Package ‘sigQC’

April 9, 2021

Title Quality Control Metrics for Gene Signatures

Version 0.1.22

Description Provides gene signature quality control metrics in publication ready plots. Namely, enables the visualization of properties such as expression, variability, correlation, and comparison of methods of standardisation and scoring metrics.

Depends R (>= 3.3.0)

License file LICENSE

License_restricts_use yes

Encoding UTF-8

RoxygenNote 7.1.1

biocViews

Imports MASS, lattice, KernSmooth, cluster, nnet, class, gridGraphics, biclust, gplots, ComplexHeatmap, RankProd, fmsb, moments, grDevices, graphics, stats, utils, mclust, GSVA, circlize

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Andrew Dhawan [aut],
Alessandro Barberis [aut],
Wei-Chen Cheng [aut],
Francesca Buffa [aut, cre]

Maintainer Francesca Buffa <francesca.buffa@oncology.ox.ac.uk>

Repository CRAN

Date/Publication 2021-04-09 19:30:02 UTC

R topics documented:

make_all_plots *make_all_plots.R*

Description

Makes all the plots for the quality control of the list(s) of genes in a specified output directory (out_dir). Plots (PDFs) are made for all combinations of gene expression datasets and gene signatures inputted. For the purposes of this protocol, gene signatures are defined as sets of genes for which there is a coherent pattern of expression, in conjunction with a biological process or clinical outcome. The methodology is based on the sigQC protocol defined in the manuscript by Dhawan et al. at: https://www.biorxiv.org/content/early/2017/11/13/203729.

Usage

```
make_all_plots(
  gene_sigs_list,
  mRNA_expr_matrix,
  names_sigs = NULL,
  names_datasets = NULL,
  covariates = NULL,
  thresholds = NULL,
  out_dir = tempdir(),
  showResults = TRUE,
  origin = NULL,
  doNegativeControl = TRUE,
  numResampling = 50
)
```

Arguments

`gene_sigs_list` A list object, for the gene signatures. The name reference for each list element should correspond to the name of the gene signature. This list consists of k-by-1 character matrices of k gene names, which comprise the gene signature. Genes must be annotated in the same manner as the rows of the data matrix; at least one gene name must be present in the rownames of the gene expression matrices for the signature to be evaluated on that dataset.

`mRNA_expr_matrix` A list of matrices of expression values for the datasets to be considered, which must contain at least 2 samples per dataset. One numeric matrix entry per dataset. Name reference of each list entry should correspond to the name of the dataset. The rows are to be labelled as the genes, all annotated in the same way, and columns are sample IDs. Expression values should be normalised, batch-corrected, standardised, and log-transformed if needed, prior to use in sigQC. We recommend normalisation, batch correction, and log-transformation prior to use. Care must be taken to remove samples displaying a high proportion of NA values, especially for signature genes.

`names_sigs` The names of the gene signatures (e.g. Hypoxia, Invasiveness), one name per each signature in gene_sigs_list. Corresponds to the names of the entries of the list.

`names_datasets` The names of the different datasets contained in mRNA_expr_matrix. Corresponds to the names of the entries of the list.

`covariates` A list containing a sub-list of 'annotations' and 'colors' which contains the annotation matrix for the given dataset and the associated colours with which to plot in the expression heatmap. This is in the same form as used by the ComplexHeatmap package. One sub-list per dataset is used, referenced by the same name as given by the dataset in the mRNA_expr_matrix list.

`thresholds` A list of expression thresholds to be considered for each data set, default is median of the data set. A gene is considered expressed if above the threshold, non-expressed otherwise. One threshold per dataset, in the same order as the dataset list. Note that this is only used for the reporting of the genes showing supra-threshold expression across each dataset. Genes are not removed from computation based on expression; but proportion above this threshold is reported to the user. This is defaulted to the median level of all genes across all samples for a given dataset.

`out_dir` A path to the directory where the resulting output files are written. Default is R temporary directory, given by tempdir().

`showResults` Tells if R should open plot windows showing the computed results. Default is TRUE. Regardless of value, all plots are saved to PDF files in the output directory.

`origin` Tells if datasets have come from different labs/experiments/machines. This is a vector of characters, with same character representing same origin. Default is assumption that all datasets come from the same source. Used in the correction of batch effects during the RankProduct computation for poorly auto-correlated signature genes. Only to be used if multiple datasets are present.

`doNegativeControl`
Logical, tells the function if negative and permutation controls should be computed. TRUE by default. Note that depending on the number of resamplings, setting this parameter to TRUE may result in much longer runtimes. Negative controls in this context refers to resampling based on random genes selected with the same length as the gene signatures in question. Permutation controls are generated by considering the same genes as each signature in each dataset, but with labels of the genes permuted for each sample.

`numResampling` Number of bootstrap re-samplings of random gene signatures of the same length as the signature from which to compute null distribution of each metric, for each dataset and gene signature combination. This is the same value used for the nubmer of permutations of dataset values to consider in the permutation controls as described above, where in each dataset the labels of the signature genes are permuted for each sample. The default value for this parameter is set to 50.

Examples

```
library(sigQC)
names = c("dataset1")
```

```
data.matrix = replicate(10, rnorm(50))#random matrix - 50 genes x 10 samples
mRNA_expr_matrix = list()
mRNA_expr_matrix[["dataset1"]] = data.matrix
row.names(mRNA_expr_matrix$dataset1) <- as.character(1:(dim(mRNA_expr_matrix$dataset1)[1]))
colnames(mRNA_expr_matrix$dataset1) <- as.character(1:(dim(mRNA_expr_matrix$dataset1)[2]))
#Define the signature
gene_sigs_list = list()
signature = "hypoxiaSig"
gene_sig = c('1', '4', '5')#gene ids
gene_sigs_list[[signature]] = as.matrix(gene_sig)
names_sigs = c(signature)
make_all_plots(gene_sigs_list = gene_sigs_list, mRNA_expr_matrix = mRNA_expr_matrix,
    doNegativeControl=FALSE, out_dir = tempdir(), showResults=FALSE)
```

Index