# Package 'roughrf'

February 20, 2015

**Type** Package

**Title** Roughened Random Forests for Binary Classification

**Version** 1.0

**Date** 2015-01-28

**Author** Kuangnan Xiong

**Maintainer** Kuangnan Xiong <kxiong@albany.edu>

**Depends** R (>= 2.5.0), randomForest, mice, nnet

**Suggests** MASS, caTools

**Description** A set of functions to support Xiong K, 'Roughened Random Forests for Binary Classification' (2014). The functions include RRFA, RRFB, RRFC1-
RRFC7, RRFD and RRFE. RRFB and RRFC6 are usually recommended. RRFB is much faster than RRFC6.

**License** GPL (>= 2)

**URL** http://pqdtopen.proquest.com/pqdtopen/doc/1554346548.html?FMT=ABS

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-01-29 11:08:31

## R topics documented:

---

`mfix` *Four single imputation methods*

---

## Description

Four single imputation methods can be implemented by this function, including median/mode imputation (median imputation on continuous variables and mode imputation on categorical variables), mean/mode imputation (mean imputation on continuous variables and mode imputation on categorical variables), minimum-value/mode imputation (minimum-value imputation on continuous variables and mode imputation on categorical variables) and maximum-value/mode imputation (maximum-value imputation on continuous variables and mode imputation on categorical variables). When `mmmm`=1, this function is the same as `na.roughfix` from R package `randomForest`.

## Usage

```
mfix(x, mmmm)
```

## Arguments

| | |
|---|---|
| `x` | A dataset with missing values. |
| `mmmm` | Its value is used to select from four different single imputation methods. `mmmm`=1 refers to median/mode imputation; `mmmm`=2 refers to mean/mode imputation; `mmmm`=3 refers to minimum-value/mode imputation; `mmmm`=4 refers to maximum-value/mode imputation. |

## Value

Imputed dataset

## Author(s)

Kuangnan Xiong

## References

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

## See Also

`rrfa`, `rrfb`, `rrfc1`, `rrfc2`, `rrfc3`, `rrfc4`,`rrfc5`, `rrfc6`, `rrfc7`, `rrfd`, `rrfe`

## Examples

```
dat=data.frame(continuous=c(1,2,3,4,5),categorical=c('a','a','a','a','b'))
dat[2,]=NA
dat
summary(dat)
mfix(dat,1)[2,]  #median/mode imputation
mfix(dat,2)[2,]  #mean/mode imputation
mfix(dat,3)[2,]  #minimum-value/mode imputation
mfix(dat,4)[2,]  #maximum-value/mode imputation
#
```

---

`rrfa` *Roughenen Random Forests - A (RRFA)*

---

## Description

RRFA algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of both training and testing datasets.

2.Impute the missing data by median imputation for continuous variables and mode imputation for categorical variables.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the imputed testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfa(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfb, rrfc1, rrfc2, rrfc3, rrfc4, rrfc5, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.4
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFA
pred.rrfa=rrfa(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfa$pred,1,mean),dat[te,yvar]))
}}
```

---

# rrfb *Roughenen Random Forests - B (RRFB)*

---

## Description

RRFB algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Impute the missing data by median imputation for continuous variables and mode imputation for categorical variables.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfb(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

`rrfa`, `rrfc1`, `rrfc2`, `rrfc3`, `rrfc4`, `rrfc5`, `rrfc6`, `rrfc7`, `rrfd`, `rrfe`

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
```

```
tr=1:200
te=201:532
mispct=0.7
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFB
pred.rrfb=rrfb(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfb$pred,1,mean),dat[te,yvar]))

}}

#
```

---

`rrfc1` *Roughenen Random Forests - C1 (RRFC1)*

---

## Description

RRFC1 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Impute the missing values in a continuous variable by its mean value and impute the missing values in a categorical variable by its mode value (Mean/mode imputation).

