Reporte_SRP090061

SRP098758

Previo a elegir el proyecto SRP090061, se comenzó a realizar el análisis para el proyecto SRP098758: A blood RNA signature for tuberculosis disease risk in household contact study - GC6 cohort.

No se pudo completar pues existían 6 muestras no clasificadas para edad, grupo, género, sitio y tejido, por lo que en estos atributos tenian datos faltantes (NAs).

En caso de reemplazar los caracteres (”NAs”) por NAs y usar droplevels(), al intentar realizar el análisis de expresion diferencial con limma voom() obtuvimos un error, pues las dimensiones de las matrices no coinciden (6 datos no clasif.). En caso de retener los NAs para que las dimensiones de las matrices coincidan obtenemos un error pues voom() no puede leer los NAs.

Por ello, se eligió un estudio diferente.

REGION-SPECIFIC NEURAL STEM CELL LINEAGES REVEALED BY SINGLE-CELL RNA-SEQ FROM HUMAN EMBRYONIC STEM CELLS [Smart-seq]

Organism: Human

Proyect_home: data_sources/sra

Proyect: SRP090061

Abstract:

During development of the human brain, multiple cell types with diverse regional identities are generated. Here we report a system to generate early human brain forebrain and mid/hindbrain cell types from human embryonic stem cells (hESCs), and infer and experimentally confirm a lineage tree for the generation of these types based on single-cell RNA-Seq analysis. We engineered SOX2Cit/+ and DCXCit/Y hESC lines to target progenitors and neurons throughout neural differentiation for single-cell transcriptomic profiling, then identified discrete cell types consisting of both rostral (cortical) and caudal (mid/hindbrain) identities. Direct comparison of the cell types were made to primary tissues using gene expression atlases and fetal human brain single-cell gene expression data, and this established that the cell types resembled early human brain cell types, including preplate cells. From the single-cell transcriptomic data a Bayesian algorithm generated a unified lineage tree, and predicted novel regulatory transcription factors. The lineage tree highlighted a prominent bifurcation between cortical and mid/hindbrain cell types, confirmed by clonal analysis experiments. We demonstrated that cell types from either branch could preferentially generated by manipulation of the canonical Wnt/beta-catenin pathway. In summary, we present an experimentally validated lineage tree that encompasses multiple brain regions, and our work sheds light on the molecular regulation of region-specific neural lineages during human brain development. During development of the human brain, multiple cell types with diverse regional identities are generated. Here we report a system to generate early human brain forebrain and mid/hindbrain cell types from human embryonic stem cells (hESCs), and infer and experimentally confirm a lineage tree for the generation of these types based on single-cell RNA-Seq analysis. We engineered SOX2Cit/+ and DCXCit/Y hESC lines to target progenitors and neurons throughout neural differentiation for single-cell transcriptomic profiling, then identified discrete cell types consisting of both rostral (cortical) and caudal (mid/hindbrain) identities. Direct comparison of the cell types were made to primary tissues using gene expression atlases and fetal human brain single-cell gene expression data, and this established that the cell types resembled early human brain cell types, including preplate cells. From the single-cell transcriptomic data a Bayesian algorithm generated a unified lineage tree, and predicted novel regulatory transcription factors. The lineage tree highlighted a prominent bifurcation between cortical and mid/hindbrain cell types, confirmed by clonal analysis experiments. We demonstrated that cell types from either branch could preferentially generated by manipulation of the canonical Wnt/beta-catenin pathway. In summary, we present an experimentally validated lineage tree that encompasses multiple brain regions, and our work sheds light on the molecular regulation of region-specific neural lineages during human brain development. During development of the human brain, multiple cell types with diverse regional identities are generated. Here we report a system to generate early human brain forebrain and mid/hindbrain cell types from human embryonic stem cells (hESCs), and infer and experimentally confirm a lineage tree for the generation of these types based on single-cell RNA-Seq analysis. We engineered SOX2Cit/+ and DCXCit/Y hESC lines to target progenitors and neurons throughout neural differentiation for single-cell transcriptomic profiling, then identified discrete cell types consisting of both rostral (cortical) and caudal (mid/hindbrain) identities.

