Conformal: an R package to calculate prediction errors in the conformal prediction framework

October 21, 2014

ConformalClassification *Conformal Prediction For Classification*

Description

R class to p.values for individual predictions according to the conformal prediction framework.

Usage

```
ConformalClassification(...)
```

Author(s)

Isidro Cort<c3><a9>s-Ciriano <isidrolauscher@gmail.com>

References

http://pubs.acs.org/doi/abs/10.1021/ci5001168

See Also

ConformalRegression

Examples

```
# Optional for parallel training
#library(doMC)
#registerDoMC(cores=4)

data(LogS)

# convert data to categorical
LogSTrain[LogSTrain > -4] <- 1
LogSTrain[LogSTrain <= -4] <- 2
LogSTest[LogSTest > -4] <- 1
LogSTest[LogSTest <= -4] <- 2
```

```
LogSTrain <- factor(LogSTrain)
LogSTest <- factor(LogSTest)

algorithm <- "rf"

trControl <- trainControl(method = "cv",  number=5,savePredictions=TRUE,
                          predict.all=TRUE,keep.forest=TRUE,norm.votes=TRUE)
set.seed(3)
model <- train(LogSDescsTrain, LogSTrain, algorithm,type="classification",
               trControl=trControl)


# Instantiate the class and get the p.values
example <- ConformalClassification$new()
example$CalculateCVAlphas(model=model)
example$CalculatePValues(new.data=LogSDescsTest)
example$p.values$P.values
example$p.values$Significance_p.values
```

ConformalRegression *Conformal Prediction for Regression*

Description

R class to create and visualize confidence intervals for individual predictions according to the conformal prediction framework.

Usage

```
ConformalRegression(...)
```

Author(s)

Isidro Cort<c3><a9>s-Ciriano <isidrolauscher@gmail.com>

References

http://pubs.acs.org/doi/abs/10.1021/ci5001168

See Also

ConformalClassification

Examples

```
##############################################
### Example
##############################################

# Optional for parallel training
#library(doMC)
#registerDoMC(cores=4)
```

```
data(LogS)

algorithm <- "svmRadial"
tune.grid <- expand.grid(.sigma = expGrid(power.from=-10, power.to=-6, power.by=1, base=2),
                         .C = expGrid(power.from=4, power.to=10, power.by=2, base=2))
trControl <- trainControl(method = "cv",  number=5,savePredictions=TRUE)
set.seed(3)
model <- train(LogSDescsTrain, LogSTrain, algorithm,
               tuneGrid=tune.grid,
               trControl=trControl)


# Train an error model
ErrorModel <- error_model(PointPredictionModel=model,x.train=,LogSDescsTrain,
                          savePredictions=TRUE,algorithm=algorithm,
                          trControl=trControl, tune.grid=tune.grid)

# Instantiate the class and get the confidence intervals
example <- ConformalRegression$new()
example$CalculateAlphas(model=model,error_model=ErrorModel,ConformityMeasure=StandardMeasure)
example$GetConfidenceIntervals(new.data=LogSDescsTest)
example$CorrelationPlot(obs=LogSTest)
example$plot
```

error_model   *Calculation of an error model*

Description

This function permits the calculation of an error model from (i) a training set, and (ii) a caret model trained on this set to predict the response variable of interest using cross-validation (point prediction model). The cross-validation predictions are extracted from the input model. The errors in prediction are then calculated, which serve as the response variable for the error model. The error model uses as descriptors the same descriptors used by the point prediction model, input through the argument "PointPredictionModel".

Usage

```
error_model(PointPredictionModel, x.train, algorithm = "svmRadial", ...)
```

Arguments

`PointPredictionModel`
Point prediction model from which the cross-validation predictions will be extracted.

`x.train` Descriptors for the datapoints in the training set used to train the point prediction model, and which will serve to train the error model.

`algorithm` The machine learning algorithm to be used to train the error model. The defaul value is Support Vector Machine with radial kernel ("svmRadial").

