

Prueba_Actividad_1

Mar Balanzá Rodríguez

2024-03-01

Resolución de la pregunta 1

Actualmente, estoy cursando el primer año de mi tesis que versa sobre las DHR, que incluyen la retinosis pigmentaria. Esta circunstancia ha condicionado mi elección del artículo de González-del Pozo *et al.* (2020), dado que he optado por una propuesta que se asemeja a mi propio proyecto de tesis.

- **Título:** *Unmasking Retinitis Pigmentosa complex cases by a whole genome sequencing algorithm based on open-access tools: hidden recessive inheritance and potential oligogenic variants*
- **Referencia del artículo (DOI):** 10.1186/s12967-020-02258-3
- **Objetivo y ámbito de aplicación:** Diagnóstico molecular de distrofias de la retina (DHR). Aplicación en clínica.
- **Tecnología ómica empleada:** *Whole Genome Sequencing* (Genómica)
- **Abstract:** *Retinitis Pigmentosa (RP) is a clinically and genetically heterogeneous disorder that results in inherited blindness. Despite the large number of genes identified, only 60% of cases receive a genetic diagnosis using targeted-sequencing. The aim of this study was to design a whole genome sequencing (WGS) based approach to increase the diagnostic yield of complex Retinitis Pigmentosa cases.*
- **Adecuación de la tecnología ómica a los objetivos del proyecto**

Las DHR son un conjunto de trastornos caracterizados por la muerte, generalmente, progresiva de fotorreceptores y células del epitelio pigmentario, conllevando una pérdida de la función visual. Esta última puede desembocar en la ceguera legal. Las DHR están catalogadas como enfermedades raras, puesto que su prevalencia oscila entre 1:3000 y 1:4000 personas. Aunque, son la primera causa de ceguera entre niños y adultos jóvenes.

Estas patologías se caracterizan por su enorme heterogeneidad clínica y genética. A este respecto, hasta la fecha, se han descrito 280 genes asociados a las DHR. El advenimiento de la secuenciación masiva (en inglés *Next Generation Sequencing*) ha supuesto un gran avance de cara al incremento del porcentaje de diagnóstico molecular. No obstante, aún restan muchos pacientes a la espera de recibir su diagnóstico genético, que permitiría: a) la confirmación de la sospecha clínica, b) el asesoramiento genético y reproductivo de las familias y c) la inclusión en ensayos clínicos basados en la terapia génica.

A modo de referencia, en mi grupo de investigación, se ha estudiado a más de 260 familias españolas mediante múltiples versiones de un panel de genes (que no es otra cosa que un custom microarray), alcanzándose un porcentaje diagnóstico de alrededor del 65%. El artículo de García-Bohórquez *et al.* (2021) que he adjuntado es un ejemplo de una publicación derivada de la aplicación de una estrategia de *target sequencing*.

El estudio de un panel de genes presenta claras limitaciones, ya que ni podemos detectar variantes en nuevos genes candidatos ni examinar las variantes estructurales o aquellas que recaen en regiones no codificantes. Como resultado, los estudios del exoma completo (*Whole Exome Sequencing*) o del genoma completo (*Whole Genome Sequencing*) se emplean cada vez más para resolver los casos que permanecen sin diagnóstico tras el análisis inicial con un panel de genes. De hecho, este razonamiento queda reflejado en la discusión planteada por González-del Pozo *et al.* (2020).

