Práctica 4: Alineamiento múltiple de secuencias

Marta Cuevas Rodríguez

Técnicas y modelos algorítmicos
Universidad de Málaga

Junio 2025

Índice

1. Introducción

El análisis comparativo de secuencias de ADN es una herramienta fundamental en bioinformática para estudiar la evolución, las relaciones filogenéticas y la funcionalidad de los genomas. En esta práctica se han analizado las similitudes genéticas entre tres especies de la familia *Felidae*: el gato doméstico (*Felis catus*), el león (*Panthera leo*) y el tigre (*Panthera tigris*). Aunque estas especies comparten un ancestro común, sus trayectorias evolutivas han generado diferencias genómicas que pueden ser cuantificadas y comparadas mediante diversas técnicas bioinformáticas.

El objetivo principal de esta práctica es estimar la distancia genética entre estas especies empleando dos enfoques complementarios. En una primera fase, se utiliza la frecuencia de *k-mers* para obtener una medida rápida y sin alineamiento de la similitud global entre las secuencias. Posteriormente, se lleva a cabo un alineamiento múltiple de secuencias, seguido de la construcción de árboles guía (filogenéticos), lo cual permite identificar con mayor precisión las relaciones evolutivas entre los organismos analizados.

Esta doble aproximación permite comparar los resultados obtenidos por métodos basados en composición de secuencia frente a aquellos que consideran la posición y el orden de los nucleótidos, resaltando las ventajas y limitaciones de cada estrategia en el contexto del análisis filogenético.

2. Alineamiento múltiple de secuencias

El alineamiento múltiple de secuencias (MSA, por sus siglas en inglés) es una técnica fundamental en bioinformática que permite comparar simultáneamente tres o más secuencias biológicas, ya sean de ADN, ARN o proteínas. Su propósito es identificar regiones conservadas entre las secuencias, lo cual puede ofrecer información sobre funciones biológicas comunes, estructuras moleculares compartidas o relaciones evolutivas entre organismos.

En un alineamiento múltiple, cada secuencia es colocada en una matriz donde las posiciones similares —por ejemplo, nucleótidos o aminoácidos homólogos— se alinean en columnas. Esto permite detectar patrones de conservación o variación entre las secuencias analizadas.

El proceso general de un alineamiento múltiple suele seguir los siguientes pasos:

- **1. Cálculo de similitudes**: Se comparan todas las secuencias entre sí para determinar su grado de similitud o distancia.
- **2. Construcción de una guía**: A partir de la información de similitud, se genera una estructura jerárquica (como un árbol filogenético) que refleja la relación entre las secuencias.
- **3. Alineamiento progresivo**: Las secuencias se alinean de forma iterativa, comenzando por las más similares. A medida que avanza el proceso, se van incorporando las secuencias restantes, respetando la estructura del árbol guía.
- **4. Inserción de huecos**: Se introducen huecos (gaps) en las secuencias cuando es necesario para mantener el alineamiento global, preservando la correspondencia entre posiciones homólogas.

El resultado final es una matriz alineada donde se pueden observar claramente regiones conservadas entre las secuencias, así como las diferencias que podrían indicar eventos evolutivos como mutaciones, inserciones o deleciones. Este tipo de análisis es clave en estudios de filogenia, identificación de genes ortólogos y análisis funcional de secuencias.

3. Descarga de datos

Para realizar el análisis de k-mers y el alineamiento múltiple de secuencias, se han descargado secuencias genómicas representativas de tres especies del género *Felidae*: el gato doméstico, el león y el tigre.

Las secuencias fueron obtenidas de la base de datos del **NCBI** (National Center for Biotechnology Information) [5], una plataforma pública mantenida por los Institutos Nacionales de Salud de los Estados Unidos (NIH), que proporciona acceso a una amplia colección de bases de datos biomoleculares, incluyendo secuencias genómicas, artículos científicos, y recursos de análisis bioinformático. El NCBI es una fuente confiable y ampliamente utilizada en estudios genómicos y filogenéticos.

A continuación, se indican los identificadores y enlaces a las secuencias utilizadas:

- ***Felis catus*** (Gato doméstico):
  Se descargó la secuencia completa del ADN mitocondrial de *Felis catus* desde GenBank [1].
- ***Panthera leo*** (León):
  Se descargó la secuencia completa del ADN mitocondrial de *Panthera leo* (león africano) [2].
- ***Panthera tigris*** (Tigre):
  Se utilizó la secuencia completa del genoma mitocondrial de *Panthera tigris* [3].

4. Estimación de la distancia genética mediante frecuencias de k-mers

Una forma de evaluar la similitud o divergencia genética entre especies es a través del análisis de *k-mers*, que consiste en contar todas las subsecuencias contiguas de longitud k presentes en una secuencia de ADN. Este enfoque no requiere alineamientos exactos y permite comparar secuencias de forma cuantitativa, incluso si presentan variaciones o mutaciones.

