Real_dataset_tutorial

1.1 Data and Goals

Human embryonic stem cells (hESCs) typically exhibit “primed” pluripotency, analogous to stem cells derived from the mouse post-implantation epiblast. Since primed hESC have limited differentiation capacity, scientists have tried various method trying to revert “primed” hESCs to a more “naive” state which have higher pluripotency capacity (could have more clinical application potentials, and can also help us understand the early embryo development). By investigating the differentially expressed genes and their involved pathways, we can understand the mechanism undelying the differentiation capacity difference between naive hESCs and primed hESCs.

In this tutorial, we will use the 8 RNAseq dataset from *William Pastor et al., 2016, Cell Stem Cell*, with 4 replicates of naive hESCs and 4 replicates of primed hESCs.

Accession	ID	Replicate	CellType
GSM2041708	1	rep1	Primed_hESC
GSM2041709	2	rep2	Primed_hESC
GSM2041710	3	rep3	Primed_hESC
GSM2041711	4	rep4	Primed_hESC
GSM2041712	5	rep1	Naive_hESC
GSM2041713	6	rep2	Naive_hESC
GSM2041714	7	rep3	Naive_hESC
GSM2041715	8	rep4	Naive_hESC

```
#read in the primed hESC RNAseq raw count
primed_hESC_rep1 <- read.delim('/Users/jesi/Documents/real_Data/GSM2041708_RNAseq_UCLA1_
Primed_rep1_readsCount.txt',row.names = 1)
primed_hESC_rep2 <- read.delim('/Users/jesi/Documents/real_Data/GSM2041709_RNAseq_UCLA1_
Primed_rep2_readsCount.txt',,row.names = 1)
primed_hESC_rep3  <-read.delim('/Users/jesi/Documents/real_Data/GSM2041710_RNAseq_UCLA1_
Primed_rep3_readsCount.txt',row.names = 1)
primed_hESC_rep4  <-read.delim('/Users/jesi/Documents/real_Data/GSM2041711_RNAseq_UCLA1_
Primed_rep4_readsCount.txt',row.names = 1)

#read in the naive hESC RNAseq raw count
naive_hESC_rep1 <- read.delim('/Users/jesi/Documents/real_Data/GSM2041712_RNAseq_SSEA4_n
eg_rep1_readCounts.txt',row.names = 1)
naive_hESC_rep2 <- read.delim('/Users/jesi/Documents/real_Data/GSM2041713_RNAseq_SSEA4_n
eg_rep2_readCounts.txt',row.names = 1)
naive_hESC_rep3  <-read.delim('/Users/jesi/Documents/real_Data/GSM2041714_RNAseq_SSEA4_n
eg_rep3_readCounts.txt',row.names = 1)
naive_hESC_rep4  <-read.delim('/Users/jesi/Documents/real_Data/GSM2041715_RNAseq_SSEA4_n
eg_rep4_readsCount.txt',row.names = 1)

head(primed_hESC_rep4)
```

	X116 <int>
A1BG	18
A1BG-AS1	15
A1CF	13
A2LD1	30
A2M	36
A2ML1	393

6 rows

```
readcount = data.frame(naive_hESC_rep1,
                       naive_hESC_rep2,
                       naive_hESC_rep3,
                       naive_hESC_rep4,
                       primed_hESC_rep1,
                       primed_hESC_rep2,
                       primed_hESC_rep3,
                       primed_hESC_rep4)

colnames(readcount) = c(paste(rep('naive_hESC_rep',4),c(1:4),sep=''),paste(rep('primed_h
ESC_rep',4),c(1:4),sep=''))
# convert variable name as str: deparse(substitute(data))
```

```
library(grnaeR)
```

```
## Loading required package: tidyverse
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.0     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.1     ✔ tibble    3.1.8
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ✔ purrr     1.0.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the ]8;;http://conflicted.r-lib.org/ conflicted package ]8;; to force all confl
icts to become errors
```

```
library(DESeq2)
```

```
## Loading required package: S4Vectors
## Loading required package: stats4
## Loading required package: BiocGenerics
##
## Attaching package: 'BiocGenerics'
##
## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union
##
## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union
##
## 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
##
##
## Attaching package: 'S4Vectors'
##
## The following objects are masked from 'package:lubridate':
##
##     second, second<-
##
## The following objects are masked from 'package:dplyr':
##
##     first, rename
##
## The following object is masked from 'package:tidyr':
##
##     expand
##
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname
##
## Loading required package: IRanges
##
## Attaching package: 'IRanges'
##
## The following object is masked from 'package:lubridate':
##
```

```
##     %within%
##
## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice
##
## The following object is masked from 'package:purrr':
##
##     reduce
##
## Loading required package: GenomicRanges
## Loading required package: GenomeInfoDb
## Loading required package: SummarizedExperiment
## Loading required package: MatrixGenerics
## Loading required package: matrixStats
##
## Attaching package: 'matrixStats'
##
## The following object is masked from 'package:dplyr':
##
##     count
##
##
## Attaching package: '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
##
## Loading required package: Biobase
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
##
##
## Attaching package: 'Biobase'
```

```
##
## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians
##
## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

