DWLS

November 26, 2018

Title Dampened Weighted Least Square for Deconvolution of Bulk mRNA Data

Version 0.1

Author Daphne Tsoucas, Qian Zhu, Rui Dong, Guo-cheng Yuan

Description Deconvolution of bulk mRNA data using single-cell RNAseq to provide cell type specific signatures

Maintainer Qian Zhu <zqian@jimmy.harvard.edu>

biocViews

Depends R (>= 2.14.0), reshape (>= 0.8.8), quadprog (>= 1.5),
e1071 (>= 1.7.0), Seurat (>= 2.3.0), ROCR (>= 1.0),
varhandle (>= 2.0.3), MAST (>= 1.0.5)

License GPL

NeedsCompilation no

Repository CRAN

Date/Publication 2018-11-25 01:00:00

R topics documented:

buildSignatureMatrixMAST

Builds a signature matrix from single-cell data by calling MAST.

Description

Given a single-cell matrix and cell type ids, the function constructs a gene by cell type matrix containing averaged expression values for each cell type, for highly differentially expressed genes. Differential expression analysis is conducted through MAST.

Usage

```
buildSignatureMatrixMAST(scdata, id, path, diff.cutoff=0.5, pval.cutoff=0.01)
```

Arguments

`scdata` Gene by cell single-cell data matrix.

`id` List containing cell type ids corresponding to each column of the `scdata`

`path` String corresponding to path in which to save DE information.

`diff.cutoff` Scalar corresponding to the log fold change cutoff to use when picking differentially expressed genes (default=0.5)

`pval.cutoff` Scalar corresponding to the pvalue cutoff to use when picking differentially expressed genes (default=0.01)

Value

Gene by cell type matrix containing the gene expression signature of each cell type.

Examples

```
buildSignatureMatrixMAST(scdata, id, path, diff.cutoff=0.5, pval.cutoff=0.01)
```

buildSignatureMatrixUsingSeurat

Builds a signature matrix from single-cell data by calling Seurat.

Description

Given a single-cell matrix and cell type ids, the function constructs a gene by cell type matrix containing averaged expression values for each cell type, for highly differentially expressed genes. Differential expression analysis is conducted through Seurat.

Usage

```
buildSignatureMatrixUsingSeurat(scdata, id, path, diff.cutoff=0.5, pval.cutoff=0.01)
```

Arguments

`scdata` Gene by cell single-cell data matrix.

`id` List containing cell type ids corresponding to each column of the `scdata`

`path` String corresponding to path in which to save DE information.

`diff.cutoff` Scalar corresponding to the log fold change cutoff to use when picking differentially expressed genes (default=0.5)

`pval.cutoff` Scalar corresponding to the pvalue cutoff to use when picking differentially expressed genes (default=0.01)

Value

Gene by cell type matrix containing the gene expression signature of each cell type.

Examples

```
buildSignatureMatrixUsingSeurat(scdata, id, path, diff.cutoff=0.5, pval.cutoff=0.01)
```

`DEAnalysis` *Calculates differentially expressed genes between each cell type and the rest for a given single-cell data set. (Using Seurat's function)*

Description

Finds differentially expressed genes between cell types using the MAST test in Seurat. Saves differential expression information to specified path, in the form of a matrix for each cell type. This is used by the buildSignatureMatrixUsingSeurat function in DWLS.

Usage

```
DEAnalysis(scdata, id, path)
```

Arguments

`scdata` Gene by cell single-cell data matrix.

`id` List containing cell type ids corresponding to each column of the `scdata`

`path` String corresponding to path in which to save DE information.

Value

Saves DE information to specified path, in the form of a matrix for each cell type.

Examples

```
DEAnalysis(scdata, id, path)
```

DEAnalysisMAST *Calculates differentially expressed genes between each cell type and the rest for a given single-cell data set. (Using MAST function directly, without using Seurat)*

Description

Finds differentially expressed genes between cell types using the MAST's lrTest function. Saves differential expression information to specified path, in the form of a matrix for each cell type. This is used by the buildSignatureMatrixMAST function in DWLS.

Usage

```
DEAnalysisMAST(scdata, id, path)
```

Arguments

scdata	Gene by cell single-cell data matrix.
id	List containing cell type ids corresponding to each column of the scdata
path	String corresponding to path in which to save DE information.

Value

Saves DE information to specified path, in the form of a matrix for each cell type.

