

digitalDLSorteR: paquete de R para la deconvolución de muestras bulk RNA-seq basado en Redes Neuronales.

Trabajo Fin de Máster

Máster en Bioinformática y Biología Computacional

Diego Mañanes Cayero

Universidad Autónoma de Madrid
Curso 2019-2020



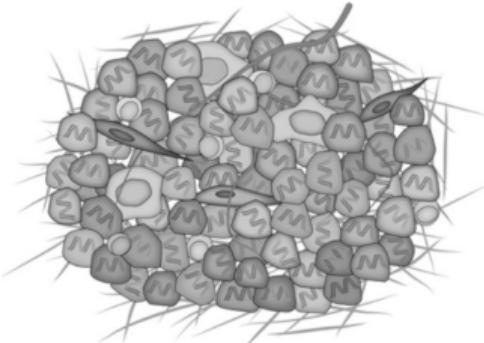
Universidad Autónoma
de Madrid



Contenidos

1. Introducción y contexto
 - 1.1. Cáncer y heterogeneidad celular
 - 1.2. Tecnologías para explorar la heterogeneidad celular: *scRNA-seq*
 - 1.3. Métodos de deconvolución de muestras *bulk RNA-seq*
2. Objetivos
3. digitalDLSorter: transformación de la *pipeline* en paquete de R
 - 2.1. Fundamento del método
 - 2.1. digitalDLSorter: flujo de trabajo
4. PhenoGraph
 1. Construcción del grafo
 2. Partición del grafo en comunidades
5. Resultados
6. Otros métodos

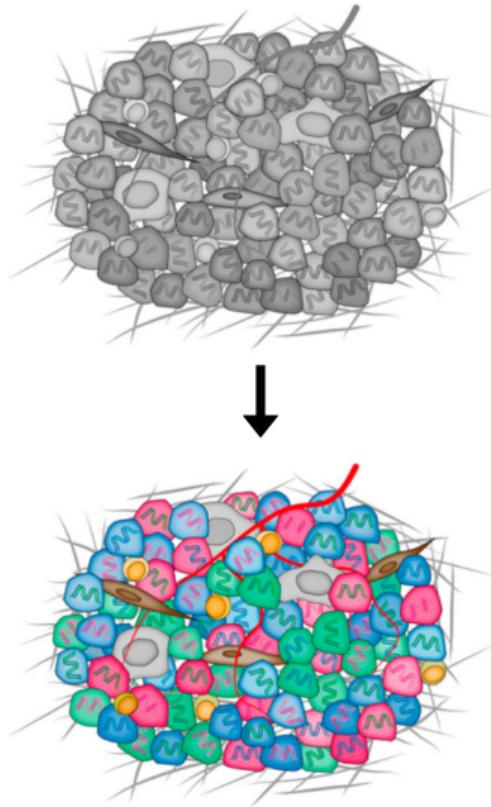
Introducción y contexto



Por qué

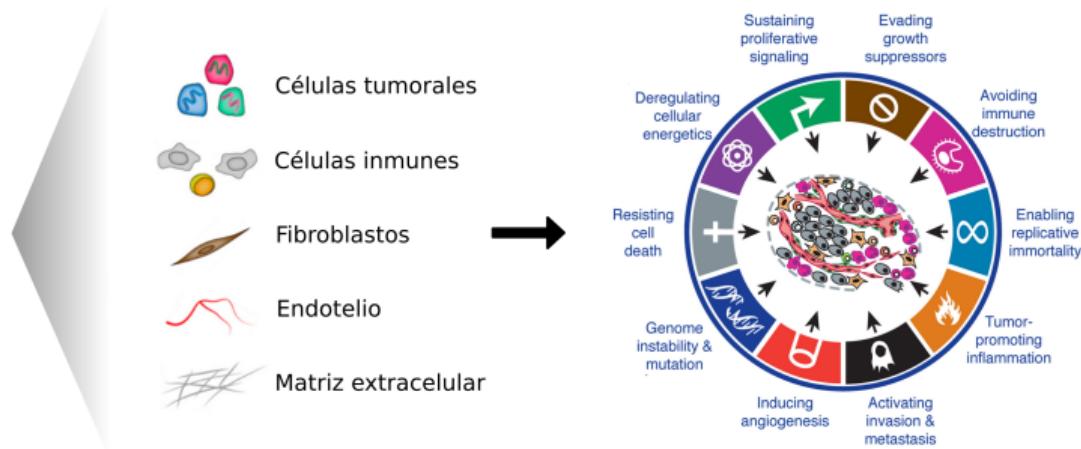
- Diferentes poblaciones tumorales.
- Micro-entorno tumoral: diferentes tipos celulares con complejas interacciones.
- Sello de Identidad del cáncer.

Cáncer y micro-entorno tumoral

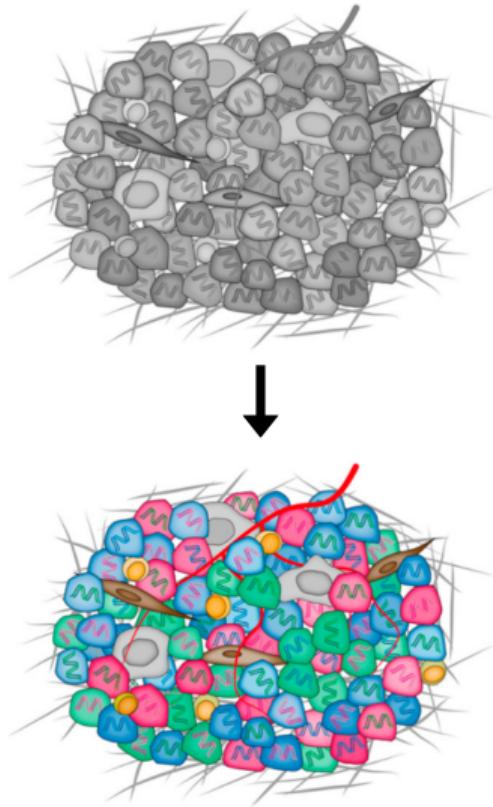


Por qué

- Diferentes poblaciones tumorales.
- Micro-entorno tumoral: diferentes tipos celulares con complejas interacciones.
- Sello de Identidad del cáncer.

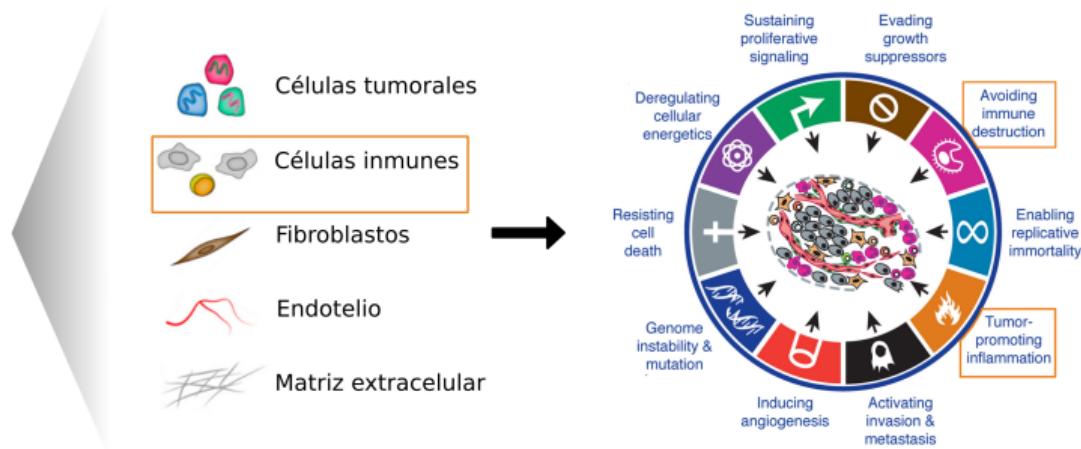


Cáncer y micro-entorno tumoral



Por qué

- Diferentes poblaciones tumorales.
- Micro-entorno tumoral: diferentes tipos celulares con complejas interacciones.
- Sello de Identidad del cáncer.



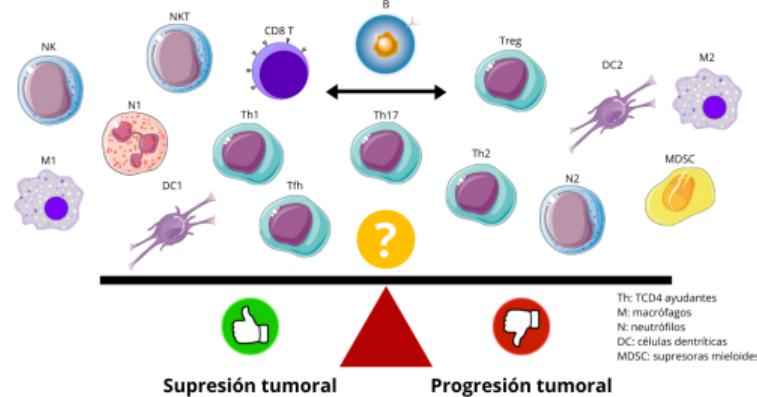
Papel del sistema inmune en la enfermedad

Papel clave

- La similitud entre células será máxima cuando sus k -vecinos son los mismos → mismo fenotipo.
- Similitud decaerá entre células que comparten menos vecinos conectados → distinto fenotipo.

Terapias

- La similitud entre células será máxima cuando sus k -vecinos son los mismos → mismo fenotipo.
- Similitud decaerá entre células que comparten menos vecinos conectados → distinto fenotipo.



Caso de estudio: cáncer de mama

Enfermedad altamente **heterogénea desde el punto de vista molecular**.

Subtipos intrínsecos del cáncer

- Luminal A (ER+).
- Luminal B (ER+/HER2+).
- HER2 enriquecido (HER2+).
- Triple negativo (TNBC).

Subtipo cáncer	Luminal A	Luminal B	HER2 enriquecido	Basal o triple negativo (TNBC)
% de cánceres de mama	50%	25%	15%	10%
Fenotipo	ER+ PR+ HER2-	ER+ PR+ HER2+	ER- PR- HER2+	ER- PR- HER2-
Prognosis				
Valor de TILs en la prognosis				
Tratamiento				

Importancia de las células inmunes infiltradas

- Luminal A (ER+).
- Luminal B (ER+/HER2+).
- HER2 enriquecido (HER2+).
- Triple negativo (TNBC).

Estudio de la heterogeneidad celular

Necesidad de métodos para el estudio de la heterogeneidad celular.

Tradicionalmente

A nivel de proteína mediante técnicas inmunohistoquímicas, inmunofluorescencia y citometría de flujo.

