```
---

title: "Mars R Documentation"

author: "David Wang"

date: "4/20/2021"

output: pdf_document

---
```

MARS {stats}

Description

A new method for regression modeling which produces continuous models with continuous derivatives.

Usage

mars(formula, data, control)

Arguments

formula - an object of class formula. A description of a model to be fitted.

data - a data frame or list that contains the variables in the model as listed by the formula.

control - an object containing the values of Mmax, D and Trace to be used by the mars function. Default values will be used if the control is not provided by the user.

Details

The function takes in three inputs. The formula, a model that typically has the form response ~ terms. The second input is the dataset the model is based upon. The third input is a list of Mmax, d and trace. Mmax is used for foward, d is for backward. The function first bundles the data frame and formula into a "model frame". The subsequent design matrix allows for the creation of dummy variables and is now ready to be parsed into forward stepwise after checking if the values for the control list is valid.

The forward stepwise function is a regression procedure that selects products of mirror-image step functions. It builds a linear prediction equation that is linear in basis functions. The function loops over the parent basis function to generate children. It then constructs the basis functions by using the products of the hinge function h. After choosing the best splits, forward stepwise returns the basis matrix and the splits list which is then fed into the backwards stepwise function.

The backwards stepwise function takes the set of basis functions from the forward algorithm and uses the LOF function which is a generalized cross validation method to find the best model by looping through the number of columns of the basis matrix. It terminates and returns the best set of basis functions.

The lm function is called for the output from backwards stepwise to inherit from the lm function. The output is then classified as class mars which inherits from lm. The mars function then returns the object and terminates.

The user has the methods print and summary for general information on the basis functions and the split list. The plot method is used for general plots made from the data returned by the mars function. The anova method is used to get the anova decomposition of the mars output. The predict function is used to predict data of the fitted model. The user can also choose to parse new data and have the predict function return predicted values of the new data.

Value

mars returns an object of class "mars" and inherits from lm.

The functions summary is used to print a summary of the splits and coefficient list of the results.

The function plot is used to print plots of multiple objects returned by mars

The function anova is used to return the anova decomposition of the mars model.

The function predict is used to return the predicted values of the fitted values from mars.

An object of class "mars" is a list containing at least the following components:

call - The call of the function, shows the user's input when calling the mars function.

formula - the description of the model to be fitted.

y - the response.

B - the basis matrix of the model.

splits - the splits list used to form the basis matrix.

x_names - the names of variables of the model matrix.

data - the data passed by the user that is used by mars.

coefficients - named vector of coefficients.

residuals - residuals. Inherited from lm.

effects - Inherited from lm.

rank - numeric rank of the fitted linear model. Inherited from lm.

fitted.values - fitted mean values. Inherited from lm.

assign - Inherited from lm.

qr - Inherited from lm.

df.residual - residual degrees of freedom. Inherited from lm.

xlevels - a record of the levels of the factors used in fitting. Inherited from lm.

terms - terms object used. Inherited from lm.

model - data frame of each basis function.

Author(s)

David Wang

301293572

dwa110@sfu.ca

References

Jerome H. Friedman. Multivariate Adaptive Regression Splines Invited Paper. The Annals of Statistics, Mar., 1991, Vol. 19, No. 1 (Mar., 1991), pp. 1-67. Institute of Mathematical Statistics. www.jstor.org/stable/2241837

Lecture 9 recording, Brad McNeney

Lecture 10 part2 recording, Brad McNeney

Lecture 12 recording, Brad McNeney

Lecture 13 recording, Brad McNeney

https://github.com/SFUStatgen/SFUStat360/blob/main/Notes/mars4.pdf, Brad McNeney

https://github.com/SFUStatgen/SFUStat360/blob/main/Notes/mars2.pdf, Brad McNeney

https://github.com/SFUStatgen/SFUStat360/blob/main/Notes/mars3.pdf, Brad McNeney

Lab 4, Week 6 solution, Pulindu Ratnasekera

https://rdrr.io/cran/ISLR/man/College.html College data set

https://rdrr.io/cran/ISLR/man/Credit.html Credit data set

https://faculty.nps.edu/sebuttre/home/R/text.html#:~:text=Special%20Characters%20in%20Strings&text=The%20most%20commonly%20used%20are,a%20(single)%20backslash%20character.

?lm()

See Also

summary.mars for summaries.

print.mars for subset information of summaries.

predict.mars for prediction.

anova.mars for the anova decomposition of the mars model

plot.mars for plots of the fitted model given by the mars function

Examples

1.

~~~
```{r}
library(ISLR)
data(Wage)
mc <- mars.control(Mmax=10)
mout <- mars(wage ~ age + education, data=Wage, control=mc)
ff <- fitted(mout)
p1 <- predict(mout)
p2 <- predict(mout,newdata=data.frame(age=Wage$age,education=Wage$education))
head(cbind(ff,p1,p2)) # columns should be identical
mout # tests print method
summary(mout) #test summary method
anova(mout) # test anova method
plot(mout) # test plot method
```
~~~

2.

```{r}
library(ISLR)
data(College)
mc <- mars.control(Mmax=10)
mout <- mars(Apps ~ Private + Accept, data=College, control=mc)
ff <- fitted(mout)
p1 <- predict(mout)
p2 <- predict(mout,newdata=data.frame(Private=College$Private,Accept=College$Accept))
head(cbind(ff,p1,p2)) #test predict method, columns should be identical
mout # test print method
summary(mout) #test summary method
anova(mout) # test anova method
plot(mout)
```

3.

~~~
```{r}
library(ISLR)
data(Credit)
mc <- mars.control(Mmax=10)
mout <- mars(Balance ~ Student + Limit, data=Credit, control=mc)
ff <- fitted(mout)
p1 <- predict(mout)
p2 <- predict(mout,newdata=data.frame(Student=Credit$Student,Limit=Credit$Limit))
head(cbind(ff,p1,p2)) #test predict method, columns should be identical
mout # test print method
summary(mout) #test summary method
anova(mout) # test anova method
plot(mout) # test plot method
```
~~~