Package ‘EBMAforecast’

February 19, 2015

Type Package
Title Ensemble BMA Forecasting
Version 0.42
Date 2013-02-25
Author Jacob M. Montgomery, Florian Hollenbach, and Michael D. Ward
Maintainer Jacob Montgomery <jacob.montgomery@wustl.edu>
Description Ensemble BMA for social science data
License GPL (>= 2)
Depends separationplot, ensembleBMA, plyr, methods, Hmisc, abind
LazyLoad yes
Collate 'forecastData.R' 'calibrateEnsemble.R' 'fitDatLogit.R'
    'fitDatNormal.R' 'compareModels.R' 'document-data.R'
    'summary.R' 'fitEnsembleLogit.R' 'fitEnsembleNormal.R'
    'makeForecastData.R' 'plot.R'
NeedsCompilation no
Repository CRAN
Date/Publication 2013-02-25 07:29:27

R topics documented:

  calibrateEnsemble .............................................. 2
  calibrationSample ........................................... 4
  compareModels ............................................... 5
  EBMAforecast .................................................. 6
  ForecastData .................................................. 7
  makeForecastData ............................................ 8
  plot .......................................................... 10
  presidentialForecast ........................................ 12
  summary ....................................................... 13
  testSample ................................................... 14

Index 16
**calibrateEnsemble**

*Calibrate an ensemble Bayesian Model Averaging model*

**Description**

This function calibrates an EBMA model based on out-of-sample performance in the calibration time period. Given a dependent variable and calibration-sample predictions from multiple component forecast models in the `forecastdata` the `calibrateEnsemble` function fits an ensemble BMA mixture model. The weights assigned to each model are derived from the individual model’s performance in the calibration period. Missing observations are allowed in the calibration period, however models with missing observations are penalized. When missing observations are prevalent in the calibration set, the EM algorithm is adjusted and model parameters are estimated by maximizing a renormalized partial expected complete-data log-likelihood (Fraley et al. 2010).

**Arguments**

- `.forecastData` An object of class `ForecastData` that will be used to calibrate the model.
- `exp` The exponential shrinkage term. When `model="logit"`, forecasts are raised to the \((1/\text{exp})\) power on the logit scale for the purposes of bias reduction. The default value is `exp=3`. When `model="logit"`, forecasts are raised to the \((1/\text{exp})\) and the default value is `exp=1`.
- `tol` Tolerance for improvements in the log-likelihood before the EM algorithm will stop optimization. The default is `\text{sqrt(.Machine\$double.\text{eps})}`.
- `maxIter` The maximum number of iterations the EM algorithm will run before stopping automatically. The default is `\text{maxIter}=\text{1e6}`.
- `model` The model type that should be used given the type of data that is being predicted (i.e., normal, binary, etc.).
- `method` The estimation method used. Currently only implements "EM".
- `predType` The prediction type used for the EBMA model under the normal model, user can choose either `posterior.\text{median}` or `posterior.\text{mean}`. Posterior median is the default.
- `useModelParams` A logical that indicates whether the regression parameters should be used to post-process the data. The default is TRUE. We recommend setting this to FALSE for sparse datasets or datasets modeling rare events.
- ... Not implemented

**Value**

Returns a data of class `FDatFitLogit` or `FDatFitNormal`, a subclass of `ForecastData`, with the following slots

- `predCalibration` An array containing the predictions of all component models and the EBMA model for all observations in the calibration period.
calibrateEnsemble

predTest
An array containing the predictions of all component models and the EBMA model for all observations in the test period.

outcomeCalibration
A vector containing the true values of the dependent variable for all observations in the calibration period.

outcomeTest
A vector containing the true values of the dependent variable for all observations in the test period.

modelName
A character vector containing the names of all component models. If no model names are specified, names will be assigned automatically.

modelWeights
A vector containing the model weights assigned to each model.

modelParams
An array containing the parameters for the individual logit models that transform the component models.

logLik
The final log-likelihood for the calibrated EBMA model.

exp
The exponential shrinkage term.

tol
Tolerance for improvements in the log-likelihood before the EM algorithm will stop optimization.

maxIter
The maximum number of iterations the EM algorithm will run before stopping automatically.

method
The estimation method used.

call
The actual call used to create the object.

Author(s)

Michael D. Ward <<michael.d.ward@duke.edu>> and Jacob M. Montgomery <<jacob.montgomery@wustl.edu>>
and Florian M. Hollenbach <<florian.hollenbach@duke.edu>>

References


## Description

This includes the calibration sample data for the insurgency forecasting example in Montgomery, Hollenbach and Ward (2012). It provides the predictions for the three models included in the Ensemble model, as well as the true values of the dependent variable for insurgency in 29 Asian countries. The calibration sample ranges from January 2008 to December 2009.

