

Reporte del Análisis de la expresión diferencial de células endoteliales dérmicas de pacientes diabéticos tipo 2

Edna Karen Rivera Zagal

2025-02-09

Introducción

La prevalencia de diabetes mellitus tipo 2 (DT2) aumenta constantemente y se han identificado diversos factores de riesgo como la obesidad, el envejecimiento, los estados nutricionales y la inactividad física, además de predisposiciones genéticas en diferentes poblaciones. Las consecuencias de la alta glucosa en sangre incluyen vasos sanguíneos dañados, lo que lleva a arteriosclerosis y microangiopatías diabéticas crónicas.[1]

Para obtener las muestras y realizar el análisis con los datos obtenidos se aislaron células endoteliales dérmicas de pacientes diabéticos (Pat) e individuos de control (Ctrl) y se realizaron RNASeq para comparar genes expresados diferencialmente. Diseño general: Se tomaron muestras quirúrgicas de piel humana de pacientes diabéticos tipo 2 y pacientes no diabéticos.[1]

Este proyecto tiene como objetivo analizar la expresión diferencial de genes en células endoteliales de personas diabéticas y sanas, utilizando datos de RNA-seq obtenidos a través del paquete Recount3[2] de biocreador[3]. Utilizamos herramientas bioinformáticas para identificar genes diferencialmente expresados y visualizamos los resultados con gráficos y heatmaps. Con la finalidad de tener una propuesta biológica de los cambios observados en la expresión de los genes.

Datos

- Número de identificación de los datos en Recount3 : **SRP095512**
- GEO accession[4] a los datos : **GSE92724**

Cargar los paquetes necesarios para todo el análisis

```
library("recount3")

## Cargando paquete requerido: SummarizedExperiment

## Cargando paquete requerido: MatrixGenerics

## Cargando paquete requerido: matrixStats

##
## Adjuntando el paquete: 'MatrixGenerics'
```

```

## The following objects are masked from 'package:matrixStats':
##
##      colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##      colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##      colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##      colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##      colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##      colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##      colWeightedMeans, colWeightedMedians, colWeightedSds,
##      colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##      rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##      rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##      rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##      rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##      rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##      rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##      rowWeightedSds, rowWeightedVars

## Cargando paquete requerido: GenomicRanges

## Cargando paquete requerido: stats4

## Cargando paquete requerido: BiocGenerics

##
## Adjuntando el paquete: '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, saveRDS, setdiff,
##      table, tapply, union, unique, unsplit, which.max, which.min

## Cargando paquete requerido: S4Vectors

##
## Adjuntando el paquete: 'S4Vectors'

## The following object is masked from 'package:utils':
##
##      findMatches

## The following objects are masked from 'package:base':
##
##      expand.grid, I, unname

```

```

## Cargando paquete requerido: IRanges

##
## Adjuntando el paquete: 'IRanges'

## The following object is masked from 'package:grDevices':
##
##      windows

## Cargando paquete requerido: GenomeInfoDb

## Cargando paquete requerido: Biobase

## Welcome to Bioconductor
##
##      Vignettes contain introductory material; view with
##      'browseVignettes()'. To cite Bioconductor, see
##      'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Adjuntando el paquete: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##      rowMedians

## The following objects are masked from 'package:matrixStats':
##
##      anyMissing, rowMedians

library("edgeR") # BiocManager::install("edgeR", update = FALSE)

## Cargando paquete requerido: limma

##
## Adjuntando el paquete: 'limma'

## The following object is masked from 'package:BiocGenerics':
##
##      plotMA

library("ggplot2")
library("limma")
library("pheatmap")
library("RColorBrewer")

