

Package ‘dbstats’

July 22, 2025

Type Package

Title Distance-Based Statistics

Version 2.0.2

Date 2024-01-26

Author Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@irb.barcelona.org>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>.

Maintainer Eva Boj <evaboj@ub.edu>

Description Prediction methods where explanatory information is coded as a matrix of distances between individuals. Distances can either be directly input as a distances matrix, a squared distances matrix, an inner-products matrix or computed from observed predictors.

License GPL-2

Repository CRAN

Depends R (>= 4.3), cluster, pls

Suggests proxy

NeedsCompilation no

Date/Publication 2024-01-26 18:40:03 UTC

Contents

dbstats-package	2
as.D2	4
as.Gram	4
D2toDist	5
D2toG	6
dbglm	7
dblm	11
dbplsr	15
disttoD2	18
GtoD2	19
ldbgglm	20
ldblm	24
plot.dblm	28

plot.dbplsr	30
plot.ldblm	32
predict.dbglm	33
predict.dblm	35
predict.dbplsr	37
predict.ldbglm	38
predict.ldblm	40
summary.dbglm	42
summary.dblm	44
summary.dbplsr	45
summary.ldbglm	47
summary.ldblm	48
Index	50

dbstats-package	<i>Distance-based statistics (dbstats)</i>
-----------------	--

Description

This package contains functions for distance-based prediction methods.

These are methods for prediction where predictor information is coded as a matrix of distances between individuals.

In the currently implemented methods the response is a univariate variable as in the ordinary linear model or in the generalized linear model.

Distances can either be directly input as an distances matrix, a squared distances matrix, an inner-products matrix (see [GtoD2](#)) or computed from observed explanatory variables.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by *Z* or *z*, from *Euclidean coordinates* which we denote by *X* or *x*. For explanation on the meaning of both terms see the bibliography references below.

Observed explanatory variables *z* are possibly a mixture of continuous and qualitative explanatory variables or more general quantities.

dbstats does not provide specific functions for computing distances, depending instead on other functions and packages, such as:

- [dist](#) in the **stats** package.
- [dist](#) in the **proxy** package. When the **proxy** package is loaded, its [dist](#) function supersedes the one in the **stats** package.
- [daisy](#) in the **cluster** package. Compared to both instances of [dist](#) above whose input must be numeric variables, the main feature of [daisy](#) is its ability to handle other variable types as well (e.g. nominal, ordinal, (a)symmetric binary) even when different types occur in the same data set.

Actually the last statement is not hundred percent true: it refers only to the default behaviour of both [dist](#) functions, whereas the [dist](#) function in the **proxy** package can evaluate distances between observations with a user-provided function, entered as a parameter, hence it can deal with any type of data. See the examples in [pr_DB](#).

Functions of **dbstats** package:

Linear and local linear models with a continuous response:

- [dblm](#) for distance-based linear models.
- [ldblm](#) for local distance-based linear models.
- [dbpls](#) for distance-based partial least squares.

Generalized linear and local generalized linear models with a numeric response:

- [dbglm](#) for distance-based generalized linear models.
- [ldbgglm](#) for local distance-based generalized linear models.

Details

Package:	dbstats
Type:	Package
Version:	2.0.2
Date:	2024-01-26
License:	GPL-2
LazyLoad:	yes

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Caballe, A., Delicado P, Esteve, A., Fortiana J (2016). *Global and local distance-based generalized linear models*. TEST 25, 170-195.
- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Implementing PLS for distance-based regression: computational issues*. Computational Statistics 22, 237-248.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

`as.D2`*D2 objects*

Description

`as.D2` attempts to turn its argument into a D2 class object.

`is.D2` tests if its argument is a (strict) D2 class object.

Usage

```
as.D2(x)
```

```
is.D2(x)
```

Arguments

`x` an R object.

Value

An object of class D2 containing the squared distances matrix between individuals.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[D2toG](#), [disttoD2](#), [D2toDist](#) and [GtoD2](#) for conversions.

`as.Gram`*Gram objects*

Description

`as.Gram` attempts to turn its argument into a Gram class object.

`is.Gram` tests if its argument is a (strict) Gram class object.

Usage

```
as.Gram(x)
```

```
is.Gram(x)
```

Arguments

x an R object.

Value

A Gram class object. Weighted centered inner products matrix of the squared distances matrix.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[D2toG](#), [disttoD2](#), [D2toDist](#) and [GtoD2](#) for conversions.

D2toDist

Distance conversion: D2 to dist

Description

Converts D2 class object into dist class object.

Usage

```
D2toDist(D2)
```

Arguments

D2 D2 object. Squared distances matrix between individuals.

Value

An object of class dist. See function [dist](#) for details.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[GtoD2](#)
[D2toG](#)
[disttoD2](#)

Examples

```
X <- matrix(rnorm(100*3),nrow=100)
distance <- daisy(X,"manhattan")
D2 <- disttoD2(distance)
distance2 <- D2toDist(D2)
```

D2toG

*Distance conversion: D2 to G***Description**

Converts D2 class object into Gram class object.

Usage

```
D2toG(D2,weights)
```

Arguments

D2	D2 object. Squared distances matrix between individuals.
weights	an optional numeric vector of weights. By default all individuals have the same weight.

Value

An object of class Gram containing the Doubly centered inner product matrix of D2.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[GtoD2](#)
[disttoD2](#)
[D2toDist](#)

Examples

```
X <- matrix(rnorm(100*3),nrow=100)
D2 <- as.matrix(dist(X)^2)
class(D2) <- "D2"
G <- D2toG(D2,weights=NULL)
```

dbglm

Distance-based generalized linear models

Description

dbglm is a variety of generalized linear model where explanatory information is coded as distances between individuals. These distances can either be computed from observed explanatory variables or directly input as a squared distances matrix.

Response and link function as in the glm function for ordinary generalized linear models.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by Z or z , from *Euclidean coordinates* which we denote by X or x . For explanation on the meaning of both terms see the bibliography references below.

Usage

```
## S3 method for class 'formula'
dbglm(formula, data, family=gaussian, method="GCV", full.search=TRUE, ...,
      metric="euclidean", weights, maxiter=100, eps1=1e-10,
      eps2=1e-10, rel.gvar=0.95, eff.rank=NULL, offset, mustart=NULL, range.eff.rank)

## S3 method for class 'dist'
dbglm(distance,y,family=gaussian, method="GCV", full.search=TRUE, weights,
      maxiter=100,eps1=1e-10,eps2=1e-10,rel.gvar=0.95,eff.rank=NULL,
      offset,mustart=NULL, range.eff.rank,...)

## S3 method for class 'D2'
dbglm(D2,y,...,family=gaussian, method="GCV", full.search=TRUE, weights,maxiter=100,
      eps1=1e-10,eps2=1e-10,rel.gvar=0.95,eff.rank=NULL,offset,
      mustart=NULL, range.eff.rank)

## S3 method for class 'Gram'
dbglm(G,y,...,family=gaussian, method="GCV", full.search=TRUE, weights,maxiter=100,
      eps1=1e-10,eps2=1e-10,rel.gvar=0.95,eff.rank=NULL,
      offset,mustart=NULL, range.eff.rank)
```

Arguments

formula	an object of class <code>formula</code> . A formula of the form $y \sim Z$. This argument is a remnant of the <code>glm</code> function, kept for compatibility.
data	an optional data frame containing the variables in the model (both response and explanatory variables, either the observed ones, Z , or a Euclidean configuration X).
y	(required if no formula is given as the principal argument). Response (dependent variable) must be numeric, factor, matrix or data.frame.
distance	a dist or dissimilarity class object. See functions <code>dist</code> in the package stats and <code>daisy</code> in the package cluster.

D2	a D2 class object. Squared distances matrix between individuals. See the Details section in dblm to learn the usage.
G	a Gram class object. Doubly centered inner product matrix of the squared distances matrix D2. See details in dblm .
family	a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See family for details of family functions.)
metric	metric function to be used when computing distances from observed explanatory variables. One of "euclidean" (the default), "manhattan", or "gower".
weights	an optional numeric vector of prior weights to be used in the fitting process. By default all individuals have the same weight.
method	sets the method to be used in deciding the <i>effective rank</i> , which is defined as the number of linearly independent Euclidean coordinates used in prediction. There are five different methods: "AIC", "BIC", "GCV"(default), "eff.rank" and "rel.gvar". GCV take the effective rank minimizing a cross-validatory quantity. AIC and BIC take the effective rank minimizing, respectively, the Akaike or Bayesian Information Criterion (see AIC for more details).
full.search	sets which optimization procedure will be used to minimize the modelling criterion specified in method. Needs to be specified only if method is "AIC", "BIC" or "GCV". If full.search=TRUE, <i>effective rank</i> is set to its global best value, after evaluating the criterion for all possible ranks. Potentially too computationally expensive. If full.search=FALSE, the optimize function is called. Then computation time is shorter, but the result may be found a local minimum.
maxiter	maximum number of iterations in the iterated dblm algorithm. (Default = 100)
eps1	stopping criterion 1, "DevStat": convergence tolerance eps1, a positive (small) number; the iterations converge when $ dev - dev_{old} / (dev) < eps1$. Stationarity of deviance has been attained.
eps2	stopping criterion 2, "mustat": convergence tolerance eps2, a positive (small) number; the iterations converge when $ \mu - \mu_{old} / (\mu) < eps2$. Stationarity of fitted.values mu has been attained.
rel.gvar	relative geometric variability (a real number between 0 and 1). In each dblm iteration, take the lowest effective rank, with a relative geometric variability higher or equal to rel.gvar. Default value (rel.gvar=0.95) uses the 95% of the total variability.
eff.rank	integer between 1 and the number of observations minus one. Number of Euclidean coordinates used for model fitting in each dblm iteration. If specified its value overrides rel.gvar. When eff.rank=NULL (default), calls to dblm are made with method=rel.gvar.
offset	this can be used to specify an a priori known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases.
mustart	starting values for the vector of means.
range.eff.rank	vector of size two defining the range of values for the effective rank with which the dblm iterations will be evaluated (must be specified when method is "AIC", "BIC" or "GCV"). The range should be restrict between $c(1, n-1)$.
...	arguments passed to or from other methods to the low level.

