Package ‘DBKGrad’

February 19, 2015

Title  Discrete Beta Kernel Graduation of Mortality Data
Version 1.6
Date 2014-10-24
Description This package allows for nonparametric graduation of mortality rates using fixed or adaptive discrete beta kernel estimator.
License GPL-2
LazyLoad yes
Depends R (>= 2.15.0)
Imports minpack.lm, SDD, TSA, lattice
Author Angelo Mazza [aut, cre], Antonio Punzo [aut]
Maintainer Angelo Mazza <a.mazza@unict.it>
NeedsCompilation no
Repository CRAN
Date/Publication 2014-10-25 08:08:31

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBKGrad-package</td>
<td>2</td>
</tr>
<tr>
<td>dbkGrad</td>
<td>3</td>
</tr>
<tr>
<td>ItalyM</td>
<td>7</td>
</tr>
<tr>
<td>plot.dbkGrad</td>
<td>7</td>
</tr>
</tbody>
</table>

Index 11
Description

This package allows for nonparametric graduation of mortality rates using the discrete beta kernel estimator in both its fixed (Mazza and Punzo, 2011) and adaptive (Mazza and Punzo, 2013a, 2013b) variants.

Details

Package: DBKGrad
Type: Package
Version: 1.6
Date: 2014-10-24
License: GNU-2

Author(s)

Angelo Mazza and Antonio Punzo
Maintainer: Angelo Mazza <a.mazza@unict.it>

References


**See Also**

dbkGrad, plot, ItalyM

---

**dbkGrad**

*Discrete Beta Kernel Graduation of Mortality Rates*

**Description**

This function performs nonparametric graduation of mortality rates using discrete beta kernel smoothing techniques.

**Usage**

```r
dbkGrad(obsq, limx, limy, exposures = NULL, transformation = c("none", "log", "logit", "Gompertz"), bwtypex = c("FX", "VC", "EX"), bwtypey = c("FX", "VC", "EX"), adaptx = c("a", "b", "ab"), adapty = c("a", "b", "ab"), hx = 0.002, hy = 0.002, sx = 0.2, sy = 0.2, cvres = c("propres", "res"), cvhx = FALSE, cvhy = FALSE, cvsx = FALSE, cvsy = FALSE, alpha = 0.05)
```

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

```r
## S3 method for class 'dbkGrad'
as.data.frame(x, row.names = x$limx[1]:x$limx[2], optional = FALSE, ...)
```

```r
## S3 method for class 'dbkGrad'
residuals(object, type = c("working", "proportional", "response", "deviance", "pearson"), ...)
```

**Arguments**

- `obsq` a matrix (or an object which can be coerced to a matrix using `as.matrix()`) of observed mortality rates. Dimnames, if provided, should be numeric; row names should be ages and column names years.
limx, limy optional vector of two integers; if provided, limx (limy) sets a lower and an upper row (column) limit. Only data within these intervals are graduated.

exposures an optional matrix containing the exposed to the risk of death for each age and year. Dimensions of exposures should correspond to those of obsq.

transformation an optional character string; the transformation specified is applied to the observed data before graduation. Graduated data are then back-transformed. Possible values are "none" for no transformation, "log", "logit" and "Gompertz".

bwtype, bwtypey an optional character string. It specifies the type of bandwidth to be adopted by row (by column) and must be:
- "FX" for a fixed bandwidth (default),
- "EX" for an adaptive bandwidth based on exposures (see Mazza A, Punzo A, 2013a, for details);
- "VC" for an adaptive bandwidth based on a vector of weights derived from the variation coefficients which, in turn, depends from the exposures (see Mazza A, Punzo A, in press, for details).

adaptx, adapty an optional character string. It is the type of adaptive bandwidth to be adopted by row (by column) and must be:
- "a" a different bandwidth for each evaluation age x at which the rates are estimated;
- "b" a different bandwidth can be attributed to each age, regardless from the evaluation point;
- "ab" a different bandwidth can be selected for each evaluation point and for each age.

hx, hy an optional scalar. It is the global bandwidth used for the variable on the rows (columns). Default value is 0.002. If cvhx=TRUE (cvhy=TRUE), then the smoothing parameter is computed by means of cross-validation using this value as an initialization.

sx, sy an optional scalar. It is the sensitive parameter used for the variable on the rows (columns). Default value is 0.2. If cvsx=TRUE (cvsy=TRUE), then the sensitive parameter is computed by means of cross-validation using this value as an initialization.

cvhx, cvhy an optional logical; if cvsx=TRUE (cvsy=TRUE) then cross-validation is used to select the smoothing parameter. Default value is TRUE. Parameter hx, (hy) is the initial value used in cross-validation.

