Package 'bestglm'

March 13, 2020

Type Package

Title Best Subset GLM and Regression Utilities

Version 0.37.3

Date 2020-03-13

Author A.I. McLeod, Changjiang Xu and Yuanhao Lai

Maintainer Yuanhao Lai <ylai72@uwo.ca>

Depends R (>= 3.1.0), leaps

Suggests MASS

Imports lattice, glmnet, grpreg, pls

Enhances caret

Description Best subset glm using information criteria or cross-validation,
carried by using 'leaps' algorithm (Furnival and Wilson, 1974) <doi:10.2307/1267601>
or complete enumeration (Morgan and Tatar, 1972) <doi:10.1080/00401706.1972.10488918>.
Implements PCR and PLS using AIC/BIC.
Implements one-standard deviation rule for use with the 'caret' package.

LazyLoad yes

LazyData yes

Classification/ACM G.3, G.4, I.5.1

Classification/MSC 62M10, 91B84

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-03-13 10:10:02 UTC

R topics documented:

`bestglm-package` *bestglm: Best Subset GLM*

Description

Provides new information criterion BICq as well as AIC, BIC and EBIC for selecting the best model. Additionally, various CV algorithms are also provided.

Details

Package:	bestglm
Type:	Package
Version:	0.33
Date:	2011-11-03
License:	GLP 2.0 or greater
LazyData:	yes
LazyLoad:	yes

bestglm is the main function. All other functions are utility functions and are not normally invoked.

Many examples are provided in the vignettes accompanying this package. The vignettes are produced using the R package `Sweave` and so R scripts can easily be extracted.

The R package `xtable` is needed for the vignette in `SimExperimentBICq.Rnw`.

Author(s)

A.I. McLeod and Changjiang Xu

References

Xu, C. and McLeod, A.I. (2009). Bayesian Information Criterion with Bernouilli Prior.

See Also

`leaps`

Examples

```
## Not run:
data(zprostate)
train<-(zprostate[zprostate[,10],])[,-10]
#Best subset using AIC
bestglm(train, IC="AIC")
#Best subset using BIC
bestglm(train, IC="BIC")
#Best subset using EBIC
bestglm(train, IC="BICg")
#Best subset using BICg with g=0.5 (tuning parameter)
bestglm(train, IC="BICg", t=0.5)
#Best subset using BICq. Note BICq with q=0.25 is default.
bestglm(train, IC="BICq")
#Best subset using BICq with q=0.5 (equivalent to BIC)
bestglm(train, IC="BICq", t=0.5)
#Remark: set seed since CV depends on it
set.seed(123321123)
bestglm(train, IC="CV", t=10)
#using HTF method
bestglm(train, IC="CV", CVArgs=list(Method="HTF", K=10, REP=1))
#Best subset, logistic regression
data(SAheart)
```

```
bestglm(SAheart, IC="BIC", family=binomial)
#Best subset, factor variables with more than 2 levels
data(AirQuality)
#subset
bestglm(AirQuality, IC="BICq")

## End(Not run)
```

`AirQuality` *Daily ozone pollution with meteorlogical and date inputs*

Description

This dataset was derived from the R built-in dataset 'airquality' by adding date information and deleting all missing values. This dataset is referred to as 'environmental' in Cleveland (1993).

Usage

```
data(AirQuality)
```

Format

A data frame with 111 observations on the following 6 variables.

`Solar.R` input, a numeric vector

`Wind` input, a numeric vector

`Temp` input, a numeric vector

`month` input, a factor with levels `May` `Jun` `Jul` `Aug` `Sep` `Oct` `Nov` `Dec` `Jan` `Feb` `Mar` `Apr`

`weekday` input, a factor with levels `Sunday` `Monday` `Tuesday` `Wednesday` `Thursday` `Friday` `Saturday`

`Ozone` output, a numeric vector

Details

Cleveland (1993, Chapter 5) presents an insightful analysis using co-plots and the scatterplot matrix. Several interesting interactions are noted. For a fixed 'Wind', the effect of 'Solar.R' changes as 'Temp' increases. And for a fixed 'Temp', as 'Wind' decreases, the effect of 'Solar.R' is less.

Source

`airquality`

References

Cleveland, W.S. (1993). Visualizing Data.

Examples

```
data(AirQuality)
#Example 1. Find best model
bestglm(AirQuality, IC="BIC")
```

---

asbinary *Binary representation of non-negative integer*

Description

A non-negative integer is represented as a binary number. The digits, 0 or 1, of this number are returned in a vector.

Usage

```
to.binary(n, k = ceiling(logb(n+1,base=2)))
```

Arguments

n	a non-negative integers
k	number of digits to be returned.

Value

A vector of length k. The first element is the least significant digit.

Author(s)

A.I. McLeod

Examples

```
to.binary(63)
to.binary(64)
#sometimes we want to pad result with 'leading' 0's
to.binary(63, k=20)
to.binary(64, k=20)
```

bestglm *Best Subset GLM using Information Criterion or Cross-Validation*

Description

Best subset selection using 'leaps' algorithm (Furnival and Wilson, 1974) or complete enumeration (Morgan and Tatar, 1972). Complete enumeration is used for the non-Gaussian and for the case where the input matrix contains factor variables with more than 2 levels. The best fit may be found using the information criterion IC: AIC, BIC, EBIC, or BICq. Alternatively, with IC='CV' various types of cross-validation may be used.

Usage

```
bestglm(Xy, family = gaussian, IC = "BIC", t = "default",
 CVArgs = "default", qLevel = 0.99, TopModels = 5,
 method = "exhaustive", intercept = TRUE, weights = NULL,
 nvmax = "default", RequireFullEnumerationQ = FALSE, ...)
```

Arguments

Xy	Dataframe containing the design matrix X and the output variable y. All columns must be named.
family	One of the glm distribution functions. The glm function is not used in the Gaussian case. Instead for efficiency either 'leaps' is used or when factor variables are present with more than 2 levels, 'lm' may be used.
IC	Information criteria to use: "AIC", "BIC", "BICg", "BICq", "LOOCV", "CV".
t	adjustable parameter for BICg, BICq or CV. For BICg, default is g=t=1. For BICq, default is q=t=0.25. For CV, default the delete-d method with d=ceil(n(1-1/(log n - 1))) and REP=t=1000. The default value of the parameter may be changed by changing t.
CVArgs	Used when IC is set to 'CV'. The default is use the delete-d algorithm with d=ceil(n(1-1/(log n - 1))) and t=100 repetitions. Note that the number of repetitions can be changed using t. More generally, CVArgs is a list with 3 named components: Method, K, REP, where Method is one of \"HTF\", \"DH\", \"d\" corresponding to using the functions CVHTM (Hastie et al., 2009, K-fold CV), CVDH (adjusted K-fold CV, Davison and Hartigan, 1997) and CVd (delete-d CV with random subsamples, Shao, 1997).
qLevel	the alpha level for determining interval for best q. Larger alpha's result in larger intervals.
TopModels	Finds the best TopModels models.
method	Method used in leaps algorithm for searching for the best subset.
intercept	Default TRUE means the intercept term is always included. If set to FALSE, no intercept term is included. If you want only include the intercept term when it is signficant then set IncludeInterceptQ=FALSE and include a column of 1's in the design matrix.