```

Obtención y análisis de los datos

```

## Revisemos todos los proyectos con datos de humano en recount3
human_projects <- available_projects()

## 2025-02-08 21:32:54.692058 caching file sra.recount_project.MD.gz.

## 2025-02-08 21:32:55.658173 caching file gtex.recount_project.MD.gz.

## 2025-02-08 21:32:56.488967 caching file tcga.recount_project.MD.gz.

## Identificador de mi proyecto de interes:SRP095512
##Obtencion del proyecto de interes
proj_info <- subset(
  human_projects,
  project == "SRP095512" & project_type == "data_sources"
)

## create_rse Crea un objeto de tipo RangedSummarizedExperiment (RSE)
## con la informaci&on a nivel de genes
rse_gene_SRP095512 <- create_rse(proj_info)

## 2025-02-08 21:33:15.347076 downloading and reading the metadata.

## 2025-02-08 21:33:16.394942 caching file sra.sra.SRP095512.MD.gz.

## 2025-02-08 21:33:17.280896 caching file sra.recount_project.SRP095512.MD.gz.

## 2025-02-08 21:33:18.270535 caching file sra.recount_qc.SRP095512.MD.gz.

## 2025-02-08 21:33:19.116443 caching file sra.recount_seq_qc.SRP095512.MD.gz.

## 2025-02-08 21:33:20.015342 caching file sra.recount_pred.SRP095512.MD.gz.

## 2025-02-08 21:33:20.333601 downloading and reading the feature information.

## 2025-02-08 21:33:20.930826 caching file human.gene_sums.G026.gtf.gz.

## 2025-02-08 21:33:22.892489 downloading and reading the counts: 10 samples across 63856 features.

## 2025-02-08 21:33:23.473634 caching file sra.gene_sums.SRP095512.G026.gz.

## Warning in grep(pattern, bfr, value = TRUE): unable to translate ' El n<a3>mero
## de serie del volumen es: 8456-AF82' 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: 8456-AF82' 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: 8456-AF82' to a wide string

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

## 2025-02-08 21:33:25.54769 constructing the RangedSummarizedExperiment (rse) object.

## Explorar el objeto RSE
rse_gene_SRP095512

## class: RangedSummarizedExperiment
## dim: 63856 10
## metadata(8): time_created recount3_version ... annotation recount3_url
## assays(1): raw_counts
## rownames(63856): ENSG00000278704.1 ENSG00000277400.1 ...
##   ENSG00000182484.15_PAR_Y ENSG00000227159.8_PAR_Y
## rowData names(10): source type ... havana_gene tag
## colnames(10): SRR5125028 SRR5125029 ... SRR5125036 SRR5125037
## colData names(175): rail_id external_id ...
##   recount_pred.curated.cell_line BigWigURL

## Convirtamos las cuentas por nucleotido a cuentas por lectura
## usando compute_read_counts().
assay(rse_gene_SRP095512, "counts") <- compute_read_counts(rse_gene_SRP095512)

## Visualizar la informacion de los atributos del objeto "rse_gene_SRP095512"

rse_gene_SRP095512 <- expand_sra_attributes(rse_gene_SRP095512)
colData(rse_gene_SRP095512)[
  ,
  grepl("^sra_attribute", colnames(colData(rse_gene_SRP095512)))
]

## DataFrame with 10 rows and 4 columns
##           sra_attribute.cell_type sra_attribute.disease_state
##           <character>          <character>
## SRR5125028      endothelial cell      Healthy control
## SRR5125029      endothelial cell      Diabetic Patient
## SRR5125030      endothelial cell      Diabetic Patient
## SRR5125031      endothelial cell      Diabetic Patient
## SRR5125032      endothelial cell      Diabetic Patient
## SRR5125033      endothelial cell      Healthy control
## SRR5125034      endothelial cell      Healthy control
## SRR5125035      endothelial cell      Healthy control
## SRR5125036      endothelial cell      Healthy control
## SRR5125037      endothelial cell      Healthy control
##           sra_attribute.gender sra_attribute.source_name
##           <character>          <character>
## SRR5125028        female    dermal blood endothe..
## SRR5125029        male     dermal blood endothe..

