GO enrichment

HBG

11/4/2018

This script perform GO enrichment using topGO for a set of genes that show high (top 10%)

Bayes factors in tumor cells, but low or moderate (0%-50%) BF in normal cells

topGO analysis

```r
# loading the libraries
library(topGO)
```

```
## Loading required package: BiocGenerics

## Loading required package: parallel

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind,
##     colMeans, colnames, colSums, dirname, do.call, duplicated,
##     eval, evalq, Filter, Find, get, grep, grepl, intersect,
##     is.unsorted, lapply, lengths, Map, mapply, match, mget, order,
##     paste, pmax, pmax.int, pmin, pmin.int, Position, rank, rbind,
##     Reduce, rowMeans, rownames, rowSums, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which, which.max,
##     which.min

## Loading required package: graph

## 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")'.

## Loading required package: GO.db
## Loading required package: AnnotationDbi

## Loading required package: stats4

## Loading required package: IRanges

## Loading required package: S4Vectors

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:base':
##
##     expand.grid

##

## Loading required package: SparseM

##
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':
##
##     backsolve

##
## groupGOTerms:    GOBPTerm, GOMFTerm, GOCCTerm environments built.

##
## Attaching package: 'topGO'

## The following object is masked from 'package:IRanges':
##
##     members
```

```r
library(biomaRt)
library(ensembldb)
```

```
## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: GenomicFeatures

##
## Attaching package: 'GenomicFeatures'

## The following object is masked from 'package:topGO':
##
##     genes

## Loading required package: AnnotationFilter

##
## Attaching package: 'ensembldb'

## The following object is masked from 'package:stats':
##
##     filter
```

```r
#library(mygene) ## it may be useful for gene name conversion
library(EnsDb.Hsapiens.v86)
library(GenomicFeatures)
library(GO.db)
```

```
## some packages are installed using the package 'BioManager'
## the package 'GenomicFeatures' is needed to install 'EnsDb.Hsapiens.v86'
## install_github('https://github.com/Bioconductor/GenomicFeatures')
## the library 'devtools' is needed to install GenomicFeatures from GitHub repository

# the BF data file
panc.file <- read.csv("../rnf43.csv",header=TRUE,stringsAsFactors=F)
panc.gene <- panc.file$gene
panc.panc <- panc.file$panc.mean
panc.np <- panc.file$nonpanc.mean
quant <- quantile(panc.panc, probs = seq(0,1,1/20))
quant.np <- quantile(panc.np, probs=seq(0,1,1/20))

# top 10% BF genes (1718) in tumor cells
panc.top10 <- panc.gene[which(panc.panc > quant[19])]
length(panc.top10)
```

```
## [1] 1718
```

```
# genes (8458) of 0% to 50% BF factors in normal cells
np.bottom50 <- panc.gene[which(panc.np < quant.np[11])]
length(np.bottom50)
```

```
## [1] 8458
```

```
# the overlap genes between top 10% tumor and (0-50%) normal cells
genes=panc.gene[which(panc.top10 %in% np.bottom50)]
length(genes)
```

```
## [1] 51
```

```
GeneNameFilter(genes)
```

```
## class: GeneNameFilter
## condition: ==
## value: FZD5 WLS HNF1A RBM15 PPCS PORCN TPK1 ADRBK1 NANS PPARG SLC2A1 STK40 MPI ALG3 WNT3 TRIB1 STX4 S
```

```
# Gene names need be converted to Ensembl IDs
library(EnsDb.Hsapiens.v86)
ensembl.id <- genes(EnsDb.Hsapiens.v86,
                    filter=list(GeneNameFilter(genes),GeneIdFilter("ENSG", "startsWith")),
                    return.type="data.frame", columns=c("gene_id"))
write.table(ensembl.id, file="hs.rnf43.gene.set.csv", sep=",", row.names = F, col.names = F, quote = F)
ensembl.id
```

```
##            gene_id gene_name
## 1  ENSG00000100412      ACO2
## 2  ENSG00000214160      ALG3
## 3  ENSG00000086848      ALG9
## 4  ENSG00000161203     AP2M1
## 5  ENSG00000137135  ARHGEF39
## 6  ENSG00000067248     DHX29
## 7  ENSG00000133884      DPF2
## 8  ENSG00000179151      EDC3
## 9  ENSG00000156030   ELMSAN1
## 10 ENSG00000204930   FAM221B
## 11 ENSG00000079459     FDFT1
## 12 ENSG00000163251      FZD5
## 13 ENSG00000117308      GALE
## 14 ENSG00000147533    GOLGA7
## 15 ENSG00000135100     HNF1A
## 16 ENSG00000186625    KATNA1
## 17 ENSG00000142515      KLK3
## 18 ENSG00000053747     LAMA3
## 19 ENSG00000132763    MMACHC
## 20 ENSG00000115275      