Package 'npbr'

July 2, 2014

Type Package

Title Nonparametric boundary regression

Version 1.0

Date 2013-11-26

Author Abdelaati Daouia <Abdelaati.Daouia@tse-fr.eu>, Thibault Laurent <thibault.laurent@univ-tlse1.fr>, Hohsuk Noh <word5810@gmail.com>

Maintainer Thibault Laurent <thibault.laurent@univ-tlse1.fr>

Depends R (>= 2.9.1), splines, graphics, Rglpk, Benchmarking

Description This package provides functions for the best known approaches to nonparametric boundary estimation. The selected methods are concerned with empirical, smooth, unconstrained as well as restricted estimates under both separate and multiple shape constraints. The package also allows for Monte Carlo comparisons among these different estimation procedures, illustrating the simulation experiments by Daouia, Noh and Park (2013), Data envelope fitting with constrained polynomial splines, TSE Working Paper, http://www.tse-fr.eu/images/doc/wp/etrie/wp_tse_449.pdf

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2013-12-12 01:35:35

R topics documented:

---

npbr-package *Nonparametric boundary regression*

Description

This package provides a variety of nonparemtric regression methods and allows for comparison among these methods via Monte Carlo experiments. The package provides also illustrations with real datasets.

Details

Suppose that we have n pairs of observations $(x_i, y_i), \tilde{} i = 1, \ldots, n$, from a bivariate distribution with a density $f(x, y)$ in R^2. The support Ψ of f is assumed to be of the form

$$\Psi = \{(x, y)|y \leq \varphi(x)\} \supseteq \{(x, y)|f(x, y) > 0\}$$

$$\{(x, y)|y > \varphi(x)\} \subseteq \{(x, y)|f(x, y) = 0\},$$

where φ is a monotone increasing and/or concave function whose graph corresponds to the locus of the curve above which the density f is zero. We consider the estimation of the frontier function φ based on the sample $\{(x_i, y_i), \tilde{} i = 1, \ldots, n\}$ in the general setting where the density f may have sudden jumps at the frontier, decay to zero or rise up to infinity as it approaches its support boundary.

The package provides functions for the best known nonparametric estimation procedures. The selected methods can be divided into a number of different categories: empirical, smooth, unconstrained and restricted estimates. The package provides some real data sets as well.

Author(s)

Abdelaati Daouia <Abdelaati.Daouia@tse-fr.eu>, Thibault Laurent <thibault.laurent@univ-tlse1.fr>, Hohsuk Noh <word5810@gmail.com>

Maintainer: Thibault Laurent <thibault.laurent@univ-tlse1.fr>

References

Daouia, A., Noh, H. and Park, B.U. (2013). Data Envelope fitting with constrained polynomial splines. *TSE Working Paper*, http://www.tse-fr.eu/images/doc/wp/etrie/wp_tse_449.pdf.

Examples

```
data("green")
plot(log(OUTPUT)~log(COST), data=green)
x <- seq(min(log(green$COST)), max(log(green$COST)), length.out=1001)
# under the separate monotonicity constraint with the knots mesh being
# obtained via AIC and BIC criteria
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=9, cv=0),
lty=1, col="red")
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=6, cv=0),
lty=1, col="blue")
# under both monotonicity and concavity constraints
# with the same optimal number of knots obtained using AIC and BIC criteria
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=1, cv=1),
lty=2)
legend("topleft",legend=c("monotonicity (kn=9)", "monotonicity (kn=6)",
"monotonicity + concavity (kn=1)"), col=c("red","blue","black"), lty=c(1,1,2))
```

---

dea_est *DEA, FDH and linearized FDH estimators.*

Description

The function implements the empirical FDH (free disposal hull), LFDH (linearized FDH) and DEA (data envelopment analysis) frontier estimators.

