npbr: an R package for nonparametric boundary regression

Abdelaati Daouia[*], Thibault Laurent[†] and Hohsuk Noh[‡]

December 11, 2013

The package **npbr** 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).

1 Introduction

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

$$\begin{aligned} \Psi = \{(x,y) | y \le \varphi(x)\} \quad &\supseteq \quad \{(x,y) | f(x,y) > 0\}, \\ \{(x,y) | y > \varphi(x)\} \quad &\subseteq \quad \{(x,y) | f(x,y) = 0\}, \end{aligned}$$

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 $\{(x_i, y_i),\ i = 1, \dots, n\}$ 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 datasets as well.

2 Data examples

Two datasets are included in this package:

- the dataset `nuclear` 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. See Daouia, Girard, and Guillou (2013) for more details.

[*] Institute of Statistics, UCL and Toulouse School of Economics (abdelaati.daouia@tse-fr.eu)

[†] Toulouse School of Economics, France (thibault.laurent@univ-tlse1.fr)

[‡] Department of Statistics, Sookmyung Women's University (word5810@gmail.com)

- the dataset `green` consists of 123 American electric utility companies. As in the set-up of Gijbels, Mammen, Park, and Simar (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).

Each of these datasets contains only two variables: one input and one output. To load these datasets, we do:

```
> require("npbr")
> data("green")
> data("nuclear")
```

The scatterplots are displayed in Figure 1 as follows:

```
> plot(log(OUTPUT)~log(COST), data=green, pch=16,col='blue2')
> plot(ytab~xtab, data=nuclear, pch=16,col='blue2',
+  xlab="temperature of the reactor vessel", ylab="fracture toughness")
```

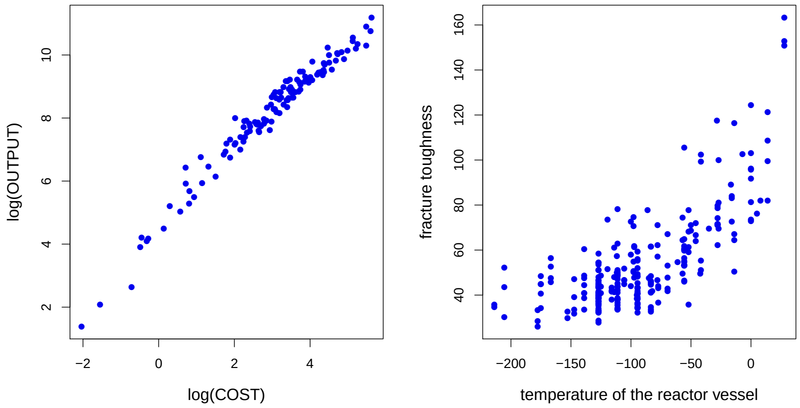

Figure 1: Scatterplots of the 123 American electric utility companies' data (left) and the 254 nuclear reactors' data (right).

3 Main functions

This section describes in detail the main functions of the **npbr** package. The two first arguments of these functions correspond to the observed inputs $x_1, \ldots, x_n$ and the observed outputs $y_1, \ldots, y_n$. The third argument is a numeric vector of evaluation points in which the estimator is to be computed. Basically, the user can generate a regular sequence of size 1000, from the minimum value of input x_i to their maximum value. The other arguments of the functions depend on the underlying statistical methods.

3.1 DEA, FDH and linearized FDH estimators

The function `dea_est` implements the empirical FDH (free disposal hull), LFDH (linearized FDH) and DEA (data envelopment analysis) frontier estimators programmed earlier in **DEA** package (Bogetoft and Otto, 2001). 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)).