Pequeña combinación de marcadores génicos

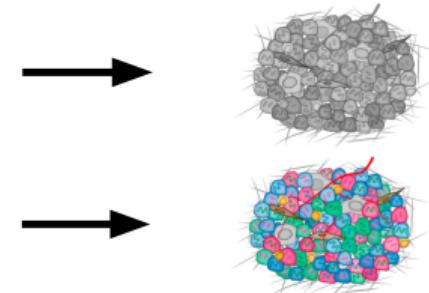
Tecnologías de alto rendimiento

A nivel transcriptómico mediante NGS: *RNA-seq*.

Estatus funcional completo

Dos variantes:

- *Bulk RNA-seq* (nivel tisular): los niveles de expresión corresponden al sumatorio de tipos celulares presentes en las muestras.
- *Single-cell RNA-seq* (nivel celular): los niveles de expresión corresponden a cada célula individual que compone la muestra.



scRNA-seq: Ventajas e inconvenientes

Ventajas

- Estatus funcional de cada célula.
- Caracterización de las poblaciones celulares. Potencial identificación de tipos celulares no esperados.
- Potencial aplicación traslacional: mejora del diagnóstico, terapias dirigidas, etc.

Inconvenientes

- Altos costes económicos y protocolos complejos.
- No es práctica su aplicación en grandes cohortes de muestras.
- Baja eficiencia de captura de ARN.
- Mayor ruido técnico.

scRNA-seq: Ventajas e inconvenientes

Ventajas

- Estatus funcional de cada célula.
- Caracterización de las poblaciones celulares. Potencial identificación de tipos celulares no esperados.
- Potencial aplicación traslacional: mejora del diagnóstico, terapias dirigidas, etc.

Inconvenientes

- Altos costes económicos y protocolos complejos.
- No es práctica su aplicación en grandes cohortes de muestras.
- Baja eficiencia de captura de ARN.
- Mayor ruido técnico.

Resultado:

Bulk RNA-seq sigue siendo el estándar.

scRNA-seq: Ventajas e inconvenientes

Ventajas

- Estatus funcional de cada célula.
- Caracterización de las poblaciones celulares. Potencial identificación de tipos celulares no esperados.
- Potencial aplicación translacional: mejora del diagnóstico, terapias dirigidas, etc.

Inconvenientes

- Altos costes económicos y protocolos complejos.
- No es práctica su aplicación en grandes cohortes de muestras.
- Baja eficiencia de captura de ARN.
- Mayor ruido técnico.

Resultado:

Bulk RNA-seq sigue siendo el estándar.

Problema: No tiene en cuenta en qué proporción contribuye cada tipo celular a los niveles de expresión medidos.

scRNA-seq: Ventajas e inconvenientes

Ventajas

- Estatus funcional de cada célula.
- Caracterización de las poblaciones celulares. Potencial identificación de tipos celulares no esperados.
- Potencial aplicación traslacional: mejora del diagnóstico, terapias dirigidas, etc.

Inconvenientes

- Altos costes económicos y protocolos complejos.
- No es práctica su aplicación en grandes cohortes de muestras.
- Baja eficiencia de captura de ARN.
- Mayor ruido técnico.

Resultado:

Bulk RNA-seq sigue siendo el estándar.

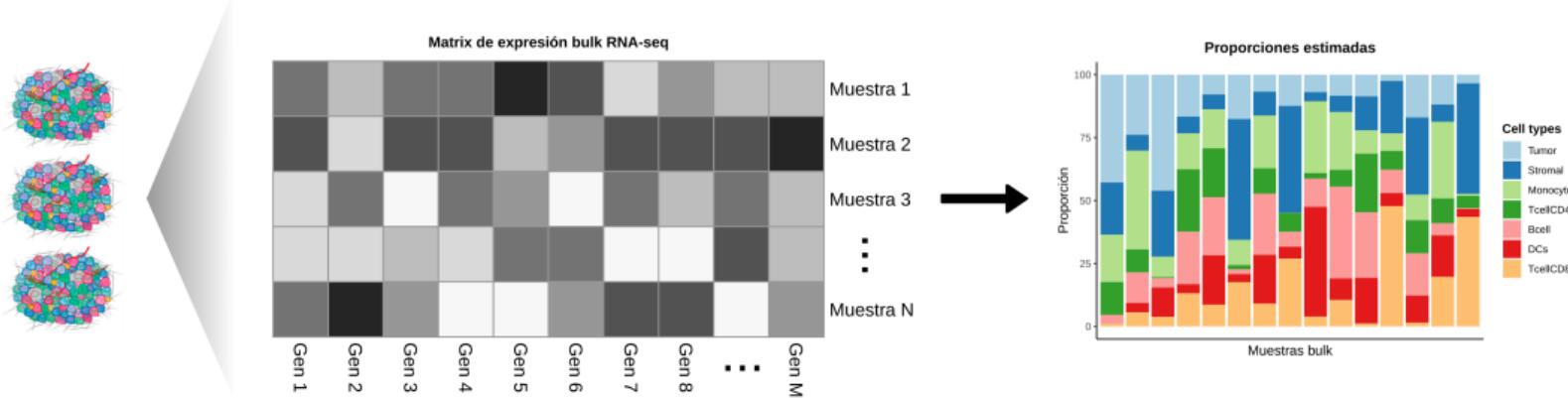
Problema: No tiene en cuenta en qué proporción contribuye cada tipo celular a los niveles de expresión medidos.

Necesidad: Métodos computacionales que permitan estimar las proporciones de cada tipo celular medidas en muestras *bulk RNA-seq*.

Deconvolución de muestras *bulk RNA-seq*

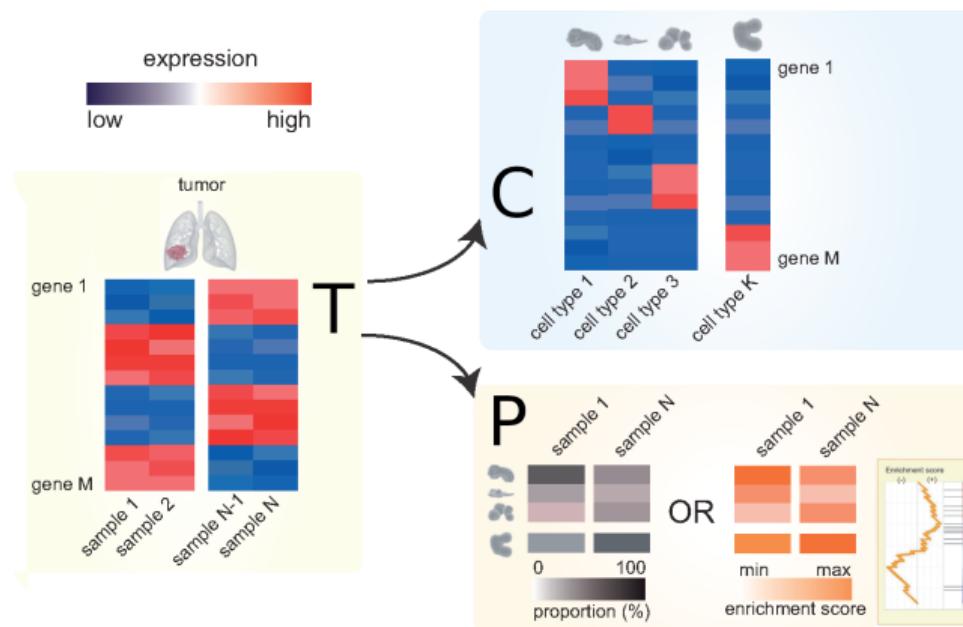
Deconvolución: estimación de la señal individual de cada uno de los componentes (tipos celulares) a partir de una mezcla de los mismos (muestra tisular).

- Sustituto de experimentos *scRNA-seq* por sus altos costes económicos o la imposibilidad de su aplicación.
- Control de la contribución de cada tipo celular a las muestras tisulares → evitar factores de confusión en análisis de expresión diferencial.



