Package 'gains'

February 19, 2015

Version 1.1

Date 2013-07-25

Title Gains Table Package

Author Craig A. Rolling <crolling@umn.edu>

Maintainer Craig A. Rolling <crolling@umn.edu>

Depends R (>= 3.0.0)

Description This package constructs gains tables and lift charts for prediction algorithms. Gains tables and lift charts are commonly used in direct marketing applications.

LazyData Yes

License GPL-3

NeedsCompilation no

Repository CRAN

Date/Publication 2013-07-25 16:03:18

R topics documented:

ciaScores 1
gains 2
plot.gains 4
print.gains 5

Index **7**

`ciaScores` *Cell Phones per Country with Predictions*

Description

This data set gives the number of cell phones per person for 194 countries, courtesy of the CIA World Factbook. The data are mostly for 2008. It also gives predicted values of this variable from 5 different methods (OLS, Lasso, Regression Tree, Random Forest, and Additive Model). Finally, there is an indicator for each country indicating whether the country was used in the model development sample or not.

Usage

```
cia.scores
```

Format

a data frame containing 194 rows and 8 columns.

- CellPhonesPP: Number of cell phones per person, from the CIA Factbook.
- PredOLS: Predicted response from an OLS regression.
- PredLasso: Predicted response from a LASSO regression.
- PredTree: Predicted response from a regression tree.
- PredRF: Predicted response from a Random Forest.
- PredSM: Predicted response from an additive model.
- PredGLM: Predicted probability (from a logistic regression) that the country has more cell phones than people.
- train: Indicator, =1 if the country was among the set used to make the predictions, =0 if the country was in the validation set (not used to make predictions).

Source

CIA - The World Factbook https://www.cia.gov/library/publications/the-world-factbook/index.html

---

gains *Gains Table for a Vector of Predictions*

Description

Takes a vector of actual responses and a vector of predictions and constructs a gains table to evaluate the predictions.

Usage

```
gains(actual, predicted, groups=10,
      ties.method=c("max","min","first","average","random"),
      conf=c("none","normal","t","boot"), boot.reps=1000, conf.level=0.95,
      optimal=FALSE,percents=FALSE)
```

Arguments

actual	a numeric vector of actual response values
predicted	a numeric vector of predicted response values. This vector must have the same length as actual, and the ith value of this vector needs to be the model score for the subject with the ith value of the actual vector as its actual response.
groups	an integer containing the number of rows in the gains table. The default value is 10.
ties.method	method of breaking ties. See the ties.method argument of the rank procedure.
conf	method to construct confidence intervals for the mean response in each row of the table. If "none", then no confidence intervals are constructed. If "normal", then critical values from the normal distribution are used. If "t", then critical values from the t distribution are used. If "boot", then 1000 bootstrap samples are drawn from each row, and the upper and lower conf.level/2 values of the distribution are used as the confidence interval endpoints.
boot.reps	the number of bootstrap replications to use for bootstrap confidence intervals. The default value is 1000.
conf.level	the 1 - alpha level of the confidence interval. The default value is 0.95.
optimal	a logical indicated whether the user wants optimal lift indices to be computed. Optimal lift indices represent the results that would be achieved from an optimal ranking of subjects.
percents	a logical that indicates whether to print the mean responses and predicted responses in percent form.

Value

gains returns an S3 object of class gains. The function print.gains can be used to print the results. The function plot.gains can be used to plot the mean response and cumulative mean response for each group. An object of class gains is a list containing the following components:

depth	cumulative percentage of file covered by each row of the gains table (e.g. 10,20,30,...,100).
obs	number of observations in each row.
cume.obs	cumulative number of observations in each row.
mean.resp	mean response in each row.
cume.mean.resp	cumulative mean response in each row.
cume.pct.of.total	cumulative percent of total response.
lift	lift index. The lift index is 100 times the mean.resp for the row divided by the cume.mean.resp for the last row.
cume.lift	cumulative lift index. It is 100 times the cume.mean.resp for the row divided by the cume.mean.resp for the last row.
mean.prediction	mean predicted response in each row.
min.prediction	minimum predicted response in each row. min.prediction and max.prediction can be used to construct decision rules for applying the model.