To illustrate the difference between these three empirical estimators on the green data, we first create the vector of evaluation points:

```
> x.green <- seq(min(log(green$COST)), max(log(green$COST)),
+  length.out=1001)
```

Then we compute the DEA, FDH and LFDH estimates:

```
> y.dea<-dea_est(log(green$COST), log(green$OUTPUT),
+  x.green, type="dea")
> y.fdh<-dea_est(log(green$COST), log(green$OUTPUT),
+  x.green, type="fdh")
> y.lfdh=dea_est(log(green$COST), log(green$OUTPUT),
+  x.green, type="lfdh")
```

The resulting piecewise linear curves are graphed in Figure 2 using the following instructions:

```
> plot(log(OUTPUT)~log(COST), data=green)
> lines(x.green, y.dea, lty=1, lwd=4, col="red")
> lines(x.green, y.fdh, lty=2, lwd=4, col="blue")
> lines(x.green, y.lfdh, lty=3, lwd=4, col="green")
> legend("topleft", legend=c("dea","fdh","lfdh"),
+  col=c("red","blue","green"), lty=1:3, lwd=4)
```

3.2 Local linear frontier estimator

The function `loc_est` computes the local linear smoothing frontier estimator of Hall, Park, and Stern (1998). The implemented estimator of $\varphi(x)$ is defined by

$$\begin{aligned}\hat{\varphi}_{n,LL}(x) &= \min\Big\{z : \text{there exists } \theta \geq 0 \text{ such that } y_i \leq z + \theta(x_i - x) \\ &\qquad \text{for all } i \text{ such that } x_i \in (x-h, x+h)\Big\}.\end{aligned}$$

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

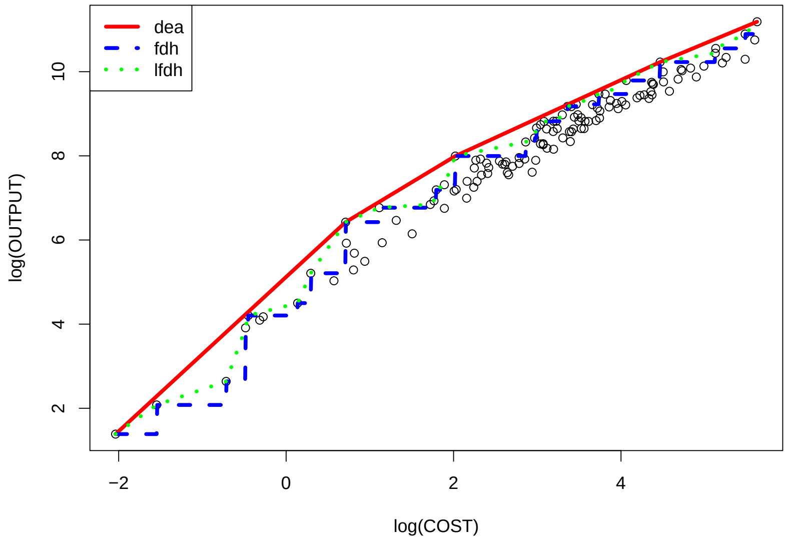

Figure 2: DEA, FDH and LFDH estimates of the optimal frontier.

Actually, to initiate Hall and Park's bootstrap device, one needs to set a pilot bandwidth, which seems to be quite critical to the quality of the local linear frontier. To see how this estimator performs in the case of `nuclear` data, we first fix the vector of evaluation points:

```
> x.nucl <- seq(min(nuclear$xtab), max(nuclear$xtab),
+  length.out=1001)
```

Then we evaluate the local linear estimate by using, for instance, the value 40 as the pilot bandwidth. The value 79.11877 is the resulting optimal bandwidth computed over 100 bootstrap replications via the function `loc_est_bw`. It should be clear that the computational time of this function is not negligible.

```
> y.loc.opt<-loc_est(nuclear$xtab, nuclear$ytab, x.nucl, h=79.11877)
> y.loc<-loc_est(nuclear$xtab, nuclear$ytab, x.nucl, h=40)
```

The obtained estimates for both initial and final bandwidths 40 and 79.11877 are superimposed in Figure 3 as follows:

```
> plot(ytab~xtab, data=nuclear)
> lines(x.nucl, y.loc.opt, lty=1, lwd=4, col="red")
> lines(x.nucl, y.loc, lty=2, lwd=4, col="blue")
> legend("topleft",legend=c("h=79.11877", "h=40"),
+  col=c("red","blue"), lwd=4, lty=c(1,2))
```