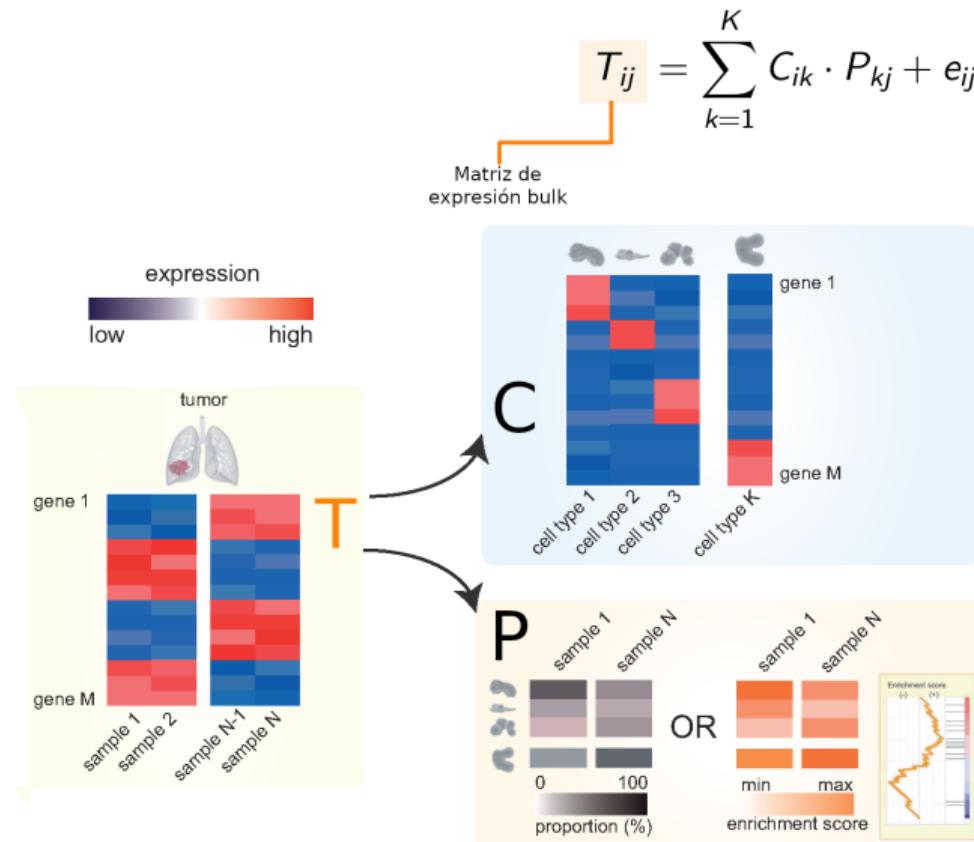
Deconvolución: marco de trabajo

$$T_{ij} = \sum_{k=1}^K C_{ik} \cdot P_{kj} + e_{ij}$$



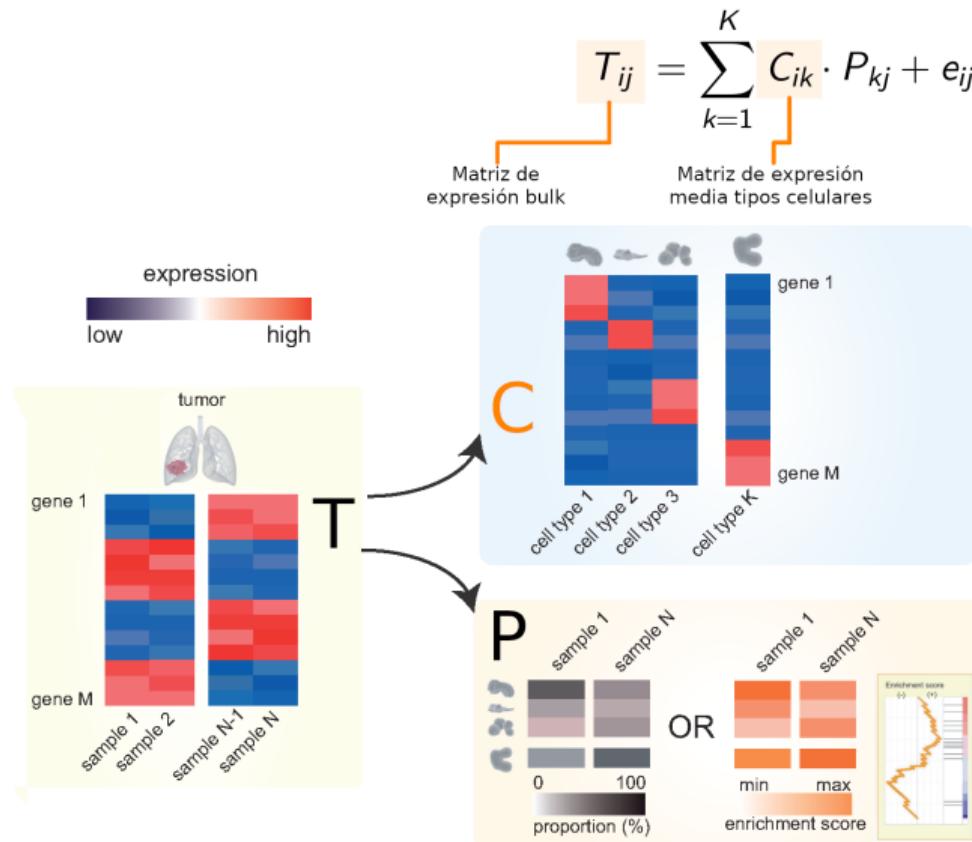
- i : genes ($i = 1 \dots M$).
- j : muestras *bulk* ($j = 1 \dots N$).
- k : tipos celulares ($k = 1 \dots K$).

Deconvolución: marco de trabajo



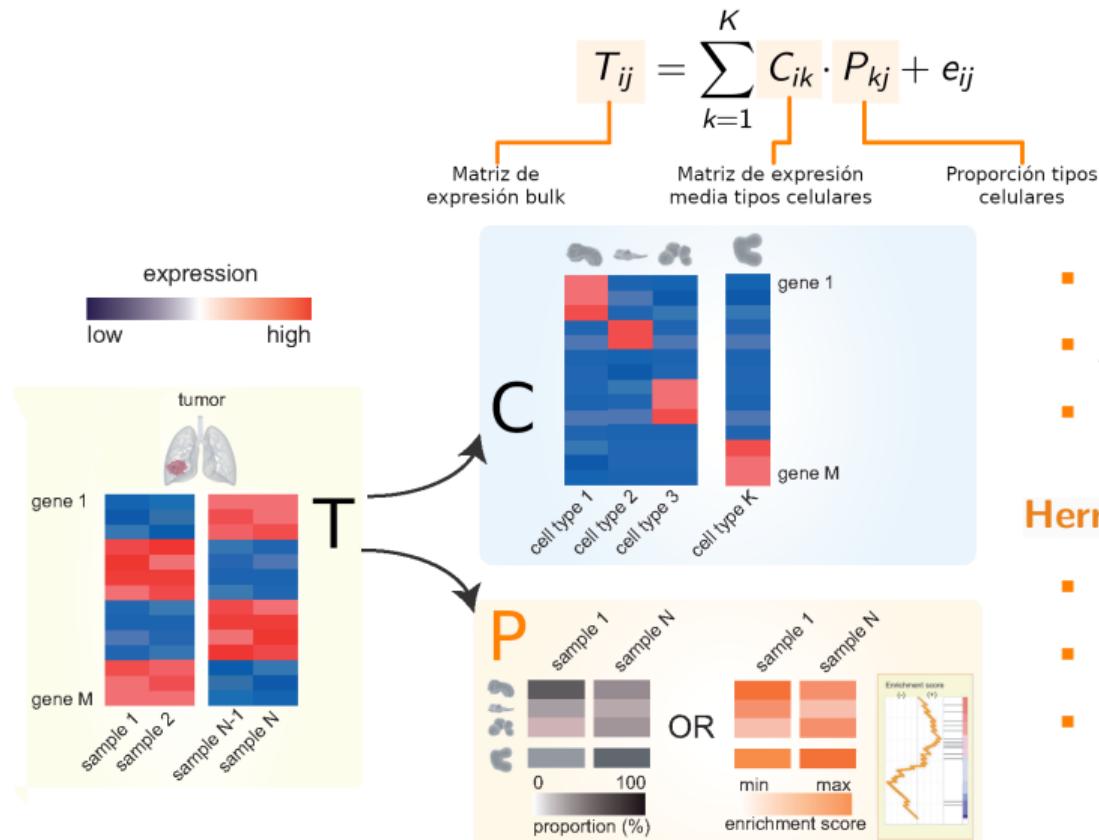
- i : genes ($i = 1 \dots M$).
- j : muestras *bulk* ($j = 1 \dots N$).
- k : tipos celulares ($k = 1 \dots K$).

Deconvolución: marco de trabajo



- i : genes ($i = 1 \dots M$).
- j : muestras *bulk* ($j = 1 \dots N$).
- k : tipos celulares ($k = 1 \dots K$).

Deconvolución: marco de trabajo



- i : genes ($i = 1 \dots M$).
- j : muestras *bulk* ($j = 1 \dots N$).
- k : tipos celulares ($k = 1 \dots K$).

Herramientas publicadas

- Enriquecimiento de sets de genes.
- Mínimos cuadrados ordinarios.
- v -SVR

Método de deconvolución digitalDLSorter

Características

- Método basado en **Aprendizaje Profundo** → revolución en el campo del Aprendizaje Automático durante los últimos años por su desempeño.
- Uso de datos **scRNA-seq** → perfiles de expresión procedentes del propio entorno de estudio (micro-entorno tumoral cáncer de mama, cáncer de colon, entorno neuronal, etc.).

Implementación

- *Pipeline* escrita en varios lenguajes de programación.
- Cada paso escribe los datos intermedios en disco como ficheros tabulados.
- No ofrece la opción de utilizar modelos preentrenados.
- Uso complicado por otros usuarios.

Objetivos

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

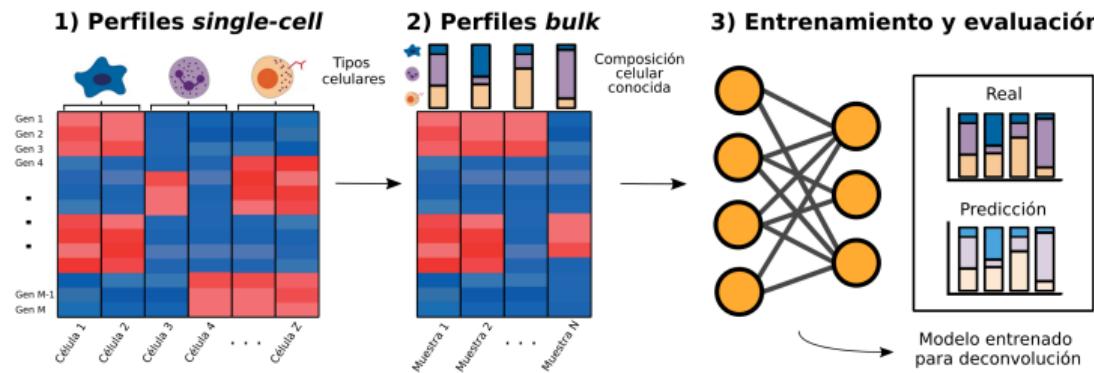
1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

1. Transformación de la *pipeline* original en un paquete de R.
 - Unificación de los lenguajes de programación → R.
 - Evitar la lectura/escritura de ficheros tabulados en disco.
 - Funcionalidades nuevas → uso de ficheros HDF5 como *back-end*, implementación de parámetros para construir modelos más personalizados.
 - Generación de documentación y viñeta → facilitar su uso.
2. Análisis de datos *scRNA-seq* procedentes de cáncer de mama.
 - Separación tipos tumorales y no tumorales.
 - Identificación de tipos celulares no tumorales.
3. Aplicación de la herramienta implementada.
 - Construcción de varios modelos y comparativa.
 - Incorporación de los mejores como modelos preentrenados en el paquete.