```
library('ggplot2')
library("pheatmap")
library("RColorBrewer")
library('AnnotationDbi')
```

```
##
## Attaching package: 'AnnotationDbi'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library('org.Hs.eg.db')
```

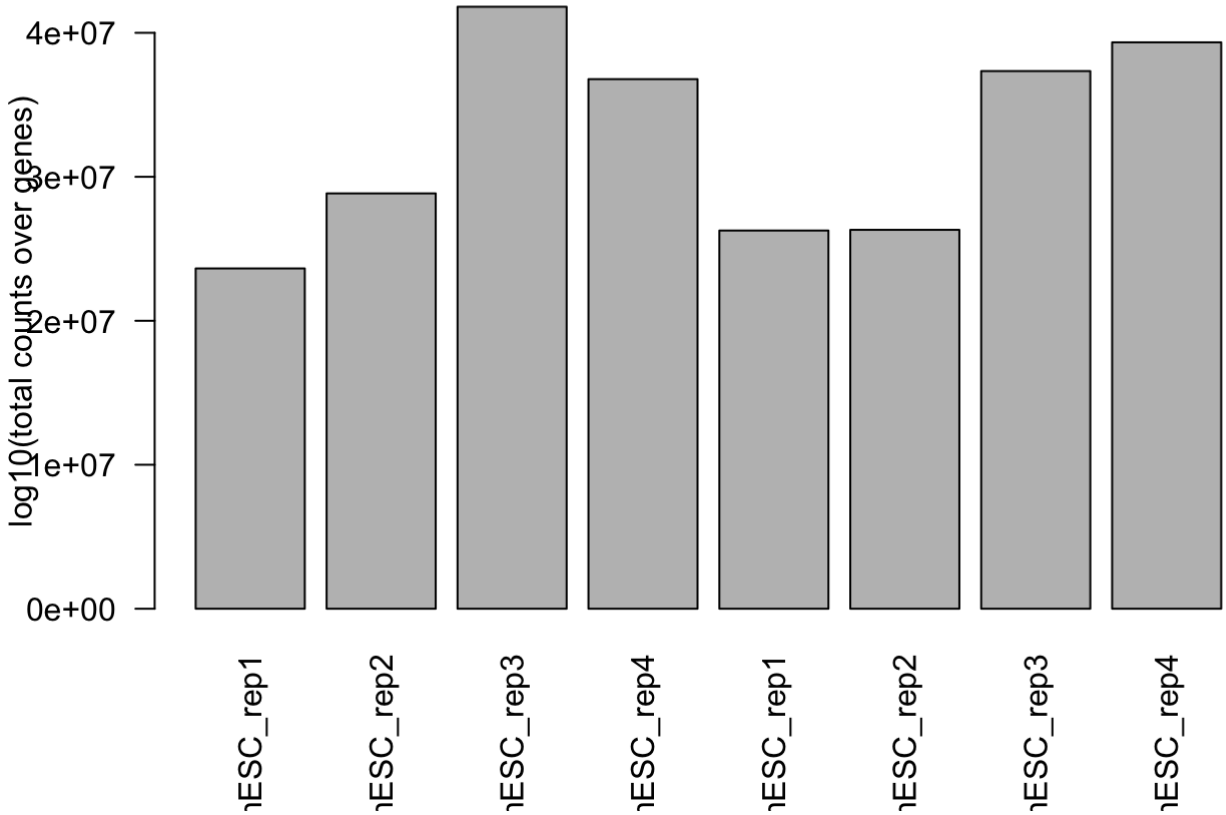
```
##
```