cvsx, cvsy an optional logical; if TRUE then cross-validation is used to select the sensitive parameter. Default value is FALSE. The value of sx (sy) is used to initialize the cross-validation process.

cvres an optional character string; if cvres="propres" (the default), then cross-validation selects the smoothing parameter and/or the sensitive parameter by minimizing the squares of the proportional differences between observed and estimated values, while if cvres="res" then the sum of square residuals is minimized.

alpha an optional scalar. When the exposures argument is provided, the function returns \((1-alpha)\times 100\%\) pointwise confidence intervals and pointwise confidence bands for fitted values. Default value is 0.05.
dbkGrad

x a dbkGrad object

row.names a NULL or a character vector giving the row names for the data frame. Missing values are not allowed. Default value is 0:x$limx, limy.

optional logical. If TRUE, setting row names and converting column names (to syntactic names: see make.names) is optional.

... additional arguments to be passed to or from methods.

type "working","proportional","response","deviance","pearson"

object a dbkGrad class object.

Details

In the cross-validation routine, minimization is performed using the Levenberg-Marquardt algorithm (More 1978) in the minpack.lm package (Elzhov, Mullen, and Bolker 2010).

Value

Returned from this function is an dbkGrad object which is a list with the following components:

- fitted.values a matrix containing the graduated values.
- residuals a matrix containing the working residuals fitted.values - obsq.
- kernels a matrix. kernels %*% obsq returns the fitted.values
- cvRSS a scalar. It is the cross-validation residual sum of squares (RSS) computed over the fitted values, using the residuals specified in cvres.
- hx (hy) a scalar. It is the global bandwidth used for the variable on the rows (columns).
- sx (sy) a scalar. It is the sensitive parameter used for the variable on the rows (columns).
- It is returned when bandwidth = "EX" or bandwidth = "VC"
- upperbound, lowerbound pointwise confidence interval. Returned when exposures is provided.
- bonferroniupperbound, bonferronilowerbound limits of the Bonferroni confidence bands. Returned when exposures is provided.
- sidakupperbound,idaklowerbound limits of the Sidak confidence bands. Returned when exposures is provided.
- obsq a matrix containing the observed mortality rates with dimensions set by limx, limy.
- exposures a matrix containing the exposures with dimensions set by limx, limy.
- limx (limy) a vector with lower and upper row (column) limits. Only data within these interval are graduated.
- call an object of class call.

Author(s)

Angelo Mazza and Antonio Punzo
References


See Also

`DBKGrad-package`, `plot`, `ItalyM`

Examples

data("ItalyM")

# unidimensional analysis

res1 <- dbkGrad(obsq=obsq, limx=c(6,71), limy=104, exposure=population, bwtype="EX", adaptx="ab")
plot(res1, plottype="obsfit", CI=FALSE, CBBonf=TRUE)
plot(res1, plottype="residuals", resType="pearson")
residuals(res1, type="pearson")

# bidimensional analysis

res2 <- dbkGrad(obsq=obsq, limx=c(6,46), limy=c(60,80), exposure=population, transformation="logit", bwtype="VC", bwtype="EX", hx=0.01, hy=0.008, adaptx="ab", adaptY="b")
plot(res2, plottype="obsfit")
plot(res2, plottype="obsfit", plotstyle="persp", col="black")
**ItalyM**  
*Mortality data for the 1906-2009 male population of Italy*

**Description**

This data set consists of probabilities of death and population size for the male Italian population aged from 0 to 95, years from 1906 to 2009.

**Usage**

data(ItalyM)

**Format**

- obsq is a $[1:96, 1:104]$ numeric matrix containing probabilities of death.
- population is a $[1:96, 1:104]$ numeric matrix containing the male Italian population.
- In both, row names are the ages $(0,1,\ldots,95)$ and column names are the years $(1905,\ldots,2009)$

**Source**

Human Mortality Database [http://www.mortality.org](http://www.mortality.org)

**References**

Human Mortality Database (2013). University of California, Berkeley (USA), and Max Planck Institute for Demographic Research (Germany). Available at [http://www.mortality.org](http://www.mortality.org)

**See Also**

- DBKGrad-package, dbkGrad

---

**plot.dbkGrad**  
*Plot Method for dbkGrad objects*

**Description**

Plotting the `dbkGrad` object produces a few different plots that are of interest. The different plots, created from various `plottype` options, are described below:
Usage

```r
## S3 method for class 'dbkGrad'
plot(x, plottype = c("obsfit", "fitted", "observed", "exposure", "residuals", "checksd"), plotstyle = c("mat", "level", "persp"), restype = c("working", "proportional", "response", "deviance", "pearson"), byage = TRUE, columns, rows, CI = TRUE, CBconf = FALSE, CBSidak = FALSE, logscale = TRUE, alphasres = 0.05, col, ...)
```

