Title       Single Cell Mapper
Version     0.1.1
Description  Statistical tools to interrogate the cell-type specificity of any gene list given a matrix of cell-types and genes associated with those cell-types (a signature matrix). Additionally, a library of over 245 signature matrices from public data are stored. Together, there are four primary functions: (i) processing scRNA-seq count data and automated cell-type naming using 'Seurat' V3 and enrichment of CellMarker and Panglao databases, (ii) tissue-by-cell-type gene set enrichment, (iii) cell-type specific enrichment of a gene list within a particular tissue, (iv) weighted cell-type specific reranking of a list of differentially expressed genes. Reference: Sokolowski DJ, Faykoo-Martinez M, Erdman L, Hou H, Uuskula-Reimand L, Yuki K, Zhu H, Goldenberg A, Wilson MD (In Preparation).

Depends     R (>= 3.6.0)
Imports     ggplot2, pheatmap, graphics, Seurat, GSVA, stats, utils, downloader, pcaMethods, grDevices, gProfileR, limSolve, gprofiler2,
License     GPL-3

URL
Encoding    UTF-8
LazyData    true
RoxygenNote  7.0.2
Suggests    testthat, knitr, rmarkdown
VignetteBuilder  knitr
NeedsCompilation no
Author       Dustin Sokolowski [aut, cre],
             Mariela Faykoo-Martinez [aut],
             Michael Wilson [aut],
             Anna Goldenberg [aut]
Maintainer   Dustin Sokolowski <dustin.sokolowski@sickkids.ca>
Repository   CRAN
Date/Publication  2020-03-05 23:10:02 UTC
R topics documented:

- cellmarker_enrich .................................................. 2
- coEnrich ............................................................. 4
- DecoRNAseq_CRAN .................................................. 5
- deconvolute_and_contextualize ................................... 6
- extract_genes_cell .................................................. 9
- generes_to_heatmap ............................................... 10
- get_gene_symbol ................................................... 11
- gmt ........................................................................... 12
- gProfiler_cellWeighted_Foldchange ............................... 13
- gsva_cellIdentify ..................................................... 14
- heatmap_generation .................................................. 15
- human_mouse_ct_marker_enrich ................................... 17
- make_TF_barplot ..................................................... 18
- pathway_enrich_internal .......................................... 19
- PBMC_example ....................................................... 21
- plotBP ................................................................. 22
- POA_example .......................................................... 23
- process dgTMatrix_lists .......................................... 24
- process_from_count ............................................... 25
- scMappR_and_pathway_analysis .................................. 27
- scMappR_tissues ...................................................... 29
- seurat_to_generes ................................................... 30
- single_gene_preferences .......................................... 31
- sm ............................................................................ 32
- tissue_by_celltype_enrichment ................................... 33
- tissue_scMappR_custom ............................................. 34
- tissue_scMappR_internal ........................................... 35
- tochr ................................................................. 37
- toNum ................................................................. 38
- topgenes_extract .................................................. 39

Index 40

---

**cellmarker_enrich**  
*Fisher’s Exact Cell-Type Identification.*

**Description**

This function uses the CellMarker and Panglao datasets to identify cell-type differentially expressed genes.
Usage

cellmarker_enrich(
  gene_list,
  p_thresh,
  gmt = "cellmarker_list.Rdata",
  fixed_length = 13000,
  min_genes = 5,
  max_genes = 3000,
  isect_size = 3
)

Arguments

gene_list A character vector of gene symbols with the same designation (e.g. mouse symbol - mouse, human symbol - human) as the gene set database.
p_thresh The Fisher’s test cutoff for a cell-marker to be enriched.
gmt Either a path to an rda file containing an object called "gmt", which is a named list where each element of the list is a vector of gene symbols website for more detail on the file type (https://software.broadinstitute.org/cancer/software/gsea/wiki/index.php/Data_form. The gmt list may also be inputted.
fixed_length Estimated number of genes in your background.
min_genes Minimum number of genes in the cell-type markers.
max_genes Maximum number of genes in the cell-type markers.
isect_size Number of genes in your list and the cell-type.

Details

Complete a Fisher’s exact test of an input list of genes against a gene set saved in an *.RData object. The RData object is storing a named list of genes called "gmt".

Value

cellmarker_enrich Gene set enrichment of cell-types on your inputted gene list.

Examples

data(POA_example)
POA_genes <- POA_example$POA_genes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- POA_Rank_signature
rowname <- get_gene_symbol(Signature)
rownames(Signature) <- rowname$rownname
genes <- rownames(Signature)[1:100]
data(gmt)
coEnrich <- cellmarker_enrich(gene_list = genes, p_thresh = 0.05, gmt = gmt)

---

**coEnrich**

*Identify co-expressed cell-types*

**Description**

This function identifies genes with similar cell-type markers and if those markers are driving enrichment.

**Usage**

```r
coEnrich(
  sig,
  gene_list_heatmap,
  background_heatmap,
  study_name,
  outDir,
  toSave = FALSE,
  path = NULL
)
```

**Arguments**

- `sig` A The number of combinations of significant cell-types to enrich.
- `gene_list_heatmap` Signature matrix of inputted genes in heatmap and the cell-type preferences – output of heatmap generation.
- `background_heatmap` Signature matrix of background matrix in heatmap and cell-type preferences – output of heatmap generation.
- `study_name` Name of the outputted table.
- `outDir` Name of the directory this table will be printed in.
- `toSave` Allow scMappR to write files in the current directory (T/F).
- `path` If toSave == TRUE, path to the directory where files will be saved.

**Details**

This function takes significantly enriched cell-types from the single CT_enrich before testing to see if the genes driving their enrichment are overlapping to a significant proportion using Fisher’s exact test. To save computational time and to not complete this with an incredible number of permutations, scMappR stops at overlapping 5 cell-types.
Value

coenrich Enrichment of cell-types that are expressed by the same genes, up to 4 sets of cell-types.

Examples

```r
# load in signature matrices
data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
sig <- get_gene_symbol(POA_Rank_signature)
Signature <- POA_Rank_signature
rownames(Signature) <- sig$rowname
genes <- rownames(Signature)[1:60]
heatmap_test <- tissue_scMappR_custom( genes, signature_matrix = Signature, output_directory = "scMappR_test", toSave = FALSE)
group_preferences <- heatmap_test$group_celltype_preferences
```

DeconRNAseq_CRAN

DeconRNASeq CRAN compatible

Description

This function runs DeconRNAseq with default parameters such that it is compatible with CRAN and scMappR

Usage

```r
DeconRNAseq_CRAN(
datasets,
signatures,
proportions = NULL,
checksig = FALSE,
known.prop = FALSE,
use.scale = TRUE,
fig = FALSE
)
```

Arguments

datasets Normalized RNA-seq dataset

signatures Signature matrix of odds ratios

proportions If cell-type proportion is already inputted - always NULL for scMappR

checksig Check to see if plotting is significant - always false for scMappR
known.prop If proportions were known - always false for scMappR
use.scale Scale and center value - always TRUE for scMappR
fig Make figures - always FALSE for scMappR

Details
This is the exact same function as the primary function in the Bioconductor package, DeconRNAseq (PMID: 23428642) except it is now compatible with CRAN packages.

