Package: cggd (via r-universe)

November 15, 2026

Version 0.8.9001

Title Continuous Generalized Gradient Descent

Maintainer Cun-Hui Zhang <czhang@stat.rutgers.edu>

Depends R (>= 2.10), grDevices, graphics, stats, utils

Description Efficient procedures for fitting an entire regression sequences with different model types. Compatibility updates by Renova.

License GPL (>= 2)

Repository https://renova-r.r-universe.dev

Date/Publication 2026-11-15 20:40:57 UTC

RemoteUrl https://github.com/renova-r/cggd

RemoteRef renova-0.8.9001

RemoteSha c0f5e3e32b5222246e029ab93dbae120266f9ea6

Contents

cggd 2
cv.cggd 3
housing 4
OJ 5
plot.cggd 6
predict.cggd 7
Wine 8

Index **9**

cggd *Fits Continuous Generalized Gradient Descent models*

Description

These CGGD models all provide the entire sequence of coefficients and fits, to the least squares fit. This package is based on the lars package.

Usage

```
cggd(x, y, beta0 = rep(0,2), kmax = 300,
                      TRR=FALSE, t0 = 1, TRACE=FALSE,
                      alpha1 = 0, alpha2 = 0, w = 1,
                      tau = 1, tautil = -1, eps = -1, fctr=1e8)
```

Arguments

x	matrix of predictors
y	response
beta0	initial coefficient values
kmax	the maximum number of steps
TRACE	If TRUE, cggd prints out its progress
TRR	If TRUE builds a Threshold Ridge Regression (TRR) model. If FALSE builds a Functional Gradient Descent Kernel model.
t0	The initial time in the TRR model.
alpha1, alpha2, w	Values used in kernel generation.
tau, tautil, eps	Values used in variable selection.
fctr	Used to tune the search routine. Smaller values tighten the search (more sampling).

Details

CGGD is a general method for performing different types of regressions while continuously varying regularization and other parameters (similar to LARS but broader). It returns the models fit along the continuous paths of parameter modification. Note that y is automatically centered and x is centered and normalized.

Value

A "cggd" object is returned, for which plot and predict methods exist.

Some significant componenets of the object are:

beta.tk	The coefficients of the model at the different steps (k).
tk	The *time* at the different steps (k).
a.set.tk	The active coefficients of the model at the different steps (k).

Author(s)

Cun-Hui Zhang and Ofer Melnik

References

Cun-Hui Zhang (2007) "Continuous Generalized Gradient Descent" *Journal of Computational and Graphical Statistics* ; see also http://stat.rutgers.edu/~cunhui/software/CGGD.html.

See Also

plot, and predict methods for cggd, and cv.cggd

Examples

```
data(housing)
attach(housing)
object <- cggd(x,y)
plot(object)
object2 <- cggd(x,y,TRR=TRUE)
plot(object2)
detach(housing)
```

---

cv.cggd *Computes K-fold cross-validated error curve for cggd*

Description

Computes the K-fold cross-validated mean squared prediction error for cggd.

Usage

```
cv.cggd(x, y, nfolds = 6, kmax=40 ,
           trace = FALSE, plot.it = TRUE, se=TRUE, ...)
```

Arguments

x	Input to cggd
y	Input to cggd
nfolds	Number of folds
kmax	Max number of iterations per model
trace	Show computations
plot.it	Plot it
se	Include standard error bands
...	Additional arguments to cggd

Value

cv	The CV loss curve at each value of k
cv.error	The standard error of the CV curve

Author(s)

Cun-Hui Zhang and Ofer Melnik

References

Cun-Hui Zhang (2007) "Continuous Generalized Gradient Descent" *Journal of Computational and Graphical Statistics* ; see also http://stat.rutgers.edu/~cunhui/software/CGGD.html.

Examples

```
data(Wine)
attach(Wine)
cv.cggd(x.learning,y.learning,kmax=40,fctr=1e3)
detach(Wine)
```

---

housing *Covariates related to housing prices*

Description

The housing data frame has 506 rows and 13 attributes.

Usage

```
data(housing)
```

Format

This data frame contains the following columns:

x a matrix with 13 columns of attributes

y a numeric vector of median value of owner-occupied homes in 1000's

Details

The x matrix has been standardized to have unit L2 norm in each column and zero mean. The matrix x2 consists of x plus certain interactions.

Source

(a) Origin: This dataset was taken from the StatLib library which is maintained at Carnegie Mellon University. (b) Creator: Harrison, D. and Rubinfeld, D.L. 'Hedonic prices and the demand for clean air', J. Environ. Economics & Management, vol.5, 81-102, 1978. (c) Date: July 7, 1993 Can be found on the UCI data repository ftp://ftp.ics.uci.edu/pub/machine-learning-databases/housing/

`OJ` *OJ*

Description

The `OJ` datafile has both learning and test data.