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc1(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc2, rrfc3, rrfc4, rrfc5, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.4
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC1
pred.rrfc1=rrfc1(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc1$pred,1,mean),dat[te,yvar]))
}}
#
```

rrfc2 *Roughenen Random Forests - C2 (RRFC2)*

## Description

RRFC2 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Impute the missing values in a continuous variable by its minimum value and impute the missing values in a categorical variable by its mode value (Minimum-value /mode imputation).

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc2(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1,rrfc3, rrfc4, rrfc5, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.2
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC2
pred.rrfc2=rrfc2(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc2$pred,1,mean),dat[te,yvar]))

}}
```

---

# rrfc3 *Roughenen Random Forests - C3 (RRFC3)*

---

## Description

RRFC3 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Impute the missing values in a continuous variable by its maximum value and impute the missing values in a categorical variable by its mode value (Maximum-value /mode imputation).

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc3(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| dat | A data frame containing both training and testing datasets |
| yvar | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| tr | Row numbers of all training data |
| te | Row numbers of all testing data |
| mispct | Rate of missing data, ranging from 0 to 1 |
| number.trees | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc4, rrfc5, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.4
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))
```

```
#AUC value for the testing dataset based on RRFC3
pred.rrfc3=rrfc3(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc3$pred,1,mean),dat[te,yvar]))
}}
#
```

---

`rrfc4` *Roughenen Random Forests - C4 (RRFC4)*

---

## Description

RRFC4 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Hot-deck imputation for all variables. For each variable, observed values are randomly selected to impute missing values.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc4(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc3, rrfc5, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.4
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC4
pred.rrfc4=rrfc4(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc4$pred,1,mean),dat[te,yvar]))
}}
#
```

---

# rrfc5 *Roughenen Random Forests - C5 (RRFC5)*

---

## Description

RRFC5 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Regression-based imputation for all variables. Linear regression is used to impute continuous variables. Logistic regression is used to impute binary variables. And multinomial logistic regression is used to impute categorical variables with three or more categories.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc5(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| dat | A data frame containing both training and testing datasets |
| yvar | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| tr | Row numbers of all training data |
| te | Row numbers of all testing data |
| mispct | Rate of missing data, ranging from 0 to 1 |
| number.trees | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc3, rrfc4, rrfc6, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.1
yvar=ncol(dat)
```

```
#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC5
pred.rrfc5=rrfc5(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc5$pred,1,mean),dat[te,yvar]))
}}
#
```

---

`rrfc6` *Roughenen Random Forests - C6 (RRFC6)*

---

## Description

RRFC6 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Multiple imputation by chained equation (implemented by mice function in R package mice) is used to produce the imputed dataset.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc6(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Van Buuren, S. & Groothuis-Oudshoorn, K., 2011. mice: Multivariate Imputation by Chained Equations in R.. Journal of Statistical Software, 45(3), pp. 1-67.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc3, rrfc4, rrfc5, rrfc7, rrfd, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){


dat=rbind(Pima.tr,Pima.te)
number.trees=5
#number.trees=500
tr=1:200
te=201:532
mispct=0.5
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC6
pred.rrfc6=rrfc6(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc6$pred,1,mean),dat[te,yvar]))
}}
#
```

---

rrfc7 *Roughenen Random Forests - C7 (RRFC7)*

---

## Description

RRFC7 algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Missing data is imputed based on proximity from random forests (implemented by rfImpute function in R package randomForest).

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfc7(dat, yvar = ncol(dat), tr, te, mispct, number.trees)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |
| `te` | Row numbers of all testing data |
| `mispct` | Rate of missing data, ranging from 0 to 1 |
| `number.trees` | Number of trees used in roughened random forests |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

`rrfa`, `rrfb`, `rrfc1`, `rrfc2`, `rrfc3`, `rrfc4`, `rrfc5`, `rrfc6`, `rrfd`, `rrfe`

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=5
#number.trees=500
tr=1:200
te=201:532
mispct=0.5
yvar=ncol(dat)