Value
DeconRNAseq_CRAN Estimated cell-type proportions with DeconRNAseq.

Examples

data(PBMC_example)
bulk_DE_cors <- PBMC_example$bulk_DE_cors
bulk_normalized <- PBMC_example$bulk_normalized
odds_ratio_in <- PBMC_example$odds_ratio_in
out <- DeconRNAseq_CRAN(as.data.frame(bulk_normalized), as.data.frame(odds_ratio_in))

deconvolute_and_contextualize
Generate cell weighted Fold-Changes (cwFold-changes)

Description
This function takes a count matrix, signature matrix, and differentially expressed genes (DEGs) before generating cwFold-changes for each cell-type.

Usage

deconvolute_and_contextualize(
  count_file,
  signature_matrix,
  DEG_list,
  case_grep,
  control_grep,
  max_proportion_change = -9,
  print_plots = T,
  plot_names = "scMappR",
  theSpecies = "human",
  make_scale = FALSE,
  FC_coef = T,
```r
sig_matrix_size = 3000,
sig_distort = 1,
drop_unknown_celltype = TRUE,
toSave = FALSE,
path = NULL
)
```

**Arguments**

- `count_file` Normalized deconvolute_and_contextualize. RNA-seq count matrix where rows are gene symbols and columns are individuals. Either the object itself or the path of a .tsv file.

- `signature_matrix` Signature matrix (recommended odds ratios) of cell-type specificity of genes. Either the object itself or a pathway to a .RData file containing an object named “wilcoxon_rank_mat_or” - generally internal.

- `DEG_list` An object with the first column as gene symbols within the bulk dataset (doesn’t have to be in signature matrix), second column is the adjusted P-value, and the third the log2FC. Path to a tsv file containing this info is also acceptable.

- `case_grep` Tag in the column name for cases (i.e. samples representing upregulated) OR an index of cases.

- `control_grep` Tag in the column name for control (i.e. samples representing downregulated) OR an index of cases.

- `max_proportion_change` Maximum cell-type proportion change. May be useful if there are many rare cell-types.

- `print_plots` Whether boxplots of the estimated CT proportion for the leave-one-out method of CT deconvolution should be printed (T/F).

- `plot_names` If plots are being printed, the pre-fix of their .pdf files.

- `theSpecies` -9 if using a precomputed count matrix from scMappR, human otherwise. Removes ensmbll symbols if appended.

- `make_scale` Convert the lowest odds ratio to 1 and scales accordingly – strongly not recommended and will produce warning if used.

- `FC_coef` Making cwFold-changes based on fold-change (TRUE) or rank := (-log10(Pval)) (FALSE) rank. After testing, we strongly recommend to keep true (T/F).

- `sig_matrix_size` Number of genes in signature matrix for cell-type deconvolution.

- `sig_distort` Exponential change of odds ratios. Strongly not recomended and will produce warnings if changed from default.

- `drop_unknown_celltype` Whether or not to remove "unknown" cell-types from the signature matrix (T/F).

- `toSave` Allow scMappR to write files in the current directory (T/F).

- `path` If toSave == TRUE, path to the directory where files will be saved.
**Details**

This function completes the cell-type contextualization in scMappR – reranking every DEG based on their fold change, likelihood the gene is in each detected cell-type, average cell-type proportion, and ratio of cell-type proportion between case and control. If a gene is upregulated then it is being controlled by control/case, otherwise it is case/control. cwFold-change’s are generated for genes that are in both the count matrix and in the list of DEGs. It does not have to also be in the signature matrix. First, this function will estimate cell-type proportions with all genes included before estimating changes in cell-type proportion between case/control using a t-test. Then, it takes a leave-one-out approach to cell-type deconvolution such that estimated cell-type proportions are computed for every inputted DEG. Optionally, the differences between cell-type proportion before and after a gene is removed is plotted in boxplots. Then, for every gene, cwFold-change’s are computed with the following formula (the example for upregulated genes) val <- cell-preferences * cell-type_proportion * cell-type_proportion_fold-change * sign*2^abs(gene_DE$log2fc). A matrix of cwFold-change’s for all DEGs are returned.

**Value**

List with the following elements:

- `cellWeighted_Foldchange` data frame of cellweightedFold changes for each gene.
- `cellType_Proportions` data frame of cell-type proportions from DeconRNA-seq.
- `leave_one_out_proportions` data frame of average cell-type proportions for case and control when gene is removed.
- `processed_signature_matrix` signature matrix used in final analysis.

**Examples**

```r
data(PBMC_example)
bulk_DE_cors <- PBMC_example$bulk_DE_cors
bulk_normalized <- PBMC_example$bulk_normalized
odds_ratio_in <- PBMC_example$odds_ratio_in
case_grep <- "_female"
control_grep <- "_male"
max_proportion_change <- 10
print_plots <- FALSE
theSpecies <- "human"
norm <- deconvolute_and_contextualize(bulk_normalized, odds_ratio_in, bulk_DE_cors,
case_grep = case_grep, control_grep = control_grep,
max_proportion_change = max_proportion_change,
print_plots = print_plots,
theSpecies = theSpecies, toSave = FALSE)
```
extract_genes_cell

Extract Markers

Description

Extracting cell-type markers from a signature matrix.

Usage

```
extract_genes_cell(
  geneHeat,
  cellTypes = "ALL",
  val = 1,
  isMax = FALSE,
  isPvalue = FALSE
)
```

Arguments

- `geneHeat`: The heatmap of ranks from your scRNA-seq dataset with your genes subsetted.
- `cellTypes`: The cell-types that you’re interested in extracting. They need to be colnames (not case sensitive).
- `val`: How associated a gene is with a particular cell type to include in your list - default is slightly associated.
- `isMax`: If you are taking the single best CT marker (T/F) – TRUE not recommended.
- `isPvalue`: If the signature matrix is raw p-value (T/F) – TRUE not recommended.