## Details

The variables included in the dataset are:

- **LMER** The calibration sample predictions of the LMER model from the insurgency prediction example in Montgomery et. al. (2012). The LMER model is a generalized linear mixed effects model using the logistic link function. It includes two random effects terms and several other covariates.

- **SAE** The calibration sample prediction of the SAE model from the insurgency prediction example in Montgomery et. al. (2012). This is a model developed as part of the ICEWS project and was designed by Strategic Analysis Enterprises. It is a simple generalized linear model with 27 independent variables.

- **GLM** The calibration sample prediction of the GLM model from the insurgency prediction example in Montgomery et. al. (2012). This is a crude logistic model with only four independent variables.

- **Insurgency** The true values of the dependent variable in the calibration sample from the insurgency prediction example in Montgomery et. al. (2012). This is a binary variable indicating the actual occurrence of insurgency for each observation in the calibration sample.

## References

**compareModels**

*Function for comparing multiple models based on predictive performance*

**Description**

This function produces statistics to compare the predictive performance of the different models component models, as well as for the EBMA model itself, for either the calibration or the test period. It currently calculates the area under the ROC (\(\text{auc}\)), the brier score, the percent of observations predicted correctly (\(\text{percCorrect}\)), as well as the proportional reduction in error compared to some baseline model (\(\text{pre}\)) for binary models. For models with normally distributed outcomes the \(\text{compareModels}\) function can be used to calculate the root mean squared error (\(\text{rmse}\)) as well as the mean absolute error (\(\text{mae}\)).

**Arguments**

- **.forecastData** An object of class `ForecastData`.
- **.period** Can take value of "calibration" or "test" and indicates the period for which the test statistics should be calculated.
- **.fitStatistics** A vector naming statistics that should be calculated. Possible values include "\(\text{auc}\)", "\(\text{brier}\)", "\(\text{percCorrect}\)" , "\(\text{pre}\)" for logit models and "\(\text{mae}\)" , "\(\text{rmse}\)" for normal models.
- **.threshold** The threshold used to calculate when a "positive" prediction is made by the model for binary dependent variables.
- **.baseModel** Vector containing predictions used to calculate proportional reduction of error ("\(\text{pre}\)").
- **...** Not implemented

**Value**

A data object of the class 'CompareModels' with the following slots:

- **fitStatistics** The output of the fit statistics for each model.
- **period** The period, "calibration" or "test", for which the statistics were calculated.
- **threshold** The threshold used to calculate when a "positive" prediction is made by the model.
- **baseModel** Vector containing predictions used to calculate proportional reduction of error ("\(\text{pre}\)").

**Author(s)**

Michael D. Ward <<michael Ndward@duke Nd.edu>> and Jacob M. Montgomery <<jacob Nmontgomery@wustl N.edu>>
References


See Also

ensembleBMA

Examples

## Not run:
data(calibrationSample)
data(testSample)

this.ForecastData <- makeForecastData(.predCalibration=calibrationSample[,c("LMER", "SAE", "GLM")], .outcomeCalibration=calibrationSample[,"Insurgency"], .predTest=testSample[,c("LMER", "SAE", "GLM")], .outcomeTest=testSample[,"Insurgency"], .modelNames=c("LMER", "SAE", "GLM"))

this.ensemble <- calibrateEnsemble(this.ForecastData, model="logit", tol=0.001, exp=3)

compareModels(this.ensemble,"calibration")

compareModels(this.ensemble,"test")

## End(Not run)

### EBMAforecast

**Description**

The EBMAforecast package (currently under development) allows users to increase the accuracy of forecasting models by pooling multiple component forecasts to generate ensemble forecasts. It includes functions to fit an ensemble Bayesian model averaging (EBMA) model using in-sample predictions, generate ensemble out-of-sample predictions, and create useful data visualizations. Currently, the package can only handle dichotomous outcomes or those with normally distributed errors, although additional models will be added to the package in the coming months. Missing observation are allowed in the calibration set, but models with many predictions missing are penalized.

