Package 'abcrf'

September 4, 2017

Type Package

Title Approximate Bayesian Computation via Random Forests

Version 1.7

Date 2017-08-31

Description Performs Approximate Bayesian Computation (ABC) model choice and parameter inference via random forests.

License GPL (>= 2)

LazyData TRUE

Imports readr, MASS, matrixStats, ranger, parallel, stringr, Rcpp (>= 0.11.2)

LinkingTo Rcpp, RcppArmadillo

Depends R(>= 3.1)

NeedsCompilation yes

Maintainer Jean-Michel Marin <jean-michel.marin@umontpellier.fr>

Repository CRAN

Author Jean-Michel Marin [aut, cre],
Louis Raynal [aut],
Pierre Pudlo [aut],
Christian P. Robert [ctb],
Arnaud Estoup [ctb]

Date/Publication 2017-09-04 12:56:56 UTC

R topics documented:

`abcrf` *Create an ABC-RF object: a classification random forest from a reference table towards performing an ABC model choice*

Description

`abcrf` constructs a random forest from a reference table towards performing an ABC model choice. Basically, the reference table (i.e. the dataset that will be treated with the present package) includes a column with the index of the models to be compared and additional columns corresponding to the values of the simulated summary statistics.

Usage

```
## S3 method for class 'formula'
abcrf(formula, data, group=list(), lda=TRUE, ntree=500, sampsize=min(1e5, nrow(data)),
paral=FALSE, ncores= if(paral) max(detectCores()-1,1) else 1, ...)
```

Arguments

`formula`	a formula: left of ~, variable representing the model index; right of ~, summary statistics of the reference table.
`data`	a data frame containing the reference table.
`group`	a list containing groups (at least 2) of model(s) on which the model choice will be performed. This is not necessarily a partition, one or more models can be excluded from the elements of the list and by default no grouping is done.
`lda`	should LDA scores be added to the list of summary statistics?
`ntree`	number of trees to grow in the forest, by default 500 trees.
`sampsize`	size of the sample from the reference table to grow a tree of the classification forest, by default the minimum between the number of elements of the reference table and 100,000.
`paral`	a boolean that indicates if the calculations of the classification random forest (forest used to assign a model to the observed dataset) should be parallelized.
`ncores`	the number of CPU cores to use. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and `detectCores` does not detect the number of CPU cores with success then 1 core is used.
`...`	additional arguments to be passed on to `ranger` used to construct the classification random forest that preditcs the selected model.

Value

An object of class `abcrf`, which is a list with the following components:

`call`	the original call to `abcrf`,
`lda`	a boolean indicating if LDA scores have been added to the list of summary statistics,
`formula`	the formula used to construct the classification random forest,
`group`	a list contining the groups of model(s) used. This list is empty if no grouping has been performed,
`model.rf`	an object of class `randomForest` containing the trained forest with the reference table,
`model.lda`	an object of class `lda` containing the Linear Discriminant Analysis based on the reference table,
`prior.err`	prior error rates of model selection on the reference table, estimated with the "out-of-bag" error of the forest.

References

Pudlo, P., Marin, J.-M., Estoup, A., Cornuet, J.-M., Gautier, M. and Robert, C.P. (2016) *Reliable ABC model choice via random forests* Bioinformatics http://bioinformatics.oxfordjournals.org/content/32/6/859

See Also

`plot.abcrf`, `predict.abcrf`, `err.abcrf`, `ranger`

Examples

```
data(snp)
modindex <- snp$modindex[1:500]
sumsta <- snp$sumsta[1:500,]
data1 <- data.frame(modindex, sumsta)
model.rf1 <- abcrf(modindex~., data = data1, ntree=100)
model.rf1
model.rf2 <- abcrf(modindex~., data = data1, group = list(c("1","2"),"3"), ntree=100)
model.rf2
```

---

covRegAbcrf *Predict posterior covariance between two parameters for new data using two reg-ABC-RF objects*

Description

Using two reg-ABC-RF objects constructed on the same reference table for two different response variables, this function predicts the posterior covariance between those two response variables, given a new dataset of summaries.

