
A Nonparametric Monotone Regression Method for Bernoulli Responses with Applications to Wafer Acceptance Tests

Jyh-Jen Horng Shiau

(Joint work with **Shuo-Huei Lin and Cheng-Chih Wen**)

Institute of Statistics

National Chiao Tung University

Hsinchu, Taiwan 30010

e-mail: jyhjen@stat.nctu.edu.tw or jyhjen@gmail.com

Presented at Conference on Nonparametric Statistics for Big Data

the WAHBA conference

June 5, 2014, UW-Madison

Grace and me

- *Came to Madison in June 1980*
- *First child born in October 1980*
- *Statistics 860 Estimation of Functions from Data*
- *Beauty of the smoothing spline*
- *A turning point—the chat in front of the mailbox*
- *All the freedom*
- *Sabbatical*
- *The telephone conference → the 11th*
- *Me in Grace's webpage: <http://www.stat.wisc.edu/~wahba/>*

Outline

- *Motivated Example*
- *Related Works*
- *Proposed Methodology*
- *Simulated Examples*
- *A Comparative Study*
- *Conclusions*

Motivated Example — WAT/EC

- *Semiconductor manufacturing*
- *Wafer Acceptance Test (WAT)*
 - *electrical characteristics of devices*
 - *voltage, current, resistance, etc.*
 - *testkeys on scribe lines (gaps between chips)*
- *Engineering Control (EC)*
 - *more stringent process control than WAT*
- *Goal: process control and improvement*

Potential Problems — WAT/EC

- *Engineers tend to choose too many test items for EC*
 - *more than tolerable number of false alarms*
- *How to set appropriate control limits?*
 - *false alarms vs. detecting power*

WAT/EC Data

Pass=1

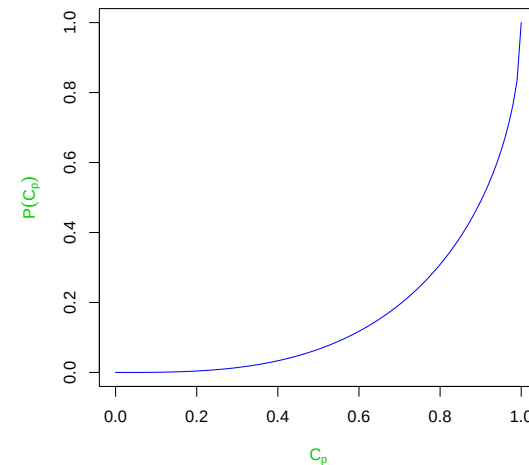
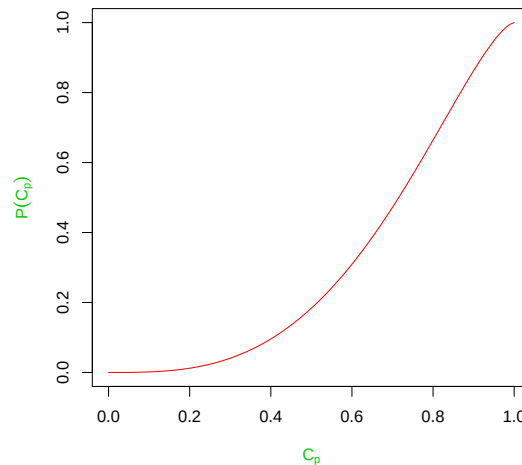
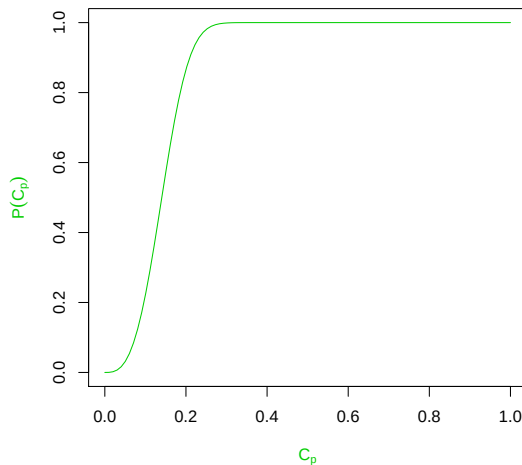
Fail=0

	Test Item1	Test Item2	...
<i>Wafer 1 ($C_p = 0.76$)</i>	<i>1</i>	<i>0</i>	<i>...</i>
<i>Wafer 2 ($C_p = 0.72$)</i>	<i>0</i>	<i>1</i>	<i>...</i>
<i>Wafer 3 ($C_p = 0.93$)</i>	<i>1</i>	<i>1</i>	<i>...</i>
<i>Wafer 4 ($C_p = 0.85$)</i>	<i>0</i>	<i>1</i>	<i>...</i>
<i>⋮</i>	<i>⋮</i>	<i>⋮</i>	<i>⋮</i>