Details

The various possible ways for inputting the model explanatory information through distances, or their squares, etc., are the same as in [dblm](#).

For gamma distributions, the domain of the canonical link function is not the same as the permitted range of the mean. In particular, the linear predictor might be negative, obtaining an impossible negative mean. Should that event occur, `dbglm` stops with an error message. Proposed alternative is to use a non-canonical link function.

Value

A list of class `dbglm` containing the following components:

<code>residuals</code>	the working residuals, that is the <code>dblm</code> residuals in the last iteration of <code>dblm</code> fit.
<code>fitted.values</code>	the fitted mean values, results of final <code>dblm</code> iteration.
<code>family</code>	the family object used.
<code>deviance</code>	measure of discrepancy or badness of fit. Proportional to twice the difference between the maximum achievable log-likelihood and that achieved by the current model.
<code>aic.model</code>	a version of Akaike's Information Criterion. Equal to minus twice the maximized log-likelihood plus twice the number of parameters. Computed by the <code>aic</code> component of the family. For binomial and Poisson families the dispersion is fixed at one and the number of parameters is the number of coefficients. For gaussian, Gamma and inverse gaussian families the dispersion is estimated from the residual deviance, and the number of parameters is the number of coefficients plus one. For a gaussian family the MLE of the dispersion is used so this is a valid value of AIC, but for Gamma and inverse gaussian families it is not. For families fitted by quasi-likelihood the value is NA.
<code>bic.model</code>	a version of the Bayesian Information Criterion. Equal to minus twice the maximized log-likelihood plus the logarithm of the number of observations by the number of parameters (see, e.g., Wood 2006).
<code>gcv.model</code>	a version of the Generalized Cross-Validation Criterion. We refer to Wood (2006) pp. 177-178 for details.
<code>null.deviance</code>	the deviance for the null model. The null model will include the offset, and an intercept if there is one in the model. Note that this will be incorrect if the link function depends on the data other than through the fitted mean: specify a zero offset to force a correct calculation.
<code>iter</code>	number of Fisher scoring (<code>dblm</code>) iterations.
<code>prior.weights</code>	the original weights.
<code>weights</code>	the working weights, that are the weights in the last iteration of <code>dblm</code> fit.
<code>df.residual</code>	the residual degrees of freedom.
<code>df.null</code>	the residual degrees of freedom for the null model.
<code>y</code>	the response vector used.
<code>convcrit</code>	convergence criterion. One of: "DevStat" (stopping criterion 1), "muStat" (stopping criterion 2), "maxiter" (maximum allowed number of iterations has been exceeded).

H	hat matrix projector of the last db1m iteration.
rel.gvar	the relative geometric variability in the last db1m iteration.
eff.rank	the working effective rank, that is the eff.rank in the last db1m iteration.
varmu	vector of estimated variance of each observation.
dev.resids	deviance residuals
call	the matched call.

Objects of class "dbglm" are actually of class c("dbglm", "db1m"), inheriting the [plot.db1m](#) method from class "db1m".

Note

When the Euclidean distance is used the dbglm model reduces to the generalized linear model (glm).

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Caballe, A., Delicado P, Esteve, A., Fortiana J (2016). *Global and local distance-based generalized linear models*. TEST 25, 170-195.
- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.
- Wood SN (2006). *Generalized Additive Models: An Introduction with R*. Chapman & Hall, Boca Raton.

See Also

[summary.dbglm](#) for summary.
[plot.dbglm](#) for plots.
[predict.dbglm](#) for predictions.
[db1m](#) for distance-based linear models.

Examples

```
## CASE POISSON
z <- rnorm(100)
y <- rpois(100, exp(1+z))
glm1 <- glm(y ~z, family = poisson(link = "log"))
D2 <- as.matrix(dist(z))^2
class(D2) <- "D2"
dbglm1 <- dbglm(D2,y,family = poisson(link = "log"), method="rel.gvar")

plot(z,y)
points(z,glm1$fitted.values,col=2)
points(z,dbglm1$fitted.values,col=3)
sum((glm1$fitted.values-y)^2)
sum((dbglm1$fitted.values-y)^2)

## CASE BINOMIAL
y <- rbinom(100, 1, plogis(z))
# needs to set a starting value for the next fit
glm2 <- glm(y ~z, family = binomial(link = "logit"))
D2 <- as.matrix(dist(z))^2
class(D2) <- "D2"
dbglm2 <- dbglm(D2,y,family = binomial(link = "logit"), method="rel.gvar")

plot(z,y)
points(z,glm2$fitted.values,col=2)
points(z,dbglm2$fitted.values,col=3)
sum((glm2$fitted.values-y)^2)
sum((dbglm2$fitted.values-y)^2)
```

dblm

Distance-based linear model

Description

dblm is a variety of linear model where explanatory information is coded as distances between individuals. These distances can either be computed from observed explanatory variables or directly input as a squared distances matrix. The response is a continuous variable as in the ordinary linear model. Since distances can be computed from a mixture of continuous and qualitative explanatory variables or, in fact, from more general quantities, dblm is a proper extension of lm.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by Z or z , from *Euclidean coordinates* which we denote by X or x . For explanation on the meaning of both terms see the bibliography references below.

Usage

```
## S3 method for class 'formula'
dblm(formula,data,...,metric="euclidean",method="OCV",full.search=TRUE,
      weights,rel.gvar=0.95,eff.rank)
```

```
## S3 method for class 'dist'
dblm(distance,y,...,method="OCV",full.search=TRUE,
      weights,rel.gvar=0.95,eff.rank)

## S3 method for class 'D2'
dblm(D2,y,...,method="OCV",full.search=TRUE,weights,rel.gvar=0.95,
      eff.rank)

## S3 method for class 'Gram'
dblm(G,y,...,method="OCV",full.search=TRUE,weights,rel.gvar=0.95,
      eff.rank)
```

Arguments

formula	an object of class formula . A formula of the form $y \sim Z$. This argument is a remnant of the lm function, kept for compatibility.
data	an optional data frame containing the variables in the model (both response and explanatory variables, either the observed ones, Z , or a Euclidean configuration X).
y	(required if no formula is given as the principal argument). Response (dependent variable) must be numeric, matrix or data.frame.
distance	a dist or dissimilarity class object. See functions dist in the package stats and daisy in the package cluster.
D2	a D2 class object. Squared distances matrix between individuals.
G	a Gram class object. Doubly centered inner product matrix of the squared distances matrix D2.
metric	metric function to be used when computing distances from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
method	sets the method to be used in deciding the <i>effective rank</i> , which is defined as the number of linearly independent Euclidean coordinates used in prediction. There are six different methods: "AIC", "BIC", "OCV" (default), "GCV", "eff.rank" and "rel.gvar". OCV and GCV take the effective rank minimizing a cross-validated quantity (either ocv or gcv). AIC and BIC take the effective rank minimizing, respectively, the Akaike or Bayesian Information Criterion (see AIC for more details). The optimization procedure to be used in the above four methods can be set with the full.search optional parameter. When method is eff.rank, the effective rank is explicitly set by the user through the eff.rank optional parameter which, in this case, becomes mandatory. When method is rel.gvar, the fraction of the data <i>geometric variability</i> for model fitting is explicitly set by the user through the rel.gvar optional parameter which, in this case, becomes mandatory.
full.search	sets which optimization procedure will be used to minimize the modelling criterion specified in method. Needs to be specified only if method is "AIC", "BIC", "OCV" or "GCV". If full.search=TRUE, <i>effective rank</i> is set to its global best value, after evaluating the criterion for all possible ranks. Potentially too computationally expensive. If full.search=FALSE, the optimize function is called. Then computation time is shorter, but the result may be found a local minimum.

<code>weights</code>	an optional numeric vector of weights to be used in the fitting process. By default all individuals have the same weight.
<code>rel.gvar</code>	relative geometric variability (real between 0 and 1). Take the lowest effective rank with a relative geometric variability higher or equal to <code>rel.gvar</code> . Default value (<code>rel.gvar=0.95</code>) uses a 95% of the total variability. Applies only <code>rel.gvar</code> if <code>method="rel.gvar"</code> .
<code>eff.rank</code>	integer between 1 and the number of observations minus one. Number of Euclidean coordinates used for model fitting. Applies only if <code>method="eff.rank"</code> .
<code>...</code>	arguments passed to or from other methods to the low level.