Usage

```
## S3 method for class 'regAbcrf'
covRegAbcrf(regForest1, regForest2, obs, training1, training2,
ntree=500, mtry=max(floor((dim(training1)[2]-1)/3), 1),
sampsize=min(1e5, dim(training1)[1]), paral = FALSE, ncores =
if(paral) max(detectCores()-1,1) else 1,
paral.predict = FALSE, ncores.predict =
if(paral.predict) max(detectCores()-1,1) else 1, ... )
```

Arguments

regForest1, regForest2	regAbcrf objects.
obs	a data frame containing the summary statistics of the observed data sets.
training1, training2	data frames containing the reference table respectively used to train the regAbcrf objects regForest1 and regForest2.
ntree	number of trees to grow in the forest, by default equal to 500 trees.
mtry	Number of variables to possibly split at in each node for the regression random forest. Default is the minimum between 1 and the number of variables divided by 3.
sampsize	size of the sample from the reference table used to grow a tree of the regression forest, by default the minimum between the number of elements of the reference table and 100,000.
paral	a boolean that indicates whether or not the calculations of the regression random forest (forest used to predict a response from the observed dataset) should be parallelized.
ncores	the number of CPU cores to use. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
paral.predict	a boolean that indicates if random forests predictions should be parallelized.
ncores.predict	the number of CPU cores to use for the regression random forest predictions. If paral.predict=TRUE, it is used the number of CPU cores minus 1. If ncores.predict is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
...	additional arguments to be passed on to ranger used to construct the regression random forest that predicts posterior covariance.

Value

covRegAbcrf returns predicted posterior covariances between response variables of two reg-ABC-RF objects, for a new data set.

References

Marin, J.-M., Raynal, L., Pudlo, P., Ribatet, M. and Robert, C.P. (2016) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

regAbcrf, predict.regAbcrf, err.regAbcrf, plot.regAbcrf, ranger, densityPlot

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
N1 <- snp$param$N1[modindex == "3"]
N1 <- N1[1:500]
data3 <- data.frame(N1, sumsta)
model.rf.N1 <- regAbcrf(N1~., data3, ntree=100)
data(snp.obs)
covRegAbcrf(model.rf.r, model.rf.N1, snp.obs,
            data2, data3 , ntree=100)
```

---

densityPlot *Plot the posterior density given a new summary statistic*

Description

Given a reg-ABC-RF object and a new value of the summary statistics, densityPlot gives the corresponding posterior density plot of the parameter.

Usage

```
## S3 method for class 'regAbcrf'
densityPlot(object, obs, training,
main="Posterior density", log="", paral=FALSE,
ncores= if(paral) max(detectCores()-1,1) else 1, ...)
```

Arguments

object	a regAbcrf object.
obs	a data frame containing the summary statistics of the observed data sets.
training	the data frame containing the reference table used to train the regAbcrf object.
main	main title to be used for the posterior density plot.
log	a character string which contains "x" if the x axis is to be logarithmic, "y" if the y axis is to be logarithmic and "xy" or "yx" if both axes are to be logarithmic. The default value "" implies no logarithmic transfomation.
paral	a boolean that indicates if random forests predictions should be parallelized.

ncores	the number of CPU cores to use for the regression random forest predictions. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
...	additional arguments to be passed on to density, as for example the smoothing bandwidth bw to be used.

References

Marin, J.-M., Raynal, L., Pudlo, P., Ribatet, M. and Robert, C.P. (2016) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

regAbcrf, predict.regAbcrf, err.regAbcrf, covRegAbcrf, ranger, plot.regAbcrf

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
data(snp.obs)
densityPlot(model.rf.r, snp.obs, data2, main = "Posterior density of r")
```

---

err.abcrf — *Calculate and plot for different numbers of tree, the out-of-bag errors associated with an ABC-RF object*

Description

err.abcrf returns out-of-bag errors and plot them.

Usage

```
err.abcrf(object, training, paral=FALSE,
ncores= if(paral) max(detectCores()-1,1) else 1)
```

Arguments

object	an abcrf object.
training	the data frame containing the reference table used to train the abcrf object.
paral	a boolean that indicates if random forests predictions should be parallelized.

ncores — the number of CPU cores to use for the random forest predictions. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.

Value

a matrix with 2 columns: the number of trees and the out-of-bag errors. Errors are computed from 40 trees to the total number.

