

5. Gene_expression_analysis

Fay

2022-08-09

load libraries

```
library(ggplot2)
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --
## v tibble 3.1.8      v dplyr 1.0.9
## v tidyr 1.2.0      v stringr 1.4.0
## v readr 2.1.2      v forcats 0.5.1
## v purrr 0.3.4

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()

library(optimx)
```

Import data:

Here, we have the experimental / field data, including imputed data

```
lab <- read.csv("output_data/gene_expression/data_products/lab_imputed_gene_expression.csv")
field <- read.csv("output_data/gene_expression/data_products/field_imputed_gene_expression.csv")
```

Selecting genes

```
# vectors for selecting gene columns
Genes_lab <- c("IFNy", "CXCR3", "IL.6", "IL.10", "IL.13", "IL1RN", "CASP1",
              "CXCL9", "IDO1", "IRGM1", "MPO", "MUC2", "MUC5AC",
              "MYD88", "NCR1", "PRF1", "RETNLB", "SOCS1", "TICAM1", "TNF")

Genes_field <- c("IFNy", "CXCR3", "IL.6", #"GBP2", "IL.12", "IRG6",
                 "IL.10", "IL.13", "IL1RN",
                 "CXCR3", "CASP1", "CXCL9",
                 "IDO1", "IRGM1", "MPO", "MUC2", "MUC5AC", "MYD88",
                 "NCR1", "PRF1", "RETNLB", "SOCS1", "TICAM1", "TNF")
```

It is time to apply the package of Alice Balard et al. on our predictions!

Let's see if we indeed have differences across the hybrid index with our predicted weight loss.

Install the package

```
##
## * checking for file '/tmp/Rtmpor4leT/remotes2e67d0799862c7/alicebalard-parasiteLoad-1b43216/DESCRIPTION' ... OK
## * preparing 'parasiteLoad':
## * checking DESCRIPTION meta-information ... OK
## * checking for LF line-endings in source and make files and shell scripts
## * checking for empty or unneeded directories
## * building 'parasiteLoad_0.1.0.tar.gz'
```

Applying Alice's package on every gene

```
x <- field$ID01

# Define function to be used to test, get the log lik and aic
tryDistrib <- function(x, distrib){
  # deals with fitdistr error:
  fit <- tryCatch(MASS::fitdistr(x, distrib), error=function(err) "fit failed")
  return(list(fit = fit,
              loglik = tryCatch(fit$loglik, error=function(err) "no loglik computed"),
              AIC = tryCatch(fit$aic, error=function(err) "no aic computed")))
}

findGoodDist <- function(x, distribs, distribs2){
  l =lapply(distribs, function(i) tryDistrib(x, i))
  names(l) <- distribs
  print(l)
  listDistr <- lapply(distribs2, function(i){
    if (i %in% "t"){
      fitdistrplus::fitdist(x, i, start = list(df =2))
    } else {
      fitdistrplus::fitdist(x,i)
    }
  })
  par(mfrow=c(2,2))
  denscomp(listDistr, legendtext=distribs2)
  cdfcomp(listDistr, legendtext=distribs2)
  qqcomp(listDistr, legendtext=distribs2)
  ppcomp(listDistr, legendtext=distribs2)
  par(mfrow=c(1,1))
}
```

```
tryDistrib(x, "normal")
```

Functions for testing distributions

```
## $fit
##      mean      sd
## 15.1432029  4.3044978
## ( 0.2278182) ( 0.1610918)
##
## $loglik
## [1] -1027.66
##
## $AIC
## NULL
```

```
tryDistrib(x, "binomial")
```

```
## $fit
## [1] "fit failed"
##
## $loglik
## [1] "no loglik computed"
##
## $AIC
## [1] "no aic computed"
```

```
tryDistrib(x, "student")
```

```
## $fit
## [1] "fit failed"
##
## $loglik
## [1] "no loglik computed"
##
## $AIC
## [1] "no aic computed"
```

```
tryDistrib(x, "weibull")
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
## $fit
##      shape      scale
## 3.6718724 16.7658776
## ( 0.1428414) ( 0.2566000)
##
## $loglik
## [1] -1032.752
##
## $AIC
## NULL
```

```
tryDistrib(x, "weibullshifted")
```

```
## $fit
## [1] "fit failed"
##
## $loglik
```

```
## [1] "no loglik computed"  
##  
## $AIC  
## [1] "no aic computed"
```