Usage

```
data(OJ)
```

Format

This data file contains the following items:

x.learning 150 spectra (lines), sampled with 700 points (columns)

y.learning Corresponding level of saccharose, in a column vector (150 lines).

x.test 68 spectra (lines), sampled with 700 points (columns)

y.test Corresponding level of saccharose, in a column vector (68 lines).

Details

The goal is to estimate the level of saccharose of an orange juice from its observed near-infrared spectrum.

Source

This database has been provided by Prof. Marc Meurens, Université catholique de Louvain, BNUT unit. Please acknowledge the origin of this database if you use it and/or publish about it. It can be found at http://www.ucl.ac.be/mlg/index.php?page=DataBases

`plot.cggd` *Plot method for cggd objects*

Description

Produce a plot of a cggd fit. The default is a complete coefficient path.

Usage

```
## S3 method for class 'cggd'
plot(x, steps=5, breaks = TRUE, first_k=1,last_k=Inf,
     xvar=c("step","t"),omit.zeros = TRUE, eps = 1e-10, ...)
```

Arguments

`x`	cggd object
`steps`	Number of steps to sample coefficients between each model iteration.
`breaks`	If `TRUE`, then vertical lines are drawn at each break point in the piecewise linear coefficient paths
`first_k`	The starting k, model iteration, to plot from.
`last_k`	The final k, model iteration, to plot from.
`xvar`	The type of x variable against which to plot. `xvar=step` plots against iteration steps of the model (default). `xvar=t` plots against the *time* along the coefficient path.
`omit.zeros`	When the number of variables is much greater than the number of observations, many coefficients will never be nonzero; this logical (default `TRUE`) avoids plotting these zero coefficents
`eps`	Definition of zero above, default is `1e-10`
`...`	Additonal arguments for generic plot. Can be used to set xlims, change colors, line widths, etc

Value

NULL

Author(s)

Cun-Hui Zhang and Ofer Melnik

References

Cun-Hui Zhang (2007) "Continuous Generalized Gradient Descent" *Journal of Computational and Graphical Statistics* ; see also http://stat.rutgers.edu/~cunhui/software/CGGD.html.

Examples

```
data(housing)
attach(housing)
object <- cggd(x,y)
plot(object)
detach(housing)
```

`predict.cggd` *Make predictions or extract coefficients from a fitted cggd model*

Description

While cggd produces the entire path of solutions, predict.cggd allows one to extract a prediction at a particular point along the path.

Usage

```
 ## S3 method for class 'cggd'
predict(object, newx, t, type = c("fit", "coefficients"), mode=c("k","t"), ...)
```

Arguments

`object`	A fitted cggd object
`newx`	If type="fit", then newx should be the x values at which the fit is required. If type="coefficients", then newx can be omitted.
`t`	A value specifying the desired model. Its values depends on the mode argument. By default (mode="k"), *t* should take on values between 0 and max k (e.g., a step of 1.3 means .3 of the way between step 1 and 2.)
`type`	If type="fit", predict returns the fitted value. If type="coefficients", predict returns the coefficients.
`mode`	Mode="k" means the t= argument indexes the cggd step number, and the coefficients will be returned corresponding to the values at step t. If mode="t", then t is a time on the coefficient path.
`...`	additional parameters.

Value

Either the fitted value, or a vector of coefficients.

Author(s)

Cun-Hui Zhang and Ofer Melnik

References

Cun-Hui Zhang (2007) "Continuous Generalized Gradient Descent" *Journal of Computational and Graphical Statistics* ; see also http://stat.rutgers.edu/~cunhui/software/CGGD.html.

See Also

plot, cggd, cv.cggd

Examples

```
data(housing)
attach(housing)
object <- cggd(x,y)
coef4.1 <- predict(object, t=4.1,type="coefficients", mode="k")
detach(housing)
```

Wine *Wine*

Description

The Wine data file has both learning and test data.

Usage

```
data(Wine)
```

Format

This data file contains the following items:

x.learning 94 spectra (lines), sampled with 256 points (columns)

y.learning Corresponding level of alcohol, in a column vector (94 lines).

x.test 30 spectra (lines), sampled with 256 points (columns)

y.test Corresponding level of alcohol, in a column vector (30 lines).

Details

To estimate the level of alcohol of a wine from its observed mean infrared spectrum.

Source

This database has been provided by Prof. Marc Meurens, Université catholique de Louvain, BNUT unit. Please acknowledge the origin of this database if you use it and/or publish about it. It can be found at http://www.ucl.ac.be/mlg/index.php?page=DataBases

Index

∗ **datasets**
housing, 4
OJ, 5
Wine, 8
∗ **hplot**
plot.cggd, 6
∗ **methods**
plot.cggd, 6
predict.cggd, 7
∗ **regression**
cggd, 2
cv.cggd, 3
predict.cggd, 7

cggd, 2
cv.cggd, 3

housing, 4

OJ, 5

plot.cggd, 6
predict.cggd, 7

Wine, 8