Usage

```
dea_est(xtab, ytab, x, type = "dea")
```

Arguments

xtab	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
ytab	numeric vectors of the same length as xtab containing the observed outputs $y_1, \ldots, y_n$
x	a numeric vector of evaluation points in which the estimator is to be computed
type	a character equal to "dea", "fdh" or "lfdh"

Details

There are mainly two usual frontier estimation methods for preserving monotonicity: the free disposal hull (FDH) introduced by Deprins *et al.* (1984) and the data envelopment analysis (DEA) initiated by Farrell (1957). The FDH boundary is the lowest "stair-case" monotone curve covering all the data points

$$\varphi_n(x) := \max\{y_i,\, i : x_i \leq x\}.$$

An improved version of this estimator, referred to as the linearized FDH (LFDH), is obtained by drawing the polygonal line smoothing the staircase FDH curve. It has been considered in Hall and Park (2002) and Jeong and Simar (2006). When the joint support of data is in addition convex, the

DEA estimator is defined as the least concave majorant of the FDH frontier (see also Gijbels *et al.* (1999)).

Employs the function `DEA` from the package **Benchmarking**.

Value

Returns a numeric vector with the same length as `x`

Author(s)

Hohsuk Noh

References

Bogetoft, P. and Otto, L. (2011), *Benchmarking with DEA, SFA and R*, Springer-Verlag

Deprins, D., Simar, L. and H. Tulkens (1984). Measuring labor efficiency in post offices, in: M. Marchand, P. Pestieau and H. Tulkens, (Eds), *The performance of Public Enterprises: Concepts and Measurements*. North-Holland, Amsterdam, pp. 243–267.

Farrell, M.J. (1957). The measurement of productive efficiency. *Journal of the Royal Statistical Society, Series A*, 120, 253–281.

Gijbels, I., Mammen, E., Park, B.U. and Simar, L. (1999). On estimation of monotone and concave frontier functions, *Journal of American Statistical Association*, 94, 220–228.

Hall, P. and Park, B.U. (2002). New methods for bias correction at endpoints and boundaries, *Annals of Statistics*, 30, 1460-1479.

Jeong, S.-O. and Simar, L. (2006). Linearly interpolated FDH efficiency score for nonconvex frontiers, *Journal of Multivariate Analysis*, 97, 2141–2161.

See Also

`quad_spline_est`

Examples

```
data("green")
plot(OUTPUT~COST, data=green)
x <- seq(min(green$COST), max(green$COST), length.out=1001)
# dea
lines(x, dea_est(green$COST, green$OUTPUT, x, type="dea"),
lty=1, col="red")
# fdh
lines(x, dea_est(green$COST, green$OUTPUT, x, type="fdh"),
lty=2, col="blue")
# lfdh
lines(x, dea_est(green$COST, green$OUTPUT, x, type="lfdh"),
lty=3, col="green")
legend("topleft",legend=c("dea","fdh","lfdh"), col=c("red","blue","green"),
lty=1:3)
```

green *American electric utility companies*

Description

The dataset consists of 123 American electric utility companies. As in the set-up of Gijbels *et al.* (1999), we used the measurements of the variables $y_i = \log(q_i)$ and $x_i = \log(c_i)$, where q_i is the production output of the company i and c_i is the total cost involved in the production. For a detailed description and analysis of these data see *e.g.* Christensen and Greene (1976) and Greene (1990).

Usage

```
data(green)
```

Format

A data frame with 123 observations on the following 2 variables.

`COST` a numeric vector

`OUTPUT` a numeric vector

Source

Gijbels *et al.* (1999).

References

Christensen, L.R. and Greene, W.H. (1976). Economies of Scale in U.S. Electric Power Generation, *Journal of Political Economy*, University of Chicago Press, 84(4), 655-76.

Gijbels, I., Mammen, E., Park, B.U. and Simar, L. (1999). On estimation of monotone and concave frontier functions. *JASA*, 94, 220-228.

Greene, W.H. (1990). A Gamma-distributed stochastic frontier model, *Journal of Econometrics*, 46(1-2), 141-163.

