Package: optRF (via r-universe)

November 23, 2024

Type Package

Title Optimising Random Forest Stability Through Selection of the Optimal Number of Trees

Version 1.0.1

Maintainer Thomas Martin Lange <thomas.lange@uni-goettingen.de>

Description Calculating the stability of random forest with certain numbers of trees. The non-linear relationship between stability and numbers of trees is described using a logistic regression model and used to estimate the optimal number of trees.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 4.1.2), minpack.lm (>= 1.2-4), ranger (>= 0.16.0), irr (>= 0.84.1)

Imports graphics, methods, stats

Suggests covr, spelling, testthat

NeedsCompilation no

Author Thomas Martin Lange [cre, aut] (<https://orcid.org/0000-0003-4351-7950>), Felix Heinrich [ctb] (<https://orcid.org/0000-0002-6093-8522>)

Repository CRAN

Date/Publication 2024-09-23 14:10:02 UTC

Contents

estimate_numtrees *Estimate the required number of trees*

Description

Estimate the number of trees required to achieve certain stability of random forest

Usage

```
estimate_numtrees(
  optRF_object,
  measure = c("selection", "importance", "prediction"),
  for_stability = 0.95
)
```

Arguments

`optRF_object` An optRF_object, either the result from the opt_importance or the opt_prediction function.

`measure` A character string indicating which stability measure is to be analysed. One of "selection" (default, analyses selection stability), "prediction" (analyses prediction stability) or "importance" (analyses variable importance stability).

`for_stability` Either a single stability value or a vector containing multiple stability values for which the number of trees should be estimated.

Value

A data frame summarising the estimated stability and run time in seconds for the given num.trees values.

Examples

```
data(SNPdata)
set.seed(123)
result_optpred = opt_prediction(y = SNPdata[,1], X=SNPdata[,-1]) # optimise random forest
estimate_numtrees(result_optpred, measure="prediction", for_stability=0.95)
```

estimate_stability *Estimate the stability of random forest*

Description

Estimate the stability of random forest with certain numbers of trees

Usage

```
estimate_stability(
  optRF_object,
  with_num.trees = c(1000, 5000, 10000, 50000, 1e+05)
)
```

Arguments

`optRF_object`	An optRF_object, either the result from the opt_importance or the opt_prediction function.
`with_num.trees`	Either a single num.trees value or a vector containing multiple num.trees values for which the stability should be estimated.

Value

A data frame summarising the estimated stability and run time in seconds for the given num.trees values.

Examples

```
data(SNPdata)
set.seed(123)
result_optpred = opt_prediction(y = SNPdata[,1], X=SNPdata[,-1]) # optimise random forest
estimate_stability(result_optpred, with_num.trees=c(1000, 5000, 10000, 50000, 100000))
```

opt_importance *Optimise random forest for estimation of variable importance*

Description

Optimising random forest for estimating the importance of variables by calculating the variable importance stability with certain numbers of trees

Usage

```
opt_importance(
  y = NULL,
  X = NULL,
  number.repetitions = 10,
  alpha = 0.05,
  num.trees_values = c(250, 500, 750, 1000, 2000),
  visualisation = c("none", "importance", "selection"),
  recommendation = c("importance", "selection", "none"),
  rec.thresh = 1e-06,
  round.recommendation = c("thousand", "hundred", "ten", "none"),
  ...
)
```

Arguments

`y` A vector containing the response variable.

`X` A data frame containing the explanatory variables. The number of rows must be equal to the number of elements in y.

`number.repetitions` Number of repetitions of random forest to estimate the variable importance stability.

`alpha` The amount of most important variables to be selected based on their estimated variable importance. If < 1, alpha will be considered the relative amount of variables in the data set.

`num.trees_values` A vector containing the numbers of trees to be analysed. If not specified, 250, 500, 750, 1000, and 2000 trees will be analysed.

`visualisation` Can be set to "importance" to draw a plot of the variable importance stability or to "selection" to draw a plot of the selection stability for the numbers of trees to be analysed.

`recommendation` If set to "importance" (default) or "selection", a recommendation will be given based on optimised variable importance or selection stability. If set to be "none", the function will analyse the stability of random forest with the inserted numbers of trees without giving a recommendation.

`rec.thresh` If the number of trees leads to an increase of stability smaller or equal to the value specified, this number of trees will be recommended. Default is 1e-6.

`round.recommendation` Setting to what number the recommended number of trees should be rounded to. Options: "none", "ten", "hundred", "thousand".

`...` Any other argument from the ranger package.

Value

An opt_importance_object containing the recommended number of trees, based on which measure the recommendation was given (importance or selection), a matrix summarising the estimated stability and computation time of a random forest with the recommended numbers of trees, a matrix containing the calculated stability and computation time for the analysed numbers of trees, and the parameters used to model the relationship between stability and numbers of trees.