Direct comparison of the cell types were made to primary tissues using gene expression atlases and fetal human brain single-cell gene expression data, and this established that the cell types resembled early human brain cell types, including preplate cells. From the single-cell transcriptomic data a Bayesian algorithm generated a unified lineage tree, and predicted novel regulatory transcription factors. The lineage tree highlighted a prominent bifurcation between cortical and mid/hindbrain cell types, confirmed by clonal analysis experiments. We demonstrated that cell types from either branch could preferentially generated by manipulation of the canonical Wnt/beta-catenin pathway. In summary, we present an experimentally validated lineage tree that encompasses multiple brain regions, and our work sheds light on the molecular regulation of region-specific neural lineages during human brain development. Overall design: The transcriptomes of 1846 single cells were profiled by SmartSeq2 at different timepoints throughout a 54-day differentiation protocol that converted H1 human embryonic stem cells to a variety of brain cell types. Some cells were positively labeled by a expression of a barcoded viral transgene to help establish clonality (marked by an "SK").

Read_and_explore_data

La descarga de datos fue realizada a través del paquete recount3 de Bioconductor, con los que se construyó un objeto de tipo RangedSummarizedExperiment (RSE) con la información a nivel de genes:

```
## Proyectos con datos de humano en recount3
human_projects <- available_projects()
## Usamos el proyecto SRP090061
proj_info <- subset(
  human_projects,
  project == "SRP090061" & project_type == "data_sources"
)
## Se crea un objeto RSE
rse_gene <- create_rse(proj_info)
```

Exploramos los datos y adaptamos la información convirtiendo las cuentas por nucleotido a cuentas por lectura usando compute_read_counts(). Además facilitamos la lectura de la información del experimento:

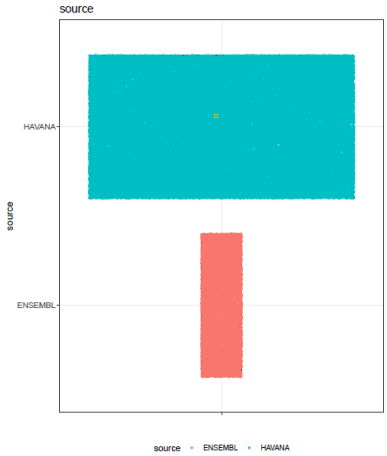
```
## Convertimos las cuentas por nucleotido a cuentas por lectura
assay(rse_gene, "counts") <- compute_read_counts(rse_gene)

## Facilitamos la lectura de la información
rse_gene <- expand_sra_attributes(rse_gene)
colData(rse_gene)[
  ,
  grepl("^sra_attribute", colnames(colData(rse_gene)))
]
```

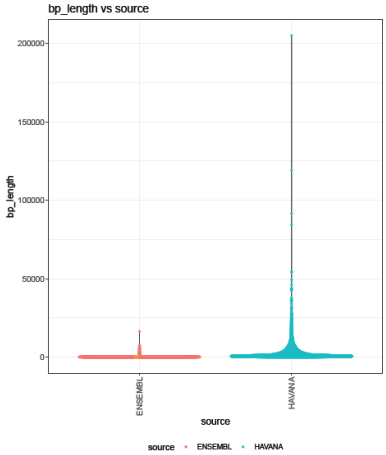
Contamos con los siguientes atributos:

- **control** - sra_attribute.control
- **cre_line** - sra_attribute.cre_line
- **days_in_culture** - sra_attribute.days_in_culture
- **source_name** - sra_attribute.source_name
- **barcoded** - sra_attribute.viral_barcoded

Exploración gráfica con iSEE

```
library("iSEE")
iSEE::iSEE(rse_gene)
```





Con la interfaz gráfica ofrecida por iSEE creamos 2 imágenes: en la primera observamos la fuente de los datos de nuestro objeto y en la segunda observamos la longitud de las secuencias (bp) dependiendo de la fuente.