References

Pudlo, P., Marin, J.-M., Estoup, A., Cornuet, J.-M., Gautier, M. and Robert, C.P. (2016) *Reliable ABC model choice via random forests* Bioinformatics http://bioinformatics.oxfordjournals.org/content/32/6/859

See Also

abcrf, predict.abcrf, plot.abcrf

Examples

```
data(snp)
modindex <- snp$modindex[1:500]
sumsta <- snp$sumsta[1:500,]
data1 <- data.frame(modindex, sumsta)
model.rf <- abcrf(modindex~., data1, ntree=100)
err.rf <- err.abcrf(model.rf, data1)
```

---

err.regAbcrf — *Calculate and plot for different numbers of tree, the out-of-bag mean squared errors associated with a REG-ABC-RF object*

Description

err.regAbcrf returns out-of-bag mean squared errors and plot them.

Usage

```
err.regAbcrf(object, training, paral=FALSE,
ncores= if(paral) max(detectCores()-1,1) else 1, what="mean")
```

Arguments

object	a regAbcrf object.
training	the data frame containing the reference table used to train the regAbcrf object.
paral	a boolean that indicates if random forests predictions should be parallelized.

ncores — the number of CPU cores to use for the random forest predictions. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.

what — a string caracter "mean" or "median" indicating if the predictions are computed with mean or median of the response variable.

Value

a matrix with 2 columns: the number of trees and the out-of-bag mean squared errors. NAs might be returned if the number of trees is too low. Errors are computed from 40 trees to the total number.

References

Raynal, L., Marin, J. Pudlo, P., Ribatet, M. Robert, C.P. and Estoup, A. (2017) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

regAbcrf, predict.regAbcrf, plot.regAbcrf, densityPlot, covRegAbcrf, ranger

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
err.regAbcrf(model.rf.r, data2)
```

---

plot.abcrf *Plot of an ABC-RF object*

Description

plot.abcrf provides both a variable importance plot of a model choice ABC-RF object and the projection of the reference table on the LDA axes.

Usage

```
## S3 method for class 'abcrf'
plot(x, training, obs=NULL, n.var=20, pdf=FALSE, xlim=NULL, ...)
```

Arguments

x	an abcrf object.
training	the data frame containing the reference table used to train the abcrf object.
obs	a vector containing the summary statistics of an observed dataset that will be added to the graph of the projected reference table (black star or vertical line).
n.var	number of variables in the variable importance representation.
pdf	a boolean that indicates if a pdf version of the graph(s) should be saved in the current directory.
xlim	range of the abscissa for the variable importance plot.
...	not used.

Note

The graph of the reference table projected on the LD axes is shown only if LD axes has been added to the set of summary statistics in the call of abcrf.

References

Pudlo, P., Marin, J.-M., Estoup, A., Cornuet, J.-M., Gautier, M. and Robert, C.P. (2016) *Reliable ABC model choice via random forests* Bioinformatics http://bioinformatics.oxfordjournals.org/content/32/6/859

See Also

abcrf, predict.abcrf, err.abcrf, variableImpPlot

Examples

```
data(snp)
modindex <- snp$modindex[1:500]
sumsta <- snp$sumsta[1:500,]
data1 <- data.frame(modindex, sumsta)
model.rf <- abcrf(modindex~., data1, ntree=100)
plot(model.rf, data1)
data(snp.obs)
plot(model.rf, data1, obs=snp.obs[1,])
```

---

plot.regAbcrf *Plot of a reg-ABC-RF object*

Description

plot.regAbcrf provides a variable importance plot used to construct the reg-ABC-RF object, as measured by ranger with the argument importance='impurity'.

Usage

```
## S3 method for class 'regAbcrf'
plot(x, n.var=min(30, length(x$model.rf$variable.importance)), xlim=NULL, main=NULL, ...)
```

Arguments

x	a regAbcrf object.
n.var	number of variables in the variable importance representation. The default value is equal to the minimum between 30 and the number of summary statistics.
xlim	range of the abscissa for the variable importance plot.
main	an overall title for the variable importance plot.
...	not used.

References

Raynal, L., Marin, J. Pudlo, P., Ribatet, M. Robert, C.P. and Estoup, A. (2017) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

regAbcrf, predict.regAbcrf, err.regAbcrf, covRegAbcrf, ranger, densityPlot

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
plot(model.rf.r)
```

---

predict.abcrf — *Predict and evaluate the posterior probability of the MAP for new data using an ABC-RF object*

Description

Based on an ABC-RF object this function predicts the best model for new data and evaluate the posterior probability of the MAP.