En esta sección se analiza la frecuencia de k-mers en las secuencias genómicas de las tres especies comentadas anteriormente. Para cada especie, se utilizó la secuencia mitocondrial completa, cuyas longitudes son las siguientes:

- **Gato** (*Felis catus*): 17 009 pb
- **León** (*Panthera leo*): 16 723 pb
- **Tigre** (*Panthera tigris*): 16 783 pb

4.1. Análisis de la variación del parámetro k en las frecuencias de k-mers

Con el objetivo de observar cómo cambia el perfil de frecuencias al modificar el tamaño del k-mer, se calcularon las frecuencias de aparición de todos los k-mers posibles para los valores de $k = 2, 3, 4$ en las secuencias mitocondriales de las tres especies: gato (*Felis catus*), león (*Panthera leo*) y tigre (*Panthera tigris*).

```
def contar_kmers(secuencia, k):
    secuencia = secuencia.upper()
    total = len(secuencia) - k + 1
    kmers = [secuencia[i:i+k] for i in range(total)]
    conteo = Counter(kmers)
```

```
    # Normalizacion
    frecuencias = {kmer: conteo[kmer]/total for kmer in conteo}
    return frecuencias

ks = [2, 3, 4]
```

Listing 1: Frecuencia de kmers 2, 3 y 4 para las especies utilizadas

A continuación, se presenta un resumen estadístico que incluye el número total de k-mers distintos encontrados (`N_kmers`), la frecuencia máxima observada (`Max_Freq`) y la frecuencia media (`Mean_Freq`) para cada combinación de especie y valor de k:

k	Especie	N_kmers	Max_Freq	Mean_Freq
2	Gato	16	0.101423	0.062500
2	León	16	0.104413	0.062500
2	Tigre	16	0.105589	0.062500
3	Gato	64	0.032869	0.015625
3	León	64	0.034747	0.015625
3	Tigre	64	0.034503	0.015625
4	Gato	256	0.012113	0.003906
4	León	256	0.013577	0.003906
4	Tigre	256	0.013290	0.003906

Cuadro 1: Resumen de frecuencias de k-mers para distintos valores de k en cada especie.

Se puede observar que:

- El número de k-mers distintos (`N_kmers`) crece exponencialmente con el valor de k, siguiendo la relación teórica 4^k.
- La frecuencia máxima disminuye al aumentar k, debido a la mayor diversidad de combinaciones posibles y la consecuente dispersión de las frecuencias.
- La frecuencia media también disminuye con k, en concordancia con la normalización y el mayor número de combinaciones observadas.

Para visualizar las distribuciones completas de las frecuencias, se han generado mapas de calor (*heatmaps*) para cada valor de k. Estas representaciones permiten comparar visualmente las diferencias en la composición de k-mers entre especies.

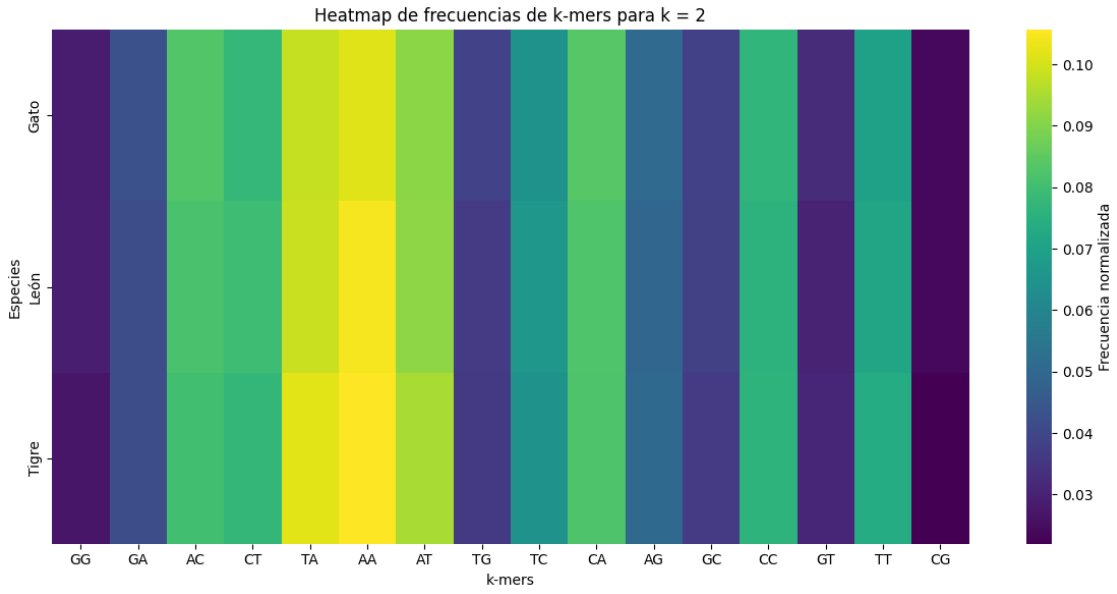

Figura 1: Mapa de calor de frecuencias para k=2.