```
library(DOSE)
```

```
## DOSE v3.24.2  For help: https://yulab-smu.top/biomedical-knowledge-mining-book/
##
## If you use DOSE in published research, please cite:
## Guangchuang Yu, Li-Gen Wang, Guang-Rong Yan, Qing-Yu He. DOSE: an R/Bioconductor pack
age for Disease Ontology Semantic and Enrichment analysis. Bioinformatics 2015, 31(4):60
8-609
```

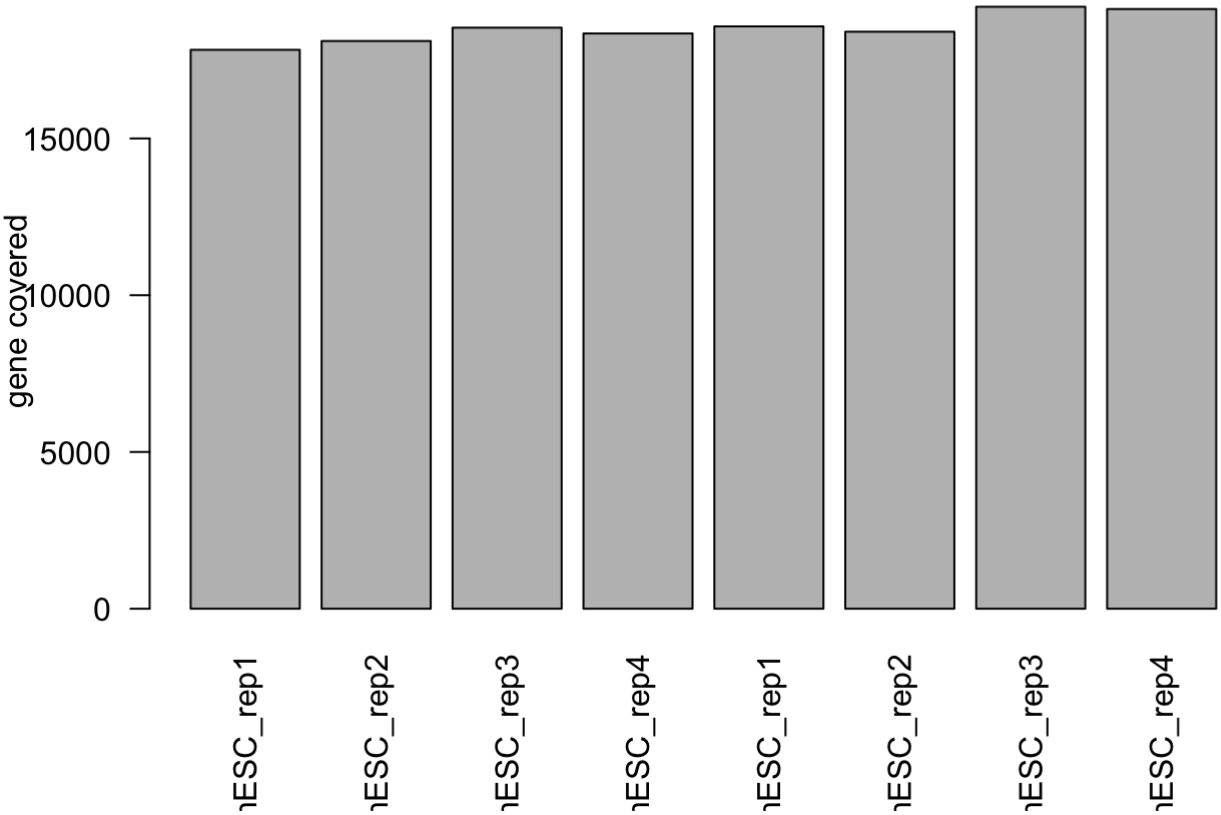
1.2 Check the data quality