Details

This function takes a signature matrix and extracts cell-type markers above a p-value or fold-change threshold.

Value

- `extract_genes_cell`: A vector of genes above the threshold for each sample.

Examples

```
data(POA_example)
Signature <- POA_example$POA_Rank_signature
RowName <- get_gene_symbol(Signature)
rownames(Signature) <- RowName$rowname
# extract genes with a -log10(Padj > 1)
Signat <- extract_genes_cell(Signature)
```
**generes_to_heatmap**  
*Generate signature matrix*

**Description**
Convert a list of cell-type markers from FindMarkers in Seurat to a signature matrix defined by odds ratio and rank.

**Usage**

```r
generes_to_heatmap(
  generes,
  species = "human",
  naming_preference = -9,
  rda_path = "",
  make_names = TRUE,
  internal = FALSE
)
```

**Arguments**

- `generes`: A list of cell-type markers with fold-changes and p-values (FindMarkers output in Seurat).
- `species`: The species of gene symbols, if not internal, "human" or "mouse".
- `naming_preference`: Likely cell-types given tissues (to be passed into human_mouse_ct_marker_enrich).
- `rda_path`: Path to output directory, if toSave is true.
- `make_names`: Identify names of cell-type markers using the Fisher’s exact test method (T/F).
- `internal`: If this function is pre-processing from Panglao (T/F).

**Details**
Take a list of compiled differentially expressed genes from different cell-types, identify what the cell-types are using the Fisher’s exact test, and then convert into a signature matrix for both the adjusted p-value and odds ratio.

**Value**
List with the following elements:

- `pVal`: A dataframe containing the signature matrix of ranks (-log10(Padj) * sign(fold-change)).
**get_gene_symbol**

OR

A dataframe containing the signature matrix of odds ratios.

cellname

A vector of the cell-labels returned from the GSVA method.

topGenes

the top 30 mos expressed genes in each cell-type.

**Examples**

```r
data(POA_example)
POA_generes <- POA_example$POA_generes
signature <- generes_to_heatmap(POA_generes,species = -9, make_names = FALSE)
```

---

**Description**

Internal – get gene symbol from Panglao.db matrix.

**Usage**

```r
get_gene_symbol(wilcoxon_rank_mat_t)
```

**Arguments**

- **wilcoxon_rank_mat_t**
  Matrix where row names are "GeneSymbol-Ensembl" (human or mouse).

**Details**

Internal: This function removes the ENGMUS/ENGS tag from Panglao created gene names (symbol-ENGS). From the ENSG/ENSMUS, this function determines if the species is mouse/human and returns the gene symbols.

**Value**

List with the following elements:

- **rownames**
  Genes in the signature matrix excluding the ensemble name.

- **species**
  "mouse" or "human" depending on appended ensembl symbols.
Examples

```r
# load signature
data(POA_example)
POA_OR_signature <- POA_example$POA_OR_signature
symbols <- get_gene_symbol(POA_OR_signature)
```

Description

Markers of 5 glial cell-types

Usage

```r
data(gmt)
```

Format

A list with 5 character vectors, each containing genes.

- **Astrocytes_panglao**  astrocyte markers identified by panglao
- **Schwann_panglao**  Schwann markers identified by panglao
- **Bergmann glia_panglao**  Bergmann glia markers identified by panglao
- **Kupffer_panglao**  Kupffer markers identified by panglao
- **Oligodendrocyte progenitor_panglao**  Oligodendrocyte progenitor markers identified by panglao

Details

A named list containing the cell-type markers of 5 glial cell types. Used for testing cell-type naming functions.

Examples

```r
data(gmt)
```
Pathway enrichment for cwFold-changes

Description

This function runs through each list of cell weighted Fold changes (cwFold-changes) and completes both pathway and transcription factor (TF) enrichment.

Usage

gProfiler_cellWeighted_Foldchange(
  cellWeighted_Foldchange_matrix,
  species,
  background,
  gene_cut,
  newGprofiler
)

Arguments

cellWeighted_Foldchange_matrix
  Matrix of cell weighted Fold changes from the deconvolute_and_contextualize functions.

species
  Human, mouse, or a character that is compatible with gProfileR.

background
  A list of background genes to test against.

gene_cut
  The top number of genes in pathway analysis.

newGprofiler
  Using gProfileR or gprofiler2, (T/F).

Details

This function takes a matrix of cellWeighted_Foldchange and a species (human, mouse, or a character directly compatible with gProfileR). Before completing pathway analysis with gProfileR. Enriched pathways are stored in a list and returned.

Value

List with the following elements:

BP   gprofiler enrichment of biological pathways for each cell-type
TF   gprofiler enrichment of transcription factors for each cell-type.
Examples

```r
data(PBMC_example)

bulk_DE_cors <- PBMC_example$bulk_DE_cors
bulk_normalized <- PBMC_example$bulk_normalized
odds_ratio_in <- PBMC_example$odds_ratio_in

case_grep <- "_female"
control_grep <- "_male"
max_proportion_change <- 10
print_plots <- FALSE
theSpecies <- "human"

norm <- deconvolute_and_contextualize(bulk_normalized, odds_ratio_in,
  bulk_DE_cors, case_grep = case_grep,
  control_grep = control_grep,
  max_proportion_change = max_proportion_change,
  print_plots = print_plots,
  theSpecies = theSpecies)

background <- rownames(bulk_normalized)
STVs <- gProfiler_cellWeighted_Foldchange(norm$cellWeighted_Foldchange, theSpecies,
  background, gene_cut = -9, newGprofiler = FALSE)
```

### gsva_cellIdentify

**Cell-type naming with GSVA**

### Description

This function computes the mean expression of every cell-type before predicting the most likely cell-type using the GSVA method.

### Usage

```r
gsva_cellIdentify(
  pbmc,
  theSpecies,
  naming_preference = -9,
  rda_path = "",
  toSave = FALSE
)
```

### Arguments

- **pbmc**
  - Processed seurat object without named cells.
heatmap_generation

theSpecies
"human" or "mouse" – it will determine which CT marker database to use – there are some differences.
naming_preference
Once top CT markers are identified, naming_preferences will then extract CT markers within a more appropriate tissue type.
rda_path
Path to pre-computed cell-type .gmt files (rda objects).
toSave
If scMappR is allowed to write files and directories.

Details

This function inputs a Seurat object and uses the average normalized expression of each gene in each cluster to identify cell-types using the GSVA method.

Value

List with the following elements:

cellMarker
Most likely cell-types predicted from cellMarker database.
panglao
Most likely cell-types predicted from panglao database.
avg_expression
Average expression of each gene in each cell-type.