Arguments

- **x**: a `dbkGrad` object
- **plottype**: an optional character string. It specifies the type of plot to display and must be one of:
  - "observed" to plot observed values;
  - "fitted" to plot fitted values;
  - "obsfit" to plot observed and fitted values (default);
  - "exposure" to plot the exposed to the risk of death;
  - "residuals" to display plots related to residuals: density of residuals, residuals versus fitted values, and residuals versus the discrete variable of interest;
  - "checksd" to plot autocorrelogram and autodependogram (see Bagnato, Punzo, Nicolis, 2012) of residuals, only for the unidimensional case.
- **plotstyle**: an optional character string. It specifies the style of plot; it has no effect when `plottype=checksd`. It must be:
  - "mat" for a matplot (default for the unidimensional case);
  - "level" for a levelplot (default for the bidimensional case);
  - "persp" for a perspective plot.
- **restype**: an optional character string. When `plottype=residuals` or `plottype=checksd`, it specifies the type of residuals displayed. It must be:
  - "working" to use working residuals: \( o-f \) (default);
  - "proportional" to use working residuals: \( o/f-1 \);
  - "response" to use response residuals: \( e*(o-f) \);
  - "deviance" to use deviance residuals: \( \text{sign}(o-f) * \sqrt{2*e*o*log(o/f) + 2*e*(1-o)*log((1-o)/(1-f))} \);
  - "pearson" to use Pearson residuals: \( e*(o-f)/\sqrt{e*f*(1-f)} \);
where \( e=\text{exposures}, o=\text{observed values} \) and \( f=\text{fitted values} \).
- **rows, columns**: an optional numeric vector. It specifies the rows (columns) to plot.
- **byage**: an optional logical. It selects the discrete variable of interest: TRUE (default) for the variable in rows (typically age) and FALSE for the other variable (years or duration).
- **logscale**: an optional logical; when TRUE (default), rates are plotted on log scale.
- **CI**: an optional logical; if TRUE, \( 1-\alpha \times 100\% \) confidence intervals for the fitted values are displayed in `plottype="obsfit"` and `plottype="fitted"`. When the `alpha` argument is not provided in `dbkGrad()`, 95% pointwise confidence intervals are displayed. Default value is TRUE.
cbbonf an optional logical; does the same as CI but Bonferroni correction is applied to obtain confidence bands. Default is FALSE.

CBSidak an optional logical; does the same as CI but Sidak correction is applied to obtain confidence bands. Default is FALSE.

alphares an optional scalar. When plottype=residuals the boundaries of the (alphares)*100% critical region are displayed. Default value is 0.05.

col a scalar or a vector with plotting colors.

... additional arguments to be passed to or from methods.

Value

No values are returned from the plot function.

Author(s)

Angelo Mazza and Antonio Punzo

References


See Also

DBKGrad-package, dbkGrad, ItalyM, TSA:acf, SDD:ADF

Examples

data("ItalyM")

# unidimensional analysis

res1 <- dbkGrad(obsq=obsq, limx=c(6,71), limy=104, exposure=population, bwtype="EX", adaptx="ab")
plot(res1, plottype="obsfit", CI=FALSE, CBonf=TRUE)
plot(res1, plottype="residuals", restype="pearson")
plot(res1, plottype="checksd", restype="pearson")
residuals(res1, type="pearson")

# bidimensional analysis
res2 <- dbkGrad(obsq=obsq, limx=c(6,46), limy=c(60,80), exposure=population, transformation="logit", bwtype="VC", bwtypey="EX", hx=0.01, hy=0.008, adaptx="ab", adapty="b")
plot(res2, plottype="obsfit")
plot(res2, plottype="obsfit", plotstyle="persp", col="black")
Index

*Topic plotNdbkgrad
   plot.dbkGrad, 7
*Topic plotNdbkgrad
   plot.dbkGrad, 7
*Topic datasets
   ItalyM, 7

as.data.frame.dbkGrad(dbkGrad), 3

dbkGrad, 3, 3, 7–9
DBKGrad=package, 2

ItalyM, 3, 6, 7, 9

obsq(ItalyM), 7

plot, 3, 6
plot.dbkGrad, 7
population(ItalyM), 7
print.dbkGrad(dbkGrad), 3

residuals.dbkGrad(dbkGrad), 3

SDD: ADF, 9

TSA: acf, 9