```

```

## SRR5125030      female    dermal blood endothe..
## SRR5125031      female    dermal blood endothe..
## SRR5125032      male     dermal blood endothe..
## SRR5125033      female    dermal blood endothe..
## SRR5125034      female    dermal blood endothe..
## SRR5125035      female    dermal blood endothe..
## SRR5125036      female    dermal blood endothe..
## SRR5125037      female    dermal blood endothe..

## Formato correcto en el que debemos tener la informacion
## Pasar de character a numeric o factor
rse_gene_SRP095512$sra_attribute.disease_state <- factor(tolower(rse_gene_SRP095512$sra_attribute.disease_state))

rse_gene_SRP095512$sra_attribute.gender <- factor(rse_gene_SRP095512$sra_attribute.gender)

## Resumen de las variables de interés
summary(as.data.frame(colData(rse_gene_SRP095512)[
  ,
  grepl("sra_attribute.[gender|disease_state]", colnames(colData(rse_gene_SRP095512))]))
])

##      sra_attribute.disease_state sra_attribute.gender sra_attribute.source_name
##  diabetic patient:4           female:8             Length:10
##  healthy control :6           male  :2              Class :character
##                                         Mode  :character

# Checar la calidad de los datos
rse_gene_SRP095512$assigned_gene_prop <-
  rse_gene_SRP095512$recount_qc.gene_fc_count_all.assigned /
  rse_gene_SRP095512$recount_qc.gene_fc_count_all.total
summary(rse_gene_SRP095512$assigned_gene_prop)

##      Min. 1st Qu. Median   Mean 3rd Qu.   Max.
##  0.3665  0.4902  0.5349  0.5189  0.5608  0.6074

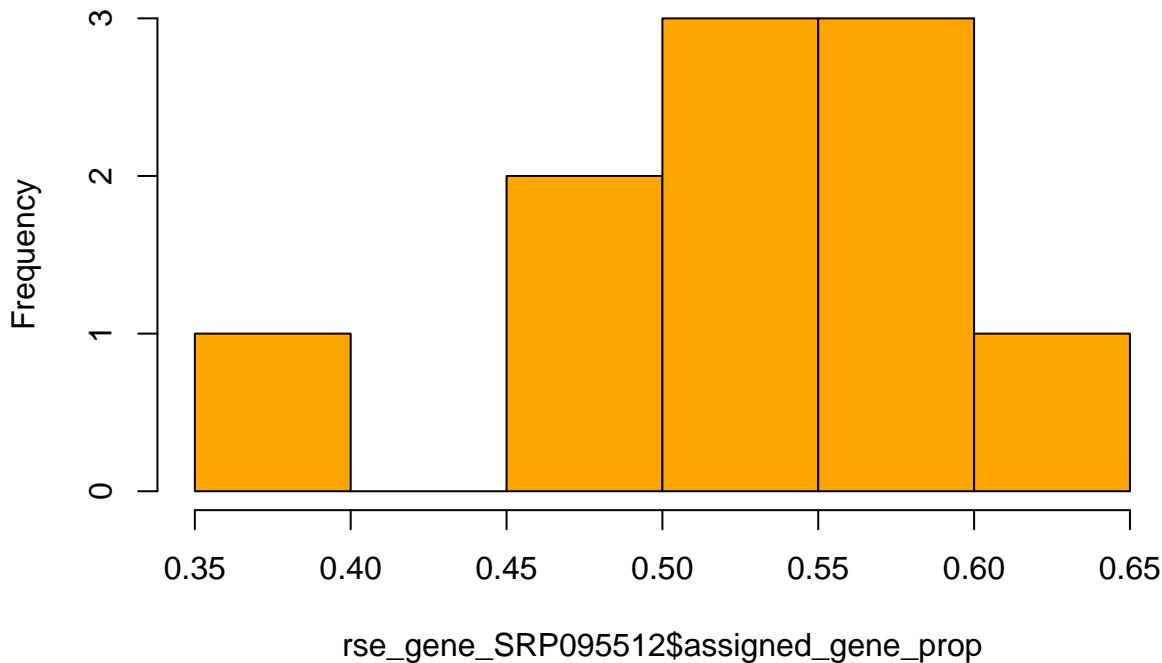
# Observar la diferencia entre las muestras
with(colData(rse_gene_SRP095512),
  tapply(assigned_gene_prop,
    sra_attribute.disease_state,
    summary))

## $`diabetic patient`
##      Min. 1st Qu. Median   Mean 3rd Qu.   Max.
##  0.4544  0.5260  0.5503  0.5298  0.5541  0.5641
##
## $`healthy control`
##      Min. 1st Qu. Median   Mean 3rd Qu.   Max.
##  0.3665  0.4902  0.5156  0.5116  0.5659  0.6074

# Grafica de la distribucion de los genes asignados
hist(rse_gene_SRP095512$assigned_gene_prop, col = "orange")

```

Histogram of rse_gene_SRP095512\$assigned_gene_prop



```
# Guardad los datos sin filtrar, en caso de querer recuperarlos despues  
rse_gene_SRP095512_unfiltered <- rse_gene_SRP095512
```

Normalización de los datos

```
# Normalización de los datos  
dge <- DGEList(  
  counts = assay(rse_gene_SRP095512, "counts"),  
  genes = rowData(rse_gene_SRP095512)  
)  
dge <- calcNormFactors(dge)
```

