circuito genético con la funcionalidad descrita. La biología sintética aún es nueva, y la investigación puede ser realizada por aquellos que no estén familiarizados con el campo, como lo demuestra la competencia iGEM.

---

25.4: ¿Qué hemos aprendido? , Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [25.4: What Have We Learned?, Bibliography](#) has no license indicated.

## Bibliografía

- 
- [1] J. Beal y J. Bachrach. Las celdas son objetivos plausibles para lenguajes espaciales de alto nivel, 2008.
  - [2] M. Elowitz y S. Leibler. Una red oscillatoria sintética de reguladores transcripcionales. *Naturaleza*, 403:335 — 338, 2000.
  - [3] iGEM. igem: Biología sintética basada en piezas estándar, diciembre de 2012.
  - [4] Registro de Partes Biológicas Estándar. Registro de partes biológicas normalizadas, diciembre de 2012.
- 

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 26: Evolución Molecular y Filogenética

- 26.1: Introducción
- 26.2: Fundamentos de la Filogenia
- 26.3: Métodos basados en la distancia
- 26.4: Métodos basados en caracteres
- 26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido
- 26.6: Hacia el proyecto final
- 26.7: ¿Qué hemos aprendido?

#### Bibliografía

---

26: Evolución Molecular y Filogenética is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

## 26.1: Introducción

La filogenética es el estudio de las relaciones entre un conjunto de objetos que tienen un origen común, basado en el conocimiento de los rasgos individuales de los objetos. Dichos objetos pueden ser especies, genes o lenguajes, y sus rasgos correspondientes pueden ser características morfológicas, secuencias, palabras, etc. En todos estos ejemplos los objetos en estudio cambian gradualmente con el tiempo y divergen de orígenes comunes a objetos actuales.

En Biología, la filogenética es particularmente relevante porque todas las especies biológicas resultan ser descendientes de un solo ancestro común que existió hace aproximadamente 3.5 a 3.8 mil millones de años. A lo largo del paso del tiempo, la variación genética, el aislamiento y la selección han creado la gran variedad de especies que observamos hoy en día. Sin embargo, no solo la especiación, sino que la extinción también ha jugado un papel clave en la conformación de la biosfera como vemos hoy en día. Estudiar la ascendencia entre especies diferentes es fundamentalmente importante para la biología porque arrojan mucha luz en la comprensión de las funciones biológicas diferentes, los mecanismos genéticos así como el propio proceso de evolución.

---

[26.1: Introducción](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [26.1: Introduction](#) has no license indicated.

## 26.2: Fundamentos de la Filogenia

### Árboles

Un árbol es una representación matemática de las relaciones entre objetos. Un árbol general se construye a partir de nodos y bordes. Cada nodo representa un objeto, y cada borde representa una relación entre dos nodos. En el caso de los árboles filogenéticos, representamos la evolución utilizando árboles. En este caso, cada nodo representa un evento de divergencia entre dos linajes ancestrales, las hojas denotan el conjunto de objetos presentes y la raíz representa al ancestro común.

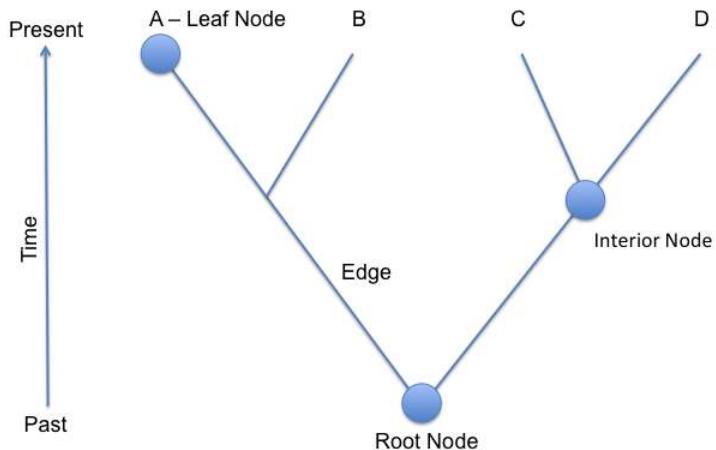


Figura 26.2: Definición de la terminología del árbol. Se representa un árbol de nodos ramificados con hojas en la parte superior y la raíz en la parte inferior. El tiempo continúa hacia arriba, hacia las hojas.

Sin embargo, a veces se refleja más información en las longitudes de las ramas, como el tiempo transcurrido o la cantidad de disimilitud. Según estas diferencias, los árboles filogenéticos biológicos pueden clasificarse en tres categorías:

**Cladograma:** no da sentido a las longitudes de ramificación; solo importa la secuencia y topología de la ramificación.

**Filograma:** Las longitudes de las ramas están directamente relacionadas con la cantidad de cambio genético. Cuanto más larga es la rama de un árbol, mayor es la cantidad de cambio filogenético que ha tenido lugar. Las hojas de este árbol no necesariamente terminan en la misma línea vertical, debido a diferentes tasas de mutación.

**Cronograma (árbol ultramétrico):** Las longitudes de las ramas están directamente relacionadas con el tiempo. Cuanto más largas sean las ramas de un árbol, mayor será la cantidad de tiempo que ha pasado. Las hojas de este árbol necesariamente terminan en la misma línea vertical (es decir, están a la misma distancia de la raíz), ya que todas están presentes a menos que se incluyan especies extintas en el árbol. Aunque existe una correlación entre las longitudes de las ramas y la distancia genética en un cronograma, no son necesariamente exactamente proporcionales porque las tasas de evolución/tasas de mutación no son constantes. Algunas especies evolucionan y mutan más rápido que otras, y algunos períodos históricos fomentan tasas de evolución más rápidas que otras.

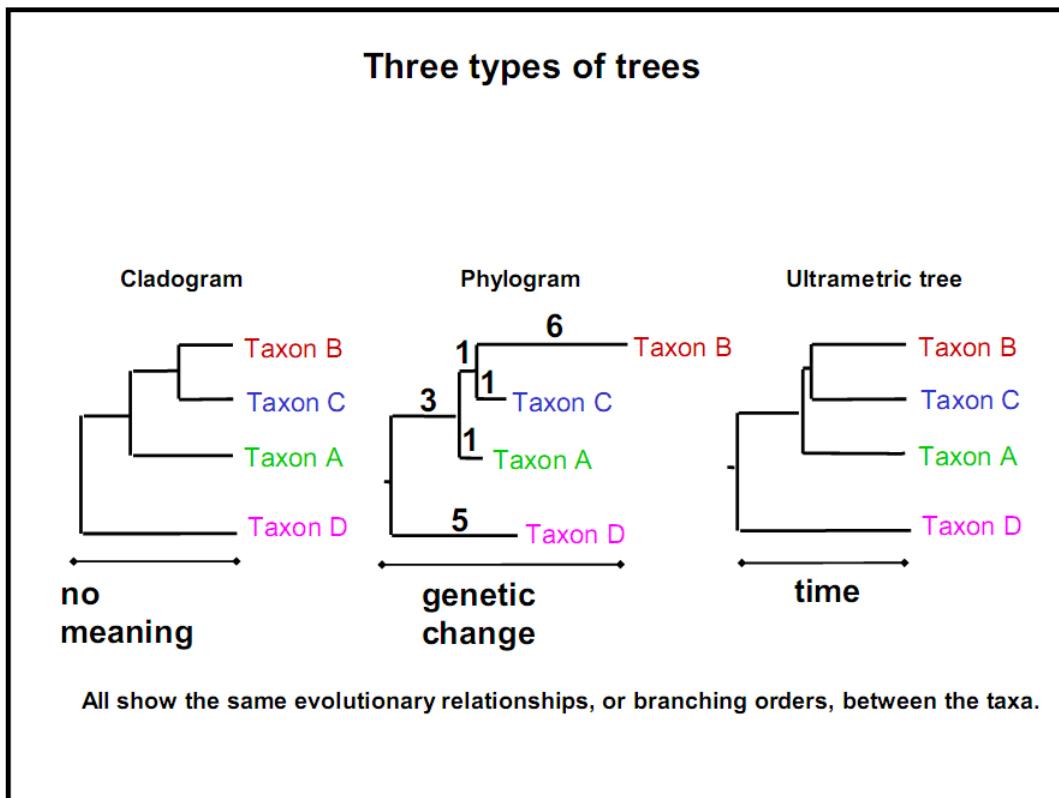


Figura 26.3: Tres tipos de árboles.

Un rasgo es cualquier característica que posea un objeto o especie. En los humanos, un ejemplo de un rasgo puede ser el bipedalismo (la capacidad de caminar erguido) o el pulgar oponible. Otro rasgo humano puede ser una secuencia de ADN específica que poseen los humanos. Los primeros ejemplos de rasgos físicos se denominan rasgos morfológicos, mientras que los últimos rasgos de ADN se denominan rasgos de secuencia. Cada uno tiene sus ventajas y desventajas para estudiar. Todos los métodos para la reconstrucción arbórea se basan en el estudio de la ocurrencia de rasgos diferentes en los objetos dados. En la filogenética tradicional se utilizaron los datos morfológicos de diferentes especies para este propósito. En los métodos modernos, se utilizan datos de secuencias genéticas en su lugar. Cada uno tiene sus ventajas y desventajas.

**Rasgos morfológicos:** Surgen de la evaluación empírica de los rasgos físicos. Esto puede ser ventajoso porque las características físicas son muy fáciles de cuantificar y entender para todos, científicos y niños por igual. Las desventajas de este enfoque son que solo podemos evaluar un pequeño conjunto de rasgos, como pelo, uñas, pezuñas, dientes, etc. Además, estos rasgos solo nos permiten construir especies. Por último, es mucho más fácil ser “engañado” por la evolución convergente. Las especies que divergieron hace millones de años pueden volver a converger en los pocos rasgos que son observables para los científicos, dando una falsa representación de cuán estrechamente relacionadas están las especies.

**Rasgos de Secuencia:** Se descubren mediante el estudio de los genomas de diferentes especies. Este enfoque puede ser ventajoso porque crea muchos más datos y permite a los científicos crear árboles genéticos además de árboles de especies. La principal dificultad con este enfoque es que el ADN solo se construye a partir de 4 bases, por lo que las retromutaciones son frecuentes. En este enfoque, los científicos deben conciliar las señales de un gran número de rasgos de mal comportamiento frente a la de un pequeño número de rasgos de buen comportamiento en el enfoque tradicional. El resto del capítulo se centrará principalmente en la construcción de árboles a partir de secuencias génicas.

Dado que este enfoque trata de comparar entre pares de genes, es útil entender el concepto de homología: Un par de genes se llaman parálogos si divergieron de un evento de duplicación, y ortólogos si divergieron de un evento de especiación.

## Preguntas frecuentes

P: ¿Sería posible utilizar secuencias de ADN de especies extintas?

R: Las tecnologías actuales solo permiten el uso de secuencias existentes. Sin embargo, ha habido algunos éxitos en el uso del ADN de especies extintas. El ADN de mamuts congelados ha sido recolectado y están siendo secuenciadas pero debido a que el ADN se descompone con el tiempo y la contaminación del ambiente, es muy difícil extraer secuencias correctas.

Una vez que hemos encontrado datos genéticos para un conjunto de especies, nos interesa conocer cómo esas especies se relacionan entre sí. Dado que en su mayor parte solo podemos obtener ADN de criaturas vivientes, debemos inferir la existencia de ancestros de cada especie, y en última instancia inferir la existencia de un ancestro común. Este es un problema desafiante, porque se dispone de datos muy limitados. Las siguientes secciones explorarán los métodos modernos para inferir ascendencia a partir de datos de secuencia. Se pueden clasificar en dos enfoques, métodos basados en distancia y métodos basados en caracteres.

Los enfoques basados en la distancia toman dos pasos para resolver el problema, es decir, cuantificar la cantidad de mutación que separa cada par de secuencias (que puede o no ser proporcional al tiempo transcurrido desde que se han separado) y ajustar el árbol más probable de acuerdo con la matriz de distancia por pares. El segundo paso suele ser un algoritmo directo, basado en algunas suposiciones, pero puede ser más complejo.

En cambio, los enfoques basados en caracteres intentan encontrar el árbol que mejor explique las secuencias observadas. A diferencia de la reconstrucción directa, estos métodos se basan en la propuesta de árboles y técnicas de puntuación para realizar una búsqueda heurística sobre el espacio de los árboles.

## ¿Sabías?

La navaja de Occam, como se discutió en capítulos anteriores, no siempre proporciona la hipótesis más precisa. En muchos casos durante la reconstrucción arbórea, la explicación más simple no es la más probable. Por ejemplo, puede ser posible un conjunto de posibles ancestros, dados algunos datos observados. En este caso, la ascendencia más simple puede no ser correcta si un rasgo surgió independientemente en dos linajes separados. Este tema será considerado en una sección posterior.

[26.2: Fundamentos de la Filogenia](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [26.2: Basics of Phylogeny](#) has no license indicated.

## 26.3: Métodos basados en la distancia

Los modelos basados en distancia secuestran los datos de secuencia en distancias por pares. Este paso pierde algo de información, pero establece la plataforma para la reconstrucción directa de árboles. Las dos etapas de este método se discuten en detalle en la presente memoria.

### De la alineación a las distancias

Para entender cómo funciona un modelo basado en la distancia, es importante pensar qué significa distancia al comparar dos secuencias. Hay tres interpretaciones principales.

La divergencia de nucleótidos es la idea de medir la distancia entre dos secuencias en función del número de lugares donde los nucleótidos no son consistentes. Esto supone que la evolución ocurre a una velocidad uniforme en todo el genoma, y que un nucleótido dado es igual de probable que evolucione hacia cualquiera de los otros tres nucleótidos. A pesar de que tiene deficiencias, esta suele ser una excelente manera de pensarlo.

Transiciones y Transversiones Esto es similar a la divergencia de nucleótidos, pero reconoce que las sustituciones A-G y T-C son las más frecuentes. Por lo tanto, mantiene dos parámetros, la probabilidad de una transición y la probabilidad de una transversión.

Sustituciones sinónimas y no sinónimas Este método mantiene un seguimiento de las sustituciones que afectan al aminoácido codificado asumiendo que las sustituciones que no cambian la proteína codificada no se seleccionarán contra, y por lo tanto tendrán una mayor probabilidad de ocurrir que aquellas sustituciones que cambien el aminoácido codificado.

La forma ingenua de interpretar la separación entre dos secuencias puede ser simplemente el número de desapareamientos, como se describe por la divergencia de nucleótidos anteriormente. Si bien esto nos proporciona una métrica de distancia (es decir,  $d(a, b) + d(b, c) = d(a, c)$ ) esto no satisface del todo nuestros requisitos, porque queremos distancias aditivas, es decir, aquellas que satisfacen  $d(a, b) + d(b, c) = d(a, c)$  para un camino a! b! c de secuencia evolutiva, porque la cantidad de mutaciones acumuladas a lo largo de un camino en el árbol debe ser la suma de la de sus componentes individuales. Sin embargo, la fracción de desajuste ingenuo no siempre tiene esta propiedad, ya que esta cantidad está limitada por 1, mientras que la suma de componentes individuales puede superar fácilmente 1.

La clave para resolver esta paradoja son las retromutaciones. Cuando un gran número de mutaciones se acumulan en una secuencia, no todas las mutaciones introducen nuevos desapareamientos, algunas de ellas pueden ocurrir en un par de bases ya mutado, dando como resultado que la puntuación de desapareamientos permanezca igual o incluso decreciente. Sin embargo, para pequeñas puntuaciones de desajuste, este efecto es estadísticamente insignificante, porque hay muchísimo más pares idénticos que pares descoincidentes. Sin embargo, para secuencias separadas por mayor distancia evolutiva, debemos corregir este efecto. El modelo Jukes-Cantor es uno de esos modelos de markov simples que toma esto en cuenta.

### Distancias Jukes-Cantor

Para ilustrar este concepto, considere un nucleótido en estado 'A' en el tiempo cero. En cada paso de tiempo, tiene una probabilidad 0.7 de retener su estado previo y probabilidad 0.1 de transición a cada uno de los otros tres estados. La probabilidad  $P(B|t)$  de observar el estado (base) B en el tiempo t sigue esencialmente a la recursión

$$P(B | t + 1) = 0.7P(B | t) + 0.1 \sum_{b \neq B} P(b | t) = 0.1 + 0.6P(B | t)$$

Figura 26.5: Cadena de Markov que da cuenta de las mutaciones de espalda

Si trazamos  $P(B|t)$  frente a t, observamos que la distribución comienza como concentrada en el estado 'A' y gradualmente se extiende al resto de los estados, yendo eventualmente hacia un equilibrio de probabilidades iguales. Esta progresión tiene sentido, intuitivamente. A lo largo de millones de años, las especies pueden evolucionar tan dramáticamente que ya no se parecen a sus antepasados. En ese extremo, una ubicación base dada en el antepasado es igual de probable que haya evolucionado a cualquiera de las cuatro bases posibles en esa ubicación a lo largo del tiempo.

```
\begin{array}{lcccccc}
\text{time: -} & 0 & 1 & 2 & 3 & 4 \\
\hline
\text{A} & 1 & 0.7 & 0.52 & 0.412 & 0.3472
\end{array}
```

```
\text{C} & 0 & 0.1 & 0.16 & 0.196 & 0.2196\\
\text{G} & 0 & 0.1 & 0.16 & 0.196 & 0.2196\\
\text{T} & 0 & 0.1 & 0.16 & 0.196 & 0.2196\\
\end{array}\nonumber]
```

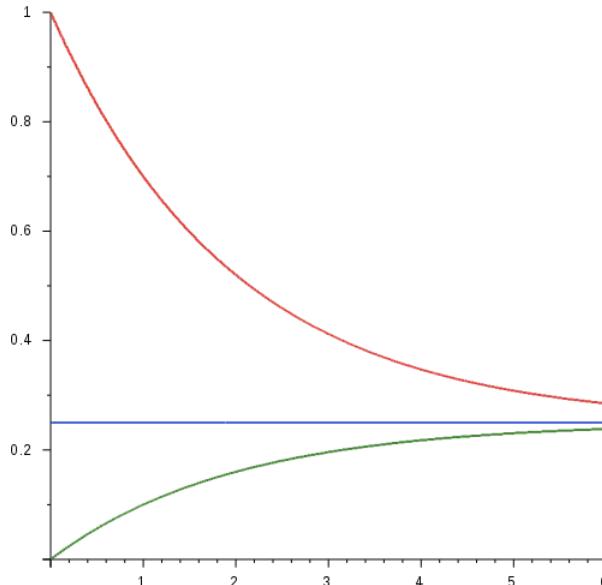


Figura 26.6: El eje y denota probabilidad de observar las bases - A (rojo), otras (verde). El eje x denota el tiempo.

La esencia del modelo Jukes Cantor es retroceder  $t$ , la cantidad de tiempo transcurrido desde la fracción de bases alteradas. Conceptualmente, esto es solo invertir los ejes x e y de la curva verde. Para modelar esto cuantitativamente, consideramos la siguiente matriz  $S(t)$  que denota las respectivas probabilidades  $P(x|y, t)$  de observar la base  $x$  dada un estado inicial de base  $y$  en el tiempo  $t$ .

```
\[S(\Delta t) = \left( \begin{array}{cccc}
P(A|\text{media A}, \Delta t) & P(A|\text{media G}, \Delta t) & \dots & P(A|\text{media T}, \Delta t) \\
P(G|\text{media A}, \Delta t) & \dots & \dots & \dots \\
\vdots & \vdots & \ddots & \vdots \\
P(T|\text{mediados de A}, \Delta t) & \dots & \dots & P(T|\text{mid T}, \Delta t)
\end{array} \right)
```

Podemos suponer que se trata de un modelo de markov estacionario, lo que implica que esta matriz es multiplicativa, i.e.

$$S(t_1 + t_2) = S(t_1) S(t_2) \quad (26.3.1)$$

Por muy poco tiempo  $\epsilon$ , podemos suponer que no hay efecto de segundo orden, es decir, no hay tiempo suficiente para que ocurran dos mutaciones en el mismo nucleótido. Entonces las probabilidades de las transiciones cruzadas son todas proporcionales a  $\epsilon$ . Además, en el modelo Jukes Cantor, asumimos que todas las tasas de transición son las mismas de cada nucleótido a otro nucleótido. Por lo tanto, por un corto tiempo  $\epsilon$

```
\[S(\epsilon) = \left( \begin{array}{cccc}
1-3\alpha\epsilon & \alpha\epsilon & \alpha\epsilon & \alpha\epsilon \\
\alpha\epsilon & 1-3\alpha\epsilon & \alpha\epsilon & \alpha\epsilon \\
\alpha\epsilon & \alpha\epsilon & 1-3\alpha\epsilon & \alpha\epsilon \\
\alpha\epsilon & \alpha\epsilon & \alpha\epsilon & 1-3\alpha\epsilon
\end{array} \right)
```

En el tiempo  $t$ , la matriz viene dada por

```
\[S(t) = \left( \begin{array}{cccc}
r(t) & s(t) & s(t) & s(t)
\end{array} \right)
```

```
s (t) & r (t) & s (t) & s (t)\
s (t) & s (t) & r (t) & s (t)\ s (t) &
s (t) & s (t) & s (t) & r (t)
\end {array}\derecha\nonumber]
```

De la ecuación  $S(t + \epsilon) = S(t)S(\epsilon)$  obtenemos

$$r(t + \epsilon) = r(t)(1 - 3\alpha\epsilon) + 3\alpha\epsilon s(t) \text{ and } s(t + \epsilon) = s(t)(1 - \alpha\epsilon) + \alpha\epsilon r(t)$$

Que se reordenan como el sistema acoplado de ecuaciones diferenciales

$$r'(t) = 3\alpha(-r(t) + s(t)) \text{ and } s'(t) = \alpha(r(t) - s(t))$$

Con las condiciones iniciales  $r(0) = 1$  y  $s(0) = 0$ . Las soluciones se pueden obtener como

$$r(t) = \frac{1}{4} (1 + 3e^{-4\alpha t}) \text{ and } s(t) = \frac{1}{4} (1 - e^{-4\alpha t})$$

Ahora bien, en una alineación dada, si tenemos la fracción  $f$  de los sitios donde difieren las bases, tenemos:

$$f = 3s(t) = \frac{3}{4} (1 - e^{-4\alpha t})$$

lo que implica

$$t \propto -\log \left(1 - \frac{4f}{3}\right)$$

Para acordar asintóticamente con  $f$ , establecemos la distancia evolutiva  $d$  para ser

$$d = -\frac{3}{4} \log \left(1 - \frac{4f}{3}\right)$$

Tenga en cuenta que la distancia es aproximadamente proporcional a  $f$  para valores pequeños de  $f$  y asintóticamente se acerca al infinito cuando  $f \neq 0.75$ . Intuitivamente esto sucede porque después de un periodo de tiempo muy largo, esperaríamos que la secuencia fuera completamente aleatoria y eso implicaría alrededor de tres cuartas partes de las bases que no coinciden con las originales. Pero los valores de incertidumbre de la distancia Jukes-Cantor también se vuelven muy grandes cuando  $f$  se acerca a 0.75.

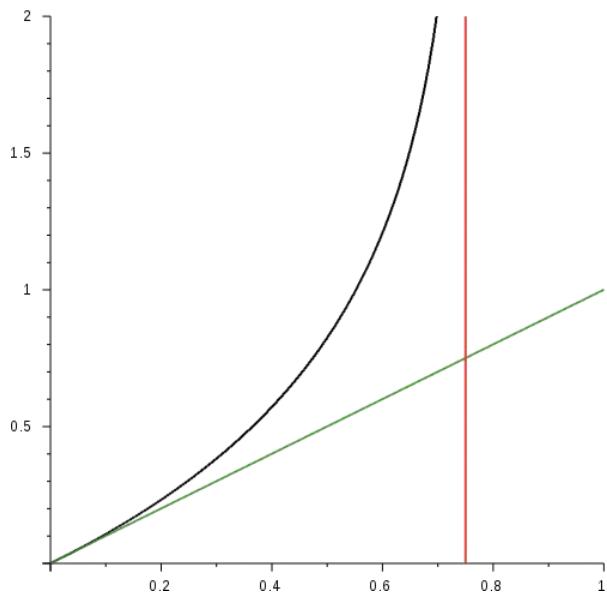
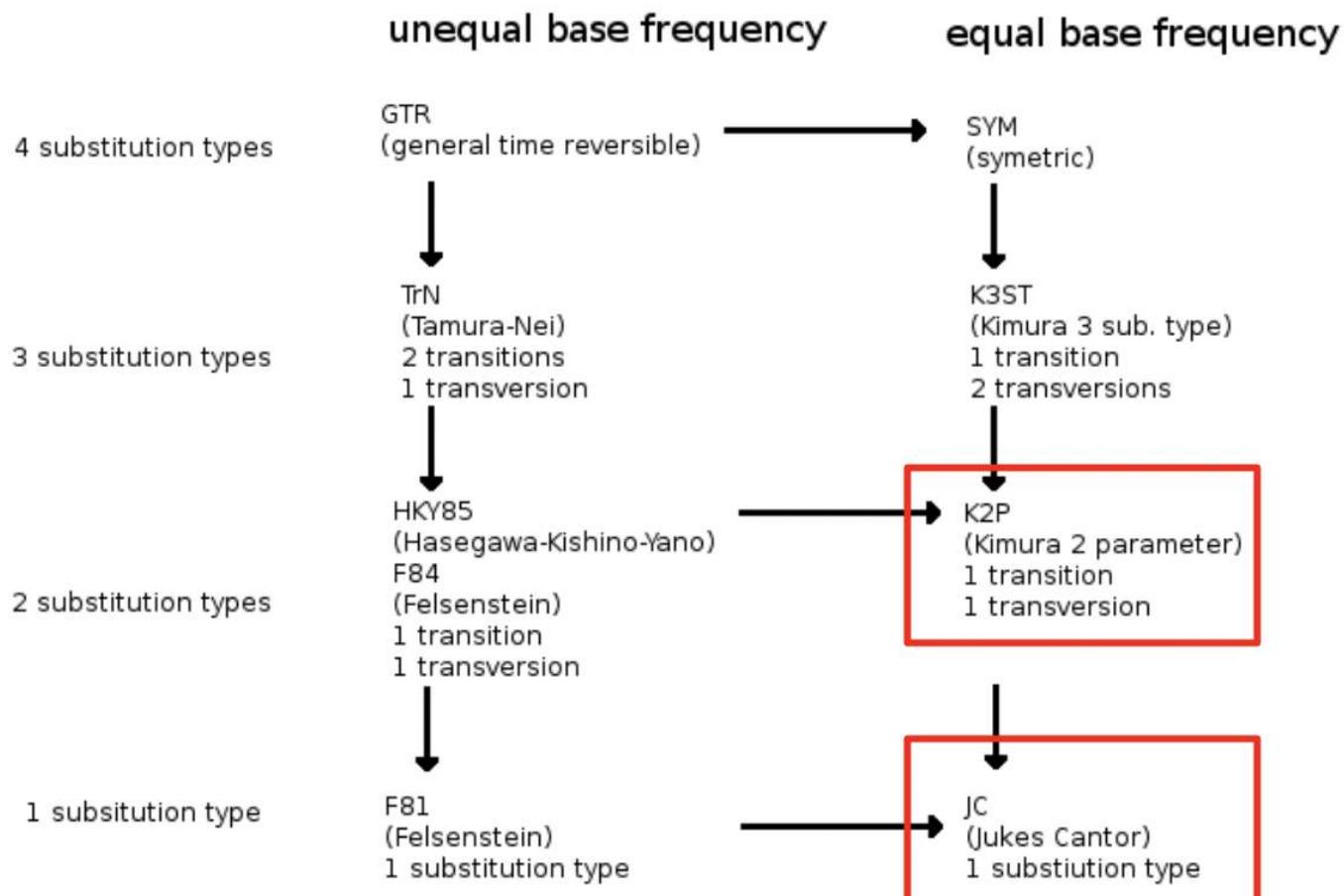


Figura 26.7: Fracción de bases alteradas (eje x) frente a la distancia Jukes Cantor (eje y).

Línea negra denota la curva, verde es la línea de tendencia para valores pequeños de  $f$  mientras que la línea roja denota el límite asintótico.

### Otros modelos

El modelo Jukes Cantor es el modelo más simple que nos da un modelo de distancia aditiva teóricamente consistente. Sin embargo, es un modelo de un parámetro que asume que las mutaciones de cada base a una base diferente tienen la misma probabilidad. Pero, los cambios entre AG o entre TC son más probables que los cambios a través de ellos. El primer tipo de sustitución se llama transiciones mientras que el segundo tipo se llama transversiones. El modelo Kimura tiene dos parámetros que tienen esto en cuenta. También hay muchas otras modificaciones de este modelo de distancia que toma en cuenta las diferentes tasas de transiciones y transversiones etc. que se representan a continuación.



## Models also exist for peptides and codons

Figura 26.8: Modelos de distancia de diferentes niveles de complejidad (parámetros).

### FAQ

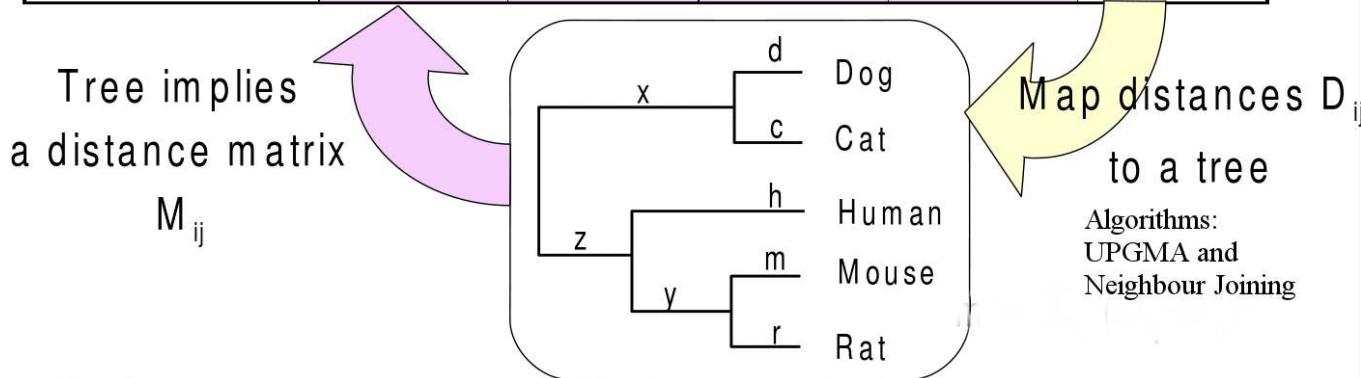
P: ¿Podemos usar diferentes parámetros para diferentes partes del árbol? ¿Para dar cuenta de diferentes tasas de mutación?

R: Es posible, es un área de investigación actual.

### Distancias a los árboles

Si tenemos un árbol filogenético ponderado, podemos encontrar el peso total (longitud) del camino más corto entre un par de hojas sumando las longitudes de rama individuales en el camino. Teniendo en cuenta todos esos pares de hojas, tenemos una matriz de distancia que representa los datos. En los métodos basados en distancia, el problema es reconstruir el árbol dada esta matriz de distancia.

|       | Hum     | Mou       | Rat       | Dog | Cat |
|-------|---------|-----------|-----------|-----|-----|
| Human | 0       | 4         | 5         | 7   | 6   |
| Mouse | h.y.m   | 0         | 3         | 8   | 5   |
| Rat   | h.y.r   | m.r       | 0         | 9   | 7   |
| Dog   | h.z.x.d | m.y.z.x.d | r.y.z.x.d | 0   | 2   |
| Cat   | h.z.x.c | m.y.z.x.c | r.y.z.x.c | d.c | 0   |



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 26.9: Mapeo de un árbol a una matriz de distancia y viceversa

#### FAQ

P: En la Figura 27.9 Las métricas de divergencia m y r pueden tener algún solapamiento por lo que la distancia entre ratón y rata no es simplemente m+r. ¿No sería ese solo el caso si no hubiera superposición?

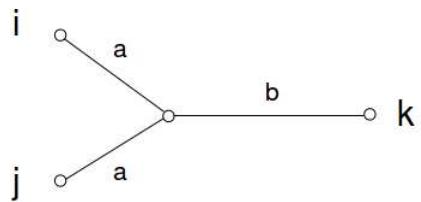
R: Si modelas la evolución correctamente, entonces obtendrías distancia evolutiva. Es una desigualdad más que una igualdad y coincidimos en que no se puede inferir exactamente que la distancia dada es la distancia precisa. Por lo tanto, la distancia de secuencias entre ratón y rata es probablemente menor que m + r debido a la superposición, evolución convergente y transversiones.

Sin embargo, tenga en cuenta que no existe una correspondencia uno a uno entre una matriz de distancia y un árbol ponderado. Cada árbol sí corresponde a una matriz de distancia, pero lo contrario no siempre es cierto. Una matriz de distancia tiene que satisfacer propiedades adicionales para corresponder a algún árbol ponderado. De hecho, hay dos modelos que asumen restricciones especiales en la matriz de distancia:

**Ultramétrico:** Para todos los trillizos (a, b, c) de hojas, dos pares entre ellos tienen la misma distancia, y la tercera distancia es menor; es decir, el triplete puede etiquetarse i, j, k de tal manera que

$$d_{ij} \leq d_{ik} = d_{jk}$$

Conceptualmente esto se debe a que las dos hojas que están más estrechamente relacionadas (digamos i, j) han divergido de la tercera (k) exactamente al mismo tiempo, y la separación temporal de la tercera debería ser igual, mientras que la separación entre ellas debería ser menor.



where  $a \leq b$

Figura 27.10: Distancias ultramétricas.

Aditivo: Las matrices aditivas de distancia satisfacen la propiedad de que todos los cuarteto de hojas pueden etiquetarse  $i, j, k, l$  de tal manera que

$$d_{ij} + d_{kl} \leq d_{ik} + d_{jl} = d_{il} + d_{jk}$$

De hecho, esto es cierto para todos los árboles de peso positivo. Para cualquier 4 hojas en un árbol, puede haber exactamente una topología, i.e.

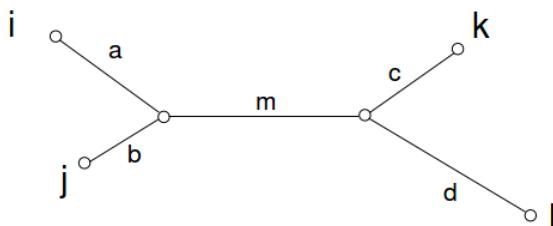


Figura 26.11: Distancias aditivas.

Entonces la condición anterior es término por término equivalente a

$$(a + b) + (c + d) \leq (a + m + c) + (b + m + d) = (a + m + d) + (b + m + c)$$

Esta igualdad corresponde a todas las distancias por pares que son posibles de atravesar este árbol.

Estos tipos de ecualidades redundantes deben ocurrir al mapear un árbol a una matriz de distancias, ya que un árbol de  $n$  nodos tiene  $n - 1$  parámetros, uno para cada longitud de rama, mientras que una matriz de distancia tiene  $n^2$  parámetros. Por lo tanto, un árbol es esencialmente una proyección de menor dimensión de un espacio dimensional superior. Un corolario de esta observación es que no todas las matrices de distancia tienen un árbol correspondiente, sino que todos los árboles se mapean a matrices de distancia únicas.

Sin embargo, los conjuntos de datos reales no satisfacen exactamente las restricciones ultramétricas o aditivas. Esto puede deberse al ruido (cuando nuestros parámetros para nuestros modelos evolutivos no son precisos), estocástico y aleatoriedad (debido a pequeñas muestras), fluctuaciones, diferentes tasas de mutaciones, conversiones de genes y transferencia horizontal. Debido a esto, necesitamos algoritmos de construcción de árboles que sean capaces de manejar matrices de distancia ruidosas.

A continuación, se discutirán dos algoritmos que se basan directamente en estos supuestos para la reconstrucción de árboles.

#### UPGMA - Método de grupo de pares no ponderados con media aritmética

Esto es exactamente lo mismo que el método de **agrupamiento jerárquico** discutido en la Conferencia 13, Clustering de expresión génica. Forma clústeres paso a paso, desde nodos estrechamente relacionados hasta aquellos que están más separados. Se forma un nodo de ramificación para cada nivel sucesivo. El algoritmo se puede describir correctamente mediante los siguientes pasos:

#### Inicialización:

1. Definir una hoja  $i$  por secuencia  $x_i$ .
2. Coloca cada hoja  $i$  a la altura 0.
3. Definir Clusters  $C_i$  cada uno teniendo una hoja  $i$ .

#### Iteración:

1. Encuentre las distancias por pares  $d_{ij}$  entre cada par de clústeres  $C_i, C_j$  tomando la media aritmética de las distancias entre sus secuencias miembro.
2. Encuentra dos cúmulos  $C_i, C_j$  de tal manera que  $d_{ij}$  se minimicen.
3. Dejar  $C_k = C_i \cup C_j$ .
4. Definir el nodo k como padre de los nodos i, j y colocarlo a la altura  $d_{ij}/2$  por encima de i, j.
5. Eliminar  $C_i, C_j$ .

**Terminación:** Cuando quedan dos clústeres  $C_i, C_j$ , coloca la raíz a la altura  $d_{ij}/2$  como padre de los nodos i, j

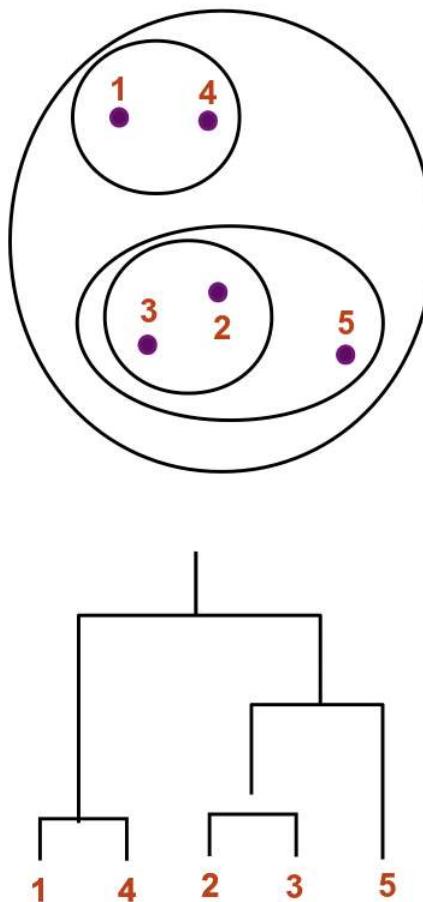


Figura 26.12: UPGMA/Clustering jerárquico

#### Ultrametrificación de árboles no ultramétricos

Si un árbol no satisface condiciones ultramétricas, podemos intentar encontrar un conjunto de alteraciones en una matriz de distancias simétricas  $n \times n$  que lo hagan ultramétrico. Esto se puede lograr construyendo una gráfica completamente conectada con pesos dados por la matriz de distancia original, encontrando un árbol de expansión mínima (MST) de esta gráfica, y luego construyendo una nueva matriz de distancia con los elementos  $D(i, j)$  dados por el mayor peso en la ruta única en el MST de  $i$  a  $j$ . El árbol de expansión del gráfico completamente conectado simplemente identifica un subconjunto de bordes que conecta todos los nodos sin crear ningún ciclo, y un árbol de expansión mínimo es un árbol de expansión que minimiza la suma total de pesos de borde. Un MST se puede encontrar usando el algoritmo de Prim's, y luego se usa para corregir un árbol no ultramétrico.

#### Debilidades de la UPGMA

Si bien este método está garantizado para encontrar el árbol correcto si la matriz de distancia obedece a la propiedad ultramérica, resulta ser un algoritmo inexacto en la práctica. Aparte de la falta de robustez, sufre de la suposición del reloj molecular de que la tasa de mutación a lo largo del tiempo es constante para todas las especies. Sin embargo, esto no es cierto ya que ciertas especies como las ratas y los ratones evolucionan mucho más rápido que otras. Tales diferencias en la tasa de mutación pueden conducir a una atracción de ramas largas; los nodos que comparten una tasa de mutación más baja pero que se encuentran en linajes distintos pueden fusionarse, dejando que aquellos nodos con tasas de mutación más altas (ramas largas) aparezcan juntos en el árbol. La siguiente figura ilustra un ejemplo donde UPGMA falla:

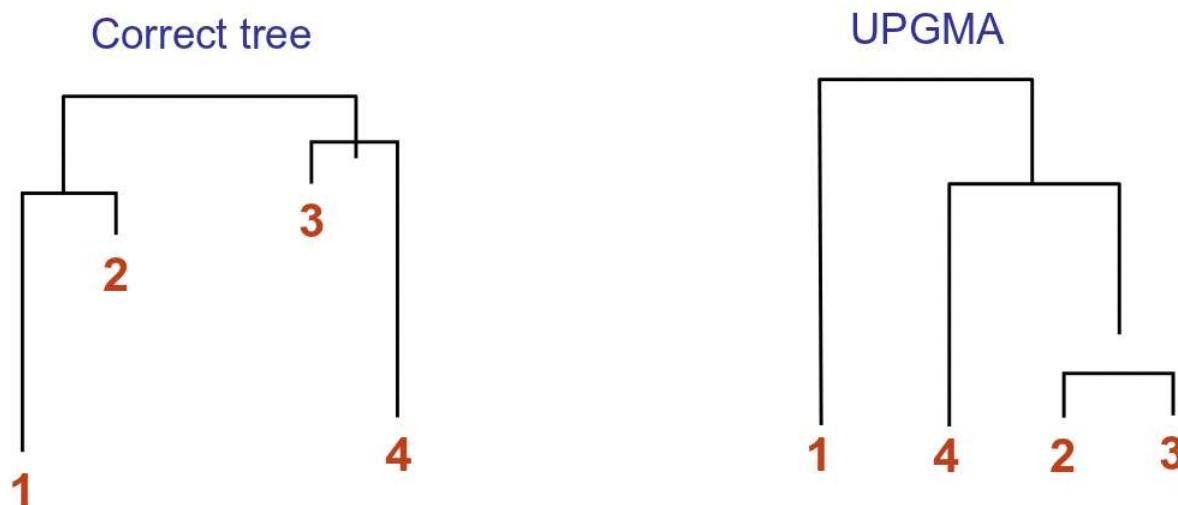


Figura 27.13: UPGMA no encuentra el árbol correcto en este caso

### Vecino Uniéndose

Se garantiza que el método de unión de vecinos produzca el árbol correcto si la matriz de distancia satisface la propiedad aditiva. También puede producir un buen árbol cuando hay algo de ruido en los datos. El algoritmo se describe a continuación:

**Encontrar las hojas vecinas:** Vamos

$$D_{ij} = d_{ij} - (r_i + r_j) \text{ where } r_a = \frac{1}{n-2} \sum_k d_{ak}, a \in \{i, j\}$$

Aquí n es el número de nodos en el árbol; por lo tanto,  $r_i$  es la distancia promedio de un nodo a los otros nodos. Se puede probar que la modificación anterior asegura que  $D_{ij}$  es mínimo solo si i, j son vecinos. (Una prueba se puede encontrar en la página 189 del libro de Durbin).

**Inicialización:** Definir T como el conjunto de nodos hoja, uno por secuencia. Dejar L = T

**Iteración:**

1. Pick i, j tal que  $D_{ij}$  se minimice.
2. Definir un nuevo nodo k, y establecer

$$d_{km} = \frac{1}{2} (d_{im} + d_{jm}) \forall m \in L$$

3. Añadir k a T, con bordes de longitudes

$$d_{ik} = \frac{1}{2} (d_{ij} + r_i r_j)$$

4. Quitar i, j de L

5. Agregar k a L

**Terminación:** Cuando L consta de dos nodos i, j, y el borde entre ellos de longitud  $d_{ij}$ , agregue el nodo raíz como padre de i y j.

## Resumen de Métodos a Distancia Pros y Contras

Se ha demostrado que los métodos descritos anteriormente capturan muchas características interesantes de las relaciones filogenéticas, y suelen ser muy rápidos en el sentido algorítmico. Sin embargo, cierta información ciertamente se pierde en la matriz de distancia, y típicamente solo se propone un solo árbol. Se pueden cometer errores graves, como la atracción de ramas largas, cuando se violan los supuestos básicos sobre la tasa de mutación, etc. Por último, los métodos a distancia no hacen inferencia sobre la historia de un sitio en particular, y por lo tanto no hacen sugerencias sobre el estado ancestral de una secuencia.

---

26.3: Métodos basados en la distancia is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 26.3: Distance Based Methods has no license indicated.

## 26.4: Métodos basados en caracteres

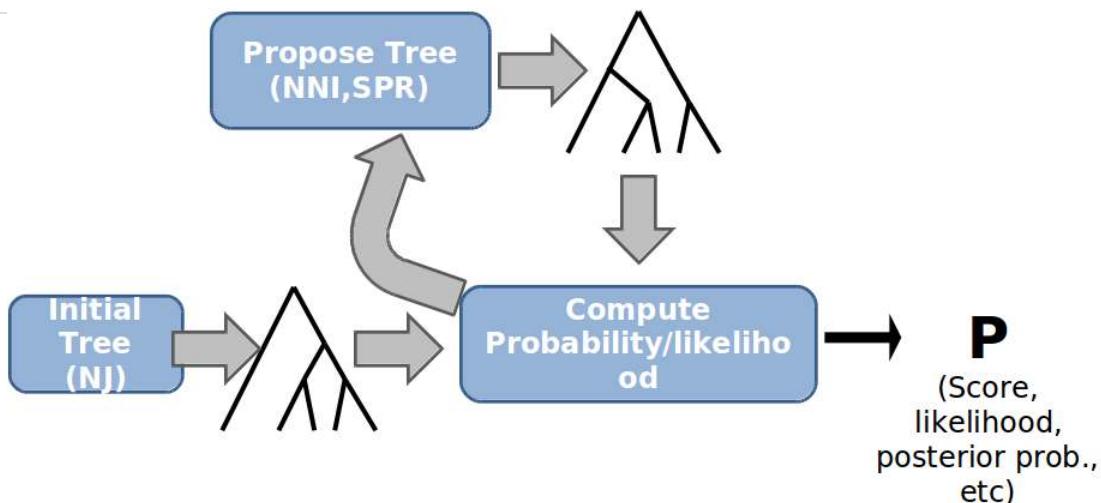


Figura 26.14: Una visión general de los métodos basados en caracteres

En los métodos basados en caracteres, el objetivo es crear primero un algoritmo válido para puntuar la probabilidad de que un árbol dado produzca las secuencias observadas en sus hojas, luego buscar a través del espacio de posibles árboles un árbol que maximice esa probabilidad. Buenos algoritmos para la puntuación de árboles, y aunque la búsqueda en el espacio de los árboles es teóricamente NP-dura (Debido a la gran cantidad de árboles posibles), los métodos de búsqueda heurística tratables pueden en muchos casos encontrar buenos árboles. Primero discutiremos los algoritmos de puntuación de árboles, luego las técnicas de búsqueda.

### Apuntar

Hay dos algoritmos principales para la puntuación de árboles. El primer enfoque, que llamaremos reconstrucción de parsimonia, se basa en la navaja de Occam, y puntuá una topología basada en el número mínimo de mutaciones que implica, dadas las secuencias (conocidas) en las hojas. Este método es simple, intuitivo y rápido. El segundo enfoque es un método de máxima verosimilitud que puntuá árboles modelando explícitamente la probabilidad de observar las secuencias en las hojas dada una topología de árbol.

### Parsimonia

Conceptualmente, este método es sencillo. Simplemente asigna un valor de para cada par de bases en cada nodo ancestral de tal manera que se minimiza el número de sustituciones. La puntuación es entonces solo la suma sobre todos los pares de bases de ese número mínimo de mutaciones en cada par de bases. (Recordemos que el objetivo final es encontrar un árbol que minimice ese marcador.)

