signatureSearch package

Author: Yuzhu Duan (yduan004@ucr.edu)

Last update: 17 December, 2018

Contents

Background

This project is about optimizing signature search and enrichment methods for the discovery of novel modes of action (MOA) of bioactive compounds from reference databases, such as LINCS, containing the genome-wide gene expression signatures (GESs) from tens of thousands of drug and genetic perturbations. The methods used by this prediction workflow can be divided into two major classes. First, gene expression signature search (GESS) methods are used to identify drugs that induce GESs similar to those of query GESs of interest. The queries can be drug- or disease-related GESs. Since the MOA of most drugs in the corresponding reference databases are known, the resulting associations are useful to gain insights into pharmacological and disease mechanisms and to develop novel drug repurposing approaches. Second, functional enrichment analysis (FEA) methods using Gene Ontology (GO) or pathway annotations have been developed to functionally interpret the vast number of GESS results generated by this project. The latter are composed of lists of drugs ranked by the similarity metric of the corresponding GESS method making the functional interpretation of their top ranking drugs challenging. Importantly, the FEA methods developed by this study also support the reconstruction of drug-target networks to guide the interpretation of the results.

Gene Expression Signature Databases

[?] generated a GES database, initially including 164 drugs screened against four mammalian cell lines (Lamb et al. 2006). A few years later it was extended to 1,309 drugs and eight cell lines. In 2017, the database was increased to 19,811 drugs by the Library of Network-Based Cellular Signatures (LINCS) Consortium [?], while the number of cell types represented in the database has increased to over 70 normal and cancer cell lines. The number of compound dosages and time points considered in the assays has also been increased by 10-20 fold. The initial database used Affymetrix Gene Chips as expression platform. To scale from a few thousand to many hundred thousand GESs, the LINCS Consortium uses now the more economic L1000 assay. This bead-based technology is a low cost, high-throughput reduced representation expression profiling assay. It measures the expression of 978 landmark genes and 80 control genes by detecting fluorescent intensity of beads after capturing the ligation-mediated amplification products of mRNAs Peck et al. (2006). The expression of 11,350 additional genes is imputed from the landmark genes by using as training data a collection of 12,063 Affymetrix gene chips (Edgar, Domrachev, and Lash 2002). In the [?] paper the authors use a subset of the LINCS database, referred to as Touchstone, containing 2,837 drug treatments of nine cell lines that can be searched on their project's website (CLUE). The substantial scale-up of the LINCS project provides now many new opportunities to explore MOAs for a vast number of small molecules.

Terminology

Gene Expression Signatures (GESs): can be Gene Expression Profiles (GEPs), which can be genome-wide profile from differential expression analysis (log2FC or Z-scores) or gene expression intensity values from perturbagen treatment in cell culture, or Differentially Expressed Gene sets (DEGs) from a treatment.

Database

The CMAP and LINCS differential expression databases can be built with `build_db` funciton or directly downloaded at CMAP, LINCS. The databases are stored as HDF5 backed `SummarizedExperiment` object. Untar the downloaded file by "tar -xzvf file.tar.gz" command, then load the `SummarizedExperiment` object via `loadHDF5SummarizedExperiment` function in `HDF5Array` package. The 'assays' slot of the loaded SummarizedExperiment object represents the logFC or Z scores generated from differential expression (DE) analysis.

The CMAP and LINCS expression databases can also be built with `build_db` funciton or directly downloaded at CMAP_expr, LINCS_expr. They stores the mean experssion values of drug treatment samples in different cells.

The custom database can also be build via `build_db` function if a `data.frame` representing genoime-wide GEPs (log2FC, Z-scores, intensity values, etc.) of compound or genetic treatments in cells is provided.

Methods for GESS

For demonstration of the use of GESS and FEA workflow, I subsetted 1000 signatures from LINCS DE database, which can be downloaded at lincs_sub. The query signature is drawn from the `lincs_sub` database for searching.

Load database and get query signature

```
db_dir <- file.path(tempdir(),"lincs42_sub")
db_path <- file.path(tempdir(), "lincs42_sub.tar.gz")
download.file("http://biocluster.ucr.edu/~yduan004/LINCS_db/lincs42_sub.tar.gz", db_path, quiet = TRUE)
untar(db_path, exdir = tempdir())
lincs_sub <- loadHDF5SummarizedExperiment(db_dir)
```

```
## get "sirolimus__HEPG2__trt_cp" signature drawn from "lincs_sub" database
query_mat <- as.matrix(assay(lincs_sub[,"sirolimus__HEPG2__trt_cp"]))
query = as.numeric(query_mat); names(query) = rownames(query_mat)
upset <- head(names(query[order(-query)]), 150)
downset <- tail(names(query[order(-query)]), 150)
```

View cell type information in lincs_sub database

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

Searching with CMAP method for GESS

Generate "qSig" object for GESS_CMAP method and 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_lincs_sub
```

Searching with LINCS method for GESS

Generate "qSig" object for GESS_LINCS method and 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_lincs_sub
result(lincs_against_lincs_sub)
```

