Package 'mogavs'

November 6, 2015

Type Package

Title Multiobjective Genetic Algorithm for Variable Selection in Regression

Version 1.0.1

Date 2015-11-04

Imports cvTools, graphics, stats

Description Functions for exploring the best subsets in regression with a genetic algorithm. The package is much faster than methods relying on complete enumeration, and is suitable for datasets with large number of variables.

License GPL-2

LazyData yes

NeedsCompilation no

Author Tommi Pajala [aut, cre],
Pekka Malo [aut],
Ankur Sinha [aut],
Timo Kuosmanen [ctb]

Maintainer Tommi Pajala <tommi.pajala@aalto.fi>

Depends R (>= 2.10)

Repository CRAN

Date/Publication 2015-11-06 18:29:17

R topics documented:

mogavs-package 2
createAdditionalPlots 2
crimeData 3
cv.mogavs 4
getBestModel 5
getBestModelVars 6
mogavs 7
mogavsToLinear 9

plotVarUsage 10
sampleData 11
summary.mogavs 11

Index **13**

`mogavs-package` *Package for regression variable selection with genetic algorithm MOGA-VS*

Description

Runs the genetic algorithm MOGA-VS for variable selection on a given data set.

Details

Package:	mogavs
Type:	Package
Version:	1.0
Date:	2015-06-17
License:	GPL-2

Author(s)

Tommi Pajala, Ankur Sinha, Pekka Malo, Timo Kuosmanen Maintainer: Tommi Pajala <tommi.pajala@aalto.fi>

References

Sinha, A., Malo, P. & Kuosmanen, T. (2015) A Multi-objective Exploratory Procedure for Regression Model Selection. *Journal of Computational and Grahical Statistics, 24(1). pp. 154-182.*

Examples

```
data(sampleData)
mod <- mogavs(y~.,data=sampleData,maxGenerations=20)
summary(mod)
createAdditionalPlots(mod,epsilonBand=0,kBest=30,"kbest")
```

`createAdditionalPlots` *Function for plotting boundaries of the archive set.*

Description

A plotting function for plotting the set of all tried models, and highlighting either all models within `epsilonBand` MSE of the efficient frontier, or the `kBest` best models for each number of variables.

Usage

```
createAdditionalPlots(mogavs, epsilonBand = 0, kBest = 1, method = c("MSE", "kBest"))
```

Arguments

`mogavs`	A model of class mogavs.
`epsilonBand`	The value of epsilonBand, ie. the mean square error inside which models are highlighted.
`kBest`	The number of models that will be highlighted for each number of variables.
`method`	Either `MSE` or `kBest` (case-insesitive). `MSE` plots the set of all tried models, with models inside the `epsilonBand` highlighted. `method="kBest"` plots the set of all tried models, with the `kBest` best models for each number of variables highlighted.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

`mogavs`

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
createAdditionalPlots(mod,epsilonBand=0,kBest=15,"kbest")
createAdditionalPlots(mod,epsilonBand=0.001,"mse")
```

---

crimeData *Crime Data Set with Imputed Values*

Description

This is the communities and crime data set, but with missing values imputed with the **mclust** package.

Usage

```
data(crimeData)
```

Source

http://archive.ics.uci.edu/ml/datasets/Communities+and+Crime

References

Redmond, M. (2009) Communities and Crime Data Set. *UCI Machine Learning Repository*

See Also

sampleData

Examples

```
data(crimeData)
head(crimeData)
```

---

cv.mogavs *k-Fold Crossvalidation for a mogavs model*

Description

Performs k-fold CV for a model of class mogavs via the cvTools package.

Usage

```
cv.mogavs(mogavs, nvar, data, y_ind, K = 10, R = 1, order = FALSE)
```

Arguments

mogavs	A model of class mogavs.
nvar	The number of variables for which you want to run k-fold CV.
data	Used data set.
y_ind	The column number for the y-variable in the dataset.
K	Number of folds in the cross-validation, default K=10.
R	Number of repeats for the CV, default R=1.
order	Logical, whether the result should be sorted by the column CVerror.

Details

Perform k-fold cross-validation for all the linear models with nvar number of variables, which have been tried during the course of the genetic algorithm.

Value

A data frame with the following columns:

archInd	The row index of the linear model in the archiveSet of the mogavs model.
formula	The formula of the linear model as a character string.
CVerror	The root mean square error of the model.
CVse	The standard error of the model across the R runs of the cross-validation. NA if R=1.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

mogavsToLinear

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
cv.mogavs(mod,nvar=3,data=sampleData,y_ind=1,K=10,R=1,order=FALSE)
```

---

getBestModel *Get the best model with nvar variables, or by AIC, BIC or knee-point.*

Description

Returns a binary vector of variables for the best model, as defined by either the AIC, BIC, or knee-point, or alternatively the best for a given number of variables.