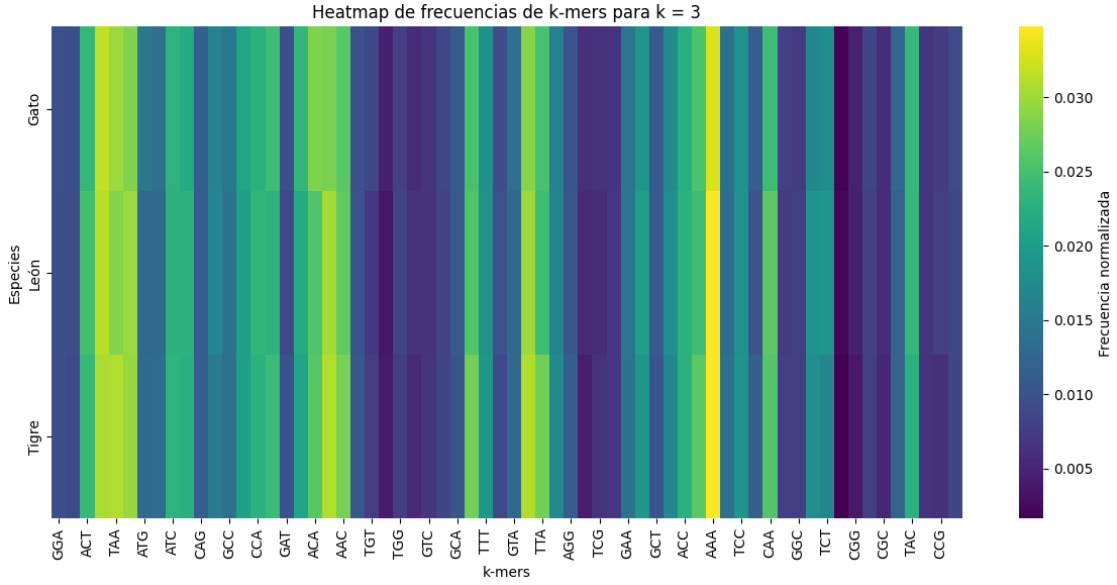

Figura 2: Mapa de calor de frecuencias para k=3.

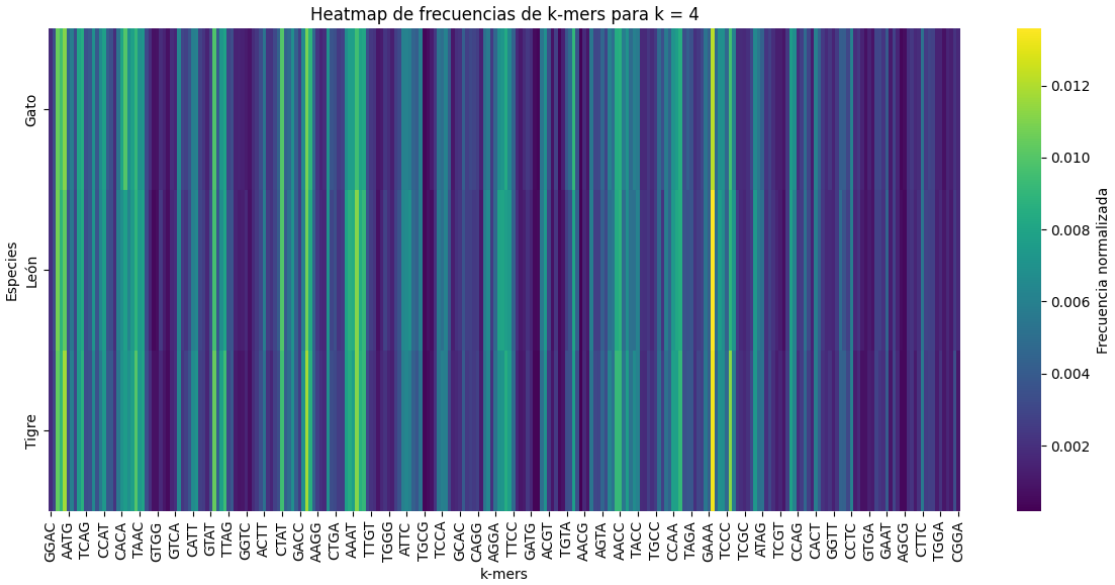

Figura 3: Mapa de calor de frecuencias para k=4.

Estos resultados permiten observar patrones conservados y diferencias específicas entre especies, que más adelante serán utilizados para estimar distancias genéticas.

4.2. Frecuencias de dímeros ($k = 2$)

En este apartado se estudian en detalle las frecuencias de aparición de todos los posibles dímeros (secuencias de dos nucleótidos consecutivos) en las secuencias mitocondriales de las tres especies analizadas. Para ello, se extrae la tabla de frecuencias correspondiente a $k = 2$, generando una matriz con las especies en las filas y los distintos dímeros en las columnas.

```
df_k2 = pd.DataFrame(resultados[2]).fillna(0).T  # filas: especie, columnas: k-
    mers
print(df_k2.head(10))
```

Listing 2: Cálculo de la matriz de frecuencias de 2-mers (dímeros).

El resultado obtenido es el siguiente:

Especie	GG	GA	AC	CT	TA	AA	AT
Gato	0.028575	0.042627	0.082843	0.077728	0.098189	0.101423	0.091075
León	0.028944	0.041502	0.081569	0.079656	0.098672	0.104413	0.091496
Tigre	0.026755	0.041473	0.080384	0.077404	0.101716	0.105589	0.094566

Especie	TG	TC	CA	AG	GC	CC	GT
Gato	0.038394	0.064440	0.083667	0.050564	0.037982	0.076611	0.032279
León	0.036359	0.066140	0.082586	0.049695	0.038213	0.075709	0.030080
Tigre	0.036348	0.064653	0.082112	0.050352	0.036408	0.075974	0.030807

Especie	TT	CG
Gato	0.069732	0.023871
León	0.071224	0.023741
Tigre	0.073472	0.021988

Cuadro 2: Frecuencias absolutas de dímeros ($k = 2$).

Como se observa, existen patrones de frecuencia bastante similares entre las tres especies, lo cual es esperable debido a su cercanía filogenética dentro del género *Felidae*. Sin embargo, se pueden apreciar ligeras diferencias, por ejemplo en la frecuencia del dímero `AA`, que es más alta en el tigre que en el gato.

4.3. Normalización de frecuencias con corrección por longitud

Con el objetivo de comparar de manera justa las frecuencias de k-mers entre organismos con secuencias de diferente longitud, es necesario aplicar una normalización. Para ello, se utiliza:

$$N = L - k - 1$$

donde L es la longitud total de la secuencia y k el tamaño del k-mer. Esta corrección elimina el sesgo introducido por la longitud de las secuencias, asegurando que las comparaciones se basen exclusivamente en la distribución relativa de los k-mers.