Examples

data(sm)
toProcess <- list(example = sm)
tst1 <- process_from_count(toProcess, "testProcess")
cellnames <- gsva_cellIdentify(tst1, theSpecies = "mouse",
naming_preference = "brain", rda_path = "")

heatmap_generation

Generate Heatmap

Description

This function takes an inputted signature matrix as well as a list of genes and overlaps them. Then, if there is overlap, it prints a heatmap or barplot (depending on the number of overlapping genes). Then, for every cell-type, genes considered over-represented are saved in a list.
Usage

heatmap_generation(
  genesIn,
  comp,
  reference,
  cex = 0.8,
  rd_path = "",
  cellTypes = "ALL",
  pVal = 0.01,
  isPval = TRUE,
  isMax = FALSE,
  isBackground = FALSE,
  which_species = "human",
  toSave = FALSE,
  path = NULL
)

Arguments

genesIn A list of gene symbols (all caps) to have their cell type enrichment.
comp The name of the comparison.
reference Path to signature matrix or the signature matrix itself.
cex The size of the genes in the column label for the heatmap.
rd_path The directory to RData files – if they are not in this directory, then the files will be downloaded.
cellTypes Colnames of the cell-types you will extract (passed to extract_genes_cell).
pVal The level of association a gene has within a cell type (passed to extract_genes_cell).
isPval If the signature matrix is raw p-value (T/F) – TRUE not recomended.
isMax If you are taking the single best CT marker (T/F) – TRUE not recomended.
isBackground If the heatmap is from the entire signature matrix or just the inputted gene list (T/F). isBackground == TRUE is used for internal.
which_species Species of gene symbols – "human" or "mouse".
toSave Allow scMappR to write files in the path directory (T/F).
path If toSave == TRUE, path to the directory where files will be saved.

Value

List with the following elements:

genesIn Vector of genes intersecting gene list and signature matrix.
genesNoIn Vector of inputted genes not in signature matrix.
geneHeat Signature matrix subsetted by inputted gene list
preferences Cell-markers mapping to cell-types.
# load in signature matrices
data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- POA_Rank_signature
rowname <- get_gene_symbol(Signature)
rownames(Signature) <- rowname$rowname
genes <- rownames(Signature)[1:100]
heatmap_test <- heatmap_generation(genesIn = genes, "scMappR_test",
                               reference = Signature, which_species = "mouse")

human_mouse_ct_marker_enrich

Consensus cell-type naming (Fisher's Exact)

Description

This function completes the Fisher's exact test cell-type naming for all cell-types.

Usage

human_mouse_ct_marker_enrich(
  gene_lists,
  theSpecies = "human",
  cell_marker_path = "",
  naming_preference = -9
)

Arguments

gene_lists A named list of vectors containing cell-type markers (mouse or human gene-symbols) which will be named as a cell-type via the Fisher's exact test method.

theSpecies The species of the gene symbols: "human" or "mouse".

cell_marker_path If local, path to cell-type marker rda files, otherwise, we will try to download data files.

naming_preference Either -9 if there is no expected cell-type or one of the categories from get_naming_preference_options(). This is useful if you previously have an idea of which cell-type you were going to enrich.
Details

Fisher’s exact test method of cell-type identification using the Panglao and CellMarker databases. It extracts significant pathways (pFDR < 0.05). Then, if naming_preference != -9, it will extract the enriched cell-types within the cell-types identified with the naming preferences option. Generally, this method seems to be biased to cell-types with a greater number of markers.

Value

List with the following elements:

- **cellTypes**: most likely marker for each cell-type from each database.
- **marker_sets**: all enriched cell-types for each cluster from each dataset.

Examples

```r
data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- POA_Rank_signature
rownames(Signature) <- rowname$rownames
genes <- rownames(Signature)[1:100]
enriched <- human_mouse_ct_marker_enrich(gene_lists = genes, theSpecies = "mouse",
                                        cell_marker_path = "", naming_preference = "brain")
```

---

**make_TF_barplot**

Plot gprofileR Barplot (TF)

Description

Make a barplot of the top transcription factors enriched by gprofileR.

Usage

```r
make_TF_barplot(ordered_back_all_tf, top_tf = 5)
```

Arguments

- **ordered_back_all_tf**: Output of the gprofileR function.
- **top_tf**: The number of pathways you want to plot.
Details

This function takes a gprofileR output and prints the top "top_tfs" most significantly enriched p-values before plotting the rank of their p-values.

Value

make_TF_barplot A barplot of the number of "top_tf" tf names (not motifs), ranked by -log10(Pfdr).

Examples

data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- as.data.frame(POA_Rank_signature)
rownames(Signature) <- get_gene_symbol(Signature)
rownames(Signature) <- rowname$rowname
ordered_back_all <- gprofiler2::gost(query = rowname$rowname[1:100], organism = "mmusculus",
ordered_query = TRUE, significant = TRUE, exclude_iae = FALSE, multi_query = FALSE,
measure_underrepresentation = FALSE, evcodes = FALSE, user_threshold = 0.05,
correction_method = "fdr", numeric_ns = ",", sources = c("GO:BP", "KEGG", "REAC"))
ordered_back_all <- ordered_back_all$result
ordered_back_all <- ordered_back_all[ordered_back_all$term_size > 15 &
ordered_back_all$term_size < 2000 & ordered_back_all$intersection_size > 2,]
ordered_back_all_tf <- gprofiler2::gost(query = rowname$rowname[1:150], organism = "mmusculus",
ordered_query = TRUE, significant = TRUE, exclude_iae = FALSE, multi_query = FALSE,
measure_underrepresentation = FALSE, evcodes = FALSE, user_threshold = 0.05,
correction_method = "fdr", numeric_ns = ",", sources = c("TF"))
ordered_back_all_tf <- ordered_back_all_tf$result
ordered_back_all_tf <- ordered_back_all_tf[ordered_back_all_tf$term_size > 15
& ordered_back_all_tf$term_size < 5000 & ordered_back_all_tf$intersection_size > 2,]
TF <- ordered_back_all_tf
BP <- ordered_back_all
bp <- plotBP(BP)
tf <- make_TF_barplot(TF)
Usage

```r
pathway_enrich_internal(
  DEGs,
  theSpecies,
  scMappR_vals,
  background_genes,
  output_directory,
  plot_names,
  number_genes = -9,
  newGprofiler = FALSE,
  toSave = FALSE,
  path = NULL
)
```

Arguments

- **DEGs**: Differentially expressed genes (gene_name, padj, log2fc).
- **theSpecies**: Human, mouse, or a character that is compatible with gProfileR.
- **scMappR_vals**: cell weighted Fold-changes of differentially expressed genes.
- **background_genes**: A list of background genes to test against.
- **output_directory**: Path to the directory where files will be saved.
- **plot_names**: Names of output.
- **number_genes**: Number of genes to if there are many, many DEGs.
- **newGprofiler**: Whether to use gProfileR or gprofiler2 (T/F).
- **toSave**: Allow scMappR to write files in the current directory (T/F).
- **path**: If toSave == TRUE, path to the directory where files will be saved.