Para reconstruir las secuencias ancestrales en los nodos internos del árbol, el algoritmo primero explora las secuencias foliares (conocidas), asignando un conjunto de bases en cada nodo interno en función de sus hijos. A continuación, itera hacia abajo en el árbol, escogiendo bases de los conjuntos permitidos en cada nodo, esta vez en función de los padres del nodo. A continuación se ilustra este algoritmo en detalle (tenga en cuenta que hay  $2N-1$  nodos totales, indexados desde la raíz, de tal manera que los nodos hoja conocidos tienen índices  $N-1$  a  $2N-1$ ):

Given a tree, and an alignment column

Label internal nodes to minimize the number of required substitutions

### Initialization:

Set cost  $C = 0$ ;  $k = 2N - 1$

### Iteration:

If  $k$  is a leaf, set  $R_k = \{x^k[u]\}$

If  $k$  is not a leaf,

Let  $i, j$  be the daughter nodes;

Set  $R_k = R_i \cap R_j$  if intersection is nonempty

Set  $R_k = R_i \cup R_j$ , and  $C += 1$ , if intersection is empty

### Termination:

Minimal cost of tree for column  $u, = C$

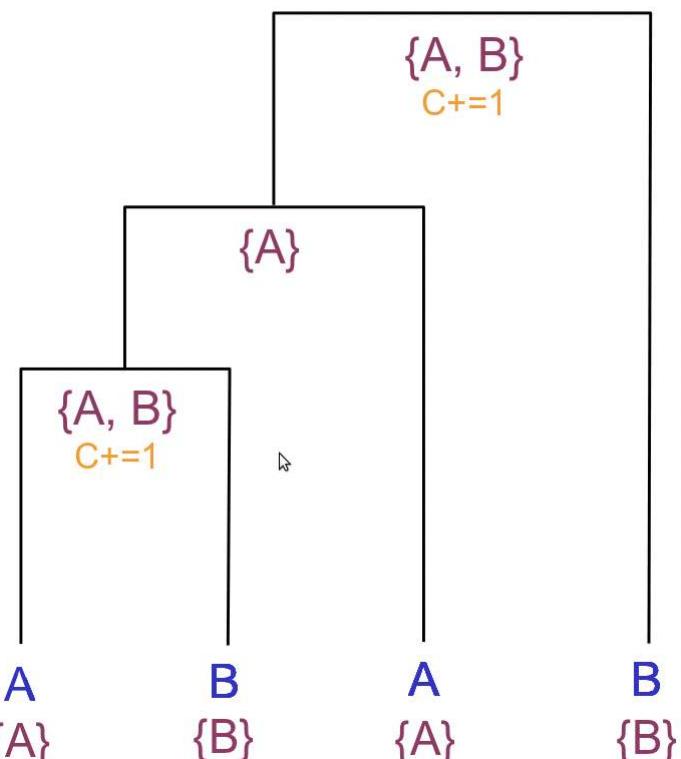


Figura 26.15: Puntuación de parsimonia: unión e intersección

### Traceback:

1. Choose an arbitrary nucleotide from  $R_{2N-1}$  for the root
  
2. Having chosen nucleotide  $r$  for parent  $k$ ,  
 If  $r \in R_i$  choose  $r$  for daughter  $i$   
 Else, choose arbitrary nucleotide from  $R_i$

Easy to see that this traceback produces some assignment of cost  $C$

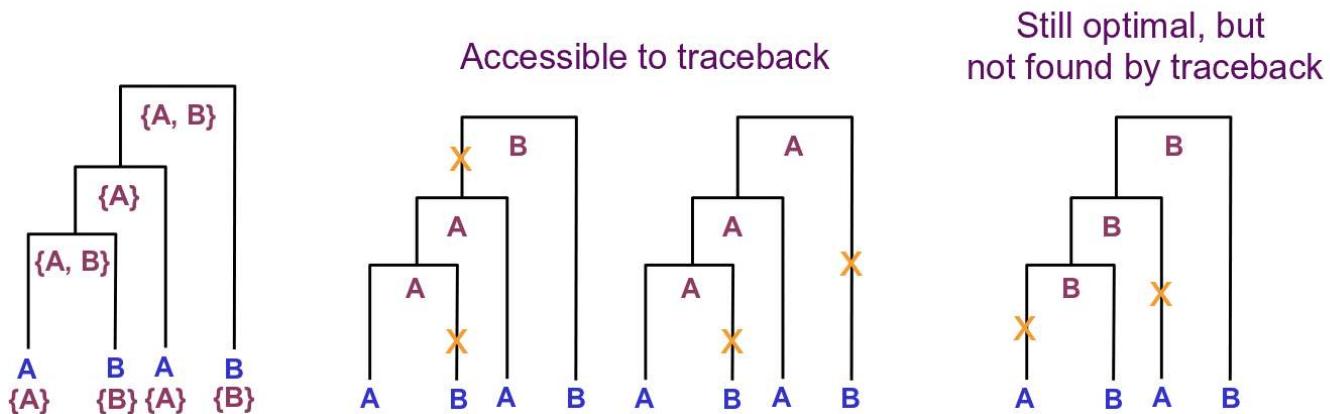


Figura 26.16: Rastreo de parsimonia para encontrar nucleótidos ancestrales

|   | M | R | B1 | H | B2 | D | B3 |
|---|---|---|----|---|----|---|----|
| A | 0 | 1 | 1  | 0 | 1  | 1 | 2  |
| C | 1 | 1 | 2  | 1 | 3  | 1 | 4  |
| G | 1 | 0 | 1  | 1 | 2  | 0 | 2  |
| T | 1 | 1 | 2  | 1 | 3  | 1 | 4  |



- Each cell  $(N, C)$  represents the min cost of the subtree rooted at  $N$ , if the label at  $N$  is  $C$ .
- Update table by walking up the tree from the leaves to the root, remembering max choices.
- **Traceback from root to leaves**

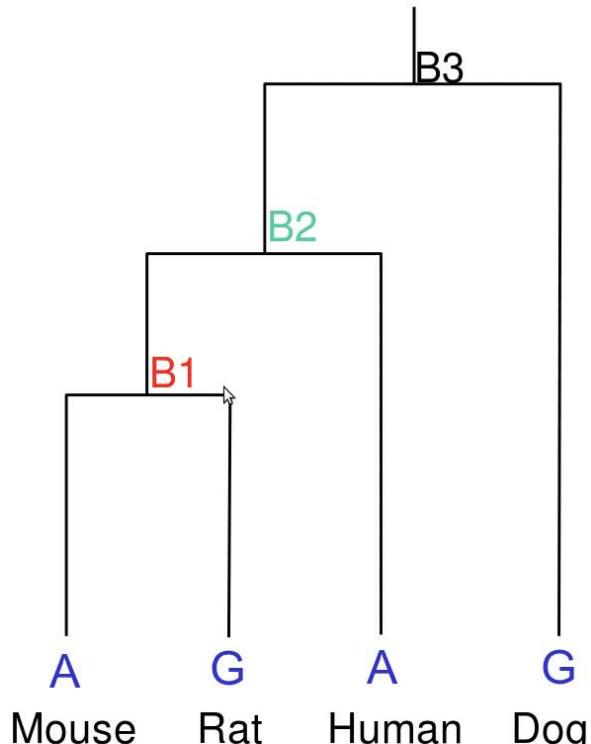


Figura 26.17: Puntuación de parsimonia por programación dinámica

Como mencionamos antes, este método es sencillo y rápido. Sin embargo, esta simplicidad puede distorsionar las puntuaciones que asigna. Por un lado, el algoritmo aquí presentado asume que un par de bases dado se somete a una sustitución a lo largo de como máximo una rama de un nodo dado, lo que puede llevarlo a ignorar muy probablemente secuencias internas que violen esta suposición. Además, este método no modela explícitamente el tiempo representado a lo largo de cada borde y, por lo tanto, no puede explicar la mayor probabilidad de una sustitución a lo largo de bordes que representan una larga duración temporal, o la posibilidad de diferentes tasas de mutación a través del árbol. Los métodos de máxima verosimilitud resuelven ampliamente estas deficiencias y, por lo tanto, se usan más comúnmente para la puntuación

### Máxima probabilidad - algoritmo de pelado

Al igual que con los métodos generales de Máxima verosimilitud, este algoritmo puntúa un árbol de acuerdo con la probabilidad conjunta (logarítmica) de observar los datos y el árbol dado, es decir  $P(D, T)$ . El algoritmo de peeling vuelve a considerar pares de bases individuales y asume que todos los sitios evolucionan de manera independiente. Al igual que en el método de parsimonia, este algoritmo considera todos los pares de bases independientemente: calcula la probabilidad de observar los caracteres dados en cada par base en los nodos hoja, dado el árbol, un conjunto de longitudes de rama, y la asignación de máxima verosimilitud de la secuencia interna, luego simplemente multiplica esta probabilidad sobre todos los pares de bases para obtener la probabilidad total de observar el árbol. Tenga en cuenta que el modelado explícito de longitudes de rama es una diferencia con respecto al enfoque anterior.

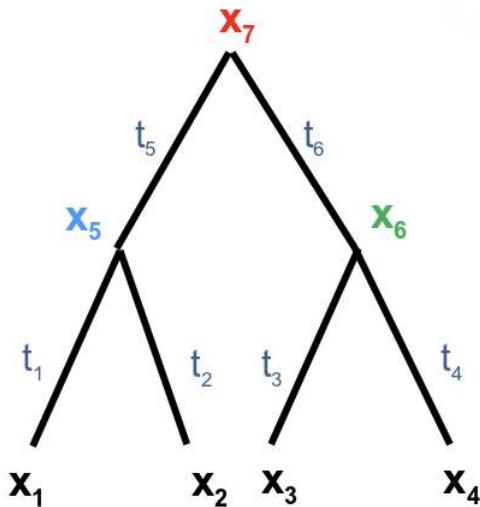


Figura 26.18: Un árbol a puntuar usando el algoritmo de peeling.  $n=4$

Aquí cada nodo tiene un carácter  $x_i$  y  $t_i$  es la longitud de rama correspondiente de su parente. Tenga en cuenta que ya conocemos los valores  $x_1, x_2 \dots x_n$ , por lo que son constantes, pero  $x_{n+1}, \dots x_{2n-1}$  son caracteres desconocidos en nodos ancestrales que son variables a las que asignaremos valores de máxima verosimilitud. (También tenga en cuenta que hemos adoptado un esquema de indexación de hojas a raíz para los nodos, lo contrario del esquema que usamos antes). Queremos computar  $P(x_1 x_2 \dots x_n | T)$ . Para ello sumamos sobre todas las combinaciones posibles de valores en los nodos ancestrales. A esto se le llama marginación. En este ejemplo particular

$$P(x_1 x_2 x_3 x_4 | T) = \sum_{x_5} \sum_{x_6} \sum_{x_7} P(x_1 x_2 \dots x_7 | T)$$

Aquí hay  $4^{n-1}$  términos, pero podemos usar el siguiente truco de factorización:

$$= \sum_{x_7} \left[ P(x_7) \left( \sum_{x_6} P(x_5 | x_7, t_5) P(x_1 | x_5, t_1) P(x_2 | x_5, t_2) \right) \right. \\ \left. \left( \sum_{x_6} P(x_6 | x_7, t_6) P(x_3 | x_6, t_3) P(x_4 | x_6, t_4) \right) \right]$$

Aquí asumimos que cada rama evoluciona de manera independiente. Y la probabilidad  $P(b|c, t)$  denota la probabilidad de que la base  $c$  muta a la base  $b$  dado el tiempo  $t$ , que se obtiene esencialmente del modelo Jukes Cantor o algún modelo más avanzado discutido anteriormente. A continuación podemos mover los factores que son independientes de la variable de suma fuera de la suma. Eso da:

$$= \sum_{x_7} \left[ P(x_7) \left( \sum_{x_6} P(x_5 | x_7, t_5) P(x_1 | x_5, t_1) P(x_2 | x_5, t_2) \right) \right. \\ \left. \left( \sum_{x_6} P(x_6 | x_7, t_6) P(x_3 | x_6, t_3) P(x_4 | x_6, t_4) \right) \right]$$

Sea  $T_i$  el subárbol debajo de  $i$ . En este caso, nuestro array de programación dinámica  $2n-1 \times 4$  calcula  $L[i, b]$ , la probabilidad  $P(T_i | x_i = b)$  de observar  $T_i$ , si el nodo  $i$  contiene base  $b$ . Entonces queremos calcular la probabilidad de observar  $T = T_{2n-1}$ , que es

$$\sum_b P(x_{2n-1} = b) L[2n-1, b]$$

Tenga en cuenta que por cada nodo ancestral  $i$  y su hijo  $j, k$ , tenemos

$$L[i, b] = \left( \sum_c P(c | b, t_j) L[j, c] \right) \left( \sum_c P(c | b, t_k) L[k, c] \right)$$

Sujeto a las condiciones iniciales para los nodos hoja, es decir, para  $i \leq n$ :

$L[i, b] = 1$  si  $x_i = b$  y 0 en caso contrario

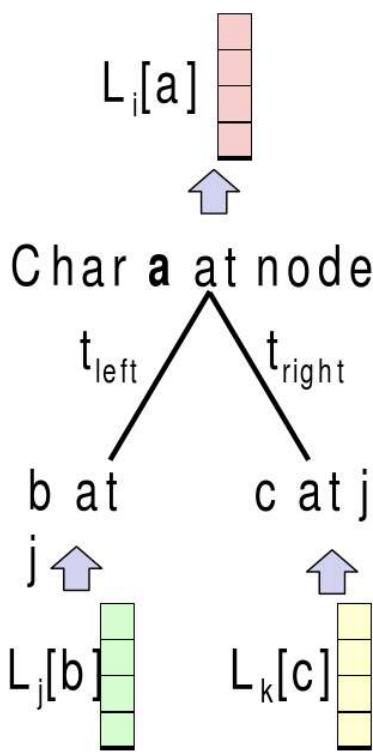


Figura 27.19: La recursión

Tenga en cuenta que todavía no tenemos los valores  $P(x_{2n-1} = b)$ . Por lo general se asigna por igual o de alguna distribución previa, pero no afecta mucho a los resultados. El paso final es, por supuesto, multiplicar todas las probabilidades para sitios individuales para obtener la probabilidad de observar el conjunto de secuencias completas. Además, una vez que hemos asignado los valores de máxima verosimilitud para cada nodo interno dada la estructura de árbol y el conjunto de longitudes de rama, podemos multiplicar la puntuación resultante por algunas probabilidades previas de la estructura de árbol y el conjunto de longitudes de rama, que a menudo se generan utilizando modelos explícitos de procesos evolutivos, como el proceso Yule o modelos de nacimiento-muerte como el proceso Moran. El resultado de esta multiplicación final se denomina probabilidad a posteriori, utilizando el lenguaje de inferencia bayesiana. La complejidad general de este algoritmo es  $O(nmk^2)$  donde  $n$  es el número de hojas (taxones),  $m$  es la longitud de la secuencia y  $k$  es el número de caracteres.

Hay advantages y desventajas de este algoritmo. Tales como

#### Ventajas:

1. Inherenteamente estadístico y basado en modelos evolutivos.
2. Por lo general, el más consistente de los métodos disponibles.
3. Se utiliza tanto para análisis de caracteres como de velocidad
4. Se puede utilizar para inferir las secuencias de los ancestros extintos.
5. Dar cuenta de los efectos de longitud de ramificación en árboles desequilibrados.
6. Secuencias de nucleótidos o aminoácidos, otro tipo de datos.

#### Desventajas:

1. No tan simples e intuitivos como muchos otros métodos.
2. Intenso computacionalmente Limitado por, número de taxones y longitud de secuencia).
3. Al igual que la parsimonia, puede ser engañada por altos niveles de homoplasía.
4. Las violaciones de los supuestos del modelo pueden dar lugar a árboles incorrectos.

## Buscar

Una búsqueda exhaustiva sobre el espacio de todos los árboles sería extremadamente costosa. El número de árboles de enraizamiento completo con  $n + 1$  hojas es el número enésimo catalán

$$\begin{aligned} C_n = & \frac{1}{n+1} \left( \begin{array}{c} c \\ 2 \\ n \end{array} \right) \\ & \approx \frac{4^n}{n^{3/2}} \sqrt{\pi} \end{aligned}$$

Además, debemos calcular el conjunto de verosimilitud máxima de longitudes de rama para cada uno de estos árboles. Por lo tanto, es un problema difícil de NP maximizar la puntuación absolutamente para todos los árboles. Afortunadamente, los algoritmos de búsqueda heurística generalmente pueden identificar buenas soluciones en el espacio del árbol. El marco general para dichos algoritmos de búsqueda es el siguiente:

**Inicialización:** Tome algún árbol como base de iteración (aleatoriamente o según algún otro anterior, o desde los algoritmos directos basados en la distancia).

**Propuesta:** Proponer un nuevo árbol modificando ligeramente al azar el árbol actual.

**Puntuación:** Calificar la nueva propuesta de acuerdo con los métodos descritos anteriormente.

**Seleccionar:** Seleccione aleatoriamente el nuevo árbol o el árbol viejo (probabilidades correspondientes según la relación de puntuación (verosimilitud)).

**Iterar:** Repita el paso de propuesta a menos que se cumplan algunos criterios de terminación (alguna puntuación umbral o número de pasos alcanzados).

La idea básica aquí es la suposición heurística de que las puntuaciones de árboles estrechamente relacionados son similares, de manera que se pueden obtener buenas soluciones mediante la optimización local sucesiva, que se espera converja hacia una buena solución general.

### Propuesta de árbol

Un método para modificar árboles es el Nearest Neighbor Exchange (NNI), ilustrado a continuación.

Figura 27.20: Un paso de unidad usando el esquema de intercambio de vecinos más cercanos

Otro método común, no descrito aquí, es la bisección y unión de árboles (TBJ). El criterio importante para tales reglas de propuesta es que:

1. a) El espacio arbóreo debe estar conectado, es decir, cualquier par de árboles debe ser obtenible entre sí mediante propuestas sucesivas.
2. b) Una nueva propuesta individual debe estar suavemente cercana a la original. Para que sea más probable que sea una buena solución en virtud de la proximidad a una buena solución ya descubierta. Si los pasos individuales son demasiado grandes, el algoritmo puede alejarse de una solución ya descubierta (también depende del paso de selección). En particular, señalar que la medida de similitud por la que la medida de estos tamaños de paso es precisamente la diferencia en las puntuaciones de verosimilitud asignadas a los dos árboles.

### Selección

Elegir si adoptar o no una propuesta dada, como el proceso de generar la propuesta misma, es inherentemente heurístico y varía. Una regla general es:

1. Si el nuevo tiene una mejor puntuación, acéptalo siempre.
2. Si tiene peor puntuación, debería haber alguna probabilidad de seleccionarlo, de lo contrario el algoritmo pronto se fijará en unos mínimos locales, ignorando mejores alternativas un poco lejos.
3. No debe haber demasiada probabilidad de seleccionar una nueva propuesta peor, de lo contrario, corre el riesgo de rechazar una buena solución conocida.

Es la compensación entre los pasos 2 y 3 lo que determina una buena regla de selección. Metropolis Hastings es un Método Montecarlo de la Cadena de Markov (MCMC) que define reglas específicas para explorar el espacio estatal de una manera que lo convierte en una muestra de la distribución posterior. Estos algoritmos funcionan algo bien en la práctica, pero no hay garantía para encontrar el árbol apropiado. Entonces se usa un método conocido como bootstrapping, que básicamente es ejecutar el algoritmo una y otra vez usando subconjuntos de los pares base en las secuencias de hoja., luego favoreciendo árboles globales que coincidan con las topologías generadas usando solo estas subsecuencias.

---

26.4: [Métodos basados en caracteres](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [26.4: Character-Based Methods](#) has no license indicated.

## 26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido

Se debe hacer un punto especial sobre las distancias. Dado que las distancias se calculan típicamente entre secuencias génicas alineadas, la mayoría de los métodos actuales de reconstrucción de árboles se basan en genes muy conservados, ya que los genes no conservados no darían información sobre especies sin esos genes. Esto provoca la ignorancia de datos que por lo demás serían útiles. Por lo tanto, hay algunos algoritmos que intentan tomar en cuenta genes menos conservados en la reconstrucción de árboles pero estos algoritmos tienden a llevar mucho tiempo debido a la naturaleza NP-dura de reconstruir árboles.

Adicionalmente, las secuencias alineadas aún no son explícitas en lo que respecta a los eventos que las crearon. Es decir, las combinaciones de eventos de especiación, duplicación, pérdida y transferencia génica horizontal (hgt) son fáciles de mezclar porque solo están disponibles las secuencias de ADN actuales. (véase [11] para un comentario sobre tales cuestiones teóricas) Una duplicación seguida de una pérdida sería muy difícil de detectar. Además, una duplicación seguida de una especiación podría parecerse a un evento HGT. Incluso las probabilidades de que ocurran eventos siguen siendo cuestionadas, especialmente los eventos de transferencia de genes horizontales.

Otro problema es que a menudo se concatenan múltiples secuencias marcadoras y la secuencia concatenada se usa para calcular la distancia y crear árboles. Sin embargo, este enfoque asume que todos los genes concatenados tenían la misma historia y existe debate sobre si este es un enfoque válido dado que eventos como hgt y duplicaciones como se describió anteriormente podrían haber ocurrido de manera diferente para diferentes genes. [8] es un artículo que muestra cómo diferentes filogenéticos se encontraron relaciones dependiendo de si el árbol fue creado usando múltiples genes concatenados juntos o si fue creado usando cada uno de los genes individuales. Por el contrario, [4] adicional afirma que si bien la hgt es prevalente, los ortólogos utilizados para la reconstrucción filogenética son consistentes con un solo árbol de la vida. Estos dos temas indican que existe un claro debate en el campo sobre una manera no arbitraria de definir especies e inferir relaciones filogenéticas para recrear el árbol de la vida.

---

26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [26.5: Possible Theoretical and Practical Issues with Discussed Approach](#) has no license indicated.

## 26.6: Hacia el proyecto final

### Ideas de Proyectos

1. Crear mejores modelos de distancia como tomar en cuenta genes duplicados o pérdida de genes. También puede ser posible analizar secuencias para regiones codificantes de péptidos y calcular distancias basándose también en cadenas peptídicas.
2. Crear un algoritmo de búsqueda más rápido/más preciso para convertir distancias en árboles.
3. Analizar secuencias para calcular probabilidades de especiación, duplicación, pérdida y eventos de transferencia genética horizontal.
4. Ampliar un algoritmo que busca HGT para buscar especies extintas. Un posible uso para los HGT es que si un programa fuera a inferir HGT entre diferentes momentos, podría significar que había una especiación donde una rama está ahora extinguida (o aún no descubierta) y esa rama había causado un HGT a la otra rama existente.

### Conjuntos de datos del proyecto

1. 1000 Genomas Proyecto <http://www.1000genomes.org/>
2. Microbes en línea <http://microbesonline.org/>

---

26.6: Hacia el proyecto final is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 26.6: Towards final project has no license indicated.

## 26.7: ¿Qué hemos aprendido?

En este capítulo, hemos aprendido diferentes métodos y enfoques para reconstruir árboles filogenéticos a partir de datos de secuencia. En el próximo capítulo se discutirá su aplicación en árboles genéticos y especies arbóreas y la relación entre ambos, así como modelar filogenias entre poblaciones dentro de una especie y entre especies estrechamente relacionadas.

---

26.7: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [26.7: What Have We Learned?](#) has no license indicated.

## Bibliografía

- 
- [1] Proyecto de 1000 genomas.
  - [2] et al Ciccarelli, Francesca. Hacia la reconstrucción automática de un árbol de la vida altamente resuelto. *Ciencia*, 311, 2006.
  - [3] Tal Dagan y William Martin. El árbol del uno por ciento. *Biología del Genoma*, Nov 2006.
  - [4] Ochman Howard Daubin Vincent, Moran Nancy A. La filogenética y la cohesión de los genomas bacterianos. *Ciencia*, 301, 2003.
  - [5] A.J. Enright, S. Van Dongen y C. A. Ouzounis. Un algoritmo eficiente para la detección a gran escala de familias de proteínas. *Investigación de ácidos nucleicos*, 30 (7) :1575—1584, Abr 2002.
  - [6] Stephanie Guindon y Olivier Gascuel. Un algoritmo simple, rápido y preciso para estimar grandes filogenias por máxima verosimilitud. *Biología de Sistemas*, 52 (5) :696—704, 2003.
  - [7] Sanderson MJ. r8s: Inferir tasas absolutas de evolución molecular y tiempos de divergencia en ausencia de un reloj molecular. *Bioinformática*, 19 (2) :301—302, ene 2003.
  - [8] R. Thane Papke, Olga Zhaxybayeva, Edward J Fiel, Katrin Sommerfeld, Denise Muise y W. Ford Doolittle. Búsqueda de especies en haloarchaea. *PNAS*, 104 (35) :14092—14097, 2007.
  - [9] Pere Puigbo, Yuri I Wolf y Eugenio V Koonin. Búsqueda de un 'árbol de la vida' en la espesura del bosque filogenético. *Revista de Biología*, 8 (59), julio de 2009.
  - [10] Sagi Snir, Yuri I Wolf y Eugene V Koonin. Marcapasos universales de la evolución del genoma. *PLoS biología tacional*, 8 (11), 2012.
  - [11] Douglas L Theobald. Una prueba formal de la teoría de la ascendencia común universal. *Naturaleza*, 465:219 —222, 2010.
- 

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 27: Filogenómica II

- 27.1: Introducción
- 27.2: SPIDR
- 27.3: Gráficas de Recombinación Ancestral
- 27.4: Conclusión
- 27.05: Inferir ortológicos
- 27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética
- 27.6: Reconstrucción
- 27.7: Modelización de Frecuencias de Poblaciones y Alelos
- 27.10 ¿Qué hemos aprendido?
- 27.9 Lectura adicional
- Bibliografía

---

27: Filogenómica II is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 27.1: Introducción

En el capítulo anterior, cubrimos técnicas de razonamiento sobre la evolución en términos de árboles de descenso. Los algoritmos que cubrimos para la construcción de árboles, la UPGMA y la unión de vecinos asumieron que estábamos comparando secciones completamente alineadas de secuencias.

En esta sección, presentamos modelos adicionales para el uso de árboles filogenéticos en diferentes contextos. Aquí aclaramos las diferencias entre especies y árboles genéticos. Luego cubrimos un marco llamado reconciliación que nos permite combinar efectivamente los dos mediante el mapeo de árboles genéticos en árboles de especies. Este mapeo nos da un medio para inferir eventos de duplicación y pérdida de genes.

También presentaremos una perspectiva filogenética para el razonamiento sobre la genética de poblaciones. Dado que la genética poblacional se ocupa de eventos de mutación relativamente recientes, ofrecemos el modelo Wright-Fisher como una herramienta para representar cambios en poblaciones enteras. Desafortunadamente, cuando se trata de datos del mundo real, generalmente solo somos capaces de secuenciar genes de los descendientes vivos actuales de un grupo. Como remedio a esta deficiencia, cubrimos el modelo Coalescent, que se puede pensar como un análogo de Wright-Fisher invertido en el tiempo.

Mediante el uso de la coalescencia, ganamos nuevos medios para estimar los tiempos de divergencia y el tamaño de la población en múltiples especies. Al final del capítulo, abordamos brevemente los desafíos del uso de árboles para modelar eventos de recombinación y resumimos el trabajo reciente en el campo junto con las fronteras abiertas a la exploración.

---

27.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.1: Introduction has no license indicated.

## 27.2: SPIDR

### Antecedentes

Como se presenta en la información complementaria para SPIDIR, una familia de genes es el conjunto de genes que son descendientes de un solo gen en el ancestro común más reciente (MRCA) de todas las especies bajo consideración. Además, las secuencias genéticas experimentan evolución a múltiples escalas, es decir, a nivel de pares de bases, y a nivel de genes. En el contexto de esta conferencia, dos genes son ortólogos si su MRCA es un evento de especiación; dos genes son parálogos si su MRCA es un evento de duplicación.

En la era genómica, a menudo se conoce la especie de genes modernos; los genes ancestrales se pueden inferir reconciliando árboles de genes y especies. Una reconciliación mapea cada nodo de árbol genético a un nodo de árbol de especies. Una técnica común es realizar la Reconciliación Máxima de Parsimonia (MPR), la cual encuentra la reconciliación R implicando el menor número de duplicaciones o pérdidas utilizando la recursión sobre los nodos internos v de un árbol genético G. MPR puño mapea cada hoja del árbol genético a la hoja de especie correspondiente del árbol de especies. Luego los nodos internos de G se mapean recursivamente:

$$R(v) = \text{MRCA}(R(\text{right}(v)), R(\text{left}(v)))$$

Si un evento de especiación y su nodo ancestral se mapean al mismo nodo en el árbol de la especie. Entonces el nodo ancestral debe ser un evento de duplicación. Usando MPR, la precisión del árbol genético es crucial. Los árboles genéticos subóptimos pueden conducir a un exceso de eventos de pérdida y duplicación. Por ejemplo, si solo una rama está fuera de lugar (como en ??) luego la conciliación infiere 3 pérdidas y 1 evento de duplicación. En [6], los autores muestran que los métodos actuales contemporáneos del árbol genético funcionan mal (60% de precisión) en genes individuales. Pero si tenemos genes concatenados más largos, entonces la precisión puede subir hacia el 100%. Además, los genes que evolucionan muy rápida o lentamente portan menos información en comparación con secuencias moderadamente divergentes (40-50% de identidad de secuencia), y tienen un desempeño correspondientemente peor. Como lo corroboran las simulaciones, los genes individuales carecen de información sucia para reproducir la especie correcta de árbol. Los genes promedio son demasiado cortos y contienen muy pocos caracteres filogenéticamente informativos. Mientras que muchos algoritmos de construcción temprana de árboles genéticos ignoraron la información de especies, algoritmos como SPIDIR capitalizan la idea de que el árbol de especies puede proporcionar información adicional que puede ser aprovechada para la construcción de árboles génicos. La sintonía se puede usar para probar independientemente la precisión relativa de las reconstrucciones de árboles génicos diferentes. Esto se debe a que los bloques sintéticos son regiones del genoma donde los organismos recientemente divergentes tienen el mismo orden de genes, y contienen mucha más información que los genes individuales.

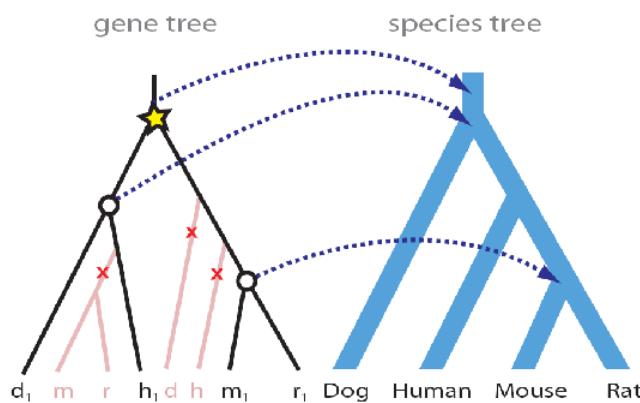


Figura 27.22: Reconciliación MPR de genes y especies arbóreas.

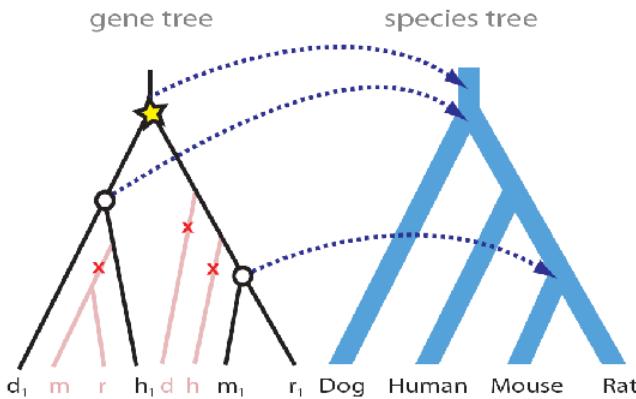


Figura 27.23: Imprecisiones en el árbol génico.

Ha habido una serie de algoritmos filogenómicos recientes que incluyen: RIO [2], que utiliza la unión de vecinos (NJ) y el bootstrapping para lidiar con las incognencias, Orthostrapper [7], que usa NJ y se reconcilia con un árbol de especies vagas, TreeFam [3], que utiliza la curación humana de árboles genéticos, así como muchos otros. Varios algoritmos toman una pista más similar a SPIDIR [6], incluyendo [4], un algoritmo de reconciliación probabilística [8], un método bayesiano con reloj, [9], y método de parsimonia usando especies arbóreas, así como desarrollos más recientes: [1] un método bayesiano con reloj relajado y [5], un método bayesiano con tasas relajadas específicas de genes y especies (una extensión a SPIDIR).

## Método y Modelo

SPIDIR ejemplifica un algoritmo iterativo para la construcción de árboles genéticos usando el árbol de especies. En SPIDIR, los autores definen un modelo generativo para la evolución del árbol genético. Esto consiste en un previo para topología de árbol genético y longitudes de ramas. SPIDIR utiliza un proceso de nacimiento y muerte para modelar duplicaciones y pérdidas (lo que informa al anterior sobre la topología) y luego aprende las tasas de sustitución específicas de genes y especies específicas (que informan la anterior sobre la longitud de las ramas). SPIDIR es un método Maximum a posteriori (MAP) y, como tal, disfruta de varios criterios de optimismo agradables.

En cuanto al problema de estimación, el modelo SPIDIR completo aparece de la siguiente manera:

$$\operatorname{argmax} L, T, RP(L, T, R | D, S, \Theta) = \operatorname{argmax} L, T, RP(D | T, L)P(L | T, R, S, \Theta)P(T, R | S, \Theta)$$

Los parámetros en la ecuación anterior son: D = datos de alineación, L = longitud de rama T = topología de árbol genético, R = reconciliación, S = árbol de especies (expresado en tiempos),  $\Theta$  = (parámetros específicos de genes y especies [estimados mediante entrenamiento EM],  $\mu$  dup/parámetros de pérdida)). Este modelo se puede entender a través de los tres términos en la expresión de la mano derecha, a saber:

1. el modelo de secuencia—  $P(D|T, L)$ . Los autores utilizaron el modelo HKY común para las sustituciones de secuencias, que unifica el modelo de dos parámetros de Kimura para transiciones y transversiones con el modelo de Felsenstein donde la tasa de sustitución depende de la frecuencia de equilibrio de nucleótidos.
2. el primer término previo, para el modelo de tasas—  $P(L|T, R, S, \Theta)$ , que los autores calculan numéricamente después de aprender las tasas específicas de especies y genes.
3. el segundo término previo, para el modelo de duplicación/pérdida—  $P(T, R|S, \Theta)$ , que los autores describen mediante un proceso de nacimiento y muerte.

Tener un modelo de tasas es muy útil el modelo de tasas, ya que las tasas de mutación son bastante variables entre los genes. En la conferencia, vimos cómo las tasas estaban bien descritas por una descomposición en tasas específicas de genes y especies. En la conferencia vimos que una distribución gamma inversa parece parametrizar las tasas de sustitución específica del gen, y se nos dijo que una distribución gamma aparentemente captura las tasas de sustitución específica de especies. Contabilizar las tasas específicas de genes y especies permite a SPIDIR construir árboles genéticos con mayor precisión que los métodos anteriores. Se puede elegir un conjunto de entrenamiento para parámetros de tasa de aprendizaje de árboles genéticos que son congruentes con el árbol de la

especie. Una preocupación algorítmica importante para las reconstrucciones de árboles genéticos es idear un método de búsqueda rápida de árboles. En la conferencia, vimos cómo la búsqueda en árboles podría acelerarse calculando solo el ArgMaxL completo,  $T, RP(L, T, R|D, S, \Theta)$  para árboles con altas probabilidades previas. Esto se logra a través de un pipeline computacional donde en cada iteración 100s de árboles son propuestos por alguna heurística. La topología anterior  $P(T, R|D, S, \Theta)$  se puede calcular rápidamente. Esto se utiliza como un filtro donde solo se seleccionan las topologías con altas probabilidades previas como candidatas para el cálculo de verosimilitud completa.

El desempeño de SPIDIR se probó en un conjunto de datos reales de 21 hongos. SPIDER recuperó más de 96% de los ortólogos de síntesis mientras que otros algoritmos encontraron menos de 65%. En consecuencia, SPIDER invocó mucho menos número de duplicaciones y pérdidas.

---

27.2: SPIDR is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.2: SPIDR has no license indicated.

## 27.3: Gráficas de Recombinación Ancestral

En la Figura 28.24 a, los dos cromosomas en la parte superior representan los cromosomas homólogos de un progenitor. El cromosoma rojo representa la información genética de la madre y el cromosoma azul representa la información genética del padre (de la generación de abuelos). Sin cruzamientos (recombinación), el progenitor transmitirá la información genética roja o azul a la descendencia. En realidad, la recombinación ocurre durante la meiosis para que uno de los padres transmita alguna información genética de ambos abuelos, transmitiendo efectivamente una mejor representación de la información genética del parente.

En cada generación, un evento de recombinación puede ocurrir en cualquier loci. La historia evolutiva de la recombinación se puede rastrear a través de una gráfica secuencial de árboles, de tal manera que el árbol  $i$ -ésimo en la gráfica representa la recombinación en el  $i$ -ésimo locus.

Rellene esta sección con base en: [www.Eecs.Berkeley.edu/yss/pub/sh-jcbo5.pdf](http://www.Eecs.Berkeley.edu/yss/pub/sh-jcbo5.pdf) y las notas del curso de 2012. Se podría agregar más sobre este tema en el futuro

### El coalescente secuencial de Markov

El modelo coalescente secuencial de Markov aborda el papel de la recombinación en la construcción de árboles. Con recombinación involucrada, una secuencia puede tener dos progenitores, lo que complica la construcción. El modelo coalescente secuencial de Markov nos dice que moverse secuencialmente de izquierda a derecha es un enfoque más simple y mucho más eficiente para analizar el árbol; el enfoque esencialmente divide el árbol en árboles locales y los superpone para describir eventos de recombinación. Se pueden leer más en el siguiente trabajo:

Elaborar sobre las complejidades del modelo en sí: <http://www.ncbi.nlm.nih.gov/pubmed/21270390>

---

27.3: Gráficas de Recombinación Ancestral is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.3: Ancestral Recombination Graphs has no license indicated.

## 27.4: Conclusión

La incorporación de información de árboles de especies en el proceso de construcción de árboles genéticos mediante la introducción de genes separados y tasas de sustitución de especies permite reconstrucciones precisas de árboles génicos parsimoniosos. Las reconstrucciones previas de árboles genéticos probablemente sobreestimaron enormemente el número de eventos de duplicación y pérdida. La reconstrucción de árboles genéticos para familias numerosas sigue siendo un problema desafiante.

---

27.4: Conclusión is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.4: Conclusion has no license indicated.

## 27.05: Inferir ortológicos

Esta página se ha generado automáticamente porque un usuario ha creado una subpágina de esta página.

[27.05: Inferir ortológicos](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética

Hay dos árboles de uso común, Árbol de especies y Árbol de genes. En esta sección se explica cómo se pueden usar estos árboles y cómo encajar un árbol genético dentro de un árbol de especies (reconciliación).

### Especies Árbol

Especies arbóreas que muestran cómo diferentes especies evolucionaron entre sí. Estos árboles se crean usando caracteres morfológicos, evidencia fósil, etc. Las hojas de cada árbol están etiquetadas como especies y el resto del árbol muestra cómo se relacionan estas especies. Un ejemplo de árbol especie se muestra en la Figura 27.1. Nota: en conferencia se menciona que una especie puede ser pensada como una "bolsa de genes", es decir, el grupo de genes comunes entre los miembros de una especie.

### Árbol de genes

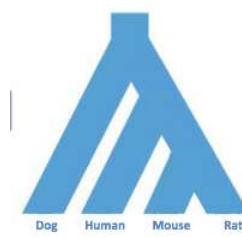


Figura 27.1: Especies arbóreas

Los árboles génicos son árboles que miran genes específicos en diferentes especies. Las hojas de los árboles génicos están marcadas con secuencias génicas o identificadores de genes asociados con secuencias específicas. La Figura 27.2 muestra un ejemplo de un árbol genético que tiene 4 genes (hojas). Las secuencias asociadas a cada gen se presentan en el lado derecho de la Figura 27.2.

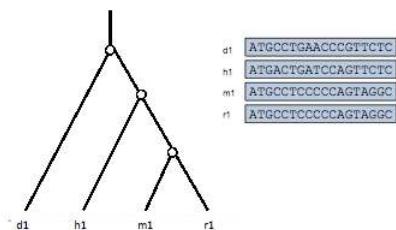


Figura 27.2: Árbol de genes

### Evolución de la Familia Génica

Los árboles genéticos evolucionan dentro de un árbol de especies. Un ejemplo de un árbol genético contenido en un árbol de especie se muestra en la Figura 27.3 a continuación.

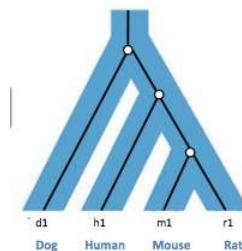


Figura 27.3: Árbol de genes dentro de un árbol de especies

En la siguiente subsección se explica cómo podemos encajar árboles genéticos dentro de árboles de una especie usando Reconciliation.

## Reconciliación

La reconciliación es un algoritmo que ayuda a comparar árboles genéticos con árboles genómicos ajustando un árbol genético dentro de un árbol de especies. Esto se hace mapeando los vértices en el árbol de genes a vértices en el árbol de la especie. Esta subsección se centrará en la Reconciliación, definiciones relacionadas, algoritmos (Conciliación de Parsimonia Máxima y SPIDIR) y ejemplos.

### Definiciones

Dos genes son ortólogos si su ancestro común más reciente (MRCA) es una especiación (dividiéndose en diferentes especies).

Los parálogos son genes cuyo MRCA es una duplicación.

La Figura 27.4 a continuación ilustra cómo se pueden representar estos tipos de genes en un árbol génico. El árbol de abajo tiene 4 nodos de especiación, una duplicación y una pérdida.

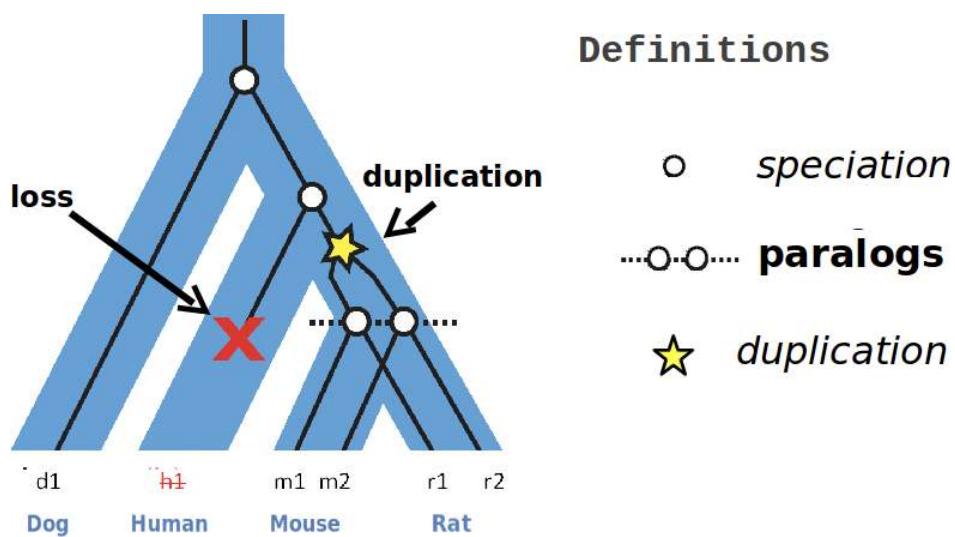


Figura 27.4: Evolución de la familia génica: Árboles genéticos y árboles de especies

Un diagrama de mapeo es un diagrama que muestra el mapeo de nodos desde el árbol génico hasta el árbol de especies. La Figura 27.5 muestra un ejemplo de un diagrama de mapeo.

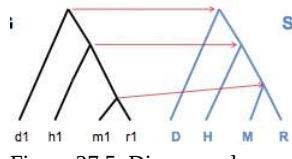


Figura 27.5: Diagrama de mapeo

Un diagrama de anidación muestra cómo se puede anidar el árbol genético dentro del árbol de la especie. Para cada diagrama de mapeo hay un diagrama de anidamiento. La Figura 27.6 muestra un ejemplo de un posible diagrama de anidamiento para el diagrama de mapeo de la Figura 27.5.

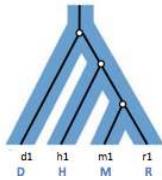


Figura 27.6: Diagrama de anidamiento

### Algoritmo de reconciliación máxima de parsimonia (MPR)

MPR es un algoritmo que ajusta un árbol genético en un árbol de especies mientras minimiza el número de duplicaciones y delecciones.

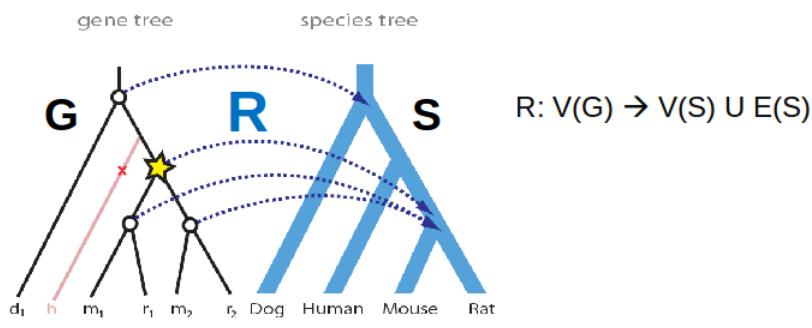


Figura 27.7: Conciliación máxima de parsimonia (MPR)

Dado un árbol genético y un árbol de especies, el algoritmo encuentra la reconciliación que minimiza el número de duplicaciones y delecciones. La Figura 27.7 anterior muestra un ejemplo de un posible mapeo de un árbol génico a un árbol de especie. La Figura 27.8 presenta el pseudocódigo para el algoritmo MPR. El caso base consiste en emparejar las hojas del árbol génico con las hojas del árbol de la especie; el algoritmo luego progresó por los vértices del árbol génico, dibujando una relación entre el MRCA de todas las hojas dentro del subárbol de un vértice dado y el vértice MRCA correspondiente en el árbol de la especie. En el pseudocódigo,  $I(G)$  representa el árbol de especies y  $L(G)$  representa el árbol génico.

#### Solve recursively:

- $R[v] = \text{species of } v \quad \text{if } v \in L(G)$
- $R[v] = \text{LCA}(\quad R[\text{right}(v)] \quad , \quad R[\text{left}(v)] \quad ) \quad \text{if } v \in I(G)$ 
  - LCA = “least common ancestor”
  - (also called “most recent common ancestor”)

#### Labeling events:

- $v$  is a dup if  $R[v] = R[\text{right}(v)]$  or  $R[\text{left}(v)]$
- Branch above  $v$  has at least one loss if  $R[\text{parent}(v)] \neq R[v]$  or  $\text{parent}[R[v]]$

Figura 27.8: Algoritmo recursivo de reconciliación de parsimonia máxima

Mapeamos las flechas lo más bajo posible, ya que un mapeo más bajo generalmente da como resultado menos eventos. Sin embargo, no podemos mapear demasiado bajo. Mapear demasiado bajo significa que estamos violando la restricción de que el MRCA de un nodo dado es al menos tan alto como el MRCA de sus hijos. Mapeamos lo más bajo posible sin violar las relaciones descendientes-ancestros. El algoritmo va recursivamente de abajo hacia arriba, comenzando desde las hojas. Dado que tomamos muestras de genes de especies conocidas para construir el árbol genético, existe un mapeo directo entre las hojas del árbol genético y las hojas del árbol de la especie. Para mapear a los antepasados, para cada nodo (subiendo recursivamente por el árbol) observamos al niño derecho y al niño izquierdo y tomamos el ancestro menos común (LCA) de las especies a las que mapean. Si un nodo se mapea a su hijo derecho o izquierdo, sabemos que hay una duplicación. Una rama esperada que no existe indica una pérdida.

#### Ejemplos de reconciliación

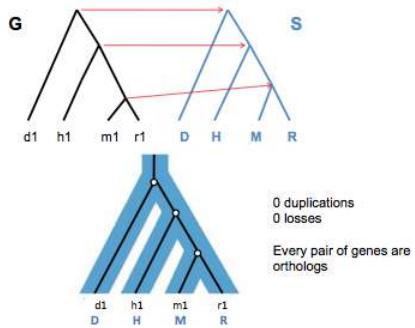


Figura 27.9: Ejemplo de reconciliación 1, caso de mapeo simple

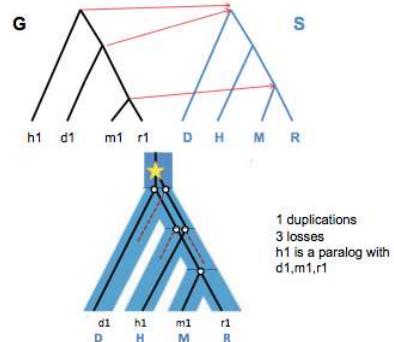


Figura 27.10: Reconciliación Ejemplo 2, reconciliación parsimoniosa para caso complejo

En la Figura 27.10, vemos una reconciliación parsimoniosa (número mínimo de pérdidas y duplicaciones) para un caso en el que los nodos del árbol de genes no pueden mapearse directamente a través. Esto es el resultado de las ubicaciones intercambiadas de h1 y d1 en el árbol génico; el ancestro menos común para d1, m1 y r1 es ahora el vértice raíz del árbol de la especie.

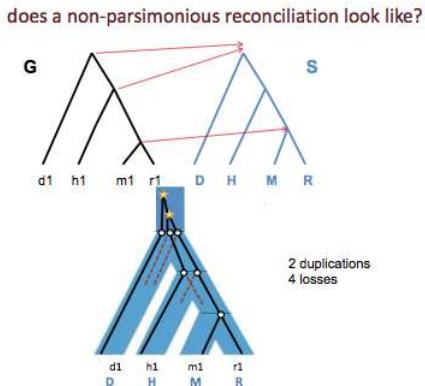


Figura 27.11: Reconciliación Ejemplo 3, reconciliación no parsimoniosa para caso complejo

La Figura 27.11 muestra una reconciliación no parsimoniosa. El mapeo parsimonioso para los mismos árboles se muestra en la Figura 27.9.

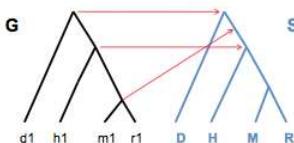


Figura 27.12: Ejemplo de reconciliación 4, reconciliación no válida

La Figura 27.12 muestra una reconciliación no válida. Esta reconciliación no es válida ya que no respeta las relaciones descendentes-ancestros. Para que esta reconciliación sea posible, el descendiente tendría que viajar atrás en el tiempo y ser creado ante su antepasado. Claramente, tal escenario sería imposible. Una conciliación válida debe satisfacer lo siguiente: **Si  $a < b$  en  $G$ , entonces  $R[a] \leq R[b]$  en  $S$ .**

### Interpretación de ejemplos de reconciliación

Los árboles genéticos, cuando se reconcilan con árboles de especies, ofrecen una visión significativa de los eventos evolutivos (es decir, duplicaciones y pérdidas). Las duplicaciones describen que el mismo gen se encuentra en loci separados -m2 o r2, en esta situación- y es un mecanismo importante para crear nuevos genes y funciones. Estas consecuencias evolutivas se dividen en tres categorías: no funcionalización, neofuncionalización y subfuncionalización. La no funcionalización es bastante común y hace que una de las copias, como era de esperar, simplemente no funcione. La neofuncionalización es cuando una de las copias desarrolla una función completamente nueva. La subfuncionalización es cuando las copias conservan diferentes partes (dividiendo la mano de obra, de alguna manera), y juntas, realizan la misma función.

En la Figura 4, vemos que se produjo un evento de duplicación antes de la divergencia de ratones y ratas como especie. Es por ello que vemos genes similares tanto en m1 como en m2, que representan dos loci separados. d2 y h2 no están incluidos en la gráfica porque en el gen que se está considerando no está presente en esos loci (ya que no se produjo ningún evento de duplicación), mientras que es tanto en m2 como r2.

Si el evento de duplicación hubiera ocurrido un nivel superior en la Figura 4, sin ver un h2 correspondiente en el árbol génico, esto implicaría una pérdida dentro de la rama h del árbol de la especie.

---

27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.5: Inferring Orthologs/Paralogs, Gene Duplication and Loss has no license indicated.

## 27.6: Reconstrucción

En la sección anterior aprendimos a comparar y combinar árboles genéticos y árboles de especies. En esta sección, utilizaremos esta información para reconstruir árboles genéticos y árboles de especies.

### Reconstrucción de árboles de especies

En el pasado, era muy difícil identificar un gen marcador que diera una idea de la diferenciación para una especie específica. A medida que mejoraba la secuenciación, comenzamos a tener muchos datos de secuenciación en varios genes. A partir de diferentes conjuntos de loci, las personas construyeron diferentes árboles, los cuales dependían en gran medida del conjunto de loci elegidos. Las posibles razones por las que los árboles difieren incluyen ruido (de errores de estimación estadística y ruido), duplicaciones y pérdidas ocultas, y clasificación de alelos en una población.

Problema de reconstrucción de árboles de especies

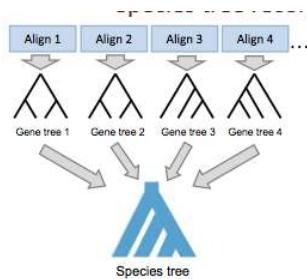


Figura 27.13: Reconstrucción de árboles de especies

Dados muchos árboles genéticos diferentes que no están de acuerdo, nuestro objetivo es convertirlos en un árbol de una especie (como se muestra en la Figura 27.13). Hay muchos algoritmos diferentes que reconstruyen árboles de especies. Estos algoritmos incluyen métodos Supermatrix (Rokas 2003, Ciccarelli 2006), métodos Supertree (Creevey & McInerney 2005), Minimizando la coalescencia profunda (Maddison & Knowles 2006) y Modelado de coalescencia (Liu & Pearl 2007).

Una forma de hacer esto, que en su mayoría es efectiva para datos ruidosos, es reunir más datos para aumentar la precisión. Esto se hace concatenando alineaciones de genes en una supermatriz.

Otro método consiste en construir un árbol para cada uno y usar un método de consenso para resumir estos árboles. Luego identificamos ramas análogas a través de la gran cantidad de árboles y construimos una especie de árbol que tiene las ramas que ocurren con mayor frecuencia.