**Author(s)**

Michael D. Ward <<michael.d.ward@duke.edu>> and Jacob M. Montgomery <<jacob.montgomery@wustl.edu>> and Florian M. Hollenbach <<florian.hollenbach@duke.edu>>
### ForecastData

**References**


### Examples

```r
## Not run:
demo(EBMAforecast)
demo(presForecast)
```

## End(Not run)

---

**ForecastData**

An ensemble forecasting data object

---

**Description**

Objects of class ForecastData are used in the calibrateEnsemble function. Datasets should be converted into an object of class ForecastData using the makeForecastData function. Individual slots of the ForecastData object can be accessed and changed using the get and set functions respectively. Missing observations in the prediction calibration set are allowed.

**Details**

A data object of the class 'ForecastData' has the following slots:

- `predCalibration` An array containing the predictions of all component models for the observations in the calibration period.
- `predTest` An array containing the predictions of all component models for all the observations in the test period.
- `outcomeCalibration` A vector containing the true values of the dependent variable for all observations in the calibration period.
- `outcomeTest` A vector containing the true values of the dependent variable for all observations in the test period.
- `modelNames` A character vector containing the names of all component models.
makeForecastData

Build a ensemble forecasting data object

Description

This function uses the component model forecasts and dependent variable observations provided by the user to create an object of class ForecastData, which can then be used to calibrate and fit the ensemble. Individual slots of the ForecastData object can be accessed and changed using the get and set functions respectively. Missing predictions are allowed in the calibration set.
**makeForecastData**

**Arguments**

- `.predCalibration` A matrix with the number of rows being the number of observations in the calibration period and a column with calibration period predictions for each model.

- `.predTest` A vector with the number of rows being the number of observations in the test period and a column with test period predictions for each model.

- `.outcomeCalibration` A vector with the true values of the dependent variable for each observation in the calibration period.

- `.outcomeTest` A vector with the true values of the dependent variable for each observation in the test period.

- `.modelName` A vector of length p with the names of the component models.

- `...` Additional arguments not implemented

**Value**

A data object of the class 'ForecastData' with the following slots:

- `predCalibration` An array containing the predictions of all component models for all observations in the calibration period.

- `predTest` An array containing the predictions of all component models for all observations in the test period.

- `outcomeCalibration` A vector containing the true values of the dependent variable for all observations in the calibration period.

- `outcomeTest` A vector containing the true values of the dependent variable for all observations in the test period.

- `modelName` A character vector containing the names of all component models. If no model names are specified, names will be assigned automatically.

**Author(s)**

Michael D. Ward <<michael.d.ward@duke.edu>> and Jacob M. Montgomery <<jacob.montgomery@wustl.edu>> and Florian M. Hollenbach <<florian.hollenbach@duke.edu>>

**References**


**See Also**

ensembleBMA
Examples

## Not run:

```r
data(calibrationSample)
```

```r
data(testSample)
this.ForecastData <- makeForecastData(predCalibration=calibrationSample[,c("LMER", "SAE", "GLM")],
outcomeCalibration=calibrationSample[,"Insurgency"],
predTest=testSample[,c("LMER", "SAE", "GLM")],
outcomeTest=testSample[,"Insurgency"], modelName=c("LMER", "SAE", "GLM"))
```

```r
### to access individual slots in the ForecastData object
getPredCalibration(this.ForecastData)
getOutcomeCalibration(this.ForecastData)
getPredTest(this.ForecastData)
getOutcomeTest(this.FORECASTData)
getModelNames(this.FORECASTData)
```

```r
### to assign individual slots, use set functions