A continuación, se muestra el código utilizado para normalizar las frecuencias de los dímeros ($k = 2$):

```python
def contar_kmers_normalizados(secuencia, k):
    secuencia = secuencia.upper()
    L = len(secuencia)
    N = L - k - 1
    kmers = [secuencia[i:i+k] for i in range(L - k + 1)]
    conteo = Counter(kmers)
    frecuencias_norm = {kmer: conteo[kmer]/N for kmer in conteo}
    return frecuencias_norm
```

Listing 3: Normalización de frecuencias de 2-mers utilizando $N = L - k - 1$

El resultado de la normalización se presenta en la siguiente tabla. Cada valor representa la frecuencia relativa de un dímero en la secuencia mitocondrial correspondiente, corregida por la longitud total de la secuencia:

Como puede observarse, las frecuencias ya presentaban valores similares antes de la normalización, pero esta corrección asegura que las comparaciones posteriores —como las distancias genéticas— se realicen en igualdad de condiciones, eliminando el posible efecto de la longitud de cada genoma.

Especie	GG	GA	AC	CT	TA	AA	AT
Gato	0.028575	0.042627	0.082843	0.077728	0.098189	0.101423	0.091075
León	0.028944	0.041502	0.081569	0.079656	0.098672	0.104413	0.091496
Tigre	0.026755	0.041473	0.080384	0.077404	0.101716	0.105589	0.094566

Especie	TG	TC	CA	AG	GC	CC	GT
Gato	0.038394	0.064440	0.083667	0.050564	0.037982	0.076611	0.032279
León	0.036359	0.066140	0.082586	0.049695	0.038213	0.075709	0.030080
Tigre	0.036348	0.064653	0.082112	0.050352	0.036408	0.075974	0.030807

Especie	TT	CG
Gato	0.069732	0.023871
León	0.071224	0.023741
Tigre	0.073472	0.021988

Cuadro 3: Frecuencias normalizadas de dímeros para cada especie con $N = L - k - 1$ (sin redondeo).

4.4. Visualización y almacenamiento de resultados

Una vez normalizadas las frecuencias de k-mers, se guardan los resultados y para facilitar su análisis comparativo entre especies se representan de diferentes formas. Estas representaciones permiten detectar patrones recurrentes y diferencias sutiles entre los genomas de los organismos del género *Felidae*.

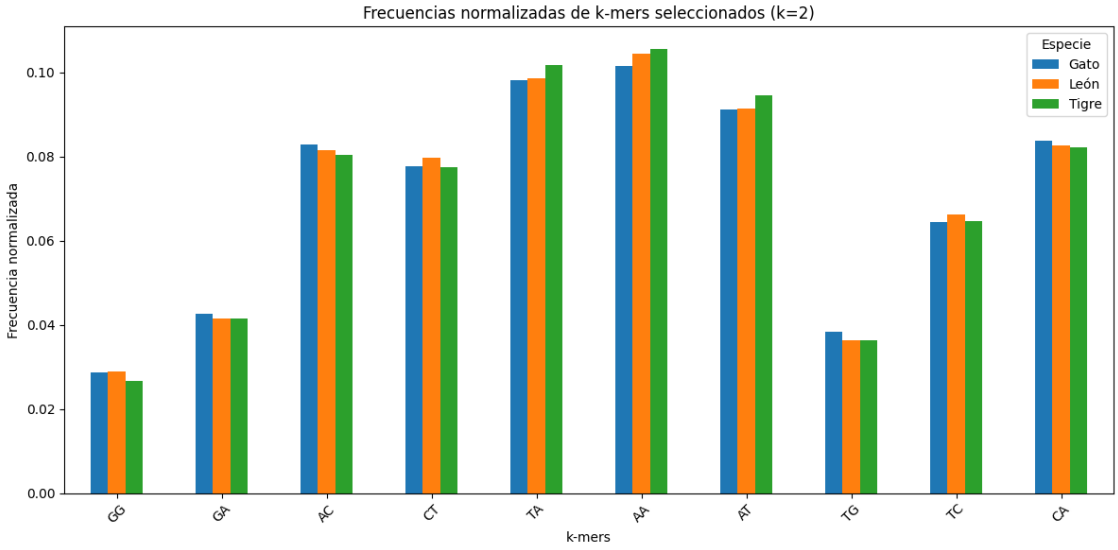

Figura 4: Frecuencias normalizadas de los k-mers más representativos ($k = 2$) en las tres especies.

En la gráfica de barras (Figura 4) se muestran las frecuencias normalizadas para un subconjunto representativo de los dímeros ($k = 2$) en las secuencias del gato, león y tigre. Podemos observar cómo ciertos patrones se conservan entre las tres especies —por ejemplo, los k-mers `AA`, `TA` y `AT` presentan frecuencias consistentemente altas—, lo que sugiere una posible conservación funcional o estructural en sus secuencias genómicas.

Por otro lado, se aprecian ligeras diferencias en la abundancia relativa de algunos dímeros. Por ejemplo, el k-mer `TA` aparece con mayor frecuencia en el tigre que en el gato o el león, mientras que `CG` (no mostrado en este gráfico) es típicamente bajo en todos los organismos, lo cual concuerda con la conocida menor abundancia

de dinucleótidos `CG` en genomas de vertebrados.