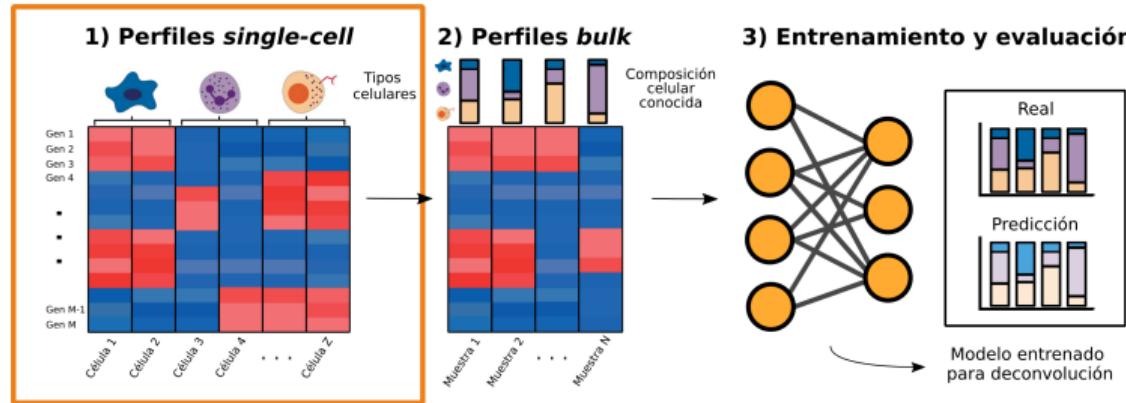
digitalDLSorteR: transformación de la *pipeline* en paquete de R

Fundamento del método



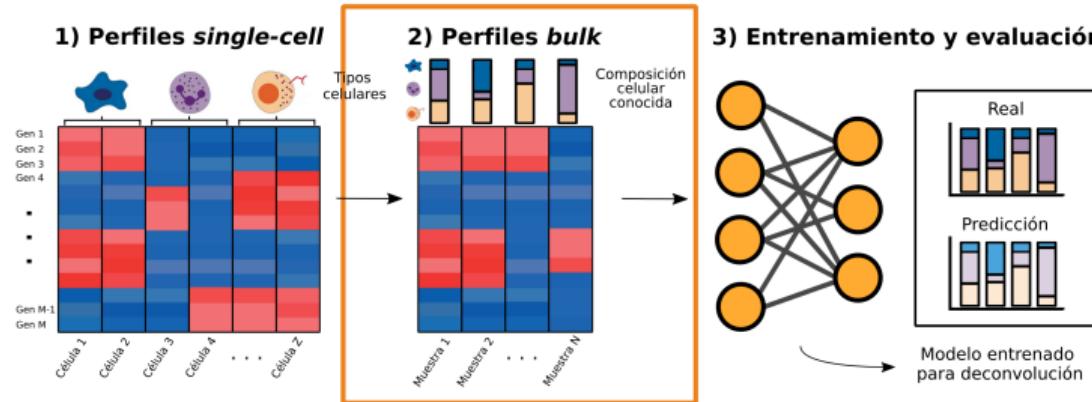
1. Simulación de perfiles *scRNA-seq* (si es necesario): uso del modelo ZINB-WaVE en caso de tipos celulares poco representados o pocas células de partida.
2. Simulación de perfiles *bulk RNA-seq* de composición celular conocida: agregación de perfiles *single-cell* en función del tipo celular al que pertenecen.
3. Entrenamiento y evaluación de la Red Neuronal Profunda: deconvolución de nuevas muestras *bulk*.

Fundamento del método



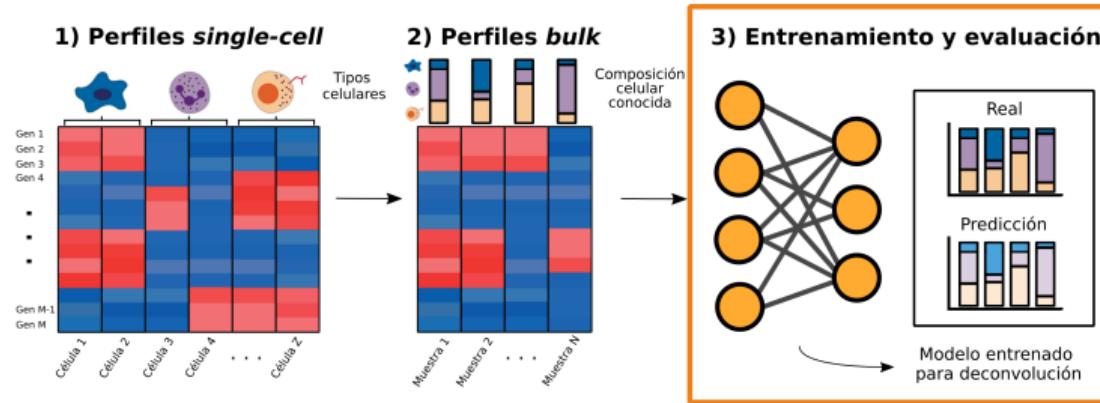
1. Simulación de perfiles *scRNA-seq* (si es necesario): uso del modelo ZINB-WaVE en caso de tipos celulares poco representados o pocas células de partida.
2. Simulación de perfiles *bulk RNA-seq* de composición celular conocida: agregación de perfiles *single-cell* en función del tipo celular al que pertenecen.
3. Entrenamiento y evaluación de la Red Neuronal Profunda: deconvolución de nuevas muestras *bulk*.

Fundamento del método



1. Simulación de perfiles *scRNA-seq* (si es necesario): uso del modelo ZINB-WaVE en caso de tipos celulares poco representados o pocas células de partida.
2. Simulación de perfiles *bulk RNA-seq* de composición celular conocida: agregación de perfiles *single-cell* en función del tipo celular al que pertenecen.
3. Entrenamiento y evaluación de la Red Neuronal Profunda: deconvolución de nuevas muestras *bulk*.

Fundamento del método



1. Simulación de perfiles *scRNA-seq* (si es necesario): uso del modelo ZINB-WaVE en caso de tipos celulares poco representados o pocas células de partida.
2. Simulación de perfiles *bulk RNA-seq* de composición celular conocida: agregación de perfiles *single-cell* en función del tipo celular al que pertenecen.
3. **Entrenamiento y evaluación de la Red Neuronal Profunda:** deconvolución de nuevas muestras *bulk*.

Fundamento del método

$$T_{ij} = \sum_{k=1}^K C_{ik} \cdot P_{kj} + e_{ij}$$

Matriz de expresión bulk Matriz de expresión media tipos celulares Proporción tipos celulares

1. Simulación de perfiles *scRNA-seq* (si es necesario): uso del modelo ZINB-WaVE en caso de tipos celulares poco representados o pocas células de partida.
2. Simulación de perfiles *bulk RNA-seq* de composición celular conocida: agregación de perfiles *single-cell* en función del tipo celular al que pertenecen.
3. Entrenamiento y evaluación de la Red Neuronal Profunda: deconvolución de nuevas muestras *bulk*.

Objetivo del método como paquete

1. Permitir la deconvolución directa de muestras *bulk RNA-seq*.
2. Permitir la construcción de nuevos modelos a partir de datos *scRNA-seq*

**digitalDLSorteR ofrece
dos flujos de trabajo**

- 
1. Uso de modelos preentrenados integrados en el paquete
 2. Construcción de nuevos modelos a partir de *scRNA-seq*

Uso de modelos preentrenados

Datos bulk RNA-seq

Genes	Muestras

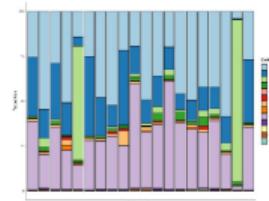
-deconvDigitalDLSorter→

Matriz de composición celular

Muestras	Tipos celulares

-barPlotCellTypes→

Proporciones predichas



```
1 deconvResults <- deconvDigitalDLSorter(  
2   data = TCGA.breast.small,  
3   model = "breast.chung.generic",  
4   normalize = TRUE  
5 )  
6 ## barplot showing results  
7 barPlotCellTypes(deconvResults)
```

- Modelos para la cuantificación de células inmunes en cáncer de mama.
 1. Modelo genérico: 7 tipos celulares.
 2. Modelo específico: 13 tipos celulares.
- Datos procedentes de Chung et al., 2017 ([GSE75688](#)).

Construcción de nuevos modelos

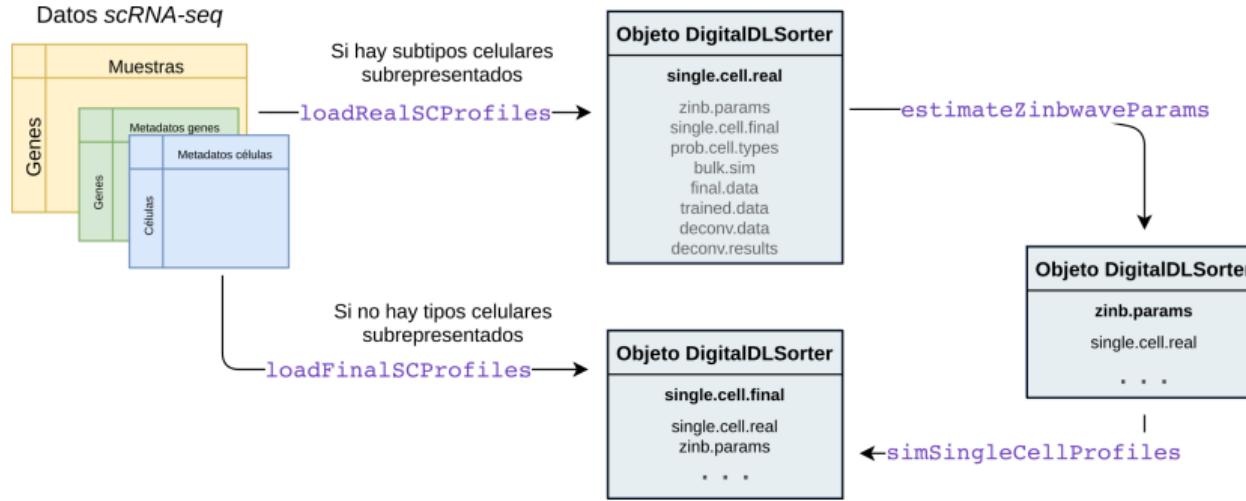
Clases

- *DigitalDLSorter*: núcleo del paquete.
- *ProbMatrixCellTypes*: información relativa a las matrices de composición celular.
- *DigitalDLSorterDNN*: información relativa a la Red Neuronal Profunda.
- Otras clases: *SingleCellExperiment*, *SummarizedExperiment*, etc.

Flujo de trabajo

1. Carga de datos y simulación de nuevos perfiles *single-cell* (si es necesario).
2. Generación de matrices de composición celular.
3. Simulación de perfiles *bulk RNA-seq* de acuerdo a las proporciones establecidas en el paso anterior. Preparación de los datos para el entrenamiento.
4. Entrenamiento de la red neuronal y evaluación.
5. Carga de nuevos datos y su deconvolución.

