# Package 'Sstack'

May 1, 2018

**Type** Package

**Title** Bootstrap Stacking of Random Forest Models for Heterogeneous Data

**Version** 1.0.1

**Author** Kevin Matlock, Raziur Rahman

**Maintainer** Kevin Matlock <kevin.matlock@gmail.com>

**Description** Generates and predicts a set of linearly stacked Random Forest models using bootstrap sampling. Individual datasets may be heterogeneous (not all samples have full sets of features). Contains support for parallelization but the user should register their cores before running. This is an extension of the method found in Matlock (2018) <doi:10.1186/s12859-018-2060-2>.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.0.1

**Depends** R (>= 2.10)

**Imports** randomForest, foreach, dplyr, parallel, doParallel

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-05-01 15:38:50 UTC

## R topics documented:

BSHorizontalStack *Horizontal stacking Random Forest models.*

## Description

Generate the weights for a horizontally stacked set of Random Forest (RF) models given a set of heterogeneous datasets. For horizontal stacking some subset of samples must be common among all datasets. Subfunction of BSstack but can be used stand-alone.

## Usage

```
BSHorizontalStack(T = 100, mtry = NA, nodesize = 5, iter = 100,
  Xn = NULL, ECHO = TRUE, Cf = NULL, Y, X1, X2, ...)
```

## Arguments

| | |
|---|---|
| T | Number of trees for the individual RF models. (int) |
| mtry | Number of variables available for splitting at each tree node. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. |
| nodesize | Minimum size of terminal nodes. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. By default all models use 5. |
| iter | The number of time to bootstrap sample the data. (int) |
| Xn | List containing each dataset to be stacked. If not supplied will be generated from X1, X2, ... |
| ECHO | Bool, enable to provide output to the user in terms of overlapping samples and runtime for CV. |
| Cf | Character vector listing set of samples common among all given datasets. If not found will generate on it's own. |
| Y | Nsample x 1 data table of responses for ALL samples. Must have matching rownames with each individual dataset. |
| X1 | Data table of first dataset to be stacked. Rownames should be contained within Y. |
| X2 | Data table of second dataset to be stacked. Rownames should be contained within Y. |
| ... | Further data tables, X3, X4, ..., Xl. |

## Details

Required Packages: dplyr, randomForest, foreach

## Value

Weights and offsets for each individual RF model.

BSstack *Bootstrap Stacking model builder.*

## Description

Creates a bootstrapped linear stacked set of Random Forest (RF) models given a set of heterogeneous datasets.

## Usage

```
BSstack(T = 50, mtry = NULL, nodesize = 5, iter = 25, CV = NA,
  Xn = NULL, ECHO = TRUE, Y, X1, X2, ...)
```

## Arguments

| | |
|---|---|
| T | Number of trees for the individual RF models. (int) |
| mtry | Number of variables available for splitting at each tree node. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. If NA then for each model it will be set to Nfeats/3 |
| nodesize | Minimum size of terminal nodes. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. By default all models use 5. |
| iter | The number of time to bootstrap sample the data. (int) |
| CV | Cross validation (CV) to measure mean-absolute error and correlation coefficient, if NA (default) no CV is performed. Otherwise the value gives the number of folds for CV. If CV<2 then leave-one-out CV is performed. CV is performed utilizing the samples that have full record. |
| Xn | List containing each dataset to be stacked. If not supplied will be generated from X1, X2, ... |
| ECHO | Bool, enable to provide output to the user in terms of overlapping samples and runtime for CV. |
| Y | Nsample x 1 data table of responses for ALL samples. Must have matching rownames with each individual dataset. |
| X1 | Data table of first dataset to be stacked. Rownames should be contained within Y. |
| X2 | Data table of second dataset to be stacked. Rownames should be contained within Y. |
| ... | Further data tables, X3, X4, ..., Xl. |

## Details

Required Packages: dplyr, randomForest, foreach

## Value

If CV != null : A list composed of: [1] List containing [1] individual RF models, [2] Nstack +1 weights and [3] feature names for full record samples. This argument is what is used for BSstack_predict [2] Mean-absolute error calculated using cross validation (scalar). [3] Pearson correlation coefficient between actual and predicted values through cross validation (scalar -1<=r<=1). [4] Individual weights calculate for each fold (CV x Nstack+1 matrix). [5] Out of fold predictions for the overlaping samples. [6] Actual values for the overlaping samples. If CV > 1 : Also [7] The fold assignments for the overlapping samples. If CV = null : Only [1] is returned.