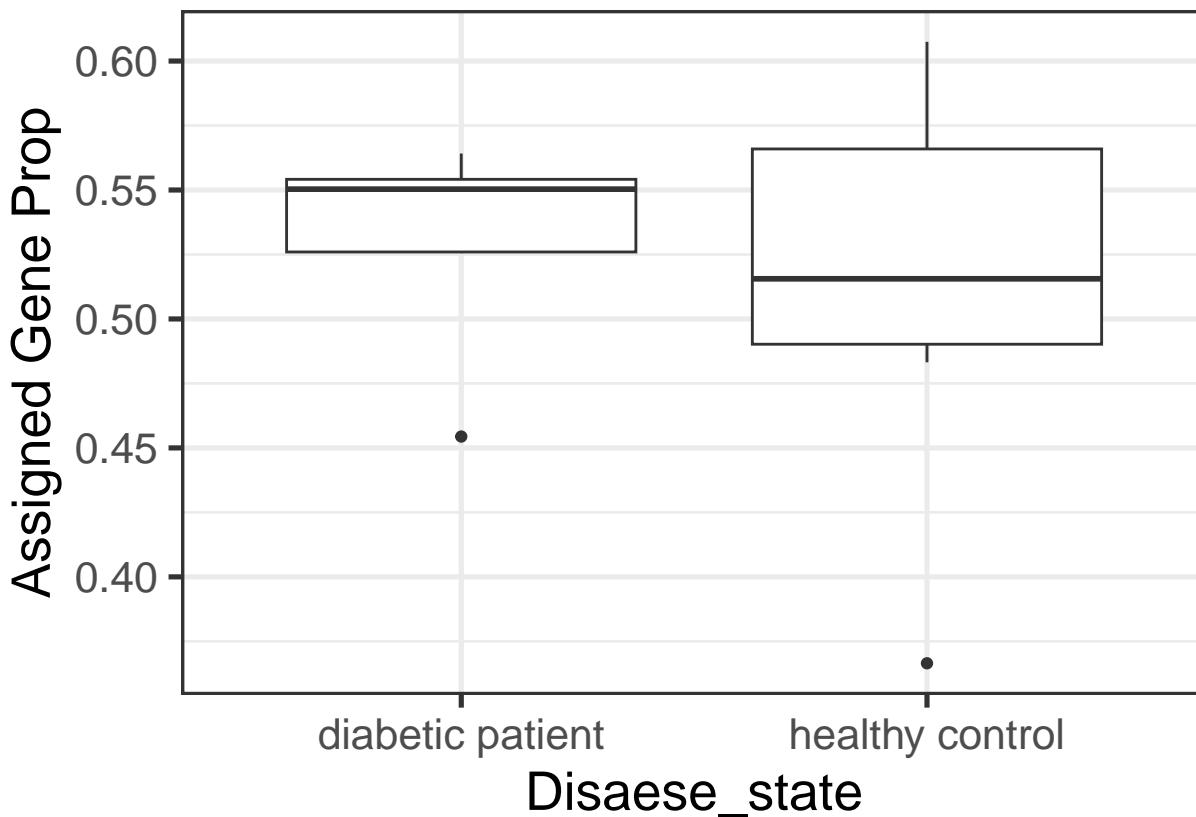
Análisis de la expresión diferencial

```
## Creacion de un boxplot para analizar la distribución de los genes en muestras  
## de pacientes enfermos y sanos  
  
ggplot(as.data.frame(colData(rse_gene_SRP095512)), aes(y = assigned_gene_prop, x = sra_attribute.disease)
```

```

ylab("Assigned Gene Prop") +
xlab("Disaese_state")