1. Carga de datos y simulación de perfiles *single-cell*



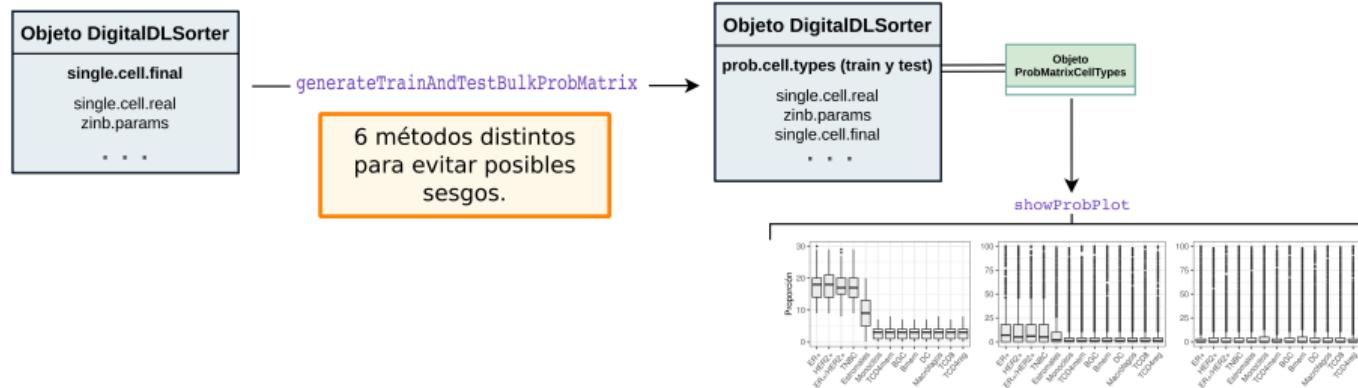
- Carga de los datos en un objeto de la clase *DigitalDLSorter* → matriz de expresión, metadatos de células y metadatos de genes.
- Si es necesario: estimación de parámetros mediante el modelo ZINB-WaVE y simulación de nuevos perfiles *single-cell*: distribución binomial negativa cero inflada.

1. Carga de datos y simulación de perfiles *single-cell*

```
1 DDLSChungSmall <- loadRealSCPProfiles(  
2   ## SingleCellExperiment object  
3   single.cell.real = sc.chung.breast,  
4   cell.ID.column = "Cell_ID",  
5   gene.ID.column = "external_gene_name",  
6   min.cells = 1,  
7   min.counts = 1,  
8   project = "Chung_example"  
9 )
```

```
1 ## estimation of ZINB-Wave parameters  
2 DDLSChungSmall <- estimateZinbwaveParams(  
3   object = DDLSChungSmall,  
4   cell.ID.column = "Cell_ID",  
5   gene.ID.column = "external_gene_name",  
6   cell.type.column = "Cell_type",  
7   cell.cov.columns = "Patient",  
8   gene.cov.columns = "gene_length"  
9 )  
10 ## simulation of new profiles  
11 DDLSChungSmall <- simSingleCellProfiles(  
12   object = DDLSChungSmall,  
13   cell.ID.column = "Cell_ID",  
14   cell.type.column = "Cell_type",  
15   n.cells = 10 # 1000 in real situations  
16 )
```

2. Generación de la matriz de composición celular



- Carga de los datos en un objeto de la clase *DigitalDLSorter* → matriz de expresión, metadatos de células y metadatos de genes.
- Si es necesario: estimación de parámetros mediante el modelo ZINB-WaVE y simulación de nuevos perfiles *single-cell*: distribución binomial negativa cero inflada.

Introducción: Clustering

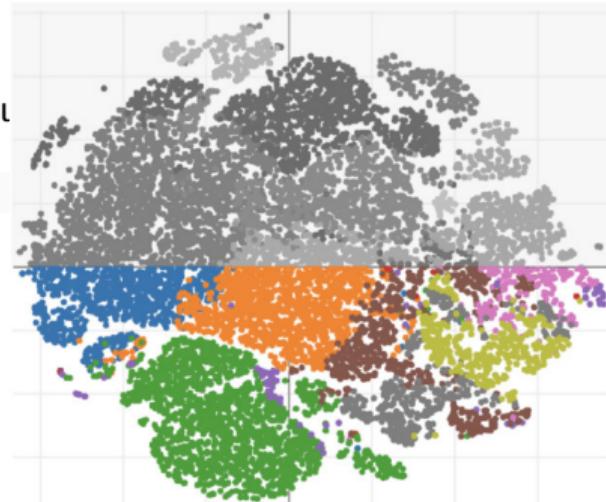
Conocer la **estructura de las poblaciones celulares**.

Por qué

- Qué fenotipos hay presentes.
- Qué perfiles de expresión.
- Qué células son similares y qué celu

Características

- Escalable.
- No paramétrico.
- Alta dimensionalidad.
- Geometrías no estándar.



Introducción: Clustering

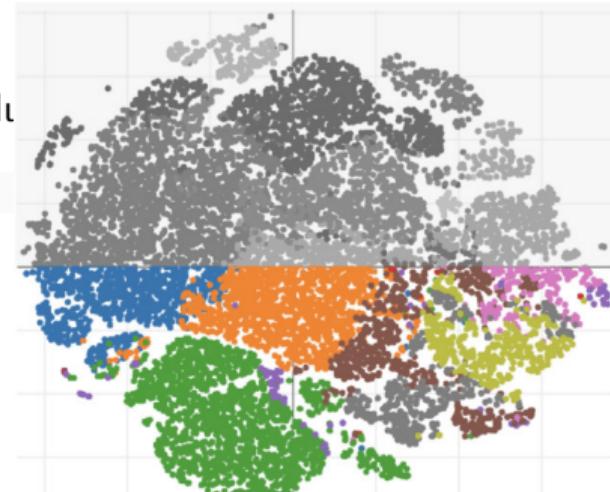
Conocer la **estructura de las poblaciones celulares.**

Por qué

- Qué fenotipos hay presentes.
- Qué perfiles de expresión.
- Qué células son similares y qué celu

Características

- Escalable.
- No paramétrico.
- Alta dimensionalidad.
- Geometrías no estándar.



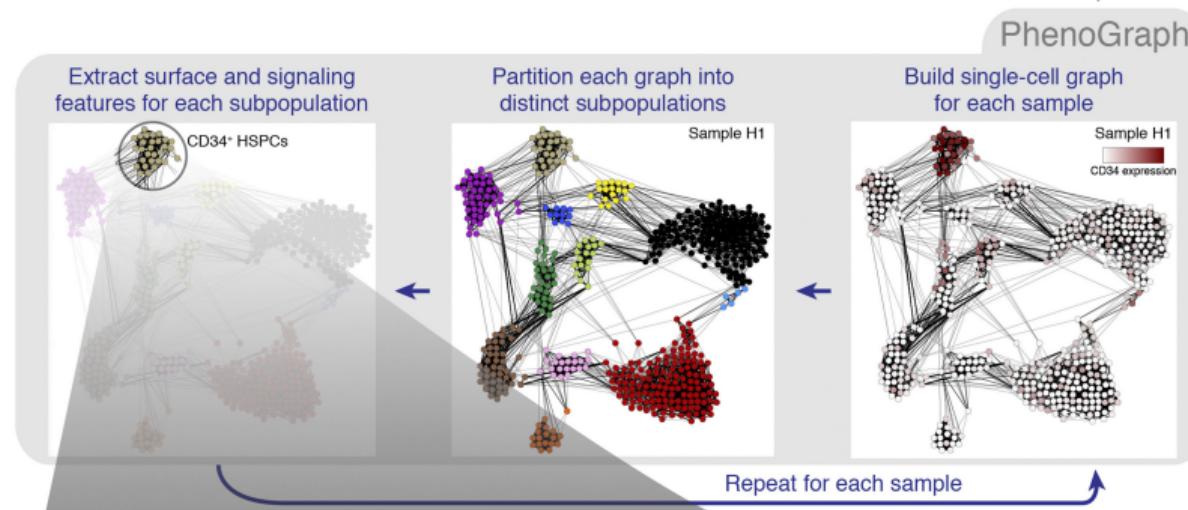
PhenoGraph

PhenoGraph

PhenoGraph

Aprendizaje no supervisado: Agrupa N células individuales en subpoblaciones (clústers) que representan los fenotipos presentes en la muestra.

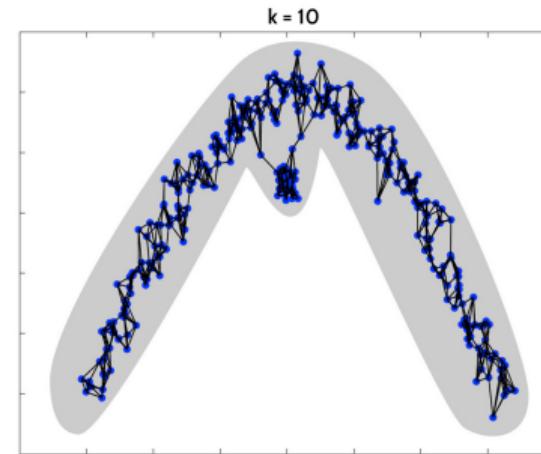
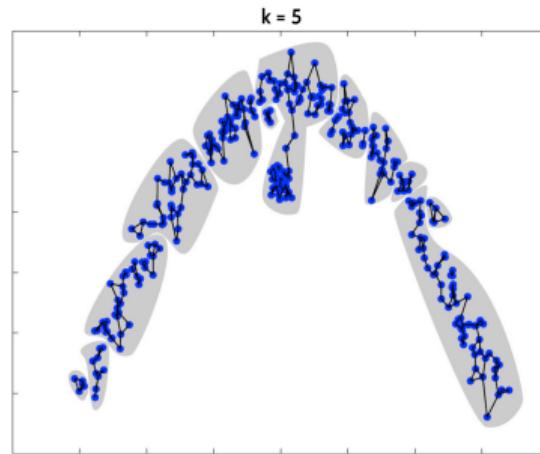
- Input: matriz $N \times D$.
- Busca regiones densas en el espacio D -dimensional → grafo.
- Busca comunidades en el grafo.
- Output: index asignando cada célula a una subpoblación.



Construcción del grafo: kNN

Paso 1: *k*-Nearest-Neighbors

- k vecinos más cercanos para cada célula con distancia Euclídea.
- Si k es bajo, baja conectividad entre las poblaciones.
- Si k es alto, resolución de poblaciones pequeñas se pierde.
- N -células con k -vecinos: N agrupaciones.

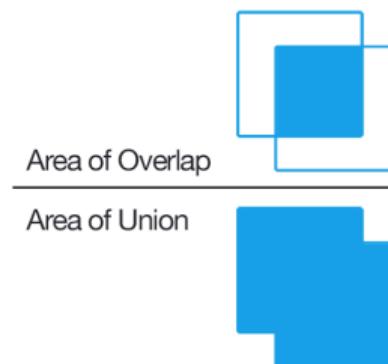


Construcción del grafo: Coeficiente de similitud de Jaccard

Paso 2: Índice de Jaccard

- Redefinición de k -vecinos de cada célula definidos por k NN.
- Índice de Jaccard: similitud entre dos conjuntos.

$$W_{i,j} = \frac{|v(i) \cap v(j)|}{|v(i) \cup v(j)|} =$$



Intuitivamente

- La similitud entre células será máxima cuando sus k -vecinos son los mismos → mismo fenotipo.
- Similitud decaerá entre células que comparten menos vecinos conectados → distinto fenotipo.