## Examples

```
library(Sstack)
library(doParallel)
data(StackData)

AUC=StackData[[1]]
GE=StackData[[2]]
RPPA=StackData[[3]]

X1 <- GE[1:400,1:75]
X2 <- GE[200:400,76:150]
Xt <- GE[401:487,]

set.seed(1)

cl <- makeCluster(2)
registerDoParallel(cl)

Hbs <- BSstack(T = 25, iter = 20, Y = AUC, X1 = X1, X2 = X2)

stopCluster(cl)

Yp <- BSstack_predict(Hbs[[1]],Xt)

maeH1 <- mean(abs(AUC[401:487,]-Yp[,1]))
maeH2 <- mean(abs(AUC[401:487,]-Yp[,2]))
maeHs <- mean(abs(AUC[401:487,]-Yp[,3]))
```

---

# BSstack_predict *Predict using a set of Stacked Random Forest models.*

---

## Description

Gives predictions for a linear bootstrapped stacked Random Forest predictors. Gives the predictions of each individual model as well as the linearlly combined predictions.

## Usage

```
BSstack_predict(BSmodel, Xi)
```

### Arguments

| | |
|---|---|
| BSmodel | List containing the individual Random Forest models, their weights, and feature names. Generated using BSstack function. |
| Xi | NxM datatable containing input features to be predicted. Must contain all features used in the individual RF models. |

### Details

Required Packages: randomForest

### Value

NxL+1 matrix where L is the number of individual RF models. Predictions for the ith RF model is found in the ith column of this matrix while predictions for the stacked model is in the final column.

---

## BSVerticalStack *Vertical stacking Random Forest models.*

---

### Description

Generate the weights for a vertically stacked set of Random Forest (RF) models given a set of heterogeneous datasets. For vertical stacking at least one dataset must contain full record (all features). Subfunction of BSstack but can be used stand-alone.

### Usage

```
BSVerticalStack(T = 50, mtry = NULL, nodesize = 5, iter = 25,
  ECHO = TRUE, Y, Xfull = NULL, Xn = NULL, X1, X2, ...)
```

### Arguments

| | |
|---|---|
| T | Number of trees for the individual RF models. (int) |
| mtry | Number of variables available for splitting at each tree node. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. |
| nodesize | Minimum size of terminal nodes. If a scalar is given then all models use the given values. If a 1D array is given then each individual model uses the given value. By default all models use 5. |
| iter | The number of time to bootstrap sample the data. (int) |
| ECHO | Bool, enable to provide output to the user in terms of overlapping samples and runtime for CV. |
| Y | Nsample x 1 data table of responses for ALL samples. Must have matching rownames with each individual dataset. |
| Xfull | Data table containing samples with full record. Used for generating the weights. Will attempt to find if not given. |

| | |
|---|---|
| Xn | List containing each dataset to be stacked. If not supplied will be generated from X1, X2, ... |
| X1 | Data table of first dataset to be stacked. Rownames should be contained within Y. |
| X2 | Data table of second dataset to be stacked. Rownames should be contained within Y. |
| ... | Further data tables, X3, X4, ..., Xl. |

## Details

Required Packages: dplyr, randomForest, foreach

## Value

Weights and offsets for each individual RF model.

---

# StackData *Sample Stack Data*

---

## Description

A demo dataset containing Gene Expression (GE), Reverse Phase Protein Array (RPPA) and drug sensitivity measure (AUC) for cancer cell lines that have been tested for the drug 17-AAG. Used to demonstrate the benefits of stacking by building an integrated model that will more effectively predict the AUC values given both GE and RPPA data.

## Usage

```
StackData
```

## Format

A list of 3 variables containing Gene Expression (GE), Reverse Phase Protein Array (RPPA) and drug sensitivity measure (AUC):

**GE** Gene Expression (GE) data corresponding to cell lines tested by the drug 17-AAG taken from the Cancer Cell Line Encyclopedia. Relieff has been run to select the top 150 genes.

**RPPA** Reverse Phase Protein Array (RPPA) data corresponding to cell lines tested by the drug 17-AAG taken from the MD Anderson Cell Lines Project. Relieff has been run to select the top 150 proteins.

**AUC** Area under the Dose-Responce Curve (AUC) corresponding to the drug 17-AAG applied to each cell line. Taken from the Cancer Cell Line Encyclopedia.

## Source

http://bioinformatics.mdanderson.org/main/MCLP:Overview https://portals.broadinstitute.org/ccle

## References

Barretina,J. and et al. (2012) The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. Nature, 483 (7391), 603–607. Li, J. and et al. (2016) Characterization of Human Cancer Cell Lines by Reverse-Phase Protein Arrays. Cancer Cell (In Press).

# Index