Details

The `dblm` model uses the distance matrix between individuals to find an appropriate prediction method. There are many ways to compute and calculate this matrix, besides the three included as parameters in this function. Several packages in R also study this problem. In particular [dist](#) in the package `stats` and [daisy](#) in the package `cluster` (the three metrics in `dblm` call the `daisy` function).

Another way to enter a distance matrix to the model is through an object of class "D2" (containing the squared distances matrix). An object of class "dist" or "dissimilarity" can easily be transformed into one of class "D2". See [disttoD2](#). Reciprocally, an object of class "D2" can be transformed into one of class "dist". See [D2toDist](#).

S3 method `Gram` uses the Doubly centered inner product matrix $G=XX'$. Its also easily to transformed into one of class "D2". See [D2toG](#) and [GtoD2](#).

The weights array is adequate when responses for different individuals have different variances. In this case the weights array should be (proportional to) the reciprocal of the variances vector.

When using `method="eff.rank"` or `method="rel.gvar"`, a compromise between possible consequences of a bad choice has to be reached. If the rank is too large, the model can be overfitted, possibly leading to an increased prediction error for new cases (even though R^2 is higher). On the other hand, a small rank suggests a model inadequacy (R^2 is small). The other four methods are less error prone (but still they do not guarantee good predictions).

Value

A list of class `dblm` containing the following components:

<code>residuals</code>	the residuals (response minus fitted values).
<code>fitted.values</code>	the fitted mean values.
<code>df.residuals</code>	the residual degrees of freedom.
<code>weights</code>	the specified weights.
<code>y</code>	the response used to fit the model.
<code>H</code>	the hat matrix projector.
<code>call</code>	the matched call.
<code>rel.gvar</code>	the relative geometric variability, used to fit the model.
<code>eff.rank</code>	the dimensions chosen to estimate the model.

ocv	the ordinary cross-validation estimate of the prediction error.
gcv	the generalized cross-validation estimate of the prediction error.
aic	the Akaike Value Criterium of the model (only if method="AIC").
bic	the Bayesian Value Criterium of the model (only if method="BIC").

Note

When the Euclidean distance is used the dblm model reduces to the linear model (lm).

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Caballe, A., Delicado P, Esteve, A., Fortiana J (2016). *Global and local distance-based generalized linear models*. TEST 25, 170-195.
- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[summary.dblm](#) for summary.
[plot.dblm](#) for plots.
[predict.dblm](#) for predictions.
[ldblm](#) for distance-based local linear models.

Examples

```
# easy example to illustrate usage of the dblm function
n <- 100
p <- 3
k <- 5

Z <- matrix(rnorm(n*p),nrow=n)
b <- matrix(runif(p)*k,nrow=p)
s <- 1
e <- rnorm(n)*s
```

```

y <- Z%*%b + e

D<-dist(Z)

dblm1 <- dblm(D,y)
lm1 <- lm(y~Z)
# the same fitted values with the lm
mean(lm1$fitted.values-dblm1$fitted.values)

```

dbplsr

Distance-based partial least squares regression

Description

dbplsr is a variety of partial least squares regression where explanatory information is coded as distances between individuals. These distances can either be computed from observed explanatory variables or directly input as a squared distances matrix.

Since distances can be computed from a mixture of continuous and qualitative explanatory variables or, in fact, from more general quantities, dbplsr is a proper extension of pls.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by Z or z , from *Euclidean coordinates* which we denote by X or x . For explanation on the meaning of both terms see the bibliography references below.

Usage

```

## S3 method for class 'formula'
dbplsr(formula,data,...,metric="euclidean",
        method="ncomp",weights,ncomp)

## S3 method for class 'dist'
dbplsr(distance,y,...,weights,ncomp=ncomp,method="ncomp")

## S3 method for class 'D2'
dbplsr(D2,y,...,weights,ncomp=ncomp,method="ncomp")

## S3 method for class 'Gram'
dbplsr(G,y,...,weights,ncomp=ncomp,method="ncomp")

```

Arguments

formula	an object of class <code>formula</code> . A formula of the form $y \sim Z$. This argument is a remnant of the <code>pls</code> function, kept for compatibility.
data	an optional data frame containing the variables in the model (both response and explanatory variables, either the observed ones, Z , or a Euclidean configuration X).

<code>y</code>	(required if no formula is given as the principal argument). Response (dependent variable) must be numeric, matrix or data.frame.
<code>distance</code>	a dist or dissimilarity class object. See functions dist in the package stats and daisy in the package cluster.
<code>D2</code>	a D2 class object. Squared distances matrix between individuals.
<code>G</code>	a Gram class object. Weighted centered inner products matrix of the squared distances matrix D2. See details in dblm .
<code>metric</code>	metric function to be used when computing distances from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
<code>method</code>	sets the method to be used in deciding how many components needed to fit the best model for new predictions. There are five different methods, "AIC", "BIC", "OCV", "GCV" and "ncomp" (default). OCV and GCV find the number of components that minimizes the Cross-validation coefficient (ocv or gcv). AIC and BIC find the number of components that minimizes the Akaike or Bayesian Information Criterion (see AIC for more details).
<code>weights</code>	an optional numeric vector of weights to be used in the fitting process. By default all individuals have the same weight.
<code>ncomp</code>	the number of components to include in the model.
<code>...</code>	arguments passed to or from other methods to the low level.

Details

Partial least squares (PLS) is a method for constructing predictive models when the factors (Z) are many and highly collinear. A PLS model will try to find the multidimensional direction in the Z space that explains the maximum multidimensional variance direction in the Y space. `dbplsr` is particularly suited when the matrix of predictors has more variables than observations. By contrast, standard regression (`dblm`) will fail in these cases.

The various possible ways for inputting the model explanatory information through distances, or their squares, etc., are the same as in [dblm](#).

The number of components to fit is specified with the argument `ncomp`.

Value

A list of class `dbplsr` containing the following components:

<code>residuals</code>	a list containing the residuals (response minus fitted values) for each iteration.
<code>fitted.values</code>	a list containing the fitted values for each iteration.
<code>fk</code>	a list containing the scores for each iteration.
<code>bk</code>	regression coefficients. <code>fitted.values = fk*bk</code>
<code>Pk</code>	orthogonal projector on the one-dimensional linear space by <code>fk</code> .
<code>ncomp</code>	number of components included in the model.
<code>ncomp.opt</code>	optimum number of components according to the selected method.
<code>weights</code>	the specified weights.
<code>method</code>	the using method.

<code>y</code>	the response used to fit the model.
<code>H</code>	the hat matrix projector.
<code>G0</code>	initial weighted centered inner products matrix of the squared distance matrix.
<code>Gk</code>	weighted centered inner products matrix in last iteration.
<code>gvar</code>	total weighted geometric variability.
<code>gvec</code>	the diagonal entries in <code>G0</code> .
<code>gvar.iter</code>	geometric variability for each iteration.
<code>ocv</code>	the ordinary cross-validation estimate of the prediction error.
<code>gcv</code>	the generalized cross-validation estimate of the prediction error.
<code>aic</code>	the Akaike Value Criterium of the model.
<code>bic</code>	the Bayesian Value Criterium of the model.

Note

When the Euclidean distance is used the `dbplsr` model reduces to the traditional partial least squares (`pls`).

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Implementing PLS for distance-based regression: computational issues*. Computational Statistics 22, 237-248.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[summary.dbplsr](#) for summary.
[plot.dbplsr](#) for plots.
[predict.dbplsr](#) for predictions.

Examples

```
#require(pls)
library(pls)
data(yarn)
## Default methods:
yarn.dbpls <- dbpls(density ~ NIR, data = yarn, ncomp=6, method="GCV")
```

disttoD2

Distance conversion: dist to D2

Description

Converts dist or dissimilarity class object into D2 class object.

Usage

```
disttoD2(distance)
```

Arguments

distance dist or dissimilarity class object. See functions [dist](#) in the package stats and [daisy](#) in the package cluster.

Value

An object of class D2 containing the squared distances matrix between individuals.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[GtoD2](#)
[D2toG](#)
[D2toDist](#)

Examples

```
X <- matrix(rnorm(100*3),nrow=100)
distance <- daisy(X,"manhattan")
D2 <- disttoD2(distance)
```

GtoD2*Distance conversion: dist to D2*

Description

Converts Gram class object into D2 class object

Usage

```
GtoD2(G)
```

Arguments

G Gram class object. Weighted centered inner products matrix of the squared distances matrix.

Value

An object of class D2 containing the squared distances matrix between individuals.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

See Also

[D2toG](#)
[disttoD2](#)
[D2toDist](#)

Examples

```
X <- matrix(rnorm(100*3),nrow=100)
D2 <- as.matrix(dist(X)^2)
class(D2) <- "D2"
G <- D2toG(D2,weights=NULL)
class(G) <- "Gram"
D22 <- GtoD2(G)
```

Description

ldbglm is a localized version of a distance-based generalized linear model. As in the global model [dbglm](#), explanatory information is coded as distances between individuals.