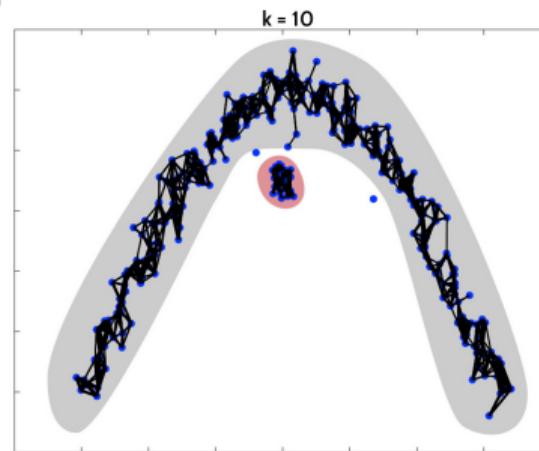
Construcción del grafo: Coeficiente de similitud de Jaccard

Resultado

Grafo ponderado con pesos basados en el número de vecinos que comparten.

Qué conseguimos

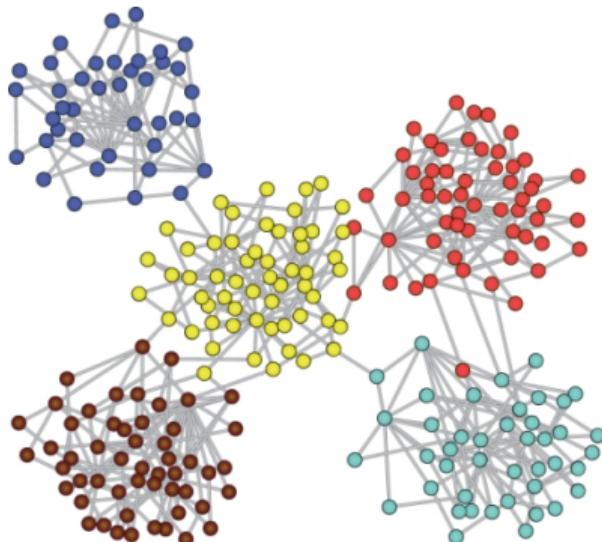
- Incorporamos la estructura de la distribución de los datos al grafo a través de los pesos.
- Estructura compacta y rica en información que captura la similitud entre células.
- Se refuerzan ejes en zonas densas.
- Se penalizan ejes en zonas dispersas.
- Poblaciones raras son mejor resueltas.
- Outliers tienden a ser excluidos.



Partición del grafo en comunidades

Comunidad

Presencia de grupos de nodos que están más densamente conectados entre sí que con otros nodos.



- Los grafos resultantes tienen esta propiedad por cómo se han construido y por el tipo de datos de los que proceden.
- Las comunidades representan células fenotípicamente similares.

Cómo encontramos las comunidades.

Cómo dividimos el grafo de forma óptima

Partición del grafo en comunidades: Modularidad

Modularidad (Q)

Medida de la calidad de una división particular de una red. Puede ser utilizada como función objetivo a maximizar por los algoritmos de detección de comunidades.

$$Q = \frac{1}{2m} \sum_{i,j} \left[W_{i,j} - \frac{s_i s_j}{2m} \right] \delta(c_i, c_j)$$

- $W_{i,j}$: peso del eje entre los nodos i y j .
- s_i : suma de los pesos de los ejes que involucran al nodo i .
- c_i : asignación de la comunidad para el nodo i .
- $\delta(u, v)$: función delta de Kronecker: 1 si $u = v$, 0 si $u \neq v$.
- $m = \frac{1}{2} \sum W_{i,j}$: peso total de la red.

Partición del grafo en comunidades: Modularidad

Modularidad (Q)

Medida de la calidad de una división particular de una red. Puede ser utilizada como función objetivo a maximizar por los algoritmos de detección de comunidades.

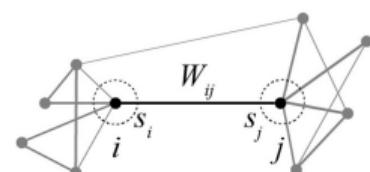
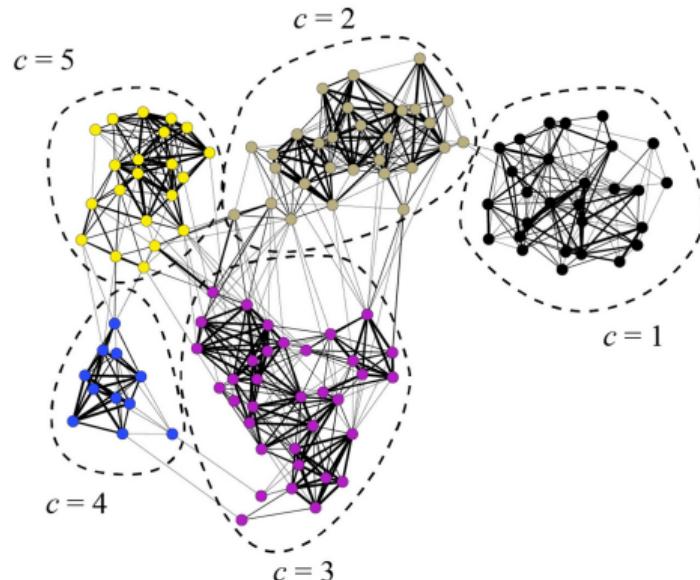
$$Q = \frac{1}{2m} \sum_{i,j} \left[W_{i,j} - \frac{s_i s_j}{2m} \right] \delta(c_i, c_j)$$

Peso de las ramas
de la comunidad Peso total de las
ramas en el grafo Nodos de la
misma comunidad

Intuitivamente

- Solo se calcula para pares de nodos de la misma comunidad.
- Cuando $W_{i,j}$ es mayor, la modularidad es más alta.
- Cuando $\frac{s_i s_j}{2m}$ es mayor, la modularidad baja.

Partición del grafo en comunidades: Modularidad



$$Q = \frac{1}{2m} \sum_{i,j} \left[W_{ij} - \frac{s_i s_j}{2m} \right] \delta(c_i, c_j)$$

Búsqueda óptima: Problema de optimización combinatorial **NP-completo**.

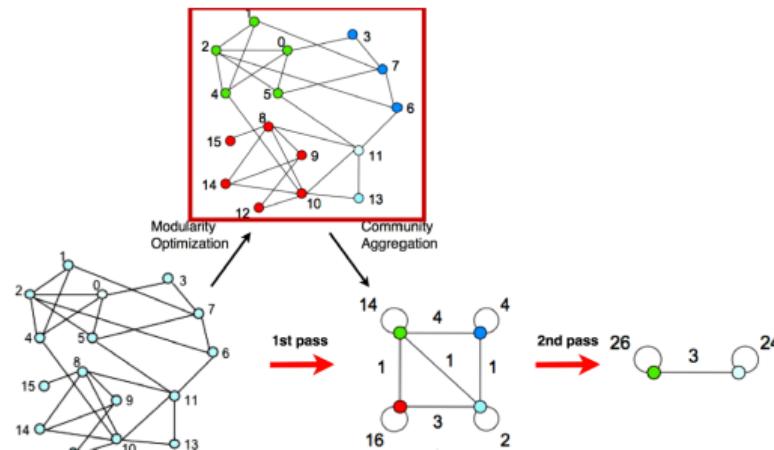
Necesitamos algoritmos heurísticos: **Método de Louvain**.

Partición del grafo en comunidades: Método de Louvain

Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).

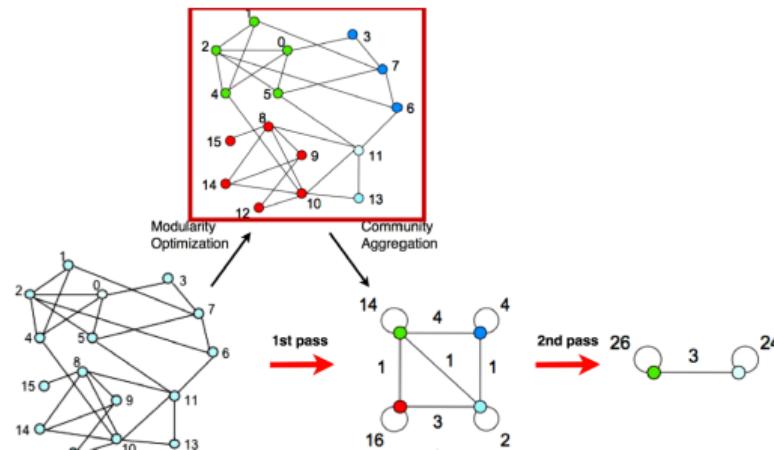


Partición del grafo en comunidades: Método de Louvain

Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).
2. Cada nodo i se asigna a la de cada vecino j y se calcula ΔQ .

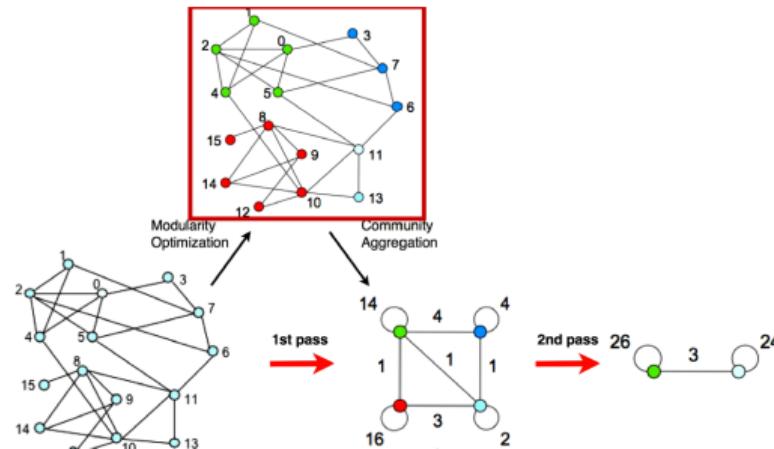


Partición del grafo en comunidades: Método de Louvain

Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).
2. Cada nodo i se asigna a la de cada vecino j y se calcula ΔQ .
3. Si ΔQ resultante es positiva, se asigna i a dicha comunidad.