Usage

```
getBestModel(mogavs, nvar, method = c("AIC", "BIC", "knee", "mse", NULL))
```

Arguments

mogavs	A model of the class mogavs.
nvar	Number of variables for the best model. Only used if method is mse or NULL. Can be omitted if method is named and is AIC, BIC or knee.
method	The desired metric for defining the best model. If nvar is omitted, method must be named.

Details

The methods AIC, BIC and knee look at the whole set of tried models, whereas mse or NULL means that the function looks for the best model with nvar variables and the lowest mean square error.

Value

A binary vector of the variables in the best model.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

getBestModelVars

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
getBestModel(mod,15,"mse")
getBestModel(mod,method="BIC")
```

---

getBestModelVars *Get variable names of the best model with nvar variables, or defined by lowest MSE, AIC, BIC or knee-point.*

Description

Returns a vector of variable names for the best model, as defined by either the AIC, BIC, or knee-point, or alternatively the best for a given number of variables.

Usage

```
getBestModelVars(mogavs, nvars, data, method=c("AIC","BIC","mse",NULL))
```

Arguments

mogavs	A model of the class mogavs.
nvars	Number of variables for the best model. Only used if method is NULL or MSE.
data	The used data set.
method	The desired metric for defining the best model. If nvar is omitted, method must be named.

Details

The methods AIC, BIC and knee look at the whole set of tried models, whereas NULL means that the function looks for the best model with $nvar$ variables and the lowest mean square error.

Value

Returns a character vector of the variable names of the best model.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

getBestModel,plotVarUsage

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
getBestModelVars(mod,nvars=15,sampleData,NULL)
getBestModelVars(mod,nvars=0,data=sampleData,method="BIC")
```

---

mogavs *Multiobjective Genetic Algorithm for Variable Selection*

Description

The main function for the mogavs genetic algorithm, returning a list containing the full archive set of regression models tried and the nondominated set.

Usage

```
## Default S3 method:
mogavs(x, y, maxGenerations = 10*ncol(x), popSize = ncol(x), noOfOffspring = ncol(x),
crossoverProbability = 0.9, mutationProbability = 1/ncol(x), kBest = 1,
plots = F, additionalPlots = F, ...)
## S3 method for class 'formula'
mogavs(formula, data, maxGenerations= 10*ncol(x), popSize = ncol(x),
noOfOffspring = ncol(x), crossoverProbability = 0.9, mutationProbability = 1/ncol(x),
kBest = 1, plots = F, additionalPlots = F, ...)
```

Arguments

formula	Formula interface with y~x1+x2 or y~. for predicting y with x1 and x2 or all predictors, respectively.
data	A data frame containing the variables mentioned in the formula.
x	An n x p matrix containing the n observations of p values used in the regression.
y	An n x 1 vector of values to fit the regression to.
maxGenerations	Number of maximum generations to be run in the evolutionary algorithm. Default is 10*ncol(x)
popSize	Population size, ie. how many regression models the population holds. Default is ncol(x).
noOfOffspring	Indicates how many offspring models are generated for each generation. Default is ncol(x).

`crossoverProbability`
: Indicates the probability of crossover for each offpring. Default is 0.9.

`mutationProbability`
: Indicates the probability of mutation for each offspring. Default is 1/ncol(x).

`kBest`
: Indicates how many best models for each number of variables are highlighted in printing at the end of the run (default=1).

`plots`
: Binary variable for turning plotting for each generation on/off.

`additionalPlots`
: Binary variable for turning additional plotting at the end of the run on/off. Plot can also be generated after the run with given `createAdditionalPlots` functions.

`...`
: Any additional arguments.

Details

Runs genetic algorithm for the linear regression model space, with predicting variables x and predicted values y. Alternatively, can be given a data frame and formula. Setting `plots=TRUE` creates for each generation a plot, showing the current efficient boundary of the models. Setting `additionalPlots=TRUE` gives out an additional plot at the end of the algorithm, showing the full set of tried models and the `kBest` best models for each number of variables. All plotting is turned off by default to make processing faster.

Value

Returns model of class `mogavs` with items

`nonDominatedSet`
: Matrix of the nondominated models.

`numOfVariables`
: Vector of the number of variables for each model in the nonDominatedSet.

`MSE`
: Vector of mean square errors for each model in the nonDominatedSet.

`archiveSet`
: The full archive set of models tried

`kBest`
: The value of kBest used

`maxGenerations`
: Number of generations used.

`crossoverProbability`
: The crossover probability used.