Neighborhood definition for localizing is done by the (semi)metric `dist1` whereas a second (semi)metric `dist2` (which may coincide with `dist1`) is used for distance-based prediction. Both `dist1` and `dist2` can either be computed from observed explanatory variables or directly input as a squared distances matrix or as a Gram matrix. Response and link function are as in the `dbglm` function for ordinary generalized linear models. The model allows for a mixture of continuous and qualitative explanatory variables or, in fact, from more general quantities such as functional data.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by Z or z , from *Euclidean coordinates* which we denote by X or x . For explanation on the meaning of both terms see the bibliography references below.

Usage

```
## S3 method for class 'formula'
ldbglm(formula,data,...,family=gaussian(),kind.of.kernel=1,
        metric1="euclidean",metric2=metric1,method.h="GCV",weights,
        user.h=NULL,h.range=NULL,noh=10,k.knn=3,
        rel.gvar=0.95,eff.rank=NULL,maxiter=100,eps1=1e-10,
        eps2=1e-10)

## S3 method for class 'dist'
ldbglm(dist1,dist2=dist1,y,family=gaussian(),kind.of.kernel=1,
        method.h="GCV",weights,user.h=quantile(dist1,.25),
        h.range=quantile(as.matrix(dist1),c(.05,.5)),noh=10,k.knn=3,
        rel.gvar=0.95,eff.rank=NULL,maxiter=100,eps1=1e-10,eps2=1e-10,...)

## S3 method for class 'D2'
ldbglm(D2.1,D2.2=D2.1,y,family=gaussian(),kind.of.kernel=1,
        method.h="GCV",weights,user.h=quantile(D2.1,.25)^.5,
        h.range=quantile(as.matrix(D2.1),c(.05,.5))^.5,noh=10,
        k.knn=3,rel.gvar=0.95,eff.rank=NULL,maxiter=100,eps1=1e-10,
        eps2=1e-10,...)

## S3 method for class 'Gram'
ldbglm(G1,G2=G1,y,family=gaussian(),kind.of.kernel=1,user.h=NULL,
        family=gaussian(),method.h="GCV",weights,h.range=NULL,noh=10,
        k.knn=3,rel.gvar=0.95,eff.rank=NULL,maxiter=100,eps1=1e-10,
        eps2=1e-10,...)
```

Arguments

formula	an object of class formula . A formula of the form $y \sim Z$. This argument is a remnant of the loess function, kept for compatibility.
data	an optional data frame containing the variables in the model (both response and explanatory variables, either the observed ones, Z , or a Euclidean configuration X).
y	(required if no formula is given as the principal argument). Response (dependent variable) must be numeric, matrix or data.frame.
dist1	a dist or dissimilarity class object. Distances between observations, used for neighborhood localizing definition. Weights for observations are computed as a decreasing function of their dist1 distances to the neighborhood center, e.g. a new observation whose response has to be predicted. These weights are then entered to a dbglm , where distances are evaluated with dist2.
dist2	a dist or dissimilarity class object. Distances between observations, used for fitting dbglm . Default dist2=dist1.
D2.1	a D2 class object. Squared distances matrix between individuals. One of the alternative ways of entering distance information to a function. See the Details section in dblm . See above dist1 for explanation of its role in this function.
D2.2	a D2 class object. Squared distances between observations. One of the alternative ways of entering distance information to a function. See the Details section in dblm . See above dist2 for explanation of its role in this function. Default D2.2=D2.1.
G1	a Gram class object. Doubly centered inner product matrix associated with the squared distances matrix D2.1.
G2	a Gram class object. Doubly centered inner product matrix associated with the squared distances matrix D2.2. Default G2=G1
family	a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See family for details of family functions.)
kind.of.kernel	integer number between 1 and 6 which determines the user's choice of smoothing kernel. (1) Epanechnikov (Default), (2) Biweight, (3) Triweight, (4) Normal, (5) Triangular, (6) Uniform.
metric1	metric function to be used when computing dist1 from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
metric2	metric function to be used when computing dist2 from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
method.h	sets the method to be used in deciding the <i>optimal bandwidth</i> h . There are four different methods, AIC, BIC, GCV (default) and user.h. GCV take the optimal bandwidth minimizing a cross-validated quantity. AIC and BIC take the optimal bandwidth minimizing, respectively, the Akaike or Bayesian Information Criterion (see AIC for more details). When method.h is user.h, the bandwidth is explicitly set by the user through the user.h optional parameter which, in this case, becomes mandatory.

<code>weights</code>	an optional numeric vector of weights to be used in the fitting process. By default all individuals have the same weight.
<code>user.h</code>	global bandwidth <code>user.h</code> , set by the user, controlling the size of the local neighborhood of <code>Z</code> . Smoothing parameter (Default: 1st quartile of all the distances $d(i,j)$ in <code>dist1</code>). Applies only if <code>method.h="user.h"</code> .
<code>h.range</code>	a vector of length 2 giving the range for automatic bandwidth choice. (Default: quantiles 0.05 and 0.5 of $d(i,j)$ in <code>dist1</code>).
<code>noh</code>	number of bandwidth <code>h</code> values within <code>h.range</code> for automatic bandwidth choice (if <code>method.h!="user.h"</code>).
<code>k.knn</code>	minimum number of observations with positive weight in neighborhood localizing. To avoid runtime errors due to a too small bandwidth originating neighborhoods with only one observation. By default <code>k.nn=3</code> .
<code>rel.gvar</code>	relative geometric variability (a real number between 0 and 1). In each <code>dblm</code> iteration, take the lowest effective rank, with a relative geometric variability higher or equal to <code>rel.gvar</code> . Default value (<code>rel.gvar=0.95</code>) uses the 95% of the total variability.
<code>eff.rank</code>	integer between 1 and the number of observations minus one. Number of Euclidean coordinates used for model fitting in each <code>dblm</code> iteration. If specified its value overrides <code>rel.gvar</code> . When <code>eff.rank=NULL</code> (default), calls to <code>dblm</code> are made with <code>method=rel.gvar</code> .
<code>maxiter</code>	maximum number of iterations in the iterated <code>dblm</code> algorithm. (Default = 100)
<code>eps1</code>	stopping criterion 1, "DevStat": convergence tolerance <code>eps1</code> , a positive (small) number; the iterations converge when $ \text{dev} - \text{dev}_{\text{old}} / (\text{dev}) < \text{eps1}$. Stationarity of deviance has been attained.
<code>eps2</code>	stopping criterion 2, "mustat": convergence tolerance <code>eps2</code> , a positive (small) number; the iterations converge when $ \mu - \mu_{\text{old}} / (\mu) < \text{eps2}$. Stationarity of fitted values <code>mu</code> has been attained.
<code>...</code>	arguments passed to or from other methods to the low level.

Details

The various possible ways for inputting the model explanatory information through distances, or their squares, etc., are the same as in [dblm](#).

The set of bandwidth `h` values checked in automatic bandwidth choice is defined by `h.range` and `noh`, together with `k.knn`. For each `h` in it a local generalized linear model is fitted and the optimal `h` is decided according to the statistic specified in `method.h`.

`kind.of.kernel` designates which kernel function is to be used in determining individual weights from `dist1` values. See [density](#) for more information.

For gamma distributions, the domain of the canonical link function is not the same as the permitted range of the mean. In particular, the linear predictor might be negative, obtaining an impossible negative mean. Should that event occur, `ldbglm` stops with an error message. Proposed alternative is to use a non-canonical link function.

Value

A list of class `ldbg1m` containing the following components:

<code>residuals</code>	the residuals (response minus fitted values).
<code>fitted.values</code>	the fitted mean values.
<code>h.opt</code>	the optimal bandwidth <code>h</code> used in the fitting proces (if <code>method.h!=user.h</code>).
<code>family</code>	the <code>family</code> object used.
<code>y</code>	the response variable used.
<code>S</code>	the Smoother hat projector.
<code>weights</code>	the specified weights.
<code>call</code>	the matched call.
<code>dist1</code>	the distance matrix (object of class <code>"D2"</code> or <code>"dist"</code>) used to calculate the weights of the observations.
<code>dist2</code>	the distance matrix (object of class <code>"D2"</code> or <code>"dist"</code>) used to fit the <code>dbg1m</code> .

Objects of class `"ldbg1m"` are actually of class `c("ldbg1m", "ldblm")`, inheriting the `plot.ldblm` and `summary.ldblm` method from class `"ldblm"`.

Note

Model fitting is repeated `n` times (`n`= number of observations) for each bandwidth (`noh*n` times). For a `noh` too large or a sample with many observations, the time of this function can be very high.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Caballe, A., Delicado P, Esteve, A., Fortiana J (2016). *Global and local distance-based generalized linear models*. TEST 25, 170-195.
- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbglm](#) for distance-based generalized linear models.
[ldblm](#) for local distance-based linear models.
[summary.ldbglm](#) for summary.
[plot.ldbglm](#) for plots.
[predict.ldbglm](#) for predictions.

Examples

```

# example of ldbgml usage
z <- rnorm(100)
y <- rbinom(100, 1, plogis(z))
D2 <- as.matrix(dist(z))^2
class(D2) <- "D2"

# Distance-based generalized linear model
dbglm2 <- dbglm(D2,y,family=binomial(link = "logit"), method="rel.gvar")
# Local Distance-based generalized linear model
ldbgml2 <- ldbgml(D2,y=y,family=binomial(link = "logit"),noh=3)

# check the difference of both
sum((y-ldbgml2$fit)^2)
sum((y-dbglm2$fit)^2)
plot(z,y)
points(z,ldbgml2$fit,col=3)
points(z,dbglm2$fit,col=2)

```

ldblm

*Local distance-based linear model***Description**

ldblm is a localized version of a distance-based linear model. As in the global model dblm, explanatory information is coded as distances between individuals.