Hay otra forma de reconstruir un árbol de especies, que es efectiva en caso de que los árboles genéticos no estén de acuerdo por duplicaciones y pérdidas. El objetivo es encontrar la especie arbórea que aplique la menor cantidad de duplicaciones. Construimos todos los árboles genéticos y luego proponemos un árbol de especies. A continuación, utilizamos la reconciliación para determinar el número de eventos que implica cada árbol génico combinado con la especie propuesta. Luego, proponemos árboles de otras especies y movemos ramas alrededor. Los árboles de especies equivocadas tienden a tener muchos eventos que no sucedieron. El árbol correcto debe tener el menor número de eventos.

### Mejorando la reconstrucción de árboles genéticos y el aprendizaje entre árboles genéticos

Podemos usar métodos similares a los descritos anteriormente para construir mejores árboles genéticos. Esto se puede hacer usando información de un árbol de especies para estudiar un árbol genético de interés. Por ejemplo, los árboles de especies pueden ser utilizados para determinar cuándo ocurrieron pérdidas y duplicaciones. La idea es que podamos usar el hecho de que las especies arbóreas a menudo se construyen a partir de todo el genoma, para obtener más información sobre árboles genéticos relacionados. Podemos usar tanto la longitud de la sucursal como el número de eventos para hacer esto.

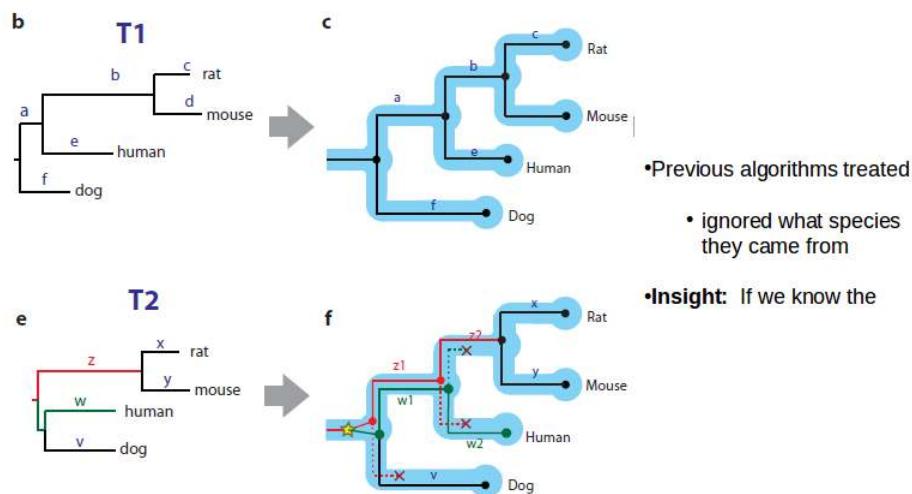


Figura 27.14: Uso de árboles de especies para mejorar la reconstrucción del árbol genético.. fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Si conocemos la especie arbórea, podemos desarrollar un modelo para qué tipo de longitudes de rama podemos esperar. Podemos usar el orden de genes conservados para contar ortólogos y construir árboles.

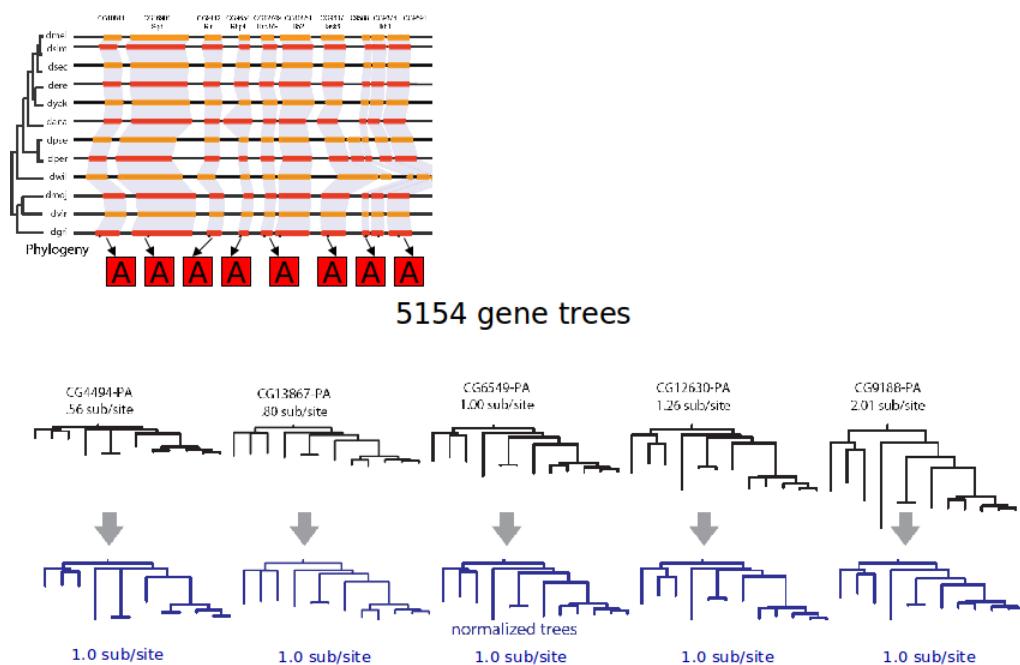


Figura 28.15: Podemos desarrollar un modelo para qué tipo de longitudes de rama podemos esperar. Podemos usar orden de genes conservados para decir ortólogos y construir árboles. fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Cuando un gen está evolucionando rápidamente en una especie, está evolucionando rápidamente en todas las especies. Podemos modelar una longitud de rama como dos componentes de velocidad diferentes. Uno es específico de gen (presente en todas las especies) y el otro es específico de especie, que se personaliza a una especie específica.

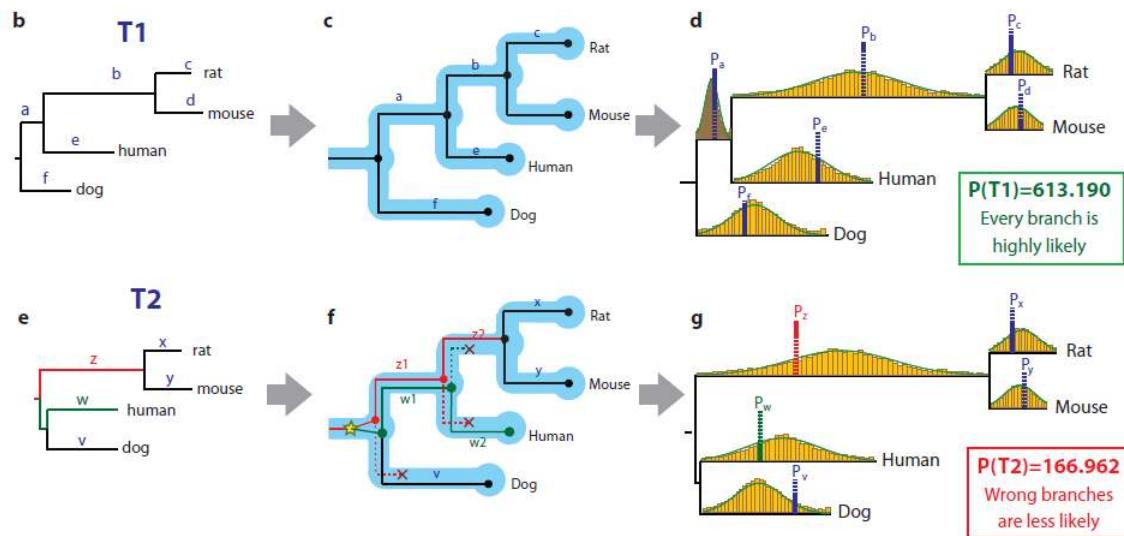


Figura 27.16: La longitud de la rama se puede modelar como dos componentes de tasa diferentes: específico de gen y específico de especie.

Este método mejora enormemente la precisión de la reconstrucción.

27.6: Reconstrucción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 27.6: Reconstruction has no license indicated.

## 27.7: Modelización de Frecuencias de Poblaciones y Alelos

Con el advenimiento de la secuenciación de próxima generación, se está volviendo económico secuenciar los genomas de muchos individuos dentro de una población. Para dar sentido a cómo los alelos se propagan a través de una población, es útil contar con un modelo con el que comparar los datos. El modelo de reproducción Wright-Fisher ha ocupado este papel durante los últimos 70 años.

### El modelo Wright-Fisher

Al igual que los HMM, Wright-Fisher es un proceso de Markov: en cada paso, el sistema progresiona aleatoriamente, y el estado actual del sistema depende únicamente del estado anterior. En este caso, las transiciones de estado representan la reproducción. Al modelar la transmisión de cromosomas a la descendencia, podemos estudiar la deriva genética.

El modelo hace una serie de suposiciones simplificadoras:

1. El tamaño de la población,  $N$ , es constante en cada generación.
2. Sólo los miembros de la misma generación se reproducen (sin solapamiento).
3. La reproducción ocurre al azar.
4. El gen que se está modelando solo tiene 2 alelos.
5. Los genes se someten a selección neutra.

Tenga en cuenta que Wright-Fisher no es una opción apropiada si está tratando de modelar el cambio en la frecuencia de un gen para el que se selecciona positiva o negativamente. Si utilizamos Wright-Fisher para modelar los cromosomas de individuos diploides, el tamaño de la población del modelo se convierte en  $2N$ .

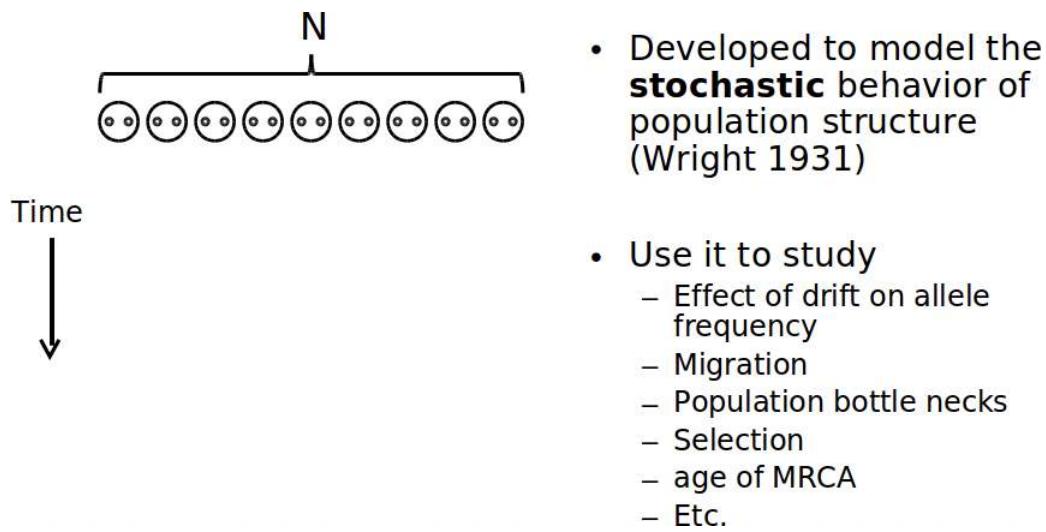


Figura 27.17: El modelo Wright-Fisher

En inglés, así es como funciona Wright-Fisher:

En cada generación, para cada niño, seleccionamos aleatoriamente de los padres (con reemplazo). El alelo del niño se convierte en el del parente seleccionado aleatoriamente.

Repetimos este proceso para muchas generaciones, con los niños sirviendo como los nuevos padres, ignorando el orden de los cromosomas.

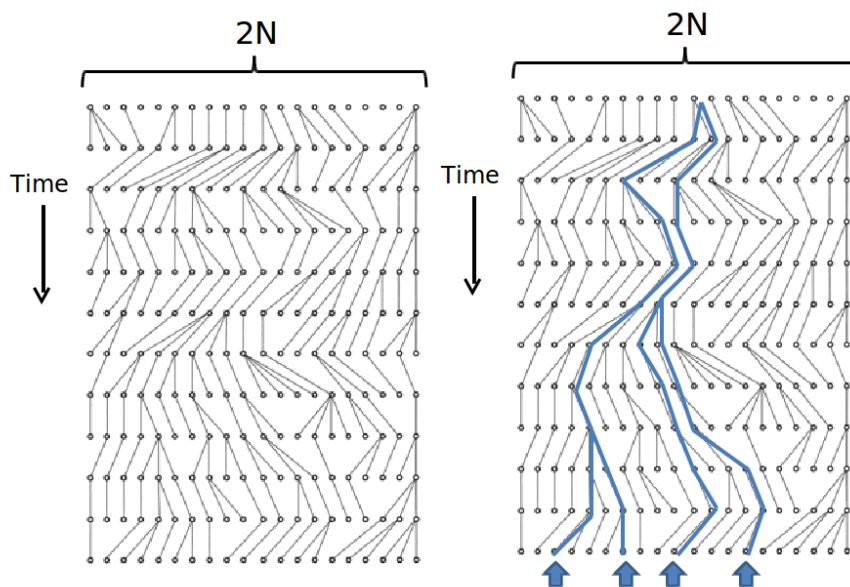
Realmente es así de simple. Para determinar la probabilidad de  $k$  copias de un alelo existente en la generación hijo cuando tenía una frecuencia de  $p$  en la generación padre, podemos usar esta fórmula:

$$\frac{1}{2} \left( \begin{array}{c} p \\ q \end{array} \right)^N$$

$$k$$

$$\frac{1}{2} \left( \begin{array}{c} k \\ n-k \end{array} \right) p^k q^{n-k}$$

Aquí,  $q = (1-p)$ . Es la frecuencia de alelos no p en la generación parental.



- Track lineages
- Genealogies
  - Randomly sample extant chromosomes
  - Trace back tree until all coalescence

Felsenstein 2004.

Sinclair Associates, Inc. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use>.

Figura 27.18: Muchas iteraciones de Wright-Fisher produciendo un árbol de linaje

Ahora podemos comenzar a explorar preguntas como: ¿qué tan probable es y cuántas generaciones se espera que tome para que un alelo dado se fije, es decir, el alelo está presente en cada miembro de la población?

El tiempo esperado (en generaciones) para la fijación, dados los supuestos hechos por Wright-Fisher, es proporcional a  $4N_E$ , donde  $N_E$  es el tamaño efectivo de la población.

Nuevamente, es importante tener en cuenta las limitaciones de este modelo y preguntar si realmente tiene sentido para el sistema que estás tratando de representar. Considere cómo podría modificar el modelo propuesto para dar cuenta de una selección coeiciente que oscila entre -1 (selección letal negativa) y 1 (selección positiva fuerte).

### El modelo coalescente

El problema con el modelo Wright-Fisher es que asume que conoces las frecuencias alélicas de la generación ancestral. Al tratar con los genomas de las especies presentes, estas cantidades son desconocidas. El Modelo Coalescente resuelve este acertijo pensando retrospectivamente. Es decir: comenzamos con los alelos de la generación actual, y trabajamos nuestro camino hacia atrás en el tiempo. El Modelo de Coalescencia básico hace las mismas suposiciones que Wright-Fisher. En cada generación, nos preguntamos: cuál es la probabilidad de que los dos alelos idénticos se unan, o compartan un parente, en la generación anterior.

Podemos plantear la probabilidad de que ocurra un evento de coalescencia en la generación anterior como la probabilidad de que la coalescencia no ocurra en ninguna de las generaciones  $t-1$  anteriores a la última, multiplicada por la probabilidad de que ocurra en la generación anterior (la  $t$ -ésima) generación. Esto equivale a la expresión:

$$P_c(t) = \left(1 - \frac{1}{2N_e}\right)^{t-1} \left(\frac{1}{2N_e}\right) \quad (27.7.1)$$

Donde  $N_e$  es el tamaño efectivo de la población.

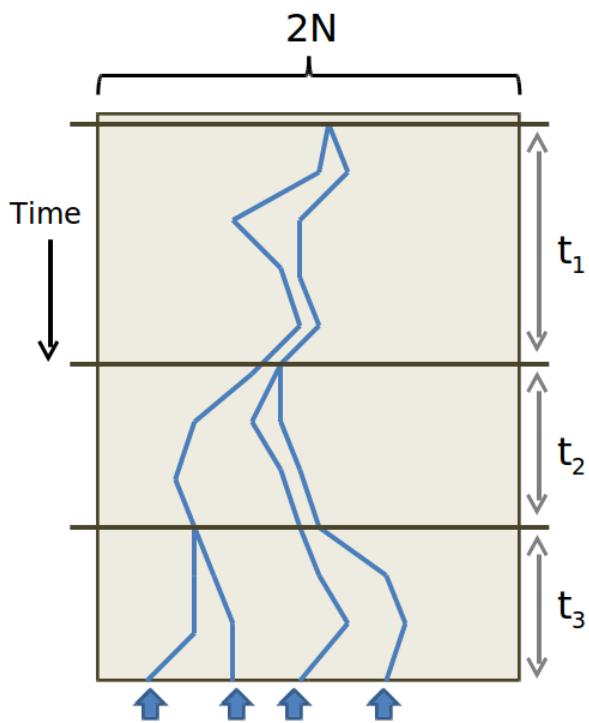


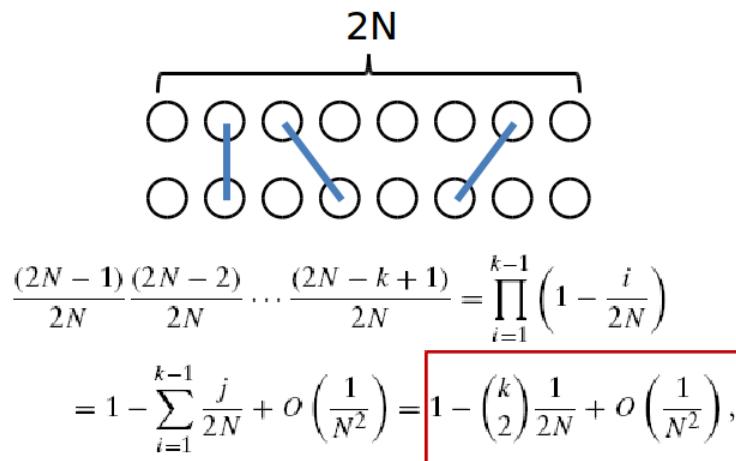
Figura 27.19: El modelo coalescente.

Al aproximar esta distribución geométrica como exponencial:  $P_c(t) = \frac{1}{2N_e} e^{-(\frac{t-1}{2N_e})}$ , podemos determinar el número esperado de generaciones atrás hasta la coalescencia, que resulta ser  $2N e$ , con una desviación estándar de  $2N e$ .

Para preguntar sobre la coalescencia de *múltiples* linajes en una generación dada, debemos, como en Wright-Fisher, usar una distribución binomial. La probabilidad de que  $k$  linajes se coalescen por primera vez en la generación  $t$  es:

$$\begin{aligned} P(\text{izquierda}(T_k) = t \text{ derecha}) &= \text{izquierda}(1 - \text{izquierda}(\begin{array}{l} k \\ 2 \end{array})) \\ &\quad \text{izquierda}(\frac{1}{2N} \text{derecha})^{t-1} \text{izquierda}(\begin{array}{l} k \\ 2 \end{array}) \\ &\quad \text{izquierda}(\frac{1}{2N}) \end{aligned}$$

Y nuevamente, esto se puede aproximar con una distribución exponencial para  $k$  suficientemente grande. El individuo en el que convergen dos linajes se conoce como el **Ancestro Común Más Reciente**. Al moverse continuamente hacia atrás hasta que todos los antepasados se unen, ¡terminamos con un nuevo tipo de árbol! Y al comparar el árbol resultante de la coalescencia con un árbol genético que hemos construido, las discrepancias entre ambos pueden indicar que se han violado ciertos supuestos del Modelo Coalescente. A saber, la selección puede estar ocurriendo.



For  $k \ll N$ ,  $O(1/N^2)$  is very small

Figura 28.20: Distribución geométrica de probabilidad para eventos coalescentes en  $k$  linajes.

### El modelo coalescente multiespecie

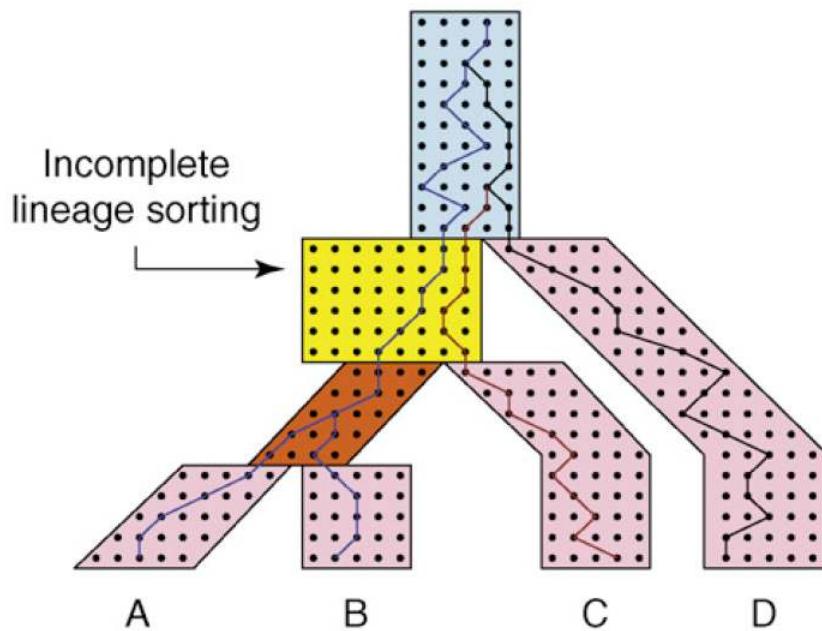


Figura 27.21: Modelo Coalescente Multiespecies.

Cortesía de Elsevier, Incorporar. Usado con permiso.

Fuente: Degnan, James H., y Noah A. Rosenberg. "La discordancia del árbol genético, filogenética

Inferencia y el Coalescente Multiespecie". *Tendencias en Ecología y Evolución* 24, núm. 6 (2009): 332-40.

Podemos llevar esta idea un paso más allá y rastrear eventos de coalescencia en múltiples especies. Aquí, cada genoma de una especie individual es tratado como linaje.

Tenga en cuenta que existe un tiempo de retraso entre la separación de dos poblaciones y el momento en que dos linajes génicos se unen en un ancestro común. También tenga en cuenta cómo la tasa de coalescencia se ralentiza a medida que  $N$  se hace más grande y para ramas cortas.

En la imagen de arriba, la coalescencia profunda se representa en azul claro para tres linajes. Las especies y árboles genéticos aquí son incongruentes ya que C y D son hermanas en el árbol genético pero no en el árbol de la especie.

Existe la  $\frac{2}{3}$  posibilidad de que ocurra incongruencia porque una vez que lleguemos a la sección azul claro, Wright- Fisher no tiene memoria y solo hay  $\frac{1}{3}$  posibilidad de que sea congruente. El efecto de la incongruencia se llama **Clasificación de Linaje Incompleto**. Al medir la frecuencia con la que ocurre el ILS, obtenemos información sobre poblaciones inusualmente grandes o longitudes de ramas insusualmente cortas dentro del árbol de la especie.

Se puede construir un árbol de especies de parsimonia máxima basado en la noción de minimizar el número de eventos de ILS en lugar de minimizar los eventos de duplicación/pérdida implícitos como se cubrió anteriormente. Incluso es posible combinar estos dos métodos para, idealmente, crear una filogenia que sea más precisa de lo que cualquiera de ellos sería individualmente.

---

[27.7: Modelización de Frecuencias de Poblaciones y Alelos](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [27.7: Modeling Population and Allele Frequencies](#) has no license indicated.

## 27.10 ¿Qué hemos aprendido?

En este capítulo, sacamos conclusiones sobre la relación entre árboles genéticos y árboles de especies. Luego exploramos métodos utilizando árboles genéticos para desarrollar árboles de especies más precisas y viceversa, involucrando las tasas de mutación de genes específicos tanto para genes como para especies. El Modelo Wright-Fisher, así como el Modelo Coalescente, nos ayudaron a interpretar mejor estas tasas de mutación y comprender la dinámica de las frecuencias alélicas dentro de una población.

---

27.10 ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [27.10 What Have We Learned?](#) has no license indicated.

## 27.9 Lectura adicional

- Ponencia sobre el descubrimiento del evento de duplicación del genoma completo en levaduras:  
[http://www.nature.com/nature/journal...ture02424.pdf](http://www.nature.com/nature/journal...ature02424.pdf)

27.9 Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [27.9 Further Reading](#) has no license indicated.

## Bibliografía

- [1] O. Akerborg, B. Sennblad, L. Arvestad, y J. Lagergren. Reconstrucción de árboles genéticos bayesianos y análisis de reconciliación. *Proc Natl Acad Sci*, 106 (14) :5714—5719, abr 2009.
- [2] Zmasek C.M. y Eddy S.R. Analizando proteomas por filogenómica automatizada utilizando inferencias remuestreadas de ortólogos. *BMC Bioinformática*, 3 (14), 2002.
- [3] Li H, Coghlan A, Ruan J, Coin LJ, Heriche JK, Osmotherly L, Li R, Liu T, Zhang Z, Bolund L, Wong GK, Zheng W, DeHal P, Wang J y Durbin R. Treefam: una base de datos curada de árboles filogenéticos de familias de genes animales. *Nucleic Acids Res*, 34, 2006.
- [4] Arvestad L., Berglund A., Lagergren J. y Sennblad B. Reconciliación de árboles de genes/especies bayesianas y análisis de ortología usando mcmc. *Bioinformática*, 19 Supl 1, 2003.
- [5] M. D. Rasmussen y M. Kellis. Un enfoque bayesiano para una reconstrucción rápida y precisa de árboles genéticos. *Mol Biol Evol*, 28 (1) :273290, ene 2011.
- [6] Matthew D. Rasmussen y Manolis Kellis. Reconstrucción precisa del árbol genético mediante el aprendizaje de las tasas de sustitución específica de genes y especies en múltiples genomas completos. *Genome Res*, 17 (12) :1932—1942, dic 2007.
- [7] C.E.V. Storm y E.L.L. Sonnhammer. Inferencia de ortólogos automatizados a partir de árboles filogenéticos y cálculo de la confiabilidad de la ortología. *Bioinformática*, 18 (1) :92—99, ene 2002.
- [8] Hollich V., Milchert L., Arvestad L. y Sonnhammer E. Evaluación de medidas de distancia proteica y métodos de construcción de árboles para la reconstrucción filogenética de árboles. *Mol Biol Evol*, 22:2257 —2264, 2005.
- [9] Wapinski, I. A. Pfeffer, N. Friedman y A. Regev. Reconstrucción automática a lo largo del genoma de árboles genéticos filogenéticos. *Bioinformática*, 23 (13) :i549—i558, 2007.

---

[Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 28: Historia de la población

- 28.1: Introducción
- 28.2: Encuesta Rápida de Variación Genética Humana
- 28.3: Flujo genético africano y europeo
- 28.4: Flujo de genes en el subcontinente indio
- 28.5: Flujo de genes entre poblaciones humanas arcaicas
- 28.6: Herramientas y Técnicas
- 28.7: Direcciones de investigación, lecturas adicionales, bibliografía
- 28.8: Ascendencia Europea y Migraciones

---

28: Historia de la población is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 28.1: Introducción

Los humanos comparten 99.9% de la misma información genética, y son 99% similares a los chimpancés. Aunque los humanos tienen menos diversidad genética que muchas otras especies [?], los polimorfismos en poblaciones pueden, sin embargo, conducir a diferencias en el riesgo de enfermedad. Aprender sobre la diferencia de 0.1% entre humanos puede usarse para comprender la historia poblacional, rastrear linajes, predecir enfermedades y analizar tendencias de selección natural.

En esta conferencia, el Dr. David Reich de la Escuela de Medicina de Harvard describe tres ejemplos históricos de flujo de genes entre poblaciones humanas: flujo de genes entre africanos y europeos debido a la trata de esclavos, mezcla india por migración y mestizaje entre neandertales, denisovanos y humanos modernos de Occidente Decente euroasiático.

---

28.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 28.1: Introduction has no license indicated.

## 28.2: Encuesta Rápida de Variación Genética Humana

En el genoma humano, generalmente hay un polimorfismo cada 1000 bases, aunque hay regiones del genoma donde esta tasa puede cuadruplicar. Estos Polimorfismos de Nucleótidos Únicos (SNP) son una manifestación de la variación genética. Cuando ocurren los SNP, se segregan de acuerdo con las tasas de recombinación, ventajas o desventajas de la mutación, y la estructura poblacional que existe y continúa durante la vida útil del SNP. Después de un evento de mezcla genética, por ejemplo, uno ve inicialmente cromosomas enteros, o cerca de cromosomas enteros, provenientes de cada constituyente. A medida que pasan las generaciones, la recombinación divide los bloques de haplotipos SNP en pices más pequeños. La tasa de cambio de la longitud de estos bloques, entonces, depende de la velocidad de recombinación y la estabilidad del producto de recombinación. Por lo tanto, la longitud de los haplotipos conservados puede ser utilizada para inferir la edad de una mutación o su selección. Sin embargo, una consideración importante es que la tasa de recombinación no es uniforme en todo el genoma; más bien, hay puntos calientes de recombinación que pueden sesgar la medida de la edad o selectividad del haplotipo. Esto hace que los bloques de haplotipos sean más largos de lo esperado bajo un modelo uniforme.

Cada lugar del genoma se puede considerar como un árbol cuando se compara entre individuos. Dependiendo de dónde se mire dentro del genoma, un árbol será diferente a otro árbol que pueda obtener de un conjunto específico de SNP. El truco es usar los datos que tenemos disponibles sobre los SNP para inferir los árboles subyacentes, y luego las relaciones filogenéticas generales. Por ejemplo, el cromosoma Y experimenta poca o ninguna recombinación y, por lo tanto, puede producir un árbol de alta precisión a medida que pasa de padre a hijo. De igual manera, podemos observar el ADN mitocondrial transmitido de madre a hijo. Si bien estos árboles pueden tener alta precisión, otros árboles autosómicos se confunden con la recombinación y, por lo tanto, muestran una menor precisión para predecir las relaciones filogenéticas. Los árboles génicos se hacen mejor observando áreas de baja recombinación, ya que la recombinación mezcla árboles. En general, hay alrededor de 1 a 2 recombinaciones por generación.

Los humanos muestran alrededor de 10,000 pares de bases de unión, ya que retrocedemos alrededor de 10,000 generaciones. Los bloques de equilibrio de ligamiento de la mosca de la fruta, por otro lado, son solo unos pocos cientos de bases. La fijación de un alelo ocurrirá con el tiempo, proporcional al tamaño de la población. Para una población de alrededor de 10 mil, tardarán alrededor de 10 mil años en llegar a ese punto. Cuando una población crece, se reduce el efecto de la deriva génica. Curiosamente, la variación en humanos se parece a lo que se habría formado en un tamaño de población de 10,000.

Si se mapean haplotipos largos a árboles genéticos, aproximadamente la mitad de la profundidad está en la primera rama; la mayoría de los cambios de morfología son profundos en el árbol porque hubo más tiempo para mutar. Un modelo simple de mutación sin selección natural es el modelo neutro de Wright-Fisher que utiliza muestreo binomial. En este modelo, un SNP alcanzará la fijación (frecuencia 1) o morirá (frecuencia 0).

En el genoma humano, hay 10-20 millones de SNP comunes. Esto es menos diversidad que los chimpancés, lo que implica que los humanos están genéticamente más cerca unos de otros.

Con esta similitud genética en mente, la comparación de subpoblaciones humanas puede dar información sobre ancestros comunes y sugerir eventos históricos. La similitud entre dos subpoblaciones se puede medir comparando frecuencias alélicas en un diagrama de dispersión. Si trazamos las frecuencias de SNP a través de diferentes poblaciones en una parcela de dispersión, vemos más propagación entre poblaciones más distantes. La siguiente gráfica, por ejemplo, muestra la relativa disimilitud de las poblaciones de los indios europeos americanos y americanos junto con la mayor similitud de las poblaciones europeas americanas y chinas. Las parcelas indican que hubo una divergencia en el pasado entre chinos y nativos americanos, evidencia del cuello de botella de la migración norteamericana que han sido hipotetizados por los arqueólogos. La propagación entre diferentes poblaciones dentro de África es bastante grande. Podemos medir la propagación por el índice de fijación ( $F_{ST}$ ) que describe la varianza.

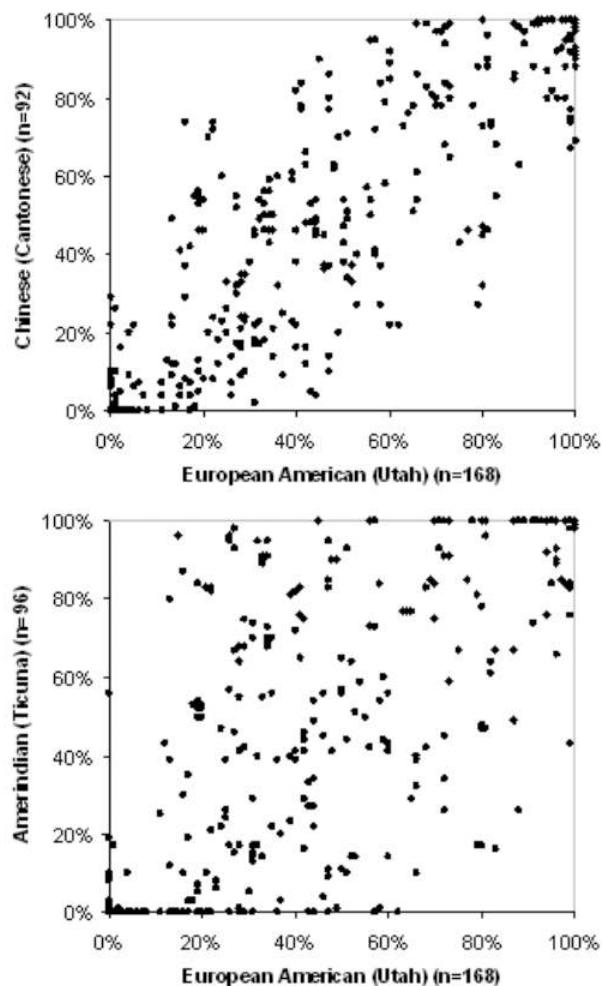


Figura 29.1: La similitud entre dos subpoblaciones se puede medir comparando frecuencias alélicas en una gráfica de dispersión. Las parcelas muestran la diferencia relativa de las poblaciones de los indios europeos americanos e indios americanos junto con una mayor similitud de las poblaciones europeas americanas y chinas.

Varios estudios actuales han demostrado que la agrupación no supervisada de datos genéticos puede recuperar etiquetas autoselecciónadas de identidad étnica. [3] El experimento de Rosenberg utilizó un algoritmo de agrupamiento bayesiano. Tomaron un tamaño de muestra de 1000 personas (50 poblaciones, 20 personas por población), y agruparon a esas personas por sus datos genéticos SNP, pero no etiquetaron a ninguna de las personas con su población, para que pudieran ver cómo se agruparía el algoritmo sin conocimiento de etnia. Intentaron muchos números diferentes de clusters para encontrar el número óptimo. Con 2 cúmulos, se separaron los asiáticos del Este y los no asiáticos del Este. Con 3 racimos, los africanos se separaron de todos los demás. Con 4, los Asiáticos Orientales y los Nativos Americanos fueron separados. Con 5, las subpoblaciones más pequeñas comenzaron a emerger.

Cuando oleadas de humanos abandonaron África, la diversidad genética disminuyó; el pequeño número de personas en los grupos que salieron de África permitió que ocurrieran eventos de fundadores en serie. Estos eventos seriales de fundadores conducen a la formación de subpoblaciones con menor diversidad genética. Este efecto fundador se demuestra por el hecho de que la diversidad genética disminuye al salir de África y que los africanos occidentales tienen la mayor diversidad de cualquier subpoblación humana.

---

28.2: Encuesta Rápida de Variación Genética Humana is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [28.2: Quick Survey of Human Genetic Variation](#) has no license indicated.

## 28.3: Flujo genético africano y europeo

La trata atlántica de esclavos se llevó a cabo del siglo XVI al siglo XIX, y trasladó a alrededor de 5 millones de personas de África a las Américas. La mayoría de los afroamericanos hoy en día tienen una mezcla de 80% de herencia africana y 20% europea. Cuando dos padres de diferentes etnias tienen hijos, sus hijos heredarán un cromosoma de cada parente, y sus nietos heredarán cromosomas que son un mosaico de las dos etnias debido a la recombinación. A medida que pasa el tiempo, el número creciente de eventos de recombinación disminuirá la longitud de los tramos de ADN “africanos” o “europeos”.

Los eventos de recombinación no se distribuyen de manera uniforme a lo largo de los cromosomas, sino que ocurren en los puntos calientes. El ADN africano y europeo tienen diferentes puntos calientes, lo que podría deberse a diferencias en la composición de aminoácidos de PRDM9, una histona H3 (K4) trimetiltransferasa que es esencial para la meiosis.

La diferencia en la susceptibilidad de la enfermedad se puede predecir para las poblaciones africanas y europeas. Con la secuenciación, este conocimiento también se puede aplicar a poblaciones mixtas. Por ejemplo, los africanos tienen un mayor riesgo de cáncer de próstata que está directamente vinculado a un área en el cromosoma 8 que mapea a un proto-oncogén del cáncer [?]. Si un individuo mixto tiene la secuencia africana en esa zona, tendrá el mayor riesgo, pero si el individuo tiene la secuencia europea, no tendrá un mayor riesgo. El mismo enfoque se puede aplicar al cáncer de mama, cáncer de colon, esclerosis múltiple y otras enfermedades.

---

28.3: Flujo genético africano y europeo is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [28.3: African and European Gene Flow](#) has no license indicated.

## 28.4: Flujo de genes en el subcontinente indio

La evidencia genética sugiere que las poblaciones modernas del subcontinente indio descendieron de dos poblaciones ancestrales diferentes que se mezclaron hace 4,000 años. Los datos de la matriz SNP se recolectaron de alrededor de 500 personas diferentes de 73 grupos indios con diferentes familias lingüísticas [? ]. Una gráfica de análisis de componentes principales revela que los grupos lingüísticos dravidianos/indoeuropeos y los grupos lingüísticos austroasiáticos se encuentran en dos grupos diferentes, lo que sugiere que tienen diferentes linajes. Dentro de los grupos lingüísticos dravidianos/indoeuropeos, existe un gradiente de relación con los grupos de Eurasia Occidental.

Se utilizó la misma técnica de mosaico utilizada en el estudio de mezcla africano/europeo para estimar la fecha de la mezcla. La población india es una mezcla de un grupo centroasiático-europeo y otro grupo más estrechamente relacionado con la gente de las Islas Andamán. El tamaño del trozo del ADN que pertenece a cada grupo sugiere una mezcla de unas 100 generaciones de antigüedad, o hace 2,000 a 4,000 años. Muchos grupos tienen este patrimonio mixto, pero la mezcla se detiene después de la creación del sistema de castas.

El conocimiento del patrimonio de los genes puede predecir enfermedades. Por ejemplo, una mutación del sur de Asia en la proteína C de unión a miosina provoca un aumento de siete veces en la insuficiencia cardíaca. Muchos grupos étnicos son endogámicos y tienen una diversidad genética baja, lo que resulta en una mayor prevalencia de enfermedades recesivas.

Encuestas anteriores en la India han estudiado aspectos como la variación antropométrica, el ADNmt y el cromosoma Y. El estudio antropométrico analizó diferencias significativas en las características físicas entre grupos separados por geografía y etnia. Los resultados mostraron una variación muy superior a la de Europa. El estudio de ADNmt fue una encuesta de linaje materno y los resultados sugirieron que había un solo árbol indio de tal manera que la edad del linaje podría inferirse por el número de mutaciones. Los datos también mostraron que las poblaciones indias estaban separadas de las poblaciones no indias al menos hace 40 mil años. Finalmente, el estudio del cromosoma Y analizó el linaje paterno y mostró una similitud más reciente con los hombres de Oriente Medio y dependencias de geografía y casta. Estos datos entran en conflicto con los resultados de ADNmt. Una posible ex- planación es que hubo una migración masculina más reciente. De cualquier manera, los estudios genéticos realizados en la India han servido para mostrar su complejidad genética. La alta variación genética, la disimilitud con otras muestras y la dificultad de obtener más muestras llevan a que la India quede fuera de HapMap, el Proyecto 1000 Genomas y el HGDP.

En el estudio de David Reich y colaboradores de la India, se eligieron 25 grupos indios para representar diversas geografías, raíces lingüísticas y etnias. Los datos brutos incluyeron cinco muestras para cada uno de los veinticinco grupos. A pesar de que este número parece pequeño, el número de SNP de cada muestra tiene mucha información. Aproximadamente quinientos mil marcadores fueron genotipados por individuo. Al observar los datos que emergen del estudio, si se usa Análisis de Componentes Principales en datos de Eurasianos Occidentales y Asiáticos, y si se comparan las poblaciones indias usando los mismos componentes, emerge el Cline de la India. Esto muestra un gradiente de similitud que podría indicar una divergencia escalonada de poblaciones indias y europeas.

### Casi todos los grupos de indios continentales son mixtos

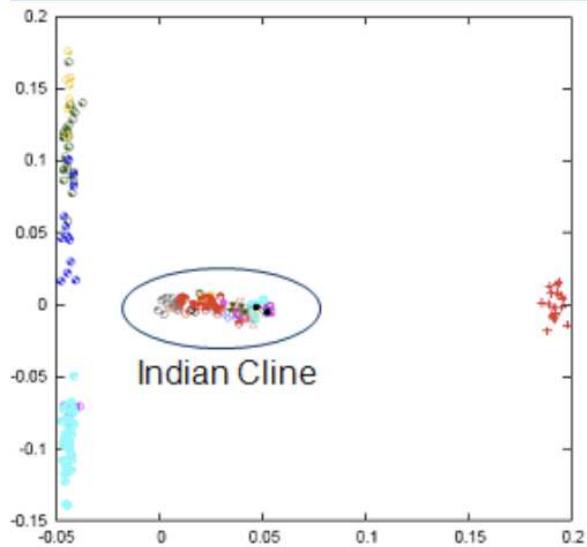
Un análisis posterior del fenómeno India Cline produce resultados interesantes. Por ejemplo, algunas subpoblaciones paquistaníes tienen ascendencia que también cae a lo largo del clino indio. Las poblaciones pueden proyectarse sobre los componentes principales de otras poblaciones: los asiáticos del sur proyectados sobre los componentes principales chinos y europeos producen un efecto lineal (el clino de la India), mientras que los europeos proyectados sobre los componentes principales del sur de Asia y China no. Una interpretación es que la ascendencia india muestra más variabilidad que los otros grupos. Una evaluación de variabilidad similar aparece al comparar poblaciones africanas con no africanas. De este análisis surgen dos hipótesis de árbol:

1. hubo eventos de fundadores en serie en la historia de la India o
2. hubo flujo de genes entre poblaciones ancestrales.

Los autores desarrollaron una prueba formal de cuatro poblaciones para probar hipótesis de ascendencia en presencia de mezcla u otros efectos de confusión. La prueba toma una topología de árbol propuesta y suma sobre todos los SNP de (P<sub>1</sub> P<sub>2</sub>) (P<sub>3</sub> P<sub>4</sub>), donde los valores de P son frecuencias para las cuatro poblaciones. Si el árbol propuesto es correcto, la correlación será 0 y las poblaciones en cuestión forman un clado. Este método es resistente a varios problemas que limitan otros modelos. Se puede

construir un modelo completo para adaptarse a la historia. La información topológica de las gráficas de mezcla se puede aumentar con valores de Fst a través de un procedimiento de ajuste. Este método no hace suposiciones sobre los tiempos de división de la población, la expansión y las contracciones, y la duración del flujo génico, lo que resulta en un procedimiento de estimación más robusto.

### Projecting South Asians onto PCA of Europeans and Chinese



### Projecting Europeans onto PCA of South Asians and Chinese

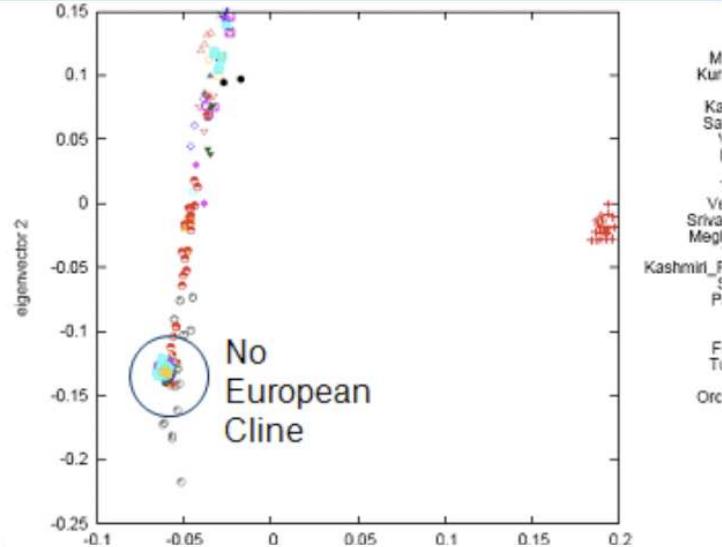


Figura 28.2: Las poblaciones pueden proyectarse sobre los componentes principales de otras poblaciones: los surasiáticos proyectados sobre los componentes principales chinos y europeos producen un efecto lineal (el clino de la India), mientras que los europeos proyectados sobre los componentes principales del sur de Asia y China no.

Además, al estimar las proporciones de la mezcla utilizando el estadístico de 4 poblaciones, se obtienen estimaciones de error para cada uno de los grupos del árbol. La historia complicada no tiene en cuenta en este cálculo, siempre y cuando la topología determinada por la prueba de 4 poblaciones sea válida.

Estas pruebas y el análisis clino permitieron a los autores determinar la fuerza relativa de la ascendencia ancestral del norte de la India y Ancestral del Sur de la India en cada muestra poblacional representativa. Encontraron

## An admixture graph that fits Indian history

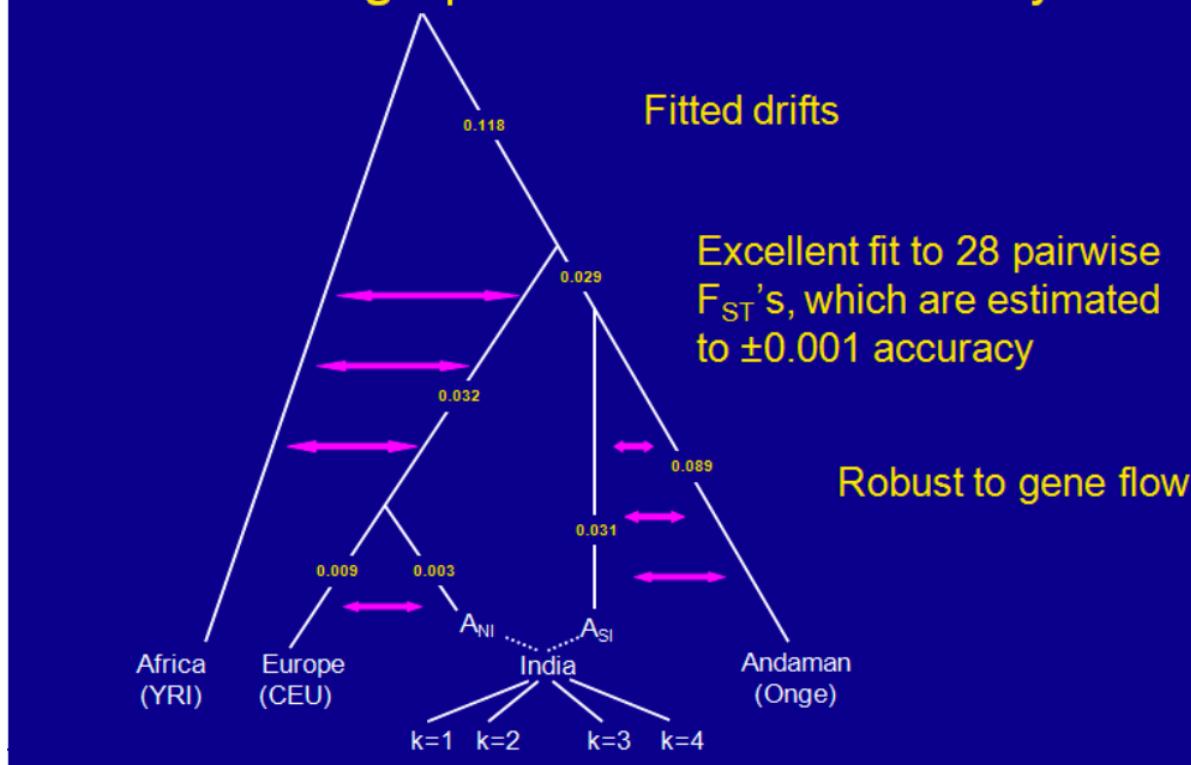


Figura 28.3: Un gráfico de mezcla que se ajusta a la historia de la India  
fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

que la alta ascendencia ancestral del norte de la India se correlaciona con la casta tradicionalmente superior y ciertas agrupaciones lingüísticas. Además, la ascendencia ancestral de la India Norte (ANI) y del sur de la India (ASI) es tan diferente de la china como la europea.

### La estructura poblacional en la India es diferente a la de Europa

La estructura poblacional en la India está mucho menos correlacionada con la geografía que en Europa. Incluso corrigiendo las poblaciones por diferencias lingüísticas, geográficas y de estatus social, el primer valor es 0.007, aproximadamente 7 veces el de las poblaciones más divergentes de Europa. Una pregunta abierta es si esto podría deberse a la falta de SNP (en gran parte específicos de la India) en las matrices de genotipado. Esto se debe a que el conjunto de SNP dirigidos se identificaron principalmente a partir del proyecto HapMap, que no incluyó fuentes indias.

La mayor parte de la variación genética india no surge de eventos fuera de la India. Adicionalmente, los matrimonios consanguíneos no pueden explicar la señal. Muchos eventos de fundadores en serie, quizás vinculados a las castas o grupos precursores, podrían contribuir. Analizando un solo grupo a la vez, se hace evidente que las castas y subcastas tienen mucha endogamia. La autocorrelación del reparto de alelos entre pares de muestras dentro de un grupo se utiliza para determinar si ocurrió un evento fundador y su edad relativa. Hay segmentos de ADN de un fundador, muchos indicando eventos de más de 1000 años de antigüedad. En la mayoría de los grupos hay evidencia de un fuerte evento fundador antiguo y posterior endogamia. Esto contrasta con la estructura poblacional en la mayor parte de Europa o África, donde se produce más mezcla poblacional (menos endogamia).

Estos eventos de fundadores en serie y su estructura resultante tienen importantes implicaciones médicas. Los fuertes eventos fundadores seguidos de la endogamia y algunas mezclas han llevado a grupos que tienen fuertes propensiones a diversas enfermedades recesivas. Esta estructura significa que los grupos indios tienen una colección de enfermedades prevalentes, similares a las ya conocidas en otros grupos, como los judíos asquenazíes o los finlandeses. La variación única dentro de la India significa

que los vínculos con los alelos de enfermedades prevalentes en la India podrían no ser detectables usando solo fuentes de datos no indias. Se necesita un pequeño número de muestras de cada grupo, y más grupos, para mapear mejor estas enfermedades recesivas. Estos mapas se pueden utilizar para predecir mejor los patrones de enfermedades en la India.