```
total.cov = check_totalcov_quality(readcount)
```



From this plot, we can see that the coverage of all those 8 samples are in the same magnitude indicating they got sequenced evenly.

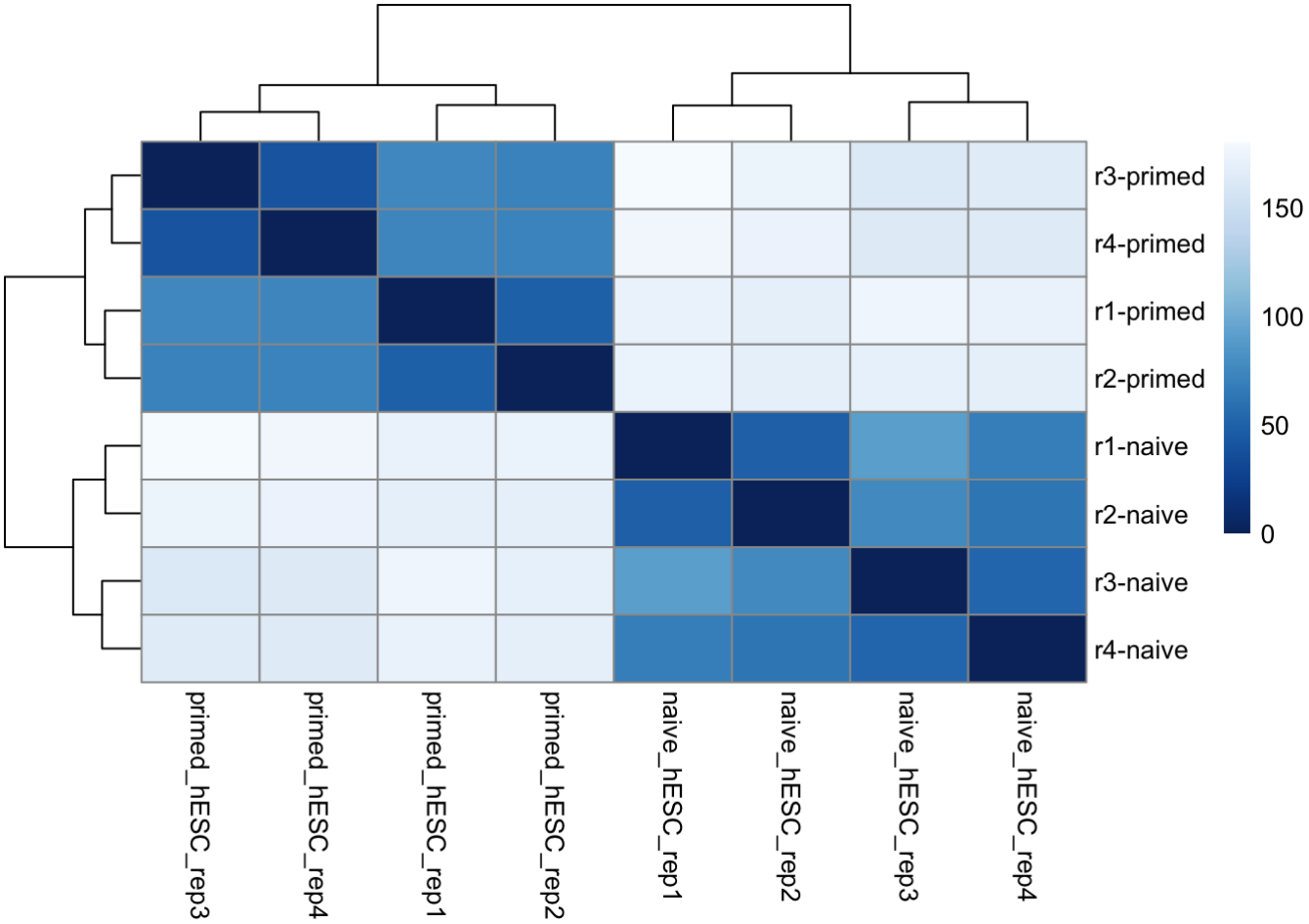
```
gene.cov = check_genecovered_quality(readcount)
```

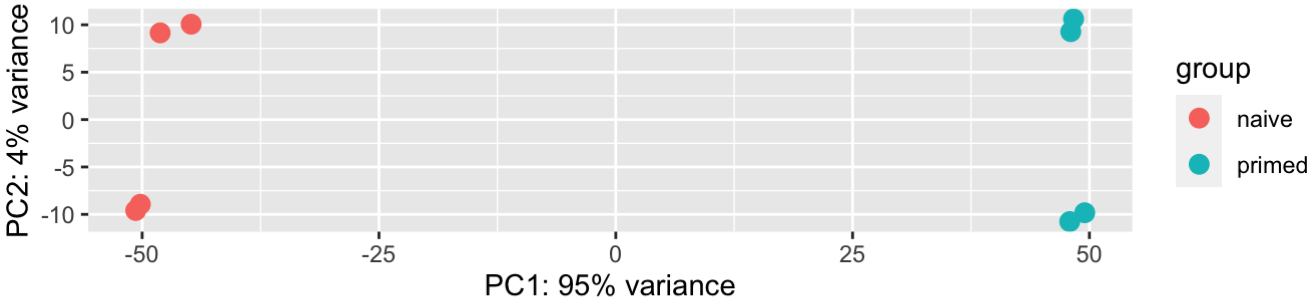


From this plot, we can see that most genes (18k+) are covered in all 8 samples, and the number of genes detected in each library is similar.

1.2 Load data into Deseq2