Neighborhood definition for localizing is done by the (semi)metric `dist1` whereas a second (semi)metric `dist2` (which may coincide with `dist1`) is used for distance-based prediction. Both `dist1` and `dist2` can either be computed from observed explanatory variables or directly input as a squared distances matrix or as a Gram matrix. The response is a continuous variable as in the ordinary linear model. The model allows for a mixture of continuous and qualitative explanatory variables or, in fact, from more general quantities such as functional data.

Notation convention: in distance-based methods we must distinguish *observed explanatory variables* which we denote by Z or z , from *Euclidean coordinates* which we denote by X or x . For explanation on the meaning of both terms see the bibliography references below.

Usage

```
## S3 method for class 'formula'
ldblm(formula,data,...,kind.of.kernel=1,
      metric1="euclidean",metric2=metric1,method.h="GCV",weights,
      user.h=NULL,h.range=NULL,noh=10,k.knn=3,rel.gvar=0.95,eff.rank=NULL)

## S3 method for class 'dist'
ldblm(dist1,dist2=dist1,y,kind.of.kernel=1,
      method.h="GCV",weights,user.h=quantile(dist1,.25),
      h.range=quantile(as.matrix(dist1),c(.05,.5)),noh=10,
      k.knn=3,rel.gvar=0.95,eff.rank=NULL,...)

## S3 method for class 'D2'
ldblm(D2.1,D2.2=D2.1,y,kind.of.kernel=1,method.h="GCV",
      weights,user.h=quantile(D2.1,.25)^.5,
      h.range=quantile(as.matrix(D2.1),c(.05,.5))^.5,noh=10,k.knn=3,
      rel.gvar=0.95,eff.rank=NULL,...)

## S3 method for class 'Gram'
ldblm(G1,G2=G1,y,kind.of.kernel=1,method.h="GCV",
      weights,user.h=NULL,h.range=NULL,noh=10,k.knn=3,rel.gvar=0.95,
      eff.rank=NULL,...)
```

Arguments

formula	an object of class formula . A formula of the form $y \sim Z$. This argument is a remnant of the loess function, kept for compatibility.
data	an optional data frame containing the variables in the model (both response and explanatory variables, either the observed ones, Z , or a Euclidean configuration X).
y	(required if no formula is given as the principal argument). Response (dependent variable) must be numeric, matrix or data.frame.
dist1	a dist or dissimilarity class object. Distances between observations, used for neighborhood localizing definition. Weights for observations are computed as a decreasing function of their dist1 distances to the neighborhood center, e.g. a new observation whose response has to be predicted. These weights are then entered to a dblm , where distances are evaluated with dist2.
dist2	a dist or dissimilarity class object. Distances between observations, used for fitting dblm . Default dist2=dist1.
D2.1	a D2 class object. Squared distances matrix between individuals. One of the alternative ways of entering distance information to a function. See the Details section in dblm . See above dist1 for explanation of its role in this function.
D2.2	a D2 class object. Squared distances between observations. One of the alternative ways of entering distance information to a function. See the Details section in dblm . See above dist2 for explanation of its role in this function. Default D2.2=D2.1.

G1	a Gram class object. Doubly centered inner product matrix associated with the squared distances matrix D2. 1.
G2	a Gram class object. Doubly centered inner product matrix associated with the squared distances matrix D2. 2. Default G2=G1
kind.of.kernel	integer number between 1 and 6 which determines the user's choice of smoothing kernel. (1) Epanechnikov (Default), (2) Biweight, (3) Triweight, (4) Normal, (5) Triangular, (6) Uniform.
metric1	metric function to be used when computing dist1 from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
metric2	metric function to be used when computing dist2 from observed explanatory variables. One of "euclidean" (default), "manhattan", or "gower".
method.h	sets the method to be used in deciding the <i>optimal bandwidth h</i> . There are five different methods, AIC, BIC, OCV, GCV (default) and user.h. OCV and GCV take the optimal bandwidth minimizing a cross-validatory quantity (either ocv or gcv). AIC and BIC take the optimal bandwidth minimizing, respectively, the Akaike or Bayesian Information Criterion (see AIC for more details). When method.h is user.h, the bandwidth is explicitly set by the user through the user.h optional parameter which, in this case, becomes mandatory.
weights	an optional numeric vector of weights to be used in the fitting process. By default all individuals have the same weight.
user.h	global bandwidth user.h, set by the user, controlling the size of the local neighborhood of Z. Smoothing parameter (Default: 1st quartile of all the distances d(i,j) in dist1). Applies only if method.h="user.h".
h.range	a vector of length 2 giving the range for automatic bandwidth choice. (Default: quantiles 0.05 and 0.5 of d(i,j) in dist1).
noh	number of bandwidth h values within h.range for automatic bandwidth choice (if method.h!="user.h").
k.knn	minimum number of observations with positive weight in neighborhood localizing. To avoid runtime errors due to a too small bandwidth originating neighborhoods with only one observation. By default k.nn=3.
rel.gvar	relative geometric variability (a real number between 0 and 1). In each dblm iteration, take the lowest effective rank, with a relative geometric variability higher or equal to rel.gvar. Default value (rel.gvar=0.95) uses the 95% of the total variability.
eff.rank	integer between 1 and the number of observations minus one. Number of Euclidean coordinates used for model fitting in each dblm iteration. If specified its value overrides rel.gvar. When eff.rank=NULL (default), calls to dblm are made with method=rel.gvar.
...	arguments passed to or from other methods to the low level.

Details

There are two semi-metrics involved in local linear distance-based estimation: dist1 and dist2.

Both semi-metrics can coincide. For instance, when $\text{dist1} = ||x_i - x_j||$ and $\text{dist2} = ||(x_i, x_i^2, x_i^3) - (x_j, x_j^2, x_j^3)||$ the estimator for new observations coincides with fitting a local cubic polynomial regression.

The set of bandwidth h values checked in automatic bandwidth choice is defined by `h.range` and `noh`, together with `k.knn`. For each h in it a local linear model is fitted and the optimal h is decided according to the statistic specified in `method.h`.

`kind.of.kernel` designates which kernel function is to be used in determining individual weights from `dist1` values. See [density](#) for more information.

Value

A list of class `ldblm` containing the following components:

<code>residuals</code>	the residuals (response minus fitted values).
<code>fitted.values</code>	the fitted mean values.
<code>h.opt</code>	the optimal bandwidth h used in the fitting proces (if <code>method.h!=user.h</code>).
<code>S</code>	the Smoother hat projector.
<code>weights</code>	the specified weights.
<code>y</code>	the response variable used.
<code>call</code>	the matched call.
<code>dist1</code>	the distance matrix (object of class "D2" or "dist") used to calculate the weights of the observations.
<code>dist2</code>	the distance matrix (object of class "D2" or "dist") used to fit the dblm .

Note

Model fitting is repeated n times (n = number of observations) for each bandwidth (`noh*n` times). For a `noh` too large or a sample with many observations, the time of this function can be very high.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Caballe, A., Delicado P, Esteve, A., Fortiana J (2016). *Global and local distance-based generalized linear models*. TEST 25, 170-195.
- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dblml](#) for distance-based linear models.
[ldbglm](#) for local distance-based generalized linear models.
[summary.ldblml](#) for summary.
[plot.ldblml](#) for plots.
[predict.ldblml](#) for predictions.

Examples

```

# example to use of the ldblml function
n <- 100
p <- 1
k <- 5

Z <- matrix(rnorm(n*p),nrow=n)
b1 <- matrix(runif(p)*k,nrow=p)
b2 <- matrix(runif(p)*k,nrow=p)
b3 <- matrix(runif(p)*k,nrow=p)

s <- 1
e <- rnorm(n)*s

y <- Z%*%b1 + Z^2%*%b2 +Z^3%*%b3 + e

D2 <- as.matrix(dist(Z)^2)
class(D2) <- "D2"

ldblml1 <- ldblml(y~Z,kind.of.kernel=1,method="GCV",noh=3,k.knn=3)
ldblml2 <- ldblml(D2.1=D2,D2.2=D2,y,kind.of.kernel=1,method="user.h",k.knn=3)

```

plot.dblm

Plots for objects of classes dblml or dbglm

Description

Six plots (selected by which) are available: a plot of residual vs fitted values, the Q-Qplot of normality, a Scale-Location plot of $\sqrt{|\text{residuals}|}$ against fitted values. A plot of Cook's distances versus row labels, a plot of residuals against leverages, and the optimal effective rank of "OCV", "GCV", "AIC" or "BIC" method (only if one of these four methods have been chosen in function `dblml`). By default, only the first three and 5 are provided.