```



```

# Modelo estadístico

mod <- model.matrix(~ sra_attribute.gender +sra_attribute.disease_state + assigned_gene_prop,
                     data = colData(rse_gene_SRP095512)
)
colnames(mod)

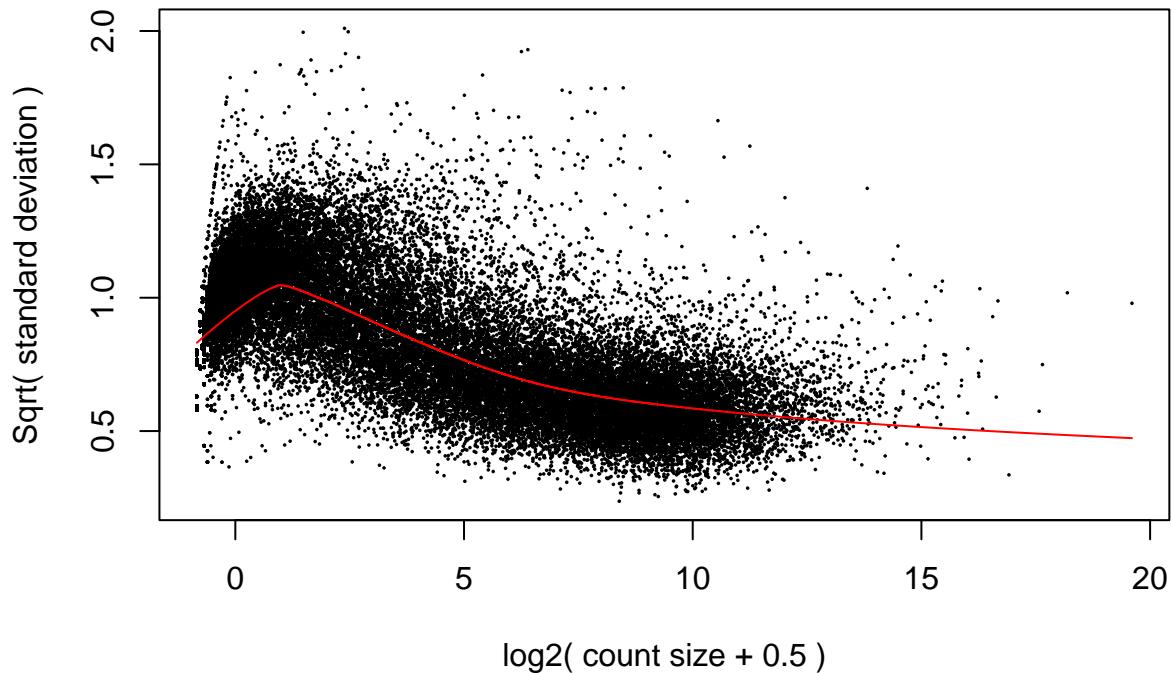
## [1] "(Intercept)"
## [2] "sra_attribute.gendermale"
## [3] "sra_attribute.disease_statehealthy control"
## [4] "assigned_gene_prop"

## Usamos limma para realizar el análisis de expresión diferencial

vGene <- voom(dge, mod, plot = TRUE)

```

