Package 'rMisbeta'

November 6, 2020

Version 1.0

Date 2020-10-5

Title A Robust Missing Imputation Method for Gene Expression Data

Author Md. Shahjaman and Md. Nurul Haque Mollah

Maintainer Shahjaman <shahjaman_brur@yahoo.com>

Depends R (>= 4.0),stats

Imports ROC

Suggests MASS

Description It was developed especially for gene expression and metabolomics data analysis when the datasets are corrupted by outliers and missing values. The beta-divergence method was used to impute the missing values and modify the outliers.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2020-11-06 12:20:02 UTC

R topics documented:

`rMisbeta-package` *A Robust Missing Imputation Method for Gene Expression and Metabolomics Data Analysis*

Description

We developed a robust missing value imputation approach for gene expression and metabolomics data analysis using minimum beta-divergence method. This approach capable of handling both missing values and outliers, simultaneously.

Details

Package:	rMisbeta
Type:	Package
Version:	1.0
Date:	2020-10-03
License:	GPL (>=2.0)

Package rMisbeta has the six following functions:

Sim2Group():	This function generates the data from the one way ANOVA model for two groups.
OutMisDat():	This function returns the outliers and missing value incorporated data.
CalcMeanVar():	This function calculates the robust mean and variance from the the matrix in presence of outliers and missing values for function RobMeanVar().
RobMeanVar():	This function calculates the robust mean and variance from the the matrix in presence of outliers and missing values. The function RobMeanVar() also produces a weight called beta-weights for each of the values to detect the outliers in the dataset.
remat():	This function returns reconstructed data matrix by modifying the outliers and missing value using beta divergence method.
performance.eval():	This is the performance evaluation function. Which calculates TPR,TNR,FPR,PNR,AUC etc. as a measure of performance index.

Author(s)

Md Shahjaman and Md. Nurul Haque Mollah; Maintainer: Md Shahjaman, shahjaman_brur@yahoo.com

Examples

```
nG=1000
n1=n2=5
pde=0.1
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
TrueDE=Simdat$DEtrue
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
```

```
misdat_zero<-MisOutdat
misdat_zero[is.na(misdat_zero)]<-0
cl=rep(c(1,2),each=n1)

res=remat(MisOutdat,cl)
up_mat<-res$remat


pTtest_zero<-pTtest_beta<-NULL
for (j1 in 1:dim(xx)[1])
{
DataYY <- data.frame(YY =misdat_zero[j1,], FactorLevels = factor(cl))
DataYY2 <- data.frame(YY2=up_mat[j1,], FactorLevels2 = factor(cl))
pTtest_zero[j1] <-  t.test(YY~FactorLevels,data=DataYY, paired=FALSE)[[3]]
pTtest_beta[j1] <-  t.test(YY2~FactorLevels2,data=DataYY2, paired=FALSE)[[3]]
}

TopDEn<-seq(nG*pde/10, pde*nG, length=10)

performance_zero<-performance.eval(pTtest_zero,TrueDE,TopDEn,decreasing=FALSE);
performance_beta<-performance.eval(pTtest_beta,TrueDE,TopDEn,decreasing=FALSE);
plot(performance_zero$FPR,performance_zero$TPR,type="o",
xlab="False Positive Rate",ylab="True Positive Rate",ylim=c(0,1))
points(performance_beta$FPR,performance_beta$TPR,type="o",col=2)
legend("bottomright", c('t_test_zero','t_test_rMisbeta'),lwd=1,cex=0.8,col=c(1,2))
```

`CalcMeanVar` *This function estimates the robust mean and variance using beta-divergence method.*

Description

The CalcMeanVar() function estimates the robust mean and variance using beta-divergence method for RobMeanVar () function. The beta-weight function confirms that is the data contain outliers or not.The larger weights indicate the good data points and the smaller weights (near to zero) indicates the outlying data points.

Usage

```
CalcMeanVar(xx,Mo)
```

Arguments

`xx`	xx denotes a vector of data matrix.
`Mo`	Mo denotes median of xx.

Value

This function returns a data frame containing 3 components

MM	Robust mean vector produced by beta-divergence method.
VV	Robust variance produced by beta-divergence method.
WW	Weights of the each data points produced by beta-divergence method using weight function.