weights	weights
nvmax	maximum number of independent variables allowed. By default, all variables
RequireFullEnumerationQ	Use exhaustive search algorithm instead of 'leaps'
...	Optional arguments which are passed to lm or glm

Details

In the Gaussian case, the loglikelihood may be written $logL = -(n/2)log(RSS/n)$, where RSS is the residual sum-of-squares and n is the number of observations. When the function 'glm' is used, the log-likelihood, logL, is obtained using 'logLik'. The penalty for EBIC and BICq depends on the tuning parameter argument, t. The argument t also controls the number of replications used when the delete-d CV is used as default. In this case, the parameter d is chosen using the formula recommended by Shao (1997). See CVd for more details.

In the binomial GLM, nonlogistic, case the last two columns of Xy are the counts of 'success' and 'failures'.

Cross-validation may also be used to select the best subset. When cross-validation is used, the best models of size k according to the log-likelihood are compared for k=0,1,...,p, where p is the number of inputs. Cross-validation is not available when there are categorical variables since in this case it is likely that the training sample may not contain all levels and in this case we can't predict the response in the validation sample. In the case of GLM, the \"DH\" method for CV is not available.

Usually it is a good idea to keep the intercept term even if it is not significant. See discussion in vignette.

Cross-validation is not available for models with no intercept term or when force.in is non-null or when nvmax is set to less than the full number of independent variables.

Please see the package vignette for more details and examples.

Value

A list with class attribute 'bestglm' and named components:

BestModel	An lm-object representing the best fitted regression.
Title	A brief title describing the algorithm used: CV(K=K), CVadj(K=K), CVd(d=K). The range of q for an equivalent BICq model is given.
Subsets	The best subsets of size, k=0,1,...,p are indicated as well the value of the log-likelihood and information criterion for each best subset. In the case of categorical variables with more than 2 levels, the degrees of freedom are also shown.
qTable	Table showing range of q for choosing each possible subset size. Assuming intercept=TRUE, k=1 corresponds to model with only an intercept term and k=p+1, where p is the number of input variables, corresponds to including all variables.
Bestq	Optimal q
ModelReport	A list with components: NullModel, LEAPSQ, glmQ, gaussianQ, NumDF, CategoricalQ, Bestk.
BestModels	Variables in the TopModels best list

Methods function 'print.bestglm' and 'summary.bestglm' are provided.

Author(s)

C. Xu and A.I. McLeod

References

Xu, C. and McLeod, A.I. (2009). Bayesian Information Criterion with Bernouilli Prior.

Chen, J. and Chen, Z. (2008). Extended Bayesian Information Criteria for Model Selection with Large Model Space. Biometrika 2008 95: 759-771.

Furnival, G.M. and Wilson, R. W. (1974). Regressions by Leaps and Bounds. Technometrics, 16, 499–511.

Morgan, J. A. and Tatar, J. F. (1972). Calculation of the Residual Sum of Squares for All Possible Regressions. Technometrics 14, 317-325.

Miller, A. J. (2002), Subset Selection in Regression, 2nd Ed. London, Chapman and Hall.

Shao, Jun (1997). An Asymptotic Theory for Linear Model Selection. Statistica Sinica 7, 221-264.

See Also

glm, lm, leaps CVHTF, CVDH, CVd

Examples

```
#Example 1.
#White noise test.
set.seed(123321123)
p<-25   #number of inputs
n<-100  #number of observations
X<-matrix(rnorm(n*p), ncol=p)
y<-rnorm(n)
Xy<-as.data.frame(cbind(X,y))
names(Xy)<-c(paste("X",1:p,sep=""),"y")
bestAIC <- bestglm(Xy, IC="AIC")
bestBIC <- bestglm(Xy, IC="BIC")
bestEBIC <- bestglm(Xy, IC="BICg")
bestBICq <- bestglm(Xy, IC="BICq")
NAIC <- length(coef(bestAIC$BestModel))-1
NBIC <- length(coef(bestBIC$BestModel))-1
NEBIC <- length(coef(bestEBIC$BestModel))-1
NBICq <- length(coef(bestBICq$BestModel))-1
ans<-c(NAIC, NBIC, NEBIC, NBICq)
names(ans)<-c("AIC", "BIC", "BICg", "BICq")
ans
# AIC  BIC EBIC BICq
#   3    1    0    0

#Example 2. bestglm with BICq
#Find best model. Default is BICq with q=0.25
data(znuclear) #standardized data.
#Rest of examples assume this dataset is loaded.
out<-bestglm(znuclear, IC="BICq")
```

```
out
#The optimal range for q
out$Bestq
#The possible models that can be chosen
out$qTable
#The best models for each subset size
out$Subsets
#The overall best models
out$BestModels
#
#Example 3. Normal probability plot, residuals, best model
ans<-bestglm(znuclear, IC="BICq")
e<-resid(ans$BestModel)
qqnorm(e, ylab="residuals, best model")
#
#To save time, none of the remaining examples are run
## Not run:
#Example 4. bestglm, using EBIC, g=1
bestglm(znuclear, IC="BICg")
#EBIC with g=0.5
bestglm(znuclear, IC="BICg", t=0.5)
#
#Example 5. bestglm, CV
data(zprostate)
train<-(zprostate[zprostate[,10],])[,-10]
#the default CV method takes too long, set t=10 to do only
# 10 replications instead of the recommended 1000
bestglm(train, IC="CV", t=10)
bestglm(train, IC="CV", CVArgs=list(Method="HTF", K=10, REP=1))
#Compare with DH Algorithm. Normally set REP=100 is recommended.
bestglm(train, IC="CV", CVArgs=list(Method="DH", K=10, REP=1))
#Compare LOOCV
bestglm(train, IC="LOOCV")
#
#Example 6. Optimal q for manpower dataset
data(manpower)
out<-bestglm(manpower)
out$Bestq
#
#Example 7. Factors with more than 2 levels
data(AirQuality)
bestglm(AirQuality)
#
#Example 8. Logistic regression
data(SAheart)
bestglm(SAheart, IC="BIC", family=binomial)
#BIC agrees with backward stepwise approach
out<-glm(chd~., data=SAheart, family=binomial)
step(out, k=log(nrow(SAheart)))
#but BICq with q=0.25
bestglm(SAheart, IC="BICq", t=0.25, family=binomial)
#
#Cross-validation with glm
```

```
#make reproducible results
set.seed(33997711)
#takes about 15 seconds and selects 5 variables
bestglm(SAheart, IC="CV", family=binomial)
#about 6 seconds and selects 2 variables
bestglm(SAheart, IC="CV", CVArgs=list(Method="HTF", K=10, REP=1), family=binomial)
#Will produce an error -- NA
\dontrun{bestglm(SAheart, IC="CV", CVArgs=list(Method="DH", K=10, REP=1), family=binomial)}
\dontrun{bestglm(SAheart, IC="LOOCV", family=binomial)}
#
#Example 9. Model with no intercept term
X<-matrix(rnorm(200*3), ncol=3)
b<-c(0, 1.5, 0)
y<-X%*%b + rnorm(40)
Xy<-data.frame(as.matrix.data.frame(X), y=y)
bestglm(Xy, intercept=FALSE)

## End(Not run)
```

CVd *Cross-validation using delete-d method.*

Description

The delete-d method for cross-validation uses a random sample of d observations as the validation sample. This is repeated many times.

