PEC 1. Análisis de datos ómicos

Javier Matos García

2025-03-24

Tabla de contenido.

Apartado	Contenido
1. Resumen.	Breve explicación de la justificación y resultados del trabajo.
2. Objetivos.	Objetivos principales y secundarios del trabajo.
3. Metodología.	Justificación objetiva y personal de la selección de los datos. Explicación paso a paso del proceso de creación del SummarizedExperiment.
4. Resultados.	Resolución a las preguntas planteadas en el enunciado y análisis del SummarizedExperiment
5. Discusión.	Valoración personal del objeto, de la base de datos utilizada y de la fase de preparación de los datos.
6. Conclusiones	Análisis objetivo del trabajo.
7. Referencias	Enlaces utilizados durante el trabajo y enlace al GitHub requerido.

1. Resumen.

Este trabajo consiste principalmente en un acercamiento al objeto SummarizedExperiment al que puede accederse tras la instalación de Bioconductor en RStudio. Este objeto posee unas características sumamente interesantes de cara a la organización de la información relevante de un experimento y supone una herramienta muy interesante desde el punto de la organización del trabajo de un bioinformático ya que permite el acceso a los datos y metadatos del estudio de forma rápida.

Este objeto puede ser considerado una versión más versatil del objeto ExpressionSet utilizado de forma nativa por Bioconductor, lo que habilita el análisis de estudios con características más específicas. De forma general, este objeto es claro y se puede utilizar directamente una vez se le han asignado los datos de estudio, pero requiere de una exhaustiva fase previa de preparación de datos y de un manejo considerable de la herramienta R/RStudio.

2. Objetivos.

El objetivo principal de este trabajo es conocer el manejoobjeto SummarizedExperiment, desde su intalación hasta su uso, comprobando en el proceso la utilidad del mismo desde el punto de vista de la Bioinformática así como sus limitaciones y dificultades de uso. De manera paralela, este trabajo permite por una parte conocer

y manejar bases de datos de metabolómica y por otra el preprocesado de los resultados de un experimento para adaptarlos a un sistema de trabajo como este, por lo que se podrían considerar estos como dos objetivos secundarios del trabajo.

3. Metodología.

Para la realización de este trabajo se ha acudido a la base de datos MetabolomicsWorkbench que contiene un repositorio de datos de metabolómica provenientes de estudios y proyectos así como de datos referentes a metabolitos de interés que pueden encontrarse en diversos organismos. En concreto se ha utilizado el estudio ST000875 que comprueba las diferencias intraespecie entre cepas terrestres y marinas de *Nesterenkonia flava*.

Esta especie de actinomiceto tiene considerable interés por su capacidad de producir nesterkoniaina, un eter ciclico con actividad antialergénica además de otros compuestos de acción similar, tal y como puede comprobarse en esta publicación de PubMed. Sin embargo, el motivo de selección de este estudio no está relacionado con esta interesante capacidad de la especie.

El estudio muestra una diferencia notable en los patrones de transcriptómica que aparecen en cepas de la misma especie que se han obtenido de entornos terrestres y acuáticos marinos. Esta diferencia intraespecie a nivel de transcriptómica suscita un planteamiento interesante desde el punto de vista de la conservación de especies que pueden habitar y/o habitan en ecosistemas diferentes ya que presenta un factor que usualmente pasa desapercibido en los programas de conservación como es la adaptación al entorno desde el punto de vista ómico.

Dado que mis objetivos profesionales se encuentran orientados al trabajo como investigador en el campo de la Biología de la Conservación, he considerado que este proyecto, por lo arriba mencionado, era el que me resultaba más interesante, especialmente frente a otros mucho más orientados a farmacología, estudios de cancer y tratamientos de patologías.

En cuanto al procedimiento llevado a cabo para realizar este trabajo, el primer paso ha sido la obtención de los ficheros descargandolos directamente de la base de datos de MetabolomicsWorkbench. En este estudio se ofrecen por un lado los metadatos del estudio en formato .json o .txt y por otro se dispone de los resultados, utilizados en este trabajo, y datos obtenidos al realizar la espectroscopia de resonancia magnética nuclear, NMR en adelante, que componen susodichos resultados.