Limpieza y normalización de datos

Calculamos la proporción de genes asignados, donde valores más cercanos a 1 representan una mayor asignación de lecturas-genes (mayor calidad)

```
## Proporcion de genes para cada grupo
rse_gene$assigned_gene_prop <- rse_gene$recount_qc.gene_fc_count_all.assigned / rse_gene$recount_qc.gene_fc_count_all.total
```

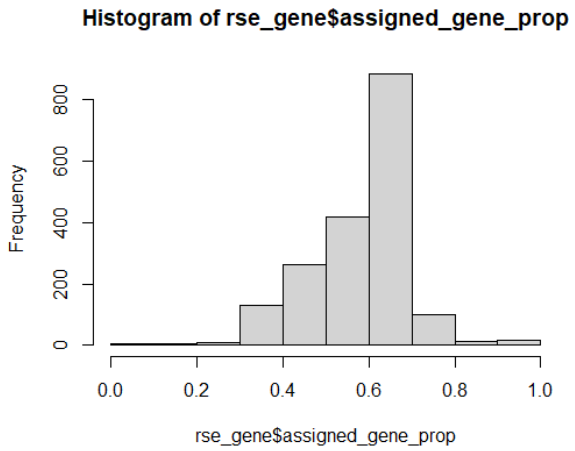
Con ella creamos un histograma con la proporcion de genes y su frecuencia.

Posteriormente tomamos un cutoff value de 0.3, eliminando las muestras de menor calidad. Esto es posible ya que no se encuentran en gran proporción, por lo que no perderemos una cantidad de información considerable.

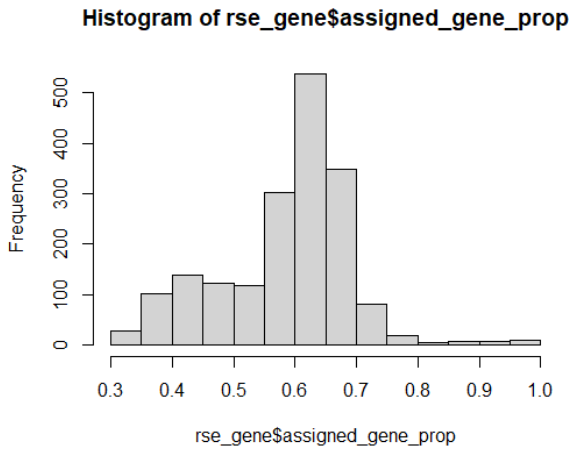
Además, obtenemos los niveles medios de expresión de los genes (en las muestras), y nos quedamos sólo con aquellos que tienen una expresión mayor a 0.1.

```
## Creamos un histograma con la proporcion de genes
hist(rse_gene$assigned_gene_prop)

## Eliminamos las muestras malas
table(rse_gene$assigned_gene_prop < 0.3)
rse_gene <- rse_gene[, rse_gene$assigned_gene_prop > 0.3]

## Niveles medios de expresión de los genes en las muestras
gene_means <- rowMeans(assay(rse_gene, "counts"))

## Eliminamos genes
rse_gene <- rse_gene[gene_means > 0.1, ]

## Creamos un histograma con la proporcion de genes despues de la limpieza de datos
hist(rse_gene$assigned_gene_prop)
```

Después del filtrado, creamos un nuevo histograma con la proporcion de genes y su frecuencia.