Usage

```
CVd(X, y, d = ceiling(n * (1 - 1/(log(n) - 1))), REP = 100, family = gaussian, ...)
```

Arguments

X	training inputs
y	training output
d	size of validation sample
REP	number of replications
family	glm family
...	optional arguments passed to glm or lm

Details

Shao (1993, 1997) suggested the delete-d algorithm implemented in this function. In this algorithm, a random sample of d observations are taken as the validation sample. This random sampling is repeated REP times. Shao (1997, p.234, eqn. 4.5 and p.236) suggests $d = n(1 - 1/(logn - 1))$, This is obtained by taking $\lambda_n = logn$ on page 236 (Shao, 1997). As shown in the table Shao's recommended choice of the d parameter corresponds to validation samples that are typically much larger that used in 10-fold or 5-fold cross-validation. LOOCV corresponds to d=1 only!

n	d	K=10	K=5
50	33	5	10
100	73	10	20
200	154	20	40
500	405	50	100
1000	831	100	200

Value

Vector of two components comprising the cross-validation MSE and its sd based on the MSE in each validation sample.

Author(s)

A.I. McLeod and C. Xu

References

Shao, Jun (1993). Linear Model Selection by Cross-Validation. Journal of the American Statistical Assocation 88, 486-494.

Shao, Jun (1997). An Asymptotic Theory for Linear Model Selection. Statistica Sinica 7, 221-264.

See Also

bestglm, CVHTF, CVDH, LOOCV

Examples

```
#Example 1. delete-d method
#For the training set, n=67. So 10-fold CV is like using delete-d
#with d=7, approximately.
data(zprostate)
train<-(zprostate[zprostate[,10],])[,-10]
X<-train[,1:2]
y<-train[,9]
set.seed(123321123)
CVd(X, y, d=7, REP=10)
#should set to 1000. Used 10 to save time in example.
```

CVDH *Adjusted K-fold Cross-Validation*

Description

An adjustment to K-fold cross-validation is made to reduce bias.

Usage

```
CVDH(X, y, K = 10, REP = 1)
```

Arguments

X	training inputs
y	training output
K	size of validation sample
REP	number of replications

Details

Algorithm 6.5 (Davison and Hinkley, p.295) is implemented.

Value

Vector of two components comprising the cross-validation MSE and its sd based on the MSE in each validation sample.

Author(s)

A.I. McLeod and C. Xu

References

Davison, A.C. and Hinkley, D.V. (1997). Bootstrap Methods and their Application. Cambridge University Press.

See Also

bestglm, CVHTF, CVd, LOOCV

Examples

```
#Example 1. Variability in 10-fold CV with Davison-Hartigan Algorithm.
#Plot the CVs obtained by using 10-fold CV on the best subset
#model of size 2 for the prostate data. We assume the best model is
#the model with the first two inputs and then we compute the CV's
#using 10-fold CV, 100 times. The result is summarized by a boxplot as well
#as the sd.
NUMSIM<-10
data(zprostate)
train<-(zprostate[zprostate[,10],])[,-10]
X<-train[,1:2]
y<-train[,9]
cvs<-numeric(NUMSIM)
set.seed(123321123)
for (isim in 1:NUMSIM)
    cvs[isim]<-CVDH(X,y,K=10,REP=1)[1]
summary(cvs)
```

CVHTF *K-fold Cross-Validation*

Description

K-fold cross-validation.

Usage

```
CVHTF(X, y, K = 10, REP = 1, family = gaussian, ...)
```

Arguments

X	training inputs
y	training output
K	size of validation sample
REP	number of replications
family	glm family
...	optional arguments passed to glm or lm

Details

HTF (2009) describe K-fold cross-validation. The observations are partitioned into K non-overlapping subsets of approximately equal size. Each subset is used as the validation sample while the remaining K-1 subsets are used as training data. When $K = n$, where n is the number of observations the algorithm is equivalent to leave-one-out CV. Normally $K = 10$ or $K = 5$ are used. When $K < n - 1$, their are may be many possible partitions and so the results of K-fold CV may vary somewhat depending on the partitions used. In our implementation, random partitions are used and we allow for many replications. Note that in the Shao's delete-d method, random samples are used to select the valiation data whereas in this method the whole partition is selected as random. This is acomplished using, `fold <-sample(rep(1:K,length=n))`. Then `fold` indicates each validation sample in the partition.

Value

Vector of two components comprising the cross-validation MSE and its sd based on the MSE in each validation sample.

Author(s)

A.I. McLeod and C. Xu

References

Hastie, T., Tibshirani, R. and Friedman, J. (2009). The Elements of Statistical Learning. 2nd Ed. Springer-Verlag.

See Also

bestglm, CVd, CVDH, LOOCV

Examples

```
#Example 1. 10-fold CV
data(zprostate)
train<-(zprostate[zprostate[,10],])[,-10]
X<-train[,1:2]
y<-train[,9]
CVHTF(X,y,K=10,REP=1)[1]
```

Detroit — *Detroit homicide data for 1961-73 used in the book Subset Regression by A.J. Miller*

Description

For convenience we have labelled the input variables 1 through 11 to be consistent with the notation used in Miller (2002). Only the first 11 variables were used in Miller's analyses. The best fitting subset regression with these 11 variables, uses only 3 inputs and has a residual sum of squares of 6.77 while using forward selection produces a best fit with 3 inputs with residual sum of squares 21.19. Backward selection and stagewise methods produce similar results. It is remarkable that there is such a big difference. Note that the usual forward and backward selection algorithms may fail since the linear regression using 11 variables gives essentially a perfect fit.

Usage

```
data(Detroit)
```

Format

A data frame with 13 observations on the following 14 variables.