Resolución de la pregunta 2

```
# Si no se dispone del paquete "GEOquery"
if (!require("GEOquery")){
  # Instala el paquete "GEOquery"
  BiocManager::install("GEOquery")
}
```

```
## Loading required package: GEOquery
```

```
## Loading required package: Biobase
```

```
## Loading required package: BiocGenerics
```

```
##
```

```
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##      colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##      get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##      match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##      Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##      table, tapply, union, unique, unsplit, which.max, which.min
```

```
## Welcome to Bioconductor
```

```
##
```

```
##      Vignettes contain introductory material; view with
```

```
##      'browseVignettes()'. To cite Bioconductor, see
```

```
##      'citation("Biobase)"', and for packages 'citation("pkgname)"'.
```

```
## Setting options('download.file.method.GEOquery'='auto')
```

```
## Setting options('GEOquery.inmemory.gpl'=FALSE)
```

```
# Carga el paquete "GEOquery"
library("GEOquery")
# Crea un objeto GDS que albergue el DataSet sugerido en el ejercicio
GDS2294<-getGEO(GEO="GDS2294", destdir=getwd(), AnnotGPL=TRUE)
```

```
## Using locally cached version of GDS2294 found here:
```

```
## C:/Users/marba/OneDrive/Escritorio/Master UOC/Análisis_Datos_Omicos/Prueba_Actividad_1/GDS2294.soft.
```

```
## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
```

```
## de serie del volumen es: 486F-DD03' to a wide string
```

```
## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid

## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 486F-DD03' to a wide string

## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid

## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 486F-DD03' to a wide string

## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid
```

```
# Número total de muestras consideradas en el DataSet
Meta(GDS2294)$sample_count
```

```
## [1] "6"
```

```
# Muestras consideradas dentro de este DataSet
Meta(GDS2294)$sample_id
```

```
## [1] "GSM130365,GSM130366,GSM130367" "GSM130368,GSM130369,GSM130370"
```

```
# Plataforma utilizada para el análisis de las siguientes muestras
Meta(GDS2294)$platform
```

```
## [1] "GPL81"
```

```
# Nombres de las variables consideradas en la tabla de
names(Columns(GDS2294))
```

```
## [1] "sample" "genotype/variation" "description"
```

```
# Cabecera de la tabla de datos fenotípicos asociada a este DataSet
head(Columns(GDS2294))
```