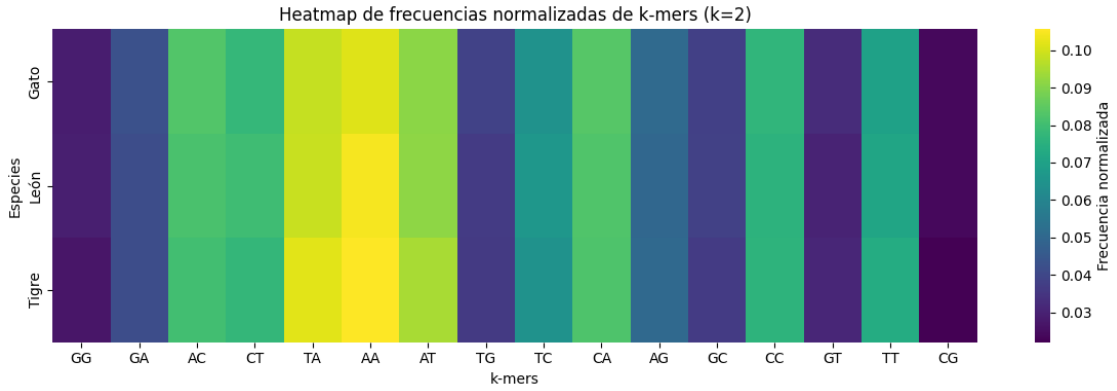

Figura 5: Mapa de calor de frecuencias normalizadas de todos los k-mers ($k = 2$) en las especies analizadas.

El mapa de calor complementa la gráfica de barras al permitir observar simultáneamente todos los k-mers y detectar agrupaciones o anomalías visuales. Los colores más intensos representan frecuencias más altas. Como se observa, la distribución es en gran medida similar entre especies, lo que es esperable dada su cercanía evolutiva.

4.5. Medidas de distancia y similitud entre especies

Para cuantificar la similitud entre las especies a partir de las frecuencias normalizadas de dímeros ($k = 2$), se calculan varias métricas de distancia y correlación: la distancia Manhattan, la distancia Euclídea y el coeficiente de correlación de Pearson. Estas métricas permiten evaluar en qué medida los patrones de k-mers difieren entre los genomas de las especies analizadas.

Cuadro 4: Distancias y correlaciones entre especies basadas en las frecuencias de k-mers ($k = 2$).

Especie 1	**Especie 2**	**Distancia Manhattan**	**Distancia Euclídea**	**Coef. Correlación (Pearson)**
Gato	León	0.019232	0.005745	0.998783
Gato	Tigre	0.030276	0.009060	0.998021
León	Tigre	0.022375	0.006797	0.998487

4.6. Visualización de distancias y correlaciones entre especies

Los resultados obtenidos de las métricas de distancia y correlación se almacenan en un fichero de salida para su análisis posterior.

Para facilitar la comparación entre pares de especies, se generaron representaciones visuales de los resultados obtenidos. A continuación, se muestran dos gráficas de barras: una correspondiente al coeficiente de correlación de Pearson (Figura 6) y otras que representan la distancia de Manhattan (Figura 7) y Euclídea (8). Estas gráficas permiten identificar de forma clara qué tan similares son las frecuencias de dímeros entre las especies analizadas.

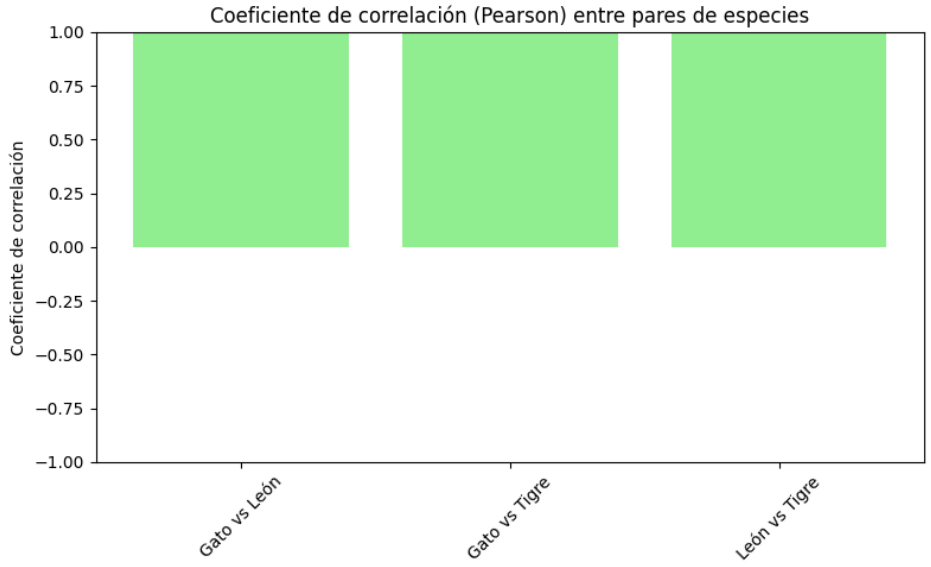

Figura 6: Coeficientes de correlación de Pearson entre pares de especies para $k = 2$.