`FTP.1` Full-time police per 100,000 population

`UEMP.2` Percent unemployed in the population

`MAN.3` Number of manufacturing workers in thousands

`LIC.4` Number of handgun licences per 100,000 population

`GR.5` Number of handgun registrations per 100,000 population

`CLEAR.6` Percent homicides cleared by arrests

`WM.7` Number of white males in the population

`NMAN.8` Number of non-manufacturing workers in thousands

`GOV.9` Number of government workers in thousands

`HE.10` Average hourly earnings

`WE.11` Average weekly earnings

ACC Death rate in accidents per 100,000 population

ASR Number of assaults per 100,000 population

HOM Number of homicides per 100,000 of population

Details

The data were orginally collected and discussed by Fisher (1976) but the complete dataset first appeared in Gunst and Mason (1980, Appendix A). Miller (2002) discusses this dataset throughout his book. The data were obtained from StatLib.

Source

http://lib.stat.cmu.edu/datasets/detroit

References

Fisher, J.C. (1976). Homicide in Detroit: The Role of Firearms. Criminology, vol.14, 387-400.

Gunst, R.F. and Mason, R.L. (1980). Regression analysis and its application: A data-oriented approach. Marcel Dekker.

Miller, A. J. (2002). Subset Selection in Regression. 2nd Ed. Chapman & Hall/CRC. Boca Raton.

Examples

```
#Detroit data example
data(Detroit)
#As in Miller (2002) columns 1-11 are used as inputs
p<-11
#For possible comparison with other algorithms such as LARS
#  it is preferable to work with the scaled inputs.
#From Miller (2002, Table 3.14), we see that the
#best six inputs are: 1, 2, 4, 6, 7, 11
X<-as.data.frame(scale(Detroit[,c(1,2,4,6,7,11)]))
y<-Detroit[,ncol(Detroit)]
Xy<-cbind(X,HOM=y)
#Use backward stepwise regression with BIC selects full model
out <- lm(HOM~., data=Xy)
step(out, k=log(nrow(Xy)))
#
#Same story with exhaustive search algorithm
out<-bestglm(Xy, IC="BIC")
out
#But many coefficients have p-values that are quite large considering
#  the selection bias. Note: 1, 6 and 7 are all about 5% only.
#We can use BICq to reduce the number of variables.
#The qTable let's choose q for other possible models,
out$qTable
#This suggest we try q=0.05 or q=0.0005
bestglm(Xy,IC="BICq", t=0.05)
bestglm(Xy,IC="BICq", t=0.00005)
#It is interesting that the subset model of size 2 is not a subset
```

```
# itself of the size 3 model. These results agree with
#Miller (2002, Table 3.14).
#
#Using delete-d CV with d=4 suggests variables 2,4,6,11
set.seed(1233211)
bestglm(Xy, IC="CV", CVArgs=list(Method="d", K=4, REP=50))
```

---

dgrid *Scaled Variables Dependency Plots: Output vs Inputs*

Description

A lattice grid plot is produced for the output vs. each input. The variables are scaled to have mean zero and variance one.

Usage

```
dgrid(XyDF, span=0.8)
```

Arguments

XyDF	Must be a dataframe with the last column corresponding to the output
span	smoothing parameter for loess

Value

a lattice plot

Author(s)

A. I. McLeod

See Also

pairs, splom,

Examples

```
data(mcdonald)
dgrid(mcdonald)
```

`Fires` *Forest fires in Montesinho natural park. Standardized inputs.*

Description

The forest fire data were collected during January 2000 to December 2003 for fires in the Montesinho natural park located in the northeast region of Portugal. The response variable of interest was area burned in ha. When the area burned as less than one-tenth of a hectare, the response variable as set to zero. In all there were 517 fires and 247 of them recorded as zero. The region was divided into a 10-by-10 grid with coordinates X and Y running from 1 to 9. The categorical variable xyarea indicates the region in this grid for the fire.

Usage

```
data(Fires)
```

Format

A data frame with 517 observations on the following 12 variables. All quantitative variables have been standardized.

`xyarea` a factor with 36 levels

`month` an ordered factor with 12 levels

`day` an ordered factor with 7 levels

`FFMC` fine fuel moisture code

`DMC` Duff moisture code

`DC` drought code

`ISI` initial spread index

`temp` average ambient temperature

`RH` a numeric vector

`wind` wind speed

`rain` rainfall

`lburned` log(x+1), x is burned area with x=0 for small fires

Details

The original data may be found at the website below as well as an analysis. The quantitative variables in this dataset have been standardized. For convenience, the original data is provided in `MontesinhoFires`.

Source

http://archive.ics.uci.edu/ml/datasets/Forest+Fires

References

P. Cortez and A. Morais, 2007. A Data Mining Approach to Predict Forest Fires using Meteorological Data. In J. Neves, M. F. Santos and J. Machado Eds., New Trends in Artificial Intelligence, Proceedings of the 13th EPIA 2007 - Portuguese Conference on Artificial Intelligence, December, Guimaraes, Portugal, pp. 512-523, 2007.

See Also

MontesinhoFires

Examples

```
data(Fires)
names(Fires)
#ANOVA for xyarea is significant at 1.1%.
summary(aov(lburned~xyarea, data=Fires))
```

---

fitted.pcreg *Fitted values in PCR and PLS.*

Description

The fitted values are returned given the output from pcreg.

Usage

```
## S3 method for class 'pcreg'
fitted(object, ...)
```

Arguments

object	object output
...	additional parameters

Details

Method function for pcreg.

Value

residuals

Author(s)

A. I. McLeod

See Also

pcreg, residuals.pcreg, plot.pcreg

Examples

```
fitted(pcreg(mcdonald, scale=TRUE))
```

`glmnetGridTable` *Multipanel Display and Table Glmnet CV Output.*

Description

Four panels.

Usage

```
glmnetGridTable(XyList, alpha = 0, nfolds=10, family = "gaussian")
```

Arguments

`XyList`	input
`alpha`	elastic net parameter
`nfolds`	Number of folds, K, in regularized K-fold CV, must be >3 and <=10.
`family`	distribution

Details

tba

Value

plot produced by side-effect. Table.

Note

Set random seed beforehand if you want reproducibility.

Author(s)

A. I. McLeod

See Also

`trainTestPartition`, `cv.glmnet`, `glmnet`, `predict.glmnet`

Examples

```
set.seed(7733551)
out <- trainTestPartition(mcdonald)
round(glmnetGridTable(out),4)
```

`glmnetPredict` *Glmnet Prediction Using CVAV.*

Description

Predict by averaging the predictions from cv.glmnet().