voom: Mean–variance trend



```
eb_results <- eBayes(lmFit(vGene))

de_results <- topTable(
  eb_results,
  coef = 2,
  number = nrow(rse_gene_SRP095512),
  sort.by = "none"
)
dim(de_results)

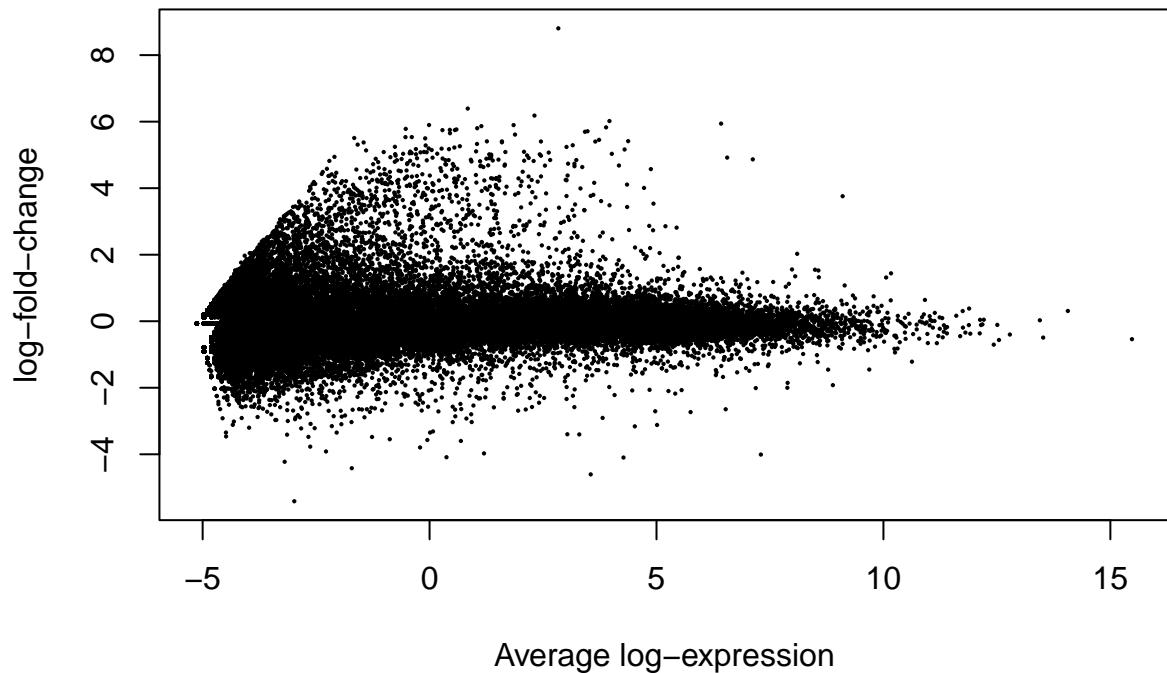
## [1] 63856     16

## Genes diferencialmente expresados con FDR < 5%
table(de_results$adj.P.Val < 0.05)

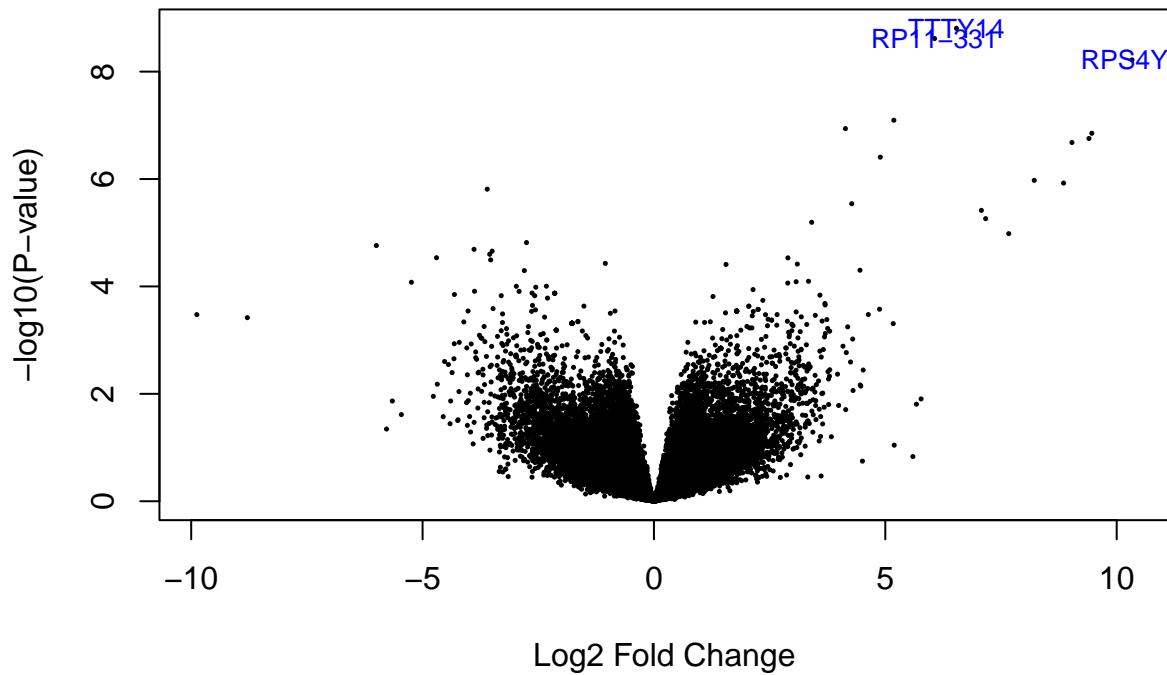
## 
## FALSE  TRUE
## 63839    17

## Visualicemos los resultados estadísticos
plotMA(eb_results, coef = 3)
```

