DE_analysis_time

Yu (Sylvia) Zhang

2022-11-13

```
library(GEOquery)
library(SummarizedExperiment)
library(DESeq2)
library(here)
library(dplyr)
library(limma)
library(nlme)
library(MASS)
library(multcomp)
```

Load data

I saved the data into a local directory to save time from downloading each time.

```
gset <- getGEO("GSE48024", GSEMatrix =TRUE, AnnotGPL=TRUE)
save(gset, file = "/Users/yu.zhang/Desktop/BIOS 784/Data/GSE48024/gset.RData")
```

```
load("/Users/yu.zhang/Desktop/BIOS 784/Data/GSE48024/gset.RData")
```

Working on male data

```
gset_male <- gset[[2]]
```

Subset to probes with gene name

Total 431 samples and 29228 probes.

```
gset_male <- gset_male[which(gset_male@featureData@data[["Gene symbol"]] != ""), ]

gset_male
```

```
## ExpressionSet (storageMode: lockedEnvironment)
## assayData: 29228 features, 431 samples
##   element names: exprs
## protocolData: none
## phenoData
##   sampleNames: GSM1165057 GSM1165058 ...
##     GSM1165487 (431 total)
##   varLabels: title geo_accession ...
##     treatment:ch1 (41 total)
##   varMetadata: labelDescription
## featureData
```

```
##   featureNames: ILMN_1343291 ILMN_1343295 ...
##     ILMN_2416019 (29228 total)
##   fvarLabels: ID Gene title ...
##     Platform_SEQUENCE (22 total)
##   fvarMetadata: Column Description
##     labelDescription
## experimentData: use 'experimentData(object)'
##   pubMedIds: 23878721
## 21357945
## Annotation: GPL6947
```

Set up summarized experiment object

```
SE_male  <- SummarizedExperiment(assay = list("exprs" = exprs(gset_male)),
                                 colData = as(gset_male@phenoData, "data.frame"),
                                 rowData = as(gset_male@featureData, "data.frame"))

table(SE_male$`time:ch1`)
```

```
##
##  Day0  Day1 Day14  Day3
##   111   110   109   101
```

Subset to participants sampled at all time

```
paticipants.all <- as.data.frame(table(SE_male$`subject:ch1`)) %>% filter(Freq == 4)

SE_male <- SE_male[, which(SE_male$`subject:ch1` %in% paticipants.all$Var1)]
```

Merge duplicate probes, averaging expression from the same gene

Said from limma for averaging expression: "This function should only be applied to normalized log-expression values, and not to raw unlogged expression values. It will generate an error message if applied to RGList or EListRaw objects."

```
rownames(SE_male)[1:10]
```

```
##  [1] "ILMN_1343291" "ILMN_1343295" "ILMN_1651209"
##  [4] "ILMN_1651210" "ILMN_1651228" "ILMN_1651229"
##  [7] "ILMN_1651232" "ILMN_1651235" "ILMN_1651237"
## [10] "ILMN_1651238"
```

```
expression.dupRM <- avereps(SE_male@assays@data@listData[["exprs"]], ID=SE_male@elementMetadata@listData

SE_male_dupRM <- SummarizedExperiment(assay = list("exprs" = expression.dupRM),
                                 colData = SE_male@colData)

SE_male_dupRM
```

```
## class: SummarizedExperiment
## dim: 19589 368
## metadata(0):
## assays(1): exprs
## rownames(19589): EEF1A1 GAPDH ... MCM10
##   ZNF703
```

```
## rowData names(0):
## colnames(368): GSM1165057 GSM1165058 ...
##   GSM1165486 GSM1165487
## colData names(41): title geo_accession ...
##   tissue:ch1 treatment:ch1
```

