deg_assignment_2

chandrima

2024-11-13

```
.libPaths("C:/MS_Bioinformatics_Fall_2024/deseq_analysis/renv/library/R-4.3/x86_64-w64-mingw32")
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(BiocManager)
```

```
## Bioconductor version '3.18' is out-of-date; the current release version '3.20'
##   is available with R version '4.4'; see https://bioconductor.org/install
```

```
library(DESeq2)
```

```
## Warning: package 'DESeq2' was built under R version 4.3.3
```

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

```
##
## Loading required package: GenomicRanges
## Loading required package: GenomeInfoDb

## Warning: package 'GenomeInfoDb' was built under R version 4.3.3

## 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
```

```r
library(clusterProfiler)
```

```
## Warning: package 'clusterProfiler' was built under R version 4.3.3

##
## clusterProfiler v4.10.1  For help: https://yulab-smu.top/biomedical-knowledge-mining-book/
##
## If you use clusterProfiler in published research, please cite:
## T Wu, E Hu, S Xu, M Chen, P Guo, Z Dai, T Feng, L Zhou, W Tang, L Zhan, X Fu, S Liu, X Bo, and G Yu.
##
## Attaching package: 'clusterProfiler'
##
## The following object is masked from 'package:IRanges':
##
##     slice
##
## The following object is masked from 'package:S4Vectors':
##
##     rename
##
## The following object is masked from 'package:purrr':
##
##     simplify
##
## The following object is masked from 'package:stats':
##
##     filter
```

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

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

```r
# Create an empty data frame to hold the combined data
data_df <- data.frame()

# Get the list of file names in the directory
file_names <- list.files("C:\\MS_Bioinformatics_Fall_2024\\deseq_analysis\\counts_csvs")

# Initialize a flag to include gene identifiers only once
include_gene_ids <- TRUE

for (files in file_names) {
  # Construct the full file path
  file_path <- paste0("C:\\MS_Bioinformatics_Fall_2024\\deseq_analysis\\counts_csvs\\", files)

  # Read the CSV file
  data_clean <- read.csv(file_path)

  # Extract the sample name (e.g., SRR22269872) using basename
  sample_name <- sub("_.*", "", basename(file_path))

  # Rename the second column to the sample name
  colnames(data_clean)[2] <- sample_name

  # For the first file, include gene identifiers (assuming they are in the first column)
  if (include_gene_ids) {
    data_df <- data_clean[, c(1, 2)] # Include GeneID and the sample column
    include_gene_ids <- FALSE        # Disable including GeneID in subsequent iterations
  } else {
    # Only bind the new sample column
    data_df <- cbind(data_df, data_clean[, 2])
  }
}

# Update column names of the final data_df to include sample names
sample_names <- c("GeneID", sapply(file_names, function(f) sub("_.*", "", basename(f))))
colnames(data_df) <- sample_names

# Display the first few rows of the combined data
head(data_df)
```

```
##            GeneID SRR22269872 SRR22269873 SRR22269874 SRR22269875 SRR22269876
## 1 ENSG00000142611           0           0           0           0           0
## 2 ENSG00000284616           0           0           0           0           0
## 3 ENSG00000157911           0           2           0           0           0
## 4 ENSG00000260972           0           0           0           0           0
## 5 ENSG00000224340           0           0           0           0           0
## 6 ENSG00000229280           0           0           0           0           0
##   SRR22269877 SRR22269878 SRR22269879 SRR22269880 SRR22269881 SRR22269882
## 1           0           0           0           0           0           0
## 2           0           0           0           0           0           0
## 3           0           0           0           0           0           0
## 4           0           0           0           0           0           0
## 5           0           0           0           0           0           0
## 6           0           0           0           0           0           0
##   SRR22269883
## 1           0
## 2           0
## 3           0
## 4           0
## 5           0
## 6           0
```

```
# Read the sample information CSV
sample_info <- read.csv("assignment_2_info.csv") %>%
```

```r
  # Clean up column names and any whitespace in values
  mutate(
    Condition = trimws(Condition),
    SRA_Accession = trimws(SRA.Accession)
  ) %>%
  # Convert to factors
  mutate(
    Condition = factor(Condition),
    Time_Point = factor(Time.Point),
    Sample = factor(Sample))
sample_info <- sample_info[, -c(3:10)]
```

```r
#
data_deseq <- as.matrix(data_df[-1])
mode(data_deseq) <- "numeric"
rownames(data_deseq) <- data_df$GeneID
data_deseq <- data_deseq[, sample_info$SRA_Accession]
# Keep only genes that have at least 10 counts in at least 3 samples
keep <- rowSums(data_deseq >= 10) >= 3
counts_filtered <- data_deseq[keep,]
```

```r
# Create DESeq2 object
dds <- DESeqDataSetFromMatrix(
  countData = counts_filtered,
  colData = sample_info,
  design = ~ Condition + Time_Point
)
```

```
## converting counts to integer mode
```

```
##   Note: levels of factors in the design contain characters other than
##   letters, numbers, ’_’ and ’.’. It is recommended (but not required) to use
##   only letters, numbers, and delimiters ’_’ or ’.’, as these are safe characters
##   for column names in R. [This is a message, not a warning or an error]
```

```r
# Run DESeq2
dds <- DESeq(dds)
```

```
## estimating size factors
```

```
##   Note: levels of factors in the design contain characters other than
##   letters, numbers, ’_’ and ’.’. It is recommended (but not required) to use
##   only letters, numbers, and delimiters ’_’ or ’.’, as these are safe characters
##   for column names in R. [This is a message, not a warning or an error]
```

```
## estimating dispersions
```