Usage

```
glmnetPredict(XyList, NREP = 15, alpha = 0, nfolds=10,
 family = c("gaussian", "binomial", "poisson", "multinomial"))
```

Arguments

`XyList`	list with components XyTr, XTr, yTr, XTe.
`NREP`	number of replications to use in average
`alpha`	elastic net parameter
`nfolds`	Number of folds, K, in regularized K-fold CV, must be >3 and <=10.
`family`	model

Value

vector with predictions

Author(s)

A. I. McLeod

See Also

`trainTestPartition`, `glmnetGridTable`, `glmnet`, `cv.glmnet`, `predict.glmnet`

Examples

```
set.seed(7733551)
out <- trainTestPartition(mcdonald)
round(glmnetGridTable(out),4)
yh <- glmnetPredict(out, NREP=5)
sqrt(mean((out$yTe - yh)^2))
```

grpregPredict *Predictions on Test Data with Grpreg*

Description

A dataframe is partitioned randomly into training and test samples. The function grpreg::grpreg() is used to fit the training data using Lasso, SCAD and MCP penalty functions. The BIC criterion is used to selecting the penalty parameter lambda.

Usage

```
grpregPredict(Xy, trainFrac = 2/3, XyList=NULL)
```

Arguments

Xy	a dataframe that may contain factor variables
trainFrac	the fraction of data to be used for training
XyList	instead of supplying Xy you can provide XyList.

Value

vector of RMSEs

See Also

glmnetPredict, glmnetGridTable, trainTestPartition, grpreg

Examples

```
grpregPredict(mcdonald)
```

hivif *Simulated Linear Regression (Train) with Nine Highly Correlated Inputs*

Description

The script that generated this data is given below.

Usage

```
data("hivif")
```

Format

A data frame with 1000 observations on the following 10 variables.

`x1` a numeric vector

`x2` a numeric vector

`x3` a numeric vector

`x4` a numeric vector

`x5` a numeric vector

`x6` a numeric vector

`x7` a numeric vector

`x8` a numeric vector

`x9` a numeric vector

`y` a numeric vector

Examples

```
#Simple example
data(hivif)
lm(y ~ ., data=hivif)
#
#This example shows how the original data was simulated and
#how additional test data may be simulated.
## Not run:
 set.seed(778851) #needed for original training data
 n <- 100
 p <- 9 #9 covariates plus intercept
 sig <- toeplitz(0.9^(0:(p-1)))
 X <- MASS::mvrnorm(n=n, rep(0, p), Sigma=sig)
 colnames(X) <- paste0("x", 1:p)
 b <- c(0,-0.3,0,0,-0.3,0,0,0.3,0.3) #
 names(b) <- paste0("x", 1:p)
 y <- 1 +  X
 Xy <- cbind(as.data.frame.matrix(X), y=y) #=hivif
#Test data
 nTe <- 10^3
 XTe <- MASS::mvrnorm(n=nTe, rep(0, p), Sigma=sig)
 colnames(XTe) <- paste0("x", 1:p)
  yTe <- 1 +  XTe
 XyTe <- cbind(as.data.frame.matrix(XTe), y=yTe) #test data
 ans <- lm(y ~ ., data=Xy) #fit training data
 mean((XyTe$y - predict(ans, newdata=XyTe))^2) #MSE on test data

## End(Not run)
```

`Iowa` *Iowa School Test*

Description

Dataset on poverty and academic performance.

Usage

```
data("Iowa")
```

Format

A data frame with 133 observations on the following 3 variables.

`City` a factor with 6 levels `Cedar Rapids` `Davenport` `Des Moines` `Iowa City` `Sioux City` `Waterloo`

`Poverty` percentage subsidized

`Test` achievement test score

Details

There are n=133 average test scores for schools in the K=6 largest cities. The test score offers a standardized measure of academic achievement. The purpose of the study is to investigate if there is a relationship between academic achievement, as measured by the test, and poverty. It is expected that students from economically disadvantaged backgrounds will do less well. Data on the average income in the school district was not available so a proxy variable for poverty was used. The percentage of students who received subsidized meals was available so this was used as the "Poverty" variable.

Source

Abraham and Ledholter, Introduction to Regression, Wiley.

Examples

```
data(Iowa)
table(Iowa$City)
```

LOOCV *Leave-one-out cross-validation*

Description

An observation is removed and the model is fit the the remaining data and this fit used to predict the value of the deleted observation. This is repeated, n times, for each of the n observations and the mean square error is computed.

Usage

```
LOOCV(X, y)
```

Arguments

X	training inputs
y	training output

Details

LOOCV for linear regression is exactly equivalent to the PRESS method suggested by Allen (1971) who also provided an efficient algorithm.

Value

Vector of two components comprising the cross-validation MSE and its sd based on the MSE in each validation sample.

Author(s)

A.I. McLeod and C. Xu

References

Hastie, T., Tibshirani, R. and Friedman, J. (2009). The Elements of Statistical Learning. 2nd Ed.

Allen, D.M. (1971). Mean Square Error of Prediction as a Criterion for Selecting Variables. Technometrics, 13, 469 -475.

See Also

bestglm, CVd, CVDH, CVHTF

Examples

```
#Example. Compare LOO CV with K-fold CV.
#Find CV MSE's for LOOCV and compare with K=5, 10, 20, 40, 50, 60
#Takes about 30 sec
## Not run:
 data(zprostate)
 train<-(zprostate[zprostate[,10],])[,-10]
 X<-train[,1:2]
 y<-train[,9]
 CVLOO<-LOOCV(X,y)
 KS<-c(5,10,20,40,50,60)
 nKS<-length(KS)
 cvs<-numeric(nKS)
 set.seed(1233211231)
 for (iK in 1:nKS)
    cvs[iK]<-CVDH(X,y,K=KS[iK],REP=10)[1]
 boxplot(cvs)
 abline(h=CVLOO, lwd=3, col="red")
 title(sub="Boxplot of CV's with K=5,10,20,40,50,60 and LOO CV in red")

## End(Not run)
```

---

manpower *Hospital manpower data*

Description

The goal of this study is to predict the manpower requirement as given in the output variable Hours given the five other input variables. Data is from Table 3.8 of Myers (1990). See also Examples 3.8, 4.5, 8.8.

Usage

```
data(manpower)
```

Format

A data frame with 17 observations. The output variable is Hours and the inputs are Load, Xray, BedDays, AreaPop and Stay. The site 1 through 17 is indicated by the row name.

`Load` a numeric vector

`Xray` a numeric vector

`BedDays` a numeric vector

`AreaPop` a numeric vector

`Stay` a numeric vector

`Hours` a numeric vector

Details

This data illustrates the multicollinearity problem and the use of VIF to identify it. It provides an illustrative example for ridge regression and more modern methods such as lasso and lars.

Source

Myers (1990) indicates the source was "Procedures and Analysis for Staffing Standards Development: Data/Regression Analysis Handbook", Navy Manpower and Material Analysis Center, San Diego, 1979.

References

Myers, R. (1990). Classical and Modern Regression with Applications. The Duxbury Advanced Series in Statistics and Decision Sciences. Boston: PWS-KENT Publishing Company.

Examples

```
data(manpower)
```

---

mcdonald *Pollution dataset from McDonald and Schwing (1973)*

Description

Regression data used to illustrate ridge regression

Usage

```
data("mcdonald")
```

Format

A data frame with 60 observations on the following 16 variables.