MOGS
## 21 ENSG00000178802       MPI
## 22 ENSG00000116984       MTR
## 23 ENSG00000124275      MTRR
## 24 ENSG00000095380      NANS
## 25 ENSG00000072864      NDE1
## 26 ENSG00000275911      NDE1
## 27 ENSG00000141458      NPC1
## 28 ENSG00000166228     PCBD1
## 29 ENSG00000013375      PGM3
## 30 ENSG00000102312     PORCN
## 31 ENSG00000132170     PPARG
## 32 ENSG00000138621     PPCDC
## 33 ENSG00000127125      PPCS
## 34 ENSG00000011485     PPP5C
## 35 ENSG00000126464     PRR12
## 36 ENSG00000117425     PTCH2
## 37 ENSG00000204764   RANBP17
## 38 ENSG00000162775     RBM15
## 39 ENSG00000080345      RIF1
## 40 ENSG00000100075   SLC25A1
## 41 ENSG00000117394    SLC2A1
## 42 ENSG00000196182     STK40
## 43 ENSG00000103496      STX4
## 44 ENSG00000029639     TFB1M
## 45 ENSG00000196511      TPK1
## 46 ENSG00000173334     TRIB1
## 47 ENSG00000119541     VPS4B
## 48 ENSG00000162923     WDR26
## 49 ENSG00000116729       WLS
## 50 ENSG00000108379      WNT3
## 51 ENSG00000277626      WNT3
## 52 ENSG00000277641      WNT3
```

```
ensembl.id$gene_id
```

```
##  [1] "ENSG00000100412" "ENSG00000214160" "ENSG00000086848"
##  [4] "ENSG00000161203" "ENSG00000137135" "ENSG00000067248"
##  [7] "ENSG00000133884" "ENSG00000179151" "ENSG00000156030"
## [10] "ENSG00000204930" "ENSG00000079459" "ENSG00000163251"
## [13] "ENSG00000117308" "ENSG00000147533" "ENSG00000135100"
## [16] "ENSG00000186625" "ENSG00000142515" "ENSG00000053747"
## [19] "ENSG00000132763" "ENSG00000115275" "ENSG00000178802"
## [22] "ENSG00000116984" "ENSG00000124275" "ENSG00000095380"
## [25] "ENSG00000072864" "ENSG00000275911" "ENSG00000141458"
## [28] "ENSG00000166228" "ENSG00000013375" "ENSG00000102312"
## [31] "ENSG00000132170" "ENSG00000138621" "ENSG00000127125"
## [34] "ENSG00000011485" "ENSG00000126464" "ENSG00000117425"
## [37] "ENSG00000204764" "ENSG00000162775" "ENSG00000080345"
## [40] "ENSG00000100075" "ENSG00000117394" "ENSG00000196182"
## [43] "ENSG00000103496" "ENSG00000029639" "ENSG00000196511"
## [46] "ENSG00000173334" "ENSG00000119541" "ENSG00000162923"
## [49] "ENSG00000116729" "ENSG00000108379" "ENSG00000277626"
## [52] "ENSG00000277641"
```

```
# Enrichment analysis
mart <- useDataset("hsapiens_gene_ensembl", mart=useMart("ensembl"))
all_ensembl_gene_id <- getBM(attributes = "ensembl_gene_id",
                             values = "*", mart = mart)
all <- factor(as.integer (all_ensembl_gene_id[,1] %in% ensembl.id$gene_id))
names(all) <- all_ensembl_gene_id[,1]

GOdata <- new("topGOdata", ontology="BP",
              allGenes = all, geneSel=function(p) p == 1,
              description = "P.not.NP", annot=annFUN.org, mapping="org.Hs.eg.db", ID="Ensembl")
```

```
##
## Building most specific GOs .....
```

```
## Loading required package: org.Hs.eg.db
```

```
##
```

```
##  ( 12078 GO terms found. )
```

```
##
## Build GO DAG topology ..........
```

```
##  ( 16113 GO terms and 38254 relations. )
```

```
##
## Annotating nodes ...............
```

```
##  ( 20504 genes annotated to the GO terms. )
```

```
result.test <- runTest(GOdata, algorithm = "classic", statistic = "fisher")
```

```
##
##           -- Classic Algorithm --
##
##       the algorithm is scoring 1966 nontrivial nodes
##       parameters:
##           test statistic: fisher
```

```
resultKS.test <- runTest(GOdata, algorithm = "classic", statistic = "ks")
```

```
##
##           -- Classic Algorithm --
##
##       the algorithm is scoring 16113 nontrivial nodes
##       parameters:
##           test statistic: ks
##           score order: increasing
```

```
resultKS.elim.test <- runTest(GOdata, algorithm = "elim", statistic = "ks")
```

```
##
##           -- Elim Algorithm --
##
##       the algorithm is scoring 16113 nontrivial nodes
##       parameters:
##           test statistic: ks
##           cutOff: 0.01
##           score order: increasing
##
##   Level 20:  1 nodes to be scored    (0 eliminated genes)
##
##   Level 19:  5 nodes to be scored    (0 eliminated genes)
##
##   Level 18:  22 nodes to be scored   (0 eliminated genes)
##
##   Level 17:  52 nodes to be scored   (13 eliminated genes)
##
##   Level 16:  120 nodes to be scored  (31 eliminated genes)
##
##   Level 15:  255 nodes to be scored  (195 eliminated genes)
##
##   Level 14:  499 nodes to be scored  (354 eliminated genes)
##
##   