La primera fase del trabajo consiste en una preparación de los datos para poder construir el objeto SummarizedExperiment posteriormente. En cuanto al preprocesado del fichero de los resultados, este solo ha requerido de modificaciones realizadas directamente en R antes de su introducción al objeto por lo que se detallarán estos más adelante. Sin embargo, respecto al fichero de metadatos, este ha requerido de una modificación manual del fichero .txt ya que la estructura original imposibilitaba su carga en RStudio. Debido a esta modificación, se adjuntarán ambas versiones a este trabajo.

Las modificaciones realizadas sobre el fichero de metadatos han consistido en el reajuste de las filas de la primera columna del mismo ya que algunos apartados, como el título y el *abstract*, presentaban la misma linea en multiples ocasiones ya que sus "valores" requerian de multiples líneas. Esto se ha solventado convirtiendo estas filas a una única que contiene todo el texto. La segunda modificación se encuentra en las muestras, donde se ha fusionado el número identificador de la segunda columna con el nombre estandarizado de la primera, haciendo que todo el fichero presente unicamente dos columnas y eliminando el problema de repetición del valor en las filas.

Antes de comenzar con el proceso de creación del objeto es imprescindible instalar la librería SummarizedExperiment, la cual requiere de Bioconductor. Es posible indicar durante la instalación del paquete que solo tenga lugar si no se dispone previamente de el, de manera que se acorte el proceso. Al trabajar en RMarkdown, es aconsejable añadir la linea message=FALSE y warning=FALSE para evitar que en el documento de salida aparezcan los mensajes de instalación de los programas.

```r
#Primero se instala Bioconductor siempre que no se disponga del paquete.
if (!requireNamespace("BiocManager", quietly = TRUE)) {
  install.packages("BiocManager")
}

#Este mismo proceso permite instalar SummarizedExperiment desde Bioconductor
if (!requireNamespace("SummarizedExperiment", quietly = TRUE)) {
  BiocManager::install("SummarizedExperiment")
}

#Finalmente se activa la librería de SummarizedExperiment para poder acceder a sus funciones.
library(SummarizedExperiment)
```

A continuación se deben cargar los ficheros que formarán parte del objeto SummarizedExperiment. Debido a la estructura del fichero que contiene los resultados de la investigación, resulta aconsejable considerar que no dispone de un header para poder acceder más facilmente a las filas ya que de otra manera se generan problemas posteriormente para obtener el nombre de las filas y columnas. Por otra parte, el fichero de metadatos es la versión modificada del anterior, tal y como se ha explicado anteriormente.

```r
resultados = read.delim(
  "C:\\Users\\vossl\\OneDrive\\Desktop\\Vräengard\\Máster\\Omicas\\PEC1\\ST000875\\ST000875_AN001412_Res
                         header = FALSE,
                         sep = "\t")

metadatos = read.delim(
  "C:\\Users\\vossl\\OneDrive\\Desktop\\Vräengard\\Máster\\Omicas\\PEC1\\ST000875\\ST000875_AN001828_mod
                        header = TRUE,
                        sep = "\t",
                        stringsAsFactors = FALSE,
                        fill = TRUE)
```

El fichero de los resultados contiene toda la información necesaria para poder crear el objeto Summarized-Experiment, pero requiere de ligeras modificaciones para poder disponer correctamente de todos los datos.

```r
#Se selecciona la segunda fila como nombres de las columnas
nombre_col = resultados[2, ]

#Se eliminan las filas que no contienen resultados del experimento.
Datos = resultados[-c(1, 2), ]

#Se reasignan los valores extraidos como nombres de las columnas.
colnames(Datos) = nombre_col

#Se transforman los datos de DataFrame a Matriz.
Exp <- as.matrix(Datos[, -1]) #Se excluye la fila que se corresponde con los factores.

#Se crean los datos correspondientes a colData del objeto SummarizedExperiment.
columna = resultados[1, -1] #Se toman los datos de la primera fila.
columna = t(columna) #Se trasponen los datos
Datos_col = DataFrame(columna) #Se transforman a DataFrame.

#Se asignan los nombres extraidos previamente.
rownames(Datos_col) = nombre_col[-1] #Se omite el primer término (Factor)
```

```
#Se toma el nombre de las filas del Data Frame original.
Datos_fila = DataFrame(Factor = Datos[, 1])

#Se asignan los nombres para evitar fallos de contenido.
colnames(Exp) = rownames(Datos_col)
rownames(Exp) = Datos_fila$Factor
```