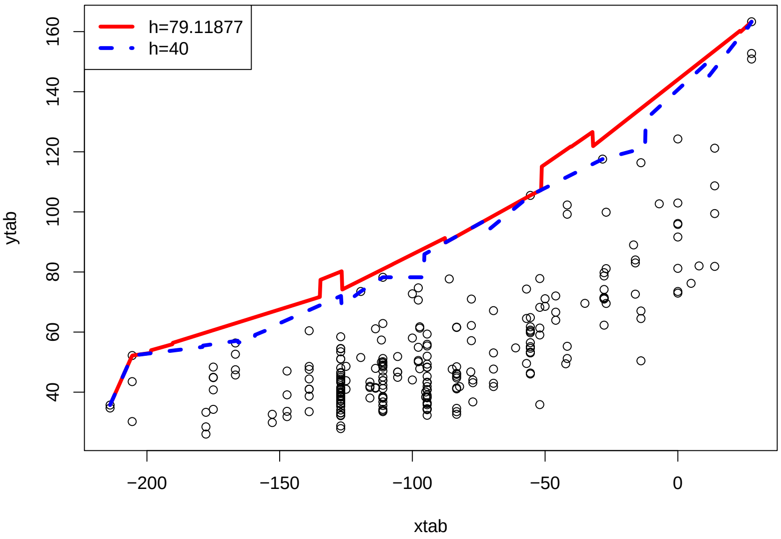

Figure 3: Local linear estimates for the upper support extremity.

3.3 Polynomial estimators

The function `poly_est` is an implementation of the polynomial-type estimators of Hall, Park, and Stern (1998) for support frontiers and boundaries.

Here, 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 $\theta \in \mathbb{R}^{p+1}$ subject to the envelopment constraints $\varphi_\theta(x_i) \geq y_i$, $i = 1, \ldots, n$.

The polynom degree p has to be fixed by the user in the 4th argument of the function. For example, the polynomial boundaries of degrees 2 and 4 can be computed in the case of `nuclear` data as follows:

```
> y.poly.2<-poly_est(nuclear$xtab, nuclear$ytab, x.nucl, deg=2)
> y.poly.4<-poly_est(nuclear$xtab, nuclear$ytab, x.nucl, deg=4)
```

The obtained estimators are graphed in Figure 4 in the following way:

```
> plot(ytab~xtab, data=nuclear)
> lines(x.nucl, y.poly.2, lty=1, lwd=4, col="red")
> lines(x.nucl, y.poly.4, lty=2, lwd=4, col="blue")
> legend("topleft",legend=c("degree=2", "degree=4"),
+  col=c("red","blue"), lwd=4, lty=c(1,2))
```

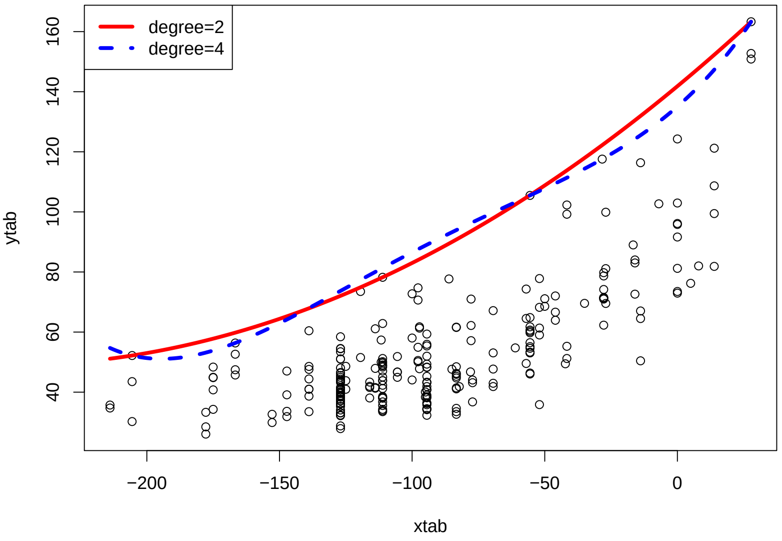

Figure 4: Polynomial estimators of degrees 2 and 4 for the data edge.