`noOfOffspring`
: Number of generated offspring for each generation.

`popSize`
: The population size.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

References

Sinha, A., Malo, P. & Kuosmanen, T. (2015) A Multi-objective Exploratory Procedure for Regression Model Selection. *Journal of Computational and Grahical Statistics, 24(1). pp. 154-182.*

See Also

createAdditionalPlots

Examples

```
data(sampleData)
#just a few generations to keep test fast
mogavs(y~.,data=sampleData,maxGenerations=5)

#with a more sensible number of generations, with all plotting on
## Not run: mogavs(y~.,data=sampleData,maxGenerations=100,plots=TRUE,additionalPlots=TRUE)
```

---

mogavsToLinear *Transform a mogavs model into a linear model.*

Description

Takes in a mogavs model and a number of variables, and transforms that into linear model as in `lm`.

Usage

```
mogavsToLinear(bestModel, y_ind, data, ...)
```

Arguments

`bestModel`	A binary vector, representing the variables in one model for a given number of variables.
`y_ind`	Column number for the y values in data.
`data`	The used data set.
`...`	Additional arguments.

Value

`lm`	A linear model of class `lm`.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

getBestModel,getBestModelVars

Examples

```
data(sampleData)
mod<-mogavs(y~.,sampleData,maxGenerations=20)

#get the best model with 15 variables
bm<-getBestModel(mod,15,method=NULL)

#transform best model into a linear model
mogavsToLinear(bm,1,sampleData)
```

`plotVarUsage` *Produce a visual summary of how many times each variable appears on the efficient frontier.*

Description

Visualizes how models on the efficient frontier use different variables. May be useful for finding out which variables seem to be most useful for explanation.

Usage

```
plotVarUsage(mogavs, method = c("hist", "plot", "table"))
```

Arguments

`mogavs`	A model of the class mogavs.
`method`	The chosen method for visualizing variable usage, `hist` for a histogram, `plot` for a plot, and `table` for just a table.

Value

In the case of `method="hist"` or `method="plot"` doesn't return anything, if `method="table"` returns a table.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

`getBestModel`,`getBestModelVars`

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
plotVarUsage(mod,"table")
plotVarUsage(mod,"hist")
plotVarUsage(mod,"plot")
```

sampleData *Simulated Data Set for MOGA-VS*

Description

A simulated data set with 100 observations, 1 dependent variable and 60 independent variables.

Usage

```
data("sampleData")
```

Details

The data frame variable y includes the dependent variables, while the x1 to x60 refer to independent variables.

Examples

```
data(sampleData)
ans <- mogavs(as.matrix(sampleData)[,-1],as.matrix(sampleData)[,1],maxGenerations=10)
```

summary.mogavs *Summary function for mogavs*

Description

S3 summary method for the mogavs class, producing output about the run and the models on the efficient frontier.

Usage

```
## S3 method for class 'mogavs'
summary(object, ...)
```

Arguments

object	A model of class mogavs.
...	Additional arguments for summary, only here to achieve S3 consistency, ie. they are ignored.

Value

A list with the following details:

maxGenerations	The number of generations run for the model.
boundary	The efficient frontier, summarized as a two-column matrix with the number of variables and MSE.
modelsTried	The number of models tried during the run.

Author(s)

Tommi Pajala <tommi.pajala@aalto.fi>

See Also

mogavs

Examples

```
data(sampleData)
mod<-mogavs(y~.,data=sampleData,maxGenerations=20)
summary(mod)
```

Index

∗Topic **datasets**
createAdditionalPlots, 2
cv.mogavs, 4
getBestModel, 5
getBestModelVars, 6
mogavs, 7
mogavsToLinear, 9
plotVarUsage, 10
summary.mogavs, 11
∗Topic **models**
createAdditionalPlots, 2
cv.mogavs, 4
getBestModel, 5
getBestModelVars, 6
mogavs, 7
mogavsToLinear, 9
plotVarUsage, 10
summary.mogavs, 11
∗Topic **package**
mogavs-package, 2
∗Topic **regression**
createAdditionalPlots, 2
cv.mogavs, 4
getBestModel, 5
getBestModelVars, 6
mogavs, 7
mogavsToLinear, 9
plotVarUsage, 10
summary.mogavs, 11

createAdditionalPlots, 2, *9*
crimeData, 3
cv.mogavs, 4

getBestModel, 5, *7*, *9, 10*
getBestModelVars, *6*, 6, *9, 10*

mogavs, *3*, 7, *12*
mogavs-package, 2
mogavsToLinear, *5*, 9

plotVarUsage, *7*, 10

sampleData, *4*, 11
summary.mogavs, 11