Details

Internal: Pathway analysis of differentially expressed genes (DEGs) and cell weighted Fold-changes (cellWeighted_Foldchanges) for each cell-type. Returns .RData objects of differential analysis as well as plots of the top bulk pathways. It is a wrapper for making barplots, bulk pathway analysis, and gProfiler_cellWeighted_Foldchange.

Value

List with the following elements:

- **BPs**: Enriched biological pathways for each cell-type.
- **TFs**: Enriched transcription factors for each cell-type.
Examples

```r
data(PBMC_example)
bulk_DE_cors <- PBMC_example$bulk_DE_cors
bulk_normalized <- PBMC_example$bulk_normalized
odds_ratio_in <- PBMC_example$odds_ratio_in
case_grep <- "_female"
control_grep <- "_male"
max_proportion_change <- 10
print_plots <- FALSE
theSpecies <- "human"
toOut <- scMappR_and_pathway_analysis(bulk_normalized, odds_ratio_in,
bulk_DE_cors, case_grep = case_grep,
control_grep = control_grep, rda_path = "",
max_proportion_change = 10, print_plots = TRUE,
plot_names = "tst1", theSpecies = "human",
output_directory = "tester",
sig_matrix_size = 3000, up_and_downregulated = FALSE,
internet = FALSE)
```

### Description

Data for scMappR_and_pathway_analysis example.

### Usage

```r
data(PBMC_example)
```

### Format

A list containing three data frames, normalized count data, a signature matrix, and a list of differentially expressed genes.

- **bulk_normalized** A 3231 x 9 matrix where rows are genes, columns are samples, and the matrix is filled with CPM normalized counts.

- **odds_ratio_in** A 2336 x 7 matrix where rows are genes, columns are cell-types and matrix is filled with the odds-ratio that a gene is in each cell-type.

- **bulk_DE_cors** A 59 x 3 matrix of sex-specific genes found between male and female PBMC samples (female biased = upregulated). row and rownames are genes, columns are gene name, FDR adjusted p-value, and log2 fold-change. DEGs were computed with DESeq2 and genes with a log2FC > 1 were kept.
Details

A named list called "PBMC_example" containing the count data, signature matrix, and DEGs. The count data and signature matrix are shortened to fit the size of the package and do not reflect the paper.

Examples

data(PBMC_example)

plotBP

Plot gProfileR Barplot

Description

Make a barplot of the top biological factors enriched by gProfileR.

Usage

plotBP(ordered_back_all, top_bp = 10)

Arguments

ordered_back_all
  Output of the gProfileR function.

top_bp
  The number of pathways you want to plot.

Details

This function takes a gProfileR output and prints the top "top_bp" most significantly enriched p-values before plotting the rank of their p-values.

Value

plotBP A barplot of the number of "top_bp" pathways, ranked by -log10(PFDR).

Examples

data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- as.data.frame(POA_Rank_signature)
rowname <- get_gene_symbol(Signature)
rownames(Signature) <- rowname$rownames
ordered_back_all <- gprofiler2::gost(query = rowname$rownames[1:100], organism = "mmusculus",
ordered_query = TRUE, significant = TRUE, exclude_ia = FALSE, multi_query = FALSE,
```r
measure_underrepresentation = FALSE, evcodes = FALSE, user_threshold = 0.05,
correction_method = "fdr", numeric_ns = "", sources = c("GO:BP", "KEGG", "REAC"))
ordered_back_all <- ordered_back_all$result
ordered_back_all <- ordered_back_all[result, ordered_back_all$term_size > 15
& ordered_back_all$term_size < 2000 & ordered_back_all$intersection_size > 2,
ordered_back_all_tf <- gprofiler2::gost(query = rowname[1:150], organism = "mmusculus",
measure_underrepresentation = FALSE, evcodes = FALSE, user_threshold = 0.05,
correction_method = "fdr", numeric_ns = "", sources = c("TF"))
ordered_back_all_tf <- ordered_back_all_tf$result
ordered_back_all_tf <- ordered_back_all_tf[ordered_back_all_tf$term_size > 15
& ordered_back_all_tf$term_size < 5000 & ordered_back_all_tf$intersection_size > 2,
TF = ordered_back_all_tf
BP <- plotBP(BP)
bp <- make_TF_barplot(TF)
```
process_dgTMatrix_lists

Count Matrix To Signature Matrix

Description

This function takes a list of count matrices, processes them, calls cell-types, and generates signature matrices.

Usage

```r
process_dgTMatrix_lists(
  dgTMatrix_list,
  name,
  species_name,
  naming_preference = -9,
  rda_path = "",
  panglao_set = FALSE,
  haveUMAP = FALSE,
  saveSCObject = FALSE,
  internal = FALSE,
  toSave = FALSE,
  path = NULL,
  use_sctransform = FALSE,
  test_ctname = "wilcox"
)
```

Arguments

dgTMatrix_list  A list of matrices in the class of dgTMatrix object – sparse object – compatible with Seurat rownames should be of the same species for each.

name  The name of the outputted signature matrices, cell-type preferences, and Seurat objects if you choose to save them.

species_name  Mouse or human symbols, -9 if internal as panglao objects have gene symbol and ensembl strappled together.

naming_preference  For cell-type naming, see if cell-types given the inputted tissues are more likely to be named within one of the categories of get_naming_preference_options().

rda_path  If saved, directory to where data from scMappR_data is downloaded.

panglao_set  If the inputted matrices are from Panglao (i.e. if they’re internal).

haveUMAP  Save the UMAPs – only use if the package is downloaded with pip.

saveSCObject  Save the Seurat object as an RData object (T/F).
process_from_count

Was this used as part of the internal processing of Panglao datasets (T/F).

toSave

Allow scMappR to write files in the current directory (T/F).

path

If toSave == TRUE, path to the directory where files will be saved.

use_sctransform

If you should use sctransform or the Normalize/VariableFeatures/ScaleData pipeline (T/F).

test_ctname

statistical test for calling CT markers – must be in Seurat

Details

This function is a one line wrapper to process count matrices into a signature matrix. It combines process_from_count, two methods of identifying cell-type identities (GSVA and Fisher’s test). Then, it takes the output of cell-type markers and converts it into a signature matrix of p-value ranks and odds ratios. Along the way, it saves the Seurat object (if chosen with saveSCObject), cell-type identities from GSVA (its own object), and the signature matrices. Cell-type marker outputs are also saved in the generes.RData list. Names of the generes objects and the signature matrices are kept.