Partición del grafo en comunidades: Método de Louvain

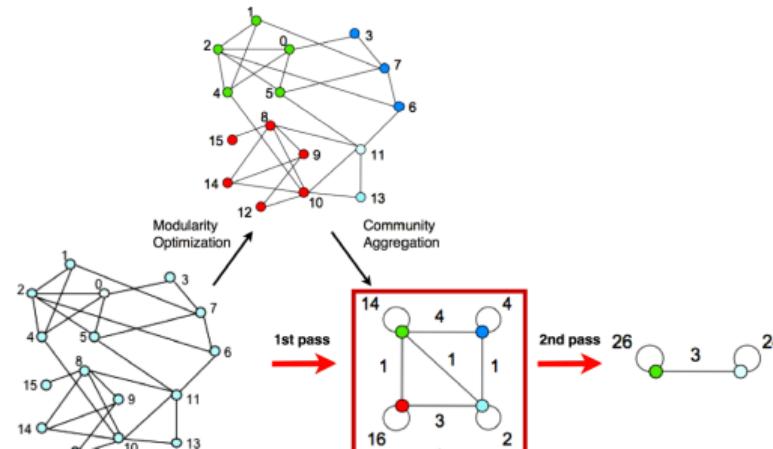
Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).
2. Cada nodo i se asigna a la de cada vecino j y se calcula ΔQ .
3. Si ΔQ resultante es positiva, se asigna i a dicha comunidad.

Segunda fase

1. Se construye una nueva red con las comunidades obtenidas anteriormente como nodos.



Partición del grafo en comunidades: Método de Louvain

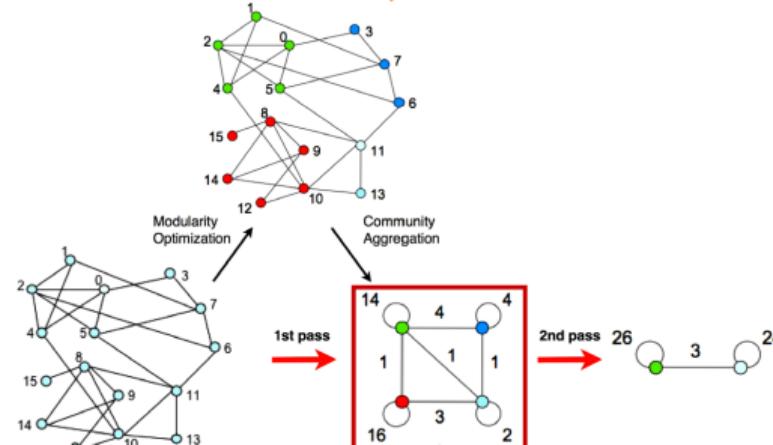
Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).
2. Cada nodo i se asigna a la de cada vecino j y se calcula ΔQ .
3. Si ΔQ resultante es positiva, se asigna i a dicha comunidad.

Segunda fase

1. Se construye una nueva red con las comunidades obtenidas anteriormente como nodos.
2. Los pesos se computan como la suma de los pesos de cada comunidad.



Partición del grafo en comunidades: Método de Louvain

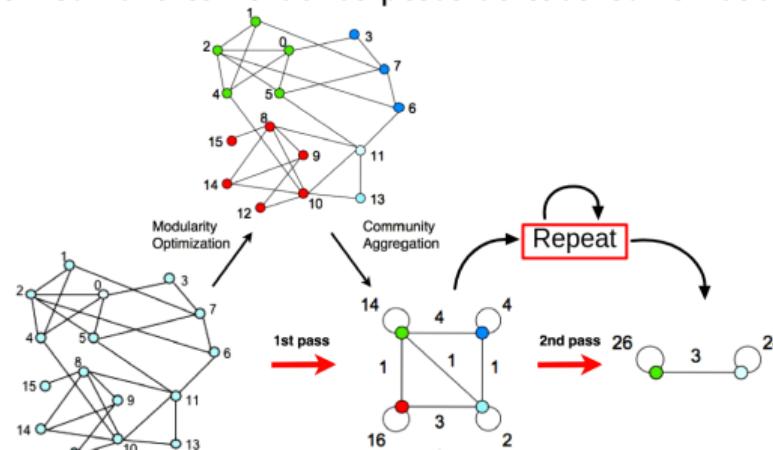
Aproximación heurística con dos fases repetidas iterativamente:

Primera fase

1. Se asigna una comunidad a cada nodo (N comunidades).
2. Cada nodo i se asigna a la de cada vecino j y se calcula ΔQ .
3. Si ΔQ resultante es positiva, se asigna i a dicha comunidad.

Segunda fase

1. Se construye una nueva red con las comunidades obtenidas anteriormente como nodos.
2. Los pesos se computan como la suma de los pesos de cada comunidad.



Pseudocódigo

Input: data set of single-cell measurements $X = \{x_1, \dots, x_N\}$

Output: subpopulation index assigning each cell in X to one of M groups

Inicialización:

```
for each cell  $x_i$ 
    find the indices (i) of the k nearest cells
```

Construcción del grafo:

```
set of vertices  $V = \{v_1, \dots, v_N\}$  to each cell in  $X$ 
```

```
set of edges  $E = \{\}$ 
```

```
for each pair of cells  $x_i$  and  $x_j$ 
```

```
    compute  $W_{ij}$ 
```

```
    if  $W_{ij} > 0$ , add  $e_{ij} = W_{ij}$  to E
```

```
return  $G = (V, E)$ 
```

Detección de comunidades:

```
for t in {1, ..., 100}
```

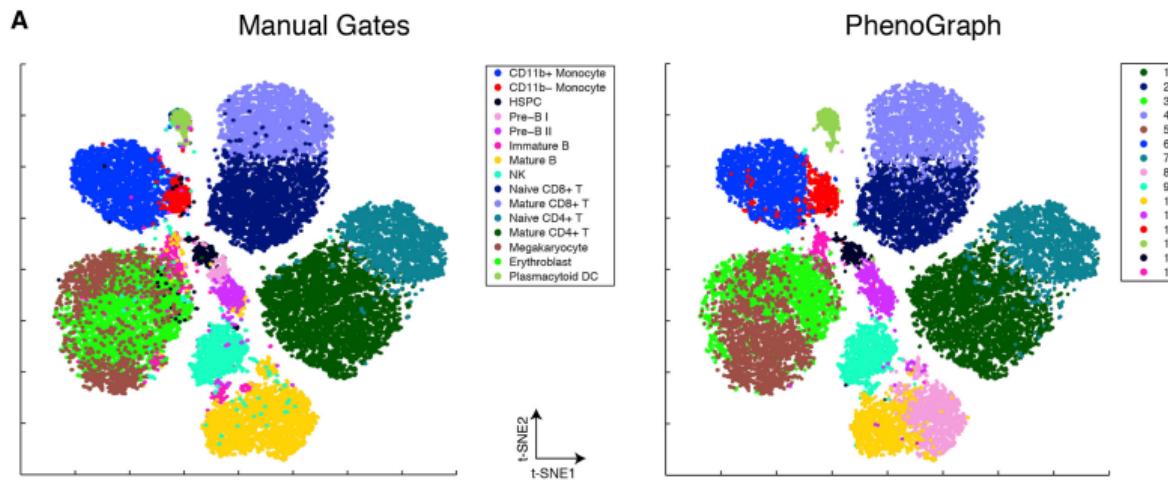
```
    decompose  $G$  into  $C_t$  by Louvain Method
```

```
    determine best  $C_t$  by maximum  $Q$ 
```

```
return  $C = C_t$ 
```

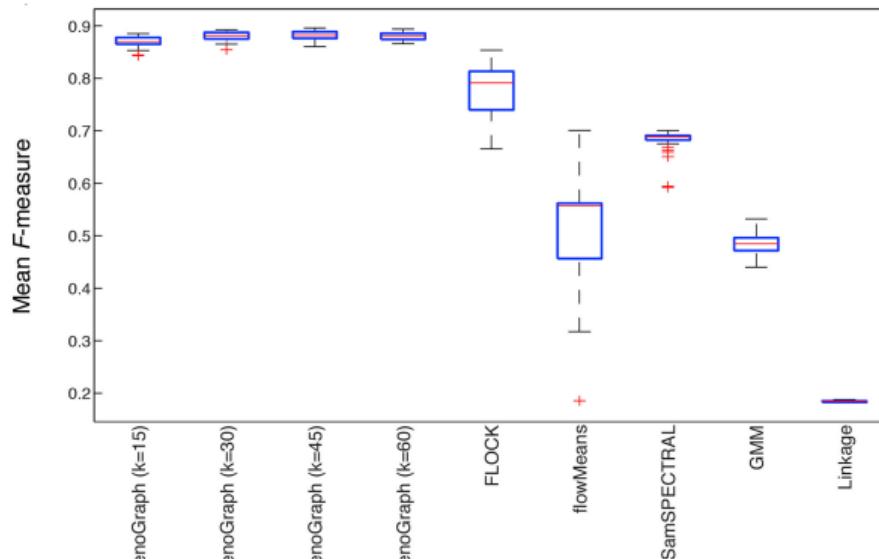
Validación en células inmunes de adulto sano

- Datos públicos: 30.000 células de médula ósea asignadas manualmente mediante técnicas estándar.
- Mayor precisión y escalabilidad que otros métodos.
- Robusto con diferentes parámetros.



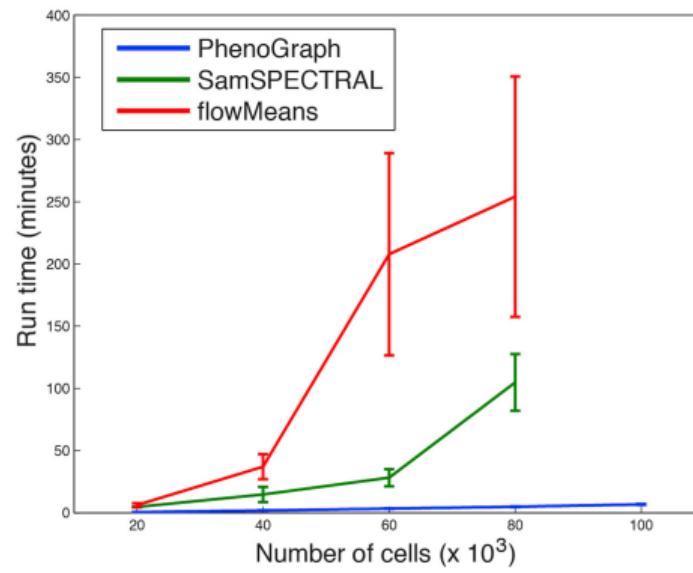
Validación en células inmunes de adulto sano

- Datos públicos: 30.000 células de médula ósea asignadas manualmente mediante técnicas estándar.
- Mayor precisión y escalabilidad que otros métodos.
- Robusto con diferentes parámetros.