```r
setPredCalibration(this.FORECASTData)<-calibrationSample[,c("LMER", "SAE", "GLM")]
setOutcomeCalibration(this.FORECASTData)<-calibrationSample[,"Insurgency"]
setPredTest(this.FORECASTData)<-testSample[,c("LMER", "SAE", "GLM")]
setOutcomeTest(this.FORECASTData)<-testSample[,"Insurgency"]
setModelNames(this.FORECASTData)<-c("LMER", "SAE", "GLM")
```

## End(Not run)

---

**plot**

Plotting function for ensemble models of the class `FDatFitLogit` or `FDatFitNormal`, which are the objects created by the `calibrateensemble` function.

### Description

Default plotting for objects created by the `calibrateensemble` function. See details below.

### Arguments

- **x**: An object of class `FDatFitLogit` or `FDatFitNormal`
- **period**: Can take value of "calibration" or "test" and indicates the period for which the plots should be produced.
- **subset**: The row names or numbers for the observations the user wishes to plot. Only implemented for the subclass "FDatFitNormal"
- **mainLabel**: A vector strings to appear at the top of each predictive posterior plot. Only implemented for the subclass "FDatFitNormal"
- **xLab**: The label for the x-axis. Only implemented for the subclass "FDatFitNormal"
plot

ylab The label for the y-axis. Only implemented for the subclass "FDatFitNormal"
cols A vector containing the color for plotting the predictive pdf of each component model forecast. Only implemented for the subclass "FDatFitNormal"

Details

For objects of the class FDatFitLogit, this function creates separation plots for each of the fitted models, including the EBMA model. Observations are ordered from left to right with increasing predicted probabilities, which is depicted by the black line. Actual occurrences are displayed by red vertical lines. Plots can be displayed for the test or calibration period. For objects of the class FDatFitNormal, this function creates a plot of the predictive density distribution containing the EBMA PDF and the PDFs for all component models (scaled by their model weights). It also plots the prediction for the ensemble and the components for the specified observations.

Value

NULL

Author(s)

Michael D. Ward <<michael.d.ward@duke.edu>> and Jacob M. Montgomery <<jacob.montgomery@wustl.edu>>

References


See Also

separationplot, ensembleBMA

Examples

## Not run:
data(calibrationSample)
data(testSample)

this.ForecastData <- makeForecastData(.predCalibration=calibrationSample[,c("LMER", "SAE", "GLM")], .outcomeCalibration=calibrationSample[,"Insurgency"], .predTest=testSample[,c("LMER", "SAE", "GLM")], .outcomeTest=testSample[,"Insurgency"], .modelNames=c("LMER", "SAE", "GLM"))

this.ensemble <- calibrateEnsemble(this.ForecastData, model="logit", tol=0.001, exp=3)

plot(this.ensemble, period="calibration")
plot(this.ensemble, period="test")
Sample data Presidential Election

Description

This includes the data for the presidential election forecasting example in Montgomery, Hollenbach and Ward (2012). The data ranges from 1952 to 2008 and includes predictions for the six different component models included in the Ensemble model. Users may split the sample into calibration and test sample.

Details

The variables included in the dataset are:

- **Campbell** Predictions of Campbell’s “Trial-Heat and Economy Model” (Campbell 2008).
- **Abramowitz** Predictions of Abramowitz’s “Time for Change Model” (Abramowitz 2008).
- **Fair** Forecasts from Fair’s presidential vote share model (2010).
- **EWT2C2** Predictions from the model in Column 2 in Table 2 by Erickson and Wlezien (2008).
- **Actual** The true values of the dependent variable, i.e. the incumbent-party vote-share in each presidential election in the sample.

References


Summary Function

Description

This function summarizes the Ensemble models that have been fit previously by the user.

Arguments

- **object**: An object of the subclass `FDatFitLogit` or `FDatFitNormal`
- **period**: The period for which the summary should be provided, either "calibration" or "test".
- **fitStatistics**: A vector naming statistics that should be calculated. Possible values for objects in the `FDatFitLogit` subclass include "auc", "brier", "percCorrect", "pre". Possible values for objects in the `FDatFitNormal` subclass include "rmse" and "mae." Additional metrics will be made available in a future release of this package.
- **threshold**: The threshold used to calculate when a "positive" prediction is made for a model. Not used for objects of the `FDatFitNormal` subclass.
- **baseModel**: A vector containing predictions used to calculate proportional reduction of error ("pre"). Not used for objects of the `FDatFitNormal` subclass.
- **showCoefs**: A logical indicating whether model coefficients from the ensemble should be shown.
- **...**: Not implemented

Value

A data object of the class 'SummaryForecastData' with the following slots:

- **summaryData**: Under the default, the function produces a matrix containing one row for each model plus one row for the EBMA forecast. The first column is always the model weights assigned to the component models. The second and third columns display the model parameters for the transformation of the component models. The remaining columns are the requested fit statistics for all models, as calculated by the `compareModels` function. If `showCoefs=FALSE`, then the model parameters will not be shown.

Author(s)

Michael D. Ward <<michael.d.ward@duke.edu>> and Jacob M. Montgomery <<jacob.montgomery@wustl.edu>>
Examples