Author(s)

Md.Shahjaman and Md. Nurul Haque Mollah shahjaman_brur@yahoo.com

References

Shahjaman M, Mollah MHM, Rahman MR, Islam SSM and Mollah NHM. Robust identification of differentially expressed genes from RNA-seq data. Genomics 2020; 112(2): 2000:2010.

Examples

```
nG=1000
n1=n2=5
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
Datao=xx
cl=rep(c(1,2),each=n1)
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
res=remat(MisOutdat,cl)
up_mat<-res$remat
```

OutMisDat *This function allows user's to add outliers and missing values in the original dataset*

Description

OutMisDat() function returns the outliers and missing value incorporated data. The percentages of outliers and missing values can be provided by the users. If pctOut and pctMis both are provided 0 then this function returns the original dataset

Usage

```
OutMisDat(xx,pctOut,pctMis)
```

Arguments

xx	xx denotes a vector of data matrix.
pctOut	percentage of outliers defined by user.
pctMis	percentage of missing values defined by user.

Value

This function returns the outlier and missing incorporated data matrix

Datao	a dataset corrupted by Outlier and missing value

Author(s)

Md.Shahjaman; shahjaman_brur@yahoo.com

References

Shahjaman M, Mollah MHM, Rahman MR, Islam SSM and Mollah NHM. Robust identification of differentially expressed genes from RNA-seq data. Genomics 2020; 112(2): 2000:2010.

Examples

```
nG=1000
n1=n2=5
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
Datao=xx
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
```

---

performance.eval *This function estimates the different performance indices like, TPR,TNR,FPR,FNR,AUC etc. for number of top genes*

Description

This function estimates the different performance indeces,like TPR,TNR,FPR,FNR,AUC etc. to asses the performance of the method

Usage

```
performance.eval(PostP, de.true, TopG, decreasing = TRUE)
```

Arguments

PostP	p-values should be given to identify the different performance index.
de.true	The true DE information should be given to calculates the performance index.
TopG	How many Top DE genes will be used to calculate the performance indices.
decreasing	Is the p-values decreasing or increasing order.

Value

The following performance indices are produced by performance.eval():

TP	Number of True positive.
TN	Number of True negative.
FP	Number of False positive.
FN	Number of False negative.
R1	Specificity.
TPR	True positive rate.
TNR	True negative rate.
FPR	False positive rate.
FNR	False negative rate.
FDR	False discovery rate.
ER	Error rate.
AUC2	Area under the curve of ROC.
pAUC2	Partial Area under the curve of ROC with FDR controlled at 0.2.

Author(s)

Md.Shahjaman and Md. Nurul Haque Mollah shahjaman_brur@yahoo.com

Examples

```
# Performance evaluation in presence of outliers and missing values
nG=1000
n1=n2=5
pde=0.1
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
TrueDE=Simdat$DEtrue
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
misdat_zero<-MisOutdat
misdat_zero[is.na(misdat_zero)]<-0
cl=rep(c(1,2),each=n1)

res=remat(MisOutdat,cl)
up_mat<-res$remat


pTtest_zero<-pTtest_beta<-NULL
for (j1 in 1:dim(xx)[1])
{
DataYY <- data.frame(YY =misdat_zero[j1,], FactorLevels = factor(cl))
DataYY2 <- data.frame(YY2=up_mat[j1,], FactorLevels2 = factor(cl))
pTtest_zero[j1] <-  t.test(YY~FactorLevels,data=DataYY, paired=FALSE)[[3]]
pTtest_beta[j1] <-  t.test(YY2~FactorLevels2,data=DataYY2, paired=FALSE)[[3]]
}
```

```
TopDEn<-seq(nG*pde/10, pde*nG, length=10)

performance_zero<-performance.eval(pTtest_zero,TrueDE,TopDEn,decreasing=FALSE);
performance_beta<-performance.eval(pTtest_beta,TrueDE,TopDEn,decreasing=FALSE);
plot(performance_zero$FPR,performance_zero$TPR,type="o",
xlab="False Positive Rate",ylab="True Positive Rate",ylim=c(0,1))
points(performance_beta$FPR,performance_beta$TPR,type="o",col=2)
legend("bottomright", c('t_test_zero','t_test_rMisbeta'),lwd=1,cex=0.8,col=c(1,2))
```

`remat` *Reformulated data matrix after modification of outliers and missing imputation using beta divergence method*

Description

remat() function returns reformulated data matrix by modifying the outliers and missing value using the robust mean produced by RobMeanVar(). This function also produces the weights of each feature. The lower weights indicate that the corresponding feature is corrupted by the outliers.