Value

List with the following elements:

wilcoxon_rank_mat_t

A dataframe containing the signature matrix of ranks (-log10(Padj) * sign(fold-change)).

wilcoxon_rank_mat_or

A dataframe containing the signature matrix of odds-ratios.

generes

All cell-type markers for each cell-type with p-value and fold changes.

Examples

data(sm)
toProcess <- list(example = sm)
tst1 <- process_dgTMatrix_lists(toProcess, name = "testProcess", species_name = -9,
naming_preference = "eye", rda_path = "")

process_from_count

Count Matrix To Seurat Object

Description

This function processes a list of count matrices (same species/gene symbols in each list) and converts them to a Seurat object.
process_from_count

Usage

process_from_count(
  countmat_list,
  name,
  theSpecies = -9,
  haveUmap = FALSE,
  saveALL = FALSE,
  panglao_set = FALSE,
  toSave = FALSE,
  path = NULL,
  use_sctransform = FALSE
)

Arguments

countmat_list A list of count matrices that will be integrated using the IntegrationAnchors features they should have the same rownames.

name The output of the normalized and fused Seurat object if you choose to keep it.

theSpecies Gene symbols for human, mouse, or -9 if internal. If your species is not human or mouse gene symbols, make sure that you have "MT-" before your mitochondrial gene names then pick "human".

haveUmap Write a UMAP (T/F).

saveALL Save the Seurat object generated (T/F).

panglao_set If the function is being used from internal (T/F).

toSave Allows scMappR to print files and make directories locally (T/F).

path If toSave == TRUE, path to the directory where files will be saved.

use_sctransform If you should use sctransform or the Normalize/VariableFeatures/ScaleData pipeline (T/F).

Details

This function takes a list of count matrices and returns a Seurat object of the count matrices integrated using Seurat v3 (with sctransform and IntegrationAnchors). Different options are used when the function is being ran internally (i.e. reprocessing count matrices from PanglaoDB) or if it is running from custom scRNA-seq data. For larger scRNA-seq datasets (~20k + cells), it is likely that this function will be required to run on an hpc.

Value

process_from_count A processed and integrated Seurat object that has been scaled and clustered. It can be returned as an internal object or also stored as an RData object if necessary.
scMappR_and_pathway_analysis

Examples

```r
data(sm)
toProcess <- list(example = sm)
tst1 <- process_from_count(toProcess, "testProcess")
```

scMappR_and_pathway_analysis

*Generate cellWeighted_Foldchange, visualize, and enrich.*

Description

This function generates cell weighted Fold-changes (cellWeighted_Foldchange), visualizes them in a heatmap, and completes pathway enrichment of cellWeighted_Foldchanges and bulk gene list.

Usage

```r
scMappR_and_pathway_analysis(
  count_file,
  signature_matrix,
  DEG_list,
  case_grep,
  control_grep,
  rda_path = "",
  max_proportion_change = -9,
  print_plots = T,
  plot_names = "scMappR",
  theSpecies = "human",
  output_directory = "scMappR_analysis",
  sig_matrix_size = 3000,
  drop_unknown_celltype = TRUE,
  internet = TRUE,
  up_and_downregulated = FALSE,
  gene_label_size = 0.4,
  number_genes = -9,
  toSave = FALSE,
  newGprofiler = FALSE,
  path = NULL
)
```

Arguments

- `count_file`: Normalized RNA-seq count matrix where rows are gene symbols and columns are individuals. Either the object itself or the path of a .tsv file.
signature_matrix
Signature matrix (recommended odds ratios) of cell-type specificity of genes. Either the object itself or a pathway to a .RData file containing an object named "wilcoxon_rank_mat_or" – generally internal.

DEG_list
An object with the first column as gene symbols within the bulk dataset (doesn’t have to be in signature matrix), second column is the adjusted p-value, and the third the log2FC path to a .tsv file containing this info is also acceptable.

case_grep
Tag in the column name for cases (i.e. samples representing upregulated) OR an index of cases.

case_grep
Tag in the column name for controls (i.e. samples representing downregulated OR an index of controls).

rda_path
If downloaded, path to where data from scMappR_data is stored.

max_proportion_change
Maximum cell-type proportion change – may be useful if there are many rare cell-types.

print_plots
Whether boxplots of the estimated CT proportion for the leave-one-out method of CT deconvolution should be printed. The same name of the plots will be completed for top pathways.

plot_names
The prefix of plot pdf files.

theSpecies
-9 if using a pre-computed count matrix from scMappR, human, mouse, or a specied directly compatible with gProfileR. Removes Ensembl symbols if appended.

output_directory
The name of the directory that will contain output of the analysis.

sig_matrix_size
Number of genes in signature matrix for cell-type deconvolution.

drop_unknown_celltype
Whether or not to remove "unknown" cell-types from the signature matrix.

internet
Whether you have stable Wifi (T/F).

up_and_downregulated
Whether you are additionally splitting up/downregulated genes (T/F).

gene_label_size
The size of the gene label on the plot.

number_genes
The number of genes to cut-off for pathway analysis (good with many DEGs).

toSave
Allow scMappR to write files in the current directory (T/F).

newGprofiler
Whether to use gProfileR or gprofiler2 (T/F).

path
If toSave == TRUE, path to the directory where files will be saved.

Details
This function generates cellWeighted_Foldchanges for every cell-type (see deconvolute_and_contextualize), as well as the relative cell-type proportions (which will be reutnred and pushed through). Then, it generates heatmaps of all cellWeighted_Foldchanges, cellWeighted_Foldchanges overlapping with the signature matrix, the entire signature matrix, the cell-type preference values from the signature
matrix that overlap with inputted differentially expressed genes. Then, if you have Wifi, it will complete gProfileR of the reordered cellWeighted_Foldchanges as well as a the ordered list of genes. This function is a wrapper for deconvolute_and_contextualize and pathway_enrich_internal.

Value

List with the following elements:

- cellWeighted_Foldchanges: Cellweighted Fold-changes for all differentially expressed genes.
- paths: Enriched biological pathways for each cell-type.
- TFs: Enirched TFs for each cell-type.

