signatureSearch package

Author: Yuzhu Duan (yduan004@ucr.edu)

Last update: 08 November, 2018

Contents

Methods for GESS

Get longevity related query signature from Peters et al. (2015)

The query signature are 150 up and 150 down regulated gene sets from 1,497 age-related genes after differential expression analysis

```
## Obtain 150 up and 150 down regulated gene sets from Peters
Peters <- read.delim("~/insync/project/longevityTools_eDRUG/data/PMID26490707_S1.xls", comment="#", che
#Peters <- read.delim("/rhome/tgirke/Projects/longevity/longevityTools/vignettes/data/PMID26490707_S1.x
Peters <- Peters[!duplicated(Peters[,"NEW-Entrez-ID"]),]
query <- Peters$Zscore; names(query) <- Peters[,"NEW-Entrez-ID"]
upset <- head(names(query[order(-query)]), 150)
downset <- tail(names(query[order(-query)]), 150)

# Load public CMAP, LINCS and LINCS_sub database
se_cmap <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/data/cmap")
se_lincs <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/data/lincs42
se_lincs_sub <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/data/lin

# Load public CMAP_expr, LINCS_expr and LINCS_sub_expr database
se_cmap_expr <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/data/cma
se_lincs_expr <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/data/li
se_lincs_sub_expr <- loadHDF5SummarizedExperiment("~/insync/project/lincs_gse92742_dataset_analysis/dat
```

View cell type information in LINCS database

```
cell_info <- metadata(se_lincs)$cell_info
cell_info
```

Searching with CMAP method for GESS

Generate “qSig” object for GESS_CMAP method and search against CMAP, LINCS, LINCS_sub reference database

```
# Search against CMAP reference database
qsig_cmap_against_cmap <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "CMAP", refdb =
cmap_against_cmap <- gess_cmap(qSig=qsig_cmap_against_cmap, chunk_size=5000)

# Search against LINCS reference database
qsig_cmap_against_lincs <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "CMAP", refdb
cmap_against_lincs <- gess_cmap(qSig=qsig_cmap_against_lincs, chunk_size=5000)

# Search against LINCS_sub reference database
qsig_cmap_against_lincs_sub <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "CMAP", re
cmap_against_lincs_sub <- gess_cmap(qSig=qsig_cmap_against_lincs_sub, chunk_size=5000)
```

```
cmap_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/cmap_against_cmap.rds")
cmap_against_cmap@result
cmap_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/cmap_against_lincs.rds")
cmap_against_lincs@result
cmap_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/cmap_against_lincs_sub.rds")
cmap_against_lincs_sub@result
```

Searching with LINCS method for GESS

Generate “qSig” object for GESS_LINCS method and search against CMAP, LINCS, LINCS_sub reference database

```
# Search against CMAP reference database
qsig_lincs_against_cmap <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "LINCS", refdb
lincs_against_cmap <- gess_lincs(qsig_lincs_against_cmap, sortby="WTCS")

# Search against LINCS reference database
qsig_lincs_against_lincs <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "LINCS", refd
lincs_against_lincs <- gess_lincs(qSig=qsig_lincs_against_lincs, ES_NULL="Default", taurefList="Default

# Search against LINCS_sub reference database
qsig_lincs_against_lincs_sub <- qSig(qsig = list(upset=upset, downset=downset), gess_method = "LINCS",
lincs_against_lincs_sub <- gess_lincs(qsig_lincs_against_lincs_sub, sortby="NCS")
```

```
lincs_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/lincs_against_cmap.rds")
lincs_against_cmap@result
lincs_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/lincs_against_lincs.rds")
lincs_against_lincs@result
lincs_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/lincs_against_lincs_sub.rds")
lincs_against_lincs_sub@result
```