sra_attribute.disease_statehealthy control



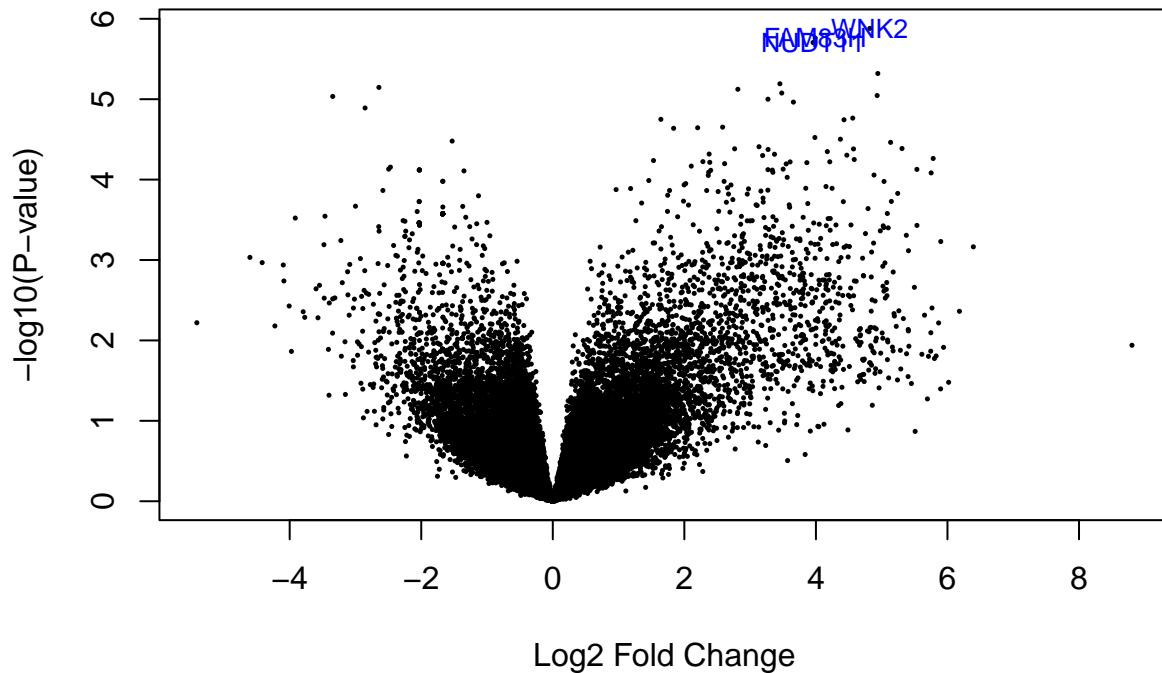
```
## Volcano plot Gender
volcanoplot(eb_results, coef = 2, highlight = 3, names = de_results$gene_name)
```



```
de_results[de_results$gene_name %in% c("TTY14", "RPS4Y1", "RP11-331"), ]
```

```
##           source type bp_length phase          gene_id
## ENSG00000129824.15 HAVANA gene      2275     NA ENSG00000129824.15
##                               gene_type gene_name level      havana_gene
## ENSG00000129824.15 protein_coding   RPS4Y1      2 OTTHUMG00000036152.4
##                               tag      logFC AveExpr       t    P.Value
## ENSG00000129824.15 overlapping_locus 10.32998 -2.382076 24.48189 6.047661e-09
##                               adj.P.Val      B
## ENSG00000129824.15 0.0001287265 5.200397
```

```
## Volcano plot Disease State
volcanoplot(eb_results, coef = 3, highlight = 3, names = de_results$gene_name)
```



```
de_results[de_results$gene_name %in% c("WNK2", "NUD111", "FAM83H"), ]
```

```
##           source type bp_length phase          gene_id
## ENSG00000273889.2 HAVANA gene      5654     NA ENSG00000273889.2
## ENSG00000180921.6 HAVANA gene      5654     NA ENSG00000180921.6
## ENSG00000165238.16 HAVANA gene     13198     NA ENSG00000165238.16
##             gene_type gene_name level          havana_gene
## ENSG00000273889.2 protein_coding FAM83H      2 OTTHUMG00000190683.1
## ENSG00000180921.6 protein_coding FAM83H      2 OTTHUMG00000133559.2
## ENSG00000165238.16 protein_coding   WNK2      1 OTTHUMG00000020247.4
##             tag      logFC AveExpr          t    P.Value
## ENSG00000273889.2 <NA> -0.16013602 -5.127480 -0.6052952 0.561392663
## ENSG00000180921.6 ncRNA_host -1.48582887 -2.016150 -3.6851075 0.005936403
## ENSG00000165238.16 <NA>  0.07310596 -2.207875  0.1486222 0.885449682
##             adj.P.Val      B
## ENSG00000273889.2  0.7356061 -5.770037
## ENSG00000180921.6  0.6400002 -1.938974
## ENSG00000165238.16 0.9355717 -5.947010
```

Visualizando la expresion diferencial a traves de un Heatmap

```
## Extraer valores de los genes de interés
exprs_heatmap <- vGene$E[rank(de_results$adj.P.Val) <= 50, ]
```

```

## Creamos una tabla con información de las muestras
## y con nombres de columnas más amigables
df <- as.data.frame(colData(rse_gene_SRP095512)[, c("sra_attribute.gender", "sra_attribute.disease_state")]
colnames(df) <- c("Gender", "disease_state")