Una vez se dispone de toda la información necesaria se puede proceder a crear el objeto SummarizedExperiment siguiendo las indicaciones del mismo que se pueden encontrar en la web de Bioconductor(1, 2) asi como en RDocumentation.

```
SE <- SummarizedExperiment(
  assays = list(counts = Exp),
  colData = Datos_col,
  rowData = Datos_fila,
  metadata = metadatos)

#Se ejecuta el objeto para comprobar en pantalla que se ha creado correctamente.
SE
```

```
## class: SummarizedExperiment
## dim: 570 72
## metadata(1):
##   X.METABOLOMICS.WORKBENCH.Jinmei_20170812_050650.DATATRACK_ID.1217.STUDY_ID.ST000875.ANALYSIS_ID.AN
## assays(1): counts
## rownames(570): 0.67595 0.6897 ... 9.48576 9.49716
## rowData names(1): Factor
## colnames(72): Strain:1K00606 | Strain source:Marine | Medium:A14
##   Strain:1K00606 | Strain source:Marine | Medium:A14 ... Strain:1A10663
##   | Strain source:Terrestrial | Medium:A3 Strain:1A10663 | Strain
##   source:Terrestrial | Medium:A3
## colData names(1): X1
```

4. Resultados.

Una vez que se dispone del objeto SummarizedExperiment con todos los datos y metadatos del experimento, se puede proceder a comprobar la estructura del mismo, haciendo uso de las funciones descritas en la web de Bioconductor orientada a mostrar la organización de los datos haciendo uso de SummarizedExperiment.

La función dim() devuelve el tamaño del dataset, mostrando filas y columnas respectivamente.

```
dim(SE)
```

```
## [1] 570  72
```

La función dimnames() muestra en pantalla el nombre de cada fila y columna lo cual es interesante para comprobar que los datos se han cargado correctamente. En datasets de gran tamaño, como el utilizado en esta actividad, el resultado puede ser dificil de analizar por su volumen, por lo que puede combinarse con la función head() para ver unicamente las primeras lineas:

```r
head(dimnames(SE)[[1]]) #Muestra el nombre de las primeras filas.
```

```
## [1] "0.67595" "0.6897"  "0.70205" "0.71415" "0.7341"  "0.7481"
```

```r
head(dimnames(SE)[[2]]) #Muestra el nombre de las primeras columnas.
```

```
## [1] "Strain:1K00606 | Strain source:Marine | Medium:A14"
## [2] "Strain:1K00606 | Strain source:Marine | Medium:A14"
## [3] "Strain:1K00606 | Strain source:Marine | Medium:A14"
## [4] "Strain:1K00606 | Strain source:Marine | Medium:A14"
## [5] "Strain:1K00606 | Strain source:Marine | Medium:A14"
## [6] "Strain:1K00606 | Strain source:Marine | Medium:A14"
```