Searching with gCMAP method for GESS

Generate “qSig” object for GESS_gCMAP method and search against CMAP, LINCS, LINCS_sub reference database

```
query <- as.matrix(assay(se_lincs)[,"sirolimus__MCF7__trt_cp"])

# Search against CMAP reference database
qsig_gcmap_against_cmap <- qSig(qsig = as.matrix(query), gess_method = "gCMAP", refdb = se_cmap)
gcmap_against_cmap <- gess_gcmap(qSig=qsig_gcmap_against_cmap, higher=1, lower=-1, chunk_size=5000)

# Search against LINCS reference database
qsig_gcmap_against_lincs <- qSig(qsig = as.matrix(query), gess_method = "gCMAP", refdb = se_lincs)
gcmap_against_lincs <- gess_gcmap(qSig=qsig_gcmap_against_lincs, higher=1, lower=-1, chunk_size=5000)

# Search against LINCS_sub reference database
qsig_gcmap_against_lincs_sub <- qSig(qsig = as.matrix(query), gess_method = "gCMAP", refdb = se_lincs_su
gcmap_against_lincs_sub <- gess_gcmap(qsig_gcmap_against_lincs_sub, higher=1, lower=-1, chunk_size=5000
```

```
gcmap_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/gcmap_against_cmap.rds")
gcmap_against_cmap@result
gcmap_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/gcmap_against_lincs.rds")
gcmap_against_lincs@result
gcmap_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/gcmap_against_lincs_sub.rds")
gcmap_against_lincs_sub@result
```

Searching with Fisher method for GESS

Generate “qSig” object for GESS_fisher method and search against CMAP, LINCS, LINCS_sub reference database

```
query <- as.matrix(assay(se_lincs)[,"sirolimus__MCF7__trt_cp"])

# Search against CMAP reference database
qsig_fisher_against_cmap <- qSig(qsig = as.matrix(query), gess_method = "Fisher", refdb = se_cmap)
fisher_against_cmap <- gess_fisher(qSig=qsig_fisher_against_cmap, higher=1, lower=-1, chunk_size=5000)

# Search against LINCS reference database
qsig_fisher_against_lincs <- qSig(qsig = as.matrix(query), gess_method = "Fisher", refdb = se_lincs)
fisher_against_lincs <- gess_fisher(qSig=qsig_fisher_against_lincs, higher=1, lower=-1, chunk_size=5000

# Search against LINCS_sub reference database
qsig_fisher_against_lincs_sub <- qSig(qsig = as.matrix(query), gess_method = "Fisher", refdb = se_lincs
fisher_against_lincs_sub <- gess_fisher(qSig=qsig_fisher_against_lincs_sub, higher=1, lower=-1, chunk_si
```

```
fisher_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/fisher_against_cmap.rds")
fisher_against_cmap@result
fisher_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/fisher_against_lincs.rds")
fisher_against_lincs@result
fisher_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/fisher_against_lincs_sub.rds
fisher_against_lincs_sub@result
```

Searching with Spearman method for GESS

Generate “qSig” object for GESS_SP method and search against CMAP, LINCS, LINCS_sub reference database

Genome-wide Spearman correlation

```r
# Use genome-wide expression data of sirolimus treatment in MCF7 cells in CMAP_expr database as query
query <- as.matrix(assay(se_cmap_expr)[,"sirolimus__MCF7__trt_cp"])

# Search against CMAP_expr reference database
qsig_sp_against_cmap <- qSig(qsig = as.matrix(query), gess_method = "SP", refdb = se_cmap_expr)
sp_against_cmap <- gess_sp(qSig=qsig_sp_against_cmap, chunk_size=5000)

# Search against LINCS_expr reference database
qsig_sp_against_lincs <- qSig(qsig = as.matrix(query), gess_method = "SP", refdb = se_lincs_expr)
sp_against_lincs <- gess_sp(qSig=qsig_sp_against_lincs, chunk_size=5000)

# Search against LINCS_sub_expr reference database
qsig_sp_against_lincs_sub <- qSig(qsig = as.matrix(query), gess_method = "SP", refdb = se_lincs_sub_expr
sp_against_lincs_sub <- gess_sp(qSig=qsig_sp_against_lincs_sub, chunk_size=5000)
```

```r
sp_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/sp_against_cmap.rds")
sp_against_cmap@result
sp_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/sp_against_lincs.rds")
sp_against_lincs@result
sp_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/sp_against_lincs_sub.rds")
sp_against_lincs_sub@result
```