```
condition_vector = c(c(rep('naive',4)),c(rep('primed',4)))
type_vector = c(rep(paste(rep('r',4),c(1:4),sep=''),2))

dds = load_data_for_DESeq2(readcount,condition_vector,type_vector)
normalized_dds = normalize_dataset(dds)
check_sample_distance(normalized_dds)
```


r3-primed
r4-primed
r1-primed
r2-primed
r1-naive
r2-naive
r3-naive
r4-naive
150
100
50
0
primed_hESC_rep3
primed_hESC_rep4
primed_hESC_rep1
primed_hESC_rep2
naive_hESC_rep1
naive_hESC_rep2
naive_hESC_rep3
naive_hESC_rep4


PC2: 4% variance
10
5
0
-5
-10
-50
-25
0
25
50
PC1: 95% variance
group
naive
primed

```
select_DEGs = select_DEG(dds = dds,filter_thresh = 0,log2_fc = 1, padjust = 0.05)
```

```
## [1] "filtering 2489 genes with low counts"
```

```
## using pre-existing size factors
```

```
## estimating dispersions
```

```
## gene-wise dispersion estimates
```

```
## mean-dispersion relationship
```

```
## final dispersion estimates
```

```
## fitting model and testing
```

Here, we displayed part of the differentially expressed genes in the primed and naive human embryonic stem cells

```
head(select_DEGs)
```

	baseMean <dbl>	log2FoldChange <dbl>	lfcSE <dbl>	stat <dbl>	pvalue <dbl>	padj <dbl>
A4GALT	398.511748	-4.370871	0.2506463	-13.448718	3.132572e-41	3.211801e-39
AADAC	3.615533	-5.495015	1.3855628	-3.244180	1.177893e-03	8.406033e-03
AADACL3	16.371728	3.365400	0.7923113	2.985443	2.831683e-03	1.884823e-02
AARS2	2285.624212	-2.587097	0.2117793	-7.494107	6.675119e-14	1.636836e-12
ABCA1	7578.004807	-3.240110	0.3196038	-7.009022	2.399900e-12	5.166086e-11
ABCA13	187.323098	-2.190139	0.2360508	-5.041876	4.609896e-07	5.622820e-06

6 rows

Later, we try to convert the DEGS from SYMBOL IS to ENSEMBLE ID to enable the further visualization

```
gene_name = mapIds(org.Hs.eg.db,
                   keys = row.names(select_DEGs),
                   column = "ENTREZID",
                   keytype = "SYMBOL",
                   multiVals = "first")
```