#### 28.4.3 Discusión

En general, los fuertes eventos de los fundadores seguidos de la endogamia han dado a la India más subestructura que a Europa. Todos los grupos tribales y castas encuestados muestran una fuerte mezcla de ascendencia ANI y ASI, variando entre 35% y 75% de identidad ANI. Estimar el tiempo y el mecanismo de la mezcla ANI-ASI es actualmente una prioridad alta. Adicionalmente, futuros estudios determinarán si se pueden aplicar nuevas técnicas como la prueba de 4 poblaciones y los gráficos de mezcla a otras poblaciones y cómo.

---

[28.4: Flujo de genes en el subcontinente indio](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [28.4: Gene Flow on the Indian Subcontinent](#) has no license indicated.

## 28.5: Flujo de genes entre poblaciones humanas arcaicas

El Dr. Reich trabajó con el Instituto Max Planck como genetista poblacional estudiando datos genéticos neandertales. En esta sección se discutirán los antecedentes de su investigación como parte del proyecto del genoma neandertal, el borrador de secuencia que ensamblaron y la evidencia que se ha compilado para el flujo de genes entre los humanos modernos y los neandertales.

### Antecedentes

Los neandertales son el único otro homínido con un cerebro tan grande como el *Homo sapiens*. Fósiles neandertales de hace 200.000 años se han encontrado en Eurasia Occidental (Europa y Asia Occidental), que es muy anterior al *Homo erectus*. Los primeros fósiles humanos provienen de Etiopía que datan de hace unos 200.000 años. Sin embargo, existe evidencia de que los neandertales y los humanos se superpusieron en el tiempo y el espacio entre 135 mil y 35 mil años atrás.

El primer lugar de contacto podría haber ocurrido en El Levante, en Israel. Hay fósiles humanos de hace 120 mil años, luego una brecha, fósiles neandertales hace unos 80 mil años, otra brecha, y luego fósiles humanos nuevamente hace 60 mil años. Esto es prueba de una superposición en su lugar, pero no en el tiempo. En la época paleolítica superior, hubo una explosión de poblaciones que salían de África (la migración hace unos 60 mil a 45 mil años). En Europa después de hace 45 mil años, hay sitios donde los neandertales y los humanos existen lado a lado en el registro fósil. Dado que hay evidencia de que las dos especies coexistieron, ¿hubo mestizaje? Esta es una pregunta que se puede responder examinando la genómica poblacional.

Consulte Herramientas y técnicas para una discusión sobre la extracción de ADN de neandertales.

#### 28.5.2 Evidencia de flujo génico entre humanos y neandertales

1. Una prueba de comparación entre el ADN neandertal y el ADN humano de poblaciones africanas y no africanas demuestra que las poblaciones no africanas están más relacionadas con los neandertales que con las poblaciones africanas. Podemos observar todos los SNP en el genoma y ver si el SNP humano de una población coincide con el SNP neandertal. Cuando se compararon diferentes poblaciones humanas con los neandertales, se encontró que los SNP franceses, chinos y de Nueva Guinea coincidían con los SNP neandertales mucho más que los SNP yoruba nigerianos coincidían con los SNP neandertales. Las poblaciones san bosquimanos y yoruba de África, a pesar de ser genéticamente muy distintas, ambas tenían la misma distancia del ADN neandertal. Esta evidencia sugiere que las poblaciones humanas que migran de África se cruzan con los neandertales.
2. Un estudio de haplotipos de largo alcance demuestra que cuando la rama más profunda de un árbol haplotípico estaba en poblaciones no africanas, las regiones coincidieron con frecuencia con el ADN de Neandertal. Las poblaciones africanas hoy en día son las poblaciones más diversas del mundo. Cuando los humanos migraron fuera de África, la diversidad disminuyó debido al efecto fundador. De esta historia, uno esperaría que si construyeras un árbol de relaciones, la división más profunda sería africana. Para mostrar la herencia neandertal, los investigadores de Berkley seleccionaron secciones de largo alcance del genoma y las compararon entre humanos elegidos al azar de varias poblaciones. La rama más profunda del árbol construida a partir de ese haplotípico es casi siempre de la población africana. Sin embargo, ocasionalmente los no africanos tienen la rama más profunda. El estudio encontró que había 12 regiones donde los no africanos tienen la rama más profunda. Cuando se utilizaron estos datos para analizar el genoma neandertal, se encontró que 10 de 12 de estas regiones en los no africanos coincidieron más con los neandertales que la secuencia de referencia humana coincidente (una compilación de secuencias de diversas poblaciones). Esto es evidencia de que ese haplotípico en realidad es de origen neandertal.
3. Por último, existe una divergencia mayor de lo esperado entre los humanos. La división promedio entre un Neandertal y un humano es de unos 800,000 años. La divergencia típica entre dos humanos es de aproximadamente 500,000 años. Al observar secuencias africanas y no africanas, las regiones de baja divergencia surgieron en secuencias no africanas en comparación con el material neandertal. Las regiones encontradas estaban altamente Enriquecidas en material neandertal (94% neandertal), lo que aumentaría la divergencia promedio entre humanos (ya que la divergencia estándar Neandertal - humana es de aproximadamente 800,000 años).

### Flujo de genes entre humanos y denisovanos

En 2010, los científicos descubrieron un hueso de dedo de 50 mil años de edad en el sur de Siberia. El ADN en esta muestra denisovana no era como ningún ADN humano anterior. El ADN mitocondrial denisovano es un grupo externo tanto para los

neandertales como para los humanos modernos. (Se utilizó ADN mitocondrial porque es aproximadamente 1000 veces más frecuente que el ADN somático. La tasa de polimorfismo también es 10 veces mayor). Los denisovanos están más estrechamente relacionados con los neandertales que con los humanos.

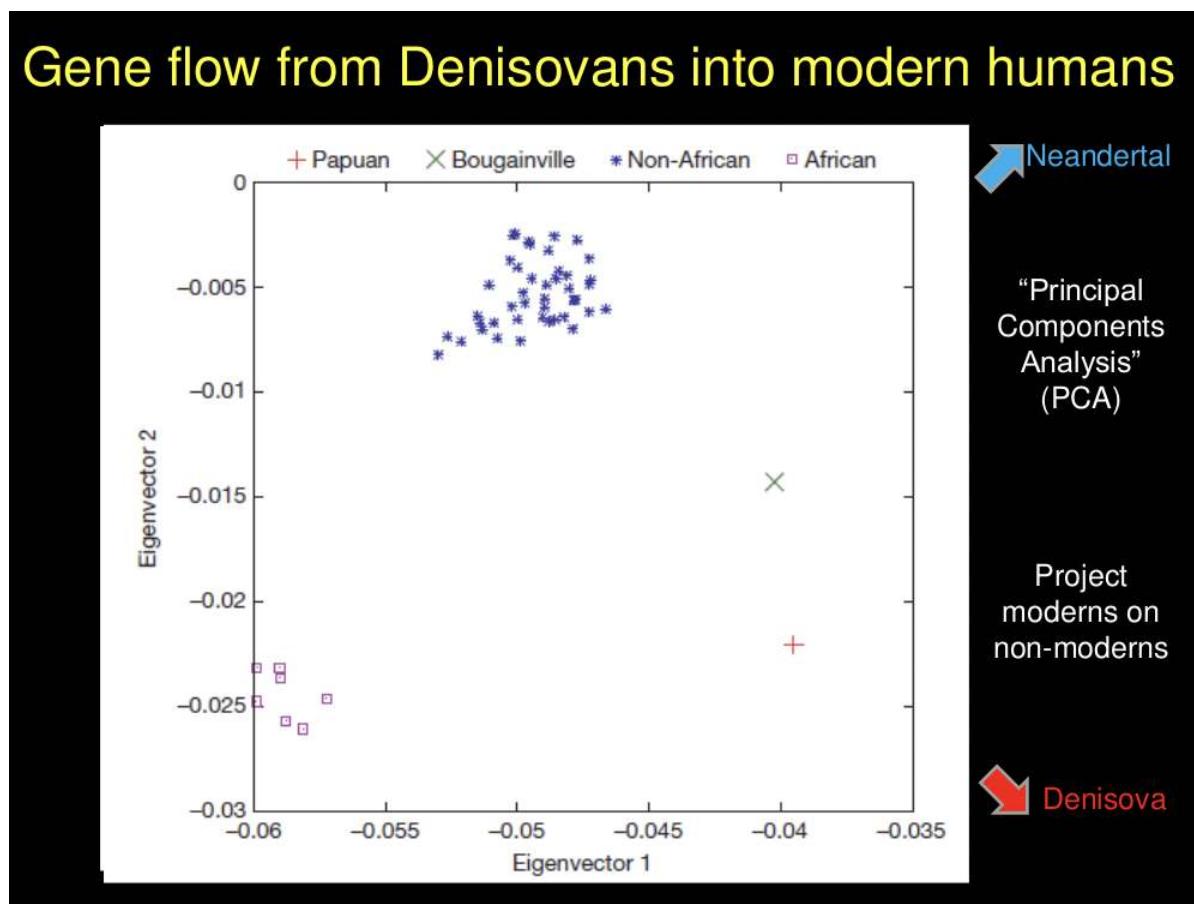


Figura 28.4: Proyección sobre dos dimensiones de un análisis de componentes principales de diferentes poblaciones humanas.

Usando la misma técnica de emparejamiento de SNP del ejemplo neandertal, se descubrió que el ADN de Deniso- van coincide más con el ADN de Nueva Guinea que el ADN chino o el ADN europeo. Se estima que los denisovanos aportaron alrededor del 5% de la ascendencia de los neoguineanos en la actualidad. Una proyección de análisis de componentes principle (ver figura) entre la relación con chimpancés, neandertales y denisovanos muestra que las poblaciones no africanas están más relacionadas con los neandertales, y los nuevos guineanos y buganvillas están más relacionadas con los denisovanos.

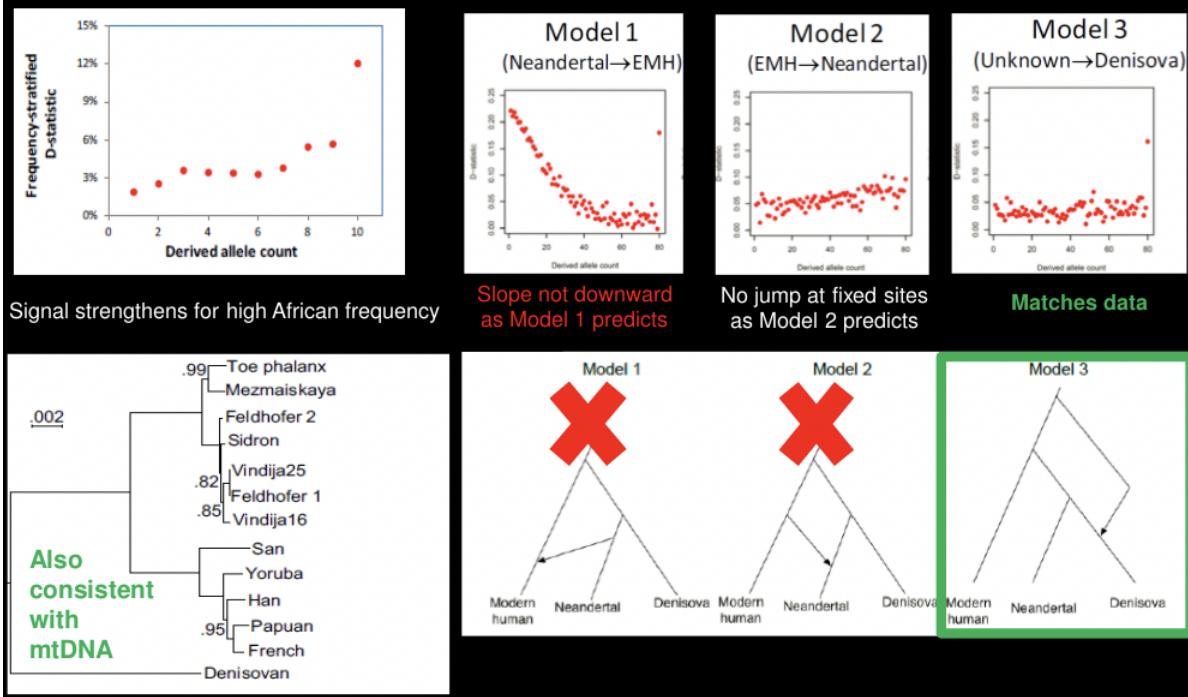
Esta evidencia sugiere un modelo para la migración humana y el mestizaje. Los humanos migraron fuera de África y se cruzaron con los neandertales, luego se extendieron por Asia y se cruzaron con denisovanos en el sudeste asiático. Es menos plausible que los humanos se crucen con denisovanos en la India porque no todas las poblaciones del sudeste asiático tienen ascendencia denisovan.

### Análisis de Genomas Arcaicos de Alta Cobertura

Los genomas arcaicos de alta cobertura pueden decirnos mucho sobre la historia de las poblaciones de homínidos. Se adquirió una secuencia de Neandertal de Altai de alta cobertura de un hueso del dedo del pie encontrado en la cueva de Denisova. A partir de esta secuencia, podemos observar el tiempo de convergencia de las dos copias de cromosomas para estimar el tamaño de la población. El ADN neandertal contiene muchos tramos largos de homocigosidad, lo que indica un pequeño tamaño poblacional persistente y endogamia. Para el Neandertal de Altai, una octava parte del genoma era homocigótico, sobre el nivel esperado de endogamia de los medios hermanos. La aplicación de la técnica a poblaciones no africanas muestra un cuello de botella hace 50 mil años y una posterior expansión poblacional, lo que es consistente con la teoría de Out Of Africa.

Los neandertales y denisovanos también se cruzaron, demostrando la notable proclividad de los humanoides a la reproducción. Aunque la mayor parte del genoma neandertal tiene una profundidad mínima de cientos de miles de años desde el genoma denisovano, al menos 0.5% del genoma denisovan tiene una distancia mucho más corta del genoma neandertal, especialmente para los genes inmunes.

## Denisovans have ancestry from an unknown archaic population unrelated to Neandertals



Nick Patterson, Fernando Racimo, Sriram Sankararaman, Montgomery Slatkin

Figura 28.5: Datos y modelos para el flujo de genes ancestrales.

Los denisovanos probablemente tienen ascendencia de una población arcaica desconocida no relacionada con los neandertales. Una secuencia africana tiene una coincidencia de 23% con ADN neandertal y 47% con ADN denisovo, lo que es estadísticamente significativo. Si estratificas la estadística D por la frecuencia de un alelo en la población, ves una pendiente creciente y un salto brusco al llegar a la fijación que más se ajusta a las predicciones que se obtendrían de una población desconocida que fluye hacia denisovanos (ver figura).

### Discusión

El cuello de botella causado por la migración de África es sólo un ejemplo de muchos. La mayoría de los científicos suelen concentrarse en la edad e intensidad de los eventos migratorios y no necesariamente en la duración, pero la duración es muy importante porque los largos cuellos de botella crean un rango menor de diversidad. Una forma de predecir la longitud de un cuello de botella es determinar si surgieron nuevas variaciones durante el mismo, lo que es más probable durante cuellos de botella más largos. El cambio en el rango de diversidad es también lo que ayudó a crear las diferentes subpoblaciones humanas que quedaron geográficamente aisladas. Esta es solo otra forma en que la genómica poblacional puede ser útil para ayudar a reconstruir las migraciones históricas.

Las diferencias genéticas entre especies (aquí dentro de los primates) se pueden utilizar para ayudar a comprender el árbol filogenético del que todos derivamos. Analizamos el estudio de caso de comparaciones con el ADN neandertal, aprendimos sobre cómo se obtienen muestras de ADN antiguo, cómo se encuentran e interpretan las secuencias, y cómo esa evidencia muestra una alta probabilidad de mestizaje entre humanos modernos (de ascendencia euroasiática) y neandertales. Esas diferencias muy

pequeñas entre una especie y otra, y dentro de las especies, nos permiten deducir gran parte de la historia humana a través de la genética poblacional.

---

28.5: Flujos de genes entre poblaciones humanas arcaicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 28.5: Gene Flow Between Archaic Human Populations has no license indicated.

## 28.6: Herramientas y Técnicas

### Técnicas para estudiar las relaciones poblacionales

Existen varios métodos diferentes para estudiar las relaciones poblacionales con datos genéticos. El primer tipo de estudio general utiliza datos tanto de filogenia como de migración. Ajusta las filogenias a valores de Fst, valores de heterocigosidad subpoblacional (pioneros por Cavalli-Sforza y Edwards en 1967 [?]). Este método también hace uso de mapas sintéticos y Análisis de Componentes Principales. [2] La principal desventaja de analizar los datos de población de esta manera es la incertidumbre sobre los resultados. Hay efectos matemáticos y de borde en el procesamiento de datos que no se pueden predecir. Además, ciertos grupos han demostrado que poblaciones mezcladas separadas y delimitadas pueden producir componentes principales de apariencia significativa por casualidad. Aunque los resultados del estudio sean correctos, entonces, también son inciertos.

El segundo método para analizar las relaciones entre subpoblaciones es el agrupamiento genético. Los clústeres se pueden formar usando ascendencia autodefinida [1] o la base de datos ESTRUCTURA. [3] Este método se utiliza en exceso y puede sobreajustar los datos; la composición de la base de datos puede sesgar los resultados de agrupamiento.

Los avances tecnológicos y el aumento de la recolección de datos, sin embargo, han producido conjuntos de datos que son 10,000 veces más grandes que antes, lo que significa que las afirmaciones más específicas pueden ser desestimadas por algún subconjunto de datos. Entonces, en efecto, muchos modelos que se predicen ya sea por filogenia y migración o agrupamiento genético serán desacreditados en algún momento, conduciendo a una confusión de resultados a gran escala. Una solución a este problema es utilizar un modelo sencillo que haga una afirmación que sea a la vez útil y que tenga menos probabilidad de ser falsificada.

### Extracción de ADN de Huesos Neandertales

Echemos un vistazo a cómo vas a encontrar y secuenciar ADN de restos antiguos. Primero, hay que obtener una muestra de hueso con ADN de un Neandertal. El ADN humano y el ADN neandertal es muy similar (somos más parecidos a ellos que a los chimpancés), así que al secuenciar lecturas cortas con ADN muy antiguo, es imposible saber si el ADN es neandertal o humano. La cueva donde se encontraron los huesos se clasifica primero como humana o no humana utilizando basura o herramientas como identificador, lo que ayuda a predecir el origen de los huesos. Incluso si tienes un hueso, sigue siendo muy poco probable que tengas algún ADN recuperable. De hecho, el 99% de la secuencia de neandertales proviene de solo tres huesos largos encontrados en un sitio: la cueva Vindija en Croacia (5.3 Gb, 1.3x cobertura completa).

A continuación, el ADN se envía a un laboratorio de ADN antiguo. Ya que son huesos de 40 mil años, les queda muy poco ADN. Entonces, primero se les hace un cribado de ADN. Si encuentran ADN, la siguiente pregunta es si es ADN de primate. Por lo general, es ADN de microbios y hongos que viven en el suelo y digieren organismos muertos. Solo alrededor del 1-10% del ADN en huesos viejos es el ADN de los primates. Si es ADN de primates, ¿es contaminación del ser humano (arqueólogo o técnico de laboratorio) que lo maneja? Solo uno de los 600 pb es diferente entre humanos y ADN de neandertales. El tamaño de las lecturas de una muestra ósea de 40 mil años de edad es de 30-40 pb. Las lecturas son casi siempre idénticas para un humano y un neandertal, por lo que es difícil distinguirlas.

En una instancia, 89 extractos de ADN fueron cribados para ADN de neandertales, pero solo 6 huesos fueron realmente secuenciados (requiere falta de contaminación y cantidad suficientemente alta de ADN). El proceso de recuperación del ADN requiere perforar debajo de la superficie ósea (para minimizar la contaminación) y tomar muestras desde adentro. Para los tres huesos largos, se pudo obtener menos de 1 gramo de polvo de hueso. Luego se secuencia el ADN y se alinea con un genoma de chimpancé de referencia. Se mapea a un chimpancé en lugar de a un humano en particular porque mapear a un humano podría causar sesgo si se busca ver cómo se relaciona la secuencia con subpoblaciones humanas específicas.

Los hallazgos más exitosos han sido en cuevas frías de piedra caliza, donde es seca y fría y quizás un poco básica. La mejor probabilidad de conservación ocurre en áreas de permafrost. Muy poco ADN es recuperable de los trópicos. Los trópicos tienen un gran registro fósil, pero el ADN es mucho más difícil de obtener. Dado que la mayoría de los huesos no producen suficiente o buen ADN, los científicos tienen las muestras de cribado una y otra vez hasta que finalmente encuentran una buena.

## Reensamblar ADN antiguo

El ADN extraído de los huesos neandertales tiene lecturas cortas, alrededor de 37 pb en promedio. Hay muchos agujeros debido a mutaciones causadas por el tiempo erosionando el ADN. Es difícil saber si una secuencia es el resultado de la contaminación porque los humanos y los neandertales sólo difieren en una de cada mil bases. Sin embargo, podemos usar el daño del ADN característico del ADN antiguo para distinguir el ADN antiguo y el nuevo. El ADN antiguo tiene una tendencia hacia los errores C a T y G a A. El error C a T es con mucho el más común, y se ve alrededor del 2% de las veces. Con el tiempo, un grupo metilo se desprende de una C, lo que hace que se asemeje a U. Cuando se usa PCR para amplificar el ADN para secuenciación, la polimerasa ve una U y la repara a una T. Para combatir este error, los científicos utilizan una enzima especial que reconoce la U, y corta la hebra en lugar de reemplazarla con una T. Esto ayuda a identificar esos sitios. Las mutaciones G a A son el resultado de verlo en la hebra opuesta.

El tamaño promedio del fragmento es bastante pequeño, y la tasa de error sigue siendo 0.1% - 0.3%. Una forma de combatir las mutaciones es notar que en un fragmento bicatenario, el ADN se deshilacha hacia los extremos, donde se vuelve monocatenario durante aproximadamente 10 pb. Suelen haber altas tasas de mutaciones en las primeras y últimas 10 bases, pero ADN de alta calidad en otros lugares, es decir, más mutaciones C a T al principio y G a A al final. En los chimpancés, las mutaciones más comunes son las transiciones (purina a purina, pirimidina a pirimidina), y las transversiones son mucho más raras. Lo mismo ocurre con los humanos. Dado que las mutaciones G a A y C a T son transiciones, se puede determinar que hay aproximadamente 4 veces más mutaciones en el ADN neandertal antiguo que si fuera fresco al anotar el número de transiciones vistas en comparación con el número de transversiones observadas (comparando el ADN de Neandertal con el ADN humano). Las transversiones tienen una tasa de ocurrencia bastante estable, por lo que esa relación ayuda a determinar cuánto error ha ocurrido a través de mutaciones C a T.

Ahora somos capaces de reducir la contaminación humana del ADN del artefacto a 1%. Cuando se introduce el ADN, tan pronto como se extrae del hueso se codifica con barras con una etiqueta de 7 pb. Esa etiqueta permite evitar la contaminación en cualquier momento posterior del experimento, pero no antes. La extracción también se realiza en una sala limpia con luz UV, después de haber lavado el hueso. El ADN mitocondrial es útil para distinguir qué porcentaje de la muestra está contaminada con ADN humano. El ADN mitocondrial está lleno de sitios de eventos característicos porque los humanos y los neandertales son recíprocamente monofilogenéticos. La contaminación se puede medir contando la proporción de esos sitios. En el ADN neandertal, la contaminación estaba presente, pero sí a 0.5%.

En la secuenciación, la tasa de error es casi siempre mayor que la tasa de polimorfismo. Por lo tanto, la mayoría de los sitios en la secuencia que son diferentes a los humanos son causados por errores de secuenciación. Entonces no podemos aprender exactamente sobre la biología neandertal a través de la secuencia generada, pero podemos analizar SNP particulares siempre y cuando sepamos dónde buscar. La probabilidad de que un SNP en particular sea cambiado debido a un error en la secuenciación es solo  $\frac{1}{300}$  de 11000, por lo que aún se pueden obtener datos utilizables.

Después de alinear las secuencias de chimpancés, neandertales y humanos modernos, podemos medir la distancia entre los neandertales y los humanos y los chimpancés. Esta distancia es sólo de aproximadamente 12.7% de la secuencia de referencia humana. Una muestra francesa mide aproximadamente 8% de distancia de la secuencia de referencia, y una bosquimana aproximadamente 10.3%. Lo que esto dice es que el ADN neandertal está dentro de nuestro rango de variación como especie.

---

28.6: Herramientas y Técnicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 28.6: Tools and Techniques has no license indicated.

## 28.7: Direcciones de investigación, lecturas adicionales, bibliografía

### Direcciones de investigación

Actualmente, la tendencia más emocionante en el campo es la existencia de cada vez más datos sobre genética poblacional tanto antigua como moderna. Con más muestras, podemos idear pruebas estadísticas más finas y generar más y más información sobre la composición de la población y la historia.

### Lectura adicional

#### Bibliografía

[1] Bowcock AM, Ruiz-Linares A, Tomfohrde J, Minch E, Kidd JR y Cavalli-Sforza LL. Alta resolución de árboles de historia evolutiva humana con microsatélites polimórficos. *Naturaleza*, 368:455 —457, 1994.

[2] Menozzi. Mapas sintéticos de frecuencias génicas humanas en europeos. *Science*, 201 (4358) :768—792, sep 1978. [3] Rosenberg N. Estructura genética de las poblaciones humanas. *Ciencia*, 298 (5602) :2381—2385, 2002.

---

28.7: Direcciones de investigación, lecturas adicionales, bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 28.7: Research Directions, Further Reading, Bibliography has no license indicated.

## 28.8: Ascendencia Europea y Migraciones

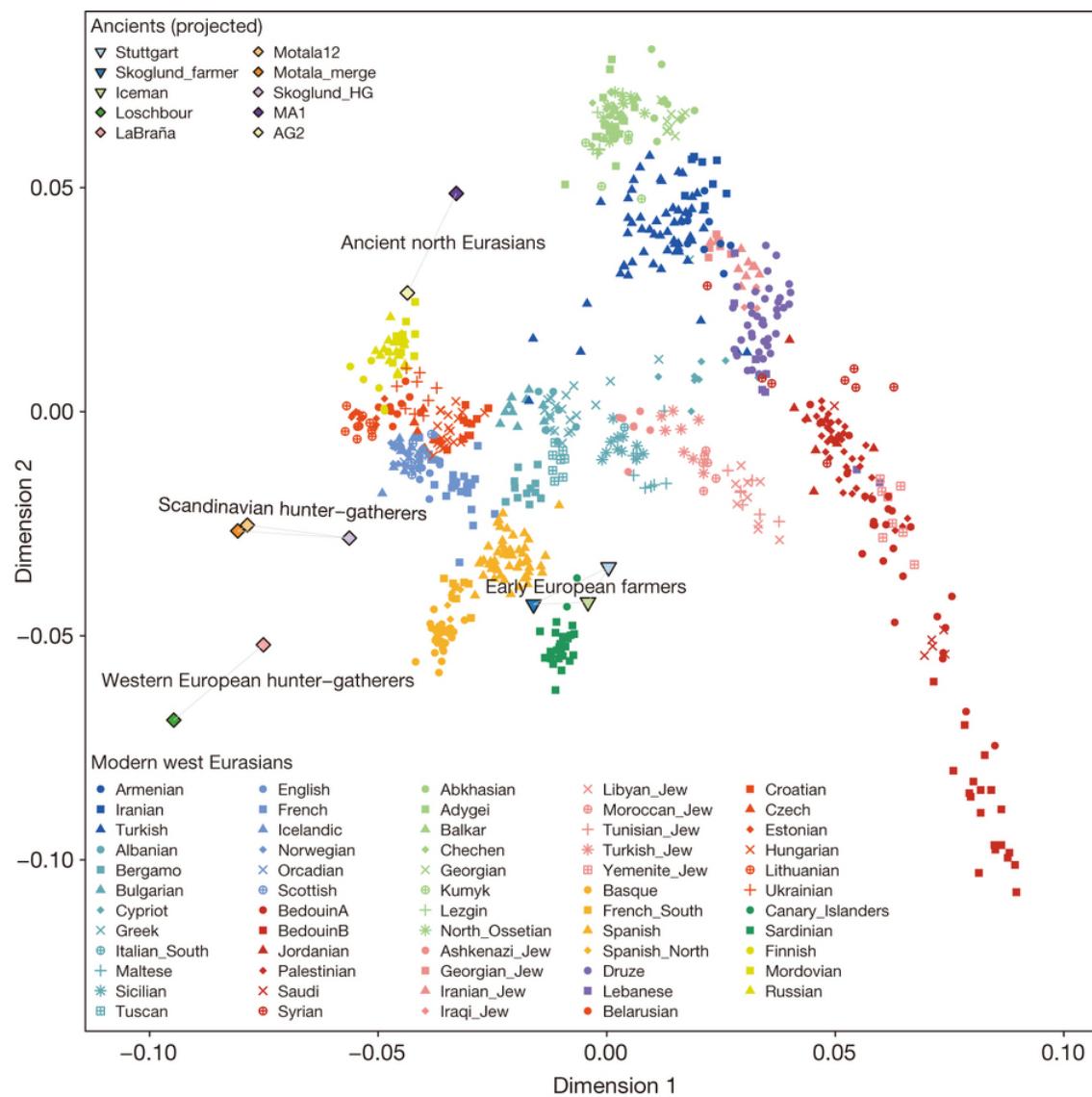
### Rastreando los orígenes de la genética europea

Antes de 2014, se creía que la genética europea moderna era principalmente una mezcla de dos poblaciones ancestrales. La primera población es lo que se conoce como la población occidental cazador-gathere (WHG), y es considerada la población indígena europea. La segunda población es conocida como la población campesina de Europa Temprana (EEF), y representa la rápida migración de los pueblos agrícolas a Europa, y la posterior mezcla de la nueva población agrícola con la población original WHG. Sin embargo, en 2012, Patterson et al. [?] utilizó el análisis de componentes principales de la Figura 29.6 para mostrar que la genética europea no coincide con ser una mezcla de solo estas dos poblaciones. Más bien, el análisis de mezcla genética mostró que algunos europeos solo podían explicarse como una mezcla de poblaciones de EEF/WHG con una tercera población cuya genética se asemejaba a los nativos americanos. Si bien esto no significa que los nativos americanos sean ancestrales de los europeos, el estudio concluyó que la hipótesis más probable fue la mezcla de estas dos poblaciones conocidas con una población de la Antigua Eurasia del Norte (ANE) que migró tanto a Asia como a Europa, y ya no se encuentra en el norte de Eurasia. Este estudio llamó a esta población misteriosa el “Fantasma del norte de Eurasia”.

Dos años después, en 2014, sin embargo, se encontró una muestra confirmando la existencia de esta población. Proclamando que “Se encuentra el fantasma”, Raghavan, Skoglund et al. [?] estudió la muestra recién encontrada de “Mal'ta” del lago Baikal (actualmente en el sur de Rusia) y determinó que coincidía con la población fantasma predicha a partir de 2012, y podría explicar la variación bidimensional en las poblaciones europeas modernas. En particular, se encontró que los europeos modernos estaban compuestos por 0-50% WHG, 32-93% EEF y 1-18% de poblaciones de ANE.

### Migración desde la estepa

Ante esta nueva población como fuente de ascendencia europea, las preguntas naturales son ¿cuándo y por qué migraron a Europa los miembros de la población del ANE? La respuesta, por supuesto, se puede sacar de más datos genéticos sobre la historia de las poblaciones europeas. La primera pista se encontró en datos de ADN mitocondrial, en un artículo de 2013 de Brandt, Haak et al. [?], que encontró que había dos discontinuidades en el ADN mitocondrial europeo: una entre el Mesolítico y el Neolítico temprano, y otra entre el Neolítico medio y el Neolítico Tardío y el Bronce. En 2014, estudios de 9, y luego 94 muestras de antiguos individuos europeos mostraron claramente los dos eventos migratorios, visualizados en la Figura 29.7. La primera migración, aproximadamente al 6500 BCE, fue una migración de la población EEF, que reemplazó a la población WHG existente a una tasa de entre 60 y 100%. La segunda migración fue una migración de pastores esteparios, conocidos como los Yamnaya, que reemplazaron a la población existente con una tasa entre 60 y 80%. En ambos



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Lazaridis, Iosif, et al. "Genomas Humanos Antiguos Sugieren Tres Ancestrales Poblaciones para los europeos actuales". *Naturaleza* 513, núm. 7518 (2014): 409-13

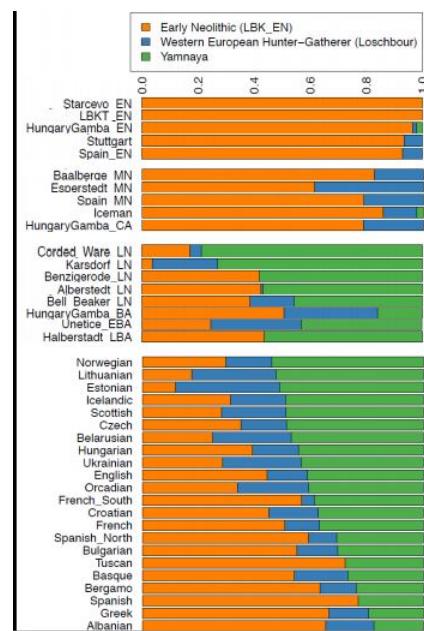
Figura 28.6: Antiguos europeos proyectados sobre el PCP bidimensional de todas las poblaciones europeas modernas. Los europeos occidentales modernos, representados principalmente por el clino inferior izquierdo, no pueden describirse como una mezcla de solo poblaciones de EEF y WHG. Sin embargo, con la adición de un componente ANE, se pueden explicar las variaciones.

casos, la población migratoria se hace cargo de una parte de la composición genética casi de inmediato, y luego la población anterior resurge gradualmente a lo largo de varios miles de años.

### Cribado para Selección Natural

Otra aplicación de los datos de ADN a la historia es en el rastreo de eventos de selección natural. Esencialmente, se pueden observar las frecuencias de varios alelos en los datos de ADN europeos modernos, y encontrar casos en los que no coincide con el modelo de mezcla ancestral de la población. Tales casos tenderán a significar alelos que han sido seleccionados a favor o en contra desde que ocurrieron los eventos ancestrales de mezcla. El ejemplo más fácil de identificar y más conocido de tal rasgo es la persistencia de lactasa. El nivel actual de prevalencia de este rasgo está muy por encima de cualquiera de los niveles representados por las poblaciones ancestrales, lo que sugiere que se sometió a selección positiva (debido a la domesticación y ordeño de animales) desde los eventos de mezcla ancestral.

También se pueden detectar varios otros rasgos como candidatos para la selección. Otro ejemplo sencillo es la pigmentación de la piel. Más interesante es el cuento de selección de altura que muestra la genética de los europeos del norte y del sur. En particular, los datos muestran que se produjeron dos efectos de selección distintos. Primero, los primeros agricultores del sur de Europa se sometieron a selección para disminuir la altura entre 8000 y 4000 años atrás. Segundo, los pueblos del norte de Europa (escandinavos modernos, etc.) fueron sometidos a una selección positiva alrededor del mismo periodo de tiempo y a través del presente. Si bien se cuestionan las explicaciones antropológicas de estos efectos, los propios efectos se muestran claramente en los datos genéticos.



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Haak, Wolfgang, et al. "Migración masiva desde la estepa fue una fuente de lenguas indoeuropeas en Europa". *Naturaleza* (2015).

Figura 29.7: La composición genética europea a lo largo del tiempo muestra dos migraciones masivas: primero, la migración de la población EEF, reemplazando casi por completo a la población nativa WHG; y segundo, la migración de la población ANE Yamnaya, reemplazando alrededor del 75% de la población nativa en ese momento.

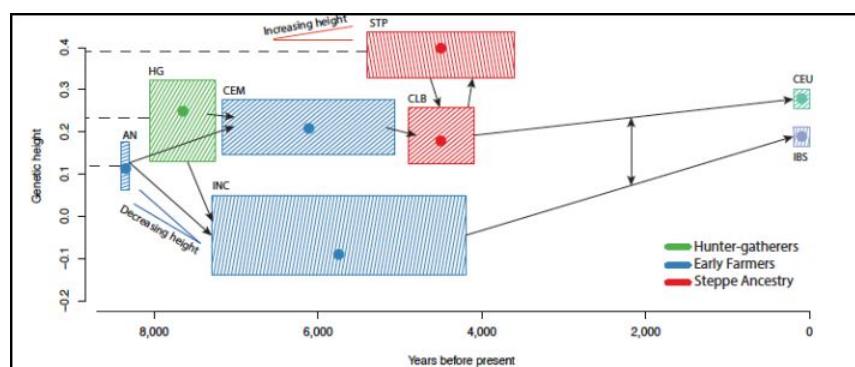


Figura 28.8: Selección de altura en poblaciones europeas desde hace 8000 años hasta la actualidad.

Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Mathieson, Iain et al. "Patrones de selección de todo el genoma en 230 antiguos eurasiáticos." *Naturaleza* 528, núm. 7583 (2015): 499-503.

---

[28.8: Ascendencia Europea y Migraciones](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [28.8: European Ancestry and Migrations](#) has no license indicated.

## CHAPTER OVERVIEW

### 29: Variación genética poblacional

- 29.1: Introducción
- 29.2: Conceptos básicos de selección de población
- 29.3: Vinculación genética
- 29.4: Selección natural
- 29.5: Evolución Humana
- 29.6: Investigación actual
- 29.7: Lectura adicional

---

29: Variación genética poblacional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 29.1: Introducción

Durante siglos, los biólogos tuvieron que confiar en las propiedades morfológicas y fenotípicas de los organismos para inferir el árbol de la vida y hacer conjeturas educadas sobre la historia evolutiva de las especies. Solo recientemente, la capacidad de secuenciar genomas enteros a bajo costo y encontrar patrones en ellos ha transformado la biología evolutiva. La secuenciación y comparación de genomas a nivel molecular se ha convertido en una herramienta fundamental que nos permite conocer la historia evolutiva mucho más antigua que antes, pero también entender la evolución a una resolución de tiempo mucho menor. Con estas nuevas herramientas, no sólo podemos aprender la relación entre clados distantes que separaron hace miles de millones de años, sino también comprender el pasado presente y reciente de especies e incluso diferentes poblaciones dentro de una especie.

En este capítulo discutiremos el estudio de la historia genética humana y la selección reciente. El marco metodológico de esta sección se basa en gran medida en los conceptos de capítulos anteriores. Más específicamente, los métodos para el mapeo de asociación de enfermedades y construcciones filogenéticas como la construcción de árboles entre especies y genes, y la historia de mutaciones usando coalescencia. Habiendo aprendido sobre estos métodos en el último capítulo, ahora estudiaremos cómo su aplicación puede informarnos sobre las relaciones, y diferencias entre las poblaciones humanas. Adicionalmente, buscaremos cómo se pueden explotar estas diferencias para buscar señales de selección natural reciente y la identificación de loci de enfermedades. También discutiremos en este capítulo lo que conocemos actualmente sobre las diferencias entre poblaciones humanas y describiremos algunos parámetros que podemos inferir que cuantifican las diferencias poblacionales, utilizando únicamente la extensión de la variación genética que observamos. En el estudio de la historia genética humana y la selección reciente, hay dos temas principales de investigación que a menudo se estudian. El primero es la historia del tamaño de la población. El segundo es la historia de interacciones entre poblaciones. A menudo se hacen preguntas sobre estas áreas porque las respuestas a menudo pueden proporcionar conocimientos para mejorar el proceso de mapeo de enfermedades. Hasta el momento, todo el conocimiento actual basado en la investigación de la historia humana se encontró investigando regiones funcionalmente neutras del genoma, y asumiendo deriva genética. La razón por la que se emplean regiones neuronales es porque las mutaciones están sujetas a presión de selección positiva, negativa y de equilibrio, cuando tienen lugar en una región funcional. Por lo tanto, investigar las regiones neuronales proporciona un proxy imparcial de selección para la deriva entre especies. En este capítulo profundizaremos en algunas de las características del proceso de selección en humanos y buscaremos patrones de variación humana en términos de comparaciones cruzadas de especies, comparación de mutaciones sinónimas y no sinónimas, y estructura de haplotipos.

---

29.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [29.1: Introduction](#) has no license indicated.

## 29.2: Conceptos básicos de selección de población

### Polimorfismos

Los polimorfismos son diferencias de apariencia entre miembros de una misma especie. Muchos de ellos surgen de mutaciones en el genoma. Estas mutaciones, o polimorfismos genéticos, se pueden caracterizar en diferentes tipos.

#### Polimorfismos de un solo nucleótido (SNP)

- La mutación de una sola base nucleotídica dentro de una secuencia. En la mayoría de los casos, estos cambios no tienen consecuencias. Sin embargo, hay algunos casos en los que la mutación de un solo nucleótido tiene un efecto mayor.
- Por ejemplo, es causada por una de A a T, que provoca un cambio de ácido glutámico (GAG) a valina (GTG) en la hemoglobina.

#### Número variable de repeticiones en tandem

- Cuando una secuencia corta se repite varias veces, la ADN Polimerasa a veces puede “deslizarse”, haciendo que haga demasiadas o muy pocas copias de la repetición. Esto se llama un.
- Por ejemplo, la **enfermedad de Huntington** que es causada por demasiadas repeticiones del trinucleótido CAG se repite en el gen HTT. Tener más de 36 repeticiones puede llevar a una pérdida gradual del control muscular y a una degradación neurológica severa. Generalmente, cuantas más repeticiones haya, más fuertes serán los síntomas.

#### Inserción/eliminación

- A través de copia defectuosa o reparación de ADN, o de uno o múltiples nucleótidos puede ocurrir.
- Si la inserción o delección está dentro de un exón (la región codificante de proteínas de un gen) y no consiste en un múltiplo de tres nucleótidos, ocurrirá a.
- El primer ejemplo son las delecciones en el gen CFTR, que codifica canales de cloruro en los pulmones y puede causar Fibrosis Quística donde el paciente no puede limpiar las mucosas en los pulmones y causa infección

#### ¿Sabías?

El perfil de ADN se basa en repeticiones en tandem de números variables cortos (STR). El ADN se corta con ciertas enzimas de restricción, dando como resultado fragmentos de longitud variable que pueden ser utilizados para identificar a un individuo. Diferentes países utilizan loci diferentes (pero a menudo superpuestos) para estos perfiles. En Norteamérica se utiliza un sistema basado en 13 loci.

### Frecuencias de alelos y genotipos

Para entender la evolución de una especie a través del análisis de alelos o genotipos, debemos tener un modelo de cómo se transmiten los alelos de una generación a otra. Es de inmensa importancia que el lector tenga una firme intuición para el modelo Hardy-Weinberg Principle y Wright Fisher antes de continuar. Por lo tanto, proporcionaremos aquí un breve recordatorio de modelar la historia de las mutaciones a través de estos métodos. Introducido por primera vez hace más de cien años, el Modelo Wright-Fisher es un modelo matemático de deriva genética en una población. Específicamente, describe la probabilidad de obtener  $k$  copias de un nuevo alelo  $p$  dentro de una población de tamaño  $N$ , con una frecuencia no mutante de  $q$ , y cuál será su frecuencia esperada en generaciones sucesivas.

#### Principio Hardy-Weinberg

Se afirma que las frecuencias de alelos y genotipos dentro de una población permanecerán constantes a menos que exista una influencia externa que los empuje lejos de ese equilibrio.

El principio Hardy-Weinberg se basa en los siguientes supuestos:

- La población observada es muy grande
- La población está aislada, es decir, no hay introducción de otra subpoblación en la población general
- Todos los individuos tienen la misma probabilidad de producir descendencia
- Todo el apareamiento en la población es al azar

- No se producen mutaciones aleatorias en la población de una generación a otra
- La frecuencia alélica impulsa la frecuencia futura del genotipo (el alelo prevalente impulsa el genotipo prevalente)

En un Equilibrio Hardy-Weinberg, para dos alelos A y a, que ocurren con probabilidad  $p$  y  $q = 1-p$ , respectivamente, las probabilidades de un individuo escogido aleatoriamente que tenga los genotipos homocigotos AA o aa ( $p^2$  o  $q^2$ , respectivamente) o heterocigóticos Aa o aA ( $2pq$ ) pueden describirse mediante la ecuación:

$$p^2 + 2pq + q^2 = 1$$

Esta ecuación da una tabla de probabilidades para cada genotipo, la cual puede compararse con las frecuencias observadas del genotipo mediante pruebas de error estadístico como la prueba de chi-cuadrado para determinar si el modelo Hardy-Weinberg es aplicable. La Figura 29.1 muestra la distribución de frecuencias de genotipos a diferentes frecuencias alélicas.

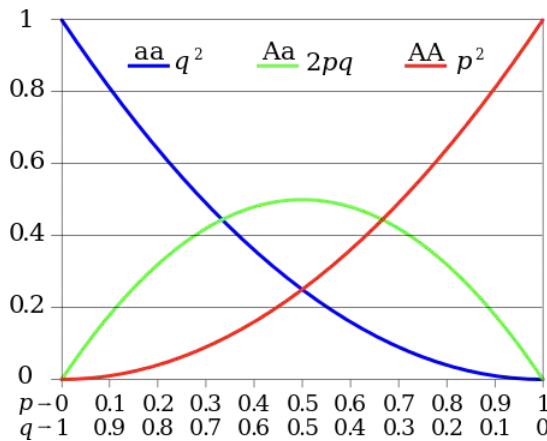


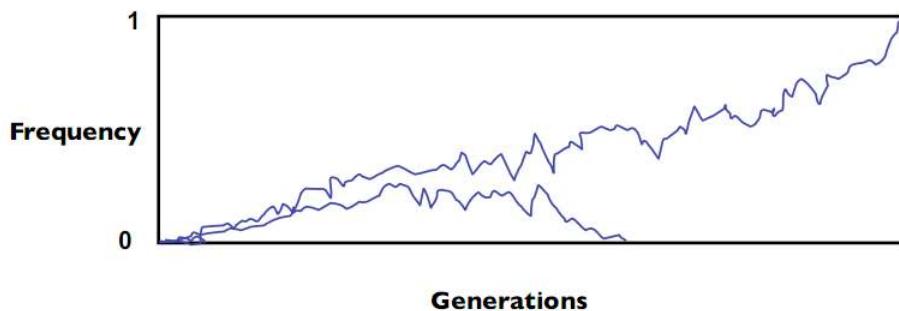
Figura 29.1: Gráfica de frecuencias de genotipo para diferentes frecuencias alélicas

En poblaciones naturales, los supuestos hechos por el principio Hardy-Weinberg rara vez se mantendrán. Se produce la selección natural, las poblaciones pequeñas sufren deriva genética, las poblaciones se dividen o se fusionan, etc. En la naturaleza una mutación siempre desaparecerá (frecuencia = 0) de la población o se volverá prevalente en una especie - esto se llama fijación; en general, el 99% de las mutaciones desaparecen. La Figura 29.2 muestra una simulación de una prevalencia de mutaciones en una población de tamaño finito a lo largo del tiempo: ambas realizan caminatas aleatorias, con una mutación desapareciendo y la otra haciéndose prevalente:

Una vez que una mutación ha desaparecido, la única manera de que reaparezca es la introducción de una nueva mutación en la población. Para los humanos, se cree que una mutación dada bajo ninguna presión selectiva debe fijarse a 0 o 1 (dentro de, por ejemplo, 5%) en unos pocos millones de años. No obstante, bajo selección esto sucederá mucho más rápido.

### Modelo Wright-Fisher

Bajo este modelo el tiempo de fijación es  $4N$  y la probabilidad de fijación es  $1/(2N)$ . En general Wright-Fisher se utiliza para responder preguntas relacionadas con la fijación de una forma u otra. Para asegurarse de que sus intuiciones sobre



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 29.2 Cambios en la frecuencia alélica a lo largo del tiempo.

el método es absolutamente claro considerando las siguientes preguntas:

#### FAQ

P: Digamos que tiene un total de 5 mutaciones en un cromosoma entre una población de tamaño 30, en promedio, ¿cuántas mutaciones estarán presentes en la próxima generación si cada entidad produce solo un hijo?

R: Si cada progenitor tiene una sola descendencia, entonces habrá, en promedio, 5 mutaciones en la siguiente generación porque la expectativa de frecuencias alélicas es que se mantengan constantes de acuerdo con el principio de equilibrio Hardy-Weinberg en biología básica.

#### FAQ

P: ¿Es razonable la suposición del principio de Equilibrio Hardy-Weinberg sobre la frecuencia constante de alelos?

R: No, la realidad es mucho más compleja ya que hay estocástico en el tamaño de la población y selección en cada generación. Una forma más apropiada de imaginarlo es dibujar imágenes de alelos de un conjunto de padres, variando la cantidad de alelos en la siguiente generación con el tamaño de la población. De ahí que la frecuencia en la próxima generación muy bien podría subir o bajar. Observe aquí que si la frecuencia alélica va a cero siempre estará en cero. La probabilidad en cada generación sucesiva es menor si está bajo selección negativa y mayor si está bajo selección positiva. De ahí que si se trata de una mutación beneficiosa el tiempo de fijación será menor, si la mutación es perjudicial la fijación será mayor. Si no hay descendencia con una mutación dada, entonces tampoco habrá ningún difunto con esa mutación. Sin embargo, si uno produce múltiples offspring, quienes a su vez producen múltiples crías propias, entonces hay una mayor probabilidad de que esta frecuencia alélica se eleve.

#### FAQ

P: Considera que el individuo humano promedio lleva aproximadamente 100 mutaciones completamente únicas. Entonces, cuando un individuo produce descendencia podríamos esperar que la mitad (o 50) de esas mutaciones puedan aparecer en el niño porque en cada espermatozoide u óvulo, 50 de esas mutaciones estarán presentes, en promedio. Por lo tanto, es probable que la descendencia de un individuo herede aproximadamente 100 mutaciones, 50 de un progenitor y 50 de otro, además de sus propias mutaciones únicas que provienen de ninguno de los padres. Con esto en mente, uno podría estar interesado en comprender cuáles son las posibilidades de que algunas mutaciones aparezcan en la próxima generación si un individuo produce, digamos, n hijos. ¿Cómo se puede hacer esto?

R: Pista: Para calcular este valor, asumimos que algún alelo se origina en el fundador, en algún cromosoma arbitrario (1 por ejemplo). Entonces nos hacemos la pregunta, ¿cuántos cromosomas 1 existen en toda la población? Por el momento, el tamaño de la población humana es de 7 mil millones, cada uno con dos copias del cromosoma 1.