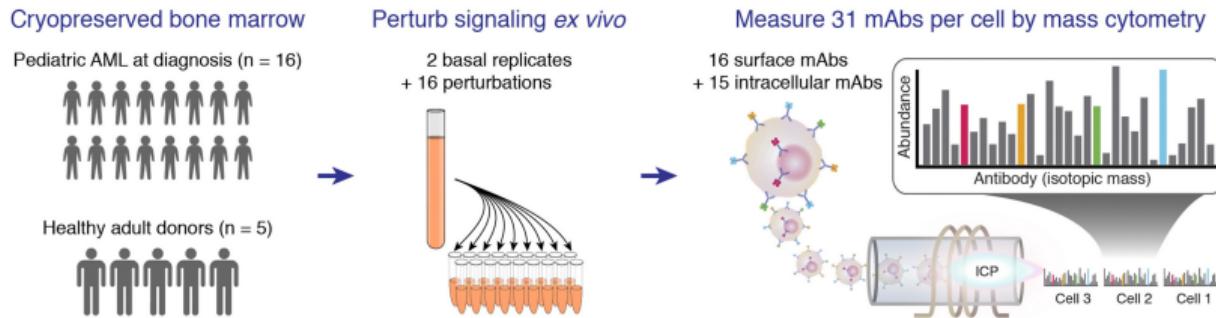
Validación en células inmunes de adulto sano

- Datos públicos: 30.000 células de médula ósea asignadas manualmente mediante técnicas estándar.
- Mayor precisión y escalabilidad que otros métodos.
- Robusto con diferentes parámetros.



Resultados

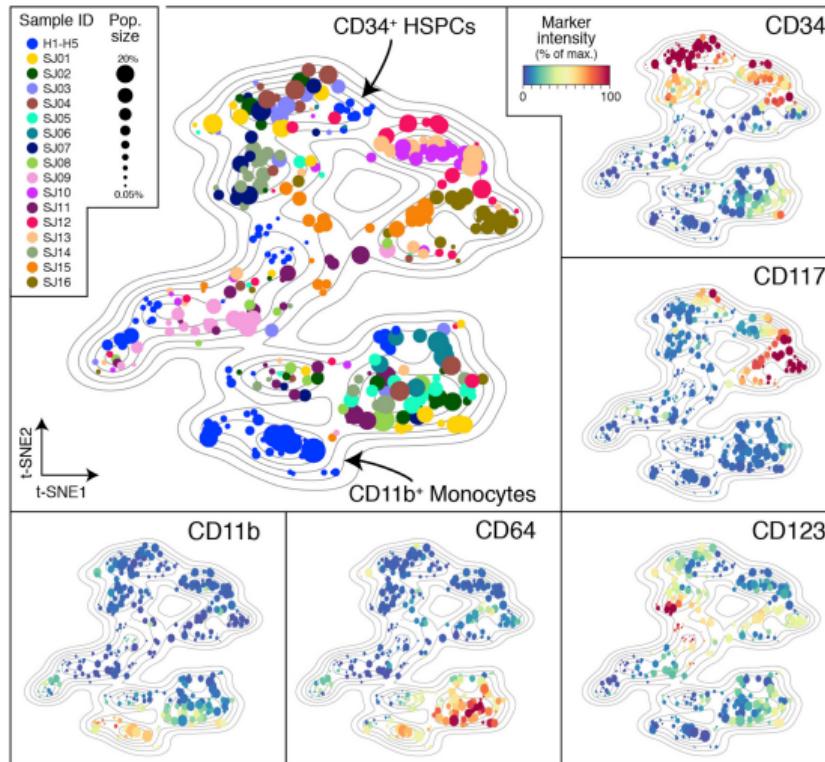
Resultados: Datos



- **21 individuos:** 16 pacientes con AML, 5 adultos donantes sanos.
- **31 dimensiones:** 16 marcadores de superficie más informativos y 14 sondas contra proteínas importantes en señalización intracelular.

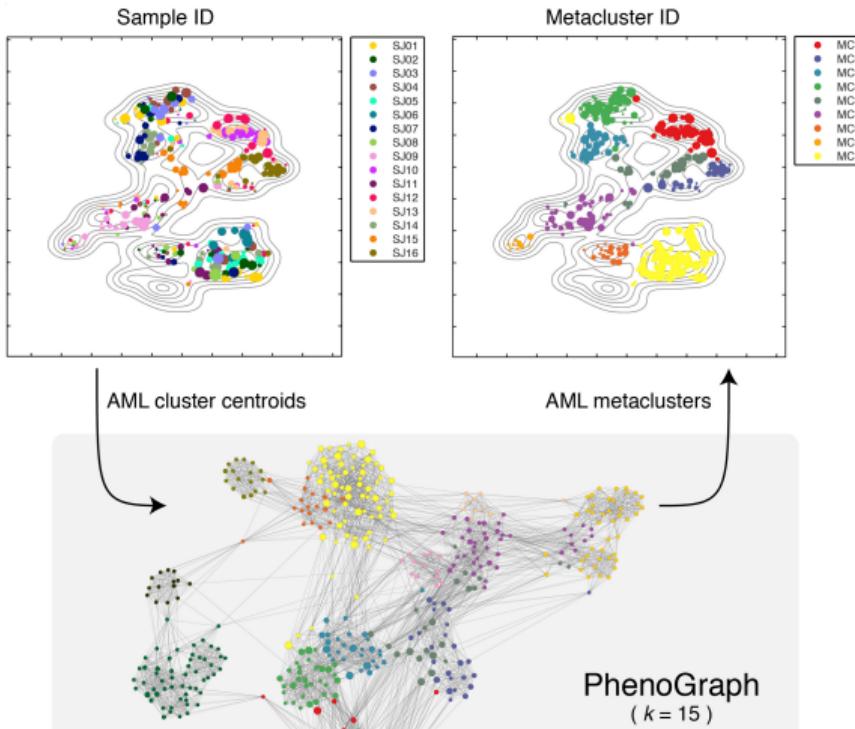
Resultados: PhenoGraph sobre set de datos AML

- PhenoGraph sobre la cohorte de pacientes AML y donantes sanos ($k = 50$).
- 28 subpoblaciones / muestra.



Resultados: PhenoGraph sobre subpoblaciones AML

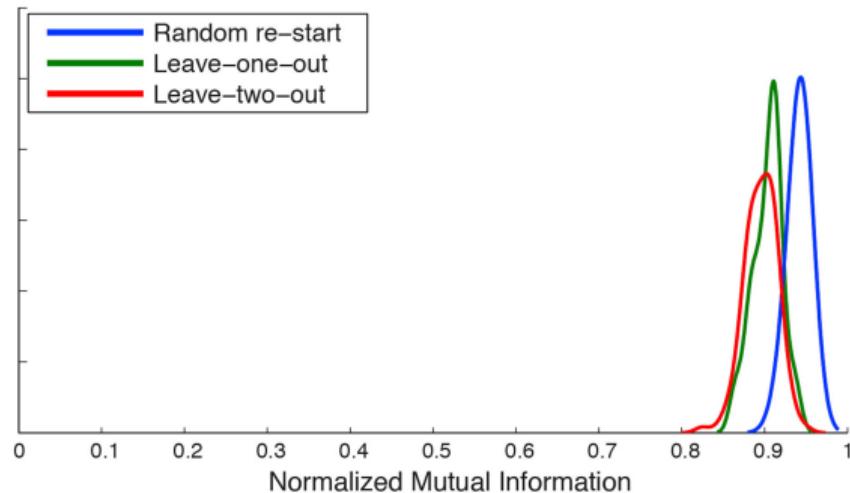
- PhenoGraph sobre subpoblaciones AML ($k = 15$).
- Cada subpoblación definida por centroides: matriz 425×16 .



Resultados: Cross-validation

Resampling de subpoblaciones AML

- 16 repeticiones aleatorias de todos los datos.
- 1 paciente fuera: 16 sets de datos.
- 2 pacientes fuera: 120 sets de datos.



Metaclústeres AML son robustos frente a cambios en el dataset.

Otros métodos

Comparación con otros métodos

Methods		Implementation tools	Description
Unsupervised	Accense	MATLAB	tSNE dimension reduction and 2D projection, kernel-based estimation of density, density-based peak-finding and partitioning
	PhenoGraph	R (cytofkit package)	Detection of k -nearest neighbors of each cell, Jaccard similarity coefficient as connectivity, community detection based on connection density
	Xshift	Vortex	Weighted k -nearest neighbor density estimation, detection of density centroids, cells linked to centroid via density-ascending paths
	FlowSOM	R	Self-organizing map (SOM) trained on scaled data, nodes of SOM connected by minimal spanning tree, consensus hierarchical meta-clustering of nodes
	flowMeans	R	K estimated by peak numbers of kernel density, kmeans clustering of estimated K , merging clusters by distance metrics
	DEPECHE	R	Tuning penalty by resampling dataset, penalized kmeans clustering
	kmeans	MATLAB	Standard kmeans procedure

Conclusiones

- Robusto frente a cambio de parámetros y remuestreo.
- Preciso y escalable.
- Primeras posiciones junto a FlowSOM, flowMeans y DEPECHE.

Implementación

MATLAB, Python, R, paquetes de análisis scRNAseq (Seurat).

Referencias

- Blondel VD, Guillaume JL, Lambiotte R and Lefebvre E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*.
- Girvan M, Newman, MEJ. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*. 99 (12):7821–7826.
- Levine JH, Simonds EF, Bendall SC, Davis KL, Amir ED, Tadmor MD, et al. (2015). Data-driven phenotypic dissection of AML reveals progenitor-like cells that correlate with prognosis. *Cell*. 162:184–97.
- Liu X, Song W, Wong BY, et al. (2019). A comparison framework and guideline of clustering methods for mass cytometry data. *Genome Biol.* 20:297–301.
- Newman, MEJ. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*. 103 (23):8577–8582
- Weber LM and Robinson MD. (2016). Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data. *Cytometry*. 89:1084–1096.

Gracias por vuestra atención.

Extra

F-measure

$$F(c_i, k_j) = \frac{2 \times Pr_{ij} \times Re_{ij}}{Pr_{ij} + Re_{ij}}$$

$$F_{mean}(C, K) = \sum_{ci} \frac{|c_i|}{N} \max_{kj} F(c_i, k_j)$$

Normalized Mutual Information

$$NMI = \frac{I(X; Y)}{\sqrt{H(X)H(Y)}}$$

Non-redundancy score (NRS)

$$NRS(A_p) = \sum_{k=1:c} |coeff(k)| \times eigenvalue(k)$$