Examples

data(PBMC_example)
bulk_DE_cors <- PBMC_example$bulk_DE_cors
bulk_normalized <- PBMC_example$bulk_normalized
odds_ratio_in <- PBMC_example$odds_ratio_in
case_grep <- "_female"
control_grep <- "_male"
max_proportion_change <- 10
print_plots <- FALSE
theSpecies <- "human"
toOut <- scMappR_and_pathway_analysis(bulk_normalized, odds_ratio_in, 
  bulk_DE_cors, case_grep = case_grep, 
  control_grep = control_grep, rda_path = "", 
  max_proportion_change = 10, print_plots = TRUE, 
  plot_names = "tst1", theSpecies = "human", 
  output_directory = "tester", 
  sig_matrix_size = 3000, up_and_downregulated = FALSE, 
  internet = FALSE)
seurat_to_generes

Format

A vector of tissue names available for tissue_scMappR_internal or to download and use in scMappR_and_pathway_analysis.

scMappR_tissues

A list of 174 tissue names from PanglaoDB.

Details

A vector of tissues available in scMappR as of February 2020.

Examples

data(scMappR_tissues)

seurat_to_generes

Identify all cell-type markers

Description

Takes processed Seurat matrix and identifies cell-type markers with FindMarkers.

Usage

seurat_to_generes(pbmc, test = "wilcox")

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pbmc</td>
<td>Processed Seurat object.</td>
</tr>
<tr>
<td>test</td>
<td>statistical test for calling CT markers – must be in Seurat</td>
</tr>
</tbody>
</table>

Details

Internal: This function runs the FindMarkers function from seurat in a loop, will use the Seurat v2 or Seurat v3 object after identifying which Seurat object is inputted. It then takes the output of the FindMarkers and puts it in a list, returning it.

Value

seurat_to_generes A list of genes where their over-representation in the i’th cell-type is computed. Each element contains the gene name, adjusted p-value, and the log2Fold-Change of each gene being present in that cell-type.
**single_gene_preferences**

**Examples**

```r
data(sm)
toProcess <- list(example = sm)
tst1 <- process_from_count(toProcess, "testProcess")
generes <- seurat_to_generes(tst1)
```

---

**single_gene_preferences**

*Single cell-type gene preferences*

**Description**

Measure enrichment of individual cell-types in a signature matrix.

Internal function as part of tissue_scMappR_internal(). This function takes genes preferentially expressed within a gene list, each cell-type and the background (i.e. all genes within the signature matrix) before completing the cell-type specific enrichment of the inputted gene list on each cell type. This function then returns a table describing the cell-type enrichments (p-value and odds ratio) of each cell-type.

**Usage**

```r
single_gene_preferences(
  hg_short,
  hg_full,
  study_name,
  outDir,
  toSave = FALSE,
  path = NULL
)
```

**Arguments**

- **hg_short** A list with two objects: a "preferences" and a "genesIn". Preferences is a list of gene symbols over-represented in each cell-type and genesIn were all the inputted genes.
- **hg_full** The same as hg_short but for every gene in the signature matrix.
- **study_name** Name of output table.
- **outDir** Directory where table is outputted.
- **toSave** Allow scMappR to write files in the current directory (T/F).
- **path** If toSave == TRUE, path to the directory where files will be saved.
Value

single_gene_preferences A gene-set enrichment table of individual cell-type enrichment.

Examples

```r
# load in signature matrices
data(POA_example)
POA_gene <- POA_example$POA_gene
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
sig <- get_gene_symbol(POA_Rank_signature)
Signature <- POA_Rank_signature
rownames(Signature) <- sig$rownames
genes <- rownames(Signature)[1:60]
heatmap_test <- tissue_scMappR_custom( genes, signature_matrix = Signature,
                                      output_directory = "scMappR_test", toSave = FALSE)
single_preferences <- heatmap_test$single_celltype_preferences
```

Description

Example data for processing scRNA-seq count data with Seurat.

Usage

data(sm)

Format

A 752 x 236 matrix of class dgCMatrix where rows are genes and columns are cells. Data matrix is filled with counts detected from scRNAseq.

**TCTCTAACACAGGCCT** Barcode of one of the sequenced cells present. Each column is the count from a scRNA-seq dataset reprocessed by PanglaoDB.

Details

A dgCMatrix object containing count data for scRNA-seq processing.

Examples

data(sm)
Description

This function uses a Fisher’s-exact-test to rank gene-set enrichment.

Usage

tissue_by_celltype_enrichment(
  gene_list,
  species,
  name = "CT_Tissue_example",
  p_thresh = 0.05,
  rda_path = "",
  isect_size = 3,
  return_gmt = FALSE
)

Arguments

gene_list A character vector of gene symbols with the same designation (e.g. mouse symbol - mouse, human symbol - human) as the gene set database.
species Species of cell-type marker to use ('human' or 'mouse').
name Name of the pdf to be printed.
p_thresh The Fisher’s test cut-off for a cell-marker to be enriched.
rda_path Path to a .rda file containing an object called "gmt". Either human or mouse cell-type markers split by experiment. If the correct file isn’t present they will be downloaded from https://github.com/DustinSokolowski/scMappR_Data.
isect_size Number of genes in your list and the cell-type.
return_gmt Return .gmt file – recommended if downloading from online as it may have updated (T/F).

Details

Complete a Fisher’s exact test of an input list of genes against one of the two curated tissue by cell-type marker datasets from scMappR.

Value

List with the following elements:

enriched Data frame of enriched cell-types from tissues.
gmt Cell-markers in enriched cell-types from tissues.
Examples

```r
data(POA_example)
POA_genres <- POA_example$POA_genres
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
Signature <- POA_Rank_signature
rowname <- get_gene_symbol(Signature)
rownames(Signature) <- rowname$rowname
genes <- rownames(Signature)[1:100]

enriched <- tissue_by_celltype_enrichment(gene_list = genes,
species = "mouse",p_thresh = 0.05, isect_size = 3)
```

---

**tissue_scMappR_custom**  
*Gene List Visualization and Enrichment with Custom Signature Matrix*

**Description**

This function visualizes signature matrix, clusters subsetted genes, completes enrichment of individual cell-types and co-enrichment.

**Usage**

```r
tissue_scMappR_custom(
  gene_list,
  signature_matrix,
  output_directory = "custom_test",
  toSave = FALSE,
  path = NULL,
  gene_cutoff = 1,
  is_pvalue = TRUE
)
```

**Arguments**

- `gene_list`: A list of gene symbols matching that of the signature_matrix. Any gene symbol is acceptable.
- `signature_matrix`: Pre-computed signature matrix with matching gene names.
- `output_directory`: Directory made containing output of functions.
- `toSave`: Allow scMappR to write files in the current directory (T/F).
path If toSave == TRUE, path to the directory where files will be saved.
gene_cutoff Value cut-off (generally rank := log10(Padj)) for a gene to be considered a marker.
is_pvalue If signature matrix is p-value before rank is applied (not recommended) (T/F).

