Package 'scNLP'

June 23, 2021

Type Package

Title Tools for applying natural language processing (NLP) techniques to single-cell (sc) omics data.

Version 0.1.0

Description
Tools for applying natural language processing (NLP) techniques to single-cell (sc) omics data.

License MIT + file LICENSE

URL `https://github.com/bschilder/scNLP`

BugReports `https://github.com/bschilder/scNLP/issues`

Encoding UTF-8

LazyData true

Depends R (>= 3.6.0)

SystemRequirements Python (>= 3.7.0)

biocViews

Imports remotes,
magrittr,
BiocManager,
dplyr,
tibble,
data.table,
tidytext,
ggplot2,
plotly,
Matrix,
ggrepel,
scales,
Seurat,
future,
SingleCellExperiment,
SummarizedExperiment,
pals

VignetteBuilder knitr

RoxygenNote 7.1.1

Suggests rmarkdown,
knitr

R topics documented:

plot_tfidf 2
pseudo_sce 3
pseudo_seurat 4
run_tfidf 4
seurat_pipeline 5
tfidf 6
wordcloud_tfidf 7

Index **8**

plot_tfidf *Plot tf-idf results in reduced dimensions*

Description

Plot tf-idf enrichment results in reduced dimensional space (e.g. PCA/tSNe/UMAP), Reduced dimensions can be computed based on single-cell data (e.g. RNA expression). .

Usage

```
plot_tfidf(
  object = NULL,
  reduction = "UMAP",
  label_var = "label",
  cluster_var = "seurat_clusters",
  replace_regex = "[.]|[_]|[-]",
  terms_per_cluster = 3,
  size_var = 1,
  color_var = "cluster",
  point_alpha = 0.7,
  point_palette = c(unname(pals::alphabet()), rev(unname(pals::alphabet2()))),
  density_palette = "Purples",
  density_adjust = 0.2,
  label_fill = alpha(c("white"), 0.7),
  show_plot = T,
  background_color = "white",
  text_color = "black",
  interact = F,
  verbose = T,
  ...
)
```

Arguments

`object`	Single-cell data object. Can be in `SingleCellExperiment` or Seurat format.
`reduction`	Name of the reduction to use (*case insensitive*).
`label_var`	Which cell metadata column to input to tf-idf enrichment analysis.
`cluster_var`	Which cell metadata column to use to identify which cluster each cell is assigned to.

`replace_regex` Characters by which to split `label_var` into terms (i.e. tokens) for tf-idf enrichment analysis.

`terms_per_cluster` The number of top significantly enriched terms to include per cluster.

`size_var` Point size variable in `object` metadata.

`color_var` Point color variable in `object` metadata.

`point_alpha` Point opacity.

`point_palette` Point palette.

`density_palette` Density palette.

`density_adjust` Density adjust (controls granularity of density plot).

`label_fill` Cluster label background color.

`show_plot` Whether to print the plot.

`background_color` Plot background color.

`text_color` Cluster label text color.

`interact` Whether to make the plot interactive with **plotly**.

`verbose` Whether to print messages.

`...` Additional arguments to be passed to `ggplot2::geom_point(aes_string(...))`.

Examples

```
data("scNLP")
data("pseudo_seurat")

res <- plot_tfidf(object = pseudo_seurat,
                  label_var = "celltype",
                  cluster_var = "cluster",
                  show_plot = T)
```

pseudo_sce *Example* `SingleCellExperiment`

Description

Contains pseudobulk data (mean expression per cell-type) from 11 different datasets. Mean expression matrices have been downsampled to 1,000/21,000 genes.

Usage

```
pseudo_sce
```

Format

An object of class `SingleCellExperiment` with 1000 rows and 801 columns.

Examples

```
## Not run:
set.seed(2021)
pseudo_sce <- scKirby::ingest_data("/Users/schilder/Desktop/model_celltype_conservation/raw_data/scRNAseq/m
SingleCellExperiment::reducedDim(pseudo_sce,"UMAP") <- data.frame(SummarizedExperiment::colData(pseudo_sce)
pseudo_sce <- pseudo_sce[sample(1:nrow(pseudo_sce),1000), ]
usethis::use_data(pseudo_sce, overwrite = T)

## End(Not run)
```

---

pseudo_seurat *Example* Seurat

Description

Contains pseudobulk data (mean expression per cell-type) from 11 different datasets. Mean expression matrices have been downsampled to 1,000/21,000 genes.

Usage

```
pseudo_seurat
```

Format

An object of class Seurat with 1000 rows and 801 columns.