Level 13:  919 nodes to be scored  (1156 eliminated genes)
##
##   Level 12:  1374 nodes to be scored (3460 eliminated genes)
##
##   Level 11:  1786 nodes to be scored (6096 eliminated genes)
##
##   Level 10:  2092 nodes to be scored (7714 eliminated genes)
##
##   Level 9:   2191 nodes to be scored (9857 eliminated genes)
##
##   Level 8:   2088 nodes to be scored (12111 eliminated genes)
##
##   Level 7:   1942 nodes to be scored (13740 eliminated genes)
##
##   Level 6:   1453 nodes to be scored (15133 eliminated genes)
##
##   Level 5:   779 nodes to be scored  (16447 eliminated genes)
##
##   Level 4:   375 nodes to be scored  (17342 eliminated genes)
##
##   Level 3:   135 nodes to be scored  (17551 eliminated genes)
```

```
##
##   Level 2:   24 nodes to be scored   (17895 eliminated genes)

##
##   Level 1:   1 nodes to be scored    (17905 eliminated genes)
```

```
allRes <- GenTable(GOdata, classicFisher = result.test, classicKS = resultKS.test, elimKS = resultKS.eli

allRes
```

```
##         GO.ID                                        Term Annotated
## 1  GO:0045944 positive regulation of transcription by ...      1273
## 2  GO:0000122 negative regulation of transcription by ...       904
## 3  GO:0008283                          cell proliferation      2322
## 4  GO:0043687     post-translational protein modification       401
## 5  GO:0051301                               cell division       625
## 6  GO:0045893 positive regulation of transcription, DN...      1621
## 7  GO:0008584                      male gonad development       137
## 8  GO:1990830 cellular response to leukemia inhibitory...        99
## 9  GO:0035556           intracellular signal transduction      3075
## 10 GO:0046777                 protein autophosphorylation       245
## 11 GO:0007411                               axon guidance       267
## 12 GO:0001701              in utero embryonic development       338
## 13 GO:0006406                    mRNA export from nucleus       123
## 14 GO:0043547      positive regulation of GTPase activity       456
## 15 GO:0018105             peptidyl-serine phosphorylation       294
## 16 GO:0007601                           visual perception       226
## 17 GO:0045471                         response to ethanol       132
## 18 GO:0010628      positive regulation of gene expression      2104
## 19 GO:0043627                        response to estrogen        74
## 20 GO:0035735 intraciliary transport involved in ciliu...        40
##    Significant Expected Rank in classicFisher classicFisher classicKS
## 1            5     3.04                  1101         0.186   5.9e-23
## 2            2     2.16                  1810         0.643   2.7e-13
## 3            9     5.55                   795         0.097   3.2e-18
## 4            0     0.96                  1966         1.000   1.2e-11
## 5            4     1.49                   631         0.062   7.3e-18
## 6            6     3.87                  1105         0.188   < 1e-30
## 7            0     0.33                  1967         1.000   1.5e-10
## 8            1     0.24                  1176         0.211   3.6e-09
## 9            6     7.35                  1888         0.764   < 1e-30
## 10           0     0.59                  1968         1.000   8.0e-09
## 11           3     0.64                   386         0.026   8.9e-10
## 12           1     0.81                  1745         0.558   6.3e-11
## 13           0     0.29                  1969         1.000   7.9e-08
## 14           0     1.09                  1970         1.000   8.6e-08
## 15           0     0.70                  1971         1.000   4.7e-11
## 16           0     0.54                  1972         1.000   1.3e-07
## 17           0     0.32                  1973         1.000   1.3e-07
## 18          10     5.03                   381         0.025   < 1e-30
## 19           1     0.18                  1029         0.163   2.3e-07
## 20           0     0.10                  1974         1.000   2.5e-07
##     elimKS
## 1  6.6e-22
## 2  2.7e-13
## 3  1.1e-12
## 4  1.2e-11
## 5  3.2e-10
## 6  9.1e-10
## 7  1.4e-09
## 8  3.6e-09
## 9  5.0e-09
## 10 8.0e-09
## 11 1.6e-08
## 12 5.4e-08
## 13 7.9e-08
## 14 8.6e-08
## 15 1.1e-07
## 16 1.3e-07
## 17 1.3e-07
## 18 1.4e-07
## 19 2.3e-07
## 20 2.5e-07
```