Examples

```
data("green")
```

`loc_est` *Local linear frontier estimator*

Description

Computes the local linear smoothing frontier estimator of Hall, Park and Stern (1998).

Usage

```
loc_est(xtab, ytab, x, h)
```

Arguments

`xtab`	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
`ytab`	numeric vectors of the same length as `xtab` containing the observed outputs $y_1, \ldots, y_n$
`x`	a numeric vector of evaluation points in which the estimator is to be computed
`h`	determines the bandwidth at which the local linear estimate will be computed

Details

The implemented local linear estimator of $\varphi(x)$ is defined by

$$\hat{\varphi}_{n,LL}(x) = \min\Big\{z : \text{there}\tilde{}\text{exists}\tilde{}\theta \geq 0\tilde{}\text{such}\tilde{}\text{that}\tilde{}\, y_i \leq z + \theta(x_i - x)$$

$$\text{for}\tilde{}\text{all}\tilde{}\, i\tilde{}\text{such}\tilde{}\text{that}\tilde{}\, x_i \in (x-h, x+h)\Big\}.$$

Hall and Park (2004) proposed a bootstrap procedure for selecting the optimal bandwidth h in $\hat{\varphi}_{n,LL}$. See the function `loc_est_bw`.

Value

Returns a numeric vector with the same length as `x`

Author(s)

Hohsuk Noh

References

Hall, P. and Park, B.U. (2004). Bandwidth choice for local polynomial estimation of smooth boundaries. *Journal of Multivariate Analysis*, 91 (2), 240-261.

Hall, P., Park, B.U. and Stern, S.E. (1998). On polynomial estimators of frontiers and boundaries. *Journal of Multivariate Analysis*, 66, 71-98.

See Also

poly_est, loc_est_bw

Examples

```
data("nuclear")
plot(ytab~xtab, data=nuclear)
x <- seq(min(nuclear$xtab), max(nuclear$xtab), length.out=1001)
lines(x, loc_est(nuclear$xtab, nuclear$ytab, x, h=79.12), lty=1, col="red")
lines(x, loc_est(nuclear$xtab, nuclear$ytab, x, h=40), lty=2, col="blue")
legend("topleft",legend=c("h=72.70", "h=40"), col=c("red","blue"),
lty=c(1,2))
```

---

loc_est_bw *Bootstrap bandwidth selection for the local linear frontier estimator*

Description

Computes the optimal bootstrap bandwidth proposed by Hall and Park (2004) for the local linear frontier estimator

Usage

```
loc_est_bw(xtab, ytab, x, hini, B = 5)
```

Arguments

xtab	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
ytab	numeric vectors of the same length as xtab containing the observed outputs $y_1, \ldots, y_n$
x	a numeric vector of evaluation points in which the estimator is to be computed
hini	the initial bandwidth at which the local linear estimate will be computed
B	number of bootstrap replications

Details

For a detailed description of the bootstrap procedure, see Hall and Park (2004)

Value

Returns the optimal bootstrap bandwidth

Author(s)

Hohsuk Noh

References

Hall, P. and Park, B.U. (2004). Bandwidth choice for local polynomial estimation of smooth boundaries. *Journal of Multivariate Analysis*, 91 (2), 240-261.

See Also

loc_est

Examples

```
data("nuclear")
x <- seq(min(nuclear$xtab), max(nuclear$xtab), length.out=1001)
# loc_est_bw(nuclear$xtab, nuclear$ytab, x, hini=40, B=100)
# long computational time
# returns the value 79.12
```

---

nuclear *Reliability programs of nuclear reactors*

Description

The dataset from the US Electric Power Research Institute (EPRI) consists of 254 toughness results obtained from non-irradiated representative steels. For each steel i, fracture toughness x_i and temperature y_i were measured.

Usage

```
data(nuclear)
```

Format

A data frame with 254 observations on the following 2 variables.