Examples

```
DEAnalysisMAST(scdata, id, path)
```

findDampeningConstant *Find a dampening constant for dampened weighted least squares using cross-validation.*

Description

For a subset of genes, the function calculates the dampened weighted least squares solution for a particular dampening constant. This is repeated 100 times per dampening constant over a range of dampening constants. The dampening constant with the lowest cross-validation variance is chosen.

Usage

```
findDampeningConstant(S, B, goldStandard)
```

Arguments

S	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
B	List containing bulk data values.
goldStandard	An initial estimate of the number of each cell type present in the bulk data set. Often this estimate is derived from the solveOLS function.

Value

Positive integer defining the dampening constant. The maximum weight value allowed will be $d = 2^(j - 1)$.

Examples

```
findDampeningConstant(S,B,solveOLS(S,B))
```

---

solveDampenedWLS *Solve overdetermined system of equations, $Sx = B$, for x, using dampened weighted least squares.*

Description

Dampened weighted least squares involves solving:

$$min_{x>0} \sum_{i=1}^{n} w_i (B_i - (Sx)_i)^2$$

through an iterative approach. S is a gene by cell-type signature matrix, B is a list containing bulk data, and x is the number of cells from each cell type present in the bulk data. x is then normalized:

$$\tilde{x} = \frac{x}{\sum_{j=1}^{n} x},$$

where j is the cell-type index.

Usage

```
solveDampenedWLS(S, B)
```

Arguments

`S`	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
`B`	List containing bulk data values.

Value

List containing proportion of each cell type present in bulk data set.

Examples

```
prop <- solveDampenedWLS(S,B)
```

`solveDampenedWLSj` *Solve overdetermined system of equations, $Sx = B$, for x, using dampened weighted least squares given a specific dampening constant*

Description

Dampened weighted least squares involves solving:

$$min_{x>0} \sum_{i=1}^{n} w_i (B_i - (Sx)_i)^2$$

through an iterative approach. The dampening constant determines the allowed maximum value of the weights. S is a gene by cell-type signature matrix, B is a list containing bulk data, and x is the number of cells from each cell type present in the bulk data. x is then normalized:

$$\tilde{x} = \frac{x}{\sum_{j=1}^{n} x},$$

where j is the cell-type index.

Usage

```
solveDampenedWLSj(S, B, goldstandard, j)
```

Arguments

`S`	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
`B`	List containing bulk data values.
`goldstandard`	An initial estimate of the number of each cell type present in the bulk data set. Often this estimate is derived from the `solveOLS` function.
`j`	Positive integer defining the dampening constant. The maximum weight value allowed will be $d = 2^(j - 1)$

Value

List containing proportion of each cell type present in bulk data set.

Examples

```
prop <- solveDampenedWLSj(S, B, solveOLS(S,B), 5)
```

solveOLS — *Solve overdetermined system of equations, Sx=B, for x, using constrained ordinary least squares*

Description

Constrained ordinary least squares is implemented using a quadratic programming approach, as such:

$$min_{x>0}||B - Sx||^2$$

S is a gene by cell-type signature matrix, B is a list containing bulk data, and x is the number of cells from each cell type present in the bulk data. x is then normalized:

$$\tilde{x} = \frac{x}{\sum_{j=1}^{n} x},$$

where j is the cell-type index.

Usage

```
solveOLS(S, B)
```

Arguments

`S`	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
`B`	List containing bulk data values.

Value

List containing proportion of each cell type present in bulk data set.

Examples

```
prop <- solveOLS(S,B)
```

solveSVR — *Solve overdetermined system of equations, $Sx = B$, for x, using nu-support vector regression with a linear kernel*

Description

Solve overdetermined system of equations, $Sx = B$, for x, using nu-support vector regression with a linear kernel. S is a gene by cell-type signature matrix, B is a list containing bulk data, and x is the number of cells from each cell type present in the bulk data. x is then normalized:

$$\tilde{x} = \frac{x}{\sum_{j=1}^{n} x},$$

where j is the cell-type index.

Usage

```
solveSVR(S, B)
```

Arguments

S	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
B	List containing bulk data values.

Value

List containing proportion of each cell type present in bulk data set.

Examples

```
prop <- solveSVR(S,B)
```

trimData *Trims bulk data and single-cell signature matrix to contain the same genes*

Description

Trims bulk data and single-cell signature matrix to contain the same genes

Usage

```
trimData(Signature, bulkData)
```

Arguments

Signature	Gene by cell-type matrix consisting of averaged expression values across cells from each cell type
bulkData	List containing bulk data values.

Value

List containing trimmed Signature (S) and bulk (B) matrices.

Examples

```
trimmed <- trimData(Signature,bulkData)
S <- trimmed$S
B <- trimmed$B
```

Index