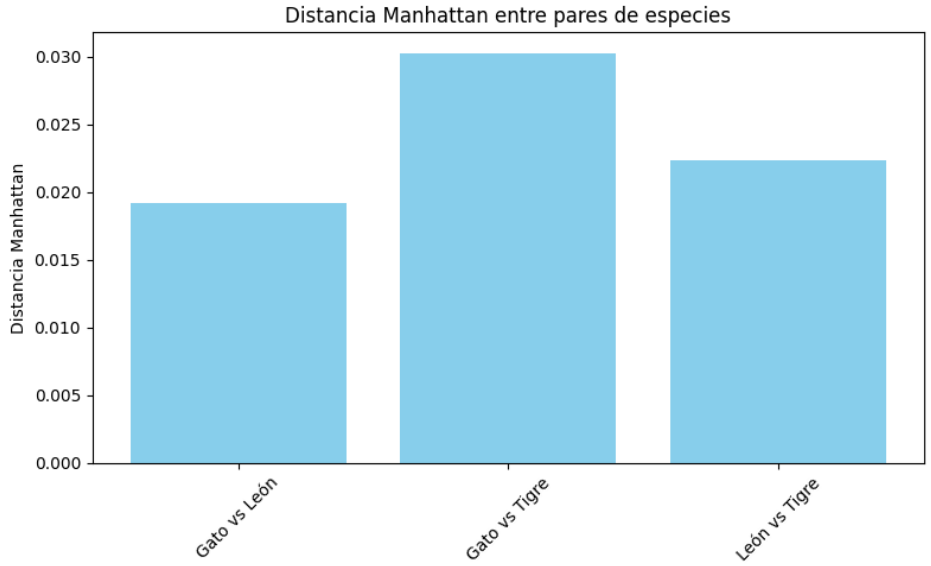

Figura 7: Distancia Manhattan entre pares de especies para $k = 2$.

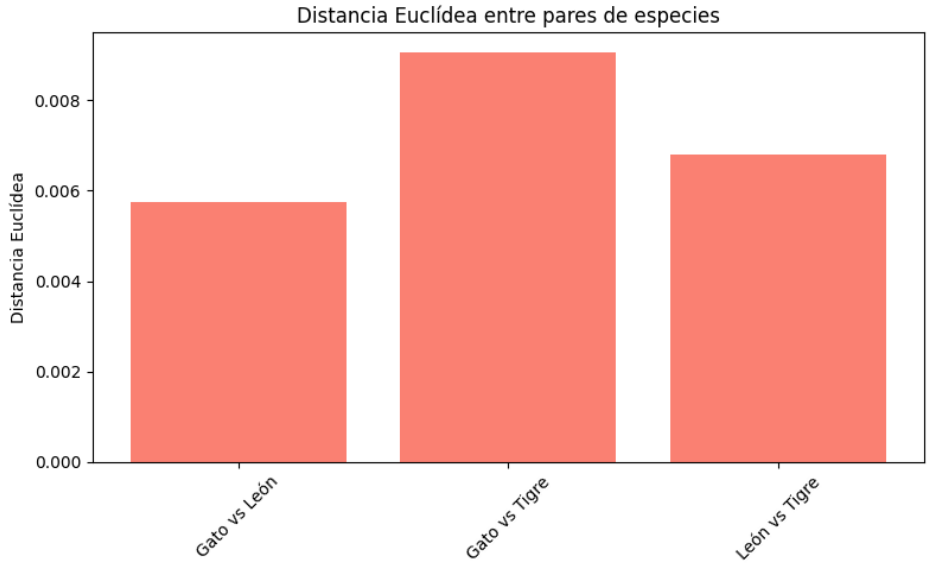

Figura 8: Distancia Euclídea entre pares de especies para $k = 2$.

Los resultados muestran que las distancias entre especies son muy pequeñas, lo cual es consistente con la cercanía evolutiva entre el gato, el león y el tigre. La menor distancia (tanto Manhattan como Euclídea) se observa entre el gato y el león, mientras que el par con mayor distancia es el formado por el gato y el tigre.

El coeficiente de correlación de Pearson también respalda esta similitud, con valores cercanos a 1 en todos los pares comparados. Esto indica que las frecuencias de los dímeros no sólo son similares en magnitud, sino también en su distribución relativa, lo que sugiere una fuerte conservación de los patrones de secuencia a nivel de dímeros entre estas especies.

5. Alineamiento múltiple de secuencias (MSA)

Para llevar a cabo este alineamiento múltiple, se utilizó el programa **Clustal Omega (ClustalO)** [4], una herramienta moderna y eficiente diseñada para el alineamiento múltiple de secuencias genómicas o proteicas. Clustal Omega utiliza una metodología progresiva optimizada y es capaz de alinear miles de secuencias de forma rápida y precisa.

El software puede ser instalado en sistemas Unix con el siguiente comando:

```
sudo apt install clustalo
```

Antes de realizar el alineamiento, se preparó un único archivo FASTA combinando las secuencias de las tres especies (gato, león y tigre). Esto se logró utilizando el siguiente comando en terminal:

```
cat sequencegato.fasta sequenceleon.fasta sequencetigre.fasta > felinos.fasta
```

El archivo resultante `felinos.fasta` contiene las tres secuencias en formato FASTA, listas para ser alineadas con Clustal Omega. Este paso asegura que todas las secuencias estén en un solo archivo, facilitando el análisis conjunto.

5.1. Alineamiento múltiple de secuencias

El primer paso en la construcción del alineamiento múltiple de secuencias (MSA) consiste en alinear simultáneamente todas las secuencias seleccionadas. Para ello, se utilizó la herramienta `Clustal Omega`, que permite generar alineamientos eficientes y escalables incluso en conjuntos de datos con múltiples secuencias.

Una vez unidas las secuencias de los tres felinos (*Felis catus*, *Panthera leo*, *Panthera tigris*) en un único archivo `felinos.fasta`, se ejecutó el siguiente comando para realizar el alineamiento:

```
clustalo -i felinos.fasta -o alineamiento.aln --outfmt=clu --force
```

El parámetro `-i` indica el archivo de entrada con las secuencias, `-o` el archivo de salida con el alineamiento, `-outfmt=clu` especifica el formato de salida (Clustal), y `-force` permite sobrescribir el archivo de salida si ya existe.