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

```
## mean-dispersion relationship
##   Note: levels of factors in the design contain characters other than
##   letters, numbers, '_' and '.'. It is recommended (but not required) to use
##   only letters, numbers, and delimiters '_' or '.', as these are safe characters
##   for column names in R. [This is a message, not a warning or an error]
```

```
## final dispersion estimates
```

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

```r
resultsNames(dds)
```

```
## [1] "Intercept"                    "Condition_SARS.CoV.2_vs_Mock"
## [3] "Time_Point_72_vs_24"
```

```r
get_results <- function(dds, contrast, name) {
  res <- results(dds, contrast=contrast, alpha=0.05)

  # Convert to data frame and add gene names
  res_df <- as.data.frame(res) %>%
    rownames_to_column("gene_id") %>%
    mutate(gene_name = mapIds(org.Hs.eg.db,
                              keys=gene_id,
                              column="SYMBOL",
                              keytype="ENSEMBL",
                              multiVals="first")) # Map Ensembl to gene names

  # Add significance columns
  res_df <- res_df %>%
    mutate(
      significant = padj < 0.05,
      regulation = case_when(
        log2FoldChange >= 1 & padj < 0.05 ~ "Up",
        log2FoldChange <= -1 & padj < 0.05 ~ "Down",
        TRUE ~ "Not Significant"
      )
    )

  # Sort by adjusted p-value
  res_df <- res_df %>%
    arrange(padj)

  # Write results to CSV
  # write.csv(res_df, paste0("DEG_analysis\\", name, "_DEGs.csv"))

  return(list(res=res, res_df=res_df))
}
```

```r
# Get results for different comparisons
res_24h <- get_results(
  dds,
  contrast=c("Condition", "SARS-CoV-2", "Mock"),
  name="mock_vs_SARS_24h"
)
```