Unfiltered



Filtered

Para la normalización de datos usamos calcNormFactors() en un objeto creado con DGEList() de la librería “edgeR”

```
######################### Normalización de datos ############################

library("edgeR")

dge <- DGEList(
  counts = assay(rse_gene, "counts"),
  genes = rowData(rse_gene)
)

dge <- calcNormFactors(dge)
```

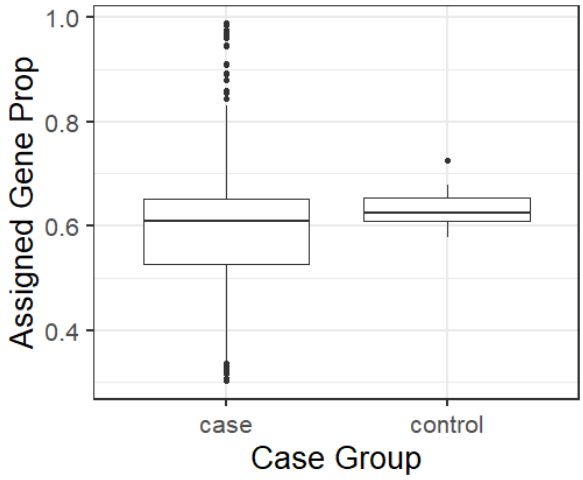
Definiendo el modelo estadístico

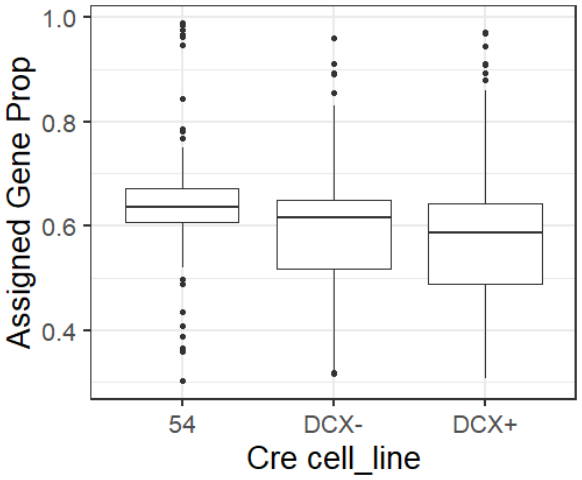
Para ayudarnos a elegir un modelo estadístico, generamos algunas graficas usando ggplot2:

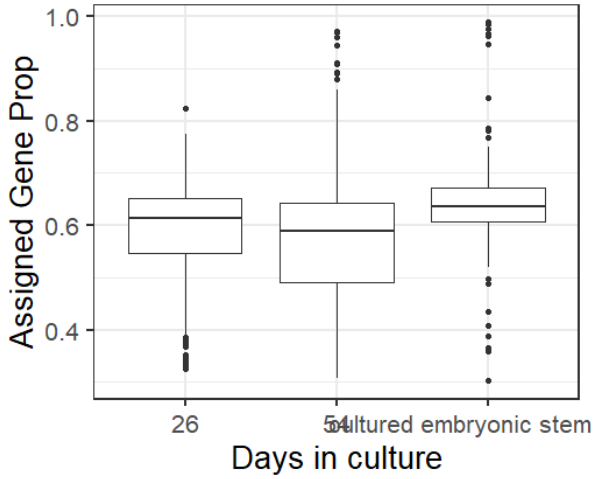
- Boxplot para comparar por casos y controles
- Boxplot para comparar por cre_line
- Boxplot para comparar por days_in_culture
- Boxplot para comparar por source

