

Package ‘malani’

July 22, 2025

Type Package
Title Machine Learning Assisted Network Inference
Version 1.0
Author Mehrab Ghanat Bari
Maintainer Mehrab Ghanat Bari <m.ghanatbari@gmail.com>
Description Find dark genes. These genes are often disregarded due to no detected mutation or differential expression, but are important in coordinating the functionality in cancer networks.
License GPL-3
LazyData TRUE
Depends e1071, stats
RoxygenNote 5.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2016-09-26 23:44:53

Contents

dat	2
grp	2
Gsvmod	2
intGenes	3
malanidata	4
pairmod	4
Index	5

dat	<i>A matrix of expression values.</i>
-----	---------------------------------------

Description

A numeric matrix 100*20.

Usage

dat

Format

matrix.

grp	<i>A vector of class labels for dat.</i>
-----	--

Description

Vector length of 20.

Usage

grp

Format

vector

Gsvmod	<i>G SVM models.</i>
--------	----------------------

Description

Returns accuracy performance of all genes. G support vector machine (SVM) classifiers trained using G different data matrixes, are used to predict labels in test data. Models are ranked based on prediction performances.

Usage

```
Gsvmod(dat.train, lab.train, dat.test, lab.test)
```

Arguments

<code>dat.train</code>	Train data with G features and $(k-1)*S/k$ samples. Parameter k comes from cross-validation scheme and is specified by user (default is 2).
<code>lab.train</code>	Class labels for train data.
<code>dat.test</code>	Test data with G features and S/k samples.
<code>lab.test</code>	Class labels for test data.

Value

Accuracy scores for models. Each model represents one gene.

<code>intGenes</code>	<i>Select initial gene list from original data matrix.</i>
-----------------------	--

Description

Train G-1 SVM models in k-fold cross validation scheme to select initial genes list.

Usage

```
intGenes(dat, grp, nfolds.out = 2, top.per = 0.05)
```

Arguments

<code>dat</code>	Original gene expression data matrix with G rows (number of genes) and S column (number of samples).
<code>grp</code>	Class labels.
<code>nfolds.out</code>	Outer cross validation number (default is 2).
<code>top.per</code>	All genes are ranked based on their models performance and top.per% of them are selected as initial genes.

Value

Selected initial genes.

Examples

```
data(malanidata)
int <- intGenes(dat,grp)
print(int$top.genes)
```

malanidata	<i>Dataset for malani package</i>
------------	-----------------------------------

Description

A numeric matrix $G \times S$ contains gene expressions data. G are the genes (rows) and S are the samples (columns).

Usage

```
malanidata
```

Format

A matrix of numeric values, 100 genes, 20 samples and class labels.

Examples

```
data(malanidata)
```

pairmod	<i>Find best performing pairs</i>
---------	-----------------------------------

Description

Combine each gene in initial set with all genes in the original set. Top `npair` pairs are selected to construct the Q matrix.

Usage

```
pairmod(X, LX, theta, npair = 10)
```

Arguments

<code>X</code>	Original gene expression data matrix. With G rows (number of genes) and S column (number of samples).
<code>LX</code>	Class labels.
<code>theta</code>	Initial gene set.
<code>npair</code>	Given a gene in initial set, top <code>npair</code> best performing pairs correspond to that gene are selected (Default is 10).

Value

Best ($npair \times G / 20$) performing pairs.

Index

* **datasets**

dat, [2](#)

grp, [2](#)

malanidata, [4](#)

dat, [2](#), [2](#)

grp, [2](#)

Gsvmod, [2](#)

intGenes, [3](#)

malanidata, [4](#)

pairmod, [4](#)