Las preguntas y respuestas anteriores deberían dejar muy claro que la suposición estándar de Hardy-Weinberg de que las frecuencias alélicas permanecen constantes de una generación a la siguiente se viola en muchos casos naturales, incluyendo migración, mutación genética y selección. En el caso de la selección, este tema se aborda modificando la definición formal para incluir un  $S$ , término que mide el sesgo en genotipos debido a la selección. Consulte el cuadro 29.1 para una comparación de las versiones originales y compensadas de selección:

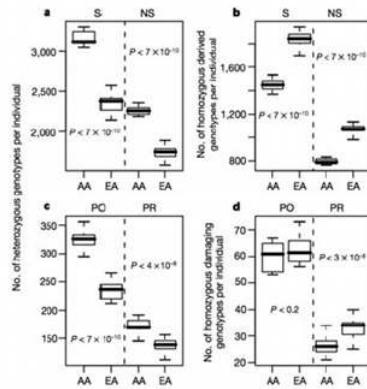
| Comportamiento             | Con solo deriva                                | Con deriva y selección   |
|----------------------------|--|--|
| n en la próxima generación | Media: $n (=2Np)$ , Dist: Binomial ( $2N, p$ ) | Media: $n \left(1 + \frac{s}{1+ns}\right)$ , Dist: Binomial ( $2N, 2N, p \frac{1+s}{1+ps}$ ) |
| Tiempo hasta la fijación   | $4N$   | $\frac{4N}{1+\frac{3}{4}Ns s } \left( \frac{1+\frac{1}{2}(\ln N) s }{1+ s } \right)$         |
| Probabilidad de fijación   | $\frac{1}{2N}$                                 | $\frac{1-e^{-2s}}{1-e^{-4Ns}}$   |

Cuadro 29.1: Comparación del Modelo Wright-Fisher con Deriva, Versus Deriva y Selección

El punto principal a quitar de la Tabla 29.1, y esta sección del capítulo es que el clima tienes selección o no, es muy poco probable que un solo alelo se fije en una población. Si tienes una población muy pequeña, sin embargo, entonces las posibilidades de que se fije un alelo son mucho mejores. Esto suele ser el caso en las poblaciones humanas, donde a menudo hay poblaciones pequeñas, entrecruzadas que permiten que las mutaciones se fijen en una población después de solo unas pocas generaciones, aunque la mutación sea de naturaleza deletérea. Precisamente por eso tendemos a ver trastornos de mandolina deletéreos recesivos en poblaciones aisladas.

## Estado Ancestral de Polimorfismos

¿Cómo podemos determinar para un polimorfismo dado qué versión fue la y cuál es la mutante? El estado ancestral se puede inferir comparando el genoma con el de una especie estrechamente relacionada (por ejemplo, humanos y chimpancés) con un árbol filogenético conocido. Las mutaciones pueden ocurrir en cualquier lugar a lo largo del árbol filogenético a veces las mutaciones en la división se fijan de manera diferente en diferentes poblaciones (“diferencia fija”), en cuyo caso las poblaciones enteras difieren en genotipo. Sin embargo, las mutaciones recientes no habrán tenido tiempo suficiente para fijarse, y un polimorfismo estará presente en una especie pero completamente ausente en la otra ya que las mutaciones simultáneas en ambas especies son muy raras. En este caso, la “variante derivada” es la versión del polimorfismo que aparece después de la división, mientras que la variante ancestral es la versión que ocurre en ambas especies.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 30.3: Comparación de los genotipos derivados y dañinos hetrozigóticos y homocigóticos por individuo en un estudio poblacional afroamericano (AA) y europeo americano (EA).

### 29.2.4 Medición de frecuencias alélicas derivadas

La frecuencia del alelo derivado en la población puede calcularse fácilmente, si asumimos que la población es homogénea. Sin embargo, esta suposición puede no sostenerse cuando existe una división invisible entre dos grupos que hace que evolucionen por separado como se muestra en la figura 29.4.

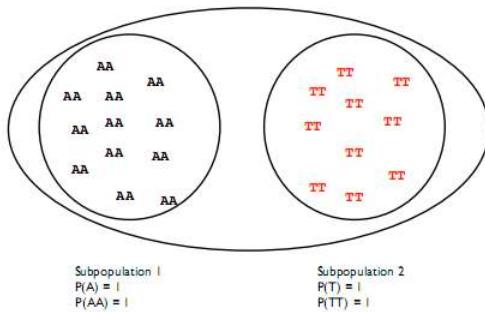


Figura 29.4: Dos poblaciones aisladas

En este caso la prevalencia de las variantes entre subpoblaciones es diferente y se viola el principio de Hardy-Weinberg.

Una forma de cuantificar esta diferencia es usar el ( $F_{st}$ ) para comparar subpoblaciones dentro de una especie. En realidad, solo una porción de la heterocigosidad total en una especie se encuentra en una subpoblación dada.  $F_{st}$  estima la reducción de heterocigosidad ( $2pq$  con alelos  $p$  y  $q$ ) esperada cuando 2 poblaciones diferentes se agrupan erróneamente juntas. Dada una población que tiene  $n$  alelos con frecuencias  $p_i$  donde ( $1 \leq i \leq n$ ), la homocigosidad  $G$  de la población se calcula como:

$$\sum_{i=1}^n p_i^2$$

La heterocigosidad total en la población viene dada por  $1-G$ .

$$F_{st} = \frac{\text{Heterozygosity(total)} - \text{Heterozygosity(subpopulation)}}{\text{Heterozygosity(total)}}$$

En el caso mostrado en la figura 29.4 no hay heterocigosidad entre las poblaciones, por lo que  $F_{st} = 1$ . En realidad el  $F_{st}$  será pequeño dentro de una especie. En humanos, por ejemplo, es sólo 0.0625. Porque en la práctica, el  $F_{st}$  se calcula agrupando subpoblaciones aleatoriamente o usando una característica obvia como etnia u origen.

29.2: Conceptos básicos de selección de población is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 29.2: Population Selection Basics has no license indicated.

## 29.3: Vinculación genética

En los modelos simples que hemos visto hasta ahora, se supone que los alelos se transmiten independientemente unos de otros. Si bien este supuesto generalmente se mantiene a largo plazo, en el corto plazo generalmente observaremos un que ciertos alelos se transmiten juntos con más frecuencia de lo esperado. Esto se denomina vinculación genética.

La, también conocida como segunda ley de Mendel establece:

*Los alelos de diferentes genes se transmiten independientemente de padres a hijos.*

Cuando esta “ley” se sostiene, no hay correlación entre diferentes polimorfismos y la probabilidad de un haplotipo (un conjunto dado de polimorfismos) es simplemente el producto de las probabilidades de cada polimorfismo individual.

En el caso en que los dos genes se encuentran en cromosomas diferentes, generalmente se mantiene esta suposición de independencia, pero si los dos genes se encuentran en el mismo cromosoma, la mayoría de las veces se transmiten juntos. Sin eventos de recombinación genética, en los que se intercambian segmentos de ADN en cromosomas homólogos (cruzamiento), los alelos de los dos genes permanecerían perfectamente correlacionados. Sin embargo, la correlación entre los genes se reducirá a lo largo de varias generaciones. Durante un intervalo de tiempo adecuadamente largo, la recombinación eliminará completamente el enlace entre dos polimorfismos; momento en el que se dice que están en equilibrio. Cuando, por otro lado, los polimorfismos están correlacionados, tenemos **Disequilibrium de Vinculación (LD)**. La cantidad de desequilibrio es la diferencia entre las frecuencias observadas de haplotipos y las predichas en equilibrio.

El desequilibrio de ligamiento se puede utilizar para medir la diferencia entre surtimientos observados y esperados. Si hay dos alelos (1 y 2) y dos loci (A y B) podemos calcular las probabilidades de haplotipos y encontrar las frecuencias alélicas esperadas.

- Frecuencias de haplotipos

$$— P(A_1) = x_{11}$$

$$— P(B_1) = x_{12}$$

$$— P(A_2) = x_{21}$$

$$— P(B_2) = x_{22}$$

- Frecuencias alélicas

$$— P_{11} = x_{11} + x_{12}$$

$$— P_{21} = x_{21} + x_{22}$$

$$— P_{12} = x_{11} + x_{21}$$

$$— P_{22} = x_{12} + x_{22}$$

- $D = P_{11} * P_{22} - P_{12} * P_{21}$

$D_{\max}$ , el valor máximo de D con frecuencias alélicas dadas, se relaciona con D en la siguiente ecuación:

$$D' = \frac{D}{D_{\max}}$$

D' es el desequilibrio máximo de ligamiento o sesgo completo para los alelos y frecuencias alélicas dadas.  $D_{\max}$  se puede encontrar tomando la menor de las frecuencias esperadas de haplotipos  $P(A_1, B_2)$  o  $P(A_2, B_1)$ . Si los dos loci están en completo equilibrio, entonces  $D' = 0$ . Si  $D' = 1$ , hay enlace completo.

El punto clave es que las mutaciones relativamente recientes no han tenido tiempo de ser desglosadas por cruces. Normalmente, tal mutación no será muy común. Sin embargo, si está bajo selección positiva, la mutación será mucho más prevalente en la población de lo esperado. Por lo tanto, combinando cuidadosamente una medida de LD y frecuencia alélica derivada, podemos determinar si una región está bajo selección positiva.

La decadencia de es impulsada por la tasa de recombinación y el tiempo (en generaciones) y tiene una decadencia exponencial. Para una mayor tasa de recombinación, el desequilibrio de ligamiento se desintegrará más rápido en un período de tiempo más

corto. Sin embargo, la tasa de recombinación de fondo es difícil de estimar y varía dependiendo de la ubicación en el genoma. La comparación de datos genómicos entre múltiples especies puede ayudar a determinar estas tasas de fondo.

### 29.3.1 Coeficiente de correlación $r^2$

Respuestas cómo predictivo es un alelo en el locus A de un alelo en el locus B

$$r^2 = \frac{D^2}{P(A_1)P(A_2)P(B_1)P(B_2)}$$

A medida que el valor de  $r^2$  se acerca a 1, se correlacionan más dos alelos en dos loci. Puede haber desequilibrio de ligamiento entre dos haplotipos, incluso si los haplotipos no están correlacionados en absoluto. La correlación coeiciente es particularmente interesante cuando se estudian asociaciones de enfermedades con genes, donde conocer el genotipo en el locus A puede no predecir una enfermedad mientras que el locus B sí. También existe la posibilidad de que ni el locus A ni el locus B sean predictivos de la enfermedad por sí solos, sino que los loci A y B juntos sean predictivos.

---

29.3: Vinculación genética is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 29.3: Genetic Linkage has no license indicated.

## 29.4: Selección natural

A mediados del siglo XIX el concepto de evolución no era una idea infrecuente, pero no fue antes de que Darwin y Wallace propusieran la selección natural como mecanismo que impulsa la evolución en la naturaleza que la teoría de la evolución obtuvo un reconocimiento generalizado. Pasaron 70 años (1948) hasta que J.B.S Haldane's Malaria Hypothesis encontró el primer ejemplo de selección natural en humanos. Mostró una correlación entre las mutaciones genéticas en los glóbulos rojos y la distribución de la prevalencia de malaria y descubrió que los individuos que tenían una mutación específica que los hacía sufrir anemia falciforme también los hacían resistentes a la malaria.

La tolerancia a la lactosa (que dura hasta la edad adulta) es otro ejemplo de selección natural. Tales ejemplos explícitos fueron difíciles de probar sin secuencias genómicas. Con la secuenciación del genoma completo fácilmente disponible, ahora podemos buscar en el genoma regiones con los mismos patrones que estos ejemplos conocidos para identificar otras regiones en proceso de selección natural.

### Señales genómicas de selección natural

- Relación Ka/Ks de cambios no sinónimos a sinónimos por gen
- Baja diversidad y muchos alelos raros en una región (ex D de Tajima con respecto a la anemia falciforme)
- Alta frecuencia alélica derivada (o baja) sobre una región (ex Fay y H de Wu)
- Diferenciación entre poblaciones más rápida de lo esperado a partir de la deriva (Medido con Fst)
- Haplótipos largos: evidencia de barrido selectivo.
- Prevalencia exponencial de una característica en generaciones secuenciales
- Mutaciones que ayudan a que una especie prospere

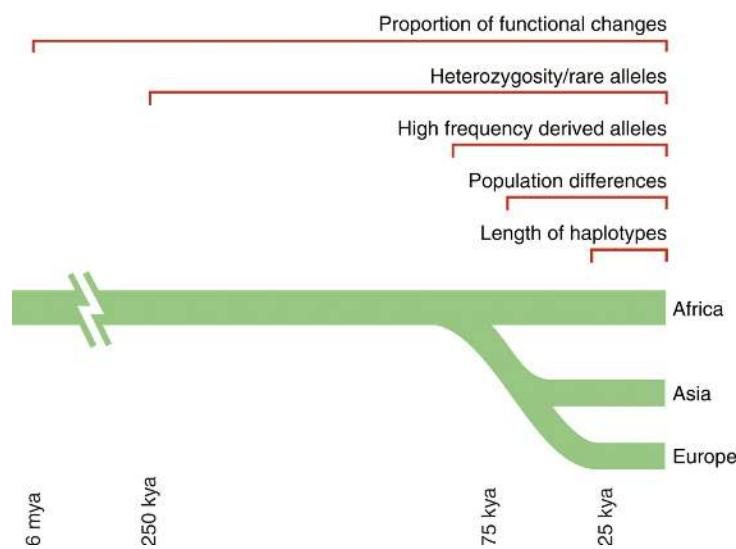


Figura 29.5: Tabla de Tiempo Aproximado de Efectos Sabeti et al. Ciencia 2006

Asociación Americana para el Avance de la Ciencia. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

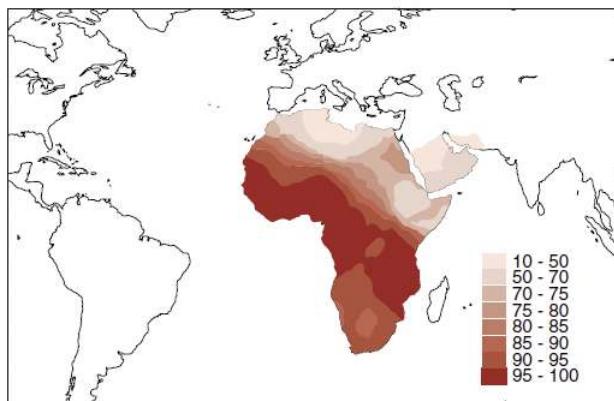
Fuente: Sabeti, P.C., et al. "Selección Natural Positiva en el Linaje Humano". *Ciencia* 312, núm. 5780 (2006): 1614-20.

### Ejemplos de Selección Negativa (Purificante)

- En **todas las especies** vemos selección negativa de nuevas mutaciones en elementos funcionales conservados (exones, etc.).
- Los **alelos nuevos** dentro de una especie tienden a tener frecuencias alélicas más bajas si el alelo no es sinónimo que sinónimo. Los alelos letales tienen frecuencias muy bajas.

### Ejemplos de selección positiva (adaptativa)

- **Síntesis de la selección negativa** en esa selección positiva más probable en elementos funcionales o no.
- En **todas las especies** en un elemento conservado, una mutación seleccionada positivamente podría ser la misma sobre la mayoría de los mamíferos, pero cambio en una especie específica porque una mutación seleccionada positivamente apareció después de la especiación o causó especiación.
- **Dentro de una especie**, los alelos seleccionados de manera positiva probablemente difieren en la frecuencia alélica ( $F_{st}$ ) entre las poblaciones. Los ejemplos incluyen la resistencia a la malaria en poblaciones africanas (29.6) y la persistencia de lactosa en poblaciones europeas (29.7).
- **La selección poligénica** dentro de las especies puede surgir cuando se selecciona un rasgo que depende de muchos genes. Un ejemplo es la altura humana donde se sabe que 139 SNP están relacionados con la altura. La mayoría no son mutaciones específicas de la población, sino alelos en todos los humanos que se seleccionan en algunas poblaciones más que en otras. (29.8)

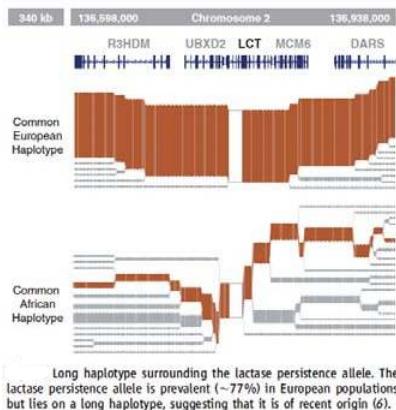


Extreme population differences in  $Fy^*$ O allele frequency. The  $Fy^*$ O allele, which confers resistance to *P. vivax* malaria, is prevalent and even fixed in many African populations, but virtually absent outside Africa (38).

Figura 29.6: Selección positiva localizada para resistencia a la malaria dentro de las especies Sabeti et al. Ciencia 2006

### Pruebas estadísticas

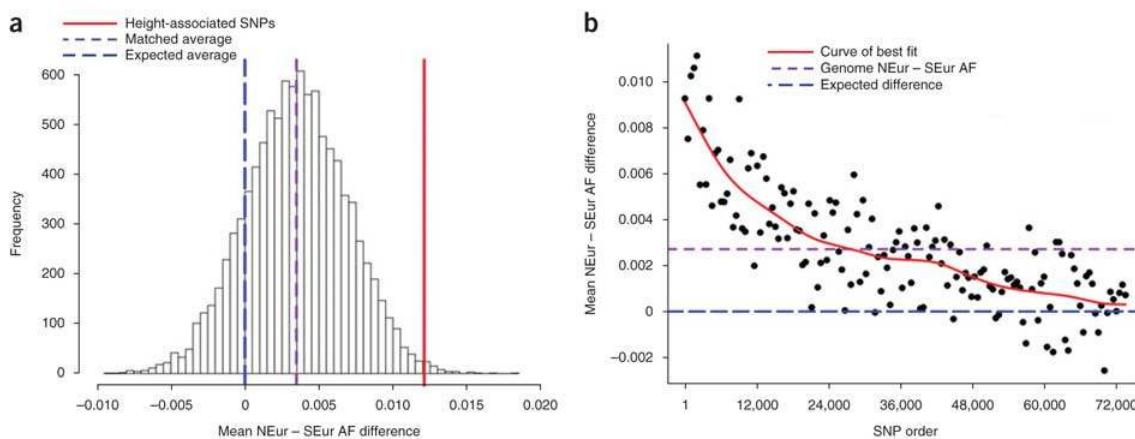
- **Correlaciones de largo alcance (IHs, Xp, EHH):** Si etiquetamos secuencias genéticas en un individuo en función de su ascendencia, terminamos con un haplotipo roto, donde el número de roturas (cambios de color) se correlaciona con el número de recombinaciones y puede decírnos cuánto tiempo hace que estuvo una ascendencia particular introducido.
- **BARREO** Un programa desarrollado por Pardis Sabeti, Ben Fry y Patrick Varilly. SWEEP detecta evidencia de selección natural mediante el análisis de estructuras de haplotipos en el genoma usando el rango largo



Asociación Americana para el Avance de la Ciencia. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Sabeti, P.C., et al. "Selección Natural Positiva en el Linaje Humano". *Ciencia* 312, núm. 5780 (2006): 1614-20.

Figura 29.7: Selección positiva localizada para el alelo de persistencia de lactasa Sabeti et al. Ciencia 2006



Cortesía de Macmillan Publishers Limited. Usado con permiso.

Fuente: Turchin, Michael C., et al. "Evidencia de selección generalizada sobre variación de pie en Europa en SNPs asociados a la Altura". *Nature Genetics* 44, núm. 9 (2012): 1015-9.

Figura 29.8: Diferencia de frecuencia alélica media de SNP de altura, SNPs emparejados y SNPs de todo el genoma entre poblaciones de Europa del Norte y del Sur  
Turchin et al., Nature Genetics (2012)

prueba de haplotipo (LRH). Busca alelos de alta frecuencia con desequilibrio de ligamiento de largo alcance que insinúa una proliferación a gran escala de un haplotipo que ocurrió a una velocidad mayor que la recombinación podría romperlo de sus marcadores.

- **Alelos Derivados de Alta Frecuencia** Busque picos grandes en la frecuencia de alelos derivados en posiciones establecidas.
- **Alta Diferenciación ( $F_{st}$ )** Grandes picos en diferenciación en ciertas posiciones.

Mediante estas pruebas, podemos encontrar regiones genómicas bajo presión selectiva. Un problema es que un solo SNP bajo selección positiva permitirá que los SNP cercanos viajen y viajen. Es difícil distinguir el SNP bajo selección de sus vecinos con una sola prueba. Bajo selección, todas las pruebas son fuertemente

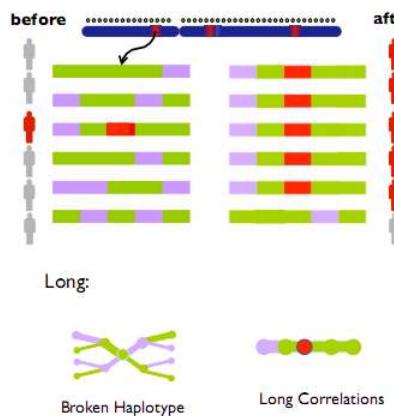


Figura 29.9: Haplótipo roto como señal de selección natural

correlacionados; sin embargo, en ausencia de selección son generalmente independientes. Por lo tanto, al emplear un estadístico compuesto construido a partir de todas estas pruebas, es posible aislar el SNP individual bajo selección.

Ejemplos en los que un solo SNP ha sido implicado en un rasgo:

- Pigmentación de la piel Chr15 en el norte de Europa
- Rasgos de cabello ChR2 en Asia
- Chr10 Rasgo desconocido en Asia
- Rasgo desconocido Chr12 en África

[29.4: Selección natural](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [29.4: Natural Selection](#) has no license indicated.

## 29.5: Evolución Humana

No es sorprendente que la comunidad científica tenga una larga y algo polémica historia de interés en la dinámica poblacional reciente. Si bien efectivamente parte de este interés se aplicó hacia fines más nefastos, como las justificaciones científicas del racismo para la eugenesia pero éstas son cada vez más la excepción y no la regla. Los primeros estudios de la dinámica poblacional fueron primitivos en muchos aspectos. La cuantificación de las diferencias entre las poblaciones humanas se realizó originalmente utilizando tipos de sangre, ya que parecían ser fenotípicamente neutros, podrían probarse para fuera del cuerpo y parecían ser polimórficos en muchas poblaciones humanas diferentes. Avance rápido hasta el presente, y la comunidad científica se ha dado cuenta de que existen otras glicoproteínas más allá de los grupos sanguíneos A, B y O que son mucho más polimórficas en la población. A medida que la ciencia continuó avanzando y la secuenciación se hizo realidad, comenzaron la secuenciación del genoma completo del cromosoma Y, marcadores mitocondriales y microsatélites alrededor de ellos. ¿Qué tienen de especial esos dos tipos de datos genéticos? En primer lugar, son bastante cortos por lo que se pueden secuenciar más fácilmente que otros cromosomas. Más allá del tamaño, la razón por la que los cromosomas Y y mitocondriales fueron de tal interés es porque no se recombinan, y pueden ser utilizados para reconstruir fácilmente árboles heredados. Esto es precisamente lo que hace que estos cromosomas sean especiales en relación con un trozo corto en un autosoma; sabemos exactamente de dónde viene porque podemos rastrear el linaje paterno o materno hacia atrás en el tiempo.

Este tipo de reconstrucción no funciona con otros cromosomas. Si uno fuera a generar un árbol usando cierto trozo de todo el cromosoma 1 en una determinada población, por ejemplo, de hecho formarían una filogenia pero esa filogenia sería escogida de ancestros aleatorios en cada uno de los árboles genealógicos.

A medida que la secuenciación continuaba desarrollándose y haciéndose más efectiva, se estaba proponiendo el proyecto del genoma humano, y junto con él hubo un fuerte impulso para incluir algún tipo de medida de diversidad en los datos genómicos. Técnicamente hablando, fue más fácil simplemente mirar los microsatélites para esta medida de diversidad porque se pueden estudiar en gel para ver polimorfismos de tamaño en lugar de inspeccionar un polimorfismo de secuencia. Como recordatorio, un microsatélite es una región de longitud variable en el genoma humano a menudo caracterizada por repeticiones cortas en tandem. Una razón para los microsatélites son los retrovirus que se insertan en el genoma, como los elementos ALU en el genoma humano. Estos elementos a veces se vuelven activos y se retrotransponen como eventos de inserción y uno puede rastrear cuándo esos eventos de inserción han ocurrido en el linaje humano. De ahí que hubo un impulso, desde el principio, para ensayar estas partes del genoma en una variedad de poblaciones diferentes. Lo realmente atractivo de los microsatélites es que son altamente polimórficos y en realidad se puede inferir su tasa de mutación. De ahí que no sólo podamos decir que existe una cierta relación entre las poblaciones en base a estas tasas, sino que también podemos decir cuánto tiempo han estado evolucionando e incluso cuándo ocurrieron ciertas mutaciones, y cuánto tiempo ha estado en ciertas ramas del árbol filogenético.

### FAQ

P: ¿No se puede hacer esto simplemente con SNP?

R: No se puede hacer muy fácilmente con los SNP.

Puedes hacerte una idea de la edad que tienen en función de su frecuencia alélica, pero también van a ser influenciados por la selección.

Después del proyecto del genoma humano, vino el proyecto Hapmap de herencia de haplotipos que analizó el genoma de los SNP en todo el genoma. Hemos discutido en detalle la herencia de haplotipos en capítulos anteriores donde aprendimos la importancia de Hapmap en el diseño de matrices de genotipado que analizan SNP que marcan haplotipos comunes en la población.

Los efectos de los cuellos de botella en la diversidad humana El uso de esta riqueza de datos a través de estudios y una pléthora de técnicas matemáticas ha llevado a la constatación de que los humanos, de hecho, tienen una diversidad muy baja dada nuestra población censal; lo que implica un pequeño tamaño poblacional efectivo. Utilizando el modelo Wright-Fisher es posible trabajar desde el nivel de diversidad y el número de mutaciones que vemos en la población actual para generar un tamaño de población fundacional. Cuando se realiza este cálculo, resulta ser alrededor de 10,000.

**FAQ**

P: ¿Por qué es esto mucho más pequeño que el tamaño de nuestra población censal?

R: Había un cuello de botella poblacional en alguna parte.

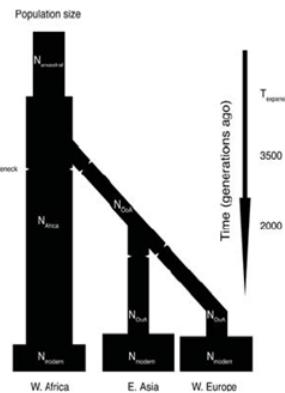


Figura 30.10: Una representación de dos eventos principales de cuello de botella, uno en la población fundadora de África, y otro, eventos de cuello de botella posteriores más pequeños en las poblaciones de Asia Oriental y Europa Occidental.

La mayor parte de la variación total entre humanos ocurre dentro del continente. Se puede medir cuánta diversidad se explica por la geografía y cuánto no lo es. Resulta que la mayor parte no se explica por la geografía. De hecho, las variantes más comunes son polimórficas en cada población y si una variante común es única para una población determinada, probablemente no haya habido tiempo suficiente para que eso suceda por deriva misma. Recordemos lo poco probable que es llegar a una alta frecuencia alélica en el transcurso de varias generaciones solo por mera casualidad. De ahí que podamos interpretar esto como una señal de selección cuando ocurre. Toda la evidencia en términos de comparación de patrones de diversidad y árboles con haplotipos ancestrales converge en una hipótesis fuera de África que es el consenso abrumador en el campo y es la lente a través de la cual revisamos todos los datos genéticos de la población. Partiendo de la población fundadora africana, se han realizado trabajos que han demostrado que es posible modelar el crecimiento poblacional utilizando el modelo wright fisher. Los estudios han demostrado que la tasa de crecimiento que vemos en las poblaciones asiáticas y europeas solo es consistente con un gran crecimiento exponencial después del evento fuera de África.

| Study   | Sample size (n)* | Time growth started (years ago)†       | Initial $N_0$ ‡ | Growth per generation (%)   |
|---|------------------|--|-----------------|-----------------------------|
| Gravel et al. (5)                                       | 60               | 23,000 <sup>§</sup><br>(21,000–27,000) | 1032            | 0.48<br>(0.30–0.75)         |
| Gutenkunst et al. (6)<br>(including New World modeling) | 22               | 26,400 <sup>§</sup><br>(21,700–30,700) | 1500            | 0.23<br>(0.16–0.34)         |
| Gutenkunst et al. (6)<br>(excluding New World modeling) | 22               | 21,200 <sup>§</sup><br>(17,600–23,900) | 1000            | 0.4<br>(0.26–0.57)          |
| Schaffner et al. (29)                                   | 62               | 8750  <br>1400<br>(900–2800)           | 1700            | 0.73  <br>9.4<br>(4.5–14.5) |
| Coventry et al. (18)                                    | 10,422           |  | 7700#           |                             |

Cuadro 30.2: Estimaciones genéticas del crecimiento poblacional reciente en Europa

Esto nos ayuda a comprender las razones de las diferencias fonotípicas entre las razas, ya que los cuellos de botella seguidos de un crecimiento exponencial pueden conducir a un exceso de alelos raros. La presente teoría sobre la diversidad humana establece que hubo eventos secundarios de cuello de botella después de que la población fundadora emigró fuera de África. Estos fundadores fueron, en algún momento anterior, sujetos a un evento de cuello de botella aún menor que ahora se refleja en cada genoma humano del planeta, independientemente de su ascendencia inmediata. Es posible estimar qué tan pequeño era el cuello de botella original observando las diferencias entre individuos de origen africano y europeo, infiriendo los efectos del cuello de botella secundario, y el término de crecimiento exponencial de la población europea. La otra forma de acercarse a la estimación de eventos de cuello de botella es simplemente inspeccionar el espectro de frecuencias alélicas necesario para construir árboles coalescentes. De esta manera, uno puede tomar haplotipos a través del genoma y preguntar cuál fue el ancestro común más reciente observando cómo varía la coalescencia a lo largo del genoma. Por ejemplo, se puede adivinar que algún haplotipo fue seleccionado positivamente para solo recientemente dada la longitud del haplotipo. Un ejemplo de una de esas mutaciones recientes en la población europea es el gen de la lactasa. Otro ejemplo para la población asiática es el locus ER.

Hay una gran cantidad de literatura que muestra que cuando uno dibuja un árbol de coalescencia para la mayoría de los haplotipos termina yendo mucho antes cuando pensamos que ocurrió la especiación. Esto indica que ciertas características se han mantenido polimórficas durante mucho tiempo. Sin embargo, se puede observar esta distribución de características en todo el genoma e inferir algo sobre la historia de la población a partir de ella. Si hubo un cuello de botella reciente en una población, se verá reflejado por los antepasados siendo muy recientes mientras que cosas más antiguas habrán sobrevivido al cuello de botella. Se puede tomar la distribución de los tiempos de coalescencia y ejecutar simulaciones de cómo el efecto del tamaño de la población habría variado con el tiempo. El modelo para realizar este tipo de estudios fue delineado por Li y Durbin. La Figura 29.11 de su estudio ilustra dos eventos de cuello de botella de este tipo. El primero es el cuello de botella que se produjo en África mucho antes de las migraciones fuera del continente. A esto le siguió un cuello de botella específico de la población que resultó de grupos migratorios fuera de África. Esto se refleja en la diversidad de las poblaciones actuales en base a su ascendencia y se puede derivar de observar un par de cromosomas de dos personas cualesquiera en estas poblaciones.

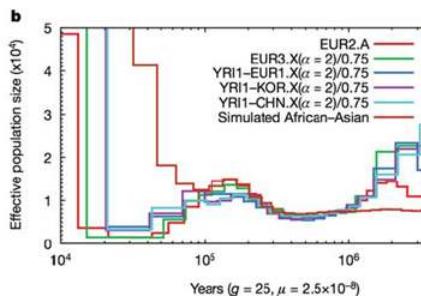


Figura 29.11: Una ilustración de dos eventos de cuello de botella

## Comprensión de la enfermedad

Comprender que las poblaciones humanas pasaron por cuellos de botella tiene implicaciones importantes para la enfermedad específica de la población subpermanente. Un estudio publicado por Tennessen et al. este año analizaba secuencias de exomas en muchas clases de individuos. El estudio tuvo como objetivo analizar cómo las variantes raras podrían estar contribuyendo a la enfermedad y, como consecuencia, fueron capaces de ajustar los modelos genéticos de poblaciones a los datos, y preguntar qué tipo de variantes deletéreas se vieron al secuenciar exomas de un amplio panel poblacional. Con este enfoque, fueron capaces de generar parámetros que describen cuánto tiempo hace que se produjo el crecimiento exponencial entre el fundador y las poblaciones ramificadas. Vea la figura 29.12 a continuación para una ilustración de esto:

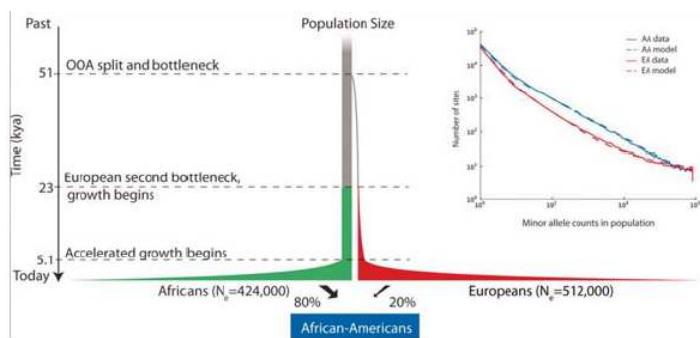


Figura 29.12: La figura ilustra los efectos de eventos de un cuello de botella sobre el número de alelos raros en una población.

## Comprensión de la mezcla poblacional reciente

Además de ver los tiempos de coalescencia, también se puede realizar Análisis de Componentes Principales en SNP para comprender las mezclas poblacionales más recientes. Ejecutar esto en la mayoría de las poblaciones muestra agrupamiento con respecto a la ubicación geográfica. Hay algunas poblaciones, sin embargo, que experimentaron una mezcla reciente por razones históricas. Los dos más comúnmente referidos en la literatura científica son: los afroamericanos, que en promedio son 20%

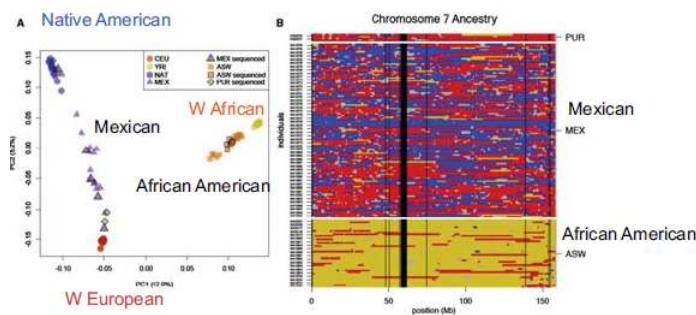


Figura 29.13: Una representación de los niveles de mezcla europea en las poblaciones mexicana y afroamericana.

Hay dos cosas importantes que uno puede decir sobre el evento de mezcla de afroamericanos y mexicoamericanos. El primero y más obvio es inferir el nivel de mezcla. El segundo, y más interesante, es inferir cuándo ocurrió el evento de mezcla basado en el nivel real de mezcla. Como hemos comentado en capítulos anteriores, los significantes raciales del genoma se descomponen con la mezcla a causa de la recombinación en cada generación. Si la población está contenida, el porcentaje de aquellos con origen europeo y de África Occidental debería permanecer igual en cada generación, pero los segmentos se acortarán, debido a la mezcla. De ahí que la longitud de los bloques de haplotipos se pueda utilizar para remontarse a cuando se produjo originalmente la mezcla. (Cuando sucedió originalmente esperaríamos trozos grandes, siendo algunos gambits enteramente de origen africano, por ejemplo). Con este enfoque, se puede observar la distribución de las trampas de ascendencia recientes y luego ajustar un modelo al momento en que estos migrantes ingresaron a una población ancestral como se muestra a continuación:

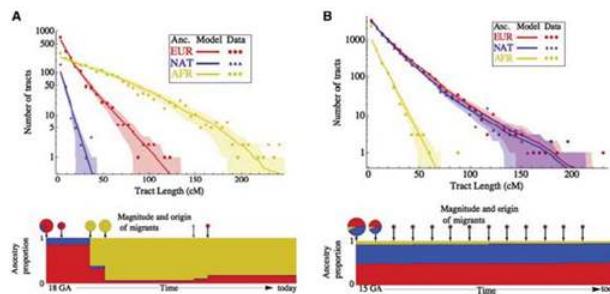


Figura 29.14: Como ilustración de la magnitud y origen de los migrantes con base en la longitud del trato y número de tratos en la población mixta.

29.5: Evolución Humana is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 29.5: Human Evolution has no license indicated.

## 29.6: Investigación actual

### Proyecto HapMap

El Proyecto Internacional tiene como objetivo catalogar los genomas de humanos de diversos países y regiones y encontrar similitudes y diferencias para ayudar a los investigadores a encontrar genes que beneficien el avance en el tratamiento de enfermedades y la administración de tecnologías relacionadas con la salud.

### proyecto genomas

El Proyecto 1000 Genomas es un consorcio internacional de investigadores con el objetivo de establecer un catálogo detallado de la variación genética humana. Su objetivo era secuenciar los genomas de más de mil participantes anónimos de una serie de etnias diferentes. En octubre de 2012, se anunció la secuenciación de 1092 genomas en un artículo de *Nature*. Se espera que los datos recopilados por este proyecto ayuden a los científicos a obtener más información sobre la evolución humana, la selección natural y las variantes raras que causan enfermedades.

---

29.6: Investigación actual is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 29.6: Current Research has no license indicated.

## 29.7: Lectura adicional

- Campbell Biology, 9<sup>a</sup> edición; Pearson; Capítulo 23: La evolución de las poblaciones 474
- The Cell, 5<sup>a</sup> edición, publicación Garland; Capítulos 5: Replicación, reparación y recombinación del ADN, Capítulo 20: Células germinales y fertilización

---

29.7: Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [29.7: Further Reading](#) has no license indicated.

## CHAPTER OVERVIEW

### 30: Genética médica: el pasado hasta el presente

- 30.1: Bibliografía
- 30.2: Introducción
- 30.3: Objetivos de investigar las bases genéticas de la enfermedad
- 30.4: Rasgos mendelianos
- 30.5: Rasgos Complejos
- 30.6: Estudios de Asociación en todo el genoma
- 30.7: Direcciones actuales de investigación
- 30.8: Herramientas y Técnicas
- 30.9: ¿Qué hemos aprendido?

---

30: Genética médica: el pasado hasta el presente is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 30.1: Bibliografía

- [1] G.R. Abecasis, S.S. Cherny, W.O. Cookson, y L.R. Cardon. Merlin: análisis rápido de mapas genéticos densos utilizando árboles de flujo génico dispersos. *Nature Genetics*, 30 (1) :97—101, 2002.
- [2] H.L. Allen et al. Cientos de variantes agrupadas en loci genómicos y vías biológicas afectan altura humana. *Naturaleza*, 467 (7317) :832—838, 2010.
- [3] Y. Benjamini e Y. Hochberg. Controlar la tasa de falsos descubrimientos: Un enfoque práctico y poderoso para múltiples pruebas. *Revista de la Real Sociedad Estadística*, 57:289 —300, 1995.
- [4] D. Botstein, R.L. White, M. Skolnick y R.W. Davis. Construcción de un mapa de ligamiento genético en el hombre mediante polimorfismos de longitud de fragmentos de restricción. *American Journal of Human Genetics*, 32:314 —331, 1980.
- [5] M.S. Brown y J.L. Goldstein. Una vía mediada por receptores para la homeostasis del colesterol. *Ciencia*, 232 (4746) :34—47, 1986.
- [6] Jonathan C. Cohen, Eric Boerwinkle, Thomas H. Mosley y Helen H. Hobbs. Variaciones de secuencia en PCSK9, LDL bajo y protección contra la enfermedad coronaria. *354* (12) :1264—1272.
- [7] B. Devlin y K. Roeder. Control genómico para estudios de asociación. *Biometría*, 55:997 —1004, 1999.
- [8] R.C. Elston y J. Stewart. Un modelo general para el análisis genético de datos de pedigree. *Herencia Humana*, 21:" 523—542", 1971.
- [9] Kyle Kai-How Farh, Alexander Marson, Jiang Zhu, Markus Kleinewietfeld, William J. Housley, Samantha Beik, Noam Shores, Holly Whitton, Russell J. H. Ryan, Alexander A. Shishkin, Meital Hatan, Marlene J. Carrasco-Alfonso, Dita Mayer, C. John Luckey, Nikolaos A. Patsopoulos, Philip L. De Jager, Vijay K. Kuchroo, Charles B. Epstein, Mark J. Daly, David A. Hafler y Bradley E. Bernstein. Mapeo fino genético y epigenético de variantes causales de enfermedades autoinmunes.
- [10] Señor R.A. Fisher. La correlación entre familiares sobre la suposición de herencia mendeliana. *Transacciones de la Real Sociedad de Edimburgo*, 52:399 —433, 1918.
- [11] D.F. Gudbjartsson, K. Jonasson, M.L. Frigge y A. Kong. Allegro, un nuevo programa informático para el análisis de ligamiento multipunto. *Nature Genetics*, 25 (1) :12—13, 2000.
- [12] D.F. Gudbjartsson, T. Thorvaldsson, A. Kong, G. Gunnarsson y A. Ingolfsdottir. Allegro versión 2. *Nature Genetics*, 37 (10) :1015—1016, 2005.
- [13] Joel T. Haas, Harland S. Winter, Elaine Lim, Andrew Kirby, Brendan Blumenstiel, Matthew DeFeo- piojos, Stacey Gabriel, Chaim Jalas, David Branski, Carrie A. Grueter, Mauro S. Toporovski, Tobías C. Walther, Mark J. Daly y Robert V. Farese. La mutación DGAT1 está ligada a un trastorno diarreico congénito. *122* (12) :4680—4684.
- [14] X. Hu, H. Kim, E. Stahl, R. Plenge, M. Daly, y S. Raychaudhuri. La integración de loci de riesgo autoinmune con datos de expresión génica identifica subconjuntos específicos de células inmunitarias patógenas. *The American Journal of Human Genetics*, 89 (4) :496—506, 2011.
- [15] R.M. Idury y R.C. Elston. Un algoritmo de modelo de markov oculto más rápido y general para cálculos de verosimilitud multipunto. *Herencia Humana*, 47:197 —202, 1997.
- [16] A. Ingolfsdottir y D. Gudbjartsson. Algoritmos de análisis de vinculación genética y su implementación. En Corrado Priami, Emanuela Merelli, Pablo Gonzalez, y Andrea Omicini, editores, *Transacciones sobre Biología de Sistemas Computacionales III*, tomo 3737 de Apuntes de conferencia en Ciencias de la Computación, páginas 123—144. Springer Berlín/Heidelberg, 2005.
- [17] L. Kruglyak, M.J. Daly, M.P. Reeve-Daly, y E.S. Lander. Análisis de vinculación paramétrico y no paramétrico: un enfoque multipunto unificado. *American Journal of Human Genetics*, 58:1347 —1363, 1996.
- [18] L. Kruglyak y E.S. Lander. Análisis de enlace multipunto más rápido utilizando transformadas de Fourier. *Revista de Biología Computacional*, 5:1 —7, 1998.

- [19] P. Kuballa, A. Huett, J.D. Rioux, M.J. Daly, y R.J. Xavier. Autofagia alterada de un patógeno intracelular inducida por una variante atg16l1 asociada a la enfermedad de Crohn. *PLoS One*, 3 (10) :e3391, 2008.
- [20] E.S. Lander y P. Green. Construcción de mapas de ligamiento genético multilocus en humanos. *Actas de la Academia Nacional de Ciencias*, 84 (8) :2363—2367, 1987.
- [21] E.S. Lander, P. Green, J. Abrahamson, A. Barlow, M.J. Daly, S.E. Lincoln, y L. Newburg. Mapmaker: Un paquete informático interactivo para la construcción de mapas de vinculación genética primaria de poblaciones experimentales y naturales. *Genómica*, 1 (2) :174—181, 1987.
- [22] Q. Li, J.B. Brown, H. Huang y P.J. Bickel. Medición de la reproducibilidad de los expertos de alto rendimiento. *Anales de Estadística Aplicada*, 5:1752 —1797, 2011.
- [23] E.Y. Liu, Q. Zhang, L. McMillan, F.P. de Villena y W. Wang. Inferencia de ascendencia genómica eficiente en pedigríes complejos con endogamia. *Bioinformática*, 26 (12) :i199—i207, 2010.
- [24] D.G. MacArthur, S. Balasubramanian, A. Frankish, N. Huang, J. Morris, K. Walter, L. Jostins, L. Habegger, J.K. Pickrell, S.B. Montgomery, et al. Un estudio sistemático de variantes de pérdida de función en genes codificadores de proteínas humanas. *Ciencia*, 335 (6070) :823—828, 2012.
- [25] B.P. McEvoy y P.M. Visscher. Genética de la estatura humana. *Economía y biología humana*, 7 (3) :294 — 306, 2009.
- [26] N.E. Morton. Pruebas secuenciales para la detección de ligamiento. *The American Journal of Human Genetics*, 7 (3) :277—318, 1955.
- [27] N. Patterson, A. Price, y D. Reich. Estructura poblacional y análisis propio. *PLoS Genetics*, 2:e190, 2006.
- [28] A. Piccolboni y D. Gusfield. Sobre la complejidad de los problemas computacionales fundamentales en el análisis pedigrí. *Revista de Biología Computacional*, 10:763 —773, octubre de 2003.
- [29] A. Price et al. El análisis de componentes principales corrige la estratificación en estudios de asociación de todo el genoma. *Nature Genetics*, 38:904 —909, 2006.
- [30] J. Pritchard, M. Stephens, N. Rosenberg y P. Donnelly. Mapeo de asociación en poblaciones estructuradas. *American Journal of Human Genetics*, 67:170 —181, 2000.
- [31] M.A. Rivas, M. Beaudoin, A. Gardet, C. Stevens, Y. Sharma, C.K. Zhang, G. Boucher, S. Ripke, D. Ellinghaus, N. Burtt, et al. La resecuenciación profunda de loci gwas identifica variantes raras independientes asociadas con enfermedad inflamatoria intestinal. *Genética de la naturaleza*, 2011.
- [32] Kaitlin E Samocha, Elise B Robinson, Stephan J Sanders, Christine Stevens, Aniko Sabo, Lauren M McGrath, Jack A Kosmicki, Karola Rehnström, Swapan Mallick, Andrew Kirby, Dennis P Wall, Daniel G MacArthur, Stacey B Gabriel, Mark Depristo, Shaun M Purcell, Aarno Palotie, Eric Boocha Erwinkle, José D Buxbaum, Edwin H Cook, Richard A Gibbs, Gerard D Schellenberg, James S Sutcliffe, Bernie Devlin, Kathryn Roeder, Benjamin M Neale y Mark J Daly. Un marco para la interpretación de la mutación de novo en la enfermedad humana. 46 (9) :944—950.
- [33] Evan A. Stein, Scott Mellis, George D. Yancopoulos, Neil Stahl, Douglas Logan, William B. Smith, Eleanor Lisboa, María Gutiérrez, Cheryle Webb, Richard Wu, Yunling Du, Therese Kranz, Evelyn Gasparino y Gary D. Swergold. EECT de un anticuerpo monoclonal contra colesterol LDL. 366 (12) :1108—1118.
- [34] T. Strachan y A.P. Leer. Genética Molecular Humana. Wiley-Liss, Nueva York, 2 edición, 1999.
- [35] S. Yang, Y. Xiao, D. Kang, J. Liu, Y. Li, E. A. B. Undheim, J. K. Clint, M. Rong, R. Lai, y G. F. King. Descubrimiento de un inhibidor selectivo de NaV1.7 a partir del veneno de ciempiés con una ecacia analgésica superior a la morfina en modelos de dolor en roedores. 110 (43) :17534—17539.

---

30.1: [Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 30.1: [Bibliography](#) has no license indicated.

## 30.2: Introducción

Mark J. Daly, Ph.D., es profesor asociado en el Massachusetts General Hospital/Harvard Medical School y miembro asociado del Broad Institute. Esta conferencia explica cómo los métodos estadísticos y computacionales pueden ayudar a los investigadores a comprender, diagnosticar y tratar enfermedades. El mapeo de asociación es el proceso de identificación de la variación genética que puede explicar la variación fenotípica, lo cual es particularmente importante para comprender los fenotipos de enfermedades (p. ej., susceptibilidad). Históricamente, el método de elección para resolver este problema fue el análisis de vinculación. Sin embargo, los avances en la tecnología genómica han permitido un método más poderoso llamado asociación genómica.

Los avances más recientes en tecnología y datos genómicos han permitido nuevos análisis integradores que pueden hacer predicciones poderosas sobre enfermedades. Cualquier discusión sobre la base de la enfermedad debe considerar los efectos genéticos y ambientales. Sin embargo, se sabe que muchos rasgos, por ejemplo los de la Figura 30.1, tienen componentes genéticos significativos. Formalmente, la heredabilidad de un fenotipo es la proporción de variación en ese fenotipo que puede explicarse por la variación genética. Los rasgos en la Figura 30.1 son todos al menos 50% heredables. Estimar con precisión la heredabilidad implica análisis estadísticos en muestras con niveles muy variados de variación genética compartida (por ejemplo, gemelos, hermanos, parientes y no relacionados). Los estudios sobre la heredabilidad de la diabetes Tipo 2, por ejemplo, han demostrado que dado que tienes diabetes, el riesgo para la persona sentada a tu lado (una persona no emparentada) aumenta en 5— 10%; el riesgo para un hermano aumenta en 30%; y el riesgo para un gemelo idéntico aumenta en 85% — 90%.

---

30.2: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 30.2: Introduction has no license indicated.

### 30.3: Objetivos de investigar las bases genéticas de la enfermedad

Habiendo establecido que hay un componente genético en los rasgos de la enfermedad, ¿cómo puede esta investigación ayudar a enfrentar desafíos médicos sobresalientes? Hay dos formas principales:

#### Medicina genómica personalizada

Las variantes se pueden usar en exámenes genéticos para evaluar el mayor riesgo de padecer el rasgo de la enfermedad y proporcionar conocimientos médicos individualizados. Un gran número de empresas ahora están brindando servicios genómicos personalizados a través de exámenes de detección del riesgo de recurrencia del cáncer, trastornos genéticos (incluido el cribado prenatal) y enfermedades comunes. La medicina genómica individualizada puede ayudar a identificar la probabilidad de beneficiarse de intervenciones terapéuticas específicas, o puede predecir respuestas adversas a medicamentos.