Searching with gCMAP method for GESS

Generate "qSig" object for GESS_gCMAP method and search against lincs_sub reference database

```
qsig_gcmap_against_lincs_sub <- qSig(qsig = query_mat, gess_method = "gCMAP", refdb = lincs_sub)
gcmap_against_lincs_sub <- gess_gcmap(qsig_gcmap_against_lincs_sub, higher=1, lower=-1, chunk_size=5000
result(gcmap_against_lincs_sub)
```

Searching with Fisher method for GESS

Generate "qSig" object for GESS_fisher method and search against lincs_sub reference database

```
qsig_fisher_against_lincs_sub <- qSig(qsig = query_mat, gess_method = "Fisher", refdb = lincs_sub)
fisher_against_lincs_sub <- gess_fisher(qSig=qsig_fisher_against_lincs_sub, higher=1, lower=-1, chunk_s
result(fisher_against_lincs_sub)
```

Searching with Spearman method for GESS

Generate "qSig" object for GESS_Cor method and search against lincs_sub reference database

Genome-wide Spearman correlation

```
qsig_sp_against_lincs_sub <- qSig(qsig = as.matrix(query), gess_method = "Cor", refdb = lincs_sub)
sp_against_lincs_sub <- gess_cor(qSig=qsig_sp_against_lincs_sub, method="spearman", chunk_size=5000)
result(sp_against_lincs_sub)
```

Spearman_sub correlatioin

```
# Subset z-scores of 150 up and down gene sets from "sirolimus__HEPG2__trt_cp" signature.
query_mat_sub <- as.matrix(query_mat[c(upset, downset),])
qsig_spsub_against_lincs_sub <- qSig(qsig = query_mat_sub, gess_method = "Cor", refdb = lincs_sub)
spsub_against_lincs_sub <- gess_cor(qSig=qsig_spsub_against_lincs_sub, method="spearman", chunk_size=500
result(spsub_against_lincs_sub)
```

Methods for FEA

Choose GESS result `lincs_against_lincs_sub` as input for the downstream functional enrichment analysis.

dup_hyperG method for TSEA

Subset top 100 ranking drugs in GESS result, get their target set (with duplication) as query for duplication support hypergeometric test for TSEA.

```
drugs <- unique(result(lincs_against_lincs_sub)$pert[1:100])
# GO annotation system
dup_hyperG_res <- tsea_dup_hyperG(drugs = drugs, universe = "Default", type = "GO", ont="MF", pvalueCut
dup_hyperG_res
result(dup_hyperG_res)
# KEGG annotation system
dup_hyperG_k_res <- tsea_dup_hyperG(drugs = drugs, universe = "Default", type = "KEGG", pvalueCutoff=0.
dup_hyperG_k_res
result(dup_hyperG_k_res)
```

m_GSEA method for TSEA

Subset top 100 ranking drugs in GESS result as query, m_GSEA method internally get their target set and turn it to scored ranked target list as query for TSEA. The scores represent weight of targets in the target set.

```
# 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, np

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

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.

```
# GO annotation system
mabs_res <- tsea_mabs(drugs=drugs, type="GO", ont="MF", nPerm=1000, pAdjustMethod="BH", pvalueCutoff=0.0
result(mabs_res)
# KEGG annotation system
```

```
mabs_k_res <- tsea_mabs(drugs=drugs, type="KEGG", nPerm=1000, pAdjustMethod="BH", pvalueCutoff=0.5, minG
result(mabs_k_res)
```

hyperG method for DSEA

Subset top 100 ranking drugs in GESS result as query for hypergeometric test for DSEA.

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

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.

```
dl <- abs(result(lincs_against_lincs_sub)$NCS); names(dl) <- result(lincs_against_lincs_sub)$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
gsea_res
result(gsea_res)
# KEGG annotation system
gsea_k_res <- dsea_GSEA(drugList=dl, type="KEGG", exponent=1, nPerm=1000, pAdjustMethod="BH", pvalueCut
gsea_k_res
result(gsea_k_res)
```

Visulization

Construct drug-target interaction networks in interesting GO categories

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

```
dtnetplot(drugs = hyperG_res@drugs, set = "GO:0043548", 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@drugs, set = "hsa05323")
```

Edgar, Ron, Michael Domrachev, and Alex E Lash. 2002. "Gene Expression Omnibus: NCBI Gene Expression and Hybridization Array Data Repository." *Nucleic Acids Res.* 30 (1): 207–10.

Lamb, Justin, Emily D Crawford, David Peck, Joshua W Modell, Irene C Blat, Matthew J Wrobel, Jim Lerner, et al. 2006. "The Connectivity Map: Using Gene-Expression Signatures to Connect Small Molecules,

Genes, and Disease." *Science* 313 (5795): 1929–35.

Peck, David, Emily D Crawford, Kenneth N Ross, Kimberly Stegmaier, Todd R Golub, and Justin Lamb. 2006. "A Method for High-Throughput Gene Expression Signature Analysis." *Genome Biol.* 7 (7): R61.