Remove lower 25% variant genes

```
SE_male_dupRM@metadata[["gene_mean"]] <- rowMeans(SE_male_dupRM@assays@data@listData[["exprs"]])
SE_male_dupRM@metadata[["gene_variance"]] <- rowVars(SE_male_dupRM@assays@data@listData[["exprs"]])

quantile(SE_male_dupRM@metadata[["gene_mean"]]) # 25% = 7.734803
```

```
##        0%       25%       50%       75%      100%
##  7.600727  7.734803  7.792982  8.081913 15.293659
```

```
quantile(SE_male_dupRM@metadata[["gene_variance"]]) # 25% =  0.0022386139, 75% = 0.0134488159
```

```
##           0%          25%          50%          75%
## 0.0003478975 0.0022386139 0.0039388560 0.0134488159
##         100%
## 2.6016623720
```

```
SE_male_dupRM_25up <- SE_male_dupRM[which(SE_male_dupRM@metadata[["gene_variance"]] > 0.0022386139), ]
SE_male_dupRM_25up
```

```
## class: SummarizedExperiment
## dim: 14692 368
## metadata(2): gene_mean gene_variance
## assays(1): exprs
## rownames(14692): EEF1A1 GAPDH ... MCM10
##   ZNF703
## rowData names(0):
## colnames(368): GSM1165057 GSM1165058 ...
##   GSM1165486 GSM1165487
## colData names(41): title geo_accession ...
##   tissue:ch1 treatment:ch1
```

```
SE_male_dupRM_75up <- SE_male_dupRM[which(SE_male_dupRM@metadata[["gene_variance"]] > 0.0134488159), ]
SE_male_dupRM_75up
```

```
## class: SummarizedExperiment
## dim: 4898 368
## metadata(2): gene_mean gene_variance
## assays(1): exprs
## rownames(4898): EEF1A1 GAPDH ... NFU1 SEP15
## rowData names(0):
## colnames(368): GSM1165057 GSM1165058 ...
##   GSM1165486 GSM1165487
## colData names(41): title geo_accession ...
##   tissue:ch1 treatment:ch1
```

```
#save(SE_male_dupRM_25up, file = "/Users/yu.zhang/Desktop/BIOS 784/Data/GSE48024/SE_male_dupRM_25up.RDa
#save(SE_male_dupRM_75up, file = "/Users/yu.zhang/Desktop/BIOS 784/Data/GSE48024/SE_male_dupRM_75up.RDa
```

After filtering out lower 25%, we are left with 14692 unique genes and 368 samples.

After filtering out lower 75%, we are left with 4898 unique genes and 368 samples.

DE analysis

Function for DE analysis

Currently using top 25% most variant genes.I have issue with mixed effect model on gene "ARRDC2" (false convergence?), which is in the upper 25%, not upper 75% in terms of variance. I need to look into the convergence problem.

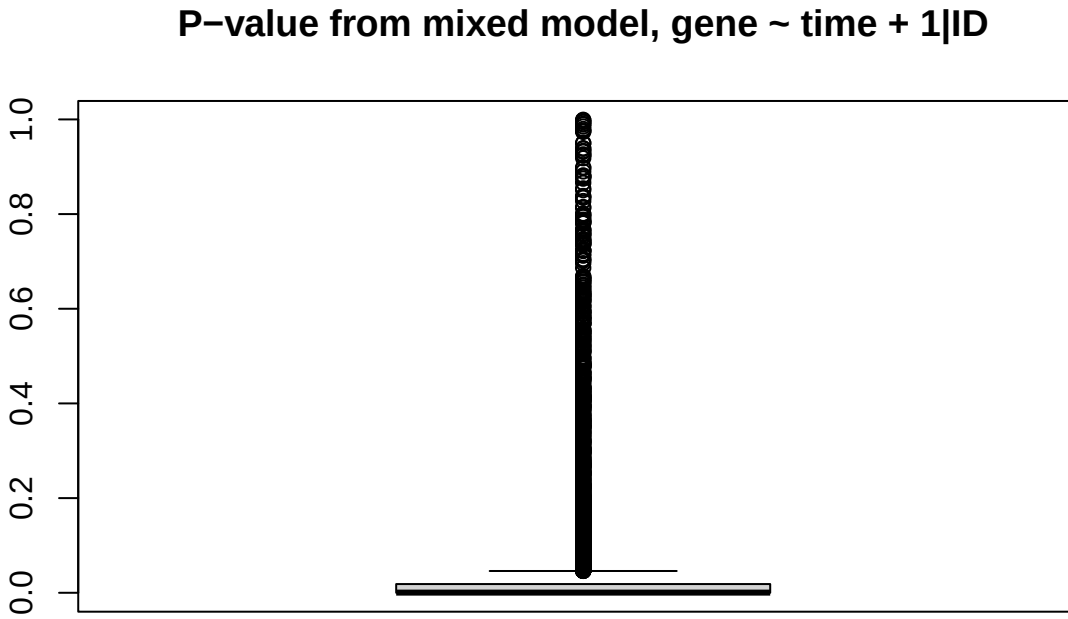
```
mixed_test <- function(gene,gene_name, covariates, fml, var_rowname){

  #gene = as.data.frame(SE_male_dupRM_25up@assays@data@listData[["exprs"]][which(rownames(SE_male_dupRM
  #gene_name = "ARRDC2"
  #covariates = cov
  #fml = as.formula(gene ~ time)
  #var_rowname = c("time")
  #print(gene_name)

  df_test <- cbind(gene , covariates)
  colnames(df_test)[1] <- "gene"
  mix_model <- lme(fml, data = df_test, random = ~1|ID, method = "ML", correlation = corCompSymm())
  df_ret <- as.data.frame(anova(mix_model, type="marginal")$`p-value`[2])
  rownames(df_ret) <- var_rowname
  colnames(df_ret) <- gene_name
  return(tibble(df_ret))
}
```