A continuación se muestra un fragmento del inicio del archivo `alineamiento.aln`, que contiene el alineamiento generado:

```
CLUSTAL O(1.2.4) multiple sequence alignment

U20753.1        ----GGACTAATGACTAATCAGCCCATGATCACACATAACTGTGGTGTCATGCATTTGGT
KP202270.1      CGATGGACTAATGACTAATCAGCCCATGATCACACATAACTGTGGTGTCATGCATTTGGT
KP202271.1      CGATGGATTAATGACTAATCAGCCCATGATCACACATAACTGTGGTGTCATGCATTTGGT
                    *** **************************************************

U20753.1        ATTTTTTATTTTTAGGGGGTCGAACTTGCTATGACTCAGCTATGACCTAAAGGTCCTGAC
KP202270.1      ATCTTTTAATTTTAGGGGGTCGAACTTGCTATGACTCAGCTATGACCTAAAGGTCCTGAC
```

Como se observa, el alineamiento muestra regiones conservadas indicadas con asteriscos (`*`) debajo de las columnas, reflejando la coincidencia de nucleótidos entre las secuencias. Esta representación permite detectar similitudes evolutivas entre las especies analizadas.

5.2. Construcción de un árbol guía a partir del alineamiento

Una vez generado el alineamiento múltiple, se procede a construir un árbol guía que representa la relación evolutiva entre las secuencias. Este árbol se utiliza para determinar el orden en que deben alinearse progresivamente las secuencias más similares entre sí, en conformidad con los métodos progresivos como el de Clustal.

Para ello, se ha utilizado la matriz de distancias obtenida a partir del alineamiento en formato `Clustal`. Las distancias se calcularon utilizando la identidad entre secuencias (proporción de posiciones idénticas).

	Gato	**León**	**Tigre**
Gato	0.000000		
León	0.108586	0.000000	
Tigre	0.114635	0.104769	0.000000

Cuadro 5: Matriz de distancias genéticas basada en identidad de secuencia entre Gato, León y Tigre.

Esta matriz muestra la divergencia entre pares de secuencias. Por ejemplo, entre `Gato` y `León` existe una divergencia del 10.8 %.

A partir de esta matriz, se construyeron dos árboles guía usando métodos distintos:

- **UPGMA (Unweighted Pair Group Method with Arithmetic Mean)**: Método jerárquico que asume una tasa de evolución constante.
- **Neighbor-Joining (NJ)**: Método que no asume tasas constantes de evolución y es más adecuado para conjuntos de datos con variaciones evolutivas.

```
# Construccion de arboles guia
constructor = DistanceTreeConstructor()
tree_upgma = constructor.upgma(dm)
tree_nj = constructor.nj(dm)
```

Listing 4: Construcción de los árboles guía utilizando UPGMA y Neighbor-Joining.

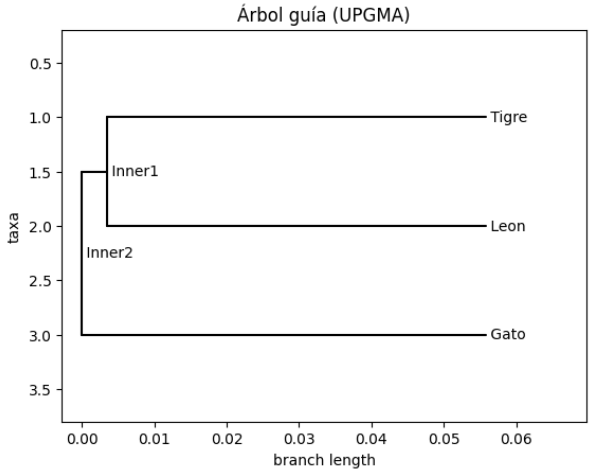

Figura 9: Árbol guía construidos a partir de la matriz de distancias UPGMA.

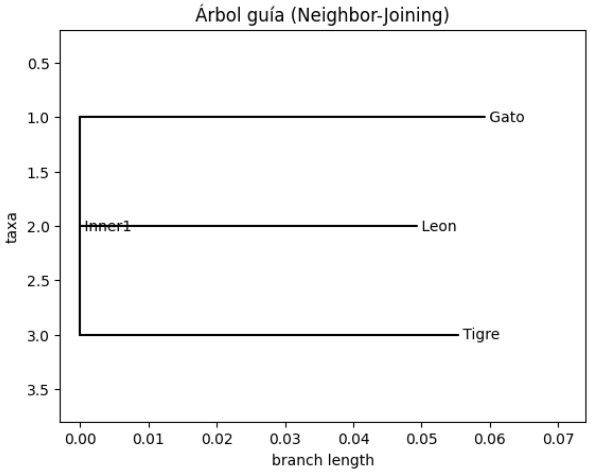

Figura 10: Árbol guía construido a partir de la matriz de distancias Neighbor-Joining.

Ambos árboles muestran que las secuencias `León` y `Tigre` están más estrechamente relacionadas entre sí, lo cual se refleja en su menor distancia evolutiva. La secuencia `Gato` se encuentra más distante de ambas, lo que sugiere una mayor divergencia.

En términos filogenéticos, esto puede interpretarse como una separación evolutiva más reciente entre `León` y `Tigre`, en comparación con su ancestro común con `Gato`. Este resultado es consistente en ambos métodos (UPGMA y NJ), lo que refuerza la confiabilidad del análisis.