`...`	Additional arguments that can be passed to the train function from the R package caret to train the error model.

Value

A list of class train containing the error model (caret model).

Author(s)

Isidro Cort<c3><a9>s-Ciriano <isidrolauscher@gmail.com>

References

http://caret.r-forge.r-project.org/training.html

Kuhn (2008), "Building Predictive Models in R Using the caret" (http://www.jstatsoft.org/v28/i05/)

See Also

`ConformalRegression` `GetCVPreds`

`expGrid` *Exponential Grid Definition*

Description

The function defines an exponential series, which can be used, e.g. when defining the parameter space when training some models such as Support Vector Machines or Gaussian Processes.

Usage

```
expGrid(power.from, power.to, power.by, base)
```

Arguments

`power.from`	The starting exponential of the series.
`power.to`	The latest exponential of the series.
`power.by`	The exponential step of the series.
`base`	The base of the exponential series.

Value

A vector with the exponential series.

Author(s)

Isidro Cortes <isidrolauscher@gmail.com>

Examples

```
expGrid(power.from=-10,power.to=10,power.by=2,base=10)
```

GetCVPreds — *Extract the cross-validation predictions for the datapoints in the training set*

Description

This function extracts the cross-validation predictions from a caret model trained with cross-validation. If grid-search is used to optimize the hyperparamter values, the \it GetCVPreds extracts the cross-validation predictions corresponding to the optimal hyperparameter values.

Usage

```
GetCVPreds(model)
```

Arguments

`model`	A caret model trained with cross-validation.

Value

A data frame containing the observed and the predicted values for the datapoints in the training set, their index according to the training set, the optimal hyperparameter values, and the fold index.

Author(s)

Isidro Cort<c3><a9>s-Ciriano <isidrolauscher@gmail.com>

LogS — *Small Molecule Solubility (LogS) Data*

Usage

```
data(LogS)
```

Details

This dataset comprises the aqueous solubility (S) values at a temperature of 20-25 $^\circ$C in mol/L, expressed as logS, for 1,708 small molecules reported by Wang \it et al.. Compound structures were standardized with the function \it StandardiseMolecules from the R package \it camb using the default parameters: (i) all molecules were kept irrespective of the numbers of fluorines, iodines, chlorine, and bromines present in their strucuture, or (ii) of their molecular mass. 905 one-dimensional topological and physicochemical descriptors were calculated with the function \it GeneratePadelDescriptors from the R package \it camb which invokes the PaDEL-Descriptor \citeppadel Java library. Near zero variance and highly-correlated descriptors were removed with the functions (i) \it RemoveNearZeroVarianceFeatures (cut-off value of 30/1), and (ii) \it RemoveHighlyCorrelatedFeatures (cut-off value of 0.95) After applying these steps the dataset consists of 1,606 molecules encoded with 211 descriptors.\

Using data(LogS) exposes 4 objects:

(i) LogSDescsTrain is a data frame with PaDEL descriptors for the datapoints in the training set (70% of the data). (ii) LogSTrain is a numeric vector containing the data solubility values for the datapoints in the training set. (iii) LogSDescsTest is a data frame with PaDEL descriptors for the datapoints in the test set (30% of the data). (iv) LogSTest is a numeric vector containing the data solubility values for the datapoints in the test set.

References

Wang et al (2007)

Examples

```
# To use the data
data(LogS)
```

`StandardMeasure` *Default Nonconformity measure used in regression (ConformalRegression class)*

Description

Nonconformity measure used by default in the class ConformalRegression to calculate nonconformity scores (alpha).

Usage

```
StandardMeasure(obs, pred, error)
```

Arguments

`obs`	Observed values for the response variable.
`pred`	Predicted values for the response variable.
`error`	Predicted errors calculated with an error model.

Value

A numeric vector with the nonconformity scores (alpha).

Author(s)

Isidro Cort<c3><a9>s-Ciriano <isidrolauscher@gmail.com>

References

http://pubs.acs.org/doi/abs/10.1021/ci5001168

See Also

ConformalRegression

Index