#### Informar el desarrollo terapéutico

La identificación de variantes genéticas que explican el rasgo de la enfermedad contribuye a nuestra capacidad de entender el mecanismo (las vías bioquímicas, etc.) por el que se manifiesta la enfermedad. Esto nos permite diseñar medicamentos que son más efectivos para apuntar a las vías causales en la enfermedad. Esto es de particular interés

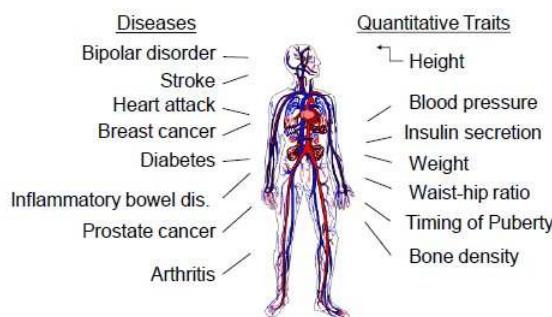


Figura 30.1: Ejemplos de enfermedades y rasgos cuantitativos que tienen componentes genéticos

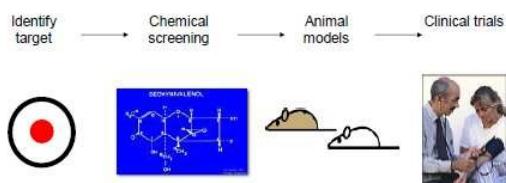


Figura 30.2: El proceso de desarrollo de fármacos

porque nuestro proceso actual de desarrollo de fármacos dificulta el desarrollo de medicamentos para ciertos trastornos. Por ejemplo, en los últimos 50 años, no se han desarrollado compuestos verdaderamente novedosos para tratar diversos trastornos psiquiátricos como la esquizofrenia. La identificación de genes genéticamente asociados puede ayudar a identificar dianas para iniciar el desarrollo de fármacos.

La Figura 30.2 representa el ciclo de desarrollo de fármacos. El proceso de desarrollo de fármacos comienza con la hipótesis de un posible objetivo de interés que podría estar relacionado con una enfermedad. Después de evaluaciones bioquímicas y desarrollo de fármacos, la diana se prueba en organismos modelo. Si el fármaco es efectivo en organismos modelo, se prueba en humanos a través de ensayos clínicos. Sin embargo, la gran mayoría de los fármacos que lo hacen pasar por este proceso terminan siendo ineficaces en el tratamiento de la enfermedad para la que fueron diseñados originalmente. Este resultado es principalmente consecuencia de una selección defectuosa de la diana como base de la enfermedad en cuestión. Las estatinas son un ejemplo destacado de fármacos altamente efectivos desarrollados después de trabajar en la comprensión de la base genética del rasgo de enfermedad al que se dirigen. El Dr. Michael Brown y el Dr. Joseph Goldstein ganaron el Premio Nobel de Fisiología o Medicina en 1985 por su trabajo sobre la regulación del metabolismo del colesterol LDL [5]. Fueron capaces de aislar la causa de hipercolesterolemia familiar extrema (FH), un trastorno mendeliano, a mutaciones de un solo gen que codifica un receptor de LDL. Además, fueron capaces de identificar la vía bioquímica que se vio afectada por la mutación para crear la condición de enfermedad.

Las estatinas se dirigen a esa vía, haciéndolas útiles no solo para los individuos que padecen FH, sino también como un tratamiento efectivo para el colesterol alto LDL en la población general.

---

30.3: [Objetivos de investigar las bases genéticas de la enfermedad](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [30.3: Goals of investigating the genetic basis of disease](#) has no license indicated.

## 30.4: Rasgos mendelianos

### Mendel

Gregor Mendel identificó la primera evidencia de herencia en 1865 usando hibridación de plantas. Reconoció unidades discretas de herencia relacionadas con rasgos fenotípicos, y señaló que la variación en estas unidades, y por lo tanto variaciones en los fenotipos, fue transmisible a través de generaciones. Sin embargo, Mendel ignoró una discrepancia en sus datos: algunos pares de fenotipos no se transmitieron de forma independiente. Esto no se entendió hasta 1913, cuando el mapeo de enlaces mostró que los genes del mismo cromosoma se pasan en tandem a menos que ocurra un evento de cruce meiótico. Además, la distancia entre genes de interés describe la probabilidad de que ocurra un evento de recombinación entre los dos loci y, por lo tanto, la probabilidad de que los dos genes se hereden juntos (**enlace**).

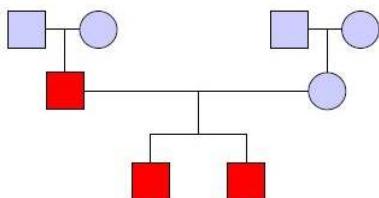


Figura 30.3: Un pedigrí que muestra la herencia de algún rasgo

### Análisis de Vinculación

Históricamente, los investigadores han utilizado la idea de **vinculación a través del análisis de ligamiento** para determinar variantes genéticas que explican la variación fenotípica. El objetivo es determinar qué variantes contribuyen al patrón observado de variación fenotípica en un pedigrí. La Figura 30.3 muestra un pedigrí ejemplar en el que los cuadrados son individuos masculinos, los círculos son individuos femeninos, las parejas y las crías están conectadas, y los individuos en rojo tienen el rasgo de interés.

El análisis de ligamiento se basa en la percepción biológica de que las variantes genéticas no se heredan de forma independiente (como lo propone Mendel). En cambio, la recombinación meiótica ocurre un número limitado de veces (aproximadamente una vez por cromosoma), por lo que muchas variantes se *cosegregan* (se heredan juntas). Este fenómeno se conoce como *desequilibrio de ligamiento* (LD).

A medida que aumenta la distancia entre dos variantes, aumenta la probabilidad de que se produzca una recombinación entre ellas. Thomas Hunt Morgan y Alfred Sturtevant desarrollaron esta idea para producir **mapas de ligamiento** que no solo pudieran determinar el orden de los genes en un cromosoma, sino también sus distancias relativas entre sí. El Morgan es la unidad de distancia genética que propusieron; los loci separados por 1 centimórgano (cM) tienen 1 de cada 100 probabilidades de ser separados por una recombinación. Los loci no enlazados tienen 50% de probabilidad de ser separados por una recombinación (se separan si ocurre un número impar de recombinaciones entre ellos). Como generalmente no conocemos a priori qué variantes son causales, en su lugar utilizamos marcadores genéticos que capturan otras variantes debido a LD. En 1980, David Botstein propuso el uso de polimorfismos de un solo nucleótido (SNP), o mutaciones de una sola base, como marcadores genéticos en humanos [4]. Si un marcador particular está en LD con la variante causal real, entonces observaremos su patrón de herencia contribuyendo a la variación fenotípica en el pedigrí y puede estrechar nuestra búsqueda.

Los fundamentos estadísticos del análisis de vinculación se desarrollaron en la primera parte del siglo XX. Ronald Fisher propuso un modelo genético que pudiera conciliar la herencia mendeliana con fenotipos continuos como la altura [10]. Newton Morton desarrolló una prueba estadística llamada puntaje LOD (logaritmo de probabilidades) para probar la hipótesis de que los datos observados resultan de la vinculación [26]. La hipótesis nula de la prueba es que la fracción de recombinación (la probabilidad de que se produzca una recombinación entre dos marcadores adyacentes)  $\theta = 1/2$  (sin vinculación) mientras que la hipótesis alternativa es que es una cantidad menor. La puntuación LOD es esencialmente una razón logarítmica de verosimilitud que captura esta prueba estadística:

$$\text{LOD} = \frac{\log(\text{likelihood of disease given linkage})}{\log(\text{likelihood of disease given no linkage})}$$

Los algoritmos para el análisis de ligamiento se desarrollaron en la última parte del siglo XX. Existen dos clases principales de análisis de ligamiento: paramétrico y *no paramétrico* [34]. El análisis de vinculación paramétrica se basa en un modelo (parámetros) de la herencia, frecuencias y penetrancia de una variante particular. Que  $F$  sea el conjunto de fundadores (ancestros originales) en el pedigree, que  $g_i$  sea el genotipo del individuo  $i$ ,  $\Phi_i$  déjese

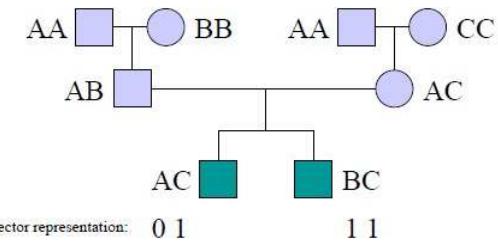


Figura 30.4: Representación de un patrón particular de herencia como vector de herencia

el fenotipo del individuo  $i$ , y dejar que  $f(i)$  y  $m(i)$  sean el padre y la madre del individuo  $i$ . Entonces, la probabilidad de observar los genotipos y fenotipos en el pedigree es:

$$L = \sum_{g_1} \dots \sum_{g_n} \prod_i \Pr(\Phi_i | g_i) \prod_{f \in F} \Pr(g_f) \prod_{i \notin F} \Pr(g_i | g_{f(i)}, g_{m(i)})$$

El tiempo requerido para calcular esta probabilidad es exponencial tanto en el número de marcadores considerados como en el número de individuos en el pedigree. Sin embargo, Elston y Stewart dieron un algoritmo para calcularlo de manera más eficiente asumiendo que no hay endogamia en el pedigree [8]. Su visión fue que condicionada a los genotipos parentales, las crías son condicionalmente independientes. En otras palabras, podemos tratar el pedigree como una red bayesiana para calcular de manera más eficiente la distribución de probabilidad conjunta. Su algoritmo escala linealmente en el tamaño del pedigree, pero exponencialmente en el número de marcadores.

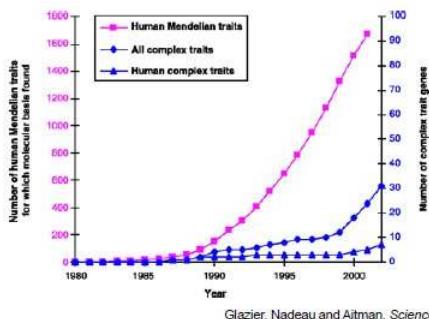
Hay varios problemas con el análisis de vinculación paramétrica. Primero, los marcadores individuales pueden no ser informativos (dar información inequívoca sobre la herencia). Por ejemplo, los padres homocigotos o el error de genotipado podrían llevar a marcadores poco informativos. Para sortear esto, podríamos escribir más marcadores, pero el algoritmo no escala bien con el número de marcadores. En segundo lugar, es sencillo idear parámetros de modelo para un trastorno mendeliano. Sin embargo, hacer lo mismo con los trastornos no mendelianos no es trivial. Finalmente, las estimaciones de LD entre marcadores no se soportan inherentemente.

El análisis de ligamiento no paramétrico no requiere un modelo genético. En cambio, primero inferimos el patrón de herencia dados los genotipos y el pedigree. Luego determinamos si el patrón de herencia puede explicar la variación fenotípica en el pedigree.

Lander y Green formularon un HMM para realizar la primera parte de este análisis [20]. Los estados de este HMM son vectores de herencia que especifican el resultado de cada meiosis en el pedigree. Cada individuo está representado por 2 bits (uno para cada parente). El valor de cada bit es 0 o 1 dependiendo de cuál de los alelos grand-parental se hereda. La Figura 30.4 muestra un ejemplo de la representación de dos individuos en un vector de herencia.

Cada paso del HMM corresponde a un marcador; una transición en el HMM corresponde a algunos bits del vector de herencia cambiando. Esto significa que el alelo heredado de alguna meiosis cambió, es decir, que se produjo una recombinación. Las probabilidades de transición en el HMM son entonces una función de la fracción de recombinación entre marcadores adyacentes y la distancia de Hamming (el número de bits que difieren, o el número de recombinaciones) entre los dos estados. Podemos usar el algoritmo adelante/atrás para calcular las probabilidades posteriores en este HMM e inferir la probabilidad de cada patrón de herencia para cada marcador.

Este algoritmo escala linealmente en el número de marcadores, pero exponencialmente en el tamaño del pedigree. El número de estados en el HMM es exponencial en la longitud del vector de herencia, que es lineal en el tamaño del pedigree. En general, se sabe que el problema es NP-duro (a lo mejor de nuestro conocimiento, no podemos hacerlo mejor que un algoritmo que escala exponencialmente en la entrada) [28]. Sin embargo, el problema es importante no



Glazier, Nadeau and Altman, Science 2002

Asociación Americana para el Avance de la Ciencia. Todos los derechos reservados. Este contenido está excluido de nuestra licencia Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Fuente: Cristalero, Anne M., et al. "Encontrar genes que subyacen a rasgos complejos". *Ciencia* 298, núm. 5602 (2002): 2345-9.

Figura 30.5: Descubrimiento de genes para diferentes tipos de enfermedad versus tiempo

sólo en este contexto, pero también en los contextos de *inferencia o fase de haplotipos* (asignación de alelos a cromosomas homólogos) e *imputación de genotipos* (inferir genotipos faltantes basados en genotipos conocidos). Ha habido muchas optimizaciones para que este análisis sea más manejable en la práctica [1, 11, 12, 15—18, 21, 23].

El análisis de ligamiento identifica una amplia región genómica que se correlaciona con el rasgo de interés. Para estrechar la región, podemos usar mapas genéticos de resolución fina de puntos de interrupción de recombinación. Luego podemos identificar el gen afectado y la mutación causal secuenciando la región y probando la función alterada.

30.4: Rasgos mendelianos is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

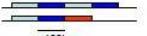
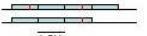
- 30.4: Mendelian Traits has no license indicated.

## 30.5: Rasgos Complejos

El análisis de ligamiento ha demostrado ser altamente efectivo en el estudio de las bases genéticas de las enfermedades mendelianas (un solo gen). En las últimas tres décadas, miles de genes han sido identificados como contribuyentes a las enfermedades mendelianas. Hemos identificado las bases genéticas de enfermedades como anemia falciforme, fibrosis quística, distrofia muscular y formas graves de enfermedades comunes como diabetes e hipertensión arterial. Para estas enfermedades, las mutaciones son graves y obvias; el ambiente, el comportamiento y el azar tienen poco efecto. La figura 30.5 muestra esta explosión en asociaciones publicadas.

Sin embargo, la mayoría de las enfermedades (y muchos otros rasgos de interés) no son mendelianos. Estos rasgos complejos surgen de las interacciones de muchos genes y posiblemente del entorno y el comportamiento. Un rasgo complejo canónico es la altura humana: es altamente heredable, pero los factores ambientales pueden afectarla. Recientemente, los investigadores han identificado cientos de variantes que están asociadas con la altura [2, 25].

El análisis de vinculación no es un enfoque viable para encontrar estas variantes. El primer mapeo de rasgos complejos ocurrió en 1920 por Altenburg y Muller e involucró la base genética del ala truncada en *D. Melanogaster*. La poligenicidad, o distribución de un rasgo complejo a través de un gran número de genes, proporciona un desafío fundamental para determinar qué genes están asociados con un fenotipo. En rasgos complejos, en lugar de un gen que determina una enfermedad o rasgo (como en la herencia mendeliana), muchos genes ejercen cada uno una pequeña influencia. El efecto de todos estos genes, así como las influencias ambientales, se combinan para determinar un resultado individual. Además, las enfermedades más comunes funcionan de esta manera. Esto se debe a que la selección contra cada diferencia genotípica individual es muy pequeña, porque no hay una diferencia que sea causal de la enfermedad. De esta manera, rasgos complejos “sobreviven” a la evolución, porque no son objetivos de selección.

|  |  |
|--|--|
| • Single nucleotide polymorphisms (SNPs)                   | TGCATT <b>I</b> GCGTAGGC   |
| – 1 every few hundred bp, mutation rate* $\approx 10^{-9}$ | TGCATT <b>C</b> CGTAGGC  |
| • Short indels (=insertion/deletion)                       | TGCATT---TAGGC   |
| – 1 every few kb, mutation rate v. variable                | TGCATT <b>CCG</b> TAGGC  |
| • Microsatellite (STR) repeat number                       | TGCTCAT <b>T</b> CATCATCAGC  |
| – 1 every few kb, mutation rate $\leq 10^{-3}$             | TGCTCAT <b>CA</b> -----GC  |
| • Minisatellites   |  |
| – 1 every few kb, mutation rate $\leq 10^{-1}$             | $\leq 100\text{bp}$  |
| • Repeated genes   |  |
| – rRNA, histones   | 1-5kb  |
| • Large deletions, duplications, inversions                |  |
| – Rare, e.g. Y chromosome                                  |  |

\* rates per generation

fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 30.6: Diferentes tipos de variación genética

30.5: Rasgos Complejos is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 30.5: Complex Traits has no license indicated.

## 30.6: Estudios de Asociación en todo el genoma

En la década de 1990, los investigadores propusieron una metodología llamada **asociación genómica** para correlacionar sistemáticamente marcadores con rasgos. Estos estudios muestran grandes grupos de casos y controles, miden sus genotipos en el orden de un millón de marcadores e intentan correlacionar la variación (SNP, CNV, indeles) en sus genotipos con su variación en el fenotipo, rastreando la enfermedad a través de la población, en lugar de pedigree.

### Eventos Habilitando Estudios de Asociación en todo el genoma

Los estudios de asociación de todo el genoma (GWASs) son posibles debido a tres avances.

En primer lugar, los avances en nuestra comprensión del genoma y la creación de recursos genómicos nos han permitido comprender y catalogar mejor la variación del genoma. A partir de estos datos, nos hemos dado cuenta de la visión biológica clave de que los humanos son una de las especies con menor diversidad genética. Del orden de decenas de millones de SNP se comparten entre diferentes subpoblaciones humanas. Para cualquier región particular del genoma, observamos solo un número limitado de **haplotipos** (combinaciones de alelos que se heredan juntas). Esto se debe a que como especie, somos relativamente nuevos, y las mutaciones no han alcanzado nuestro rápido crecimiento. Debido a esta alta redundancia, solo necesitamos medir una fracción de todas las variantes en el genoma humano para capturarlas todas con LD. Luego podemos adaptar los algoritmos para inferir patrones de herencia en el análisis de ligamientos para imputar genotipos para los marcadores que no tienen genotipo. Además, los recursos genómicos nos permiten elegir cuidadosamente marcadores para medir y hacer predicciones basadas en marcadores que muestran asociación estadísticamente significativa. Ahora tenemos la secuencia de referencia del genoma humano (permitiendo alineaciones, genotipos y llamadas SNP) y HapMap, un catálogo completo de SNP en humanos. También tenemos anotaciones genómicas de genes y elementos reguladores.

En segundo lugar, los avances en la tecnología de genotipado como los microarrays y la secuenciación de alto rendimiento nos han dado la oportunidad de comparar los genomas de los afectados con diversos fenotipos con los controles. También son los más fáciles y económicos de medir utilizando estas tecnologías. Aunque existen muchos tipos de variación en el genoma humano (la Figura 30.6 muestra algunos ejemplos), los SNP son la gran mayoría. Adicionalmente, para dar cuenta de los otros tipos de variantes, recientemente se han desarrollado microarrays de ADN para detectar la variación del número de copias además de los SNP, después de lo cual podemos imputar los datos no observados.

El tercer avance es una nueva expectativa de colaboración entre investigadores. Los GWAs se basan en tamaños de muestra grandes para aumentar la potencia (probabilidad de un verdadero positivo) de las pruebas estadísticas. La explosión en el número de GWASs publicados ha permitido un nuevo tipo de **metaanálisis** que combina los resultados de varios GWAs para el mismo fenotipo para hacer asociaciones más poderosas. El metaanálisis da cuenta de diversos sesgos técnicos y genéticos poblacionales en estudios individuales. Se espera que los investigadores que realizan GWASs colaboren con otros que han realizado GWAs en el mismo rasgo para mostrar la replicabilidad de los resultados. Al juntar los datos, también tenemos más confianza en las asociaciones reportadas, y los genes que se descubren pueden conducir al reconocimiento de vías y procesos clave.

#### ¿Sabías?

Modificado del Wellcome Trust Sanger Institute: La enfermedad de Crohn y la Colitis Ulcerosa han sido focos para la genética de enfermedades complejas, y los esfuerzos masivos de colaboración del Consorcio Internacional de Genética de Enfermedades Inflamatorias Intestinales (IIBDGC) fortalecen el éxito de la investigación. Con aproximadamente 40,000 muestras de ADN de pacientes con EII y 20,000 controles sanos, el IIBDGC ha descubierto 99 loci definidos de IBD. En total, los 71 loci de la enfermedad de Crohn y 47 loci de CU representan el 23% y 16% de la heredabilidad de la enfermedad respectivamente. Los conocimientos clave sobre la biología de la enfermedad ya han resultado del descubrimiento de genes (por ejemplo, la autofagia en la enfermedad de Crohn, la función de barrera defectuosa en la UC y la señalización de IL23 en la EII y la enfermedad inmunomediada). Se anticipa que de las muchas dianas farmacológicas novedosas identificadas por el descubrimiento de genes, algunas darán como resultado, en última instancia, una terapéutica mejorada para estas condiciones devastadoras. La mejora del diagnóstico, el pronóstico y la terapéutica son objetivos, con miras a una **terapia personalizada** (la práctica de usar el perfil genético de un individuo como guía para las decisiones de tratamiento) en el futuro.

## Controles de Calidad

El principal problema en la realización de GWASs es eliminar los factores de confusión, pero las mejores prácticas se pueden usar para respaldar datos de calidad.

En primer lugar, existe el error de genotipado, que es lo suficientemente común como para requerir un tratamiento especial independientemente de la tecnología que se utilice. Este es un control técnico de calidad, y para dar cuenta de dichos errores, utilizamos umbrales en métricas como la frecuencia de alelos menores y la desviación del **equilibrio Hardy-Weinberg** y desecharmos SNP que no cumplen con los criterios.

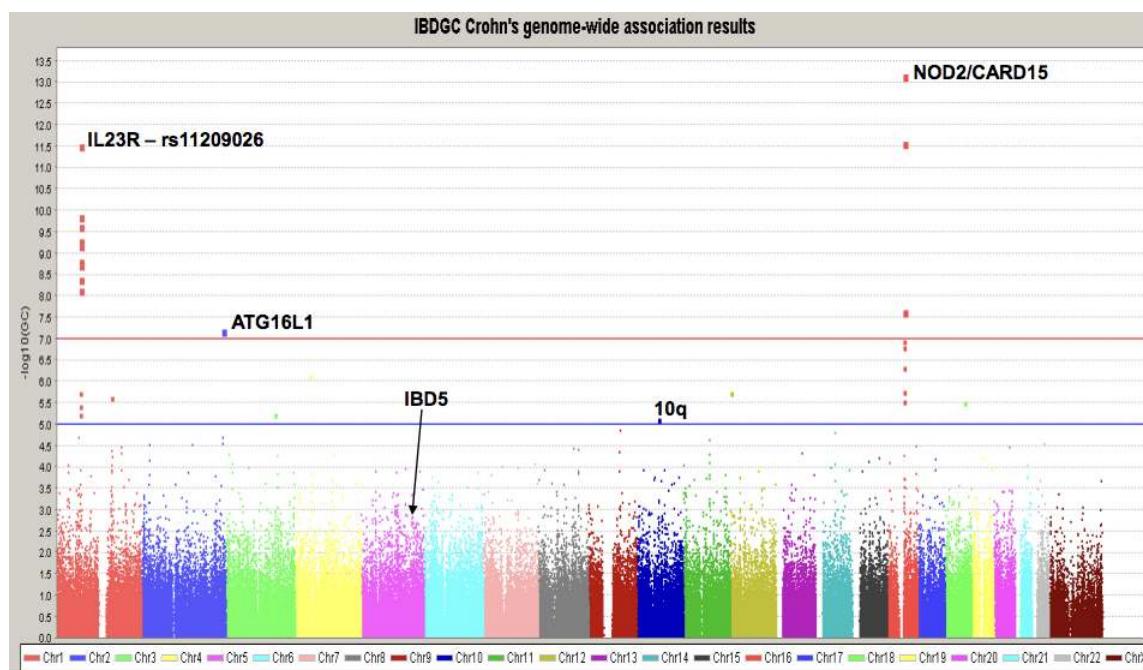
En segundo lugar, las diferencias genéticas sistemáticas entre subpoblaciones humanas requieren un control de calidad genética. Existen varios métodos para dar cuenta de esta **subestructura poblacional**, como el control genómico [7], pruebas de inconsistencias mendelianas, asociación estructurada [30] y análisis de componentes principales [27, 29].

Tercero, covariables como los efectos ambientales y conductuales o el género pueden sesgar los datos. Podemos contabilizarlos incluyéndolos en nuestro modelo estadístico.

## Pruebas para la Asociación

Después de realizar los controles de calidad, el análisis estadístico involucrado en GWAS es bastante sencillo, siendo las pruebas más simples la **regresión de un solo marcador** o una **prueba de chi-cuadrado**. De hecho, los resultados de asociación que requieren estadísticas arcanas/modelos multimarcadores complejos suelen ser menos confiables.

Primero, asumimos que el efecto de cada SNP es independiente y aditivo para hacer que el análisis sea manejable. Para cada SNP, realizamos una prueba de hipótesis cuya hipótesis nula es que la variación observada en el genotipo en ese SNP entre los sujetos no se correlaciona con la variación observada en el fenotipo entre los sujetos. Debido a que realizamos una prueba para cada SNP, necesitamos lidiar con el **problema de múltiples pruebas**. Cada prueba tiene alguna probabilidad de dar un resultado falso positivo, y a medida que aumentamos el número de pruebas,



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative

Licencia Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 30.7: Umbrales de significancia de GWAS en la línea azul y líneas rojas para un estudio del IBDGC sobre la enfermedad de Crohn. La línea azul representa un valor p de 5e-8 y la línea roja representa aproximadamente 7.2e-8.

la probabilidad de obtener un falso positivo en cualquiera de ellos aumenta. Esencialmente, con ligamiento,  $p = 0.001 (.05/ 50$  brazos cromosómicos) se consideraría potencialmente significativo, pero GWAS implica realizar pruebas O ( $10e6$ ) que son en gran

medida independientes. Cada estudio tendría cientos de  $p < 0.001$  puramente por casualidad estadística, sin relación real con la enfermedad. Existen varios métodos para dar cuenta de múltiples pruebas como la corrección de Bonferroni y medidas como la tasa de falsos descubrimientos [3] y la tasa de descubrimiento irreproducible [22]. Por lo general, la **significancia de todo el genoma** se establece en  $p = 5*10^{-8}$  (= .05/1 millón de pruebas), propuestas por primera vez por Risch y Merikangas (1996) [1]. En 2008, tres grupos [1] publicaron estimaciones derivadas empíricamente basadas en mapas densos de todo el genoma de ADN común y estimaron que los números de mapas densos apropiados estaban en el rango de 2.5 a 7.2e-8. Estos se pueden visualizar en la Figura 30.7. Debido a estos diferentes umbrales, es importante observar múltiples estudios para validar asociaciones, ya que incluso con un estricto control de calidad puede haber artefactos que pueden afectar a uno de cada mil o diez mil SNP y evadirse al aviso. Adicionalmente, la significación estricta de todo el genoma generalmente no se excede dramáticamente, si se alcanza, en un solo estudio.

Además de reportar SNP que muestran las asociaciones más fuertes, normalmente también usamos *parcelas de Manhattan* para mostrar dónde se encuentran estos SNP en el genoma y las parcelas de *cuantil-cuantil (Q-Q)* para detectar sesgos que no han sido debidamente contabilizados. Una gráfica de Manhattan es una gráfica de dispersión de valores  $p$  transformados logarítmicamente contra la posición genómica (concatenando los cromosomas). En la Figura 30.8A, los puntos en rojo son aquellos que cumplen con el umbral de significancia. Se etiquetan con genes candidatos que están cerca. Una gráfica Q-Q es una gráfica de dispersión de los valores  $p$  observados transformados logarítmicamente frente a los valores  $p$  esperados transformados logarítmicamente. Utilizamos cuantiles uniformes como los valores  $p$  esperados: suponiendo que no hay asociación, esperamos que los valores  $p$  se distribuyan uniformemente. La desviación de la diagonal sugiere que los valores de  $p$  son más significativos de lo esperado. Sin embargo, la desviación temprana y consistente de la diagonal sugiere que demasiados valores  $p$  son demasiado significativos, es decir, hay algún sesgo que está confundiendo la prueba. En la Figura 30.8B, la gráfica muestra el estadístico de prueba observado contra el estadístico de prueba esperado (que es equivalente). Considerando todos los marcadores incluye el **Complejo Mayor de Histocompatibilidad (MHC)**, que es la región asociada con la respuesta inmune. Esta región tiene una estructura LD única que confunde el análisis estadístico, como se desprende de la desviación de los puntos negros de la diagonal (el área gris). Tirar el MHC elimina gran parte de este sesgo de los resultados (los puntos azules).

GWAS identifica marcadores que se correlacionan con el rasgo de interés. Sin embargo, cada marcador captura un vecindario de SNP con el que se encuentra en LD, dificultando el problema de identificar la variante causal. Por lo general, el gen candidato para un marcador es el que está más cerca de él. A partir de aquí, tenemos que hacer más estudios para identificar la relevancia de las variantes que identificamos. Sin embargo, esto sigue siendo un problema desafiante por algunas razones:

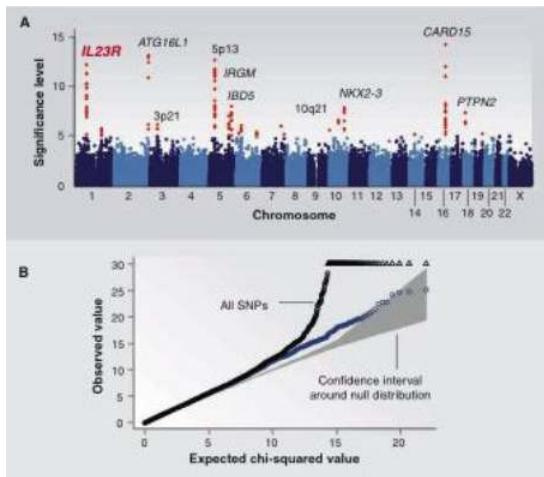


Figura 30.8: (A) parcela de Manhattan y (B) gráfica Q-Q para GWAS de la enfermedad de Crohn

- Las regiones de interés identificadas por asociación a menudo implican múltiples genes
- Algunas de estas asociaciones no están cerca de ningún segmento codificante de proteínas y no tienen un alelo obviamente funcional como su origen
- Vincular estas regiones con las vías biológicas subyacentes es difícil

## Interpretación: ¿Cómo puede el GWAS informar la biología de la enfermedad?

Nuestro objetivo principal es utilizar estas asociaciones encontradas para comprender la biología de la enfermedad de una manera accionable, ya que esto ayudará a guiar las terapias para tratar estas enfermedades. La mayoría de las asociaciones no identifican genes específicos y mutaciones causales, sino que son solo punteros a regiones pequeñas con influencias causales en la enfermedad. Para poder desarrollar y actuar sobre una hipótesis terapéutica, debemos ir mucho más allá, y responder a estas preguntas:

- ¿Qué gen está conectado con la enfermedad?
- ¿Qué proceso biológico está implicado con ello?
- ¿Cuál es el contexto celular en el que ese proceso actúa y es relevante para la enfermedad?
- ¿Cuáles son los alelos funcionales específicos que perturban el proceso y promueven o protegen de enfermedades?

Esto se puede abordar de una de dos maneras: el enfoque de *abajo hacia arriba*, o el enfoque de *arriba hacia abajo*.

### De abajo hacia arriba

El enfoque de abajo hacia arriba se utiliza para investigar un gen particular que tiene una asociación conocida con una enfermedad, e investigar su importancia biológica dentro de una célula. Kuballa et al. [19] pudieron utilizar este enfoque de abajo hacia arriba para aprender que una variante de riesgo particular asociada a la enfermedad de Crohn conduce a un deterioro de

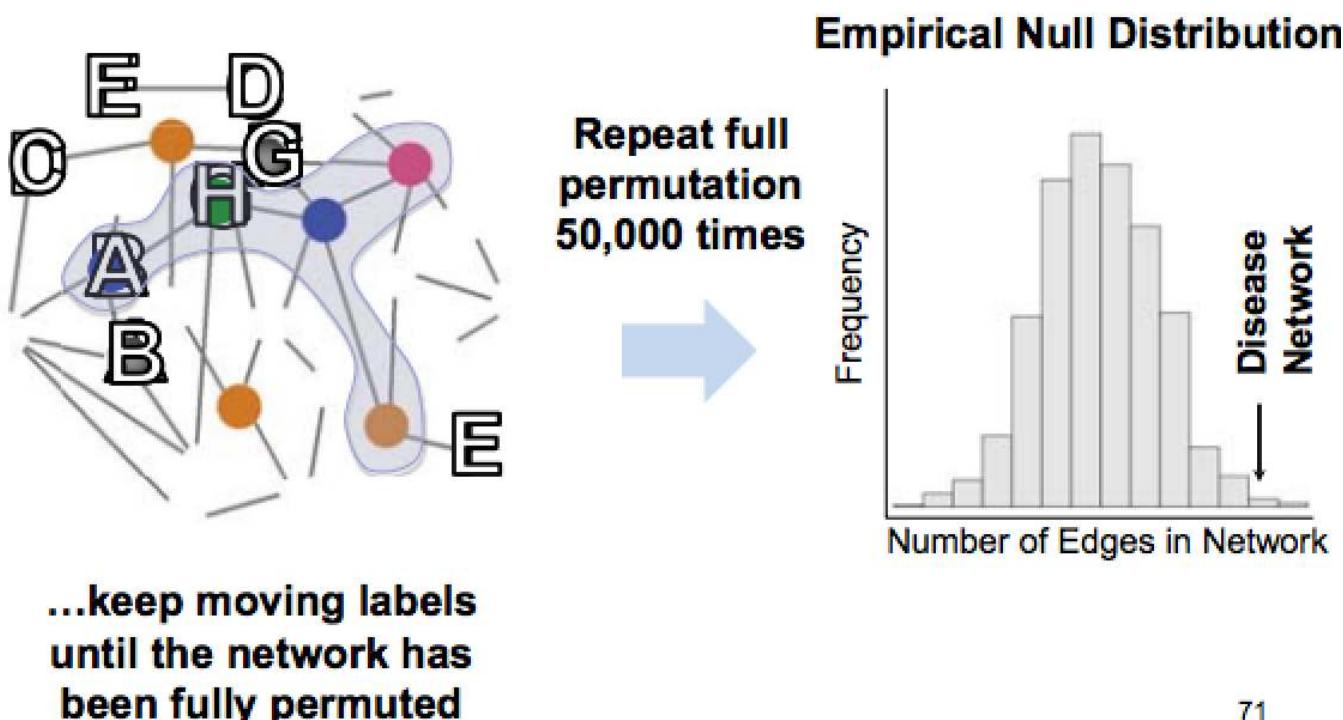


Figura 30.9: Evaluación de la importancia de la red de enfermedades

autofagia de ciertos patógenos. Además, los autores pudieron crear un modelo de ratón de la misma variante de riesgo que se encuentra en humanos. Identificar las implicaciones biológicas de las variantes de riesgo a nivel celular y crear estos modelos es invaluable, ya que los modelos pueden usarse directamente para probar nuevos compuestos potenciales de tratamiento.

### De arriba hacia abajo

En contraste, el enfoque de arriba hacia abajo implica observar todas las asociaciones conocidas, utilizar el conjunto completo de resultados de GWAS y tratar de vincularlos a procesos/vías biológicas compartidas implicadas en la patogénesis de la enfermedad. Este enfoque se basa en la idea de que muchos de los genes asociados a una enfermedad comparten vías biológicas relevantes. Esto

se hace comúnmente tomando redes existentes como redes de interacción proteína-proteína, y superponiendo los genes asociados sobre ellas. Sin embargo, estas redes de enfermedades resultantes pueden no ser significativas debido al sesgo tanto en el descubrimiento de asociaciones como por el sesgo experimental de los datos con los que se están integrando las asociaciones. Esta significación se puede estimar permutando las etiquetas para los nodos en la red muchas veces, y luego calculando cuán raro es el nivel de conectividad para la red de enfermedades dada. Este proceso se ilustra en la Figura 30.9. Como los genes conectados en la red deben ser coexpresados, se ha demostrado que estas redes de enfermedades pueden validarse aún más a partir de perfiles de expresión génica [14].

### Comparación con Análisis de Vinculación

Es importante tener en cuenta que GWAS captura más variantes que el análisis de ligamiento. El análisis de ligamiento identifica variantes raras que tienen efectos negativos, y se utilizan estudios de ligamiento cuando se dispone de pedigríes de individuos relacionados con información fenotípica. Pueden identificar alelos raros que están presentes en un número menor de familias, generalmente debido a mutaciones fundadoras y se han utilizado para identificar mutaciones como BRCA1, asociadas con cáncer de mama. Alternativamente, se utilizan estudios de asociación para este propósito y también para encontrar cambios genéticos más comunes que confieren menor influencia en la susceptibilidad, como variantes raras que tienen efectos protectores. El análisis de ligamiento no puede identificar estas variantes porque están anticorrelacionadas con el estado de la enfermedad. Además, el análisis de ligamiento se basa en la suposición de que una sola variante explica la enfermedad, una suposición que no se sostiene para rasgos complejos como la enfermedad. En cambio, necesitamos considerar muchos marcadores para explicar la base genética de estos rasgos.

Si bien la medicina genómica promete nuevos descubrimientos en mecanismos de enfermedades, genes diana, terapéutica y medicina personalizada, quedan varios desafíos, incluyendo que 90+% de los éxitos no son codificantes.

Para solucionar esto, el genoma no codificante ha sido anotado a través de codificación/hoja de ruta y los potenciadores se han vinculado a reguladores y genes diana. Una vez que cada locus GWAS se expande usando el desequilibrio de ligamiento SNP (LD), se puede usar para reconocer tipos de células relevantes, factores de transcripción de controladores y genes diana. Esto lleva a una vinculación de rasgos con sus tipos relevantes de células y tejidos.

### Conclusiones

Hemos aprendido varias lecciones de GWAS. Primero, menos de un tercio de las asociaciones reportadas son variantes codificantes o obviamente funcionales. En segundo lugar, solo algunas fracciones de variantes no codificantes asociadas están significativamente asociadas al nivel de expresión de un gen cercano. En tercer lugar, muchos están asociados a regiones sin gen codificador cercano. Finalmente, la mayoría de las variantes reportadas están asociadas a múltiples enfermedades autoinmunes o inflamatorias. Estas revelaciones indican que todavía hay muchos misterios acechando en el genoma esperando ser descubiertos.

---

30.6: Estudios de Asociación en todo el genoma is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 30.6: Genome-wide Association Studies has no license indicated.

## 30.7: Direcciones actuales de investigación

Un reto actual en genética médica es el de la traducción. En particular, nos preocupa si GWAS puede informar el desarrollo de nuevas terapias. Los estudios de GWAS han tenido éxito en la identificación de loci asociados a enfermedades. Sin embargo, proporcionan poca información sobre los alelos causales, vías, complejos o tipos celulares que están involucrados. Sin embargo, muchos objetivos drogables conocidos están asociados con impactos GWAS. Por lo tanto, esperamos que GWAS tenga un gran potencial para guiar el desarrollo terapéutico.

Una nueva herramienta en nuestra búsqueda de mayor conocimiento de las perturbaciones genéticas es la secuenciación de próxima generación (NGS). NGS ha hecho que la secuenciación del genoma de un individuo sea una tarea mucho menos costosa y que requiere mucho tiempo. NGS tiene varios usos en el contexto de la genética médica, incluyendo la secuenciación exoma/genoma de enfermedades raras y graves, así como la secuenciación exoma/genoma para la finalización de la arquitectura alélica en GWAS locis. Sin embargo, NGS a su vez ha traído consigo nuevos desafíos en computación e interpretación.

Una aplicación de NGS al estudio de enfermedades humanas es en la identificación y caracterización de variantes de pérdida de función (LoF). Las variantes LoF interrumpen el marco de lectura de los genes que codifican proteínas y, por lo tanto, se espera que sean de interés científico y clínico. Sin embargo, la identificación de estas variantes se complica por errores en la llamada automática de variantes y la anotación génica. Por lo tanto, es probable que muchas variantes putativas de LoF sean falsos positivos. En 2012, MacArthur et al. se propusieron describir un conjunto riguroso de variantes de LoF. Sus resultados sugieren que los genomas humanos típicos contienen alrededor de 100 variantes de LoF. También presentaron un método para priorizar genes candidatos en función de sus características funcionales y evolutivas [24].

El laboratorio MacArthur también está involucrado en un esfuerzo continuo del Exome Aggregation Consortium para ensamblar un catálogo de variación de codificación de proteínas humanas para la minería de datos. Actualmente, el catálogo incluye datos de secuenciación de más de 60 mil individuos. Dichos datos permiten la identificación de genes que carecen significativamente de variación de codificación funcional. Esto es importante porque se espera que los genes bajo restricciones excepcionales sean deletéreos. Con base en este principio, Samocha et al. fueron capaces de identificar 1000 genes involucrados en trastornos del espectro autista que carecían significativamente de variación de codificación funcional.

Esto se hizo usando un marco estadístico que describía un modelo de mutación de novo [32]. De igual manera, De Rubeis et al. fueron capaces de identificar 107 genes bajo una restricción evolutiva excepcional que ocurrió en 5% de los sujetos autistas. Se encontró que muchos de estos genes codifican proteínas involucradas en la transcripción y el corte y empalme, la remodelación de la cromatina y la función sináptica, avanzando así en nuestra comprensión del mecanismo de la enfermedad de estas variantes.

NGS también se puede utilizar para estudiar enfermedades raras y graves, como en el caso de la mutación DGAT1. En un estudio de Haas et al., se utilizó la secuenciación del exoma para identificar una mutación rara en el sitio de empalme en el gen DGAT1. Esto había resultado en trastornos diarreicos congénitos en los hijos de una familia de ascendencia judía asquenazí [13]. En este caso, la secuenciación no solo tuvo aplicaciones terapéuticas para el niño sobreviviente, sino que también proporcionó información sobre un ensayo clínico de inhibición de DGAT1 en curso.

Si bien NGS nos permite estudiar variantes altamente penetrantes que resultan en enfermedades mendelianas graves, también existen estudios genéticos que entregan hipótesis para la intervención. Un ejemplo de ello es el descubrimiento de SCN9A. La pérdida completa de función de SCN9A, también conocida como NaV1.7, resulta en indiferencia congénita al dolor. Esto ha dado como resultado el desarrollo de nuevos analgésicos con una ecacia superior a la de la morfina, como en el caso de μ-SLPTX-SSM6a, un inhibidor selectivo de NaV1.7 [35]. Otro ejemplo es la variante de pérdida de función de PCSK9, que disminuye el LDL y protege contra la enfermedad arterial coronaria. Esto ha llevado al desarrollo del inhibidor de PCSK9 REGN727, el cual ha demostrado ser seguro y efectivo en los senderos clínicos de fase 1 [6].

El NGS también es importante para el mapeo fino de loci identificados en estudios de GWAS. Por ejemplo, estudios de GWAS de 2010 que analizaron la enfermedad de Crohn implicaron una región en el cromosoma 15 que contenía múltiples genes. Después del mapeo fino, el Consorcio Internacional de Genética de Enfermedades Inflamatorias Inflamatorias del Intestino (IIBDGC) pudo refinar la asociación con elementos funcionales no codificantes SMAD3. Otro ejemplo es un estudio de Farh et al. que analizó variantes causales candidatas para 21 enfermedades autoinmunes. Mostraron que 90% de las variantes causales son no codificantes, pero solo 10-20% alteran los motivos de unión al factor de transcripción, lo que implica que los modelos reguladores de genes

actuales no pueden explicar el mecanismo de estas variantes [9]. Finalmente, un estudio de Rivas et al. que analizó una resecuenciación profunda de loci GWAS asociados a enfermedad inflamatoria intestinal encontró no solo nuevos factores de riesgo sino también variantes protectoras. Por ejemplo, se demostró que una variante de empalme protector en CARD9 que causa truncamiento prematuro de proteína protege fuertemente contra el desarrollo de la enfermedad de Crohn [31].

---

30.7: Direcciones actuales de investigación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 30.7: Current Research Directions has no license indicated.

## 30.8: Herramientas y Técnicas

- HapMap, un completo catálogo de SNP humanos.
- PLINK, un conjunto de herramientas GWAS C/C++ de código abierto que puede analizar grandes conjuntos de datos con cientos de miles de marcadores genotipados para miles de individuos para examinar posibles vías.
- GRASS (Gene set Ridge regression in Association Studies), resume la estructura genética de cada gen como EigenSNP y utiliza regresión de cresta grupal para seleccionar los propios SNP representativos para cada gen, evaluando su asociación con el riesgo de enfermedad y reduciendo la alta dimensionalidad de los datos de GWAS.
- GWAMA (Genome-Wide Association Meta-Analysis), realiza metaanálisis de GWAS de fenotipos dicotómicos o rasgos cuantitativos.

---

30.8: Herramientas y Técnicas is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [30.8: Tools and Techniques](#) has no license indicated.

## 30.9: ¿Qué hemos aprendido?

En las últimas décadas, hemos logrado grandes avances en el desarrollo de técnicas para investigar las bases genéticas de la enfermedad. Históricamente, hemos utilizado el análisis de ligamiento para encontrar variantes causales para la enfermedad mendeliana con gran éxito. Más recientemente, hemos utilizado estudios de asociación de todo el genoma para comenzar a investigar rasgos más complejos con cierto éxito. Sin embargo, se necesita más trabajo en el desarrollo de métodos para interpretar estos GWAS e identificar variantes causales y su papel en el mecanismo de la enfermedad. Mejorar nuestra comprensión de la base genética de la enfermedad nos permitirá desarrollar diagnósticos y tratamientos más efectivos.

---

30.9: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [30.9: What Have We Learned?](#) has no license indicated.

## CHAPTER OVERVIEW

### 31: Variación 2- Mapeo cuantitativo de rasgos, EQTLs, Variación de Rasgo Molecular

- 31.1: Introducción
- 31.2: Conceptos básicos de eQTL
- 31.3: Estructura de un estudio eQTL
- 31.4: Direcciones actuales de investigación
- 31.5: ¿Qué hemos aprendido?
- 31.6: Lectura adicional
- 31.7: Herramientas y Recursos
- 31.8: Bibliografía

---

31: Variación 2- Mapeo cuantitativo de rasgos, EQTLs, Variación de Rasgo Molecular is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

## 31.1: Introducción

Las diferencias en las regiones codificadoras de genes entre diferentes organismos no explican completamente la variación fenotípica que vemos. Por ejemplo, aunque la diferencia fenotípica es alta entre humanos y chimpancés y baja entre diferentes especies de ardillas, hay más variación genética entre las especies de ardilla [1]. Estas observaciones nos llevan a concluir que debe haber algo más que una simple variación codificadora de genes que explique la variación fenotípica; específicamente, la variación no codificante también influye en la forma en que se expresan los genes y, en consecuencia, influye en el fenotipo de un organismo. De hecho, investigaciones previas han demostrado que la mayor parte de la variación genética ocurre en regiones no codificantes [2]. Además, se ha encontrado que la mayoría de los patrones de expresión son rasgos heredables.

Comprender cómo la variación en las regiones no codificantes afecta a los genes correguladores nos permitiría no solo comprender sino también controlar la expresión de estos y otros genes relacionados. Esto es especialmente relevante para el control de expresiones de rasgos indeseables como enfermedades poligénicas complejas (Figura 31.1). En la enfermedad mendeliana, la mayoría del riesgo de enfermedad se predice mediante la variación de codificación, mientras que en las enfermedades poligénicas la gran mayoría de la variación causal se encuentra fuera de las regiones codificantes. Esto sugiere que la variación en la regulación de la expresión génica puede jugar un papel mayor que la variación genotípica en estas enfermedades poligénicas. Así, el estudio de estas variantes asociadas a rasgos es un paso en la dirección de entender cómo las secuencias genéticas codifican y controlan la expresión de tales enfermedades y sus fenotipos asociados.

Los eQTLs (loci de rasgos cuantitativos de expresión) encapsulan la idea de regiones no codificantes que influyen en la expresión de ARNm introducida anteriormente: podemos definir un eQTL como una región de variantes en un genoma que están correlacionadas cuantitativamente con la expresión de otro gen codificado por el organismo. Por lo general, veremos que ciertos SNP en ciertas regiones no codificantes mejorarán o interrumpirán la expresión de un determinado gen. El campo de la identificación, análisis e interpretación de los eQTLs en el genoma ha crecido inmensamente en los últimos años con cientos de trabajos de investigación publicados.

Existen cuatro mecanismos principales de cómo los eQTLs influyen en la expresión de sus genes asociados:

1. Alteración de la unión al factor de transcripción Modificaciones histonas
3. Corte y empalme alternativo de ARNm
4. Silenciamiento de miARN

### FAQ

P: ¿Cuál es la diferencia entre un estudio eQTL y un GWAS?

R: Hay dos diferencias fundamentales. El primero se encuentra en la naturaleza del fenotipo que se está examinando. En un eQTL, el fenotipo comprobado suele estar en un nivel inferior de abstracción biológica (niveles normalizados de expresión génica) en lugar de un fenotipo de nivel más alto, a veces visible utilizado en GWAS, como “pelo negro”). En segundo lugar, en GWAS, generalmente porque el fenotipo correlacionado con varios SNP es un fenotipo de nivel superior, muy raramente vemos GWAS específicos de tejido. Sin embargo, en los eQTLs, los patrones de expresión del ARNm podrían variar mucho entre los tipos de tejido dentro del mismo individuo, y se pueden realizar estudios de eQTL para un tipo de tejido específico, como las células neuronales y gliales (Figura 31.2)

31.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 31.1: Introduction has no license indicated.

## 31.2: Conceptos básicos de eQTL

### CIS-EQTLs

El uso del análisis de eQTL del genoma completo ha separado los eQTL en dos tipos distintos de manifestación. El primero es un cis-eQTL (Figura 31.3) en el que la posición del eQTL mapea cerca de la posición física del gen. Debido a la proximidad, los efectos cis-eQTL tienden a ser mucho más fuertes y, por lo tanto, pueden detectarse más fácilmente mediante estudios de GWAS y eQTL. A menudo, estos funcionan como promotores de ciertos polimorfismos, afectan la metilación y conformación de la cromatina (aumentando o disminuyendo así el acceso a la transcripción), y pueden manifestarse como inserciones y delecciones al genoma. Los cis-EQTLs generalmente se clasifican como variantes que se encuentran dentro de 1 millón de pares de bases del gen de interés. Sin embargo, esto es de hecho un corte arbitrario y puede ser alterado en un orden de magnitud, por ejemplo.

### Trans-EQTLs

El segundo tipo distinto de eQTL es un trans-eQTL (Figura 31.4). Un trans-eQTL no mapea cerca de la posición física del gen que regula. Sus funciones son generalmente más indirectas en su efecto sobre la expresión génica (no potenciando o inhibiendo directamente la transcripción sino que afectan a la cinética, vías de señalización, etc.). Dado que tales efectos son más difíciles de determinar explícitamente, son más difíciles de encontrar en el análisis de eQTL; además, tales redes pueden ser extremadamente complejas, limitando aún más el análisis trans-eQTL. Sin embargo, el análisis eQTL ha llevado al descubrimiento de puntos calientes trans que se refieren a loci que tienen efectos transcripcionales generalizados [11].