max.prediction	maximum predicted response in each row.
conf	the argument given for conf.
optimal	the argument given for optimal.
num.groups	the number of rows in the gains table. This will equal groups unless there are fewer distinct predicted values than groups.
percents	the argument given for percents.
conf.lower	lower confidence limit for each row. Only included if confidence intervals are requested in the gains table.
conf.upper	upper confidence limit for each row. Only included if confidence intervals are requested in the gains table.
opt.lift	optimal lift index. The lift index achieved by an optimal ranking of subjects in the file. Only included if optimal lift is requested in the gains table.
opt.cume.lift	optimal cumulative lift index. The cumulative lift by an optimal ranking. Only included if optimal lift is requested in the gains table.

See Also

print.gains for printing the table in a nice way. plot.gains for drawing a graph representing the output. (This graph is sometimes called a lift chart.)

Examples

```
data(ciaScores)
with(subset(ciaScores,train==0),
      gains(actual=CellPhonesPP, predicted=PredOLS, optimal=TRUE))
```

---

plot.gains *Plotting Gains Table Objects*

Description

Plot method for objects of class gains. These plots are sometimes called lift charts.

Usage

```
## S3 method for class 'gains'
plot(x, y=NULL, xlab="Depth of File", ylab="Mean Response",
    type="b", col=c("red3","bisque4","blue4"), pch=c(1,1,1), lty=c(1,1,1),
    legend=c(
    "Mean Response","Cumulative Mean Response","Mean Predicted Response"),
    ylim=c(min(c(x$mean.resp,x$mean.prediction)),
    max(c(x$mean.resp,x$mean.prediction))), main="Gains Table Plot", ...)
```

Arguments

x	an object of class gains.
y	included for compatability with the plot generic but is not used.
xlab	a title for the x axis. See title.
ylab	a title for the y axis. See title.
type	what type of plot should be drawn. The default is "b" for points and lines.
col	vector of length 3 specifying the colors for the series of mean response rates, cumulative mean response rates, and mean predicted response rates, respectively.
pch	vector of length 3 specifying the plotting characters for the series of mean response rates, cumulative mean response rates, and mean predicted response rates, respectively.
lty	vector of length 3 specifying the line types for the series of mean response rates, cumulative mean response rates, and mean predicted response rates, respectively.
legend	character or expression vector of length 3 specifying the legend descriptions for the series of mean response rates, cumulative mean response rates, and mean predicted response rates, respectively.
ylim	the y limits of the plot. See plot.window.
main	an overall title for the plot. See title.
...	additional arguments to plot.

See Also

gains, plot.

Examples

```
data(ciaScores)
## Not run: plot(with(subset(ciaScores,train==0),
      gains(actual=CellPhonesPP, predicted=PredOLS, optimal=TRUE)),
      main="Test Gains Table Plot")
## End(Not run)
```

---

print.gains *Printing Gains Table Objects*

Description

Print method for objects of class gains.

Usage

```
## S3 method for class 'gains'
print(x, digits=2, ...)
```

Arguments

x	an object of class gains.
digits	minimum number of significant digits to print. See print.default.
...	additional arguments to print.

See Also

gains, print.

Examples

```
data(ciaScores)
print(with(subset(ciaScores,train==0),
      gains(actual=CellPhonesPP, predicted=PredOLS, optimal=TRUE)),digits=2)
```

Index

∗Topic **datasets**
ciaScores, 1
∗Topic **misc**
gains, 2
plot.gains, 4
print.gains, 5

ciaScores, 1

gains, 2

plot.gains, 4
print.gains, 5