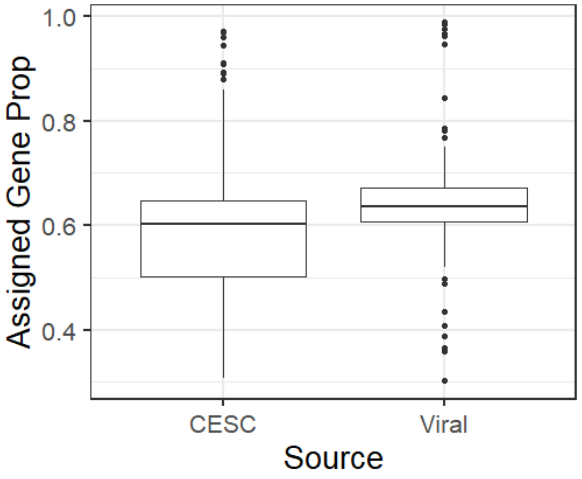
```
library("ggplot2")

## Boxplot para comparar por casos y controles
ggplot(as.data.frame(colData(rse_gene)), aes(y = assigned_gene_prop, x = case)) +
  geom_boxplot() +
  theme_bw(base_size = 20) +
  ylab("Assigned Gene Prop") +
  xlab("Case Group")

## Boxplot para comparar por cre_line
ggplot(as.data.frame(colData(rse_gene)), aes(y = assigned_gene_prop, x = sra_attribute.cre_line)) +
  geom_boxplot() +
  theme_bw(base_size = 20) +
  ylab("Assigned Gene Prop") +
  xlab("Cre cell_line")

## Boxplot para comparar por days_in_culture
ggplot(as.data.frame(colData(rse_gene)), aes(y = assigned_gene_prop, x = sra_attribute.days_in_culture)) +
  geom_boxplot() +
```

```
  theme_bw(base_size = 20) +
  ylab("Assigned Gene Prop") +
  xlab("Days in culture")

## Boxplot para comparar por source
ggplot(as.data.frame(colData(rse_gene)), aes(y = assigned_gene_prop, x = source)) +
  geom_boxplot() +
  theme_bw(base_size = 20) +
  ylab("Assigned Gene Prop") +
  xlab("Source")
```









No existe una gran diferencia entre las medias de todos los genes de los distintos grupos según cada atributo. Si bien existe una diferencia en cada uno de ellos, a mi parecer ninguno es visualmente (ni numéricamente) particularmente el atributo de mayor peso.

La elección del modelo a usar se definió además de por la comparación gráfica y numérica dependiendo del peso biologico que cada una supondría representar.

El atributo de control me parece el más relevante pues las diferencias más significativa debería estar dada por las diferencias entre casos y controles. La siguiente variable más significativa sería la linea celular usada, pues estas fueron creadas como parte del experimento con el mismo sistema para generar distintos tipos celulares del cerebro a partir de células madre humanas embrionarias.

Generamos nuestro modelo con model.matrix

Nuestras variables a tomar en cuenta de nuestro modelo son el grupo (caso-control), Cre cell_line, la fuente, los días de cultivo y viral_barcoded.

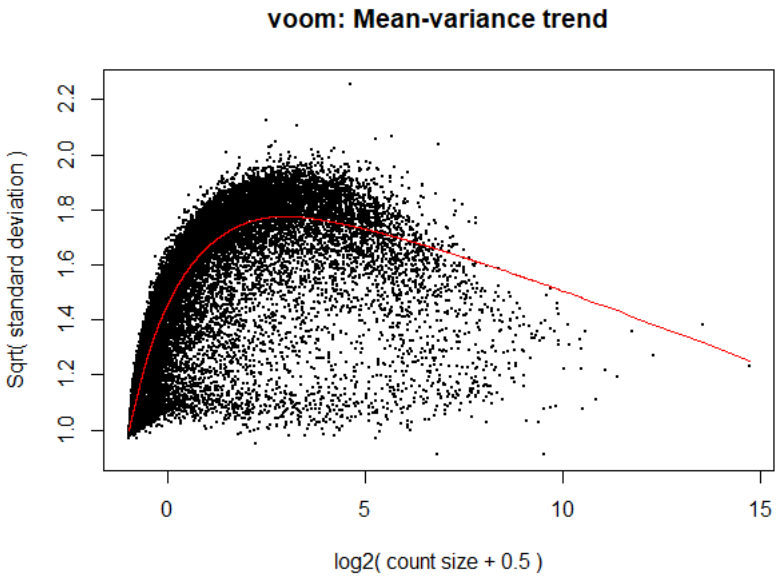
```
## Modelo estadistico con model.matrix
mod <- model.matrix(~ case + sra_attribute.cre_line + sra_attribute.days_in_culture + sra_attribute.source_name + sra_attribute.viral_barco
```