```
## 'select()' returned 1:many mapping between keys and columns
```

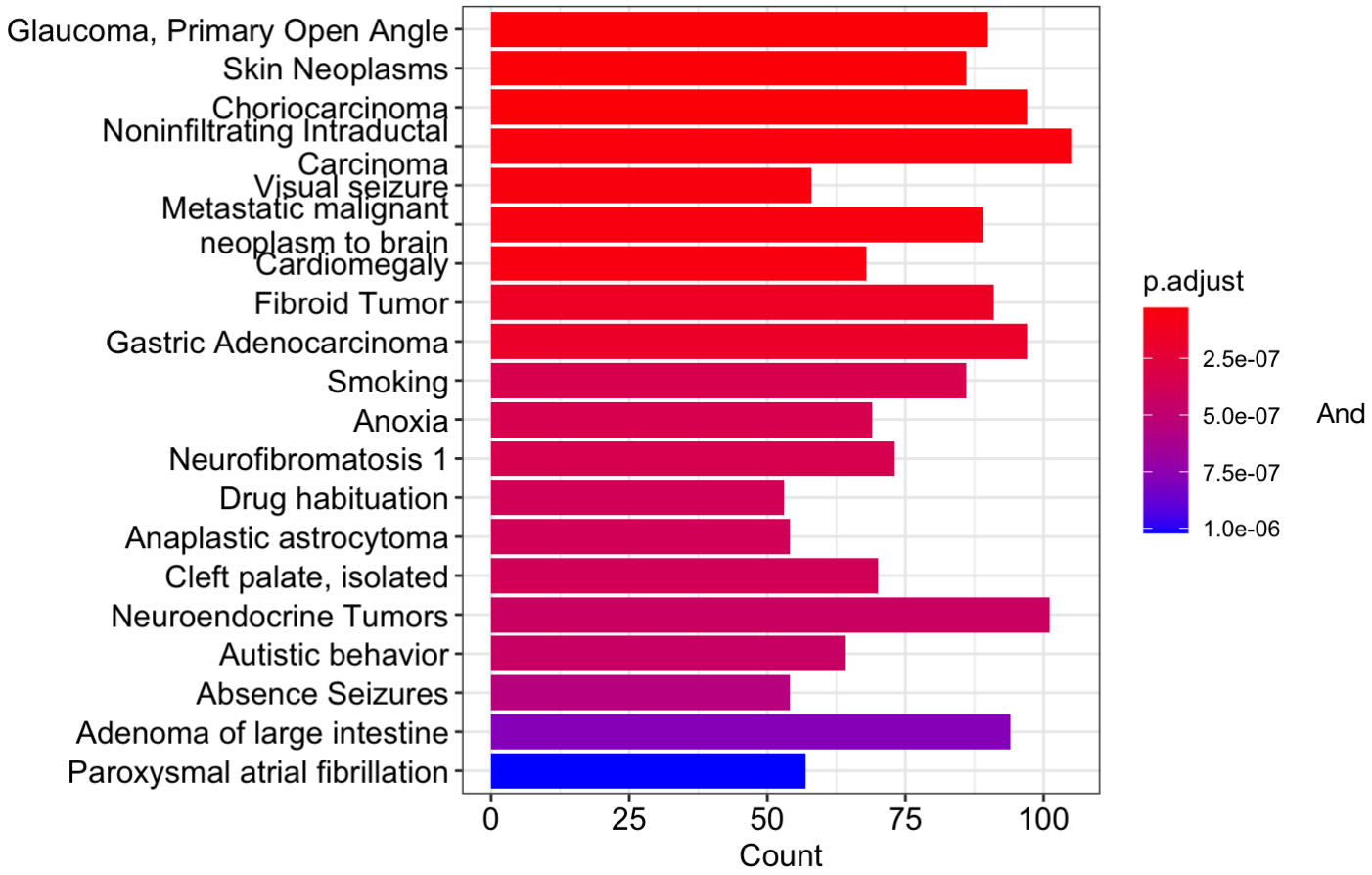
```
target_genes = as.data.frame(cbind('symbol_name' = row.names(select_DEGs),gene_name))
target_genes <- target_genes[complete.cases(target_genes), ]

DEGS <- subset(normalized_dds@assays@data@listData[[1]],rownames(normalized_dds@assays@d
ata@listData[[1]]) %in% rownames(target_genes)==TRUE)
```

```
DEG_genename <- c()
for(i in rownames(DEGS)){
  genename = subset(target_genes,symbol_name == i)$gene_name
  DEG_genename <- c(DEG_genename,genename)
}
rownames(DEGS) <- DEG_genename
naive_mean <- rowMeans(DEGS[,1:4])
edo <- filter_genelist(naive_mean,standard_fc = 2)
```

```
## [1] "enrichResult object generated"
```

We can see that the differentially expressed genes in naive cells are involved in following enriched terms. It depicts the enrichment scores (e.g. p values) and gene count or ratio as bar height and color.

```
barplot <- show_barplot(edo,showCategory_num = 20)
barplot
```



we can view the over representation analysis and gene set enrichment analysis in the naive human embryonic stem cells, to see the enriched pathways.

