

## 07\_PSS-translation-prioritisation

zagor

23 12 2021

```
rm(list = ls(all = TRUE))
gc()

##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells 468881 25.1    1017805 54.4   643708 34.4
## Vcells 846290  6.5     8388608 64.0   1649073 12.6

library(crayon)

fp = file.path('..', 'intermediate')
list.files(fp)

## [1] "edges_to_preprocess.txt"
## [2] "FC_Ath_Stu_combo-redundant.txt"
## [3] "FC_Stu_combo.txt"
## [4] "nodes_for_prioritisation.txt"
## [5] "nodes_to_preprocess.txt"
## [6] "preprocessed_PSS2022-10-26.cys"
## [7] "preprocessed_PSS2022-10-26.graphml"
## [8] "preprocessed_PSS2022-10-26_edges.txt"
## [9] "preprocessed_PSS2022-10-26_nodes.txt"

fn = 'FC_Stu_combo.txt'
PSS.translate <- read.table(file.path(fp, fn),
                           sep="\t",
                           fill=TRUE,
                           header = TRUE,
                           stringsAsFactors = FALSE,
                           comment.char = "~",
                           quote = NULL,
                           strip.white = TRUE,
                           allowEscapes = TRUE)

fp = file.path('..', '..', '_A_01_networkExpress-R', 'input')
fn = 'mixedEXP2_AND_monoEXP1_logFC_padj_TMMcpm_narrow.txt'

mylen = count.fields(file.path(fp,fn), sep = "\t")
no_col <- max(mylen, na.rm = TRUE)
exp.values = read.table(file.path(fp,fn),
                        sep="\t",
```

```

      skip = 1,
      fill=TRUE,
      header = TRUE,
      # col.names=paste0("col", 1:no_col),
      stringsAsFactors = FALSE,
      comment.char = "@",
      na.strings = '#N/A',
      quote = NULL)

rownames(exp.values) = exp.values$GeneID
colnames(exp.values)[1] = 'geneID'
names(exp.values) = colnames(exp.values)

logFC = (exp.values[, grep('logFC', colnames(exp.values))])
logFC = as.matrix(logFC)
pval = (exp.values[, grep('adj\\.P\\.Val', colnames(exp.values))])
pval = as.matrix(pval)

pval[is.na(pval)] = 1
pval[pval >= 0.05] = 1

logFC[is.na(logFC)] = 0
logFC[which(pval == 1)] = 0

PSS.translate.x = PSS.translate
colnames(PSS.translate)

## [1] "nodeID"           "athDescription" "geneID"

colnames(exp.values)[1:4]

## [1] "geneID"        "BINCODE"       "NAME"          "DESCRIPTION"

colnames(logFC)

## [1] "leaves_inoculated_PS216_26h.leaves_noninoculated...logFC"
## [2] "leaves_inoculated_PS218_26h.leaves_noninoculated...logFC"
## [3] "roots_inoculated_PS216_2h_A.roots_noninoculated_A...logFC"
## [4] "roots_inoculated_PS218_2h_A.roots_noninoculated_A...logFC"
## [5] "roots_inoculated_PS216_26h_B.roots_noninoculated_B...logFC"
## [6] "roots_inoculated_PS218_26h_B.roots_noninoculated_B...logFC"
## [7] "roots_inoculated_PS_216_YFP.roots_noninoculated...logFC"
## [8] "roots_inoculated_PS_218_YFP.roots_noninoculated...logFC"
## [9] "roots_inoculated_PS_218_MKATE.roots_noninoculated...logFC"
## [10] "roots_inoculated_PS_218_YFP__PS_218_MKATE.roots_noninoculated...logFC"
## [11] "roots_inoculated_PS_216_YFP__PS_218_MKATE.roots_noninoculated...logFC"

logFC = as.data.frame(logFC)
colnames(logFC) = gsub('\\\\.\\\\.\\\\.logFC', '', colnames(logFC))
logFC$geneID = rownames(logFC)