Spearman sub correlatioin

```r
# Subset expression values of 150 up and down gene sets from "sirolimus__MCF7__trt_cp" signature.

## get 150 up and down gene sets from "sirolimus__MCF7__trt_cp" signature.
sig <- sort(as.matrix(assay(se_lincs)[,"sirolimus__MCF7__trt_cp"]), decreasing = TRUE)
upset <- names(head(sig, 150)); downset <- names(tail(sig,150))
## get expression values of genes in up and down set
query <- as.matrix(assay(se_cmap_expr)[,"sirolimus__MCF7__trt_cp"])
query_sub <- query[c(upset, downset)]

# Search against CMAP_expr reference database
qsig_spsub_against_cmap <- qSig(qsig = as.matrix(query_sub), gess_method = "SP", refdb = se_cmap_expr)
spsub_against_cmap <- gess_sp(qSig=qsig_spsub_against_cmap, chunk_size=5000)

# Search against LINCS_expr reference database
qsig_spsub_against_lincs <- qSig(qsig = as.matrix(query_sub), gess_method = "SP", refdb = se_lincs_expr
spsub_against_lincs <- gess_sp(qSig=qsig_spsub_against_lincs, chunk_size=5000)

# Search against LINCS_sub_expr reference database
qsig_spsub_against_lincs_sub <- qSig(qsig = as.matrix(query_sub), gess_method = "SP", refdb = se_lincs_
spsub_against_lincs_sub <- gess_sp(qSig=qsig_spsub_against_lincs_sub, chunk_size=5000)
```

```r
spsub_against_cmap <- readRDS("~/insync/project/GESS_and_FEA/results/spsub_against_cmap.rds")
spsub_against_cmap@result
spsub_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/spsub_against_lincs.rds")
spsub_against_lincs@result
spsub_against_lincs_sub <- readRDS("~/insync/project/GESS_and_FEA/results/spsub_against_lincs_sub.rds")
spsub_against_lincs_sub@result
```

Methods for FEA

dup_hyperG method for TSEA

Subset top 100 ranking drugs in GESS result, get their target set (with duplication) as query for dup_hypergeometric test for TSEA. Here choose `lincs_res_vs_lincs` as GESS result.

```
lincs_against_lincs <- readRDS("~/insync/project/GESS_and_FEA/results/lincs_against_lincs.rds")
drugs <- unique(result(lincs_against_lincs)$pert[1:100])
# GO annotation system
dup_hyperG_res <- tsea_dup_hyperG(drugs = drugs, universe = "Default", type = "GO", ont="MF", pvalueCut
# KEGG annotation system
dup_hyperG_k_res <- tsea_dup_hyperG(drugs = drugs, universe = "Default", type = "KEGG", pvalueCutoff=0.
```

m_GSEA method for TSEA

Subset top 100 ranking drugs in GESS result as query, m_GSEA method internally get their target set and ranked target list with scores as query for TSEA. The scores represent weight of targets in the target set. Here choose `lincs_res_vs_lincs` as GESS result.

```
# GO annotation system
geneSets <- readRDS("~/insync/project/GESS_and_FEA/data/geneSets_470.rds")
geneList <- readRDS("~/insync/project/GESS_and_FEA/data/geneList.rds")
tmp_res <- fgsea2(pathways=geneSets, stats=geneList, nperm=1000, minSize=5, maxSize=500, gseaParam=1, n

mgsea_res <- tsea_mGSEA(drugs=drugs, type="GO", ont="MF", exponent=1, nPerm=1000, pAdjustMethod="BH", p
# KEGG annotation system
mgsea_k_res <- tsea_mGSEA(drugs=drugs, type="KEGG", exponent=1, nPerm=1000, pAdjustMethod="BH", pvalueC
```