```
dotplot <- show_dotplot(edo,sort(naive_mean,decreasing = T),showCategory_num=30)
```

```
## preparing geneSet collections...
```

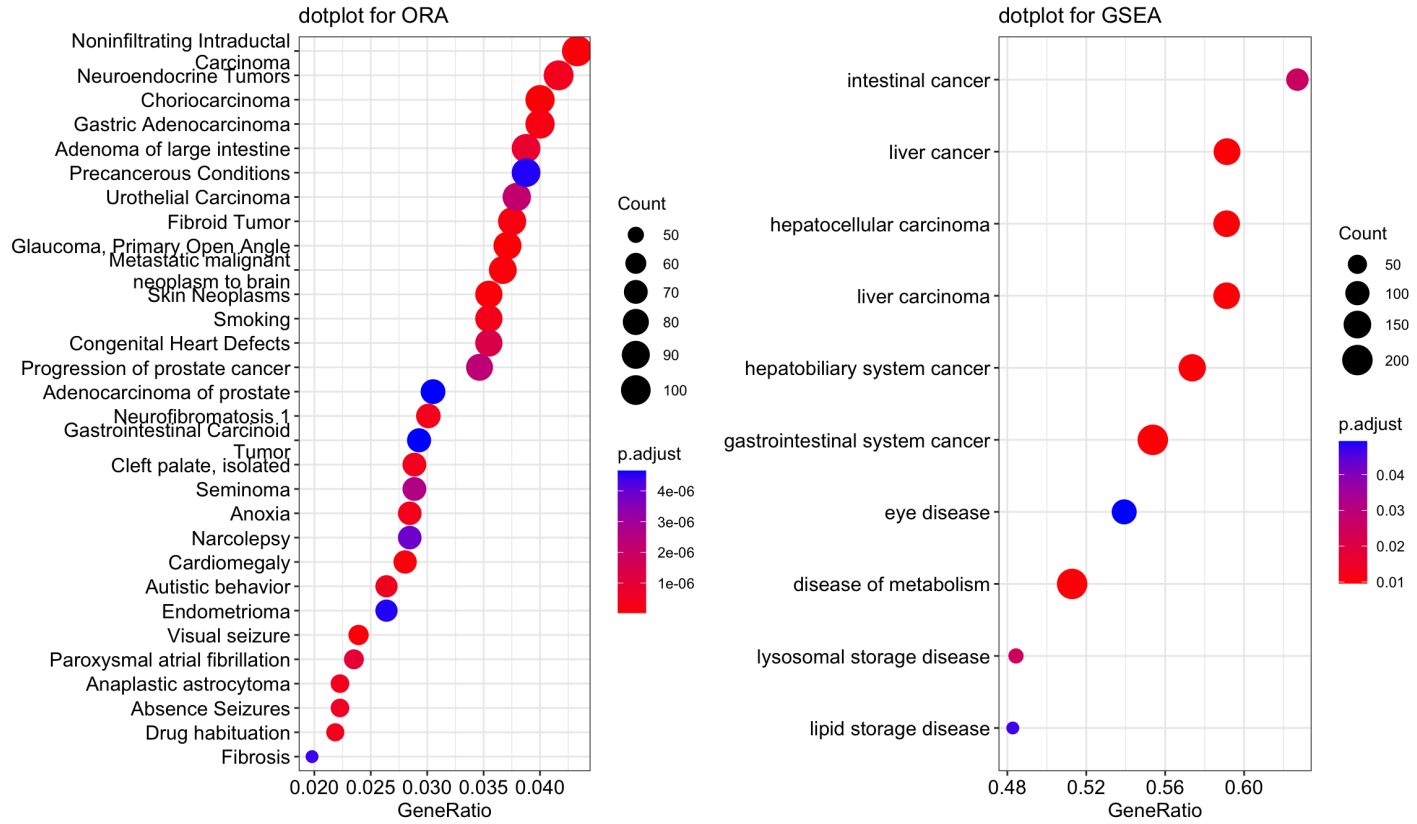
```
## GSEA analysis...
```

```
## Warning in preparePathwaysAndStats(pathways, stats, minSize, maxSize,
## gseaParam, : All values in the stats vector are greater than zero and scoreType
## is "std", maybe you should switch to scoreType = "pos".
```

