MAR 14, 2024

OPEN ACCESS

Protocol Citation: Karina Jhingan 2024. DESEq2 for time series. **protocols.io** https://protocols.io/view/deseq2-for-time-series-dajh2cj6

Protocol status: Working
We use this protocol and it's working

Created: Mar 12, 2024

Last Modified: Mar 14, 2024

PROTOCOL integer ID: 96585

DESEq2 for time series

Karina Jhingan[1]

[1]Fred Hutch

Karina Jhingan
Fred Hutch

DISCLAIMER

This code in this protocol is an altered version of https://alexslemonade.github.io/refinebio-examples/03-rnaseq/differential-expression_rnaseq_01.html

ABSTRACT

This is a differential analysis for a time series experiment using DESeq2.

Introduction

1 This protocol is heavily based off of the protocol provided by https://alexslemonade.github.io/refinebio-examples/03-rnaseq/differential-expression_rnaseq_01.html.

Note that DESeq and any other differential analysis requires replicates.

Imports/Libraries

2 Citation: “ashr” for LFC shrinkage
Stephens, M. (2016) False discovery rates: a new
deal. Biostatistics, 18:2. https://doi.org/10.1093/biostatistics/kxw041

```
if (!("DESeq2" %in% installed.packages())) {
  # Install this package if it isn't installed yet
  BiocManager::install("DESeq2", update = FALSE)
}
if (!("apeglm" %in% installed.packages())) {
  # Install this package if it isn't installed yet
  BiocManager::install("apeglm", update = FALSE)
}
if (!("ashr" %in% installed.packages())) {
  # Install this package if it isn't installed yet
  BiocManager::install("ashr", update = FALSE)
}
```

```
# Attach the DESeq2 library
library(DESeq2)

# Attach the ggplot2 library for plotting
library(ggplot2)

# We will need this so we can use the pipe: %>%
library(magrittr)

library(tidyverse)

#set seed for DESeq2::plotCounts() for reproducibility
set.seed(12345)
```

Data Set up

3 **Set Project Path**

```
#replace with your directory
projPath="/fh/fast/greenberg_p/user/kjhingan/DESeq2"
```

4 **Gene Expression Data Matrix**

A csv file where rows are genes and columns are samples/time stamps

```
# Replace with the path to your dataset file
data_file <- file.path(projPath, "gsva_data.csv")
```

5 **Metadata File**

A csv file with nrows(metadata) == ncols(datafile)
three columns:
-id: sample name, i.e the column names from the
data file
-group (i.e time stamp or control group)
-replicate

```
# Replace with the path to your metadata file
metadata_file <- file.path(projPath, "gsva_metadata.csv")
```

6

```
# Check if the gene expression matrix file is at the path stored in
`data_file`
file.exists(data_file)

# Check if the metadata file is at the file path stored in `metadata_file`
file.exists(metadata_file)
```

You should see TRUE outputted twice

7 **Read Data**

```
# Read in metadata CSV file
metadata <- readr::read_csv(metadata_file)

# Read in data CSV file
expression_df <- readr::read_csv(data_file)
```

8 **Set up data matrix**

```
# ensure unique row names for the next function
expression_df <- expression_df[!(duplicated(expression_df[[1]]) |
                                   duplicated(expression_df[[1]],
fromLast=TRUE)),]


expression_df <-expression_df %>%
  # store the gene IDs as row names so we have a numeric matrix to perform
calculations on later
  tibble::column_to_rownames("Gene")
```

9 **Set up metadata**

We are using the naive samples ("0 hours") as a reference/control.

```
metadata <- metadata %>%
  dplyr::mutate(time_stamp = dplyr::case_when(
    stringr::str_detect(group, "naïve") ~ "reference",
    stringr::str_detect(group, "24") ~ "24_hours",
    stringr::str_detect(group, "48") ~ "48_hours",
    stringr::str_detect(group, "72") ~ "72_hours"
  ))
```

10

```
# Let's take a look at the original metadata column's info
# and our new `mutation_status` column
dplyr::select(metadata, group, time_stamp)
```

```
# A tibble: 12 × 2
   group time_stamp
   <chr> <chr>
 1 naïve reference
 2 naïve reference
 3 naïve reference
 4 24    24_hours
 5 24    24_hours
 6 24    24_hours
 7 48    48_hours
 8 48    48_hours
 9 48    48_hours
10 72    72_hours
11 72    72_hours
12 72    72_hours
```

```
# Print out a preview of `mutation_status`
str(metadata$time_stamp)
```

```
> str(metadata$time_stamp)
 chr [1:12] "reference" "reference" "reference" ...
```

11

```
# Make mutation_status a factor and set the levels appropriately
metadata <- metadata %>%
  dplyr::mutate(
    # Here we define the values our factor variable can have and their
order.
    time_stamp = factor(time_stamp, levels = c("reference", "24_hours",
"48_hours", "72_hours"))
  )

levels(metadata$time_stamp)
```

You should see the following printed out: [1] "reference" "24_hours" "48_hours" "72_hours"

12

```
# Define a minimum counts cutoff and filter the data to include
# only rows (genes) that have total counts above the cutoff
filtered_expression_df <- expression_df %>%
  dplyr::filter(rowSums(.) >= 10)

# round all expression counts
gene_matrix <- round(filtered_expression_df)
```

DESeq2

13 **Create DESeq2 Object**

```
ddset <- DESeqDataSetFromMatrix(
  # Here we supply non-normalized count data
  countData = gene_matrix,
  # Supply the `colData` with our metadata data frame
  colData = metadata,
  # Supply our experimental variable to `design`
  design = ~time_stamp
)
```

