Package ‘demuxSNP’

May 14, 2024

**Title**  scRNAseq demultiplexing using cell hashing and SNPs

**Version**  1.2.0

**Description**

This package assists in demultiplexing scRNAseq data using both cell hashing and SNPs data. The SNP profile of each group is learned using high confidence assignments from the cell hashing data. Cells which cannot be assigned with high confidence from the cell hashing data are assigned to their most similar group based on their SNPs. We also provide some helper functions to optimise SNP selection, create training data and merge SNP data into the SingleCellExperiment framework.

**URL**  https://github.com/michaelplynch/demuxSNP

**BugReports**  https://github.com/michaelplynch/demuxSNP/issues

**License**  GPL-3

**Encoding**  UTF-8

**Roxygen**  list(markdown = TRUE)

**RoxygenNote**  7.3.1

**Depends**  R (>= 4.3.0), SingleCellExperiment, VariantAnnotation, ensembldb

**Imports**  MatrixGenerics, BiocGenerics, class, GenomeInfoDb, IRanges, Matrix, SummarizedExperiment, demuxmix, methods, KernelKnn, dplyr

**Suggests**  knitr, rmarkdown, ComplexHeatmap, viridisLite, ggpubar, dittoSeq, EnsDb.Hsapiens.v86, BiocStyle, RefManageR, testthat (>= 3.0.0), Seurat

**biocViews**  Classification, SingleCell

**VignetteBuilder**  knitr

**LazyData**  false

**Config/testthat/edition**  3

**git_url**  https://git.bioconductor.org/packages/demuxSNP

**git_branch**  RELEASE_3_19
Description

Add SNPs to SingleCellExperiment object

Usage

add_snps(sce, mat, thresh = 0.8)

Arguments

sce object of class SingleCellExperiment
mat object of class matrix, output from VarTrix in 'consensus' mode (default)
thresh threshold presence of SNP, defaults to 0.8

Value

Updated SingleCellExperiment object with snps in altExp slot
Examples

```r
data(multiplexed_scrrnaseq_sce, vartrix_consensus_snps)
multiplexed_scrrnaseq_sce <- add_snps(sce = multiplexed_scrrnaseq_sce,
mat = vartrix_consensus_snps,
thresh = 0.8)
```

**commonvariants_1kgenomes_subset**

*Sample vcf file*

**Description**

VCF file containing SNPs from a subset of the 1k Genomes common variants HG38 genome build.

**Usage**

```r
data(commonvariants_1kgenomes_subset)
```

**Format**

An object of class CollapsedVCF with 2609 rows and 0 columns.

**Value**

`commonvariants_1kgenomes_subset`:
An object of class CollapsedVcf

**Source**


---

**common_genes**

Return a character vector of top n most frequent genes from a SingleCellExperiment object.

**Description**

Returns a character vector of the top n most frequently expressed genes from the counts of the SingleCellExperiment object. Expression is based on having a count > 0 in a given cell.

**Usage**

```r
common_genes(sce, n = 100)
```
Arguments

sce  a SingleCellExperiment object
n   number of genes to be returned. Defaults to n=100.

Value

character vector of n most frequently expressed genes.

Examples

data(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- common_genes(multiplexed_scrnaseq_sce)

data(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- high_conf_calls(multiplexed_scrnaseq_sce)

high_conf_calls

Run demuxmix to determine high-confidence calls

Description

Run demuxmix to determine high-confidence calls

Usage

high_conf_calls(sce, assay = "HTO", pacpt = 0.95)

Arguments

sce  Object of class SingleCellExperiment with HTO (or similar) altExp assay
assay  Name of altExp for cell hashing counts to be retrieved from
pacpt  acceptance probability for demuxmix model

Value

Updated SingleCellExperiment object with logical vector indicating training data, data to be classified (all cells) and assigned labels for all cells.

Examples

data(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- high_conf_calls(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce

SingleCellExperiment object containing multiplexed RNA and HTO data from six biological samples

Description

Example SingleCellExperiment object containing demultiplexed scRNAseq data from six donors, used throughout and built upon in demuxSNP workflow.

Usage

data(multiplexed_scrnaseq_sce)

Format

An object of class SingleCellExperiment with 259 rows and 2000 columns.

Value

multiplexed_scrnaseq_sce:
An object of class SingleCellExperiment

reassign

Reassign cells using knn

Description

k-nearest neighbour classification of cells. Training data is intended to be labels of cells confidently called using cell hashing based methods and their corresponding SNPs. Prediction data can be remaining cells but can also include the training data. Doublets are simulated by randomly combining 'd' SNP profiles from each grouping combination.