Examples

```
## Not run:
set.seed(2021)
pseudo_seurat <- scKirby::ingest_data("/Users/schilder/Desktop/model_celltype_conservation/raw_data/scRNAse
pseudo_seurat <- pseudo_seurat[sample(1:nrow(pseudo_seurat),1000), ]
usethis::use_data(pseudo_seurat, overwrite = T)

## End(Not run)
```

---

run_tfidf *Run tf-idf on single-cell data*

Description

Run tf-idf on single-cell data

Usage

```
run_tfidf(
  object = NULL,
  reduction = "UMAP",
  label_var = "label",
  cluster_var = "seurat_clusters",
  replace_regex = "[.]|[_]|[-]",
  terms_per_cluster = 3,
  force_new = F,
  return_all_results = F,
  verbose = T
)
```

Arguments

`object` Single-cell data object. Can be in `SingleCellExperiment` or Seurat format.

`reduction` Name of the reduction to use (*case insensitive*).

`label_var` Which cell metadata column to input to tf-idf enrichment analysis.

`cluster_var` Which cell metadata column to use to identify which cluster each cell is assigned to.

`replace_regex` Characters by which to split `label_var` into terms (i.e. tokens) for tf-idf enrichment analysis.

`terms_per_cluster` The number of top significantly enriched terms to include per cluster.

`force_new` If tf-idf results are already detected the metadata, set `force_new=T` to replace them with new results.

`return_all_results` Whether to return just the `object` with updated metadata (`return_all_results=F`), or all intermediate results (`return_all_results=F`).

`verbose` Whether to print messages.

Examples

```
library(scNLP)
data("pseudo_seurat")
pseudo_seurat_tfidf <- run_tfidf(object = pseudo_seurat,
                                 reduction = "UMAP",
                                 cluster_var = "cluster",
                                 label_var = "celltype")
head(pseudo_seurat_tfidf@meta.data)
```

`seurat_pipeline` *Run standardized* **Seurat** *pipeline*

Description

Run **Seurat** pipeline on **Seurat** object or raw `counts` and `meta.data`.

Usage

```
seurat_pipeline(
  seurat_obj = NULL,
  counts = NULL,
  meta.data = NULL,
  nfeatures = 2000,
  vars.to.regress = NULL,
  dims = 1:50,
  add_specificity = F,
  assay_name = "RNA",
  default_assay = NULL,
  n.components = 2L,
  log_norm = F,
  parallelize = T,
  seed = 2020
)
```

Details

Automatically performs

`FindVariableFeatures` Variable feature selection

`NormalizeData` Data normalization

`RunPCA` PCA

`RunUMAP` UMAP

`FindNeighbors` K-nearest neighbors

`FindClusters` Clustering

`tfidf` *tfidf*

Description

Run tf-idf on a metadata table.

Usage

```
tfidf(
  clusts,
  label_var = "dataset",
  cluster_var = "seurat_clusters",
  terms_per_cluster = 1,
  replace_regex = "[.]|[_]|[-]",
  force_new = F
)
```

Arguments

`clusts` data.frame/data.table with the per-cell meteadata and cluster assignments.

`label_var` Which cell metadata column to input to tf-idf enrichment analysis.

`cluster_var` Which cell metadata column to use to identify which cluster each cell is assigned to.

`terms_per_cluster` The number of top significantly enriched terms to include per cluster.

`replace_regex` Characters by which to split `label_var` into terms (i.e. tokens) for tf-idf enrichment analysis.

`force_new` If tf-idf results are already detected the metadata, set `force_new=T` to replace them with new results.

wordcloud_tfidf *Wordcloud from tf-idf results*

Description

Wordcloud from tf-idf results

Usage

```
wordcloud_tfidf(
  object,
  label_var = "celltype",
  cluster_var = "cluster",
  terms_per_cluster = 10,
  show_plot = T,
  ...
)
```

Arguments

`object` Single-cell data object. Can be in `SingleCellExperiment` or Seurat format.

`label_var` Which cell metadata column to input to tf-idf enrichment analysis.

`cluster_var` Which cell metadata column to use to identify which cluster each cell is assigned to.

`terms_per_cluster` The number of top significantly enriched terms to include per cluster.

`...` Additional parameters to pass to `ggplot2::ggplot(aes_string(...))`.

Index

∗ **datasets**
    pseudo_sce, 3
    pseudo_seurat, 4

plot_tfidf, 2
pseudo_sce, 3
pseudo_seurat, 4

run_tfidf, 4

seurat_pipeline, 5

tfidf, 6

wordcloud_tfidf, 7