```
## leading edge analysis...
```

```
## done...
```

```
dotplot
```



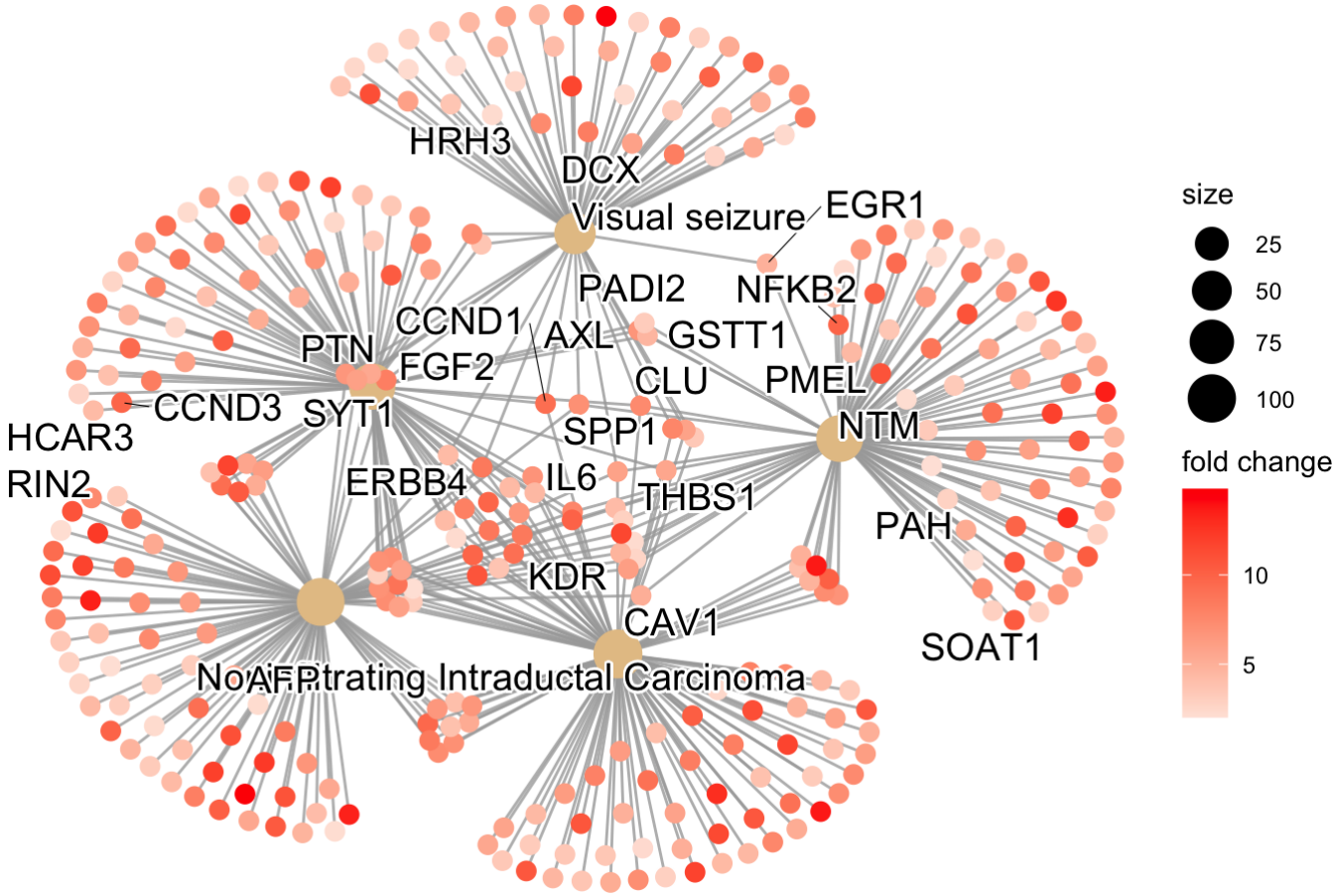
Through visualization of gene network, we can more directly understand the interaction between genes and their role in involved pathways.

```
gene_network <-develop_Gene_Network(edo,naive_mean)
```

```
## Scale for size is already present.
## Adding another scale for size, which will replace the existing scale.
```

```
gene_network
```

```
## Warning: ggrepel: 311 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```


HRH3
DCX
Visual seizure
EGR1
PADI2
NFKB2
CCND1
AXL
GSTT1
PTN
FGF2
CLU
PMEL
CCND3
SYT1
SPP1
NTM
HCAR3
RIN2
ERBB4
IL6
THBS1
PAH
KDR
CAV1
SOAT1
No AFP rating Intraductal Carcinoma
size
25
50
75
100
fold change
10
5