Usage

```

## S3 method for class 'dblml'
plot(x,which=c(1:3, 5),id.n=3,main="",

```

```
cook.levels = c(0.5, 1), cex.id = 0.75,
type.pred=c("link", "response"), ...)
```

Arguments

x	an object of class <code>dbl</code> m or <code>dbglm</code> .
which	if a subset of the plots is required, specify a subset of the numbers 1:6.
id.n	number of points to be labelled in each plot, starting with the most extreme.
main	an overall title for the plot. Only if one of the six plots is selected.
cook.levels	levels of Cook's distance at which to draw contours.
cex.id	magnification of point labels.
type.pred	the type of prediction (required only for a <code>dbglm</code> class object). Like <code>predict.dbglm</code> , the default "link" is on the scale of the linear predictors; the alternative "response" is on the scale of the response variable.
...	other parameters to be passed through to plotting functions.

Details

The five first plots are very useful to the residual analysis and are the same that `plot.lm`. A plot of residuals against fitted values sees if the variance is constant. The qq-plot checks if the residuals are normal (see `qqnorm`). The plot between "Scale-Location" and the fitted values takes the square root of the absolute residuals in order to diminish skewness. The Cook's distance against the row labels, measures the effect of deleting a given observation (estimate of the influence of a data point). Points with a large Cook's distance are considered to merit closer examination in the analysis. Finally, the Residual-Leverage plot also shows the most influence points (labelled by Cook's distance). See `cooks.distance`.

The last plot, allows to view the "OCV" (just for `dbl`m), "GCV", "AIC" or "BIC" criterion according to the used rank in the `dbl`m or `dbglm` functions, and chosen the minimum. Applies only if the parameter `full.search` its TRUE.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.

Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

Belsley, D. A., Kuh, E. and Welsch, R. E. (1980). *Regression Diagnostics*. New York: Wiley.

See Also

[dblm](#) for distance-based linear models.

[dbglm](#) for distance-based generalized linear models.

Examples

```
n <- 64
p <- 4
k <- 3

Z <- matrix(rnorm(n*p),nrow=n)
b <- matrix(runif(p)*k,nrow=p)
s <- 1
e <- rnorm(n)*s
y <- Z%*%b + e

dblm1 <- dblm(y~Z,metric="gower",method="GCV", full.search=FALSE)
plot(dblm1)
plot(dblm1,which=4)
```

plot.dbplsr

Plots for a dbplsr object

Description

Four plots (selected by which) are available: plot of scores, response vs scores, R2 contribution in each component and the value of "OCV", "GCV", "AIC" or "BIC" vs the number of component chosen.

Usage

```
## S3 method for class 'dbplsr'
plot(x,which=c(1L:4L),main="",scores.comps=1:2,
      component=1,method=c("OCV","GCV","AIC","BIC"),...)
```

Arguments

x	an object of class dbplsr.
which	if a subset of the plots is required, specify a subset of the numbers 1:4.
main	an overall title for the plot. Only if one of the four plots is selected.

scores.comps	array containing the component scores crossed in the first plot (default the first two).
component	numeric value. Component vs response in the second plot (Default the first component).
method	chosen method "OCV", "GCV", "AIC" or "BIC" in the last plot.
...	other parameters to be passed through to plotting functions.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Implementing PLS for distance-based regression: computational issues*. Computational Statistics 22, 237-248.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.
- Belsley, D. A., Kuh, E. and Welsch, R. E. (1980). *Regression Diagnostics*. New York: Wiley.

See Also

[dbplsr](#) for distance-based partial least squares.

Examples

```
#require(pls)
library(pls)
data(yarn)
## Default methods:
yarn.dbplsr <- dbplsr(density ~ NIR, data = yarn, ncomp=6, method="GCV")
plot(yarn.dbplsr,scores.comps=1:3)
```

plot.ldblm

*Plots for objects of classes ldblm or ldbglm***Description**

Three plots (selected by which) are available: a plot of fitted values vs response, a plot of residuals vs fitted and the optimal bandwidth h of "OCV", "GCV", "AIC" or "BIC" criterion (only if one of these four methods have been chosen in the `ldblm` function). By default, only the first and the second are provided.

Usage

```
## S3 method for class 'ldblm'
plot(x, which=c(1,2), id.n=3, main="", ...)
```

Arguments

<code>x</code>	an object of class <code>ldblm</code> or <code>ldbglm</code> .
<code>which</code>	if a subset of the plots is required, specify a subset of the numbers 1:3.
<code>id.n</code>	number of points to be labelled in each plot, starting with the most extreme.
<code>main</code>	an overall title for the plot. Only if one of the three plots is selected.
<code>...</code>	other parameters to be passed through to plotting functions.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.
- Belsley, D. A., Kuh, E. and Welsch, R. E. (1980). *Regression Diagnostics*. New York: Wiley.

See Also

`ldblm` for local distance-based linear models.
`ldbglm` for local distance-based generalized linear models.

Examples

```
# example to use of the ldblm function
n <- 100
p <- 1
k <- 5

Z <- matrix(rnorm(n*p),nrow=n)
b1 <- matrix(runif(p)*k,nrow=p)
b2 <- matrix(runif(p)*k,nrow=p)
b3 <- matrix(runif(p)*k,nrow=p)

s <- 1
e <- rnorm(n)*s

y <- Z%*%b1 + Z^2%*%b2 +Z^3%*%b3 + e

D2 <- as.matrix(dist(Z))^2
class(D2) <- "D2"

ldblm1 <- ldblm(D2,y=y,kind.of.kernel=1,method.h="AIC",noh=5,h.knn=NULL)
plot(ldblm1)
plot(ldblm1,which=3)
```

predict.dbglm

Predicted values for a dbglm object

Description

`predict.dbglm` returns the predicted values, obtained by tested the generalized distance regression function in the new data (`newdata`).

Usage

```
## S3 method for class 'dbglm'
predict(object,newdata,type.pred=c("link", "response"),
        type.var="Z",...)
```

Arguments

<code>object</code>	an object of class <code>dbglm</code> . Result of dbglm .
<code>newdata</code>	<code>data.frame</code> or matrix which contains the values of Z (if <code>type.var="Z"</code>). The squared distances between k new individuals and the original n individuals (only if <code>type.var="D2"</code>). Finally, the G inner products matrix (if <code>type.var="G"</code>).
<code>type.pred</code>	the type of prediction (required). The default <code>"link"</code> is on the scale of the linear predictors; the alternative <code>"response"</code> is on the scale of the response variable.
<code>type.var</code>	set de type of <code>newdata</code> . Can be <code>"Z"</code> if <code>newdata</code> contains the values of the explanatory variables, <code>"D2"</code> if contains the squared distances matrix or <code>"G"</code> if contains the inner products matrix.
<code>...</code>	arguments passed to or from other methods to the low level.

Details

The predicted values may be the expected mean values of response for the new data (`type.pred="response"`), or the linear predictors evaluated in the estimated `dblm` of the last iteration.

In classical linear models the mean and the linear predictor are the same (makes use of the identity link). However, other distributions such as Poisson or binomial, the link could change. It's easy to get the predicted mean values, as these are calculated by the inverse link of linear predictors. See [family](#) to view how to use `linkfun` and `linkinv`.

Value

`predict.dbglm` produces a vector of predictions for the k new individuals.

Note

Look at which way (or `type.var`) was made the `dbglm` call. The parameter `type.var` must be consistent with the data type that is introduced to `dbglm`.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.

Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbglm](#) for distance-based generalized linear models.

Examples

```
z <- rnorm(100)
y <- rpois(100, exp(1+z))
glm1 <- glm(y ~z, family=quasi("identity"))
dbglm1 <- dbglm(y~z,family=quasi("identity"), method="rel.gvar")

newdata<-0

pr1 <- predict(dbglm1,newdata,type.pred="response",type.var="Z")
print(pr1)
plot(z,y)
points(z,dbglm1$fitt,col=2)
points(0,pr1,col=2)
abline(v=0,col=2)
abline(h=pr1,col=2)
```

predict.dblm

Predicted values for a dblm object

Description

`predict.dblm` returns the predicted values, obtained by evaluating the distance regression function in the new data (`newdata`). `newdata` can be the values of the explanatory variables of these new cases, the squared distances between these new individuals and the originals ones, or rows of new doubly weighted and centered inner products matrix `G`.

Usage

```
## S3 method for class 'dblml'
predict(object,newdata,type.var="Z",...)
```

Arguments

<code>object</code>	an object of class <code>dblml</code> . Result of dbglm .
<code>newdata</code>	data.frame or matrix which contains the values of <code>Z</code> (if <code>type.var="Z"</code>). The squared distances between <code>k</code> new individuals and the original <code>n</code> individuals (only if <code>type.var="D2"</code>). Finally, the <code>G</code> inner products matrix (if <code>type.var="G"</code>).

`type.var` set de `type.var` of `newdata`. Can be "Z" if `newdata` contains the values of the explanatory variables, "D2" if contains the squared distances matrix or "G" if contains the inner products matrix.

... arguments passed to or from other methods to the low level.

Value

`predict.dblm` produces a vector of predictions for the `k` new individuals.

Note

Look at which way (or `type.var`) was made the `dbl`m call. The parameter `type.var` must be consistent with the data type that is introduced to `dbl`m.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbl](#)m for distance-based linear models.