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFC7
pred.rrfc7=rrfc7(dat,yvar,tr,te,mispct,number.trees)
print(colAUC(apply(pred.rrfc7$pred,1,mean),dat[te,yvar]))
}}
#
```

---

`rrfd` *Roughenen Random Forests - D (RRFD)*

---

## Description

RRFD algorithm

1.Impose missing values under the mechanism of missing completely at random on all covariates of the training dataset.

2.Impute the missing data by median imputation for continuous variables and mode imputation for categorical variables.

3.Build one tree with a certain m value using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for `number.trees` times.

## Usage

```
rrfd(dat, yvar = ncol(dat), tr, te, mispct, number.trees, m)
```

## Arguments

| | |
|---|---|
| `dat` | A data frame containing both training and testing datasets |
| `yvar` | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| `tr` | Row numbers of all training data |

| | |
|---|---|
| te | Row numbers of all testing data |
| mispct | Rate of missing data, ranging from 0 to 1 |
| number.trees | Number of trees used in roughened random forests |
| m | The number of covariates selected at each tree node |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc3, rrfc4, rrfc5, rrfc6, rrfc7, rrfe

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.7
yvar=ncol(dat)
m=5

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFD
pred.rrfd=rrfd(dat,yvar,tr,te,mispct,number.trees,m)
print(colAUC(apply(pred.rrfd$pred,1,mean),dat[te,yvar]))
}}
#
```

rrfe *Roughenen Random Forests - E (RRFE)*

## Description

RRFE algorithm

1.Impose missing values under the mechanism of missing completely at random on selected covariates of the training dataset, and the probability that missing data is imposed on a certain variable is based on the k -th power of its relative importance. The relative importance of a variable is defined as its variable importance divided by the maximum variable importance among all available covariates according to the original random forests. Here, the variable importance is based on the mean decrease in node impurity.

2.Impute the missing data by median imputation for continuous variables and mode imputation for categorical variables.

3.Build one tree in random forests using the above imputed training dataset, and then use it to predict the binary outcomes in the original testing dataset.

4.Repeat 1 to 3 for number.trees times.

## Usage

```
rrfe(dat, yvar = ncol(dat), tr, te, mispct, number.trees, k)
```

## Arguments

| | |
|---|---|
| dat | A data frame containing both training and testing datasets |
| yvar | The column number of the binary outcome variable, a factor variable. The default value is set as ncol(dat) |
| tr | Row numbers of all training data |
| te | Row numbers of all testing data |
| mispct | Rate of missing data, ranging from 0 to 1 |
| number.trees | Number of trees used in roughened random forests |
| k | The k-th power of a variable's relative importance is used for deciding the probability of imposing missing data on this variable |

## Value

A prediction matrix. Each column shows the predicted values by a single tree. Each row is sequentially associated with the observations in the testing dataset. Each cell value is either 0 or 1.

## Author(s)

Kuangnan Xiong

## References

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Liaw, A. & Wiener, M., 2002. Classification and regression by randomForest. R News, 2(3), pp. 18-22.

Xiong, Kuangnan. "Roughened Random Forests for Binary Classification." PhD diss., State University of New York at Albany, 2014.

## See Also

rrfa, rrfb, rrfc1, rrfc2, rrfc3, rrfc4, rrfc5, rrfc6, rrfc7, rrfd

## Examples

```
if(require(MASS)){
if(require(caTools)){

dat=rbind(Pima.tr,Pima.te)
number.trees=50
#number.trees=500
tr=1:200
te=201:532
mispct=0.5
yvar=ncol(dat)
k=2

#AUC value for the testing dataset based on the original random forests
rf=randomForest(dat[tr,-yvar],dat[tr,yvar],dat[te,-yvar],ntree=number.trees)
print(colAUC(rf$test$votes[,2],dat[te,yvar]))

#AUC value for the testing dataset based on RRFE
pred.rrfe=rrfe(dat,yvar,tr,te,mispct,number.trees,k)
print(colAUC(apply(pred.rrfe$pred,1,mean),dat[te,yvar]))
}}
#
```

# Index