Differential Expression analisis

Realizamos nuestro análisis de expresión diferencial con la librería limma

Usando voom() observamos la tendencia de la varianza:

```
library("limma")
## voom
vGene <- voom(dge, mod, plot = TRUE)
```

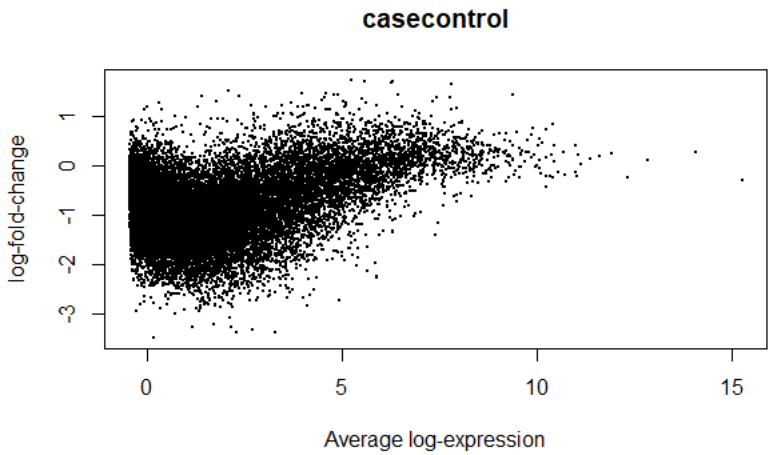


Realizamos el ajuste de datos con eBayes() usando un coeficiente de 2 (case-control):

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

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

Con un plotMA Visualizamos los resultados estadísticos:

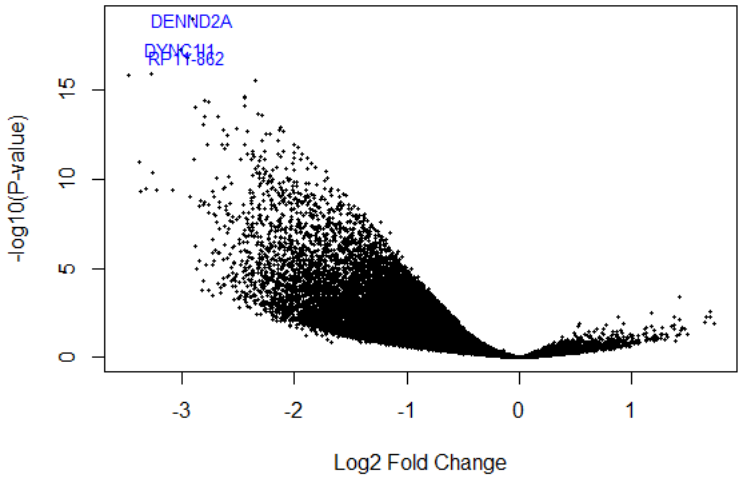
```
## Visualizamos los resultados estadísticos
plotMA(eb_results, coef = 2)
```



Para la mayoría de los genes podemos notar un cambio en el nivel de expresión (reducción).

Generamos un volcano plot con los resultados:

```
volcanoplot(eb_results, coef = 2, highlight = 3, names = de_results$gene_name)
```



Nuevamente podemos observar una reducción en la expresión en la mayoría de los genes diferencialmente expresados.

