Documentation

Yufei Yin

Description

An implementation of Friedman (1990)'s multivariate adaptive regression splines (MARS).

Usage

mars(formula, data, control = NULL, ...)

Arguments

- formula

an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.

- data

an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula), typically the environment from which lm is called.

- control

an object of class 'mars.control'.

- ...

additional arguments to be passed to the low level regression fitting functions.

Details

A new method is presented for flexible regression modeling of high dimensional data. The model takes the form of an expansion in product spline basis functions, where the number of basis functions as well as the parameters associated with each one (product degree and knot locations) are automatically determined by the data.

Value

mars returns an object of class "mars".

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

coefficients: a named vector of coefficients

residuals: the residuals, that is response minus fitted values.

fitted.values: the fitted mean values.

call: the matched call

split: the split points

Author

Name: Yufei Yin

email: yufei_yin@sfu.ca

References

Friedman, J. (1991). Multivariate Adaptive Regression Splines. The Annals of Statistics, 19(1), 1-67. Retrieved April 21, 2021, from http://www.jstor.org/stable/2241837

See Also

anova.mars for ANOVA

plot.mars for residuals vs fitted plot

predict.mars for prediction

print.mars for coefficients

summary.mars for summaries

Examples

library(ISLR)

example 1

data(Wage)

mc<- mars.control(Mmax=10)

mout <- mars(wage ~ age + education ,Wage,mc)

ff <- fitted(mout)

p1 <- predict(mout)

p2 <- predict(mout, newdata = data.frame(age=Wage$age, education = Wage$education))

res = head(cbind(ff,p1,p2)) # columns are identical

mout # tests print method

summary(mout)# test the summary function

anova(mout)#test anova function

plot(mout) # test plot method

example 2

data = read.csv('Real estate.csv')

mc<- mars.control(Mmax=10)

mout = mars(Y.house.price.of.unit.area ~ X3.distance.to.the.nearest.MRT.station,data,mc)

ff = fitted(mout)

p1 = predict(mout)

p2 = predict(mout, newdata=data.frame(X3.distance.to.the.nearest.MRT.station=data$X3.distance.to.the.nearest.MRT.statio

res = head(cbind(ff,p1,p2))

mout

summary(mout)

anova(mout)

plot(mout)

example 3

data = read.csv('insurance.csv')

mc<- mars.control(Mmax=10)

mout = mars(charges ~ bmi + age, data, mc)

ff = fitted(mout)

p1 = predict(mout)

p2 = predict(mout, newdata=data.frame(bmi = data$bmi, age = data$age))

res = head(cbind(ff,p1,p2))

summary(mout)

plot(mout)

anova(mout)

Note:

datasets for example 2 and 3 can be found at https://lionbridge.ai/datasets/10-open-datasets-for-linear-regression/