Otra función interesante para analizar un SummarizedExperiment es assay() que imprime en pantalla los valores de las primeras filas para cada columna. Sin embargo, al igual que con la función dimnames(), esta función genera un gran volumen de datos por lo que puede combinarse con otras funciones como class() y head(). En este caso la función head se puede acompañar de intervalos para indicar cuanta información se quiere mostrar.

```r
class(assay(SE)) #Imprime en pantalla el tipo de valor que contiene el SummarizedExperiment.
```

```
## [1] "matrix" "array"
```

```r
head(assay(SE)[1:5, 1:5]) #Muestra las primeras cinco filas de las primeras cinco columnas.
```

```
##         Strain:1K00606 | Strain source:Marine | Medium:A14
## 0.67595 "128.7525819"
## 0.6897  "394.8150527"
## 0.70205 "261.7162479"
## 0.71415 "259.7580799"
## 0.7341  "1888.819776"
##         Strain:1K00606 | Strain source:Marine | Medium:A14
## 0.67595 "125.6252901"
## 0.6897  "392.1212998"
## 0.70205 "253.4285078"
## 0.71415 "268.829209"
## 0.7341  "1863.044471"
##         Strain:1K00606 | Strain source:Marine | Medium:A14
## 0.67595 "176.797635"
## 0.6897  "545.4448804"
## 0.70205 "361.5180018"
## 0.71415 "309.6723149"
## 0.7341  "1409.464796"
##         Strain:1K00606 | Strain source:Marine | Medium:A14
## 0.67595 "200.8924656"
## 0.6897  "587.1165141"
## 0.70205 "392.9114948"
## 0.71415 "367.0853183"
## 0.7341  "1209.893835"
##         Strain:1K00606 | Strain source:Marine | Medium:A14
```

```
## 0.67595 "133.4270631"
## 0.6897  "411.9488165"
## 0.70205 "295.6889231"
## 0.71415 "272.1396754"
## 0.7341  "1751.99225"
```

Una función que puede ser muy util cuando se trabaja con genomas es rowRanges() ya que esta devuelve un objeto de tipo GRanges. Sin embargo en estudios de transcriptómica donde no se especifican los rangos genómicos esta información no se puede adicionar al SummarizedExperiment, por lo que al utilizar esta función se devuelve un valor NULL como se observa a continuación:

```
rowRanges(SE)
```

```
## NULL
```

Finalmente, un último analisis que puede realizarse sobre el objeto Summarized consiste en hacer uso de la función colData() que permite mostrar metainformación asociada a los datos de las columnas. En este caso, aparece que existen 72 filas lo que se debe a que el DataFrame del SummarizedExperiment contiene 72 columnas mientras que solo aparece una fila ya que en este caso solo se está considerando el factor de transcripción como variable.

```
colnames(colData(SE))[colnames(colData(SE)) == "X1"] = "Factors"
#Se cambia el nombre del parámetro por el que se definen las muestras.
colData(SE)
```

```
## DataFrame with 72 rows and 1 column
##                                                          Factors
##                                                      <character>
## Strain:1K00606 | Strain source:Marine | Medium:A14             1
## Strain:1K00606 | Strain source:Marine | Medium:A14             2
## Strain:1K00606 | Strain source:Marine | Medium:A14             3
## Strain:1K00606 | Strain source:Marine | Medium:A14             4
## Strain:1K00606 | Strain source:Marine | Medium:A14             5
## ...                                                         ...
## Strain:1A10663 | Strain source:Terrestrial | Medium:A3        68
## Strain:1A10663 | Strain source:Terrestrial | Medium:A3        69
## Strain:1A10663 | Strain source:Terrestrial | Medium:A3        70
## Strain:1A10663 | Strain source:Terrestrial | Medium:A3        71
## Strain:1A10663 | Strain source:Terrestrial | Medium:A3        72
```

Finalmente, para responder a la pregunta planteada en el enunciado de la PEC respecto a las diferencias entre el objeto SummarizedExperiment y la clase ExpressionSet se ha acudido nuevamente a la web de Bioconductor. En esta, se explica que ambos objetos son muy similares tanto en su uso como en su estructura, siendo la principal diferencia entre ambos que SummarizedExperiment es más versatil ya que permite mayor flexibilidad en cuanto a la información de las lineas, permitiendo incluso hacer uso de los ya mencionados GRanges que contienen la localización genómica. Esta diferencia es la que convierte a SummarizedExperiment en una herramienta más útil a la hora de trabajar con experimentos basados en la secuenciación de genomas (RNA-seq y ChIp-seq entre otros) frente a ExpressionSet que no tiene tal capacidad.