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

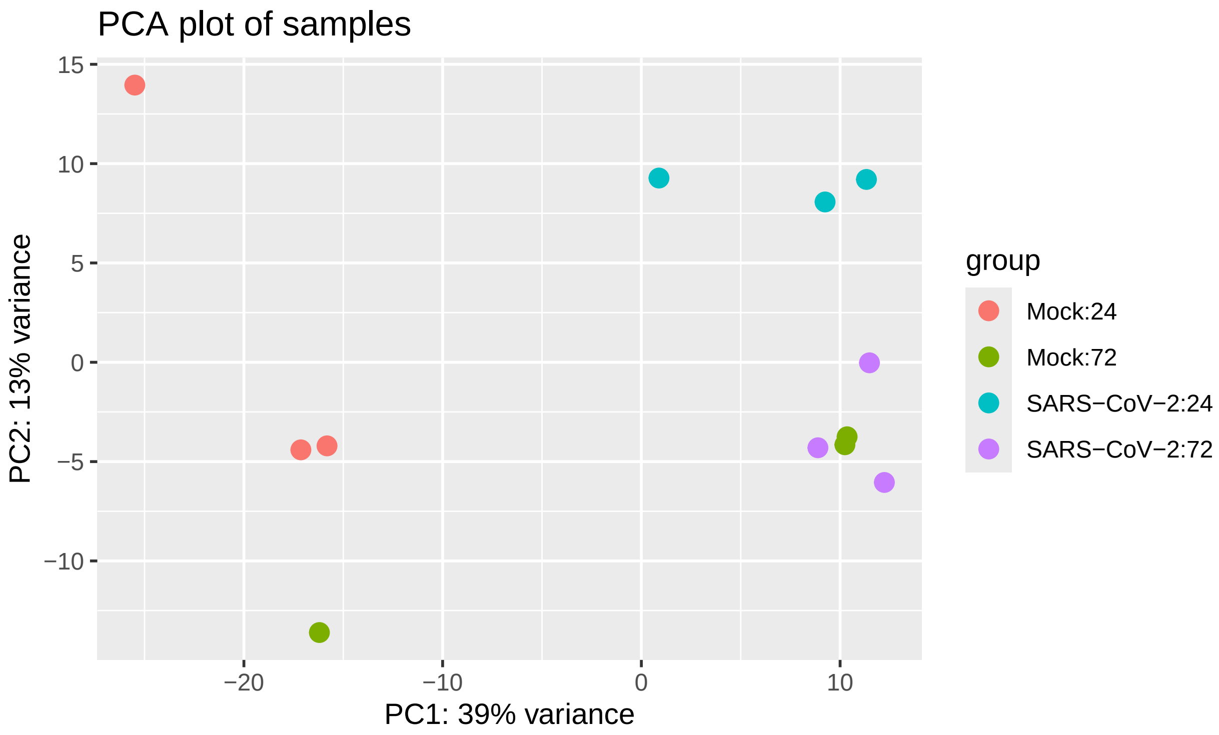
```r
res_time <- get_results(
  dds,
  contrast=c("Time_Point", "72", "24"),
  name="time_24h_vs_72h"
)
```

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

```r
# Generate summary statistics
summary_stats <- data.frame(
  Comparison = c("SARS vs Mock (Main effect)",
                 "Time 72h vs 24h"),

  Total_Genes = c(nrow(res_24h$res),
                  nrow(res_time$res)),

  Significant_DEGs = c(sum(res_24h$res_df$padj < 0.05, na.rm=TRUE),
                       sum(res_time$res_df$padj < 0.05, na.rm=TRUE)),

  Upregulated = c(sum(res_24h$res_df$regulation == "Up", na.rm=TRUE),
                  sum(res_time$res_df$regulation == "Up", na.rm=TRUE)),

  Downregulated = c(sum(res_24h$res_df$regulation == "Down", na.rm=TRUE),
                    sum(res_time$res_df$regulation == "Down", na.rm=TRUE))
)

# Save summary statistics
write.csv(summary_stats, "summary_statistics.csv", row.names=FALSE)

# Save normalized counts
normalized_counts <- counts(dds, normalized=TRUE)
write.csv(normalized_counts, "normalized_counts.csv")
```

```r
# # Generate diagnostic plots

# 1. PCA plot
vsd <- varianceStabilizingTransformation(dds, blind=FALSE)
plotPCA(vsd, intgroup=c("Condition", "Time_Point")) +
  ggtitle("PCA plot of samples")
```

```
## using ntop=500 top features by variance
```