Usage

reassign(
    sce,
    k = 10,
    d = 10,
    train_cells = sce$train,
    predict_cells = sce$predict
)
Arguments

sce: object of class SingleCellExperiment
k: number of neighbours used in knn, defaults to 10
d: number of doublets per group combination to simulate, defaults to 10
train_cells: logical vector specifying which cells to use to train classifier
predict_cells: logical vector specifying which cells to classify

Value

A SingleCellExperiment with updated group assignments called `knn`

Examples

data(multiplexed_scrnaseq_sce, vartrix_consensus_snps)
multiplexed_scrnaseq_sce <- high_conf_calls(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- add_snps(sce = multiplexed_scrnaseq_sce, mat = vartrix_consensus_snps, thresh = 0.8)
multiplexed_scrnaseq_sce <- reassign(sce = multiplexed_scrnaseq_sce, k = 10)

reassign_balanced

Reassign cells using balanced knn with jaccard distance

Description

k-nearest neighbour classification of cells. Training data is intended to be labels of cells confidently called using cell hashing based methods and their corresponding SNPs. Prediction data can be remaining cells but can also include the training data. Doublets are simulated by randomly combining `d` SNP profiles from each grouping combination.

Usage

reassign_balanced(
  sce,
  k = 10,
  d = 10,
  train_cells = sce$train,
  predict_cells = sce$predict,
  n = NULL,
  nmin = 50
)
reassign_jaccard

Arguments

sce  object of class SingleCellExperiment  
k  number of neighbours used in knn, defaults to 10  
d  number of doublets per group combination to simulate, defaults to 10  
train_cells  logical vector specifying which cells to use to train classifier 
predict_cells  logical vector specifying which cells to classify 
n  number of cells per group (otherwise will be calculated from data)  
nmin  min n per class (where available)  

Value

A SingleCellExperiment with updated group assignments called `knn`  

Examples

```r
data(multiplexed_scrnaseq_sce, vartrix_consensus_snps)
multiplexed_scrnaseq_sce <- high_conf_calls(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- add_snps(sce = multiplexed_scrnaseq_sce,
mat = vartrix_consensus_snps,
thresh = 0.8)
multiplexed_scrnaseq_sce <- reassign_balanced(sce = multiplexed_scrnaseq_sce, k = 10)
```

---

Description

k-nearest neighbour classification of cells. Training data is intended to be labels of cells confidently called using cell hashing based methods and their corresponding SNPs. Prediction data can be remaining cells but can also include the training data. Doublets are simulated by randomly combining 'd' SNP profiles from each grouping combination.

Usage

```r
reassign_jaccard(
  sce,
  k = 10,
  d = 10,
  train_cells = sce$train,
  predict_cells = sce$predict
)
```
**subset_vcf**

**Arguments**

- **sce**
  object of class SingleCellExperiment
- **k**
  number of neighbours used in knn, defaults to 10
- **d**
  number of doublets per group combination to simulate, defaults to 10
- **train_cells**
  logical vector specifying which cells to use to train classifier
- **predict_cells**
  logical vector specifying which cells to classify

**Value**

A SingleCellExperiment with updated group assignments called 'knn'

**Examples**

```r
data(multiplexed_scrnaseq_sce, vartrix_consensus_snps)
multiplexed_scrnaseq_sce <- high_conf_calls(multiplexed_scrnaseq_sce)
multiplexed_scrnaseq_sce <- add_snps(sce = multiplexed_scrnaseq_sce,
mat = vartrix_consensus_snps, thresh = 0.8)
multiplexed_scrnaseq_sce <- reassign(sce = multiplexed_scrnaseq_sce, k = 10)
```

**subset_vcf**

Subset common variants vcf file to only SNPs seen in 'top_genes'

**Description**

Subset common variants vcf file to only SNPs seen in 'top_genes'

**Usage**

```r
subset_vcf(vcf, top_genes, ensdb)
```

**Arguments**

- **vcf**
  object of class CollapsedVCF
- **top_genes**
  output from 'common_genes' function, alternatively character vector containing custom gene names.
- **ensdb**
  object of class EnsDb corresponding to organism, genome of data

**Value**

object of class CollapsedVCF containing subset of SNPs from supplied vcf seen in commonly expressed genes
Examples

```r
data(multiplexed_scrnaseq_sce, commonvariants_1kgenomes_subset)
top_genes <- common_genes(multiplexed_scrnaseq_sce)
ensdb <- EnsDb.Hsapiens.v86::EnsDb.Hsapiens.v86
small_vcf <- subset_vcf(commonvariants_1kgenomes_subset, top_genes, ensdb)
```

---

### vartrix_consensus_snps

#### Sample VarTrix output

**Description**

A sample output from VarTrix corresponding to the sce SingleCellExperiment object for a subset of SNPs located in well observed genes.

**Usage**

```r
data(vartrix_consensus_snps)
```

**Format**

An object of class `matrix` (inherits from `array`) with 2542 rows and 2000 columns.

**Value**

```r
vartrix_consensus_snps:
An object of class `matrix`
```
Index

* datasets
  commonvariants_1kgenomes_subset, 3
  multiplexed_scnnaseq_sce, 5
  vartrix_consensus_snps, 9

add_snps, 2
common_genes, 3
commonvariants_1kgenomes_subset, 3
high_conf_calls, 4
multiplexed_scnnaseq_sce, 5
reassign, 5
reassign_balanced, 6
reassign_jaccard, 7
subset_vcf, 8
vartrix_consensus_snps, 9