5.3. Alineamiento progresivo utilizando el árbol guía

Una vez construido el árbol guía, se alinean las secuencias siguiendo el orden jerárquico de semejanza evolutiva. Este enfoque, conocido como alineamiento progresivo, comienza alineando las secuencias más estrechamente relacionadas entre sí y luego incorpora progresivamente las más distantes. Para ello, es necesario insertar huecos en las posiciones necesarias para mantener la coherencia del alineamiento múltiple.

Configuración del alineador

Para realizar el alineamiento entre secuencias, se utiliza el módulo `PairwiseAligner` de Biopython en modo global. Se ajustan los parámetros del alineador para reflejar penalizaciones y recompensas razonables:

```
# Configurar alineador
aligner = PairwiseAligner()
aligner.mode = 'global'
aligner.open_gap_score = -2
aligner.extend_gap_score = -1
aligner.match_score = 1
aligner.mismatch_score = -1
```

Listing 5: Configuración del alineador global

Alineamiento progresivo según el árbol guía

Para alinear recursivamente las secuencias en función del árbol guía previamente generado, el algoritmo comienza en las hojas del árbol y asciende hasta la raíz, alineando pares de secuencias o alineamientos previamente combinados. En cada paso se calcula una secuencia consenso para representar el nodo actual.

```
# ------------- Alineamiento recursivo usando  rbol  gu a -------------
def alinear_nodos(node):
    if node.left is None and node.right is None:
        node.seq = str(secuencias[node.name].seq)
        return node.seq

    seq_left = alinear_nodos(node.left)
    seq_right = alinear_nodos(node.right)

    alignment = aligner.align(seq_left, seq_right)[0]
    aligned_left, aligned_right = reconstruir_alineacion(alignment)

    # Guardar alineaciones en hijos
    node.left.seq = aligned_left
    node.right.seq = aligned_right

    # Crear secuencia consenso para nodo actual
    aln = MultipleSeqAlignment([
        SeqRecord(Seq(aligned_left), id="left"),
        SeqRecord(Seq(aligned_right), id="right")
    ])
    consensus = AlignInfo.SummaryInfo(aln).dumb_consensus()
    node.seq = str(consensus)

    return node.seq
```

Listing 6: Alineamiento recursivo basado en el árbol guía

Como resultado del proceso de alineamiento progresivo guiado por el árbol, se obtiene un archivo que contiene las tres secuencias alineadas. A continuación, se muestra un fragmento representativo de dicho alineamiento múltiple:

```
--- Alineamiento final ---

>Gato
CACGTACACACGTACACACGTACACACGTACACACGTACACACGTACACACGTACACACGTACACACGTACACACGTACACA
>Leon
--CGT--ACACGT--ACACGT--ACACGTACACACGTACACACGTATACACGT--ACACGT--ACACGTACACACGTA------TACAC-
>Tigre
CACGT--ACACGT--ACACGT--ACACGT--ACACGT--ACACGT--ACACGT--ACACGTACACACGTACACACGTACACACGTAC---
```

Conclusión

En este trabajo se han aplicado dos enfoques distintos para estimar la relación genética entre tres especies: el gato, el león y el tigre. En primer lugar, se estimaron las distancias genéticas a partir de las frecuencias de *kmers*, lo cual permitió obtener una medida rápida y global de similitud entre las secuencias sin necesidad de alinearlas. Este método sugirió que el gato y el león presentaban una mayor cercanía genética entre sí, comparado con la relación entre el león y el tigre.

Sin embargo, al realizar un alineamiento múltiple de las secuencias —utilizando un enfoque progresivo guiado por un árbol filogenético basado en la identidad de secuencias— se observó un resultado distinto: el león y el tigre aparecen como las especies más estrechamente relacionadas, mientras que el gato se muestra más divergente respecto a ambas. Este resultado concuerda mejor con la evidencia evolutiva conocida, ya que el león y el tigre pertenecen al mismo género (*Panthera*), mientras que el gato doméstico pertenece a otro género distinto (*Felis*).

La discrepancia entre ambos métodos puede explicarse por la naturaleza de las aproximaciones utilizadas. El análisis basado en *kmers* es sensible a la composición global de secuencia, pero no considera el orden ni la posición relativa de los nucleótidos, lo cual puede llevar a inferencias erróneas en casos donde hay regiones conservadas pero dispuestas de forma diferente. En cambio, el alineamiento múltiple proporciona una visión más detallada y estructurada de la similitud entre secuencias, reflejando con mayor precisión las relaciones filogenéticas reales.

En conclusión, aunque el análisis por *kmers* puede ser útil como una primera aproximación rápida, los resultados obtenidos mediante alineamiento múltiple son más fiables y consistentes con el conocimiento biológico y evolutivo actual.

Referencias

[1] NCBI GenBank. Felis catus mitochondrial dna sequence u20753.1. https://www.ncbi.nlm.nih.gov/nuccore/U20753.1, 2024.

[2] NCBI GenBank. Panthera leo mitochondrial dna sequence kp202270.1. https://www.ncbi.nlm.nih.gov/nuccore/KP202270.1, 2024.

[3] NCBI GenBank. Panthera tigris mitochondrial dna sequence kp202271.1. https://www.ncbi.nlm.nih.gov/nuccore/KP202271.1, 2024.

[4] European Bioinformatics Institute. Clustal omega. https://www.ebi.ac.uk/Tools/msa/clustalo/, 2024. Accedido el 5 de junio de 2025.

[5] National Center for Biotechnology Information. Ncbi - national center for biotechnology information, 2024. Accessed: 2025-05-27.