`PREC` Average annual precipitation in inches

`JANT` Average January temperature in degrees F

`JULT` Same for July

`OVR65` Percent of 1960 SMSA population aged 65 or older

`POPN` Average household size

`EDUC` Median school years completed by those over 22

`HOUS` Percent of housing units which are sound & with all facilities

`DENS` Population per sq. mile in urbanized areas, 1960

`NONW` Percent non-white population in urbanized areas, 1960

`WWDRK` Percent employed in white collar occupations

`POOR` Percent of families with income < $3000

HC Relative hydrocarbon pollution potential

NOX Same for nitric oxides

SOx Same for sulphur dioxide

HUMID Annual average percent relative humidity at 1pm

MORT Total age-adjusted mortality rate per 100,000

Details

Ridge regression example

Source

Gary C. McDonald and Richard C. Schwing (1973), Instabilities of Regression Estimates Relating Air Pollution to Mortality, Technometrics 15/3, 463-481.

Examples

```
data(mcdonald)
vifx(mcdonald[, -ncol(mcdonald)])
```

MontesinhoFires *Forest fires in Montesinho natural park*

Description

The forest fire data were collected during January 2000 to December 2003 for fires in the Montesinho natural park located in the northeast region of Portugal. The response variable of interest was area burned in ha. When the area burned as less than one-tenth of a hectare, the response variable as set to zero. In all there were 517 fires and 247 of them recorded as zero. The region was divided into a 10-by-10 grid with coordinates X and Y running from 1 to 9.

Usage

```
data(MontesinhoFires)
```

Format

A data frame with 517 observations on the following 13 variables.

X X coordinate for region, 0-10

Y X coordinate for region, 0-10

month an ordered factor with 12 levels

day an ordered factor with 7 levels

FFMC fine fuel moisture code

DMC Duff moisture code

`DC` drought code

`ISI` initial spread index

`temp` average ambient temperature

`RH` a numeric vector

`wind` wind speed

`rain` rainfall

`burned` area burned in hectares

Details

This is the original data taken from the website below.

Source

http://archive.ics.uci.edu/ml/datasets/Forest+Fires

References

P. Cortez and A. Morais, 2007. A Data Mining Approach to Predict Forest Fires using Meteorological Data. In J. Neves, M. F. Santos and J. Machado Eds., New Trends in Artificial Intelligence, Proceedings of the 13th EPIA 2007 - Portuguese Conference on Artificial Intelligence, December, Guimaraes, Portugal, pp. 512-523, 2007.

See Also

`Fires`

Examples

```
data(MontesinhoFires)
names(MontesinhoFires)
data(Fires)
names(Fires)
#Anova for month
summary(aov(burned~month, data=MontesinhoFires))
```

NNPredict *Nearest Neighbour Regression Prediction*

Description

Given training/test data in the predictions on the test data computed. L1, L2 and correlation distances may be used. The data is sphered prior to making the NN predictions.

Usage

```
NNPredict(XyList, dist = c("L2", "COR", "L1"))
```

Arguments

XyList	list with six elements
dist	distance used

Value

vector of predictions

Author(s)

A. I. McLeod

See Also

sphereX

Examples

```
AQ <- airquality[complete.cases(airquality),c(2,3,4,1)]
XyList <- trainTestPartition(AQ)
NNPredict(XyList)
```

---

oneSDRule *Utility function. Implements the 1-sd rule.*

Description

The CV and its standard devation are provided for a range of models ordered by the number of parameters estimated.

Usage

```
oneSDRule(CVout)
```

Arguments

CVout	A matrix with two columns. First column is the CV and second, its sd. Row ordering is from fewest parameter to most.

Value

The row corresponding to the best model.

Author(s)

A.I. McLeod and C. Xu

References

Hastie, T., Tibshirani, R. and Friedman, J. (2009). The Elements of Statistical Learning. 2nd Ed. Springer-Verlag.

Examples

```
CV<-c(1.4637799,0.7036285,0.6242480,0.6069406,0.6006877,0.6005472,0.5707958,
      0.5907897,0.5895489)
CVsd<-c(0.24878992,0.14160499,0.08714908,0.11376041,0.08522291,
 0.11897327,0.07960879,0.09235052,0.12860983)
CVout <- matrix(c(CV,CVsd), ncol=2)
oneSDRule(CVout)
```

`pcreg` *Principal Component and Partial Least Squares Regression*

Description

Regression using the principal components or latent variables as inputs. The best model is selected using components 1, 2, ..., r, where r, the number of components to use is determined by the AIC or BIC.

Usage

```
pcreg(Xy, scale = TRUE, method = c("PC", "LV"), ic = c("BIC", "AIC"))
```

Arguments

`Xy`	dataframe with variable names in columns
`scale`	Whether or not to scale. Default is TRUE.
`method`	either principal components, "PC", or partial least squares latent variables, "LV"
`ic`	"BIC" or "AIC"

Value

An S3 class list "pcreg" with components

`lmfit`	lm model
`PLSFit`	column sd
`Z`	matrix of principal components or latent vector
`method`	'pcr' or 'pls'

Author(s)

A. I. McLeod

See Also

predict.pcreg, summary.pcreg, plot.pcreg, fitted.pcreg, residuals.pcreg

Examples

```
pcreg(mcdonald, scale=TRUE, method="PC")
pcreg(mcdonald, scale=TRUE, method="LV")
```

---

plot.pcreg *Diagnostic plots for PCR and PLS*

Description

Diagnostic plots available with lm-objects are provided.

Usage

```
## S3 method for class 'pcreg'
plot(x, ...)
```

Arguments

x	x output from pcreg(). It has S3 class 'pcreg'.
...	additional parameters

Details

See plot method for S3 class 'lm'.

Value

Nothing. The plot is produced.

Author(s)

A. I. McLeod

See Also

pcreg, fitted.pcreg, residuals.pcreg

Examples

```
ans <- pcreg(mcdonald, scale=TRUE)
plot(ans)
```

plot1SDRule *Plot Regularization Path and One Standard Deviation Rule*

Description

Takes input either matrix with 2 columns or output from caret::train() and produces a plot showing the best model selected using the 1 SD rule.

Usage

```
plot1SDRule(ans, main = "", sub = "", xlab = "df", ylab = "EPE")
```

Arguments

ans	matrix or output from train
main	optional plot title
sub	optional plot subtitle
xlab	optional x-axis label
ylab	optional y-axis label

Value

tuning parameter value for best model

Author(s)

A. I. McLeod

References

Hastie, Tibsharani and Friedman, "Elements of Statistical Learning".

See Also

oneSDRule

Examples

```
CV<-c(1.4637799,0.7036285,0.6242480,0.6069406,0.6006877,0.6005472,0.5707958,
      0.5907897,0.5895489)
CVsd<-c(0.24878992,0.14160499,0.08714908,0.11376041,0.08522291,
 0.11897327,0.07960879,0.09235052,0.12860983)
CVout <- matrix(c(CV,CVsd), ncol=2)
oneSDRule(CVout)
```

predict.pcreg *Predict Method for Pcreg.*

Description

Prediction for models fit using `pcreg()`.