Obtenemos los 3 genes con mayor DE:

- DENND2A

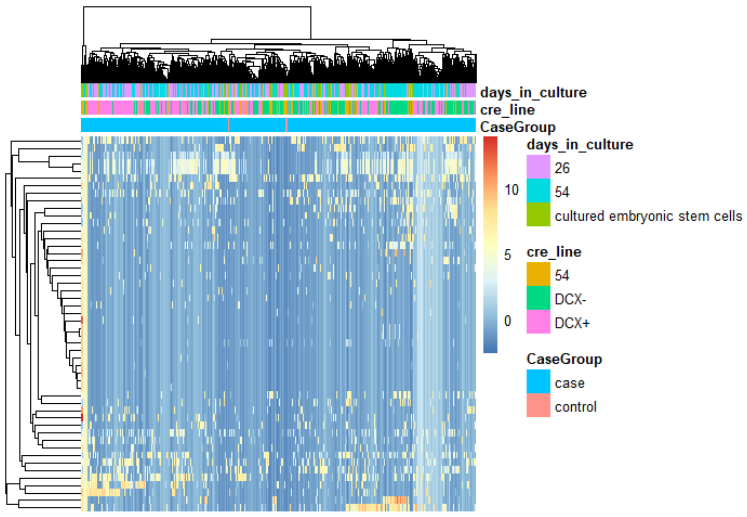
- DYNCH1
- RP11-862

```
## Obtenemos más informacion de los 3 genes con mas DE
de_results[de_results$gene_name %in% c("DENND2A", "DYNCH1", "RP11-862"), ]
```

Visualizar genes con DE

Para visualizar los 50 genes con mayor DE generamos un heatmap combinando las categorías CaseGroup, cre_line y days_in_culture:

```
########################## Visualizando genes DE ############################

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

## Creemos una tabla con información de las muestras
df <- as.data.frame(colData(rse_gene)[, c("case", "sra_attribute.cre_line", "sra_attribute.days_in_culture")])
## con nombres de columnas mas amigables
colnames(df) <- c("CaseGroup", "cre_line", "days_in_culture")

## Creamos un heatmap con los 50 genes con mayor DE
library("pheatmap")

pheatmap(
  exprs_heatmap,
  cluster_rows = TRUE,
  cluster_cols = TRUE,
  show_rownames = FALSE,
  show_colnames = FALSE,
  annotation_col = df
)
```



No se logra observar un cluster especifico para days_in_culture o cre_line, y tampoco notoriamente para CaseGroup, aunque el último puede deberse en parte a que la gran mayoría de muestras corresponden a Casos y muy pocas a controles.

Los genes agrupados mayormente sobreexpresados corresponden al grupo de Caso, DCX- cre cell_line y 54 days_in_culture

DE transcript interpretation

DENND2A - DENN domain containing 2A *(gene)*

Summary: Enables guanyl-nucleotide exchange factor activity. Involved in retrograde transport, endosome to Golgi. Located in actin cytoskeleton.

DYNCH1 - dynein cytoplasmic 1 heavy chain 1 *(gene)*

Summary: Cytoplasmic dynein 1 acts as a motor for the intracellular retrograde motility of vesicles and organelles along microtubules. Dynein has ATPase activity; the force-producing power stroke is thought to occur on release of ADP. Plays a role in mitotic spindle assembly and metaphase plate congression

RP11-862

C*orresponde a un lncRNA*

**m6A-induced lncRNA RP11 triggers the dissemination of colorectal cancer cells via upregulation of Zeb1

Los 2 primeros genes encontrados corresponden al transporte celular y se encuentran en el citoesqueleto, por lo que posiblemente sean fundamentales para que ciertos organelos se vean más favorecidos que otros así como la remodelación espacial de los mismos.

RP11-862 corresponde a un long non-coding RNA, del cual no pude encontrar mucha información. Uno de los artículos que menciona al grupo de los lncRNAs RP11 lo asocia con la diseminación de celulas cancerosas (colonrectales), por lo que su función posiblemente se asocie con la proliferación y diseminación celular, la cual en el neurodesarrollo resulta fundamental.