Examples

```
# prediction of new observations newdata
n <- 100
p <- 3
k <- 5

Z <- matrix(rnorm(n*p),nrow=n)
b <- matrix(runif(p)*k,nrow=p)
s <- 1
e <- rnorm(n)*s
y <- Z*%b + e
```

```

D <- dist(Z)
D2 <- disttoD2(D)
D2_train <- D2[1:90,1:90]
class(D2_train)<-"D2"

dblm1 <- dblm(D2_train,y[1:90])

newdata <- D2[91:100,1:90]
predict(dblm1,newdata,type.var="D2")

```

predict.dbplsr	<i>Predicted values for a dbpls object</i>
----------------	--

Description

predict.dbplsr returns the predicted values, obtained by evaluating the Distance-based partial least squares function in the new data (newdata). newdata can be the values of the explanatory variables of these new cases, the squared distances between these new individuals and the originals ones, or rows of new doubly weighted and centered inner products matrix G.

Usage

```

## S3 method for class 'dbplsr'
predict(object,newdata,type.var="Z",...)

```

Arguments

object	an object of class dbplsr. Result of dbplsr .
newdata	data.frame or matrix which contains the values of Z (if type.var="Z". The squared distances between k new individuals and the original n individuals (only if type.var="D2"). Finally, the G inner products matrix (if type.var="G").
type.var	set de type of newdata. Can be "Z" if newdata contains the values of the explanatory variables, "D2" if contains the squared distances matrix or "G" if contains the inner products matrix.
...	arguments passed to or from other methods to the low level.

Value

predict.dbplsr produces a vector of predictions for the k new individuals.

Note

Look at which way (or type.var) was made the dbplsr call. The parameter type.var must be consistent with the data type that is introduced to dbplsr.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Implementing PLS for distance-based regression: computational issues*. Computational Statistics 22, 237-248.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbplsr](#) for distance-based partial least squares.

Examples

```
#require(pls)
# prediction of new observations newdata
library(pls)
data(yarn)
## Default methods:
yarn.dbplsr <- dbplsr(density[1:27] ~ NIR[1:27,], data = yarn, ncomp=6, method="GCV")
pr_yarn_28 <- predict(yarn.dbplsr, newdata=t(as.matrix(yarn$NIR[28,])))
print(pr_yarn_28)
print(yarn$density[28])
```

predict.ldbglm

Predicted values for a ldbglm object

Description

predict.ldbglm returns the predicted values, obtained by evaluating the local distance-based generalized linear model in the new data (newdata2), using newdata1 to estimate the "kernel weights".

Usage

```
## S3 method for class 'ldbglm'
predict(object,newdata1,newdata2=newdata1,
        new.k.knn=3,type.pred=c("link","response"),
        type.var="Z",...)
```

Arguments

object	an object of class <code>ldbglm</code> . Result of <code>ldbglm</code> .
newdata1	data.frame or matrix which contains the values of Z (if <code>type.var="Z"</code>). The squared distances between k new individuals and the original n individuals (only if <code>type.var="D2"</code>). Finally, the G inner products matrix (if <code>type.var="G"</code>). newdata1 is used to compute kernels and local weights.
newdata2	the same logic as newdata1. newdata2 is used to compute the distance-based generalized regressions with (<code>dbglm</code>). If <code>newdata2=NULL</code> , <code>newdata2 <- newdata1</code> .
new.k.knn	setting a minimum bandwidth in order to check that a candidate bandwidth h doesn't contains DB linear models with only one observation. If <code>new.h.knn=NULL</code> , takes the distance that includes the 3 nearest neighbors for each new individual row.
type.pred	the type of prediction (required). The default link is on the scale of the linear predictors; the alternative "response" is on the scale of the response variable.
type.var	set de type of the newdata paramater. Can be "Z" if newdata contains the values of the explanatory variables, "D2" if contains the squared distances matrix or "G" if contains the inner products matrix.
...	arguments passed to or from other methods to the low level.

Value

A list of class `predict.ldbglm` containing the following components:

fit	predicted values for the k new individuals.
newS	matrix (with dimension (k,n)) of weights used to compute the predictions.

Note

Look at which way (or `type.var`) was made the `ldbglm` call. The parameter `type.var` must be consistent with the data type that is introduced to `ldbglm`.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[ldbglm](#) for local distance-based generalized linear models.

Examples

```
# example to use of the predict.ldbglm function
z <- rnorm(100)
y <- rpois(100, exp(1+z))
glm5 <- glm(y ~z, family=quasi("identity"))
ldbglm5 <- ldbglm(dist(z),y=y,family=quasi("identity"),noh=3)
plot(z,y)
points(z,glm5$fitt,col=2)
points(z,ldbglm5$fitt,col=3)

pr_ldbglm5 <- predict(ldbglm5,as.matrix(dist(z)^2),type.pred="response",type.var="D2")
max(pr_ldbglm5$fit-ldbglm5$fitt)
```

predict.ldblm

Predicted values for a ldblm object

Description

predict.ldblm returns the predicted values, obtained by evaluating the local distance-based linear model in the new data (newdata2), using newdata1 to estimate the "kernel weights".

Usage

```
## S3 method for class 'ldblm'
predict(object,newdata1,newdata2=newdata1,
        new.k.knn=3,type.var="Z",...)
```


Arguments

object	an object of class ldblm. Result of <code>ldblm</code> .
newdata1	data.frame or matrix which contains the values of Z (if <code>type.var="Z"</code>). The squared distances between k new individuals and the original n individuals (only if <code>type.var="D2"</code>). Finally, the G inner products matrix (if <code>type.var="G"</code>). newdata1 is used to compute kernels and local weights.
newdata2	the same logic as newdata1. newdata2 is used to compute the Distance-based Regressions with (<code>dblm</code>). If <code>newdata2=NULL</code> , <code>newdata2 <- newdata1</code> .
new.k.knn	setting a minimum bandwidth in order to check that a candidate bandwidth h doesn't contains DB linear models with only one observation. If <code>new.h.knn=NULL</code> , takes the distance that includes the 3 nearest neighbors for each new individual row.
type.var	set de type of the newdata paramater. Can be "Z" if newdata contains the values of the explanatory variables, "D2" if contains the squared distances matrix or "G" if contains the inner products matrix.
...	arguments passed to or from other methods to the low level.

Value

A list of class `predict.ldblm` containing the following components:

fit	predicted values for the k new individuals.
newS	matrix (with dimension (k,n)) of weights used to compute the predictions.

Note

Look at which way (or `type.var`) was made the `ldblm` call. The parameter `type.var` must be consistent with the data type that is introduced to `ldblm`.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[ldblm](#) for local distance-based linear models.

Examples

```
# example to use of the predict.ldblm function

n <- 100
p <- 1
k <- 5

Z <- matrix(rnorm(n*p),nrow=n)
b1 <- matrix(runif(p)*k,nrow=p)
b2 <- matrix(runif(p)*k,nrow=p)
b3 <- matrix(runif(p)*k,nrow=p)

s <- 1
e <- rnorm(n)*s

y <- Z%*%b1 + Z^2%*%b2 + Z^3%*%b3 + e

D <- as.matrix(dist(Z))
D2 <- D^2

newdata1 <- 0

ldblm1 <- ldblml(y~Z,kind.of.kernel=1,method="GCV",noh=3,k.knn=3)
pr1 <- predict(ldblm1,newdata1)
print(pr1)
plot(Z,y)
points(0,pr1$fit,col=2)
abline(v=0,col=2)
abline(h=pr1$fit,col=2)
```

summary.dbglm

Summarizing distance-based generalized linear model fits

Description

summary method for class "dbglm"

Usage

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

Arguments

object	an object of class dbglm. Result of dbglm .
dispersion	the dispersion parameter for the family used. Either a single numerical value or NULL (the default)
...	arguments passed to or from other methods to the low level.

Value

A list of class `summary.dbglm` containing the following components:

call	the matched call.
family	the family object used.
deviance	measure of discrepancy or goodness of fit. Proportional to twice the difference between the maximum log likelihood achievable and that achieved by the model under investigation.
aic	Akaike's An Information Criterion.
df.residual	the residual degrees of freedom.
null.deviance	the deviance for the null model.
df.null	the residual degrees of freedom for the null model.
iter	number of Fisher Scoring (dblm) iterations.
deviance.resid	the deviance residuals for each observation: $\text{sign}(y - \mu) \cdot \sqrt{di}$.
pears.resid	the raw residual scaled by the estimated standard deviation of y .
dispersion	the dispersion is taken as 1 for the binomial and Poisson families, and otherwise estimated by the residual Chisquared statistic (calculated from cases with non-zero weights) divided by the residual degrees of freedom.
gvar	weighted geometric variability of the squared distance matrix.
gvec	diagonal entries in weighted inner products matrix G .
convcrit	convergence criterion. One of: "DevStat" (stopping criterion 1), "muStat" (stopping criterion 2), "maxiter" (maximum allowed number of iterations has been exceeded).

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.

Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.

Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbglm](#) for distance-based generalized linear models.

summary.dblm	<i>Summarizing distance-based linear model fits</i>
--------------	---

Description

summary method for class "dblml"

Usage

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

Arguments

object	an object of class dblml. Result of dbglm .
...	arguments passed to or from other methods to the low level.