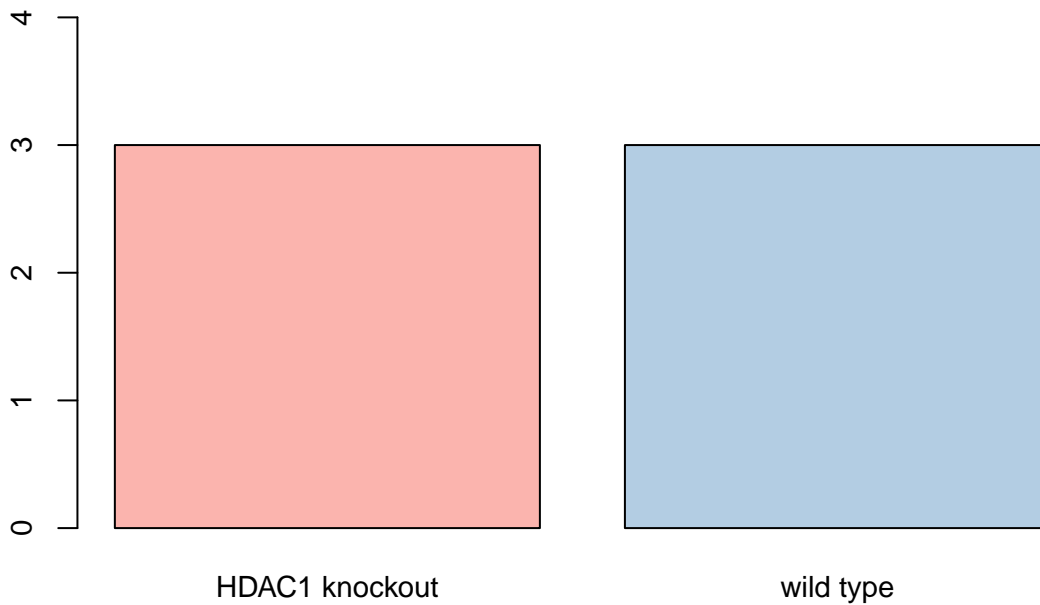
```
##      sample genotype/variation
## 1 GSM130365      wild type
## 2 GSM130366      wild type
## 3 GSM130367      wild type
## 4 GSM130368  HDAC1 knockout
## 5 GSM130369  HDAC1 knockout
## 6 GSM130370  HDAC1 knockout
##
## 1      Value for GSM130365: wild_type_embryonic_stem cells_biological_rep1; src: Mus musculus
## 2      Value for GSM130366: wild_type_embryonic_stem cells_biological_rep2; src: Mus musculus
## 3      Value for GSM130367: wild_type_embryonic_stem cells_biological_rep3; src: Mus musculus
## 4 Value for GSM130368: HDAC1_knock_out_embryonic_stem_cells_biological_rep1; src: Mus musculus embry
## 5 Value for GSM130369: HDAC1_knock_out_embryonic_stem_cells_biological_rep2; src: Mus musculus embry
## 6 Value for GSM130370: HDAC1_knock_out_embryonic_stem_cells_biological_rep3; src: Mus musculus embry
```

```
# Muestras de cada tipo incluidas en el estudio
table(Columns(GDS2294)$"genotype/variation")
```

```
##
## HDAC1 knockout      wild type
##           3           3
```

```
# Diagrama de barras de la variable "genotype/variation"
```

```
barplot(table(Columns(GDS2294)$"genotype/variation"), col=palette.colors(n=2, palette="Pastel 1"), cex.
```



```
# Convierte el objeto GDS a un objeto de tipo ExpressionSet
expr_GDS2294<-GDS2eSet(GDS2294)
```

```
## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 486F-DD03' to a wide string
```

```
## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid
```

```
## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 486F-DD03' to a wide string
```

```
## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid
```

```
## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 486F-DD03' to a wide string
```

```
## Warning in grep(pattern, bfr, value = TRUE): input string 2 is invalid
```

```
# Dimensiones de la matriz de expresión
dim(exprs(expr_GDS2294))
```

```
## [1] 12488      6
```

```
# Cabecera de la matriz de expresión
head(exprs(expr_GDS2294))
```

```
##          GSM130365 GSM130366 GSM130367 GSM130368 GSM130369 GSM130370
## 100001_at      11.5      5.6      69.1      15.7      36.0      42.0
## 100002_at      20.5     32.4     93.3     31.8     14.4     22.9
## 100003_at      72.4     89.0     79.2     80.5    130.1     86.7
## 100004_at     261.0    226.2    365.1    432.0    447.3    288.1
## 100005_at    1086.2   1555.6   1487.1   1062.2   1365.9   1436.2
## 100006_at      49.7     52.9     15.0     25.8     48.8     54.8
```

```
# Dimensiones de la matriz de datos fenotípicos
dim(pData(expr_GDS2294))
```

```
## [1] 6 3
```

```
# Cabecera de la matriz de datos fenotípicos
head(pData(expr_GDS2294))
```

```
##          sample genotype/variation
## GSM130365 GSM130365      wild type
## GSM130366 GSM130366      wild type
## GSM130367 GSM130367      wild type
## GSM130368 GSM130368  HDAC1 knockout
## GSM130369 GSM130369  HDAC1 knockout
## GSM130370 GSM130370  HDAC1 knockout
##
## GSM130365      Value for GSM130365: wild_type_embryonic_stem cells_biological_rep1; src: Mus musculus
## GSM130366      Value for GSM130366: wild_type_embryonic_stem cells_biological_rep2; src: Mus musculus
## GSM130367      Value for GSM130367: wild_type_embryonic_stem cells_biological_rep3; src: Mus musculus
## GSM130368 Value for GSM130368: HDAC1_knock_out_embryonic_stem_cells_biological_rep1; src: Mus musculus
## GSM130369 Value for GSM130369: HDAC1_knock_out_embryonic_stem_cells_biological_rep2; src: Mus musculus
## GSM130370 Value for GSM130370: HDAC1_knock_out_embryonic_stem_cells_biological_rep3; src: Mus musculus
```

```
# Verifica que las filas de la matriz de datos fenotípicos se corresponden con las columnas de la matriz de expresión
all(rownames(pData(expr_GDS2294))==colnames(exprs(expr_GDS2294)))
```

```
## [1] TRUE
```