`xtab` Temperature

`ytab` fracture toughness of each material

Source

US Electric Power Research Institute (EPRI)

References

Daouia, A., Girard, S. and Guillou, A. (2013). A Gamma-moment approach to monotonic boundary estimation. *Journal of Econometrics*. http://dx.doi.org/10.1016/j.jeconom.2013.10.013

Examples

```
data("nuclear")
```

poly_est *Polynomial-based edge estimators*

Description

Computes polynomial-type estimators of frontiers and boundaries (Hall et al., 1998)

Usage

```
poly_est(xtab, ytab, x, deg)
```

Arguments

xtab	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
ytab	numeric vectors of the same length as xtab containing the observed outputs $y_1, \ldots, y_n$
x	a numeric vector of evaluation points in which the estimator is to be computed
deg	an integer (polynomial degree)

Details

The data edge is modeled by a single polynomial $\varphi_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_p x^p$ of known degree p that envelopes the full data and minimizes the area under its graph for $x \in [a, b]$, with a and b being respectively the lower and upper endpoints of the design points $x_1, \ldots, x_n$. The function is the estimate $\hat{\varphi}_{n,P}(x) = \hat{\theta}_0 + \hat{\theta}_1 x + \cdots + \hat{\theta}_p x^p$ of $\varphi(x)$, where $\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1, \cdots, \hat{\theta}_p)^T$ minimizes $\int_a^b \varphi_\theta(x)\, dx$ over $\theta \in \mathsf{R}^{p+1}$ subject to the envelopment constraints $\varphi_\theta(x_i) \geq y_i$, $i = 1, \ldots, n$.

Value

Returns a vector of numeric with the same length than x

Author(s)

Hohsuk Noh

References

Hall, P., Park, B.U. and Stern, S.E. (1998). On polynomial estimators of frontiers and boundaries. *Journal of Multivariate Analysis*, 66, 71-98.

See Also

loc_est

Examples

```
data("nuclear")
plot(ytab~xtab, data=nuclear)
x <- seq(min(nuclear$xtab), max(nuclear$xtab), length.out=1001)
lines(x, poly_est(nuclear$xtab, nuclear$ytab, x, deg=2), lty=1, col="red")
lines(x, poly_est(nuclear$xtab, nuclear$ytab, x, deg=4), lty=2, col="blue")
legend("topleft",legend=c("degree=2", "degree=4"), col=c("red","blue"),
lty=c(1,2))
```

`quad_spline_est` *Constrained quadratic spline frontiers*

Description

This function is an implementation of the constrained quadratic spline smoother proposed by Daouia, Noh and Park (2013).

Usage

```
quad_spline_est(xtab, ytab, x, kn = ceiling((length(xtab))^(1/4)), cv = 0,
all.dea = FALSE)
```

Arguments

`xtab`	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
`ytab`	numeric vectors of the same length as `xtab` containing the observed outputs $y_1, \ldots, y_n$
`x`	a numeric vector of evaluation points in which the estimator is to be computed
`kn`	an integer specifying the number of knots at which the spline estimate will be computed
`cv`	an integer equal to 0 (under the monotonicity constraint) or 1 (under simultaneous monotonicity and concavity constraints)
`all.dea`	a boolean

Details

Let a and b be, respectively, the minimum and maximum of the design points $x_1, \ldots, x_n$. Denote a partition of $[a, b]$ by $a = t_0 < t_1 < \cdots < t_{k_n} = b$ (see below the selection process). Let $N = k_n + 1$ and $\pi(x) = (\pi_1(x), \ldots, \pi_N(x))^T$ be the vector of normalized B-splines of order 3 based on the knot mesh $\{t_j\}$ (see, *e.g.*, Schumaker (2007)). When the true frontier $\varphi(x)$ is known or required to be monotone nondecreasing (option `cv=0`), its constrained quadratic spline estimate is defined by $\hat{\varphi}_n(x) = \pi(x)^T \hat{\alpha}$, where $\hat{\alpha}$ minimizes