Value

A list of class summary.dblml containing the following components:

residuals	the residuals (response minus fitted values).
sigma	the residual standard error.
r.squared	the coefficient of determination R ² .
adj.r.squared	adjusted R-squared.
rdf	the residual degrees of freedom.
call	the matched call.
gvar	weighted geometric variability of the squared distance matrix.
gvec	diagonal entries in weighted inner products matrix G.
method	method used to decide the <i>effective rank</i> .
eff.rank	integer between 1 and the number of observations minus one. Number of Euclidean coordinates used for model fitting. Applies only if method="eff.rank".
rel.gvar	relative geometric variability (real between 0 and 1). Take the lowest effective rank with a relative geometric variability higher or equal to rel.gvar. Default value (rel.gvar=0.95) uses a 95% of the total variability. Applies only rel.gvar if method="rel.gvar".
crit.value	value of criterion defined in method.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dblm](#) for distance-based linear models.

summary.dbplsr

Summarizing distance-based partial least squares fits

Description

summary method for class "dbplsr"

Usage

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

Arguments

object an object of class dbplsr. Result of [dbplsr](#).

... arguments passed to or from other methods to the low level.

Value

A list of class `summary.dbplsr` containing the following components:

<code>ncomp</code>	the number of components of the model.
<code>r.squared</code>	the coefficient of determination R^2 .
<code>adj.r.squared</code>	adjusted R -squared.
<code>call</code>	the matched call.
<code>residuals</code>	a list containing the residuals for each iteration (response minus fitted values).
<code>sigma</code>	the residual standard error.
<code>gvar</code>	total weighted geometric variability.
<code>gvec</code>	the diagonal entries in G_0 .
<code>gvar.iter</code>	geometric variability for each iteration.
<code>method</code>	the using method to set <code>ncomp</code> .
<code>crit.value</code>	value of criterion defined in method.
<code>ncomp.opt</code>	optimum number of components according to the selected method.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Implementing PLS for distance-based regression: computational issues*. Computational Statistics 22, 237-248.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[dbplsr](#) for distance-based partial least squares.

Examples

```
# require(pls)
library(pls)
data(yarn)
## Default methods:
yarn.dbplsr <- dbplsr(density ~ NIR, data = yarn, ncomp=6, method="GCV")
summary(yarn.dbplsr)
```

summary.ldbglm

*Summarizing local distance-based generalized linear model fits***Description**

summary method for class "ldbglm".

Usage

```
## S3 method for class 'ldbglm'
summary(object, dispersion = NULL, ...)
```

Arguments

object	an object of class <code>ldbglm</code> . Result of <code>ldbglm</code> .
dispersion	the dispersion parameter for the family used. Either a single numerical value or NULL (the default)
...	arguments passed to or from other methods to the low level.

Value

A list of class `summary.ldbglm` containing the following components:

nobs	number of observations.
trace.hat	Trace of smoother matrix.
call	the matched call.
family	the <code>family</code> object used.
deviance	measure of discrepancy or goodness of fit. Proportional to twice the difference between the maximum log likelihood achievable and that achieved by the model under investigation.
df.residual	the residual degrees of freedom.
null.deviance	the deviance for the null model.
df.null	the residual degrees of freedom for the null model.
iter	number of Fisher Scoring (<code>dblm</code>) iterations.
deviance.resid	the deviance residuals for each observation: $\text{sign}(y - \mu) \cdot \sqrt{di}$.
pears.resid	the raw residual scaled by the estimated standard deviation of y .

dispersion	the dispersion is taken as 1 for the binomial and Poisson families, and otherwise estimated by the residual Chisquared statistic (calculated from cases with non-zero weights) divided by the residual degrees of freedom.
kind.kernel	smoothing kernel function.
method.h	method used to decide the optimal bandwidth.
h.opt	the optimal bandwidth h used in the fitting proces (if method.h!=user.h).
crit.value	value of criterion defined in method.h.

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

- Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.
- Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.
- Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.
- Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.
- Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

[ldbg1m](#) for local distance-based generalized linear models.

summary.ldblm

Summarizing local distance-based linear model fits

Description

summary method for class "ldblm".

Usage

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


Arguments

object	an object of class <code>ldblm</code> . Result of <code>ldblm</code> .
...	arguments passed to or from other methods to the low level.

Value

A list of class `summary.ldblm` containing the following components:

nobs	number of observations.
r.squared	the coefficient of determination R^2 .
trace.hat	Trace of smoother matrix .
call	the matched call.
residuals	the residuals (the response minus fitted values).
family	the <code>family</code> object used.
kind.kernel	smoothing kernel function.
method.h	method used to decide the optimal bandwidth.
h.opt	the optimal bandwidth h used in the fitting proces (if <code>method.h!=user.h</code>).
crit.value	value of criterion defined in <code>method.h</code> .

Author(s)

Boj, Eva <evaboj@ub.edu>, Caballe, Adria <adria.caballe@upc.edu>, Delicado, Pedro <pedro.delicado@upc.edu> and Fortiana, Josep <fortiana@ub.edu>

References

Boj E, Delicado P, Fortiana J (2010). *Distance-based local linear regression for functional predictors*. Computational Statistics and Data Analysis 54, 429-437.

Boj E, Grane A, Fortiana J, Claramunt MM (2007). *Selection of predictors in distance-based regression*. Communications in Statistics B - Simulation and Computation 36, 87-98.

Cuadras CM, Arenas C, Fortiana J (1996). *Some computational aspects of a distance-based model for prediction*. Communications in Statistics B - Simulation and Computation 25, 593-609.

Cuadras C, Arenas C (1990). *A distance-based regression model for prediction with mixed data*. Communications in Statistics A - Theory and Methods 19, 2261-2279.

Cuadras CM (1989). *Distance analysis in discrimination and classification using both continuous and categorical variables*. In: Y. Dodge (ed.), *Statistical Data Analysis and Inference*. Amsterdam, The Netherlands: North-Holland Publishing Co., pp. 459-473.

See Also

`ldblm` for local distance-based linear models.

Index

AIC, [8](#), [12](#), [16](#), [21](#), [26](#)
as.D2, [4](#)
as.Gram, [4](#)

cooks.distance, [29](#)

D2toDist, [4](#), [5](#), [6](#), [13](#), [18](#), [19](#)
D2toG, [4](#), [5](#), [6](#), [13](#), [18](#), [19](#)
daisy, [2](#), [7](#), [12](#), [13](#), [16](#), [18](#)
dbglm, [3](#), [7](#), [20](#), [21](#), [23](#), [24](#), [29](#), [30](#), [34](#), [35](#), [39](#),
[43](#), [44](#)
dblm, [3](#), [8–10](#), [11](#), [16](#), [21](#), [22](#), [25](#), [27–30](#), [35](#),
[36](#), [41](#), [44](#), [45](#)
dbplsr, [3](#), [15](#), [31](#), [37](#), [38](#), [45](#), [46](#)
dbstats (dbstats-package), [2](#)
dbstats-package, [2](#)
density, [22](#), [27](#)
dist, [2](#), [5](#), [7](#), [12](#), [13](#), [16](#), [18](#)
disttoD2, [4–6](#), [13](#), [18](#), [19](#)

family, [8](#), [9](#), [21](#), [23](#), [34](#), [43](#), [47](#), [49](#)
formula, [7](#), [12](#), [15](#), [21](#), [25](#)

glm, [7](#)
GtoD2, [2](#), [4–6](#), [13](#), [18](#), [19](#)

is.D2 (as.D2), [4](#)
is.Gram (as.Gram), [4](#)

ldbgglm, [3](#), [20](#), [28](#), [32](#), [33](#), [39](#), [40](#), [47](#), [48](#)
ldblm, [3](#), [14](#), [24](#), [24](#), [32](#), [33](#), [41](#), [42](#), [49](#)
lm, [12](#)
loess, [21](#), [25](#)

optimize, [8](#), [12](#)

plot.dbglm, [10](#)
plot.dbglm(plot.dblm), [28](#)
plot.dblm, [10](#), [14](#), [28](#)
plot.dbplsr, [17](#), [30](#)
plot.ldbgglm, [24](#)

plot.ldbgglm(plot.ldblm), [32](#)
plot.ldblm, [23](#), [28](#), [32](#)
plot.lm, [29](#)
plsr, [15](#)
pr_DB, [2](#)
predict.dbglm, [10](#), [29](#), [33](#)
predict.dblm, [14](#), [35](#)
predict.dbplsr, [17](#), [37](#)
predict.ldbgglm, [24](#), [38](#)
predict.ldblm, [28](#), [40](#)
print.dbglm (dbglm), [7](#)
print.dblm (dblm), [11](#)
print.dbplsr (dbplsr), [15](#)
print.ldbgglm (ldbgglm), [20](#)
print.ldblm (ldblm), [24](#)
print.predict.ldbgglm (predict.ldbgglm),
[38](#)
print.predict.ldblm (predict.ldblm), [40](#)
print.summary.dbglm (summary.dbglm), [42](#)
print.summary.dblm (summary.dblm), [44](#)
print.summary.dbplsr (summary.dbplsr),
[45](#)
print.summary.ldbgglm (summary.ldbgglm),
[47](#)
print.summary.ldblm (summary.ldblm), [48](#)

qqnorm, [29](#)

summary.dbglm, [10](#), [42](#)
summary.dblm, [14](#), [44](#)
summary.dbplsr, [17](#), [45](#)
summary.ldbgglm, [24](#), [47](#)
summary.ldblm, [23](#), [28](#), [48](#)