3.4 Quadratic spline estimators

Description of the method

The function `quad_spline_est` is an implementation of the constrained quadratic spline smoother proposed by Daouia et al. (2013). 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 $N = k_n + 2$ 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 $\alpha \in \mathbb{R}^N$ 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.* 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 $\hat{\alpha}^{\star} \in \mathbb{R}^N$ 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 $\hat{\varphi}_n$ is 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`.

Optimal number of knots

The function `quad_spline_est_kn` computes the optimal number of knots for the constrained quadratic spline fit proposed by Daouia *et al.*. For the implementation of the monotone quadratic spline smoother $\hat{\varphi}_n$, the authors first suggest using the set of knots $\{t_j = \mathcal{X}_{[j\mathcal{N}/k_n]},\ j = 1,\ldots,k_n-1\}$ among the FDH points $(\mathcal{X}_\ell,\mathcal{Y}_\ell)$, $\ell = 1,\ldots,\mathcal{N}$, as described above. 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:

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

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 $(\mathcal{X}_\ell^*,\mathcal{Y}_\ell^*)$.

Practical guidelines

We describe here how to effect the necessary computations of the quadratic spline fit under both separate and simultaneous shape constraints by making use of the `green` data. When only the monotonicity constraint is of interest, we first determine the optimal number of knots via the AIC criterion:

```
> (kn.aic.mono<-quad_spline_est_kn(log(green$COST), log(green$OUTPUT),
+  x.green, cv=0, type="AIC"))

[1] 6
```

We get the same optimal number of knots by applying the BIC criterion. The monotonic spline $\hat{\varphi}_n$ can then be produced as follows:

```
> y.quad.1<-quad_spline_est(log(green$COST), log(green$OUTPUT),
+  x.green, kn=kn.aic.mono, cv=0)
```

When the concavity constraint is also of interest, we obtain the optimal number of knots via the BIC criterion and the corresponding constrained spline $\hat{\varphi}_n^\star$ by proceeding as follows:

```
> (kn.bic.conca<-quad_spline_est_kn(log(green$COST), log(green$OUTPUT),
+  x.green, cv=1, type="BIC"))

[1] 1

> y.quad.2<-quad_spline_est(log(green$COST), log(green$OUTPUT),
+  x.green, kn=kn.bic.conca, cv=1)
```

To compute the smoother $\hat{\varphi}_n^\star$ by employing all the DEA points as knots, we use:

```
> y.quad.3<-quad_spline_est(log(green$COST), log(green$OUTPUT),
+  x.green, cv=1,
+  all.dea=TRUE)
```

The resulting three constrained estimators of the econometric frontier (*i.e.* the set of the most efficient electric utility companies) are graphed in Figure 5.

```
> plot(log(OUTPUT)~log(COST), data=green)
> lines(x.green, y.quad.1, lty=2, lwd=4, col="red")
> lines(x.green, y.quad.2, lty=2, lwd=4, col="blue")
> lines(x.green, y.quad.3, lwd=4, lty=1)
> legend("topleft", col=c("red","blue","black"), lty=c(2,2,1),
+  legend=c("mono(kn=6)", "mono + concav (kn=1)",
+  "mono + concav (kn=all DEA points)"), lwd=4, cex=0.8)
```

4 Numerical illustrations

A comparison among the different estimation methods described above has been undertaken by Daouia et al. (2013) via some simulation experiments. To encourage others to explore these methods, we provide in this section guidelines that can help the users to reproduce the Monte Carlo results obtained in Daouia *et al.*

4.1 Simulated data

The function `simulate.data` is an implementation of the experimental method by Daouia *et al.* It generates a random sample following the model $y_i = \varphi(x_i)\, v_i$, where x_i is uniform on $[0,1]$ and v_i, independent of x_i, is Beta(β,β) with values of $\beta = 0.5, 1$ and 3 (corresponding, respectively, to a joint density of the (x_i, y_i)'s increasing toward infinity, having a jump or decreasing to zero as it approaches the support boundary). The function has three arguments: the first one n is the sample size, the second one `funs` specifies the frontier function φ and takes 3 values:

- if `funs=0`, the true frontier is linear and equal to $\varphi(x) = x$,
- if `funs=1`, the true frontier is concave and equal to $\varphi(x) = \sqrt{x}$,
- if `funs=1`, the true frontier is not concave and equal to $\varphi(x) = \frac{exp(-5+10x)}{1+exp(-5+10x)}$.