$$\int_0^1 \pi(x)^T \alpha \, dx = \sum_{j=1}^{N} \alpha_j \int_0^1 \pi_j(x) \, dx$$

over $\alpha \in \mathsf{R}^N$ subject to the envelopment and monotonicity constraints $\pi(x_i)^T\alpha \geq y_i$, $i = 1, \ldots, n$, and $\pi'(t_j)^T\alpha \geq 0$, $j = 0, 1, \ldots, k_n$, with π' being the derivative of π.

Considering the special connection of the spline smoother $\hat{\varphi}_n$ with the traditional FDH frontier φ_n (see the function dea_est), Daouia *et al.* (2013) propose an easy way of choosing the knot mesh. Let $(\mathcal{X}_1, \mathcal{Y}_1), \ldots, (\mathcal{X}_\mathcal{N}, \mathcal{Y}_\mathcal{N})$ be the observations (x_i, y_i) lying on the FDH boundary (*i.e.* $y_i = \varphi_n(x_i)$). The basic idea is to pick out a set of knots equally spaced in percentile ranks among the $\mathcal{N}$ FDH points $(\mathcal{X}_\ell, \mathcal{Y}_\ell)$ by taking $t_j = \mathcal{X}_{[j\mathcal{N}/k_n]}$, the j/k_nth quantile of the values of $\mathcal{X}_\ell$ for $j = 1, \ldots, k_n - 1$. The choice of the number of internal knots is then viewed as model selection through the minimization of the AIC and BIC information criteria (see the function quad_spline_est_kn).

When the monotone boundary $\varphi(x)$ is also believed to be concave (option cv=1), its constrained fit is defined as $\hat{\varphi}_n^\star(x) = \pi(x)^T\hat{\alpha}^\star$, where $\hat{\alpha}^\star \in \mathsf{R}^N$ minimizes the same objective function as $\hat{\alpha}$ subject to the same envelopment and monotonicity constraints and the additional concavity constraints $\pi''(t_j^*)^T\alpha \leq 0$, $j = 1, \ldots, k_n$, where π'' is the constant second derivative of π on each inter-knot interval and t_j^* is the midpoint of $(t_{j-1}, t_j]$.

Regarding the choice of knots, the same scheme as for $\hat{\varphi}_n$ can be applied by replacing the FDH points $(\mathcal{X}_1, \mathcal{Y}_1), \ldots, (\mathcal{X}_\mathcal{N}, \mathcal{Y}_\mathcal{N})$ with the DEA points $(\mathcal{X}_1^*, \mathcal{Y}_1^*), \ldots, (\mathcal{X}_\mathcal{M}^*, \mathcal{Y}_\mathcal{M}^*)$, that is, the observations (x_i, y_i) lying on the piecewise linear DEA frontier (see the function dea_est). Alternatively, the strategy of just using all the DEA points as knots is also working quite well for datasets of modest size as shown in Daouia *et al.* (2013). In this case, the user has to choose the option all.dea=TRUE.

Value

Returns a vector of numeric with the same length than x

Author(s)

Hohsuk Noh

References

Daouia, A., Noh, H. and Park, B.U. (2013). Data Envelope Fitting with Constrained Polynomial splines. *TSE Working Paper*, http://www.tse-fr.eu/images/doc/wp/etrie/wp_tse_449.pdf.

Schumaker, L.L. (2007). *Spline Functions: Basic Theory*, 3rd edition, Cambridge University Press.