Usage

```
remat(Datao,cl)
```

Arguments

`Datao`	Datao denotes a vector of data matrix with missing values and outliers.
`cl`	Binary class level. Usually 1 and 2.

Value

This function returns the following two components

`remat`	reformulated data matrix after modification of outliers and imputed the missing values
`betawt`	The weights of each feature. The lower weights indicate that the corresponding feature is corrupted by the outliers

Author(s)

Md.Shahjaman; shahjaman_brur@yahoo.com

References

Shahjaman M, Mollah MHM, Rahman MR, Islam SSM and Mollah NHM. Robust identification of differentially expressed genes from RNA-seq data. Genomics 2020; 112(2): 2000:2010.

Examples

```
nG=1000
n1=n2=5
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
Datao=xx
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
cl=rep(c(1,2),each=n1)
res=remat(MisOutdat,cl)
up_mat<-res$remat
```

RobMeanVar *This function estimates the robust mean and variance using beta-divergence method to reconstruct the data matrix.*

Description

The RobMeanVar() function estimates the robust mean and variance using beta-divergence method. If the gene expression data corrupted with outliers or missing values then this function calculates the robust mean for the corresponding outliers or missing gene vector and if the gene expression data does not contain outliers or missing values then it calculates the classical mean and variance.The beta-weight function confirms that, the data contain outliers or not. The larger weights indicate the good data points and the smaller weights (near to zero) indicates the outlying data points.

Usage

```
RobMeanVar(xx)
```

Arguments

xx	xx denotes a vector of data matrix with missing values or outliers.

Value

This function returns a data frame containing eight components

xx	Reconstructed data matrix by updating outlying or missing values using robust mean.
mu	Robust mean vector produced by beta-divergence method.
Var	Robust variance produced by beta-divergence method.
sd	Robust standard deviation sqrt(Var) produced by beta-divergence method.
out	Outlying indices produced by function RobMeanVar() using beta-divergence method.
Wt	Weights of the each data points produced by beta-divergence method using weight function.
Wt.out	Weights of the Outlier data points produced by beta-divergence method.
out.Thr	Outliers threshold.

Author(s)

Md.Shahjaman and Md.Nurul Haque Mollah; shahjaman_brur@yahoo.com

References

Shahjaman M, Mollah MHM, Rahman MR, Islam SSM and Mollah NHM. Robust identification of differentially expressed genes from RNA-seq data. Genomics 2020; 112(2): 2000:2010.

Examples

```
nG=1000
n1=n2=5
Simdat=Sim2Group(nG,n1,n2,var0=0.1,pde=0.1)
xx=Simdat$outmat
Datao=xx
cl=rep(c(1,2),each=n1)
MisOutdat<-OutMisDat(xx,pctOut=0.1,pctMis=0.1)
res=remat(MisOutdat,cl)
betawt<-res$betawt
plot(betawt)
```

`Sim2Group` *This function Sim2Group() simulates the gene expression data for two groups using one-way ANOVA model*

Description

Generates the gene expression data using one-way ANOVA model with two groups. The variance of both group should be same and the percentage of the DE genes will be given

Usage

```
Sim2Group(ng, n1, n2, var0 = 0.1, pde = 0.05)
```

Arguments

`ng`	The total number of genes to be generated.
`n1`	Number of samples in the first group.
`n2`	Number of samples in the second group.
`var0`	The variance of the both group.
`pde`	The proportion of the differentially expressed(DE) genes.

Value

This function returns the following components:

`outmat`	Simulated gene expression data for two groups.
`DEtrue`	True DE index.

Author(s)

Md.Shahjaman; shahjaman_brur@yahoo.com

Examples

```
n1=10;n2=10;
nG=1000
TSimDat<-Sim2Group(ng=nG,n1,n2,var0=0.1,pde=0.1)
Simdat<-TSimDat[[1]]
TrueDE<-TSimDat[[2]]
```

Index