Usage

```
## S3 method for class 'pcreg'
predict(object, newdata, ...)
```

Arguments

`object`	the S3 class object produced as output from the function pcreg()
`newdata`	dataframe with new data and with same column names as used in the original argument to pcreg.
`...`	additional arguments

Details

The prediction method, `predict.mvr()`, which is available in the pls package is used. We take advantage of this since it avoids fussing with scaling issues since it is automatically handled for us by `predict.mvr()`

Value

the predicted values

Author(s)

A. I. McLeod

See Also

`predict.pcreg`, `summary.pcreg`, `plot.pcreg`, `fitted.pcreg`, `residuals.pcreg`

Examples

```
XyList <- trainTestPartition(mcdonald)
XyTr <- XyList$XyTr
XyTe <- XyList$XyTe
ans <- pcreg(XyTr, scale=TRUE)
predict(ans, newdata=XyTe)
```

print.bestglm *Print method for 'bestglm' object*

Description

A brief description of the best fit is given.

Usage

```
## S3 method for class 'bestglm'
print(x, ...)
```

Arguments

x	Output from the bestglm function
...	optional arguments

Value

No value. Output to terminal only.

Author(s)

A.I. McLeod and C. Xu

See Also

bestglm, summary.bestglm

Examples

```
data(znuclear)
bestglm(znuclear)
```

print.pcreg *Print method for 'pcreg' object*

Description

A brief description of the best fit is given.

Usage

```
## S3 method for class 'pcreg'
print(x, ...)
```

Arguments

x	Output from the pcreg function
...	optional arguments

Value

No value. Output to terminal only.

Author(s)

A.I. McLeod and C. Xu

See Also

pcreg, summary.pcreg

Examples

```
pcreg(znuclear, scale=TRUE)
```

---

residuals.pcreg *Residuals Fitted PCR or PLS*

Description

The residuals from a model fitted using pcreg are returned.

Usage

```
## S3 method for class 'pcreg'
residuals(object, ...)
```

Arguments

object	object output
...	additional parameters

Details

Method function for pcreg.

Value

residuals

Author(s)

A. I. McLeod

See Also

pcreg, fitted, plot

Examples

```
resid(pcreg(mcdonald, scale=TRUE))
```

---

rubber *Abrasion loss for various hardness and tensile strength*

Description

The data come from an experiment to investigate how the resistance of rubber to abrasion is affected by the hardness of the rubber and its tensile strength.

Usage

```
data(rubber)
```

Format

A data frame with 30 observations on the following 3 variables.

`hardness` hardness in degree Shore

`tensile.strength` tensile strength in kg per square meter

`abrasion.loss` abrasion loss in gram per hour

`ts.low` tensile strength minus the breakpoint 180 km/m^2

`ts.high` tensile strength minus the breakpoint 180 km/m^2

Source

Hand, D.J., Daly, F., Lunn, A.D., McConway, K.J. and Ostrowski, E. (1993). A Handbook of Small Datasets. Chapman and Hall.

References

Cleveland, W. S. (1993). Visualizing data. Hobart Press, Summit: New Jersey.

Davies, O.L. and Goldsmith, P.L.(1972) Statistical methods in Research and Production.

Examples

```
data(rubber)
ans <- lm(abrasion.loss~hardness+tensile.strength, data=rubber)
```

SAheart *South African Hearth Disease Data*

Description

A retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa.

Usage

```
data(SAheart)
```

Format

A data frame with 462 observations on the following 10 variables.

sbp systolic blood pressure

tobacco cumulative tobacco (kg)

ldl low density lipoprotein cholesterol

adiposity a numeric vector

famhist family history of heart disease, a factor with levels `Absent` `Present`

typea type-A behavior

obesity a numeric vector

alcohol current alcohol consumption

age age at onset

chd response, coronary heart disease

Details

A retrospective sample of males in a heart-disease high-risk region of the Western Cape, South Africa. There are roughly two controls per case of CHD. Many of the CHD positive men have undergone blood pressure reduction treatment and other programs to reduce their risk factors after their CHD event. In some cases the measurements were made after these treatments. These data are taken from a larger dataset, described in Rousseauw et al, 1983, South African Medical Journal.

Source

Rousseauw, J., du Plessis, J., Benade, A., Jordaan, P., Kotze, J. and Ferreira, J. (1983). Coronary risk factor screening in three rural communities, South African Medical Journal 64: 430–436.

Examples

```
data(SAheart)
str(SAheart)
summary(SAheart)
```

Shao *Simulated Regression Data*

Description

Data a simulation study reported by Shao (1993, Table 1). The linear regression model Shao (1993, Table 2) reported 4 simulation experiments using 4 different values for the regression coefficients:

$$y = 2 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + e,$$

where e is an independent normal error with unit variance.

The four regression coefficients for the four experiments are shown in the table below,

Experiment	β_2	β_3	β_4	β_5
1	0	0	4	0
2	0	0	4	8
3	9	0	4	8
4	9	6	4	8

The table below summarizes the probability of correct model selection in the experiment reported by Shao (1993, Table 2). Three model selection methods are compared: LOOCV (leave-one-out CV), CV(d=25) or the delete-d method with d=25 and APCV which is a very efficient computation CV method but specialized to the case of linear regression.

Experiment	LOOCV	CV(d=25)	APCV
1	0.484	0.934	0.501
2	0.641	0.947	0.651
3	0.801	0.965	0.818
4	0.985	0.948	0.999

The CV(d=25) outperforms LOOCV in all cases and it also outforms APCV by a large margin in Experiments 1, 2 and 3 but in case 4 APCV is slightly better.

Usage

```
data(Shao)
```

Format

A data frame with 40 observations on the following 4 inputs.

`x2` a numeric vector

`x3` a numeric vector

`x4` a numeric vector

`x5` a numeric vector

Source

Shao, Jun (1993). Linear Model Selection by Cross-Validation. Journal of the American Statistical Assocation 88, 486-494.

Examples

```
#In this example BICq(q=0.25) selects the correct model but BIC does not
data(Shao)
X<-as.matrix.data.frame(Shao)
b<-c(0,0,4,0)
set.seed(123321123)
#Note: matrix multiplication must be escaped in Rd file
y<-X%*%b+rnorm(40)
Xy<-data.frame(Shao, y=y)
bestglm(Xy)
bestglm(Xy, IC="BICq")
```

---

sphereX *Sphere Data Matrix*

Description

The data matrix is scaled and sphered so it is orthonormal. The Cholesky decomposition is used.