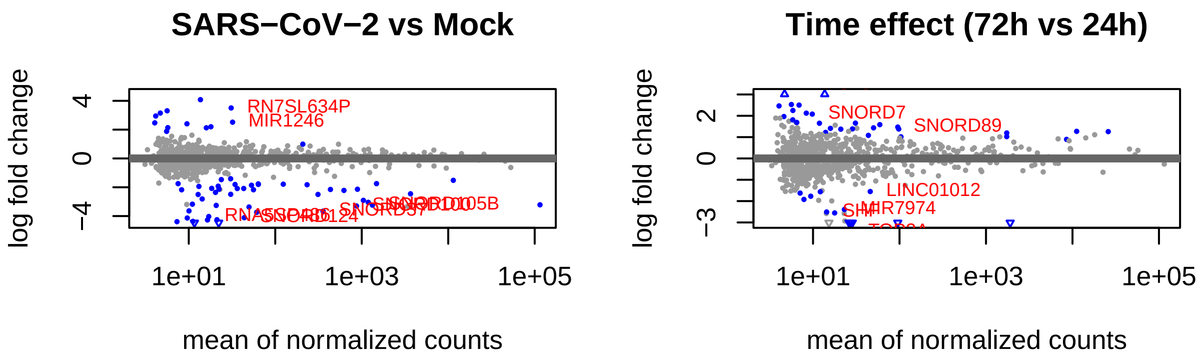
```
label_significant_genes <- function(res, top_n = 10) {
  # Filter significant genes
  sig_genes <- as.data.frame(res) %>%
    rownames_to_column("gene_id") %>%
    mutate(gene_name = mapIds(org.Hs.eg.db,
                              keys=gene_id,
                              column="SYMBOL",
                              keytype="ENSEMBL",
                              multiVals="first")) %>%
    filter(padj < 0.05) %>%
    arrange(padj) %>%
    head(top_n) # Select top N significant genes

  # Add text labels to the plot
  with(sig_genes, {
    text(baseMean, log2FoldChange, labels=gene_name, pos=4, cex=0.7, col="red")
  })
}
```

```
par(mfrow=c(2,2))

# SARS-CoV-2 vs Mock (Main effect)
plotMA(res_24h$res, main="SARS-CoV-2 vs Mock")
label_significant_genes(res_24h$res, top_n=10)
```

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

```r
# Time 72h vs 24h
plotMA(res_time$res, main="Time effect (72h vs 24h)")
label_significant_genes(res_time$res, top_n=10)
```