Quizás la mayor sorpresa de la investigación de eQTL es que, a pesar de la ubicación de puntos calientes trans y cis-eQTLs, no se han encontrado loci trans importantes para genes específicos en humanos [12]. Esto probablemente se le atribuya al proceso actual de análisis de eQTL del genoma completo en sí mismo. Como es útil y generalizado el análisis eQTL del genoma completo, encontramos que la significación de todo el genoma ocurre  $p = 5 \times 10^{-8}$  con múltiples pruebas en aproximadamente 20,000 genes. Por lo tanto, los estudios generalmente utilizan un tamaño muestral inadecuado para determinar la significancia de muchas asociaciones trans-eQTL, que comienzan con previos de muy baja probabilidad para comenzar en comparación con cis-EQTLs [4]. Además, los métodos de reducción de sesgo descritos en secciones anteriores desinflan la varianza, la cual es integral para capturar las asociaciones de microrasgos inherentes a los loci trans. Finalmente, las distribuciones no normales limitan la significación estadística de las asociaciones entre trans-EQTLs y la expresión génica [4]. Esto ha sido ligeramente remediado por el uso de metaanálisis de fenotipo cruzado (CPMA) [5] que se basa en las estadísticas resumidas de GWAS en lugar de datos individuales. Este análisis de rasgos cruzados es efectivo porque los trans-EQTLs afectan a muchos genes y por lo tanto tienen múltiples asociaciones que se originan a partir de un solo marcador. El código CPMA de muestra se puede encontrar en Herramientas y Recursos.

Sin embargo, aunque no se han encontrado loci trans, se han encontrado variantes que actúan en trans. Dado que se puede inferir que los trans-EQTLs afectan a muchos genes, CPMA y ChIP-seq pueden usarse para detectar tales variantes de rasgos cruzados. De hecho, se determinaron 24 diferentes factores de transcripción de acción trans significativos a partir de un grupo de 1311 variantes de SNP que actúan en trans mediante la observación de efectos alélicos en las poblaciones y las interacciones/conexiones de genes diana.

---

31.2: Conceptos básicos de eQTL is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [31.2: eQTL Basics](#) has no license indicated.

### 31.3: Estructura de un estudio eQTL

El enfoque básico detrás de un estudio eQTL es considerar la expresión de cada gen como un rasgo cuantitativo multifactor y retroceder sobre los componentes principales que explican la varianza en la expresión. Primero, se extraen células del tejido de interés y se extrae su ARN. La expresión de proteínas de interés se mide por micromatriz o mediante análisis de RNA-seq. Los niveles de expresión de cada gen se retroceden sobre los genotipos, controlando el ruido biológico y técnico, de tal manera que

$$Y_i = \alpha + X_i\beta + \epsilon_i$$

Donde  $Y_i$  es la expresión génica del gen  $i$ ,  $X_i$  es un vector que contiene la composición alélica de cada SNP asociado con el gen ( $y$  puede tomar valores 0, 1 o 2 dado un alelo de referencia),  $\alpha$  y  $\beta$  son vectores de columna que contienen los coeficientes de regresión, y  $\epsilon_i$  es el error residual (Ver Figura 31.5) [9]. En concepto, tal estudio es sumamente sencillo. En la práctica, existen cientos de posibles factores de confusión e incertidumbres estadísticas que deben contabilizarse en cada paso del proceso. Sin embargo, se puede utilizar el mismo modelo de regresión para dar cuenta de estas covariables.

La Figura 31.9 contiene un ejemplo de estudio eQTL realizado sobre el asma. El resultado clave del estudio es el modelo lineal en la parte superior derecha: podemos ver como el genotipo tiende más hacia la variante “A”, la expresión del gen diana disminuye.

#### Consideraciones para los datos de expresión

La cuantificación de la expresión de genes está plagada de desafíos experimentales. Para una discusión más detallada de estos temas, véase el Capítulo 14. Una consideración importante para este tipo de análisis de expresión es el **SNP- bajo sonda efecto**: las secuencias de sonda que mapean a regiones con variantes comunes proporcionan resultados inconsistentes debido al efecto de la variación dentro de la propia sonda sobre la dinámica de unión. Así, los experimentos repetidos con múltiples conjuntos de sondas producirán un resultado más confiable. El análisis de expresión también debería excluir generalmente **los genes constitutivos**, que no están regulados diferencialmente entre los miembros de una población y/o tipos celulares, ya que estos solo diluirían el poder estadístico del estudio.

#### Consideraciones para los datos genómicos

Existen dos consideraciones principales para el análisis de los datos genómicos: la frecuencia de alelos menores y el radio de búsqueda. El **radio de búsqueda** determina la generalidad del efecto que se está considerando: un radio de búsqueda infinito corresponde a una exploración cis y trans-EQTL de genoma completo, mientras que los radios más pequeños restringen el análisis a cis-EQTLs. La **frecuencia alélica menor** (MAF) determina el punto de corte bajo el cual no se considera un sitio SNP: es un determinante mayor del poder estadístico del estudio. Un mayor corte de MAF generalmente conduce a una mayor potencia estadística, pero MAF y radio de búsqueda interactúan de manera no lineal para determinar el número de alelos significativos detectados (ver Figura 31.6).

#### Ajuste Covariable

Hay muchos posibles factores de confusión estadísticos en un estudio eQTL, tanto biológicos como técnicos. Muchos factores biológicos pueden afectar la expresión observada de cualquier ARNm dado en un individuo; esto se ve exacerbado por la imposibilidad de controlar las circunstancias de prueba de las grandes muestras de población necesarias para lograr significación. La estratificación poblacional y las diferencias genómicas entre grupos raciales son factores contribuyentes adicionales. La variabilidad estadística también existe en el lado técnico. Incluso las muestras ejecutadas en la misma máquina en diferentes momentos muestran un agrupamiento marcadamente diferente de resultados de expresión. (Figura 31.7).

Los investigadores han utilizado con éxito la técnica del **Análisis de Componentes Principales** (ACP) para separar los efectos de estos factores de confusión. El PCA puede producir nuevos ejes de coordenadas a lo largo de los cuales los datos de expresión génica asociados a SNP tienen la mayor varianza, aislando así fuentes no deseadas de variación consistente (ver Capítulo 20.4 para una descripción detallada del Análisis de Componentes Principales). Despues de extraer los componentes principales de los datos de expresión génica, podemos extender el modelo de regresión lineal para dar cuenta de estos factores de confusión y producir una regresión más precisa.

## FAQ

P: ¿Por qué es PCA una herramienta estadística apropiada para usar en este entorno y por qué la necesitamos?

R: Desafortunadamente, nuestros datos brutos tienen varios sesgos y factores externos que dificultarán inferir buenos eQTLs. Sin embargo, podemos pensar en estos sesgos como influencias independientes en los conjuntos de datos que crean varianza artificial en los niveles de expresión que vemos, confundiendo los factores que dan lugar a la varianza real. Usando PCA, podemos descomponer e identificar estas varianzas en sus componentes principales, y filtrarlos adecuadamente. Además, debido a la naturaleza compleja de los rasgos que se analizan, el PCA puede ayudar a reducir la dimensionalidad de los datos y así facilitar el análisis computacional.

## FAQ

P: ¿Cómo decidimos cuántos componentes principales usar?

R: Este es un problema difícil; una posible solución sería probar un número diferente de componentes principales y examinar los eQTLs encontrados después, muy este número para futuras pruebas al ver si los eQTLs generados son viables. Tenga en cuenta que "sería difícil" optimizar diferentes parámetros para el estudio eQTL porque cada conjunto de datos tendrá un número óptimo de componentes principales, un mejor valor para MAF, etc...

## Puntos a considerar

Los siguientes son algunos puntos a considerar al realizar un estudio eQTL.

- La estrategia óptima para el descubrimiento de eQTL en un conjunto de datos específico de todas las diferentes formas de realizar procedimientos de normalización, filtrado de genes no específicos, selección de radio de búsqueda y cortes de frecuencia de alelos menores puede no ser transferible a otro estudio de eQTL. Muchos científicos superan esto usando el ajuste codicioso de estos parámetros, ejecutando el estudio eQTL iterativamente hasta que se encuentre un número máximo de eQTLs significativos.
- Es importante señalar que los estudios eQTL solo encuentran correlación entre marcadores genéticos y patrones de expresión génica, y no implican causalidad.
- Al realizar un estudio de eQTL, tenga en cuenta que los eQTL más significativos se encuentran dentro de unos pocos kb del gen regulado.
- Históricamente, se ha encontrado que la mayoría de los estudios eQTL son aproximadamente 30-40% reproducibles, y esta es una reliquia de cómo se estructura el conjunto de datos y las diferentes estrategias de normalización y filtrado que utilizan los investigadores respectivos. Sin embargo, los eQTL que se encuentran en dos o más cohortes siguen de manera consistente una influencia de expresión similar dentro de cada una de las cohortes.
- Muchos eQTLs son específicos de tejido; es decir, su influencia en la expresión génica podría ocurrir en un tejido pero no en otro, y una posible explicación de esto es la corregulación de un solo gen por múltiples eQTL que depende de que un gen tenga múltiples alelos.

---

31.3: Estructura de un estudio eQTL is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [31.3: Structure of an eQTL Study](#) has no license indicated.

## 31.4: Direcciones actuales de investigación

### Cuantificación de variación de rasgo

Debido a que el estudio de los eQTLs es un estudio en el nivel de expresión de un gen, el paso principal hacia la realización de un estudio informativo es seleccionar rasgos que tienen diferentes niveles de expresión en lugar de expresión binaria. Ejemplos de tales rasgos cuantitativamente viables son el índice de masa corporal (IMC) y la altura. A finales de la década de 1980 y principios de los noventa, los primeros estudios de expresión génica a través de estudios de mapeo de todo el genoma fueron iniciados por Damerval y de Vienne [8] [6]. Sin embargo, su uso de electroforesis 2-D para la separación de proteínas fue ineficiente y poco confiable ya que introdujo mucho ruido y no pudo resumirse sistemática y cuantitativamente. Fue solo a principios de la década de 2000 cuando la introducción de métodos basados en matrices de alto rendimiento para medir la incidencia de ARNm aceleró el uso exitoso de este método, destacado por primera vez en un estudio de Brem [10].

### Nuevas aplicaciones

Hay dos direcciones que encabezan los estudios de eQTL. En primer lugar, hay prisa por utilizar el análisis eQTL del genoma completo para validar asociaciones entre varianzas en la población humana como las diferencias en la expresión génica entre grupos étnicos, ya que el poder estadístico para poder hacerlo comienza a alcanzar el umbral de significación. Una segunda dirección de investigación busca dislocar asociaciones genéticas con diferentes fenotipos y diferencias poblacionales basadas en una base no genética. Estos factores no genéticos incluyen el ambiente, la preparación de la línea celular y los efectos del lote.

31.4: Direcciones actuales de investigación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 31.4: Current Research Directions has no license indicated.

## 31.5: ¿Qué hemos aprendido?

En resumen, la variación más causal para enfermedades poligénicas complejas que hemos descubierto hasta ahora es la no codificación. Además, las diferencias fenotípicas entre especies no se explican bien por la variación codificante, mientras que la expresión génica es altamente heredable entre generaciones. Así, se propone que el control genético de los niveles de expresión es un factor crucial para determinar la varianza fenotípica.

Los eQTLs son loci variantes de SNP que se correlacionan con los niveles de expresión génica. Vienen en una de dos formas. Los cis-eQTLs son sitios cuyos loci se mapean cerca de los genes afectados, son relativamente fáciles de detectar debido a su proximidad, y generalmente tienen claros mecanismos de acción. Los trans-eQTLs mapean a áreas distanciadas del genoma, son más difíciles de detectar y sus mecanismos no son tan directos.

Los estudios de eQTL combinan un enfoque de genoma completo similar a GWAS con un ensayo de expresión, ya sea microarray o RNA-seq. Los niveles de expresión de cada gen se correlacionan por regresión lineal con genotipos después de usar PCA para extraer factores de confusión. Determinar los parámetros óptimos para MAF, radio de búsqueda y normalización de confusores es una pregunta de investigación abierta. Las aplicaciones de los eQTLs incluyen la identificación de variantes asociadas a enfermedades así como variantes asociadas a subespecies poblacionales y la varianza genética y ambiental que da lugar a rasgos complejos,

---

31.5: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [31.5: What Have We Learned?](#) has no license indicated.

## 31.6: Lectura adicional

La siguiente es una muy buena revisión introductoria de la literatura sobre los EQTLs, incluyendo su historia y aplicaciones actuales:

El papel de la variación regulatoria en los rasgos complejos y la enfermedad

Frank W. Albert y Leonid Kruglyak Nature Comentarios Genética 16 2015

También hay algunos trabajos de investigación que son pioneros en lo que es actual en los estudios de eQTL. Uno de estos artículos es informativo sobre la ocurrencia de metilación del ADN que afecta la expresión génica en el cerebro humano. Otro estudio es un estudio sobre los cambios en la expresión durante el desarrollo en el nematodo *C. elegans*, utilizando la edad como covariable durante el mapeo de eQTL:

1. Existen abundantes loci de rasgos cuantitativos para la metilación del ADN y la expresión génica en el cerebro humano

Gibbs JR, van der Brug MP, Hernández DG, Traynor BJ, Nalls MA, et al. PLOS Genet 6 2010

2. Los efectos de la variación genética sobre la dinámica de expresión génica durante el desarrollo Francesconi, M. y Lehner, B. Nature 505 2013

Además, recientemente se ha encontrado que las variantes de eQTL están implicadas en enfermedades como la enfermedad de Crohn y la esclerosis múltiple [4].

Como se menciona en la Sección 4.2, también ha habido un aumento reciente en los estudios que aplican estudios eQTL para delinejar diferencias entre subpoblaciones humanas y caracterizar las contribuciones del ambiente hacia la variación de rasgos:

1. Variantes genéticas comunes explican diferencias en la expresión génica entre los grupos étnicos Spielman RS, Bastone LA, Burdick JT, Morley M, Ewens WJ, Cheung VG.

Nature Genetics 2007

2. Variación de la expresión génica dentro y entre las poblaciones humanas Storey JD, Madeoy J, Strout JL, Wurfel M, Ronald J, Akey JM

The American Journal of Human Genetics 2007

3. Genómica poblacional de la expresión génica humana [12] Stranger BE, Nica AC, Forrest MS, et. al.

The American Journal of Human Genetics 2007

4. Evaluación de la variación genética que contribuye a las diferencias en la expresión génica entre poblaciones

Zhang W, Duan S, Kistner EO, Bleibel WK, Huang RS, Clark TA, Chen TX, Schweitzer AC, Blume JE, Cox NJ, Dolan ME  
The American Journal of Human genetics 2008

5. Una firma de expresión génica en todo el genoma de la geografía ambiental en leucocitos de amazighs marroquíes

Isaghdour Y, Storey JD, Jadallah SJ, Gibson G

PLoS 2008

6. Sobre el diseño y análisis de estudios de expresión génica en poblaciones humanas Joshua M Akey, Shameek Biswas, Jeffrey T

Leek, John D Storey

Nature Genetics 2007

---

31.6: Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 31.6: Further Reading has no license indicated.

## 31.7: Herramientas y Recursos

- El Laboratorio Costapas distribuye código para calcular CPMA a partir de los valores p de la asociación GWAS que se pueden encontrar aquí: [www.cotsapaslab.info/index.php/software/cpma/](http://www.cotsapaslab.info/index.php/software/cpma/)
- El laboratorio Pritchard cuenta con varios recursos (que se encuentran aquí: <http://eqtl.uchicago.edu/Home.html>) para la investigación de eQTL y la regulación génica, incluyendo:
  - Datos DNase-seq de 70 líneas celulares linfoblastoides YRI
  - Descarga de posiciones de sitios de unión al factor de transcripción inferidos en el linfoblastoide HapMap líneas celulares por CENTIPEDE
  - Datos sin procesar y mapeados de RNA-seq de Pickrell et al.
  - Surtido de guiones para identificar lecturas de secuenciación que cubren genes, sitios de poliadenilación y uniones exón- exón
  - Datos y resultados de MeQTL para datos de metilación de Illumina27K en líneas celulares linfoblastoides HapMap.
  - Archivos para ignorar áreas del genoma que son propensas a causar falsos positivos en Chip-seq y otros ensayos funcionales basados en secuenciación
  - Navegador para EQTLs identificados en estudios recientes en múltiples tejidos
- El Wellcome Trust Sanger Institute ha desarrollado bases de datos empaquetadas y servicios web (Genevar) que están diseñados para ayudar al análisis integrador y visualización de asociaciones de genes SNP en estudios eQTL. Esta información se puede encontrar aquí: [www.sanger.ac.uk/resources/software/genevar/](http://www.sanger.ac.uk/resources/software/genevar/)
- El Instituto Wellcome Trust Sanger también ha desarrollado bases de datos que contienen información relevante para estudios eQTL, como encontrar e identificar todos los elementos funcionales en la secuencia del genoma humano y mantener anotaciones automáticas en genomas eucariotas seleccionados. Esta información se puede encontrar aquí: <http://www.sanger.ac.uk/resources/databases/>.
- Finalmente, el NIH está progresando en el Proyecto de Expansión de Tejido Genotípico (GTeX). Actualmente, el proyecto se ubica en 35 tejidos de 50 donantes; el objetivo es adquirir y analizar 20 mil tejidos de 900 donantes, con la esperanza de recopilar aún más datos para posteriores análisis genéticos, especialmente para análisis eQTL y trans-EQTL que requieren tamaños de muestra más grandes.

31.7: Herramientas y Recursos is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 31.7: Tools and Resources has no license indicated.

## 31.8: Bibliografía

- [1] King, Mary-Claire y Wilson, A.C. (abril de 1975) Evolución a Dos Niveles en Humanos y Chimpánzés Ciencia Vol.188 Núm. 4184
- [2] 1000 Genomas Proyecto Consorcio. Naturaleza. 2010; 467:1061-73.
- [3] Cheung Vivien G. y Spielman Richard S. (2009) Genética de la expresión génica humana: mapeo de ADN Variantes que Influyen en la Expresión Génica Naturaleza Reseñas
- [4] C. Cotsapas, Variación regulatoria y EQTLs. 2012 Nov 1.
- [5] C. Cotsapas, BF Voight, E Rossin, K Lage, BM Neale, et al. (2011) Pervasive Sharing of Genetic Effects in Autoimmune Disease. PLoS Genet 7 (8) :e1002254. doi:10.1371/journal.pgen.1002254
- [6] Damerval C, Maurice A, Josse JM, de Vienne D (mayo de 1994). Variación del producto génico subyacente a los loci de rasgos cuantitativos: una nueva perspectiva para analizar la regulación de la genética de la expresión del genoma 137 (1): 289301.PMC 1205945. PMID 7914503.
- [7] Dimas AS, et. al. (Sept. 2009) La variación reguladora común impacta la expresión génica de una manera dependiente del tipo celular. Ciencia 325 (5945) :1246-50. 2 Epub 2009 Jul 30.
- [8] D. de Vienne, A. Leonard, C. Damerval (nov 1988). Aspectos genéticos de la variación de las cantidades proteicas en maíz y guisante. Electroforesis 9 (11): 742750. doi:10.1002/elps.1150091110. PMID 3250877.
- [9] Shengjie Yang, Yiyuan Liu, Ning Jiang, Jing Chen, Lindsey Leach, Zewei Luo, Minghui Wang. Genome- EQTLs amplios y heredabilidad para rasgos de expresión génica en individuos no relacionados. BMC Genómica 15 (1): 13. 2014 Ene 9.
- [10] Rachel B. Brem y Leonid Kruglyak. El panorama de complejidad genética a través de 5,700 rasgos de expresión génica en levaduras. PNAS 102 (5): 15721577. 23 nov 2004.
- [11] Michael Morley, Cliona M. Molony, Teresa M. Weber, James L. Devlin, Kathryn G. Ewens, Richard S. Spielman, Vivian G. Cheung. Análisis genético de la variación genómica en la expresión génica humana. Naturaleza 430:743-747. 12 ago 2004.
- [12] Barbara E Extraño, Alexandra C Nica, Matthew S Forrest, Antígona Dimas, Christine P Pájaro, Claude Beazley, Catherine E Ingle, Mark Dunning, Paul Flück, Daphne Koller, Stephen Montgomery, Simon Tavar, Panos Deloukas, Emmanouil T Dermitzakis. Genómica poblacional de la expresión génica humana. Nature Genetics 39:1217 - 1224. 16 sep 2007.

31.8: Bibliografía is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 31.8: Bibliography has no license indicated.

## CHAPTER OVERVIEW

### 32: Genomas Personales, Genomas Sintéticos, Computación en C vs Si

[32.1: Introducción](#)

[32.2: Genomas de Lectura y Escritura](#)

[32.3: Genomas personales](#)

[32.4: Lectura adicional](#)

[32.5: Bibliografía](#)

---

[32: Genomas Personales, Genomas Sintéticos, Computación en C vs Si](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 32.1: Introducción

George Church discutió una variedad de temas que han motivado su investigación pasada y presente. Primero discutió sobre lectura y escritura de genomas, incluyendo su propia participación en el desarrollo de la secuenciación y el Proyecto Genoma Humano. En esa segunda mitad, discutió sobre su emprendimiento más reciente, el Proyecto Genoma Personal, que inició en 2005.

32.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 32.1: Introduction has no license indicated.

## 32.2: Genomas de Lectura y Escritura

Como motivación, considera la siguiente pregunta: ¿Existe alguna tecnología que no esté motivada o inspirada biológicamente? La biología y nuestras observaciones de la misma influyen en nuestras vidas de manera generalizada. Por ejemplo, dentro del sector energético, la biomasa y la bioenergía siempre han existido y cada vez se están convirtiendo cada vez más en el foco de atención. Incluso en las telecomunicaciones, el potencial de la computación molecular de nivel cuántico es prometedor, y se espera que sea un actor importante en el futuro.

Church ha estado involucrado en la computación molecular en su propia investigación, y afirma que una vez aprovechada, tiene grandes ventajas sobre sus actuales contrapartes de silicio. Por ejemplo, la computación molecular puede proporcionar al menos un 10% mayor de eficiencia por Joule en la computación. Más profundo quizás sea su efecto potencial en el almacenamiento de datos. Los medios actuales de almacenamiento de datos (disco magnético, unidades de estado sólido, etc.) son mucho menos densos (miles de millones de veces) que el ADN. La limitación del ADN como almacenamiento de datos es que tiene una alta tasa de error. Church está actualmente involucrada en un proyecto que explora el almacenamiento confiable mediante el uso de la corrección de errores y otras técnicas.

En un artículo de revisión de *Nature Biotechnology* de 2009 [1], Church explora el potencial de métodos eficientes para leer y escribir al ADN. Observa que en la última década ha habido una curva exponencial de 10 tanto en secuenciación como en síntesis de oligo, con síntesis bicatenaria rezagada pero en constante aumento. En comparación con la curva exponencial 1.5 para VLSI (Ley de Moore), el aumento en el lado biológico es más dramático, y aún no hay argumento teórico de por qué la tendencia debería disminuir. En resumen, existe un gran potencial para la síntesis e ingeniería del genoma.

### ¿Sabías?

George Church fue uno de los primeros pioneros de la secuenciación del genoma. En 1978, Church pudo secuenciar plásmidos a \$10 por base. Para 1984, junto con Walter Gilbert, desarrolló el primer método de secuenciación genómica directa [3]. Con este avance, ayudó a iniciar el Proyecto Genoma Humano en 1984. Esta propuesta tenía como objetivo secuenciar un genoma haploide humano completo a \$1 por base, requiriendo un presupuesto total de 3 mil millones de dólares. Esto rápidamente jugó en la conocida carrera entre Celera y UCSC-Broad-Sanger. Aunque estos últimos apenas ganaron al final, su secuencia tuvo muchos errores y brechas, mientras que la versión de Celera era de mucha mayor calidad. Celera inicialmente planeó liberar el genoma en fragmentos de 50 kb, en los que los investigadores podrían realizar alineaciones, al igual que BLAST. Church una vez se acercó al fundador de Celera, Craig Venter, y recibió la promesa de obtener todo el genoma en DVD después del lanzamiento. No obstante, cuestionando la promesa, Church decidió en cambio descargar el genoma directamente de Celera aprovechando los lanzamientos de fragmentos cortos. Usando scripts automatizados de rastreo y descarga, Church logró descargar todo el genoma en fragmentos de 50 kb en tres días!

32.2: Genomas de Lectura y Escritura is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [32.2: Reading and Writing Genomes](#) has no license indicated.

## 32.3: Genomas personales

En 2005, George Church inició el Proyecto Genoma Personal [2]. Ahora que los costos de secuenciación han disminuido rápidamente hasta el punto de que actualmente podemos obtener todo el genoma humano diploide por \$4000 (en comparación con \$3 mil millones para un genoma humano haploide en el Proyecto Genoma Humano), la información personal del genoma y la secuencia se está volviendo cada vez más asequible.

Una aplicación importante para esta información es en la medicina personalizada. Aunque muchas enfermedades siguen siendo complicadas de predecir, diagnosticar y estudiar, actualmente ya tenemos una pequeña lista de enfermedades que son altamente predecibles a partir de los datos del genoma. Los ejemplos incluyen fenilcetonuria (PKU), cáncer de mama relacionado con la mutación BRCA y miocardiopatía hipertrófica (HCM). Muchas de estas y otras enfermedades similares son inciertas (aparición repentina sin síntomas de advertencia) y normalmente no se revisan (debido a su relativa rareza). Como tales, son particularmente adecuados como dianas para la medicina personalizada por genomas personales, ya que los datos genómicos proporcionan información precisa que de otra manera no se puede obtener. Ya hay más de 2500 enfermedades (debido a ~ 6000 genes) que son altamente predecibles y médicalemente procesables, y empresas como 23andMe están explorando estas oportunidades.

Como comentario final sobre el tema, Church remarcó algunos de su filosofía personal respecto a la medicina personalizada. Él encuentra muchas personas reacias a obtener su información genómica, y lo atribuye a una visión negativa entre el público en general hacia GWAS y medicina personalizada. Cree que los medios de comunicación se centran demasiado en el fracaso de GWAS. El argumento de larga data en contra de la medicina personalizada es que debemos enfocarnos primero en enfermedades comunes y variantes antes de estudiar eventos raros. Church contrarresta que de hecho no existe tal cosa como una enfermedad común. Fenómenos como presión arterial alta o colesterol alto solo cuentan como síntomas; muchas 'enfermedades comunes' como enfermedades cardíacas y cáncer tienen muchos subtipos y categorías más finas. Todo el tiempo, agrupar estas enfermedades en una gran categoría solo tiene el beneficio de enseñar a estudiantes de medicina y vender productos farmacéuticos (por ejemplo, estatinas, a las que les ha ido bien comercialmente pero solo son muy pocas). Church sostiene que la acumulación implica una pérdida de poder estadístico, y sólo es útil si realmente tiene sentido. En última instancia, cada uno muere debido a su propia constelación de genes y enfermedades, por lo que Church ve que la división (genómica personalizada) es la manera de proceder.

La genómica personal proporciona información para la planeación y la investigación. Como modelo de negocio, es análogo a una póliza de seguro, que brinda gestión de riesgos. Sin embargo, como beneficio adicional, la información recibida permite la detección temprana, y las consecuencias pueden incluso ser evitables. El acceso a la información genómica permite tomar decisiones más informadas.

---

32.3: Genomas personales is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 32.3: Personal Genomes has no license indicated.

## 32.4: Lectura adicional

Proyecto Genoma Personal: <http://www.personalgenomes.org/>

32.4: Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 32.4: Further Reading has no license indicated.

## 32.5: Bibliografía

- 
- [1] Peter A. Carr y George M. Church. Ingeniería del genoma. Biotecnología de la naturaleza, 27 (12) :1151—1162, diciembre de 2009.
  - [2] G. M. Iglesia. El Proyecto Genoma Personal. Biología de Sistemas Moleculares, 1 (1) :MSB4100040—E1— MSB4100040—E3, diciembre de 2005.
  - [3] G. M. Church y W. Gilbert. Secuenciación genómica. Actas de la Academia Nacional de Ciencias de los Estados Unidos de América, 81 (7) :1991—1995, abril de 1984.
- 

[32.5: Bibliografía](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [32.5: Bibliography](#) has no license indicated.

## CHAPTER OVERVIEW

### 33: Genómica personal

- 33.1: Introducción
- 33.2: Epidemiología- Una visión general
- 33.3: Epidemiología Genética
- 33.4: Epidemiología Molecular
- 33.5: Modelado y Pruebas de Causalidad
- 33.6: ¿Qué hemos aprendido?

---

33: Genómica personal is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 33.1: Introducción

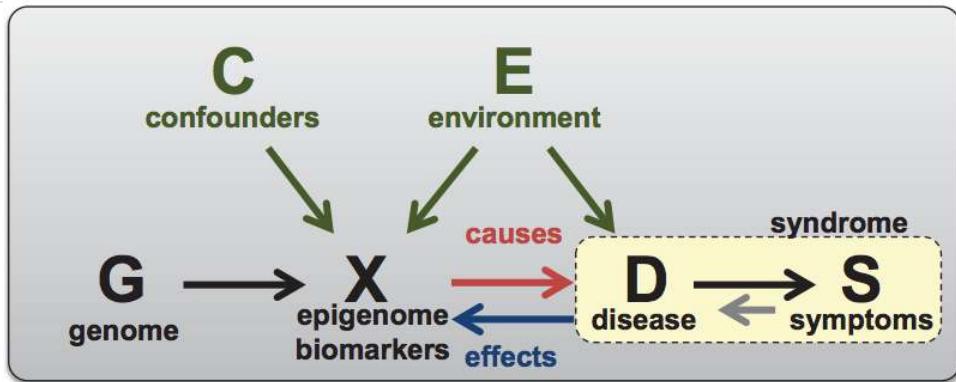
La genómica personalizada se centra en el análisis de los genomas de los individuos y sus predisposiciones para enfermedades en lugar de mirar a nivel poblacional. La medicina personalizada solo es posible con información sobre genética junto con información sobre muchos otros factores como la edad, la nutrición, el estilo de vida o los marcadores epigenéticos (como la metilación). Para que la medicina personalizada sea más realidad, necesitamos aprender más sobre las causas y patrones de enfermedades en poblaciones e individuos.

---

33.1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 33.1: Introduction has no license indicated.

## 33.2: Epidemiología- Una visión general



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative

Licencia Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 33.1: Factores que contribuyen a la probabilidad de contraer una enfermedad. Cada relación mostrada representa correlación excepto el vínculo entre genoma y enfermedad. Correlación no significa causalidad, pero podemos usar el genoma para resolver la causalidad.

La epidemiología es el estudio de patrones, causas y efectos de las condiciones de salud y enfermedad en poblaciones definidas. Para hablar de epidemiología, primero debemos entender algunas definiciones y términos básicos: El **nivel de morbilidad** es lo enfermo que está un individuo mientras que la mortalidad es si un individuo está muerto o no. La incidencia es una tasa que describe el número de nuevos casos/personas con una enfermedad que aparece durante un periodo de tiempo. La **prevalecia** es el número total de casos en estado estacionario en la población. El **riesgo atribuible** es la diferencia en la tasa de una enfermedad entre los expuestos a la enfermedad y los no expuestos a la enfermedad. La **carga poblacional** se refiere a los años de vida potencial perdida (YPLL), año de vida ajustado por calidad o ajustado por discapacidad (QALY/DALY). El **síndrome** se refiere a los signos o síntomas concurrentes de una enfermedad que se observan. El **reto de la prevención** es determinar una enfermedad y su causa y entender si, cuándo y cómo intervenir.

Para determinar las causas de la enfermedad, los estudios deben diseñarse de acuerdo con ciertos principios del diseño experimental. Estos principios incluyen control, aleatorización, replicación, agrupación, ortogonalidad y combinatoria. Se necesitan grupos de control para que se pueda hacer la comparación con una línea basal. El efecto placebo es real, por lo que es necesario contar con un grupo de control. Las personas que reciben el tratamiento putativo que se está probando también deben ser aleatorias para que no haya sesgo. El estudio también necesita ser replicado para controlar la variabilidad en la muestra inicial. (Esto es similar a la maldición de los ganadores. Alguien puede ganar una carrera porque lo hizo sobresaliente en esa ronda en particular y superó su promedio personal, pero en la siguiente ronda probablemente regresarán a desempeñarse cerca de su promedio). Comprender la variación entre diferentes subgrupos también puede desempeñar un papel importante en los resultados de los experimentos. Estos pueden incluir subgrupos basados en la edad, el género o la demografía. Un subgrupo de la población puede estar aportando de una manera más profunda que ellos descansan, por lo que es importante mirar a cada subgrupo específicamente. La ortogonalidad, o la combinación de todos los factores y tratamientos, y la combinatoria, el diseño factorial, también se debe tener en cuenta a la hora de diseñar un experimento. Con los estudios de enfermedades en particular, se debe tomar en cuenta la ética al tratar con sujetos humanos. Existen limitaciones legales y éticas que son supervisadas por las juntas de revisión. Los ensayos clínicos deben realizarse ya sea ciego (el paciente no sabe si está recibiendo tratamiento o no) o doble ciego (el médico tampoco lo sabe). Un paciente que sepa si ha recibido un tratamiento puede cambiar sus hábitos causando sesgos, o un médico que sepa que un paciente recibió el tratamiento puede tratarlo de manera diferente o analizar sus resultados de manera diferente. Ambas consideraciones deben tenerse en cuenta para disminuir el sesgo que puede ocasionar diferentes resultados de un ensayo clínico.

**Ejemplo** Un ejemplo de la necesidad de un ensayo de control aleatorio es el tratamiento del ébola. Un tratamiento debe distribuirse aleatoriamente a las personas atendidas en diferentes hospitales y debe ser ciego. Si alguien cree que está recibiendo la vacuna, puede alterar sus hábitos para protegerse lo que puede afectar el resultado. Si sólo los pacientes de un hospital reciben la vacuna, existe la posibilidad de que los efectos vistos sean solo de ese hospital siendo más cuidadosos.

## FAQ

P: En experimentos mal diseñados, ¿hay algún aspecto que más comúnmente se pasa por alto?

R: La más comúnmente omitida es la estructura de subgrupos. A veces no es obvio cuáles podrían ser los diferentes subgrupos. Para ayudar con esto, los investigadores pueden observar las propiedades generales de un predictor tratando de agrupar casos y controles de forma independiente y visualizar la agrupación. Si hay una subestructura distinta de caso/control en la agrupación, los investigadores pueden buscar variables dentro de cada clúster para ver qué es la subestructura impulsora.

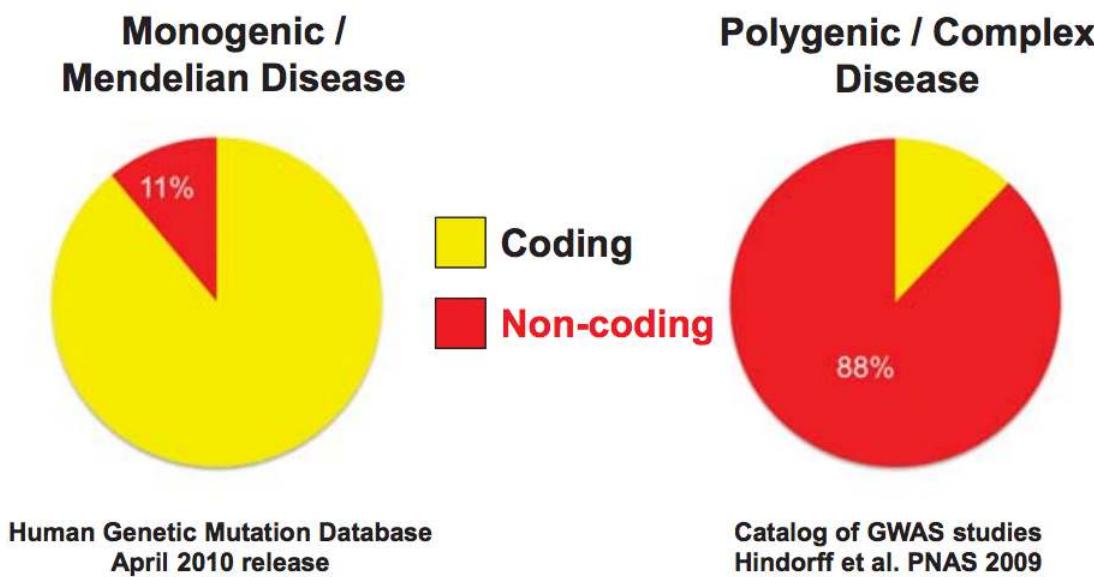
---

[33.2: Epidemiología- Una visión general](#) is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [33.2: Epidemiology- An Overview](#) has no license indicated.

### 33.3: Epidemiología Genética

La epidemiología genética se centra en los factores genéticos que contribuyen a la enfermedad. Los estudios de asociación Genome-Wide (GWAS), descritos previamente en profundidad, identifican variantes genéticas que están asociadas con una enfermedad en particular mientras ignoran todo lo demás que pueda ser un factor. Con la disminución de la secuenciación del genoma completo, este tipo de estudios son cada vez más frecuentes.



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 33.2: Los SNP asociados a enfermedades mendelianas a menudo se encuentran en regiones codificantes, mientras que los asociados con enfermedades poligénicas generalmente se encuentran en regiones no codificantes. Esto se debe a que las variantes de gran efecto, variantes codificantes de proteínas, asociadas a enfermedades mendelianas se encuentran en bajas frecuencias debido a la selección. Las variantes comunes asociadas a enfermedades poligénicas, tienden a tener un menor efecto, por lo que la selección no juega un papel tan importante.

En epidemiología genética existen muchos factores genéticos que puedes probar para identificar enfermedades en un individuo en particular. Se pueden observar alelos de riesgo familiar que se heredan con un rasgo común en genes o variantes específicos. Se pueden estudiar mutaciones monogénicas y procesables codificadoras de proteínas que son las más entendidas, tendrían el mayor impacto y serían las más fáciles de interpretar. Existe la posibilidad de probar todos los SNP codificantes (polimorfismos de un solo nucleótido) con una asociación de enfermedad conocida. Hay debates sobre si un paciente necesita o querría conocer esta información a veces especialmente si la enfermedad no es tratable. La calidad de vida de una persona puede disminuir solo por saber que puede tener la enfermedad intratable aunque no se presenten síntomas. También puedes probar todas las asociaciones codificantes y no codificantes de GWAS, todos los SNP comunes independientemente de la asociación con cualquier enfermedad, o todo el genoma.

#### ¿Sabías?

23andMe es una compañía de genómica personal que ofrece pruebas de genoma directo al consumidor basadas en saliva. 23andMe brinda a los consumidores datos genéticos brutos, resultados relacionados con la ascendencia y estimaciones de predisposición para más de 90 rasgos y afecciones. En 2010, la FDA notificó a varias empresas de pruebas genéticas, entre ellas 23andMe, que sus pruebas genéticas se consideran dispositivos médicos y se requiere la aprobación federal para comercializarlas. En 2013, la FDA ordenó a 23andMe que dejara de comercializar su Kit de Recogida de Saliva y Servicio de Genoma Personal (PGS) ya que 23andMe no había demostrado que hayan “validado analítica o clínicamente el PGS para sus usos previstos” y la “FDA está preocupada por las consecuencias para la salud pública de resultados inexactos desde el

dispositivo PGS” [? ]. La FDA expresó su preocupación por los resultados de riesgo genético tanto falsos negativos como falsos positivos, diciendo que un falso positivo puede hacer que los consumidores se sometan a cirugía, detección intensiva o quimioprevención en el caso de riesgo relacionado con BRCA, por ejemplo, mientras que un falso negativo puede impedir que los consumidores obtengan la cuidados que necesitan. En clase, discutimos si las personas deben ser informadas sobre los alelos de riesgo potenciales que puedan portar. A menudo, las personas pueden malinterpretar las probabilidades que se les brindan y subestimar o sobreestimar lo preocupadas que deberían estar. También se planteó el argumento de que no se debe decir a las personas que están en riesgo si no hay nada que la medicina y la tecnología actuales puedan hacer para mitigar el riesgo. Si se va a informar a las personas sobre un riesgo, el riesgo debe ser accionable; es decir, deberían poder hacer algo al respecto, en lugar de simplemente vivir en la preocupación, ya que ese estrés agregado puede causarles otros problemas de salud.

No sólo existe la elección de qué probar, existe la cuestión de cuándo hacer una prueba a alguien para una condición en particular. Las pruebas diagnósticas ocurren después de que se muestran los síntomas para confirmar una hipótesis o distinguir entre diferentes posibilidades de tener una afección. También puede probar el riesgo predictivo que ocurre antes de que un paciente muestre los síntomas. Se pueden realizar pruebas a los recién nacidos con el fin de intervenir temprano o incluso hacer pruebas prenatales a través de una ecografía, suero materno, sondas o muestreo de vellosidades coriónicas. Para evaluar qué trastornos puede transmitir a su hijo, puede hacer pruebas previas a la concepción. También puedes hacer pruebas de portador para determinar si eres portador de un alelo mutante en particular que pueda aparecer en tu historia familiar. Hacer pruebas genéticas y biomarcadores puede ser complicado porque puede desconocerse si la genética o biomarcador visto está causando la enfermedad o es consecuencia de tener la enfermedad.

Para interpretar las asociaciones de enfermedades, necesitamos utilizar la epigenómica y la genómica funcional. Las asociaciones genéticas siguen siendo solo probabilísticas: si tienes una variante genética, aún existe la posibilidad de que no contraigas la enfermedad. Sin embargo, con base en estadísticas bayesianas, la probabilidad posterior aumenta si aumenta la previa. A medida que encontramos cada vez más asociaciones y variantes, el valor predictivo aumentará.

---

33.3: Epidemiología Genética is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 33.3: Genetic Epidemiology has no license indicated.

### 33.4: Epidemiología Molecular

La Epidemiología Molecular implica observar los biomarcadores moleculares de un estado de enfermedad. Esto incluye observar los perfiles de expresión génica, los patrones de metilación del ADN, es decir, la epigenómica y la estructura y organización de la cromatina en tipos celulares específicos. En capítulos anteriores, discutimos el vínculo entre la expresión génica (como ARN o proteínas) y los SNP en el contexto de estudios de eQTL. Como recordatorio, los EQTLs (loci de rasgos cuantitativos de expresión) buscan correlaciones lineales entre los niveles de expresión génica y diferentes variantes de un locus genético.

Esta sección se enfocará en comprender el papel de los marcadores epigenómicos como indicadores moleculares de una enfermedad. Es importante entender que múltiples factores, y por lo tanto múltiples conjuntos de datos entran en juego para comprender las bases epigenómicas de la enfermedad: patrones de metilación de pacientes de muestra (M), información genómica (G) para los mismos individuos, datos ambientales (E, cubriendo covariables como edad, género, tabaquismo hábitos etc.), y las cuantificaciones de fenotipos (P, pueden capturar múltiples marcadores fenotípicos, por ejemplo en la Enfermedad de Alzheimer, el número de placas neuronales por paciente). Además, necesitamos comprender las diversas interconexiones y dependencias entre estos conjuntos de datos para sacar conclusiones significativas sobre la influencia de la metilación para una determinada enfermedad.

Para eliminar covariantes experimentales, técnicas o ambientales, nos basamos en correcciones conocidas o inferidas por ICA (análisis de componentes pendientes). Para vincular los datos genéticos a los patrones de metilación, buscamos MeQTLs (loci tratí cuantitativo de metilación), que es equivalente a los EQTLs. Los fenotipos moleculares como el nivel de expresión o el nivel de metilación también son rasgos cuantitativos. Finalmente, para vincular patrones de metilación con enfermedades, implementamos EWAS (Epigenome-wide association studies).

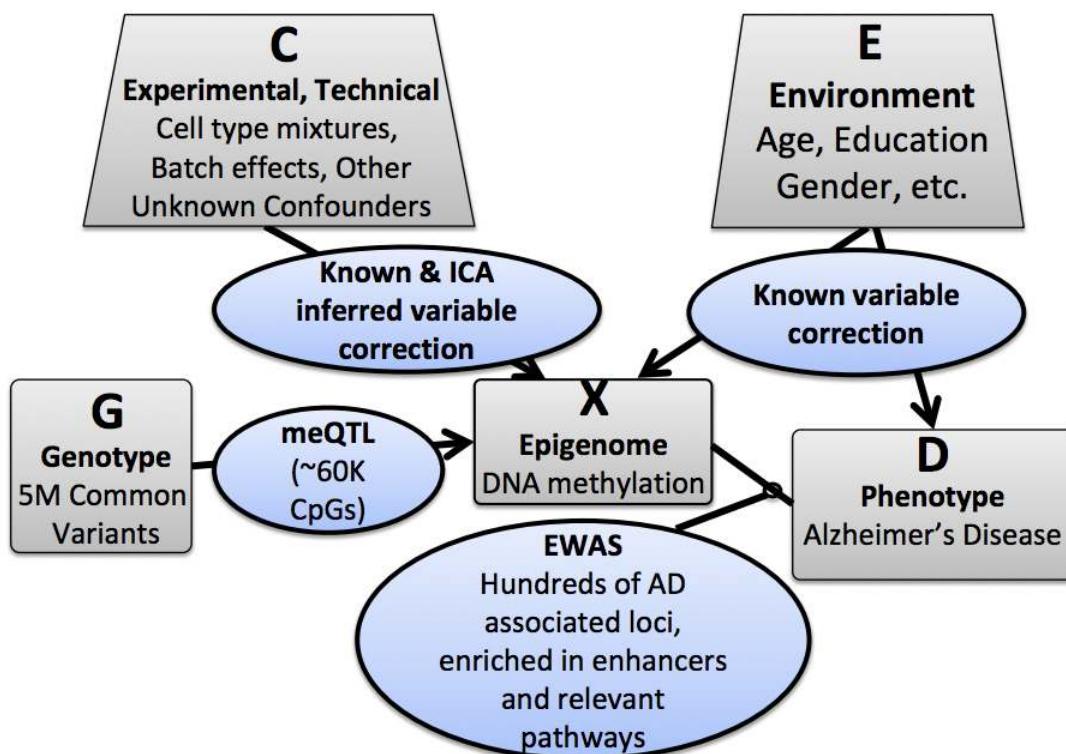


Figura 33.3: Los múltiples factores y conjuntos de datos para determinar el papel de la metilación en los estados de enfermedad, y los métodos para vincular estos conjuntos de datos.

#### MEQTLs

El descubrimiento de MEQTLs sigue un proceso que es muy similar a la metodología utilizada para descubrir los EQTLs. Para descubrir cis-MEQTLs (es decir, MEQTLs donde el efecto sobre la metilación es proximal al locus probado) seleccionamos una ventana genómica, y utilizamos un modelo lineal para probar si vemos o no una correlación entre la metilación y las variantes de

SNP en esa región. Probamos para ver si la correlación es significativa a través de una prueba F, donde nuestra hipótesis nula es que la complejidad del modelo adicional introducida a través de la información genómica no explica una porción significativa de variación en los patrones de metilación. Otros métodos para descubrir los MEQTLs incluyen la permutación y los modelos mixtos lineales (LMM).

**Ejemplo** Un ejemplo del uso de los MEQTLs para descubrir la conexión entre la metilación, el genotipo y la enfermedad es el Proyecto Memoria y Envejecimiento. 750 personas mayores inscritas en el proyecto hace muchos años y hoy en día, en su mayoría han muerto y han entregado su cerebro a la ciencia. Se determinó el genotipo y metilación de la corteza prefrontal lateral dorsal para estudiar la conexión entre la metilación y el fenotipo de Alzheimer y cómo el genotipo puede afectar el perfil de metilación. Se tomaron en cuenta los datos de SNP, metilación, factores ambientales (como edad, género, lote muestral, estado tabáquico, etc.) y fenotipo. Primeras covariantes necesitaron ser descubiertas y excluidas para asegurarse de que los resultados obtenidos no se deben a factores de confusión. Esto se hace descomponiendo la matriz de datos de metilación haciendo ICA. Esto permite el descubrimiento de variables que están impulsando la mayor variabilidad en el rasgo. La muestra discontinua y la mezcla celular pueden tener el mayor efecto en la variación entre individuos. Después de que esto se corrija, se utilizan modelos lineales, pruebas de permutación y modelos mixtos lineales para determinar cis-MEQTL, en qué medida el genotipo explica el nivel de metilación.

## EWAS

Los estudios del genoma de todo el epigenoma (EWAS) tienen como objetivo encontrar conexiones entre el patrón de metilación de un paciente y su fenotipo. Al igual que GWAS, EWAS se basa en modelos lineales y pruebas de valor p para encontrar vínculos entre los perfiles epigenómicos y los estados de enfermedad. Junto con los MEQTLs, los EWAS también pueden arrojar luz sobre si un patrón de metilación dado es la causa o el resultado de una enfermedad. Idealmente, la idea es poder generar modelos que nos permitan predecir estados de enfermedad (fenotipos) basados en la metilación.

Hay algunos inconvenientes en EWAS. Primero, la varianza en los patrones de metilación por fenotipo suele ser muy pequeña, lo que dificulta la vinculación de estados epigenómicos con estados de enfermedad, similar a buscar una aguja en un pajar. Para mejorar esta situación, necesitamos controlar por otras fuentes de varianza en nuestros datos de metilación, como género, edad etc. El género, por ejemplo, incorpora una gran varianza para el caso de la Enfermedad de Alzheimer. Adicionalmente, necesitamos tener en cuenta la varianza por genotipo (en forma de MEQTLs). Además, la variabilidad entre las muestras es un problema importante en la recolección de datos de metilación para EWAS [? ]. Como diferentes tipos de células en un mismo individuo tendrán diferentes firmas epigenómicas, es importante que se colecten muestras de tejido relevantes y se corrijan los datos para los diferentes tipos de células/tejidos involucrados en un estudio.

---

33.4: Epidemiología Molecular is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 33.4: Molecular Epidemiology has no license indicated.

### 33.5: Modelado y Pruebas de Causalidad

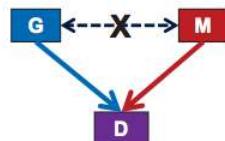
Una cuestión central para la genómica personal es la cuestión de qué marcadores son causales de enfermedad. Por ejemplo, uno podría preguntarse si la metilación en ciertos loci, o una cierta modificación de histonas, aumenta el riesgo de una persona de una determinada enfermedad. Esta pregunta es difícil porque necesitamos separar las correlaciones espurias de los efectos causales -por ejemplo, es posible que una mutación en otra parte del genoma cause la enfermedad, y también aumente la probabilidad de observar un marcador en particular, pero que el marcador no tenga ningún efecto causal sobre la enfermedad. En este caso, encontraríamos una correlación entre el fenotipo de la enfermedad y la presencia del marcador a pesar de la falta de algún efecto causal.

La visión clave que nos permite determinar los efectos causales, a diferencia de las meras correlaciones, es la observación de que si bien el genotipo puede influir en el riesgo de una persona de una enfermedad en particular, la enfermedad no modificará el genotipo de una persona. Esto nos permite utilizar el genotipo como variable instrumental para la metilación. Esto limita el número de modelos posibles para que podamos probar estadísticamente qué modelo es más consistente con los datos observados.

Existen tres posibilidades para modelar enfermedades humanas complejas: el modelo de asociaciones independientes, el modelo de interacción y el modelo de vía causal, representados en la Figura 33.4. Utilizaremos el ejemplo de estudiar la relación causal entre la metilación en ciertos loci y la enfermedad para demostrar cómo probar un efecto causal.

- **Three possible models:**

1. Independent Associations

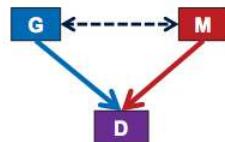


2. Causal Pathway Model



3. Interaction Model

- |          |             |
|----------|-------------|
| <b>G</b> | Genotype    |
| <b>M</b> | Methylation |
| <b>D</b> | Disease     |



fuente desconocida. Todos los derechos reservados. Este contenido está excluido de nuestro Creative Commons. Para obtener más información, consulte <http://ocw.mit.edu/help/faq-fair-use/>.