Usage

```
sphereX(X)
```

Arguments

X	X rectangular data matrix

Value

sphered matrix

Author(s)

A. I. McLeod

See Also

scale, NNPredict

Examples

```
data(longley)
longley.x <- data.matrix(longley[, 1:6])
sphereX(longley.x)
```

summary.bestglm *summary of 'bestglm' object*

Description

An analysis of deviance and a likelihood-ratio test with p-value. The p-value is greatly exagerated due to selection.

Usage

```
## S3 method for class 'bestglm'
summary(object, SubsetsQ=FALSE, ...)
```

Arguments

object	Output from the bestglm function
SubsetsQ	List best subsets of each size
...	optional arguments

Value

No value. Output to terminal only.

Author(s)

A.I. McLeod and C. Xu

See Also

bestglm, print.bestglm

Examples

```
data(znuclear)
summary(bestglm(znuclear))
#
#find statistical signficance of overall regression
data(Fires)
summary(bestglm(Fires, IC="BICq", t=1))
```

summary.pcreg *Summary Method for Pcreg.*

Description

The summary is based on the summary method for S3 class 'lm'.

Usage

```
## S3 method for class 'pcreg'
summary(object, ...)
```

Arguments

object	object output
...	additional parameters

Details

Method function for pcreg.

Value

residuals

Note

The standard errors and p-values are wrong due to selection bias.

Author(s)

A. I. McLeod

See Also

pcreg, fitted, plot

Examples

```
resid(pcreg(mcdonald, scale=TRUE))
```

`trainTestPartition` *Partition Dataframe into Train/Test Samples*

Description

Dataframe used to create training and test datasets using specified fraction for the training sample. The data matrix must be comprised of continuous variables only (no factors).

Usage

```
trainTestPartition(Xy, trainFrac = 2/3)
```

Arguments

`Xy`	Dataframe with column names, last column is the response variable and others are the regression input variables. The data matrix must be comprised of continuous variables only (no factors).
`trainFrac`	Fraction to be used for the training sample.

Value

A list with components

`XyTr`	Training dataframe.
`XTr`	Matrix, input training variables.
`yTr`	Vector, output training variable.
`XyTe`	Training dataframe.
`XTe`	Matrix, input test variables.
`yTe`	Vector, output test variable.
`XyTr`	Training dataframe.
`XyTr`	Training dataframe.
`XyTr`	Training dataframe.

Author(s)

A. I. McLeod

Examples

```
set.seed(7733551)
out <- trainTestPartition(mcdonald)
round(glmnetGridTable(out),4)
```

vifx *Variance Inflation Factor for a Design Matrix*

Description

Barplot of the VIF is produced

Usage

```
vifx(X)
```

Arguments

X	A design matrix

Details

The VIF are the diagonal elements in the inverse $t(X*)X*$, where X* is the rescaled design matrix.

Value

vector with VIF's

Author(s)

A. I. McLeod

References

Marquardt, D. W. (1970). Generalized Inverses, Ridge Regression, Biased Linear Estimation, and Nonlinear Estimation. Technometrics 12(3), 591-612.

Examples

```
data(mcdonald)
vifx(mcdonald[, -ncol(mcdonald)])
```

znuclear *Nuclear plant data. Quantitative inputs logged and standardized.*

Description

Data on 32 nuclear power plants. The response variable is cost and there are ten covariates.

Usage

```
data(znuclear)
```

Format

A data frame with 32 observations on the following 12 variables. All quantitative variables, except date, have been logged and standardized to have mean 0 and variance 1.

`date` Quantitative covariate. The date on which the construction permit was issued. The data are measured in years since January 1 1990 to the nearest month.

`T1` Quantitative covariate. The time between application for and issue of the construction permit.

`T2` Quantitative covariate. The time between issue of operating license and construction permit.

`capacity` Quantitative covariate. The net capacity of the power plant (MWe).

`PR` Binary covariate. Value 1, indicates the prior existence of a LWR plant at the same site.

`NE` Binary covariate, located in North-East USA

`CT` Binary covariate, presence of cooling tower

`BW` Binary covariate, where 1 indicates that the nuclear steam supply system was manufactured by Babcock-Wilcox.

`N` Quantitative covariate. The cumulative number of power plants constructed by each architect-engineer.

`PT` Binary covariate, partial turnkey guarantee.

`cost` Outcome. The capital cost of construction in millions of dollars adjusted to 1976 base.

Details

Davison (2003) explores fitting models to this data using forward and backward stepwise regression. In this modelling logs of quantiative variablesare used. We have also standardized this data to facilitate comparison with other techniques such as LARS and principal component regression.

Davison and Hinkley (1997, Example 6.8, 6.10, 6.12) use this data in a series of examples. Example 6.8: estimation of prediction error. Example 6.10: prediction error using cross-validation and bootstrapping. Example 6.12: subset model selection using cross-validation.

Source

Obtained from the CRAN package boot.

References

Davison, A. C. (2003). Statistical Models. Cambridge: Cambridge University Press.

Davison, A.C. and Hinkley, D.V. (1997). Bootstrap Methods and their Application. Cambridge University Press.

Examples

```
data(znuclear)
bestglm(znuclear, IC="BICq")
```

`zprostate` *Prostate cancer data. Standardized.*

Description

Data with 8 inputs and one output used to illustrate the prediction problem and regression in the textbook of Hastie, Tibshirani and Freedman (2009).

Usage

```
data(zprostate)
```

Format

A data frame with 97 observations, 9 inputs and 1 output. All input variables have been standardized.

`lcavol` log-cancer volume

`lweight` log prostate weight

`age` age in years

`lbph` log benign prostatic hyperplasia

`svi` seminal vesicle invasion

`lcp` log of capsular penetration

`gleason` Gleason score

`pgg45` percent of Gleascores 4/5

`lpsa` Outcome. Log of PSA

`train` TRUE or FALSE

Details

A study of 97 men with prostate cancer examined the correlation between PSA (prostate specific antigen) and a number of clinical measurements: lcavol, lweight, lbph, svi, lcp, gleason, pgg45

References

Hastie, Tibshirani & Friedman. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Ed. Springer.

Examples

```
#Prostate data. Table 3.3 HTF.
data(zprostate)
#full dataset
trainQ<-zprostate[,10]
train <-zprostate[trainQ,-10]
test <-zprostate[!trainQ,-10]
ans<-lm(lpsa~., data=train)
sig<-summary(ans)$sigma
yHat<-predict(ans, newdata=test)
yTest<-zprostate$lpsa[!trainQ]
TE<-mean((yTest-yHat)^2)
#subset
ansSub<-bestglm(train, IC="BICq")$BestModel
sigSub<-summary(ansSub)$sigma
yHatSub<-predict(ansSub, newdata=test)
TESub<-mean((yTest-yHatSub)^2)
m<-matrix(c(TE,sig,TESub,sigSub), ncol=2)
dimnames(m)<-list(c("TestErr","Sd"),c("LS","Best"))
m
```

Index