See Also

quad_spline_est_kn

Examples

```
data("green")
plot(log(OUTPUT)~log(COST), data=green)
x <- seq(min(log(green$COST)), max(log(green$COST)), length.out=1001)
# under the separate monotonicity constraint with the knots mesh being
# obtained via AIC and BIC criteria
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=9, cv=0),
lty=1, col="red")
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=6, cv=0),
```

```
lty=1, col="blue")
# under both monotonicity and concavity constraints
# with the same optimal number of knots obtained using AIC and BIC criteria
lines(x, quad_spline_est(log(green$COST), log(green$OUTPUT), x, kn=1, cv=1),
lty=2)
legend("topleft",legend=c("monotonicity (kn=9)", "monotonicity (kn=6)",
"monotonicity + concavity (kn=1)"), col=c("red","blue","black"), lty=c(1,1,2))
```

`quad_spline_est_kn` *AIC (or BIC) criterion for choosing the number of knots for quadratic splines*

Description

Computes the optimal number of knots for the constrained quadratic spline fit proposed by Daouia, Noh and Park (2013)

Usage

```
quad_spline_est_kn(xtab, ytab, x, cv, krange = 1:20, type = "AIC")
```

Arguments

`xtab`	numeric vectors containing the observed inputs $x_1, \ldots, x_n$
`ytab`	numeric vectors of the same length as `xtab` containing the observed outputs $y_1, \ldots, y_n$
`x`	a numeric vector of evaluation points in which the estimator is to be computed
`cv`	an integer equal to 0 (constraint of monotonicity only) or 1 (both constrainst of monotonicity and concavity)
`krange`	a vector of integer specifying the number of knots at which the spline estimate will be computed
`type`	a character equal to "AIC" or "BIC"

Details

For the implementation of the monotone quadratic spline smoother $\hat{\varphi}_n$, Daouia *et al.* (2013) first suggest using the set of knots $\{t_j = \mathcal{X}_{[j\mathcal{N}/k_n]}, \tilde{} j = 1, \ldots, k_n - 1\}$ among the FDH points $(\mathcal{X}_\ell, \mathcal{Y}_\ell)$, $\ell = 1, \ldots, \mathcal{N}$ (function `quad_spline_est`). Because the number of knots k_n determines the complexity of the spline approximation, its choice may then be viewed as model selection through the minimization of the following two information criteria:

$$AIC(k) = \log\left(\sum_{i=1}^{n} |y_i - \hat{\varphi}_n(x_i)|\right) + 2(k+2)/n,$$

$$BIC(k) = \log\left(\sum_{i=1}^{n} |y_i - \hat{\varphi}_n(x_i)|\right) + \log n \cdot (k+2)/n.$$

The first one (option `type = "AIC"`) is similar to the famous Akaike information criterion (Akaike, 1973) and the second one (option `type = "BIC"`) to the Bayesian information criterion (Schwartz, 1978). A small number of knots is typically needed as elucidated by the asymptotic theory.

For the implementation of the monotone and concave spline estimator $\hat{\varphi}_n^\star$, just apply the same scheme as above by replacing the FDH points $(\mathcal{X}_\ell, \mathcal{Y}_\ell)$ with the DEA points $(\mathcal{X}_\ell^*, \mathcal{Y}_\ell^*)$ (see `dea_est`).

Value

Returns an integer

Author(s)

Hohsuk Noh

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle, in *Second International Symposium of Information Theory*, eds. B. N. Petrov and F. Csaki, Budapest: Akademia Kiado, 267–281.

Daouia, A., Noh, H. and Park, B.U. (2013). Data Envelope Fitting with Constrained Polynomial splines. *TSE Working Paper*, http://www.tse-fr.eu/images/doc/wp/etrie/wp_tse_449.pdf.

Schwartz, G. (1978). Estimating the dimension of a model, *Annals of Statistics*, 6, 461–464.

See Also

`quad_spline_est`

Examples

```
data("green")
x <- seq(min(log(green$COST)), max(log(green$COST)), length.out=1001)
quad_spline_est_kn(log(green$COST), log(green$OUTPUT), x, cv=1, type="AIC")
quad_spline_est_kn(log(green$COST), log(green$OUTPUT), x, cv=1, type="BIC")
```

Index