Figura 33.4: Modelado de enfermedades humanas

Bajo el modelo de asociaciones independientes, los datos no deben contener correlación entre el genotipo y la enfermedad, lo que distingue a este modelo de los modelos de interacción y vía causal. Sin embargo, habrá correlaciones entre cada uno de los factores y la enfermedad por separado. Por lo tanto, este modelo es sencillo de probar. Un ejemplo de esto serían dos genes de riesgo independientes.

Bajo el modelo de interacción, el efecto del factor B sobre una enfermedad puede variar dependiendo del valor de A. Por ejemplo, un efecto de drogas en alguien puede ser diferente según su genotipo. Para probar esto, determinaremos la significancia estadística del efecto del término de interacción,  $\beta_2$ , en la regresión  $D = \beta_0A + \beta_1B + \beta_2A * B$ . Si hay un efecto de interacción significativo, podemos aislar los efectos separados por estratificación a través de diferentes niveles de A.

El modelo de vía causal es un poco más complejo. Si observamos una correlación entre un factor de riesgo y una enfermedad, podemos preguntarnos si existe un vínculo directo entre el factor de riesgo A y una enfermedad, o si el factor de riesgo A afecta al factor de riesgo B que luego afecta a la enfermedad. En el caso de que el factor de riesgo A solo tenga un efecto sobre la enfermedad a través del B, observaremos que después de condicionar a B, la correlación entre A y D desaparece, es decir, B

“media” esta interacción. En realidad, el efecto de A sobre una enfermedad suele estar mediado solo parcialmente a través de B, por lo que podemos buscar si el tamaño del efecto de A sobre la enfermedad disminuye cuando se observa B.

## Predicción de riesgo poligénico

Una de las cuestiones más centrales de la genómica personal es la predicción de predisposiciones genéticas a diversos rasgos genéticos, utilizando múltiples genes para informar nuestras predicciones. El enfoque básico se explica en la Figura 33.5. Primero, el conjunto de datos se divide en un conjunto de entrenamiento y prueba, y en la cohorte de entrenamiento, seleccionamos qué SNPs son más importantes y sus ponderaciones adecuadas. Luego usamos el conjunto de pruebas para evaluar la precisión de nuestras predicciones. Finalmente, utilizamos este modelo para predecir predisposiciones genéticas para la cohorte objetivo mediante el uso de las confidencias que determinamos a partir del conjunto de pruebas.

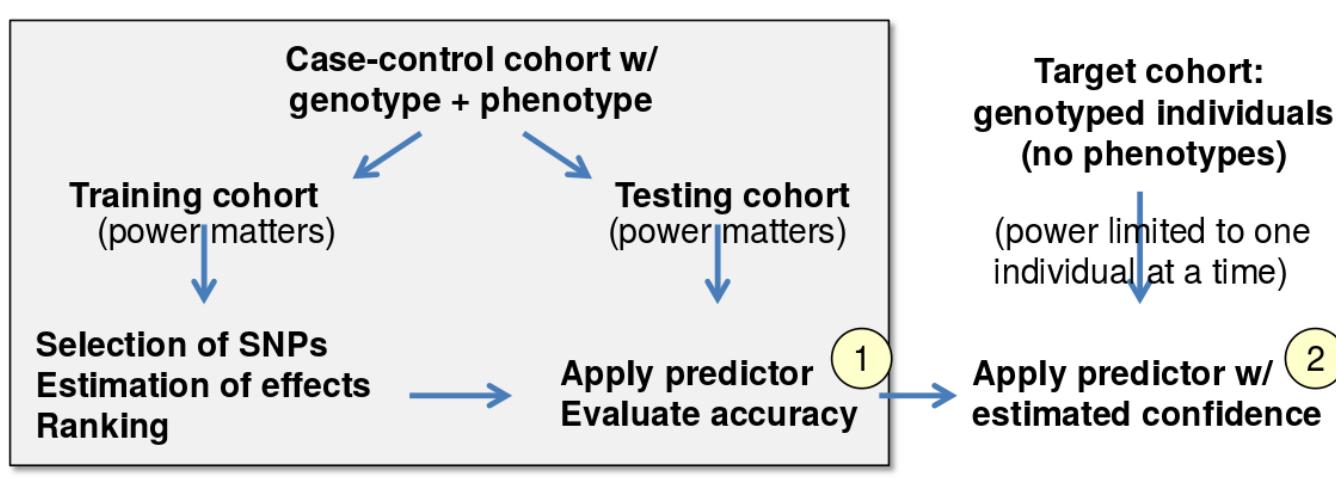


Figura 33.5: Predicción de riesgo poligénico

33.5: Modelado y Pruebas de Causalidad is shared under a not declared license and was authored, remixed, and/or curated by LibreTexts.

- [33.5: Causality Modeling and Testing](#) has no license indicated.

## 33.6: ¿Qué hemos aprendido?

En esta sección hemos aprendido sobre los fundamentos de la epidemiología, tanto genética como molecular. Hemos aprendido técnicas de diseño de un experimento epidemiológico y cómo y cuándo utilizar las pantallas genéticas para identificar enfermedades. Por último, nos enfocamos en resolver causalidad vs. correlación entre marcadores epigenéticos y enfermedades utilizando la genética como variable instrumento.

---

33.6: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [33.6: What Have We Learned?](#) has no license indicated.

## CHAPTER OVERVIEW

### 34: Genómica del Cáncer

[Sección 1: Introducción](#)

[Sección 2: Caracterización](#)

[Sección 3: Interpretación](#)

[Sección 5: Lectura adicional](#)

[Sección 6: ¿Qué hemos aprendido?](#)

---

34: Genómica del Cáncer is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## Sección 1: Introducción

¿Qué es el cáncer? El cáncer representa un grupo de enfermedades o tumores que desencadenan un crecimiento celular anormal y tienen el potencial de diseminarse a muchas partes del cuerpo. Un cáncer suele comenzar con mutaciones en uno o más “genes conductores” que son genes que pueden impulsar la tumorigénesis. Estas mutaciones se denominan eventos conductores, lo que significa que proporcionan una ventaja de aptitud selectiva para el individuo; otras mutaciones que no proporcionan ventajas de aptitud se denominan mutaciones pasajeras.

El objetivo principal de la genómica del cáncer es generar un catálogo integral de genes y caminos del cáncer. Muchos proyectos del genoma del cáncer se han iniciado en los últimos diez años (principalmente debido a la caída en los costos de secuenciación del genoma); por ejemplo, el Atlas del Genoma del Cáncer se inició en 2006 con el objetivo de analizar 20-25 tipos de tumores con 500 pares tumor/ normales cada uno a través de una gran cantidad de experimentos (matrices SNP, enteros- secuenciación de exomas, secuencia de ARN y otros). El ICGC (consorcio internacional del genoma del cáncer) es una organización paraguas más grande que organiza proyectos similares en todo el mundo con el objetivo final de estudiar 50 tipos de tumores con 500 tipos de tumor/normales cada uno.

---

Sección 1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 1: Introduction](#) has no license indicated.

## Sección 2: Caracterización

Para cada tumor, nuestro objetivo es obtener una caracterización completa y a nivel básico de ese tumor, su historia evolutiva y los mecanismos que lo configuraron. Podemos usar secuenciación masivamente paralela para obtener la caracterización del genoma a nivel base, pero este enfoque trae consigo algunos desafíos asociados.

1. Cantidad masivas de datos El principal desafío con el aumento de las cantidades de datos es un aumento en la potencia computacional requerida para analizar estos datos, así como los costos de almacenamiento asociados con el seguimiento de todos los genomas secuenciados. También se necesita un pipeline de análisis (automatizado, estandarizado, reproducible) para tener hallazgos consistentes en los diferentes esfuerzos de caracterización. Finalmente, necesitamos encontrar nuevas formas de visualizar y reportar datos a gran escala.
2. Sensibilidad/Especificidad La caracterización del cáncer comienza con la identificación adecuada de las mutaciones SNP presentes en las células cancerosas, y la eliminación máxima de lecturas falsas positivas. Al seleccionar muestras tumorales, el ADN extraído es una mezcla de genomas normales y genomas tumorales complejos. La fracción alélica mutacional (la fracción de moléculas de ADN de un locus que porta una mutación), se utiliza para estudiar la significación de una mutación y su prevalencia en el subtipo de cáncer. Esta fracción depende de la pureza, el número de copias locales, la multiplicidad de la muestra tumoral y la fracción de células cancerosas (CCF, cantidad de células cancerosas que portan la mutación). Las mutaciones clonales son portadas por todas las células cancerosas, y las mutaciones subcloniales son portadas por un subconjunto de las células tumorales.

Además de detectar la presencia de mutaciones clonales y subcloniales, el análisis adecuado requiere la eliminación de eventos mutagénicos falsos positivos. Dos tipos de falsos positivos incluyen errores de secuenciación y mutaciones de la línea germinal. Los errores de secuenciación pueden provenir de bases mal leídas, artefactos de secuenciación y lecturas desalineadas, mientras que las mutaciones de la línea germinal generalmente ocurren en lugares predecibles del genoma (1000/MB conocidos, novela 10-20/MB). Al tener múltiples lecturas de la misma secuencia, la probabilidad de errores repetidos en la secuenciación disminuye rápidamente, y al saber en qué parte del genoma es probable una mutación de la línea germinal, un filtro puede corregir la probabilidad de falsos positivos adicionales. La sensibilidad general de detectar variaciones de un solo nucleótido depende de la frecuencia de mutaciones de fondo y del número de lecturas alternativas.

Un tercer tipo de falso positivo puede provenir de la contaminación cruzada del paciente si la muestra del tumor contiene ADN de otra persona. ConTEST es un método para detectar con precisión la contaminación por comparación con una matriz SNP.

Un llamador de mutaciones es un clasificador preguntando en cada locus genómico, ¿Hay alguna mutación aquí?. Estos clasificadores se evalúan utilizando muchas curvas de Característica de Operadores Receptores (ROC), que dependen de la fracción alélica, cobertura de tumor y muestra normal, y ruido de secuenciación y alineación. MutECT es un llamador de mutación somática altamente sensible. El pipeline MutECT es el siguiente: Las muestras tumorales y normales se pasan a un estadístico de detección de variantes (que compara el modelo variante con la hipótesis nula), que se pasa a través de filtros basados en sitio (brecha proximal, sesgo de hebra, mapeo deficiente, sitio trialélico, posición agrupada, observada en control), luego se compararon con un panel de muestras normales, y finalmente se clasificaron como variantes candidatas. MutECT puede detectar mutaciones de fracción alélica baja y, por lo tanto, es adecuado para estudiar tumores impuros y heterogéneos.

### 3. Descubriendo procesos mutacionales

En lugar de detectar la presencia de mutaciones en los genes del cáncer, un enfoque diferente podría ser descubrir si hubiera patrones específicos entre mutaciones en las muestras de cáncer. Una “trama de Lego” es una forma de visualizar patrones de mutaciones, en la que las alturas de cada uno de los colores representan frecuencias de los 6 tipos de sustituciones de pares de bases, y la frecuencia de cada una se grafica en relación con los 16 contextos diferentes en los que podría ocurrir esta mutación (nucleótidos vecinos). Los tipos específicos de eventos mutagénicos en cada tipo de cáncer pueden representarse y analizarse. Como ejemplo, se encuentra un nuevo patrón de mutación (AA → AC) en el cáncer de esófago. Los cánceres pueden agruparse por estos espectros mutacionales específicos. Las reducciones de dimensionalidad usando factorización matricial no negativa (NMF) de datos de parcelas lego se pueden usar para identificar firmas espectrales fundamentales.

### 4. Estimación de la pureza, la ploidía y las funciones de las células cancerosas

Además de detectar mutaciones en células cancerosas, eliminar falsos positivos y detectar patrones de mutaciones, se requiere una caracterización adecuada de cada muestra tumoral. Debido a la heterogeneidad e impurezas de la muestra, es necesario estimar la pureza, el número absoluto de copias y la fracción de células cancerosas (CCF) de la muestra tumoral que se está secuenciando para obtener el número total correcto y la prevalencia de los alelos mutados.

## 5. Heterogeneidad y evolución tumoral

Las muestras pueden tener grandes distribuciones de mutaciones puntuales y alteraciones del número de copias, pero un algoritmo de agrupamiento bayesiano puede ayudar a identificar las mutaciones y alteraciones del número de copias en distintas subpoblaciones.

---

Sección 2: Caracterización is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 2: Characterization](#) has no license indicated.

## Sección 3: Interpretación

El desafío fundamental en la interpretación de los resultados de la secuenciación radica en diferenciar las mutaciones conductoras de las mutaciones pasajeras. Para lograr esto, necesitamos modelar los procesos mutacionales de fondo de las secuencias analizadas e identificar trayectorias/regiones con más mutaciones de las que habrían sido predichas únicamente por el modelo de fondo. Esas regiones se convierten entonces en nuestros genes candidatos al cáncer.

Sin embargo, nos encontramos con el problema potencial de seleccionar un modelo de fondo incorrecto o podemos encontrar artefactos sistemáticos en la llamada de mutaciones. En este caso, tenemos que volver al tablero de dibujo e intentar llegar a un mejor modelo de fondo antes de poder proceder con la identificación de genes candidatos.

Se han desarrollado muchas herramientas en un esfuerzo por detectar con precisión genes y vías de cáncer candidatos (subredes), incluyendo NetSIG, GISTIC y MutSIG. NetSIG se utiliza para identificar grupos de genes mutados en redes de interacción proteína-proteína. GISTIC se puede utilizar para puntuar regiones de acuerdo con la frecuencia y la frecuencia de los eventos de números de copia. MutSIG: se utiliza para puntuar genes de acuerdo al número y tipo de mutaciones. Los principales pasos de análisis para encontrar genes candidatos de cáncer son 1) estimación de la tasa de mutación de fondo (que varía entre muestras, 2) calcular valores p basados en modelos estadísticos y 3) corregir hipótesis de prueba múltiple (N genes).

A medida que aumenta el tamaño de la muestra y/o la tasa de mutación, la lista de genes significativos para los genes del cáncer aumenta y contiene muchos genes a pescado. Un gran avance para reducir los genes a pescado ha sido el modelado adecuado de las mutaciones de fondo. Las herramientas estándar utilizan una tasa de fondo consistente (tasas para CpG, C/G, A/T, indel) mientras ignoran la heterogeneidad entre muestras, contextos de secuencia adicionales y el genoma. Pero se encubrió que la tasa de mutación a través del cáncer varía 1000 veces, la tasa de mutación es menor en genes altamente expresados y la frecuencia de mutaciones somáticas se correlaciona con el tiempo de replicación del ADN. Hay más mutaciones en áreas del genoma que se replican más tarde que las que se dividen temprano. MutSigCV es una herramienta que corrige esta variación en las tasas de mutación de fondo.

---

Sección 3: Interpretación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 3: Interpretation](#) has no license indicated.

## Sección 5: Lectura adicional

1. <http://www.broadinstitute.org/cancer/cga/mutect>
2. <http://www.broadinstitute.org/cancer/cga/ABSOLUTE>

---

Sección 5: Lectura adicional is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 5: Further Reading](#) has no license indicated.

## Sección 6: ¿Qué hemos aprendido?

La caída en los costos de secuenciación en los últimos diez años ha llevado a la necesidad de tuberías de análisis automatizadas y más potencia computacional/de almacenamiento para manejar la gran avalancha de datos que se generan por una multitud de esfuerzos de secuenciación paralela. Dos tareas principales de los proyectos del genoma del cáncer en el futuro se pueden agrupar aproximadamente en dos áreas: caracterización e interpretación.

Para la caracterización, parece que aún se necesita un benchmark sistemático de métodos de análisis (un ejemplo son las curvas ROC, curvas que ilustran el desempeño de un clasificador con un umbral de discriminación variable). Vimos que las tasas de mutación del cáncer tienden a variar más de mil veces entre diferentes tipos de tumores. También aprendimos que las mutaciones clonales y subclonales podrían ser utilizadas para estudiar la evolución tumoral y la heterogeneidad.

Al ejecutar un análisis de significancia sobre los resultados de la secuenciación, se identificó una distribución de cola larga de genes mutados significativamente. Dado que estamos tratando con una distribución de cola larga, podemos aumentar el poder predictivo de nuestros modelos y detectar más genes de cáncer integrando múltiples fuentes de evidencia. Sin embargo, hay que tener en cuenta que las tasas de mutación difieren según la muestra original, el gen y la categoría de cada estudio.

---

Sección 6: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Section 6: What Have We Learned?](#) has no license indicated.

## CHAPTER OVERVIEW

### 35: Edición del genoma

- 1: Introducción
- 2: Direcciones actuales de investigación
- 3: ¿Qué hemos aprendido?

---

35: Edición del genoma is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## 1: Introducción

---

### ¿Qué es CRISPR/Cas?

El sistema CRISPR/Cas es el sistema inmune procariota. Cuando un virus u otro atacante extraño intenta infectar una célula procariota e inyectar su propio ADN en el genoma de un procariota, el sistema CRISPR/Cas del procariota es responsable de eliminar el ADN extraño. ¿Cómo hace esto? El sistema CRISPR/Cas tiene dos partes, CRISPR y Cas. La parte CRISPR (una matriz CRISPR), es responsable de “recordar” el ADN extraño, mientras que la parte Cas (proteínas Cas), es la encargada de cortar el ADN extraño reconocido. Una matriz CRISPR está compuesta por segmentos de ADN espaciador corto, que son el resultado de una exposición previa a ADN extraño. Estos ADN espaciadores se transcriben a ARN, los cuales pueden usarse para hacer coincidir el ADN extraño del que se construyó el ADN espaciador. Estos ARN son luego recogidos por las proteínas Cas. Cuando una proteína Cas recoge un ARN particular, se vuelve sensible a las secuencias de ADN coincidentes. La próxima vez que se inserte el mismo ADN extraño en el procariota, las proteínas Cas sensibles a él coincidirán con el ADN extraño y lo cortarán del genoma, haciendo que se vuelva inactivo.

### ¿Por qué el CRISPR/Cas es importante para nosotros?

¡Porque la naturaleza nos está dando una manera efectiva de editar un genoma! Para editar con precisión un genoma, es importante poder cortar una secuencia precisamente en la ubicación objetivo. Una vez que se realiza un corte, el mecanismo de reparación puede entrar y hacer una modificación en el sitio objetivo. El sistema CRISPR/Cas es un método natural probado en el tiempo para hacer alteraciones en las secuencias de ADN.

Actualmente, la capacidad de los investigadores para perturbar e interrogar al genoma está rezagada con respecto al nivel actual de técnicas de lectura. CRISPR proporciona una manera efectiva de escribir al genoma que somos capaces de leer, lo que nos permite determinar qué variaciones en el código genético dan lugar a enfermedades de interés.

### Cas-9

El sistema CRISPR/Cas-9 es un sistema que ha sido de particular interés. Cas-9 es una endonucleasa que puede desencadenar la reparación génica realizando cortes en sitios diana específicos, guiados por un ARNsg de 20 nucleótidos. Cuando se encuentra un sitio diana que es complementario al ARNsg guía y es seguido por una región PAM de NGG, la proteína Cas-9 cortará el ADN en ese sitio diana. Al programar Cas-9 con ARNsg específico, se puede programar para crear roturas bicanarias en dianas específicas, mientras que la región PAM juega un papel en la prevención de la focalización de su propio genoma. Se ha demostrado que Cas-9 es mucho más eficiente en la focalización que los métodos más establecidos. Desafortunadamente, un inconveniente de Cas-9 es que podría hacer cortes en sitios fuera de la diana que no son completamente complementarios a la guía de ARN, lo que lo convierte en un desafío para la edición precisa del genoma.

---

1: Introducción is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 1: Introduction has no license indicated.

## 2: Direcciones actuales de investigación

---

### Mejora de Cas-9

Investigaciones recientes han producido una variante de Cas-9 que mejora en gran medida la especificidad de Cas-9, reduciendo la probabilidad de errores externos.

### Investigación actual que se está realizando con CRISPR/CAS-9

La reciente mejora de Cas-9 ha abierto nuevas vías de investigación. Por ejemplo, se puede utilizar para analizar las funciones de genes específicos mediante el uso de CRISPR/Cas-9 para eliminar solo ese gen y observar el efecto de la eliminación. Un ejemplo de una aplicación de esto es en el estudio de células cancerosas de melanoma.

El vemurafenib es un medicamento aprobado por la FDA para tratar el melanoma, y se ha demostrado que es eficaz en células de melanoma que tienen una mutación V600E BRAF al interrumpir la vía BRAF e inducir la muerte celular programada.

Desafortunadamente, en muchos casos el cáncer se volverá resistente al medicamento al crear vías alternativas de supervivencia. Se puede usar CRISPR/Cas-9 para determinar los genes que permiten que las células cancerosas desarrollen vías alternativas. Al programar las proteínas Cas-9 para que se dirijan a cada gen individualmente y etiquetando las proteínas para que sea posible determinar qué proteína afectó a qué célula, es posible determinar los genes que se requieren para la supervivencia.

---

2: Direcciones actuales de investigación is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- 2: Current Research Directions has no license indicated.

### 3: ¿Qué hemos aprendido?

CRISPR/Cas-9 produce roturas de doble cadena en el ADN, y tiene dos componentes principales:

ADN de 20 pb

PAM

---

3: ¿Qué hemos aprendido? is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [3: What Have We Learned?](#) has no license indicated.

## CHAPTER OVERVIEW

### Volver Materia

[Índice](#)

[Índice](#)

[Glosario](#)

---

This page titled [Volver Materia](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al.](#) ([MIT OpenCourseWare](#)) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

## Índice

### D

dynamic programming

[2.4: Programación dinámica](#)

### E

expectation maximization algorithms

[15.3: Algoritmos de Clustering](#)

### F

fuzzy clustering

[15.3: Algoritmos de Clustering](#)

### G

generative model

[15.3: Algoritmos de Clustering](#)

### H

hidden Markov models

[7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#)

hierarchical clustering

[15.3: Algoritmos de Clustering](#)

### M

Markov Chains

[7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#)

Metabolic flux analysis (MFA)

[23.3: Análisis de Flujo Metabólico](#)

microRNA

[10.3: Origen y Funciones del ARN](#)

miRNA

[18.4: Genes y dianas de microARN](#)

### R

riboswitches

[10.3: Origen y Funciones del ARN](#)

RNA World Hypothesis

[10.3: Origen y Funciones del ARN](#)

This page titled [Índice](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Manolis Kellis et al. \(MIT OpenCourseWare\)](#) via source content that was edited to the style and standards of the LibreTexts platform.

## Índice

### D

dynamic programming

[2.4: Programación dinámica](#)

### E

expectation maximization algorithms

[15.3: Algoritmos de Clustering](#)

### F

fuzzy clustering

[15.3: Algoritmos de Clustering](#)

### G

generative model

[15.3: Algoritmos de Clustering](#)

### H

hidden Markov models

[7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#)

hierarchical clustering

[15.3: Algoritmos de Clustering](#)

### M

Markov Chains

[7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización](#)

Metabolic flux analysis (MFA)

[23.3: Análisis de Flujo Metabólico](#)

microRNA

[10.3: Origen y Funciones del ARN](#)

miRNA

[18.4: Genes y dianas de microARN](#)

### R

riboswitches

[10.3: Origen y Funciones del ARN](#)

RNA World Hypothesis

[10.3: Origen y Funciones del ARN](#)

---

Índice is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

## Glosario

Ejemplo y Direcciones

| (Opcional)   |  |  |                            |  |                           | Entradas en el glosario |                        |        |         |        |        |
|--|--|--|----------------------------|--|---------------------------|-------------------------|------------------------|--------|---------|--------|--------|
| Palabras (o palabras que tienen la misma definición) | La definición de las que se mencionan y el minús culas | definición de reconoce y se mayúsculas y el definición | (Opciónal) [No Leye image] | (Opciónal) Leye interna para extero o no o | (Opciónal) Enlace interno | Palabra (s)             | Definición             | Imagen | Leyenda | Enlace | Fuente |
| Palabras (o palabras que tienen la misma definición) | La definición de las que se mencionan y el minús culas | definición de reconoce y se mayúsculas y el definición | (Opciónal) [No Leye image] | (Opciónal) Leye interna para extero o no o | (Opciónal) Enlace interno | Palabra de muesta 1     | Definición de muesta 1 |        |         |        |        |
| (Ej. “Genético, Hereditario, ADN ...”)               | (Ej. “Relacionado con genes o herencia”)               |  |                            | La infame doble hélice                     | bio.librete xtso rg/      | CC-BY-SA; Delmar Larson |                        |        |         |        |        |

*Glosario* is shared under a [not declared](#) license and was authored, remixed, and/or curated by LibreTexts.

- [Glossary](#) has no license indicated.

# Index

## D

dynamic programming  
2.4: Programación dinámica

## E

expectation maximization algorithms  
15.3: Algoritmos de Clustering

## F

fuzzy clustering  
15.3: Algoritmos de Clustering

## G

generative model  
15.3: Algoritmos de Clustering

## H

hidden Markov models  
7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización  
hierarchical clustering  
15.3: Algoritmos de Clustering

## M

Markov Chains  
7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización

## Metabolic flux analysis (MFA)

23.3: Análisis de Flujo Metabólico

## microRNA

10.3: Origen y Funciones del ARN

## miRNA

18.4: Genes y dianas de microARN

## R

## riboswitches

10.3: Origen y Funciones del ARN

## RNA World Hypothesis

10.3: Origen y Funciones del ARN

## Glossary

---

**Sample Word 1** | Sample Definition 1

## Detailed Licensing

### Overview

**Title:** Libro: Biología Computacional - Genomas, Redes y Evolución (Kellis et al.)

**Webpages:** 308

**Applicable Restrictions:** Noncommercial

**All licenses found:**

- [CC BY-NC-SA 4.0](#): 57.1% (176 pages)
- [Undeclared](#): 42.9% (132 pages)

### By Page

- Libro: Biología Computacional - Genomas, Redes y Evolución (Kellis et al.) - [CC BY-NC-SA 4.0](#)
  - Front Matter - [Undeclared](#)
    - [TitlePage](#) - [Undeclared](#)
    - [InfoPage](#) - [Undeclared](#)
    - [Table of Contents](#) - [Undeclared](#)
    - [Licensing](#) - [Undeclared](#)
  - Materia Frontal - [CC BY-NC-SA 4.0](#)
    - [TitlePage](#) - [CC BY-NC-SA 4.0](#)
    - [InfoPage](#) - [CC BY-NC-SA 4.0](#)
    - [Tabla de Contenidos](#) - [Undeclared](#)
  - 1: Introducción al Curso - [CC BY-NC-SA 4.0](#)
    - 1.1: Introducción y Objetivos - [CC BY-NC-SA 4.0](#)
    - 1.2: Proyecto Final - Introducción a la Investigación en Biología Computacional - [CC BY-NC-SA 4.0](#)
    - 1.3: Materiales adicionales - [CC BY-NC-SA 4.0](#)
    - 1.4: Curso Crash en Biología Molecular - [CC BY-NC-SA 4.0](#)
    - 1.5: Introducción a algoritmos e inferencia probabilística - [CC BY-NC-SA 4.0](#)
  - 2: Alineación de Secuencias y Programación Dinámica - [CC BY-NC-SA 4.0](#)
    - 2.1: Introducción - [CC BY-NC-SA 4.0](#)
    - 2.2: Alineación de secuencias - [CC BY-NC-SA 4.0](#)
    - 2.3: Formulaciones de problemas - [CC BY-NC-SA 4.0](#)
    - 2.4: Programación dinámica - [CC BY-NC-SA 4.0](#)
    - 2.5: El algoritmo de Needleman-Wunsch - [CC BY-NC-SA 4.0](#)
    - 2.6: Alineación múltiple - [CC BY-NC-SA 4.0](#)
    - 2.7: Herramientas y Técnicas - [CC BY-NC-SA 4.0](#)
    - 2.8: Apéndice - [CC BY-NC-SA 4.0](#)
    - 2.9: Bibliografía - [CC BY-NC-SA 4.0](#)
  - 3: Alineación rápida de secuencias y búsqueda de bases de datos - [CC BY-NC-SA 4.0](#)
    - 3.1: ¿Qué hemos aprendido? - [Undeclared](#)
    - 3.2: Introducción - [CC BY-NC-SA 4.0](#)
    - 3.3: Alineación global vs. alineación local vs. alineación semi-global - [CC BY-NC-SA 4.0](#)
    - 3.4: Coincidencia exacta de cadenas en tiempo lineal - [CC BY-NC-SA 4.0](#)
    - 3.5: El algoritmo BLAST (Herramienta Básica de Búsqueda de Alineación Local) - [CC BY-NC-SA 4.0](#)
    - 3.6: Preprocesamiento para la coincidencia de cadenas en tiempo lineal - [CC BY-NC-SA 4.0](#)
    - 3.7: Fundamentos probabilísticos del alineamiento de secuencias - [CC BY-NC-SA 4.0](#)
  - 4: Genómica Comparada I- Anotación del Genoma - [CC BY-NC-SA 4.0](#)
    - 4.1: Introducción - [CC BY-NC-SA 4.0](#)
    - 4.2: Conservación de secuencias genómicas - [CC BY-NC-SA 4.0](#)
    - 4.3: Restricción en exceso - [CC BY-NC-SA 4.0](#)
    - 4.4: Diversidad de firmas evolutivas- Una visión general de los patrones de selección - [CC BY-NC-SA 4.0](#)
    - 4.5: Firmas de codificación de proteínas - [CC BY-NC-SA 4.0](#)
    - 4.6: Firmas génicas de microARN (miARN) - [CC BY-NC-SA 4.0](#)
    - 4.7: Motivos Regulatorios - [CC BY-NC-SA 4.0](#)
    - 4.8: Lectura adicional - [CC BY-NC-SA 4.0](#)
    - 4.9: Herramientas y Técnicas - [Undeclared](#)
    - Bibliografía - [Undeclared](#)
  - 5: Ensamblaje del Genoma y Alineación del Genoma - [CC BY-NC-SA 4.0](#)
    - 5.1: Introducción - [CC BY-NC-SA 4.0](#)
    - 5.2: Asamblea Genómica I- Superposición-Diseño-Enfoque de Consenso - [CC BY-NC-SA 4.0](#)

- 5.3: Ensamblaje Genómico II- Métodos de gráfico de cuerdas - [CC BY-NC-SA 4.0](#)
- 5.4: Alineación del Genoma Completo - [CC BY-NC-SA 4.0](#)
- 5.5: Alineación regional basada en genes - [CC BY-NC-SA 4.0](#)
- 5.6: Mecanismos de Evolución Genómica - [CC BY-NC-SA 4.0](#)
- 5.7: Duplicación del genoma completo - [CC BY-NC-SA 4.0](#)
- 5.8: Recursos adicionales y bibliografía - [CC BY-NC-SA 4.0](#)
- Bibliografía - [Undeclared](#)
- 6: Genómica Bacteriana—Evolución Molecular a Nivel de Ecosistemas - [CC BY-NC-SA 4.0](#)
  - 6.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 6.2: Estudio 1- Evolución de la vida en la tierra - [CC BY-NC-SA 4.0](#)
  - 6.3: Estudio 2- Estudio pediátrico de EII con Athos Boudvaros - [CC BY-NC-SA 4.0](#)
  - 6.4: Estudio 3- Proyecto de Ecología Gut Humana (HuGE) - [CC BY-NC-SA 4.0](#)
  - 6.5: Estudio 4- El microbioma como la conexión entre dieta y fenotipo - [CC BY-NC-SA 4.0](#)
  - 6.6: Estudio 5- Transferencia Génica Horizontal (HGT) entre grupos bacterianos y su efecto sobre la resistencia a antibióticos - [CC BY-NC-SA 4.0](#)
  - 6.7: Estudio 6- Identificación de factores de virulencia en Meningitis - [CC BY-NC-SA 4.0](#)
  - 6.08: Q - [Undeclared](#)
    - 6.8: Q/A - [CC BY-NC-SA 4.0](#)
  - 6.9: Direcciones actuales de investigación - [CC BY-NC-SA 4.0](#)
  - 6.10 Lectura adicional - [Undeclared](#)
  - 6.12 ¿Qué hemos aprendido? - [Undeclared](#)
  - Bibliografía - [Undeclared](#)
- 7: Modelos ocultos de Markov I - [CC BY-NC-SA 4.0](#)
  - 7.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 7.2: Motivación - [CC BY-NC-SA 4.0](#)
  - 7.3: Cadenas de Markov y HMMS - Del ejemplo a la formalización - [CC BY-NC-SA 4.0](#)
  - 7.4: Aplicar HMM al Mundo Real- Del Casino a la Biología - [CC BY-NC-SA 4.0](#)
  - 7.5: Ajustes algorítmicos para HMM - [CC BY-NC-SA 4.0](#)
  - 7.6: Una pregunta interesante- ¿Podemos incorporar la memoria en nuestro modelo? - [Undeclared](#)
  - 7.7: Lectura adicional, ¿qué hemos aprendido? - [Undeclared](#)
- 8: Modelos Ocultos de Markov II-Decodificación posterior y aprendizaje - [CC BY-NC-SA 4.0](#)
  - 8.1: Revisión de la conferencia anterior - [CC BY-NC-SA 4.0](#)
  - 8.2: Decodificación posterior - [CC BY-NC-SA 4.0](#)
  - 8.3: Memoria de codificación en un HMM- Detección de islas CpG - [CC BY-NC-SA 4.0](#)
  - 8.4: Aprendizaje - [CC BY-NC-SA 4.0](#)
  - 8.5: Uso de HMM para alinear secuencias con penalizaciones por hueco afín - [CC BY-NC-SA 4.0](#)
  - 8.6: Direcciones actuales de investigación, ¿qué hemos aprendido? , Bibliografía - [Undeclared](#)
  - 8.9 ¿Qué hemos aprendido? - [Undeclared](#)
  - Bibliografía - [Undeclared](#)
- 9: Identificación Génica- Estructura Génica, Semi-Markov, CRFS - [CC BY-NC-SA 4.0](#)
  - 9.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 9.2: Descripción general de los contenidos del capítulo - [CC BY-NC-SA 4.0](#)
  - 9.3: Genes eucariotas: una introducción - [CC BY-NC-SA 4.0](#)
  - 9.4: Supuestos para la identificación computacional de genes - [CC BY-NC-SA 4.0](#)
  - 9.5: Cadenas Ocultas de Markov - [CC BY-NC-SA 4.0](#)
  - 9.6: Campos aleatorios condicionales - [CC BY-NC-SA 4.0](#)
  - 9.7: Otros métodos - [CC BY-NC-SA 4.0](#)
  - 9.8: Conclusión, Bibliografía - [CC BY-NC-SA 4.0](#)
  - Bibliografía - [Undeclared](#)
- 10: Plegamiento de ARN - [CC BY-NC-SA 4.0](#)
  - 10.1: Motivación y Propósito - [CC BY-NC-SA 4.0](#)
  - 10.2: Química del ARN - [CC BY-NC-SA 4.0](#)
  - 10.3: Origen y Funciones del ARN - [CC BY-NC-SA 4.0](#)
  - 10.4: Estructura del ARN - [CC BY-NC-SA 4.0](#)
  - 10.5: Problema de plegamiento de ARN y enfoques - [CC BY-NC-SA 4.0](#)
  - 10.6: Evolución del ARN - [CC BY-NC-SA 4.0](#)
  - 10.7: Aproximación probabilística al problema del plegamiento del ARN - [CC BY-NC-SA 4.0](#)
  - 10.8: Temas avanzados, Resumen y puntos clave, Lectura adicional, Bibliografía - [CC BY-NC-SA 4.0](#)
- 11: Modificaciones de ARN - [CC BY-NC-SA 4.0](#)
  - 11.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 11.2: Regulación Postranscripcional - [CC BY-NC-SA 4.0](#)
  - 11.3: ¿Qué hemos aprendido? - [Undeclared](#)

- 12: ARN intergénicos grandes no codificantes - [CC BY-NC-SA 4.0](#)
  - 12.1: Bibliografía - [CC BY-NC-SA 4.0](#)
  - 12.2: Introducción - [CC BY-NC-SA 4.0](#)
  - 12.3: ARN no codificantes de plantas a mamíferos - [CC BY-NC-SA 4.0](#)
  - 12.4: Tema práctico- RNaseQ - [CC BY-NC-SA 4.0](#)
  - 12.5: ARN largos no codificantes en la regulación epigenética - [CC BY-NC-SA 4.0](#)
  - 12.6: ARN intergergénicos no codificantes: ¿faltan lincs en células madre o cancerosas? - [CC BY-NC-SA 4.0](#)
  - 12.7: Tecnologías- en el laboratorio húmedo, ¿cómo podemos encontrarlas? - [Undeclared](#)
- 13: ARN pequeño - [CC BY-NC-SA 4.0](#)
  - 13.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 13.2: Interferencia de ARN - [CC BY-NC-SA 4.0](#)
  - 13.3: Bibliografía - [CC BY-NC-SA 4.0](#)
- 14: Secuenciación de ARNm para análisis de expresión y descubrimiento de transcritos - [CC BY-NC-SA 4.0](#)
  - 14.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 14.2: Microarrays de expresión - [CC BY-NC-SA 4.0](#)
  - 14.3: La biología de la secuenciación de ARNm - [CC BY-NC-SA 4.0](#)
  - 14.4: Mapeo de Lectura - Alineación Espaciada de - [CC BY-NC-SA 4.0](#)
  - 14.5: Reconstrucción - [CC BY-NC-SA 4.0](#)
  - 14.6: Cuantificación - [CC BY-NC-SA 4.0](#)
- 15: Regulación Génica I - Agrupación de Expresión Génica - [CC BY-NC-SA 4.0](#)
  - 15.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 15.2: Métodos para medir la expresión génica - [CC BY-NC-SA 4.0](#)
  - 15.3: Algoritmos de Clustering - [CC BY-NC-SA 4.0](#)
  - 15.4: Direcciones actuales de investigación - [CC BY-NC-SA 4.0](#)
  - 15.5: Lectura adicional - [CC BY-NC-SA 4.0](#)
  - 15.6: Recursos - [CC BY-NC-SA 4.0](#)
  - 15.7: Qué hemos aprendido, Bibliografía - [CC BY-NC-SA 4.0](#)
- 16: Regulación Génica II - Clasificación - [CC BY-NC-SA 4.0](#)
  - 16.1: Introducción - [CC BY-NC-SA 4.0](#)
  - 16.2: Clasificación—Técnicas Bayesianas - [CC BY-NC-SA 4.0](#)
  - 16.3: Máquinas vectoriales de soporte de clasificación - [CC BY-NC-SA 4.0](#)
- 16.4: Clasificación Tumoral con SVMs - [CC BY-NC-SA 4.0](#)
- 16.5: Aprendizaje Semi-Supervisado - [CC BY-NC-SA 4.0](#)
- 16.6: Lectura adicional, Recursos, Bibliografía - [CC BY-NC-SA 4.0](#)
- 17: Motivos Regulatorios, Muestreo de Gibbs y EM - [CC BY-NC-SA 4.0](#)
  - 17.1: Representación de Motivos y Contenido de Información - [CC BY-NC-SA 4.0](#)
  - 17.2: Introducción a los motivos reguladores y la regulación génica - [CC BY-NC-SA 4.0](#)
  - 17.3: Maximización de expectativas - [CC BY-NC-SA 4.0](#)
  - 17.4: Muestreo de Gibbs- Muestra de distribución conjunta (M, Zij) - [CC BY-NC-SA 4.0](#)
  - 17.5: Descubrimiento del motivo de novo - [CC BY-NC-SA 4.0](#)
  - 17.6: Posiblemente cosas en desuso por debajo- - [CC BY-NC-SA 4.0](#)
  - 17.7: Comparando diferentes métodos - [CC BY-NC-SA 4.0](#)
  - 17.8: OOPS, ZOOPS, MTC - [CC BY-NC-SA 4.0](#)
  - 17.9: Ampliación del Enfoque EM - [CC BY-NC-SA 4.0](#)
- 18: Genómica Regulatoria - [CC BY-NC-SA 4.0](#)
  - 18.1: Introducción a la Genómica Regulatoria - [CC BY-NC-SA 4.0](#)
  - 18.2: Descubrimiento de Motivos De Novo - [CC BY-NC-SA 4.0](#)
  - 18.3: Predecir objetivos regulares - [CC BY-NC-SA 4.0](#)
  - 18.4: Genes y dianas de microARN - [CC BY-NC-SA 4.0](#)
- 19: Epigenómica - [Undeclared](#)
  - 19: Epigenómica/Estados de cromatina - [CC BY-NC-SA 4.0](#)
    - 19.1: Introducción - [CC BY-NC-SA 4.0](#)
    - 19.2: Información Epigenética en Nucleosomas - [CC BY-NC-SA 4.0](#)
    - 19.3: Ensayos Epigenómicos - [CC BY-NC-SA 4.0](#)
    - 19.4: Procesamiento primario de datos de ChIP - [CC BY-NC-SA 4.0](#)
    - 19.5: Anotar el genoma usando firmas de cromatina - [CC BY-NC-SA 4.0](#)
    - 19.6: Direcciones actuales de investigación - [CC BY-NC-SA 4.0](#)
    - 19.7: Lectura adicional, herramientas y técnicas - [CC BY-NC-SA 4.0](#)

- 19.8: ¿Qué hemos aprendido? , Bibliografía - *Undeclared*
- 20: Redes I- Inferencia, Estructura, Métodos Espectrales - *CC BY-NC-SA 4.0*
  - 20.1: Introducción - *CC BY-NC-SA 4.0*
  - 20.2: Medidas de Centralidad de Red - *CC BY-NC-SA 4.0*
  - 20.3: Revisión de álgebra lineal - *CC BY-NC-SA 4.0*
  - 20.4: Análisis de componentes principales dispersos - *CC BY-NC-SA 4.0*
  - 20.5: Comunidades y Módulos de Red - *CC BY-NC-SA 4.0*
  - 20.6: Núcleo de Difusión en Red - *CC BY-NC-SA 4.0*
  - 20.7: Redes neuronales - *CC BY-NC-SA 4.0*
  - 20.8: Temas abiertos y desafíos - *CC BY-NC-SA 4.0*
  - 20.9: Lectura adicional, ¿qué hemos aprendido? , Bibliografía - *Undeclared*
  - 20.10: ¿Qué hemos aprendido? - *Undeclared*
  - Bibliografía - *Undeclared*
- 21: Redes Regulatorias- Inferencia, Análisis, Aplicación - *CC BY-NC-SA 4.0*
  - 21.1: Redes Regulatorias- Inferencia, Análisis, Aplicación - *CC BY-NC-SA 4.0*
  - 21.2: Inferencia de estructura - *CC BY-NC-SA 4.0*
  - 21.3: Visión general de la tarea de aprendizaje PGM - *CC BY-NC-SA 4.0*
  - 21.4: Aplicación de Redes - *CC BY-NC-SA 4.0*
  - 21.5: Propiedades Estructurales de Redes - *CC BY-NC-SA 4.0*
  - 21.6: Clustering de redes, Bibliografía - *CC BY-NC-SA 4.0*
  - Bibliografía - *Undeclared*
- 22: Interacciones de cromatina - *CC BY-NC-SA 4.0*
  - 22.1: Introducción - *CC BY-NC-SA 4.0*
  - 22.2: Terminología relevante - *CC BY-NC-SA 4.0*
  - 22.3: Métodos moleculares para estudiar la organización del genoma nuclear - *CC BY-NC-SA 4.0*
  - 22.4: Mapeo de interacciones genoma-lámina nuclear (LADs) - *CC BY-NC-SA 4.0*
  - 22.5: Métodos Computacionales para Estudiar la Organización del Genoma Nuclear - *CC BY-NC-SA 4.0*
  - 22.6: Arquitectura de la Organización del Genoma - *CC BY-NC-SA 4.0*
  - 22.7: Comprensión mecanicista de la arquitectura del genoma - *CC BY-NC-SA 4.0*
  - 22.8: Direcciones actuales de investigación - *CC BY-NC-SA 4.0*
- 23: Introducción al Modelado Metabólico en Estado Estable - *CC BY-NC-SA 4.0*
  - 23.1: Introducción - *CC BY-NC-SA 4.0*
  - 23.2: Construcción de modelos - *CC BY-NC-SA 4.0*
  - 23.3: Análisis de Flujo Metabólico - *CC BY-NC-SA 4.0*
  - 23.4: Aplicaciones - *CC BY-NC-SA 4.0*
  - 23.5: Lectura adicional, Herramientas y Técnicas, Bibliografía - *Undeclared*
  - 23.6: Herramientas y Techniques - *Undeclared*
  - Bibliografía - *Undeclared*
- 24: El Proyecto Encode- Experimentación Sistemática y Genómica Integrativa - *Undeclared*
  - 24.1: Introducción - *Undeclared*
  - 24.2: Técnicas Experimentales - *Undeclared*
  - 24.3: Técnicas Computacionales - *Undeclared*
  - 24.4: Direcciones actuales de investigación - *Undeclared*
  - 24.5: Lectura adicional, Herramientas y técnicas, Bibliografía - *Undeclared*
  - 24.6: Herramientas y Técnicas - *Undeclared*
  - Bibliografía - *Undeclared*
  - Sección 7: ¿Qué hemos aprendido? - *Undeclared*
- 25: Biología Sintética - *Undeclared*
  - 25.1: Introducción a la Biología Sintética - *Undeclared*
  - 25.2: Direcciones actuales de investigación - *Undeclared*
  - 25.3: Herramientas y Técnicas - *Undeclared*
  - 25.4: ¿Qué hemos aprendido? , Bibliografía - *Undeclared*
  - Bibliografía - *Undeclared*
- 26: Evolución Molecular y Filogenética - *Undeclared*
  - 26.1: Introducción - *Undeclared*
  - 26.2: Fundamentos de la Filogenia - *Undeclared*
  - 26.3: Métodos basados en la distancia - *Undeclared*
  - 26.4: Métodos basados en caracteres - *Undeclared*
  - 26.5: Posibles cuestiones teóricas y prácticas con enfoque discutido - *Undeclared*
  - 26.6: Hacia el proyecto final - *Undeclared*
  - 26.7: ¿Qué hemos aprendido? - *Undeclared*
  - Bibliografía - *Undeclared*
- 27: Filogenómica II - *Undeclared*
  - 27.1: Introducción - *Undeclared*
  - 27.2: SPIDR - *Undeclared*
  - 27.3: Gráficas de Recombinación Ancestral - *Undeclared*
  - 27.4: Conclusión - *Undeclared*

- 27.05: Inferir ortológicos - *Undeclared*
  - 27.5: Inferir ortológicos/Parálogos, Duplicación y Pérdida Genética - *Undeclared*
  - 27.6: Reconstrucción - *Undeclared*
  - 27.7: Modelización de Frecuencias de Poblaciones y Alelos - *Undeclared*
  - 27.9 Lectura adicional - *Undeclared*
  - 27.10 ¿Qué hemos aprendido? - *Undeclared*
  - Bibliografía - *Undeclared*
- 28: Historia de la población - *Undeclared*
  - 28.1: Introducción - *Undeclared*
  - 28.2: Encuesta Rápida de Variación Genética Humana - *Undeclared*
  - 28.3: Flujo genético africano y europeo - *Undeclared*
  - 28.4: Flujo de genes en el subcontinente indio - *Undeclared*
  - 28.5: Flujo de genes entre poblaciones humanas arcaicas - *Undeclared*
  - 28.6: Herramientas y Técnicas - *Undeclared*
  - 28.7: Direcciones de investigación, lecturas adicionales, bibliografía - *Undeclared*
  - 28.8: Ascendencia Europea y Migraciones - *Undeclared*
- 29: Variación genética poblacional - *Undeclared*
  - 29.1: Introducción - *Undeclared*
  - 29.2: Conceptos básicos de selección de población - *Undeclared*
  - 29.3: Vinculación genética - *Undeclared*
  - 29.4: Selección natural - *Undeclared*
  - 29.5: Evolución Humana - *Undeclared*
  - 29.6: Investigación actual - *Undeclared*
  - 29.7: Lectura adicional - *Undeclared*
- 30: Genética médica: el pasado hasta el presente - *Undeclared*
  - 30.1: Bibliografía - *Undeclared*
  - 30.2: Introducción - *Undeclared*
  - 30.3: Objetivos de investigar las bases genéticas de la enfermedad - *Undeclared*
  - 30.4: Rasgos mendelianos - *Undeclared*
  - 30.5: Rasgos Complejos - *Undeclared*
  - 30.6: Estudios de Asociación en todo el genoma - *Undeclared*
  - 30.7: Direcciones actuales de investigación - *Undeclared*
  - 30.8: Herramientas y Técnicas - *Undeclared*
  - 30.9: ¿Qué hemos aprendido? - *Undeclared*
- 31: Variación 2- Mapeo cuantitativo de rasgos, EQTLs, Variación de Rasgo Molecular - *Undeclared*
  - 31.1: Introducción - *Undeclared*
  - 31.2: Conceptos básicos de eQTL - *Undeclared*
  - 31.3: Estructura de un estudio eQTL - *Undeclared*
  - 31.4: Direcciones actuales de investigación - *Undeclared*
  - 31.5: ¿Qué hemos aprendido? - *Undeclared*
  - 31.6: Lectura adicional - *Undeclared*
  - 31.7: Herramientas y Recursos - *Undeclared*
  - 31.8: Bibliografía - *Undeclared*
- 32: Genomas Personales, Genomas Sintéticos, Computación en C vs Si - *Undeclared*
  - 32.1: Introducción - *Undeclared*
  - 32.2: Genomas de Lectura y Escritura - *Undeclared*
  - 32.3: Genomas personales - *Undeclared*
  - 32.4: Lectura adicional - *Undeclared*
  - 32.5: Bibliografía - *Undeclared*
- 33: Genómica personal - *Undeclared*
  - 33.1: Introducción - *Undeclared*
  - 33.2: Epidemiología- Una visión general - *Undeclared*
  - 33.3: Epidemiología Genética - *Undeclared*
  - 33.4: Epidemiología Molecular - *Undeclared*
  - 33.5: Modelado y Pruebas de Causalidad - *Undeclared*
  - 33.6: ¿Qué hemos aprendido? - *Undeclared*
- 34: Genómica del Cáncer - *Undeclared*
  - Sección 1: Introducción - *Undeclared*
  - Sección 2: Caracterización - *Undeclared*
  - Sección 3: Interpretación - *Undeclared*
  - Sección 5: Lectura adicional - *Undeclared*
  - Sección 6: ¿Qué hemos aprendido? - *Undeclared*
- 35: Edición del genoma - *Undeclared*
  - 1: Introducción - *Undeclared*
  - 2: Direcciones actuales de investigación - *Undeclared*
  - 3: ¿Qué hemos aprendido? - *Undeclared*
- Back Matter - *Undeclared*
  - Index - *Undeclared*
  - Glossary - *Undeclared*
  - Detailed Licensing - *Undeclared*
- Volver Materia - CC BY-NC-SA 4.0
  - Índice - CC BY-NC-SA 4.0
  - Índice - *Undeclared*
  - Glosario - *Undeclared*