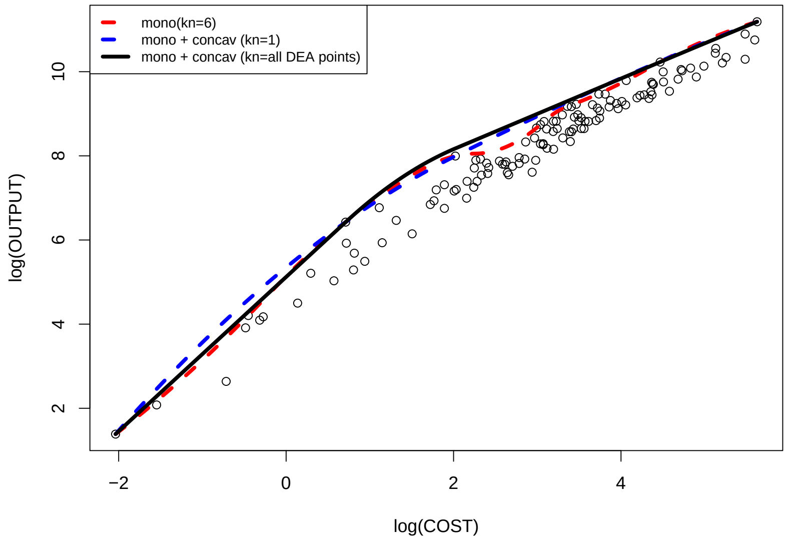

Figure 5: The constrained quadratic spline boundaries $\hat{\varphi}_n$ and $\hat{\varphi}_n^\star$.

The last argument `betav` is the parameter β.

```
> simulate.data<- function(n, funs=1, betav=0.5)
+ {
+  # internal function
+  Fron<-function(x,funs)
+  {
+   if (funs==1) { return(exp (-5 + 10*x)/(1 + exp(-5 + 10*x)))}
+   if (funs==2) { return(sqrt(x))}
+   if (funs==3) { return(x)}
+  }
+
+   xtab <- runif(n, 0, 1)
+   V <-rbeta(n, betav, betav)
+   ytab <- Fron(xtab, funs)*V
+
+   return(data.frame(xtab=xtab, ytab=ytab))
+ }
```

4.2 Some Monte Carlo evidence

To reproduce, for instance, the Monte Carlo estimates obtained in Table 1 of Daouia et al. (2013), we first generate 200 simulated samples in the case $n = 25$, $\beta = 0.5$ and $\varphi(x) = x$. Results are saved in `y.dea`.

```
> N<-200
> x.sim <- seq(0, 1, length.out=1000)
> y.dea<-matrix(0, N, 1000)
> for(k in 1:N)
+ {
+  don<-simulate.data(25)
+  y.dea[k,]<-dea_est(don$xtab, don$ytab, x.sim, type="dea")
+ }
```

We replace in `y.dea`, values equal to `-Inf` by 0:

```
> y.dea[is.infinite(y.dea)]<-0
```

To get the residuals of each simulation:

```
> error.dea<-matrix(x.sim, N, 1000, byrow=TRUE)-y.dea
```

To assess the empirical mean integrated squared error (MISE), the empirical integrated squared bias (IBIAS2) and the empirical integrated variance (IVAR):

```
> (IBIAS2<-mean((x.sim-apply(y.dea,2,mean))^2))
> (IVAR<-mean((y.dea-matrix(apply(y.dea,2,mean),N,1000,byrow=TRUE))^2))
> (MISE<-IBIAS2+IVAR)
```

References

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

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

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

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

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

Deprins, D., Simar, L. and Tulkens H. (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., Park, B.U. and Stern, S.E. (1998). On polynomial estimators of frontiers and boundaries. *Journal of Multivariate Analysis*, **66**, 71-98.

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

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

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

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

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