```

```

PSS.translate.x = merge(PSS.translate.x, exp.values[, 1:4], all.x = FALSE, all.y = FALSE, by = 'geneID')
PSS.translate.x = merge(PSS.translate.x, logFC, all.x = FALSE, all.y = FALSE, by = 'geneID')

fp = file.path('..', 'intermediate')
fn = 'nodes_for_prioritisation.txt'
write.table(PSS.translate.x,
            file = file.path(fp, fn),
            append = FALSE,
            quote = FALSE,
            sep = "\t",
            eol = "\n",
            na = "-",
            dec = ".",
            row.names = FALSE,
            col.names = TRUE,
            qmethod = c("escape"),
            fileEncoding = "UTF-8")

rm(list=ls()[! ls() %in% c("PSS.translate.x")])
gc()

##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells  481700 25.8    1017805 54.4   1017805 54.4
## Vcells  880001  6.8    8388608 64.0   5193080 39.7

library(crayon)

logFC.starts.from.col = min(grep('\\.', colnames(PSS.translate.x)))

cat(red('0.00 to NA obligatory'))

## 0.00 to NA obligatory

PSS.translate.x[PSS.translate.x == 0.00] = NA

PSSgenes = sort(unique(PSS.translate.x$nodeID))
tmp = NULL

for (i in PSSgenes) {
  ind = which(PSS.translate.x$nodeID == i)
  if (length(ind) > 1) { # (length(ind) > 1) for found, not length(ind) > 1 for multiple (1 or more, do
    mysubset = PSS.translate.x[ind, ]
    e = sapply(1:nrow(mysubset), function(x) sum(!is.na(mysubset[x, logFC.starts.from.col:ncol(mysubset)])))
    f1 = sapply(1:nrow(mysubset),
               function(x) ifelse(all(is.na(mysubset[x, logFC.starts.from.col:ncol(mysubset)])),
                                    0,
                                    mean(colMeans(abs(mysubset[x, logFC.starts.from.col:ncol(mysubset)]))), n
    f2 = sapply(1:nrow(mysubset),
               function(x) ifelse(all(is.na(mysubset[x, logFC.starts.from.col:ncol(mysubset)])),

```

```

          0,
max(abs(mysubset[x, logFC.starts.from.col:ncol(mysubset)])), na.rm = TRUE)
g = which(e == max(e))
h1 = which(f1[g] == max(f1[g]))
h2 = which(f2[g] == max(f2[g]))
k2 = ifelse(length(h1) > 1, h2, h1)
k = ifelse(length(g) > 1, g[k2], g)

tmp = rbind(tmp, mysubset[k,])
} else { # no match
  tmp = rbind(tmp, PSS.translate.x[ind,]) # take first
}
}

nrow(tmp) == length(PSSgenes)

## [1] TRUE

prioritisedPSS = tmp

prioritisedPSS[prioritisedPSS$nodeID == 'LOX',]

##           geneID nodeID athDescription BINCODE
## 99 Sotub03g034620    LOX lipoxygenase 3 17.7.1.2
##                                     NAME DESCRIPTION
## 99 hormone metabolism.jasmonate.synthesis-degradation.lipoxygenase Lipoxygenase
##   leaves_inoculated_PS216_26h.leaves_noninoculated
##   99                                         2.63
##   leaves_inoculated_PS218_26h.leaves_noninoculated
##   99                                         3.53
##   roots_inoculated_PS216_2h_A.roots_noninoculated_A
##   99                                         NA
##   roots_inoculated_PS218_2h_A.roots_noninoculated_A
##   99                                         2.62
##   roots_inoculated_PS216_26h_B.roots_noninoculated_B
##   99                                         3.41
##   roots_inoculated_PS218_26h_B.roots_noninoculated_B
##   99                                         2.62
##   roots_inoculated_PS_216_YFP.roots_noninoculated
##   99                                         5.65
##   roots_inoculated_PS_218_YFP.roots_noninoculated
##   99                                         6.22
##   roots_inoculated_PS_218_MKATE.roots_noninoculated
##   99                                         6.68
##   roots_inoculated_PS_218_YFP__PS_218_MKATE.roots_noninoculated
##   99                                         5.26
##   roots_inoculated_PS_216_YFP__PS_218_MKATE.roots_noninoculated
##   99                                         7.1

cat(red('NA to 0.00'))

## NA to 0.00