5. Discusión.

A modo de resumen, este trabajo ha permitido conocer las aplicaciones y utilidad del objeto SummarizedExperiment, así como sus limitaciones. Además, a lo largo del proceso se ha podido descubrir la base de datos

MetabolomicsWorkbench y se han practicado algunas técnicas para el preprocesado de datos obtenidos de una base de datos. De forma global, se puede considerar que los objetivos planteados al comienzo de esta tarea se han cumplido.

Como se ha comentado en el apartado de Resultados, SummarizedExperiment es un objeto más versatil que su contraparte ExpressionSet, permitiendo trabajar con diversos tipos de experimentos,lo que lo convierte en una herramienta muy interesante desde el punto de vista de la bioinformática porque permite acceder rapidamente a la información de un experimento y/o proyecto, mostrando los datos utilizados en el mismo, detalles sobre el equipo investigador y el procedimiento llevado a cabo..

A pesar de sus ventajas, y como consideración personal, este objeto podría ser considerado poco intuitivo ya que su utilización puede suponer un reto para investigadores que carezcan de las habilidades informáticas suficientes. Esta critica está relacionada proceso que se desea llevar a cabo ya que cargando los datos en formato .txt, .csv o .tsv directamente a R se puede realizar una gran cantidad de operaciones sobre los mismos de manera más sencilla y sin necesidad de hacer uso de este objeto.

Tambien es importante decir, respecto al preprocesado de los datos, que este proceso depende enormenente de la estandarización de la base de datos o repositorio de donde se obtengan. Una vez se comprende la estructura de un SummarizedExperiment es sencillo crear, componer o modificar datos propios para adaptarlos a la estructura del objeto pero este proceso se complica cuando requiere de obtener los datos de una base de datos ajena.

En este caso el repositorio de MetabolomicsWorkbench presenta los resultados en un formato relativamente accesible que permite adaptarlos con cierta facilidad a los requisitos del objeto pero esto no ocurre con los ficheros que contienen la información de los metadatos. Una caracteristica a destacar de estos ficheros es que se reparten en multiples lineas los títulos y resúmenes, repitiendo en la primera columna el mismo nombre, lo cual imposibilita la lectura por parte de R.

Adicionalmente, las muestras tambien tienen una estructura similar, donde la primera columna simplemente contiene el nombre “muestra” y su identificador pasa a la segunda columna. Esta forma de organizar el documento hace necesaria una modificación manual del mismo para lograr que RStudio pueda cargar su contenido y que posteriormente se pueda cargar al objeto SummarizedExperiment.

6. Conclusiones.

Finalmente, a modo de conclusión, el trabajo ha demostrado que, pese a requerir de ciertos conocimientos prácticos, el objeto SummarizedExperiment es una herramienta valiosa para el análisis de datos bioinformáticos de estudios propios o ajenos con gran versatilidad y capacidad para trabajar en multiples dimensiones de datos.

De manera paralela es posible afirmar que MetabolomicWorkbench es una base de datos muy bien estructurada y unificada que permite el acceso a la información de estudios de diversa indole de manera rápida, aunque la manera de estructurar esta información puede ser optimizada.

7. Referencias.

1. Base de datos de MetabolomicWorkbench: https://www.metabolomicsworkbench.org/
2. Estudio seleccionado: https://www.metabolomicsworkbench.org/data/DRCCMetadata.php?Mode=Study&StudyID=ST000875
3. Artículo sobre Nesterenkonia flava: https://pubmed.ncbi.nlm.nih.gov/28335419/
4. Web de Bioconductor sobre el objeto SummarizedExperiment: https://bioconductor.org/packages/release/bioc/vignettes/SummarizedExperiment/inst/doc/SummarizedExperiment.html#row-regions-of-interest-data
5. Web de Bioconductor sobre el análisis del objeto SummarizedExperiment: https://bioconductor.org/help/course-materials/2019/BSS2019/04_Practical_CoreApproachesInBioconductor.html

6. Web de RDocumentation sobre SummarizedExperiment: https://www.rdocumentation.org/packages/SummarizedExperiment/versions/1.2.3/topics/SummarizedExperiment-class
7. Enlace al github de la PEC: https://github.com/JMatosGar/Matos-Garcia-Javier-PEC1.git