Examples

```
data(SNPdata)
set.seed(123)
result_optimp = opt_importance(y = SNPdata[,1], X=SNPdata[,-1]) # optimise random forest
summary(result_optimp)
```

opt_prediction *Optimise random forest for prediction*

Description

Optimising random forest predictions by calculating the prediction stability with certain numbers of trees

Usage

```
opt_prediction(
  y = NULL,
  X = NULL,
  X_Test = NULL,
  number.repetitions = 10,
  alpha = 0.15,
  num.trees_values = c(250, 500, 750, 1000, 2000),
  visualisation = c("none", "prediction", "selection"),
  select_for = c("high", "low", "zero"),
  recommendation = c("prediction", "selection", "none"),
  rec.thresh = 1e-06,
  round.recommendation = c("thousand", "hundred", "ten", "none"),
  ...
)
```

Arguments

`y` A vector containing the response variable in the training data set.

`X` A data frame containing the explanatory variables in the training data set. The number of rows must be equal to the number of elements in y.

`X_Test` A data frame containing the explanatory variables of the test data set. If not entered, a test data set will be randomly generated.

`number.repetitions`
Number of repetitions of random forest to estimate the prediction stability.

`alpha` The number of best individuals to be selected in the test data set based on their predicted response values. If < 1, alpha will be considered to be the relative amount of individuals in the test data set.

`num.trees_values` A vector containing the numbers of trees to be analysed. If not specified, 250, 500, 750, 1000, and 2000 trees will be analysed.

`visualisation` Can be set to "prediction" to draw a plot of the prediction stability or "selection" to draw a plot of the selection stability for the numbers of trees to be analysed.

`select_for` What should be selected? In random forest classification, this must be set to the value of the desired class. In random forest regression, this can be set as "high" (default) to select the individuals with the highest predicted value, "low" to select the individuals with the lowest predicted value, or "zero" to select the individuals which predicted value is closest to zero.

`recommendation` If set to "prediction" (default) or "selection", a recommendation will be given based on optimised prediction or selection stability. If set to be "none", the function will analyse the stability of random forest with the inserted numbers of trees without giving a recommendation.

`rec.thresh` If the number of trees leads to an increase of stability smaller or equal to the value specified, this number of trees will be recommended. Default is 1e-6.

`round.recommendation` Setting to what number the recommended number of trees should be rounded to. Options: "none", "ten", "hundred", "thousand" (default).

`...` Any other argument from the ranger function.

Value

An opt_prediction_object containing the recommended number of trees, based on which measure the recommendation was given (prediction or selection), a matrix summarising the estimated stability and computation time of a random forest with the recommended numbers of trees, a matrix containing the calculated stability and computation time for the analysed numbers of trees, and the parameters used to model the relationship between stability and numbers of trees.

Examples

```
data(SNPdata)
set.seed(123)
result_optpred = opt_prediction(y = SNPdata[,1], X=SNPdata[,-1]) # optimise random forest
summary(result_optpred)
```

plot_stability *Plot random forest stability*

Description

Plot the estimated stability of random forest against certain numbers of trees

Usage

```
plot_stability(
  optRF_object,
  measure = c("selection", "importance", "prediction"),
  from = 0,
  to = 1e+05,
  add.recommendation = TRUE,
  add = FALSE,
  ...
)
```

Arguments

`optRF_object` An optRF_object, either the result from the opt_importance or the opt_prediction function.

`measure` A character string indicating which stability measure is to be plotted. One of "selection" (default, visualises selection stability), "prediction" (visualises prediction stability) or "importance" (visualises variable importance stability).

`from` Smallest num.trees value to be plotted.

`to` Greatest num.trees value to be plotted.

`add.recommendation` When set as TRUE, if a recommendation was stated within the opt_prediction or opt_importance function, the recommended num.trees value as well as the expected random forest stability will be highlighted in the graph

`add` If FALSE, a new plot will be created, if TRUE, the graph will be added to an existing plot.

`...` Any other arguments from the plot function.

Value

A plot showing the estimated stability of random forest for the given num.trees values.

Examples

```
data(SNPdata)
set.seed(123)
result_optpred = opt_prediction(y = SNPdata[,1], X=SNPdata[,-1]) # optimise random forest
plot_stability(result_optpred, measure = "prediction", add.recommendation = TRUE, add=FALSE)
plot_stability(result_optpred, measure = "selection", add.recommendation = FALSE, add=TRUE)
```

SNPdata *Simulated data of wheat yield and genomic markers*

Description

Data set containing simulated data of wheat yield in g/m^2 of 250 wheat lines and 7,500 SNP markers being coded as 0 for homozygous form of the major allele and 2 for homozygous form of the minor allele.

Usage

```
data(SNPdata)
```

Format

An object of class "data.frame"

yield Simulated wheat yield in g/m^2

SNP_0001 to SNP_7500 Simulated values for 7,500 single nucleotide polymorphism (SNP) markers

References

This artificial data set was created for the optRF package.

Examples

```
data(SNPdata)
SNPdata[1:5,1:5]
```

Index