```r
## Not run:
data(calibrationSample)
data(testSample)
this.ForecastData <- makeForecastData(.predCalibration=calibrationSample[,c("LMER", "SAE", "GLM")], .outcomeCalibration=calibrationSample[,"Insurgency"], .predTest=testSample[,c("LMER", "SAE", "GLM")], .outcomeTest=testSample[,"Insurgency"], .modelName=c("LMER", "SAE", "GLM"))
this.ensemble <- calibrateEnsemble(this.ForecastData, model="logit", tol=0.001,exp=3)
summary(this.ensemble, period="calibration")
summary(this.ensemble, period="test",showCoefs=FALSE)
## End(Not run)
```

---

testSample

Exemplar test sample data

Description

This includes the test sample data for the insurgency forecasting example in Montgomery, Hollenbach and Ward (2012). It provides the predictions for the three models included in the Ensemble model, as well as the true values of the dependent variable for insurgency in 29 Asian countries. The test sample ranges from January 2010 to December 2010.

Details

The variables included in the dataset are:

- **LMER** The test sample predictions of the LMER model from the insurgency prediction example in Montgomery et. al. (2012). The LMER model is a generalized linear mixed effects model using the logistic link function. It includes two random effects terms and several other covariates.

- **SAE** The test sample prediction of the SAE model from the insurgency prediction example in Montgomery et. al. (2012). This is a model developed as part of the ICEWS project and was designed by *Strategic Analysis Enterprises*. It is a simple generalized linear model with 27 independent variables.

- **GLM** The test sample prediction of the GLM model from the insurgency prediction example in Montgomery et. al. (2012). This is a crude logistic model with only four independent variables.

- **Insurgency** The true values of the dependent variable in the test sample from the insurgency prediction example in Montgomery et. al. (2012). This is a binary variable indicating the actual occurrence of insurgency for each observation in the test sample.
References

Index

* Topic EBMA
  - calibrateensemble, 2
* Topic calibrate
  - calibrateensemble, 2
* Topic datasets
  - calibrationSample, 4
  - presidentialForecast, 12
  - testSample, 14
  - calibrateensemble, 2
  - calibrateensemble, ForecastData-method (calibrateensemble), 2
  - calibrationSample, 4
  - compareModels, 5
  - compareModels, ForecastData-method (compareModels), 5
  - CompareModels-class (compareModels), 5
  - EBMAforecast, 6
  - EBMAforecast-package (EBMAforecast), 6
  - FDatFitLogit-class (calibrateensemble), 2
  - FDatFitNormal-class (calibrateensemble), 2
  - fitEnsemble (calibrateensemble), 2
  - fitEnsemble, ForecastDataLogit-method (calibrateensemble), 2
  - fitEnsemble, ForecastDataNormal-method (calibrateensemble), 2
  - ForecastData, 7
  - ForecastData-class (ForecastData), 7
  - ForecastDataLogit-class (calibrateensemble), 2
  - ForecastDataNormal-class (calibrateensemble), 2
  - getOutcomeCalibration (ForecastData), 7
  - getOutcomeCalibration, ForecastData-method (ForecastData), 7
  - getOutcomeTest (ForecastData), 7
  - getOutcomeTest, ForecastData-method (ForecastData), 7
  - getPredCalibration (ForecastData), 7
  - getPredCalibration, ForecastData-method (ForecastData), 7
  - getPredTest (ForecastData), 7
  - getPredTest, ForecastData-method (ForecastData), 7
  - initialize, ForecastData-method (ForecastData), 7
  - makeForecastData, 8
  - makeForecastData, ANY-method (ForecastData), 7
  - plot, 10
  - plot, FDatFitLogit-method (plot), 10
  - plot, FDatFitNormal-method (plot), 10
  - presidentialForecast, 12
  - print, ForecastData-method (ForecastData), 7
  - print, SummaryForecastData-method (summary), 13
  - setModelNames, ForecastData-method (ForecastData), 7
  - setModelNames <- (ForecastData), 7
  - setModelNames<-, ForecastData-method (ForecastData), 7
  - setOutcomeCalibration, ForecastData-method (ForecastData), 7
  - setOutcomeCalibration <- (ForecastData), 7
  - setOutcomeCalibration<-, ForecastData-method (ForecastData), 7
setOutcomeTest,ForecastData-method
  (ForecastData), 7
setOutcomeTest<- (ForecastData), 7
setOutcomeTest<-,ForecastData-method
  (ForecastData), 7
setPredCalibration,ForecastData-method
  (ForecastData), 7
setPredCalibration<- (ForecastData), 7
setPredCalibration<-,ForecastData-method
  (ForecastData), 7
setPredTest,ForecastData-method
  (ForecastData), 7
setPredTest<- (ForecastData), 7
setPredTest<-,ForecastData-method
  (ForecastData), 7
show,ForecastData-method
  (ForecastData), 7
show,SummaryForecastData-method
  (summary), 13
summary, 13
summary,FDatFitLogit-method (summary),
  13
summary,FDatFitNormal-method (summary),
  13
SummaryForecastData-class (summary), 13
testSample, 14