```

```

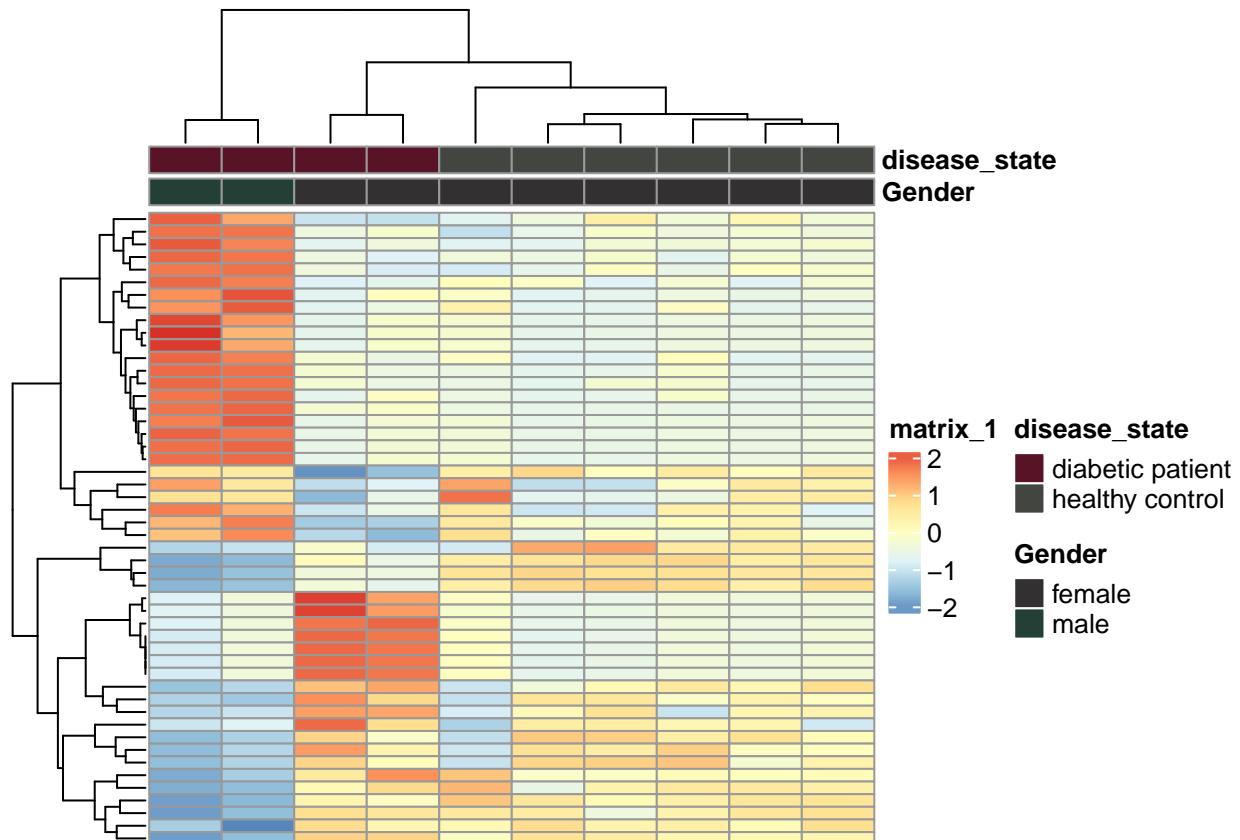
## Hagamos un heatmap

```

```

ComplexHeatmap::pheatmap(
  exprs_heatmap,
  cluster_rows = TRUE,
  cluster_cols = TRUE,
  show_rownames = FALSE,
  show_colnames = FALSE,
  annotation_col = df,
  scale = "row"
)

```



Interpretación biológica de los resultados

La expresión genética en las células epiteliales de personas con diabetes puede verse afectada por factores como la variación genética, la epigenética y la resistencia a la insulina[5]. Además de se ha comprobado que

la insulina juega un papel clave en la activación o inactivación de determinados genes, por lo que no es de extrañar que exista una diferencia en la expresión genética cuando está presente la enfermedad y cuando no.

En el análisis anterior se observó una clara diferencia en la expresión genética de las muestras con la enfermedad y las que no, contrastada con el género de los pacientes de las que fueron tomadas.

Existe una mayor expresión en los pacientes diabéticos masculinos, en comparación de los pacientes diabéticos femeninos, teniendo estos una expresión, aunque no tan débil, sí en menor cantidad de genes expresados. Esto podría indicar un efecto del sexo en la expresión génica.

Por otro lado, se observa una mayor expresión genética en las muestras de los pacientes enfermos, en comparación con los sanos. Esto sugiere que ciertos genes están regulados diferencialmente en la diabetes.

Si hay genes altamente expresados solo en diabéticos o solo en controles sanos, podrían ser biomarcadores relevantes para la enfermedad.

Bibliografía

1. <https://jhubiostatistics.shinyapps.io/recount3-study-explorer/>
2. <https://bioconductor.org/packages/release/bioc/html/genefilter.html>
3. <https://rna.recount.bio/>
4. <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE92724>
5. <https://www.dreamgenics.com/2022/04/04/papel-de-la-genetica-en-la-diabetes-mellitus/#:~:text=tipos%20de%20diabetes%20polig%C3%A9nicas,a%20desarrollar%20diabetes%20tipo%202.>