14 **Run DESeq2**

```
deseq_object <- DESeq(ddset)
```

15 **View Results**

```
deseq_results <- results(deseq_object)
resultsNames(deseq_object)
```

[1] "Intercept" "time_stamp_24_hours_vs_reference" "time_stamp_48_hours_vs_reference"
[4] "time_stamp_72_hours_vs_reference"

```
head(deseq_results)
```

```
log2 fold change (MLE): time stamp 72 hours vs reference
Wald test p-value: time stamp 72 hours vs reference
DataFrame with 6 rows and 6 columns
       baseMean log2FoldChange     lfcSE      stat       pvalue         padj
      <numeric>      <numeric> <numeric> <numeric>    <numeric>    <numeric>
IL2RA   177.902      10.406913  1.182509  8.800703  1.35962e-18  3.80387e-17
IL24     11.200       5.768774  1.615050  3.571885  3.54422e-04  2.20676e-03
GZMB   2148.532       6.432327  0.212220 30.309658 8.55203e-202 1.76257e-198
IFNG    518.404      -0.272068  0.453657 -0.599723  5.48691e-01  7.74733e-01
LIF      29.043       7.480184  1.248842  5.989698  2.10231e-09  2.92632e-08
NRN1     26.553       6.839140  1.264260  5.409600  6.31657e-08  7.28644e-07
```

16 Normalize results by log fold change

Here we are shrinking log fold change by ashr, for more information on ashr go to step 2 and view the citation.

```
deseq_results <- lfcShrink(
  deseq_object, # The original DESeq2 object after running DESeq()
  contrast=c("reference","24_hours","48_hours", "72_hours"),
  type = "ashr",
  res = deseq_results # The original DESeq2 results table
)
head(deseq_results)
```

```
log2 fold change (MMSE): time stamp 72 hours vs reference
Wald test p-value: time stamp 72 hours vs reference
DataFrame with 6 rows and 5 columns
       baseMean log2FoldChange     lfcSE       pvalue         padj
      <numeric>      <numeric> <numeric>    <numeric>    <numeric>
IL2RA   177.902       9.412737  1.129324  1.35962e-18  3.80387e-17
IL24     11.200       3.127357  1.967306  3.54422e-04  2.20676e-03
GZMB   2148.532       6.407684  0.212188 8.55203e-202 1.76257e-198
IFNG    518.404      -0.207718  0.396938  5.48691e-01  7.74733e-01
LIF      29.043       6.565746  1.252638  2.10231e-09  2.92632e-08
NRN1     26.553       5.868113  1.325249  6.31657e-08  7.28644e-07
```

17 Set up DESeq results for visualization

Convert the results into a dataframe

```
deseq_df <- deseq_results %>%
  # make into data.frame
  as.data.frame() %>%
  # the gene names are row names -- let's make them a column for easy
display
  tibble::rownames_to_column("Gene") %>%
  # add a column for significance threshold results
  dplyr::mutate(threshold = padj < 0.05) %>%
  # sort by statistic -- the highest values will be genes with
  # higher expression in RPL10 mutated samples
  dplyr::arrange(dplyr::desc(log2FoldChange))

head(deseq_df)
```

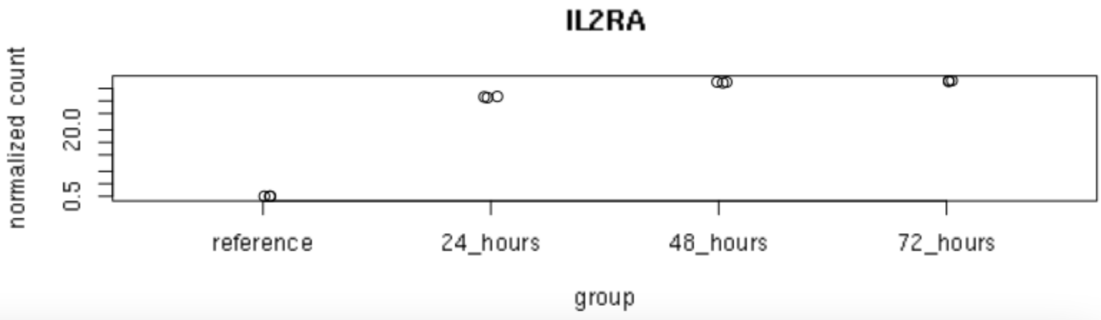
	Gene	baseMean	log2FoldChange	lfcSE	pvalue	padj	threshold
1	IL2RA	177.90182	9.412737	1.1293238	1.359622e-18	3.803865e-17	TRUE
2	TNFRSF8	27.25316	7.492022	1.2237200	1.885657e-11	3.220723e-10	TRUE
3	ZBTB32	47.95367	6.942746	1.1913659	1.018535e-10	1.618921e-09	TRUE
4	ASNS	35.56209	6.859988	1.2149153	2.733904e-10	4.132941e-09	TRUE
5	LIF	29.04295	6.565746	1.2526382	2.102312e-09	2.926319e-08	TRUE
6	GZMB	2148.53206	6.407684	0.2121883	8.552031e-202	1.762574e-198	TRUE

Visualizations

18 Plot counts

IL2RA was only chosen as an example, change to your gene of interest into the gene parameter.

```
plotCounts(ddset, gene = "IL2RA", intgroup = "time_stamp")
```



Save Results

19 **Save DESeq2 results**

```
readr::write_tsv(
  deseq_df,
  file.path(
    projPath,
    "SRP123625_diff_expr_results.tsv" # Replace with a relevant output file
name
  )
)
```