Usage

```
## S3 method for class 'abcrf'
predict(object, obs, training, ntree = 1000,
sampsize = min(1e5,  object$model.rf$num.samples ),
paral = FALSE, ncores = if(paral) max(detectCores()-1,1) else 1,paral.predict = FALSE,
ncores.predict = if(paral.predict) max(detectCores()-1,1) else 1 , ...)
```

Arguments

object	an abcrf object.
obs	a data frame containing the summary statistics of the observed data sets.
training	the data frame containing the reference table used to train the abcrf object.
ntree	number of trees to grow in the regression forest, by default 1,000 trees.
sampsize	size of the sample from the reference table used to grow a tree of the forest, by default the minimum between the number of elements of the reference table and 100,000.
paral	a boolean that indicates if the calculations of the regression random forest (forest that returns the posterior probability of the selected model) should be parallelized.
ncores	the number of CPU cores to use for the regression random forest construction. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
paral.predict	a boolean that indicates if random forests predictions should be parallelized.
ncores.predict	the number of CPU cores to use for random forest predictions (classification and regression). If paral.predict=TRUE, it is used the number of CPU cores minus 1. If ncores.predict is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
...	additional arguments to be passed on to ranger used to construct the regression random forest that estimates the posterior probability of the selected model.

Value

An object of class abcrfpredict, which is a list with the following components:

allocation	indices of the selected models for each observed data set,
vote	votes for each observed dataset,
post.prob	ABC-RF approximations of the posterior probability of the selected model for each observed dataset.

References

Pudlo, P., Marin, J.-M., Estoup, A., Cornuet, J.-M., Gautier, M. and Robert C.P. (2016) *Reliable ABC model choice via random forests* Bioinformatics http://bioinformatics.oxfordjournals.org/content/32/6/859

See Also

abcrf, plot.abcrf, err.abcrf

Examples

```
data(snp)
modindex <- snp$modindex[1:500]
sumsta <- snp$sumsta[1:500,]
data1 <- data.frame(modindex, sumsta)
model.rf <- abcrf(modindex~., data1, ntree=100)
data(snp.obs)
predict(model.rf, snp.obs, data1, ntree=100)
```

---

predict.regAbcrf *Predict posterior expectation, median, variance and quantiles given a new dataset using a reg-ABC-RF object*

Description

Based on a reg-ABC-RF object this function predicts the posterior expectation, median, variance, quantiles for the corresponding parameter given new dataset.

Usage

```
## S3 method for class 'regAbcrf'
predict(object, obs, training, quantiles=c(0.025,0.975),
paral = FALSE, ncores = if(paral) max(detectCores()-1,1) else 1,
rf.weights = FALSE,...)
```

Arguments

object	a regAbcrf object.
obs	a data frame containing the summary statistics of the observed data sets.
training	the data frame containing the reference table used to train the regAbcrf object.
quantiles	numeric vector of probabilities with values in [0,1]. The default value is equal to c(0.025, 0.975).
paral	a boolean that indicates if random forests predictions should be parallelized.
ncores	the number of CPU cores to use for the regression random forest predictions. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
rf.weights	a boolean that indicates if the random forest weights used to predict quantities of interest should we returned. The default value is FALSE.
...	optional arguments to be passed on to the function predict.ranger.

Value

An object of class `regAbcrfpredict`, which is a list with the following components:

`expectation`	predicted posterior expectation for each oberved data set,
`med`	predicted posterior median for each oberved data set,
`variance`	predicted posterior variance for each observed data set, computed by reusing weights,
`variance.cdf`	predicted posterior variance for each observed data set, computed by approximation of the cumulative distribution function,
`quantiles`	predicted posterior quantiles for each observed data set,
`weights`	a matrix composed of the weights used to predict quantities of interest. Returned if rf.weights is TRUE.