Details
This function is roughly the same as tissue_scMappR_internal, however now there is a custom signature matrix. It generates a heatmap of the signature matrix and your inputted gene list, as well as single cell-type and co-celltype enrichment.

Value
List with the following elements:
background_heatmap Data frame of the entire gene by cell-type signature matrix inputted.
gene_list_heatmap Data frame of inputted signature matrix subsetted by input genes.
single_celltype_preferences Data frame of enriched cell-types.
group_celltype_preference Data frame of groups of cell-types enriched by the same genes.

Examples

# load in signature matrices
data(POA_example)
POA_generes <- POA_example$POA_generes
POA_OR_signature <- POA_example$POA_OR_signature
POA_Rank_signature <- POA_example$POA_Rank_signature
sig <- get_gene_symbol(POA_Rank_signature)
Signature <- POA_Rank_signature
rownames(Signature) <- sig$rownames
genes <- rownames(Signature)[1:60]
heatmap_test <- tissue_scMappR_custom(genes, signature_matrix = Signature,
                                       output_directory = "scMappR_test", toSave = FALSE)
tissue_scMappR_internal

tissue_scMappR_internal(
    gene_list,
    species,
    output_directory,
    tissue,
    rda_path = "",
    cluster = "Pval",
    genecex = 0.01,
    raw_pval = FALSE,
    path = NULL,
    toSave = FALSE,
    drop_unkown_celltype = FALSE
)

Arguments

gene_list A list of gene symbols, mouse or human.
species "mouse", "human" or "-9" if using a precomputed signature matrix.
output_directory If toSave = TRUE, the name of the output directory that would be built.
tissue Name of the tissue in "get_tissues".
rda_path Path to the .rda file containing all of the signature matrices.
cluster 'Pval' or 'OR' depending on if you want to cluster odds ratios or pvalues of CT preferences.
genecex The size of the gene names of the rows in the heatmap.
raw_pval If the inputted signature matrix are raw (untransformed) p-values – recommended to generate rank first (T/F).
path If toSave == TRUE, path to the directory where files will be saved.
toSave Allow scMappR to write files in the current directory (T/F).
drop_unkown_celltype Whether or not to remove "unknown" cell-types from the signature matrix (T/F).

Details

This function takes a list of genes and a tissue that is contained in current signature matrices before and generating heatmaps of cell-type preferences. It then completes cell-type enrichment of each individual cell-type, then, if more than two cell-types are significantly enriched, co-enrichment of those enriched cell-types is then computed.

Value

List with the following elements:

background_heatmap Data frame of the entire gene by cell-type signature matrix inputted.
gene_list_heatmap
   Data frame of inputted signature matrix subsetted by input genes.

single_celltype_preferences
   Data frame of enriched cell-types.

group_celtype_preference
   Data frame of groups of cell-types enriched by the same genes.

Examples

```r
data(POA_example) # region to preoptic area
Signature <- POA_example$POA_Rank_signature # signature matrix
rowname <- get_gene_symbol(Signature) # get signature
rownames(Signature) <- rowname$rownames
genes <- rownames(Signature)[1:60]
rda_path1 = "" # data directory (if it exists)

# set toSave = TRUE and path = output directory of your choice
internal <- tissue_scMappR_internal(genes, "mouse", output_directory = "scMappR_TesInternal",
tissue = "hypothalamus", toSave = FALSE)
```

---

**tochr**

*To Character.*

**Description**

This function checks if your vector is not a character and if not, will convert it to a character.

**Usage**

```r
tochr(x)
```

**Arguments**

- `x` A character, factor or numeric vector.

**Value**

`tochr` Returns a character vector.
Examples

```r
# vector of factors
fact <- factor(c("a", "b", "c", "d"))
# convert to character
char <- tochr(fact)
```

### Description

This function checks if your vector is not a character and if it is, then converts it to a numeric.

### Usage

```r
toNum(x)
```

### Arguments

- **x**  
  A character, factor, or numeric vector.

### Value

- **toNum** Returns a numeric vector.

### Examples

```r
# vector of factors
fact <- factor(c("1", "2", "3", "4"))
# convert to numeric
num <- toNum(fact)
```
topgenes_extract

Extract Top Markers

Description
Internal – Extracts strongest cell-type markers from a Seurat object.

Usage

```r
topgenes_extract(generes, padj = 0.05, FC = 1.5, topNum = 30)
```

Arguments

- `generes`: A list of cell-type markers with fold-changes and p-values (FindMarkers output in Seurat).
- `padj`: The p-value (FDR) cutoff.
- `FC`: The fold-change cutoff.
- `topNum`: The number of genes to extract.

Details
Internal, this function runs through a list of outputs from FindMarkers objects in Seurat and will extract genes past a padj and fold-change threshold. Then it extracts the topNum number of genes. If you have not used the FindMarkers function, then a list of summary statistics with fold-change designated by `avg_logFC` and p-val by `p_val_adj`.

Value

topgenes_extract Returns a list of character vectors with the top (topNum) of gene markers for each cell-type.

Examples

```r
# load generes object
data(POA_example)
topGenes <- topgenes_extract(POA_example$POA_generes)
```
Index

*Topic datasets
gmt, 12
PBMC_example, 21
POA_example, 23
scMappR_tissues, 29
sm, 32
cellmarker_enrich, 2
coEnrich, 4
DeconRNAseq_CRAN, 5
deconvolute_and_contextualize, 6
extract_genes_cell, 9
generes_to_heatmap, 10
get_gene_symbol, 11
gmt, 12
gProfiler_cellWeighted_Foldchange, 13
gsva_cellIdentify, 14
heatmap_generation, 15
human_mouse_ct_marker_enrich, 17
make_TF_barplot, 18
pathway_enrich_internal, 19
PBMC_example, 21
plotBP, 22
POA_example, 23
process dgTMatrix_lists, 24
process_from_count, 25
scMappR_and_pathway_analysis, 27
scMappR_tissues, 29
seurat_toGeneres, 30
single_gene_preferences, 31
sm, 32
tissue_by_celltype_enrichment, 33
tissue_scMappR_custom, 34
tissue_scMappR_internal, 35
tochr, 37
toNum, 38
topgenes_extract, 39