mabs method for TSEA

Subset top 100 ranking drugs in GESS result as query, mabs method internally get their target set and ranked target list with scores as query for TSEA. The scores represent weight of targets in the target set. Here choose `lincs_res_vs_lincs` as GESS result.

```
# GO annotation system
mabs_res <- tsea_mabs(drugs=drugs, type="GO", ont="MF", nPerm=1000, pAdjustMethod="BH", pvalueCutoff=0.
# KEGG annotation system
mabs_k_res <- tsea_mabs(drugs=drugs, type="KEGG", nPerm=1000, pAdjustMethod="BH", pvalueCutoff=0.5, min
```

```
dup_hyperG_res <- readRDS("~/insync/project/GESS_and_FEA/results/dup_hyperG_res.rds")
dup_hyperG_res@result
dup_hyperG_k_res <- readRDS("~/insync/project/GESS_and_FEA/results/dup_hyperG_k_res.rds")
dup_hyperG_k_res@result

mgsea_res <- readRDS("~/insync/project/GESS_and_FEA/results/mgsea_res.rds")
mgsea_res@result
mgsea_k_res <- readRDS("~/insync/project/GESS_and_FEA/results/mgsea_k_res.rds")
mgsea_k_res@result

mabs_res <- readRDS("~/insync/project/GESS_and_FEA/results/mabs_res.rds")
mabs_res@result
mabs_k_res <- readRDS("~/insync/project/GESS_and_FEA/results/mabs_k_res.rds")
mabs_k_res@result
```

hyperG method for DSEA

Subset top 100 ranking drugs in GESS result as query for hypergeometric test for DSEA, here choose `lincs_res_vs_lincs` as GESS result.

```
drugs <- unique(result(lincs_against_lincs)$pert[1:100])
# GO annotation system
hyperG_res <- dsea_hyperG(drugs = drugs, type = "GO", ont="MF", pvalueCutoff=0.1, pAdjustMethod="BH", q
# KEGG annotation system
hyperG_k_res <- dsea_hyperG(drugs = drugs, type = "KEGG", pvalueCutoff=0.1, pAdjustMethod="BH", qvalueCu
```

GSEA method for DSEA

Use ranked drug list in GESS result as query for GSEA for DSEA, the scores are similarity scores of corresponding GESS methods. Zeros are removed. Here choose `lincs_res_vs_lincs` as GESS result.

```
dl <- abs(lincs_against_lincs@result$NCS); names(dl) <- lincs_against_lincs@result$Pert
dl <- dl[dl>0]
dl <- dl[!duplicated(names(dl))]
# GO annotation system
gsea_res <- dsea_GSEA(drugList=dl, type="GO", ont="MF", exponent=1, nPerm=1000, pAdjustMethod="BH", pval

# KEGG annotation system
gsea_k_res <- dsea_GSEA(drugList=dl, type="KEGG", exponent=1, nPerm=1000, pAdjustMethod="BH", pvalueCut
```

```
hyperG_res <- readRDS("~/insync/project/GESS_and_FEA/results/hyperG_res.rds")
hyperG_res@result
hyperG_k_res <- readRDS("~/insync/project/GESS_and_FEA/results/hyperG_k_res.rds")
hyperG_k_res@result

gsea_res <- readRDS("~/insync/project/GESS_and_FEA/results/gsea_res.rds")
gsea_res@result
gsea_k_res <- readRDS("~/insync/project/GESS_and_FEA/results/gsea_k_res.rds")
gsea_k_res@result
```

Visulization

Construct drug-target interaction networks in interested GO categories

Build drug-target networks in top ranking GO categories in `hyperG_res`

```
dtnetplot(drugs = hyperG_res@drug, set = "GO:0030169", ont = "MF")
```

Construct drug-target interaction networks in interested KEGG pathways or defined other gene/protein sets

Build drug-target networks in top ranking KEGG pathways in `hyperG_k_res`

```
dtnetplot(drugs = hyperG_k_res@drug, set = "hsa04979")
```