```

```

fp = file.path('..', 'output')
fn = 'cytoscape_prioritisedPSS.txt'

write.table(prioritisedPSS,
            file = file.path(fp, fn),
            append = FALSE,
            quote = FALSE,
            sep = "\t",
            eol = "\n",
            na = "NA",
            dec = ".",
            row.names = FALSE,
            col.names = TRUE,
            qmethod = c("escape"),
            fileEncoding = "UTF-8")

fpo = file.path('..', 'output', 'dummyFilesForDiNAR_PSS')

dir.create(fpo)

## Warning in dir.create(fpo): '..\output\dummyFilesForDiNAR_PSS' already exists

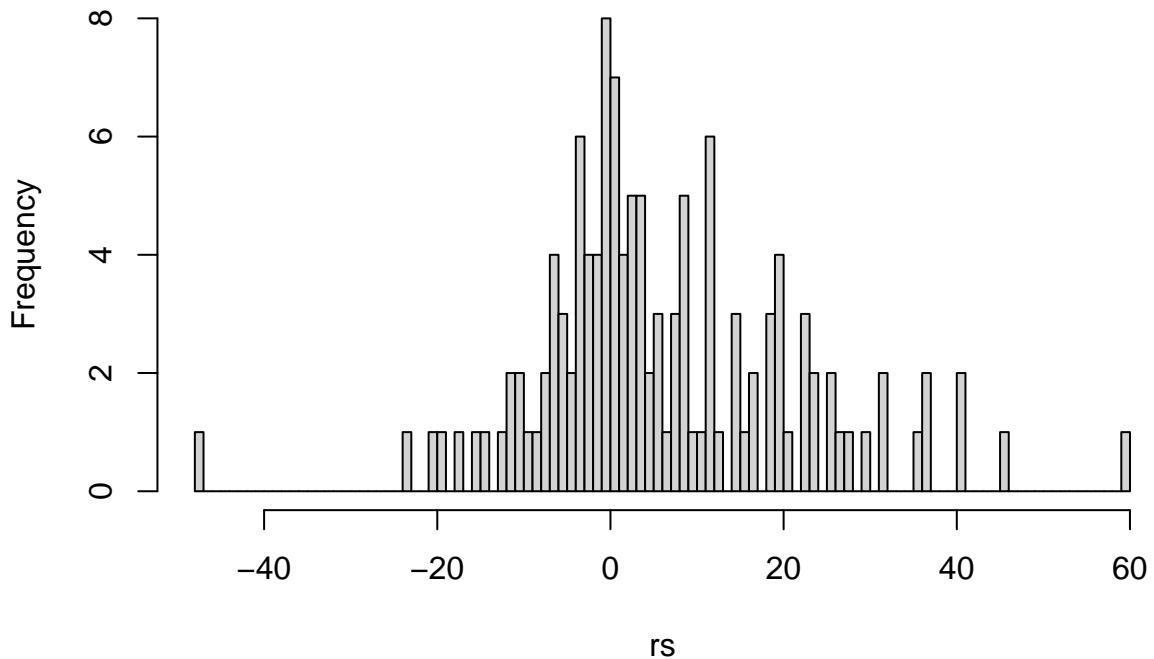
combo = prioritisedPSS[, c(grep('nodeID', colnames(prioritisedPSS)),
                           grep('\\.', colnames(prioritisedPSS)))]]

rs = rowSums(combo[, 2:ncol(combo)], na.rm = TRUE)

hist(rs, breaks = 100)

```

## Histogram of rs



```
# combo = combo[rs != 0,]

combo[is.na(combo)] = 0.0

cnt = 1

for (i in 2:ncol(combo)) {
  print(colnames(combo)[i])
  tmp = combo[, c(1, i, i)]
  tmp[,3] = 0
  colnames(tmp) = c('geneID', 'logFC', 'p.value')

  write.table(tmp,
              file = file.path(fpo, paste0(cnt, '_',
              colnames(combo)[i], '.txt')),
              append = FALSE,
              quote = FALSE,
              sep = "\t",
              eol = "\n", na = "",
              dec = ".",
              row.names = FALSE,
              col.names = TRUE, qmethod = c("escape"),
              fileEncoding = "UTF-8")
  cnt = cnt + 1
}

## [1] "leaves_inoculated_PS216_26h.leaves_noninoculated"
```

```
## [1] "leaves_inoculated_PS218_26h.leaves_noninoculated"
## [1] "roots_inoculated_PS216_2h_A.roots_noninoculated_A"
## [1] "roots_inoculated_PS218_2h_A.roots_noninoculated_A"
## [1] "roots_inoculated_PS216_26h_B.roots_noninoculated_B"
## [1] "roots_inoculated_PS218_26h_B.roots_noninoculated_B"
## [1] "roots_inoculated_PS_216_YFP.roots_noninoculated"
## [1] "roots_inoculated_PS_218_YFP.roots_noninoculated"
## [1] "roots_inoculated_PS_218_MKATE.roots_noninoculated"
## [1] "roots_inoculated_PS_218_YFP__PS_218_MKATE.roots_noninoculated"
## [1] "roots_inoculated_PS_216_YFP__PS_218_MKATE.roots_noninoculated"
```