References

Marin, J.-M., Raynal, L., Pudlo, P., Ribatet, M. and Robert, C.P. (2016) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

`regAbcrf`, `plot.regAbcrf`, `err.regAbcrf`, `covRegAbcrf`, `ranger`, `densityPlot`

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
data(snp.obs)
predict(model.rf.r, snp.obs, data2)
```

---

predictOOB *Predict out-of-bag posterior expectation, median, variance, quantiles and error measures using a reg-ABC-RF object*

Description

Based on a reg-ABC-RF object this function predicts the out-of-bag posterior expectation, median, variance, quantiles, mean squared errors, normalized mean absolute errors, credible interval coverage and relative mean range, for the corresponding parameter using the out-of-bag observations of the training data set.

Mean squared errors and normalized mean absolute errors are computed both with mean and median of the response variable.

Memory allocation issues might be encountered when the size of the training data set is large.

Usage

```
## S3 method for class 'regAbcrf'
predictOOB(object, training, quantiles=c(0.025,0.975), paral = FALSE,
ncores = if(paral) max(detectCores()-1,1) else 1, rf.weights = FALSE,...)
```

Arguments

object	a regAbcrf object.
training	the data frame containing the reference table used to train the regAbcrf object.
quantiles	numeric vector of probabilities with values in [0,1]. The default value is equal to c(0.025, 0.975).
paral	a boolean that indicates if random forests predictions should be parallelized.
ncores	the number of CPU cores to use for the regression random forest predictions. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
rf.weights	a boolean that indicates if the random forest weights used to predict quantities of interest should we returned. The default value is FALSE.
...	optional arguments to be passed on to the function predict.ranger.

Value

An object of class regAbcrfOOBpredict, which is a list with the following components:

expectation	predicted posterior expectation for each oberved data set,
med	predicted posterior median for each oberved data set,
variance	predicted posterior variance for each observed data set, computed by reusing weights,
variance.cdf	predicted posterior variance for each observed data set, computed by approximation of the cumulative distribution function,
quantiles	predicted posterior quantiles for each observed data set,
weights	a matrix composed of the weights used to predict quantities of interest. Returned if rf.weights is TRUE,
MSE	mean squared error computed with prediction based on mean of response variable,
NMAE	normalized mean absolute error computed with predictions based on mean of response variable,
med_MSE	mean squared error computed with predictions based on median of response variable,
med_NMAE	normalized mean absolute error with predictions based on median of response variable,
coverage	credible inteval coverage if only two quantiles are of interest, NULL otherwise,
mean.q.range	relative mean range if only two quantiles are of interest, NULL otherwise.

References

Marin, J.-M., Raynal, L., Pudlo, P., Ribatet, M. and Robert, C.P. (2016) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

regAbcrf, predict.regAbcrf, plot.regAbcrf, err.regAbcrf, covRegAbcrf, ranger, densityPlot

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
res <- predictOOB(model.rf.r, data2)
```

---

readRefTable *Read a reference table simulated from DIYABC*

Description

readRefTable reads a reference table simulated from DIYABC thanks to a .bin and a .txt file, respectively containing the reference table and its header.

Usage

```
readRefTable(filename = "reftable.bin", header = "header.txt", N = 0)
```

Arguments

filename	a .bin file from DIYABC containing the reference table.
header	a .txt file containing the header of the reference table in filename.
N	an integer indicating the number of observations to extract from the reference table. The default is 0 indicating that the whole reference table is used. Warning: if N is specified, nrecscen returns is obsolete.

Value

A list with the following components:

nrec	number of individuals of the reference table,
nscen	number of scenarios in the reference table,
nrecscen	number of individuals by scenario,

nparam	number of parameters by scenario,
scenarios	a vector of factor containing the scenario indices,
params	a matrix with the parameters,
stats	a matrix with the summary statistics.

References

Cornuet, J.-M., Pudlo, P., Veyssier, J., Dehne-Garcia, A., Gautier, M., Leblois, R. Marin J.-M. and Estoup A. (2014) DIYABC v2.0: a software to make Approximate Bayesian Computation inferences about population history using *Single Nucleotide Polymorphism, DNA sequence and microsatellite data* Bioinformatics http://bioinformatics.oxfordjournals.org/content/24/23/2713

`regAbcrf` *Create a reg-ABC-RF object: a regression random forest from a reference table aimed out predicting posterior expectation, variance and quantiles for a parameter*

Description

`regAbcrf` constructs a regression random forest from a reference table towards predicting posterior expectations, variances and quantiles of a parameter.