Mixed effect model, gene ~ time + 1|ID

```
# "ARRDC2" has convergence problem when running mixed effect model.

genes <- rownames(SE_male_dupRM_75up)

cov <- data.frame(ID = SE_male_dupRM_75up$`subject:ch1`,
                  gender = SE_male_dupRM_75up$`gender:ch1`,
                  time = as.character(SE_male_dupRM_75up$`time:ch1`))

mix_test_time <- sapply( 1:nrow(SE_male_dupRM_75up), function(i) mixed_test(   gene = as.data.frame(SE_
  gene_name = genes[i],
  covariates = cov,
  fml = as.formula(gene ~ time),
  var_rowname = c("time")) )

mix_test_time <- as.data.frame(unlist(mix_test_time))

colnames(mix_test_time) <- c("p_value_mix_compsym")

write.csv(mix_test_time, file = paste0(here(), "/code/2_DE_analysis/mix_pvalue_compsym.csv"))
```

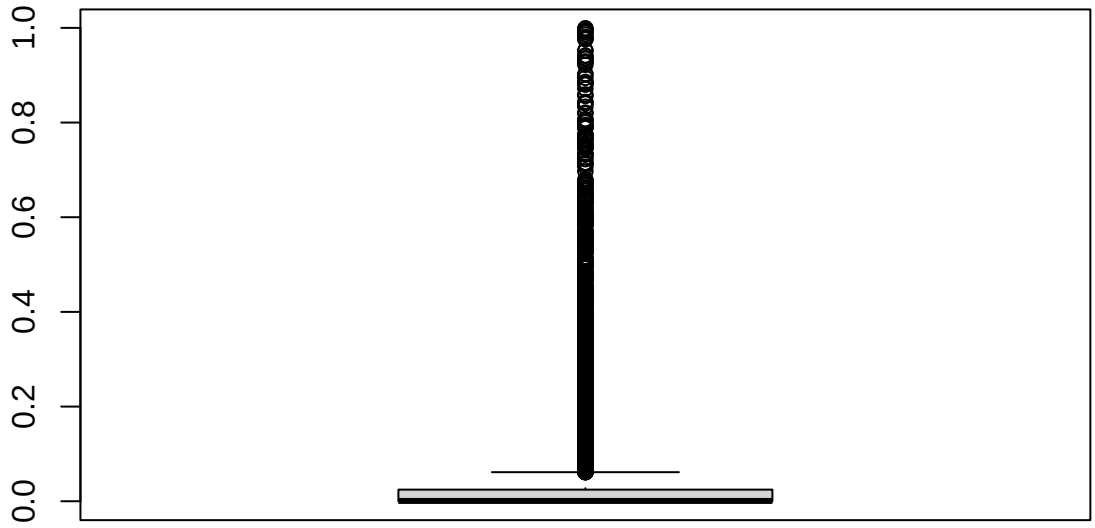
P-value from mixed effect model

```
mix_test_time <- read.csv(file = paste0(here(), "/code/2_DE_analysis/mix_pvalue_compsym.csv"))
rownames(mix_test_time) <- mix_test_time$X

boxplot(mix_test_time$p_value_mix_compsym, main ="P-value from mixed model, gene ~ time + 1|ID")
```



BH-adjusted p-value

```
# BH adjusted p-value
mix_test_time$BH_pvalue <- p.adjust(mix_test_time$p_value_mix_compsym, method =  "BH", n = length(mix_t

boxplot(mix_test_time$BH_pvalue, main ="BH adjusted P-value from mixed model, gene ~ time + 1|ID")
```




```
DEgene_pvalue_BH <- mix_test_time %>% as.data.frame() %>% filter(BH_pvalue < 0.05)

nrow(DEgene_pvalue_BH)
```

```
## [1] 3935
```