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



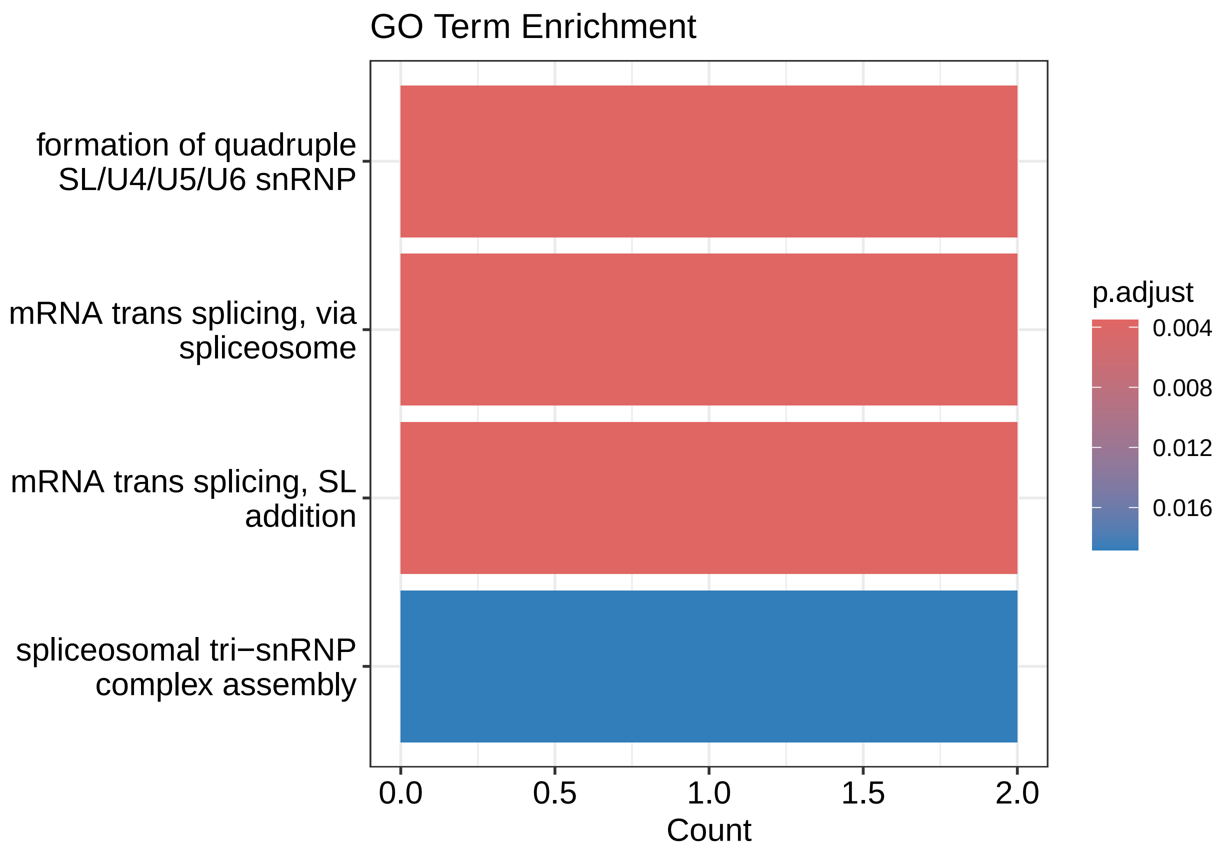
```r
degs <- res_24h$res_df
rownames(degs) <- degs$gene_id
# Map Ensembl IDs to Gene Symbols
gene_id <- rownames(degs)
gene_name <- mapIds(
  org.Hs.eg.db,
  keys=gene_id,
  column="SYMBOL",
  keytype="ENSEMBL",
  multiVals="first"
)
```

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

```r
# Add mapped gene symbols to the DEGs dataframe
degs$SYMBOL <- gene_name

# Select significant genes with p-adjusted value < 0.05
```

```r
genes <- degs$SYMBOL[degs$padj < 0.05]
genes <- na.omit(genes)  # Remove any NA values

# Perform GO Term Enrichment
ego <- enrichGO(
  gene=genes,
  OrgDb=org.Hs.eg.db,
  keyType="SYMBOL",
  ont="BP",  # Biological Process
  pAdjustMethod="BH",
  pvalueCutoff=0.05
)

# Check Results
if (!is.null(ego)) {
  print(head(as.data.frame(ego)))

  # Visualize results (optional)
  barplot(ego, showCategory=10, title="GO Term Enrichment")
} else {
  print("No enriched terms found.")
}
```

```
##                    ID                              Description GeneRatio
## GO:0000353 GO:0000353 formation of quadruple SL/U4/U5/U6 snRNP      2/30
## GO:0000365 GO:0000365     mRNA trans splicing, via spliceosome      2/30
## GO:0045291 GO:0045291         mRNA trans splicing, SL addition      2/30
## GO:0000244 GO:0000244  spliceosomal tri-snRNP complex assembly      2/30
##              BgRatio       pvalue    p.adjust      qvalue          geneID Count
## GO:0000353 10/18870 0.0001090870 0.003527146 0.002755882 RNU5A-1/RNU5B-1     2
## GO:0000365 10/18870 0.0001090870 0.003527146 0.002755882 RNU5A-1/RNU5B-1     2
## GO:0045291 10/18870 0.0001090870 0.003527146 0.002755882 RNU5A-1/RNU5B-1     2
## GO:0000244 26/18870 0.0007754877 0.018805576 0.014693451 RNU5A-1/RNU5B-1     2
```


GO Term Enrichment
formation of quadruple SL/U4/U5/U6 snRNP
mRNA trans splicing, via spliceosome
mRNA trans splicing, SL addition
spliceosomal tri−snRNP complex assembly
0.0
0.5
1.0
1.5
2.0
Count
p.adjust
0.004
0.008
0.012
0.016