Usage

```
## S3 method for class 'formula'
regAbcrf(formula, data, ntree=500,
mtry=max(floor((dim(data)[2]-1)/3), 1),
sampsize=min(1e5, nrow(data)),paral=FALSE,
ncores= if(paral) max(detectCores()-1,1) else 1, ...)
```

Arguments

formula	a formula: left of ~, variable representing the response variable; right of ~, summary statistics of the reference table.
data	a data frame containing the reference table, composed of response variable (parameter) and summary statistics.
ntree	number of trees to grow in the forest, by default 500 trees.
mtry	Number of variables to possibly split at in each node. Default is the minimum between 1 and the number of variables divided by 3.
sampsize	size of the sample from the reference table used to grow a tree of the regression forest, by default the minimum between the number of elements of the reference table and 100,000.
paral	a boolean that indicates if the calculations of the regression random forest should be parallelized.

ncores	the number of CPU cores to use. If paral=TRUE, it is used the number of CPU cores minus 1. If ncores is not specified and detectCores does not detect the number of CPU cores with success then 1 core is used.
...	additional arguments to be passed on to ranger used to construct the regression random forest that predicts the response variable.

Value

An object of class regAbcrf, which is a list with the following components:

call	the original call to regAbcrf,
formula	the formula used to construct the regression random forest,
model.rf	an object of class ranger containing the trained forest with the reference table.

References

Raynal, L., Marin, J. Pudlo, P., Ribatet, M. Robert, C.P. and Estoup, A. (2017) *ABC random forests for Bayesian parameter inference* https://arxiv.org/pdf/1605.05537.pdf

See Also

plot.regAbcrf, err.regAbcrf, predict.regAbcrf, covRegAbcrf, ranger, densityPlot, predictOOB

Examples

```
data(snp)
modindex <- snp$modindex
sumsta <- snp$sumsta[modindex == "3",]
r <- snp$param$r[modindex == "3"]
r <- r[1:500]
sumsta <- sumsta[1:500,]
data2 <- data.frame(r, sumsta)
model.rf.r <- regAbcrf(r~., data2, ntree=100)
model.rf.r
```

---

snp *A simulated example in population genetics*

Description

The simulated example of population genetics with SNP loci used in Pudlo et al. (2016): snp contains the reference table on which to perform ABC model choice, it also contains the simulated parameters to perform regression random forest. snp.obs contains two pseudo-observed data sets. The first one (favorable) should be easily allocated to a model, while that is not the case for the second one (unfavorable).

Usage

```
data(snp)
data(snp.obs)
```

Format

snp is a list containing an ABC reference table of 10,000 simulations from a Bayesian prior predictive model (see Pudlo et al., 2016, for a description of the model choice issue). The first element, named modindex is a factor containing the model indices, the second element, param, is a data frame with seven simulated parameters. The last element of this list, named sumsta, contains the reference table on which to perform ABC model choice and parameter estimation.

snp.obs is a data frame containing the summary statistics of two pseudo-observed data sets.

Source

Pudlo, P., Marin, J.-M., Estoup, A., Cornuet, J.-M., Gautier, M. and Robert, C.P. (2016) *Reliable ABC model choice via random forests* Bioinformatics http://bioinformatics.oxfordjournals.org/content/32/6/859

Examples

```
data(snp)
data(snp.obs)
```

---

variableImpPlot *Variable importance plot from a random forest*

Description

variableImpPlot provides a dotchart of variable importance as measured by ranger with the argument importance='impurity'.

Usage

```
variableImpPlot(object,
n.var=min(30, length(object$model.rf$variable.importance)),
xlim=NULL, main=NULL)
```

Arguments

object	an abcrf or regAbcrf object.
n.var	number of variables in the variable importance representation.
xlim	range of the abscissa.
main	an overall title for the variable importance plot.

Value

Invisibly, the importance of the variables that were plotted.

See Also

abcrf, plot.abcrf, plot.regAbcrf

Examples

```
data(snp)
modindex <- snp$modindex[1:500]
sumsta <- snp$sumsta[1:500,]
data1 <- data.frame(modindex, sumsta)
model.rf <- abcrf(modindex~., data1, ntree=100)
variableImpPlot(model.rf)
```

Index