C_p (circuit probe yield): proportion of chips on one wafer passing the circuit probe test

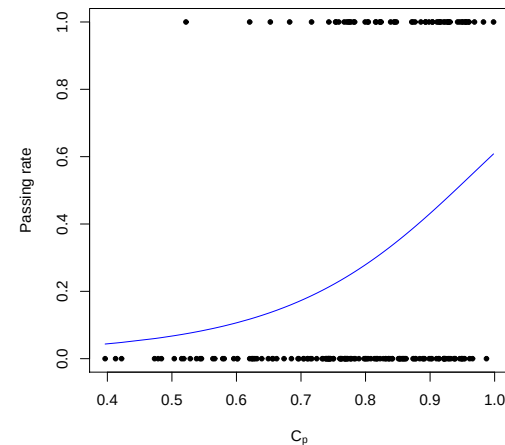
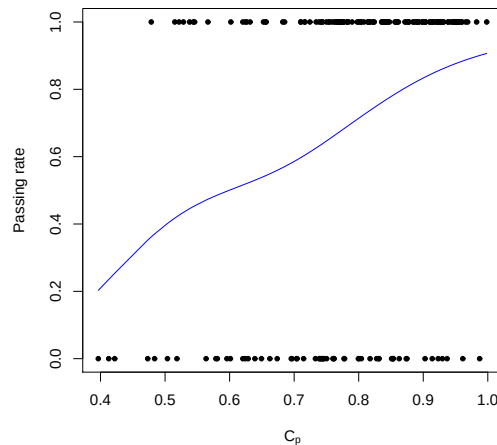
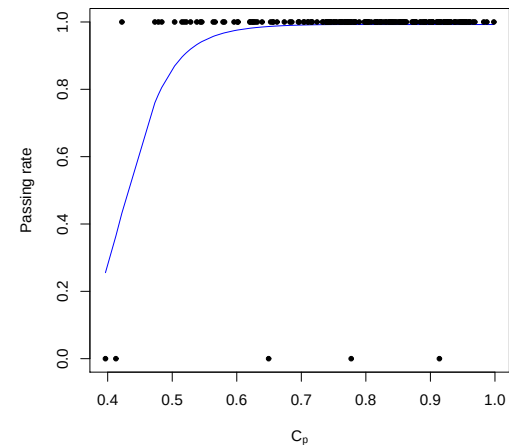
EC Performance Curves

- For each EC test item,
- Express *passing probability* as a function of C_p yield
- Establish the relationship between passing probability and C_p yield by *nonparametric regression*
- Three examples:



Setting Control Limits

- For the same process, EC performance curve varies with control limits
- For control limits $(0.88, 1.12)$, $(0.93, 1.07)$, and $(0.97, 1.03)$:



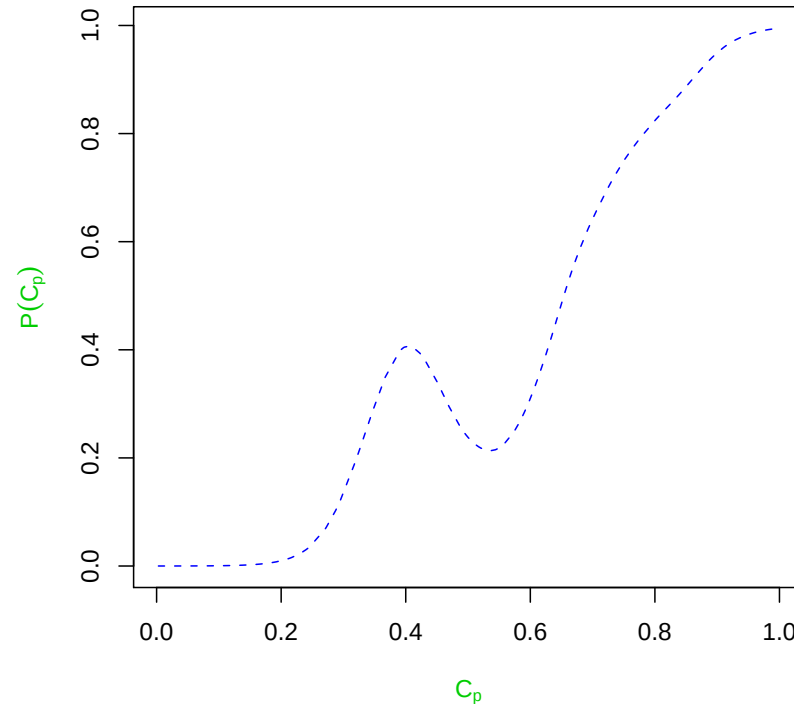
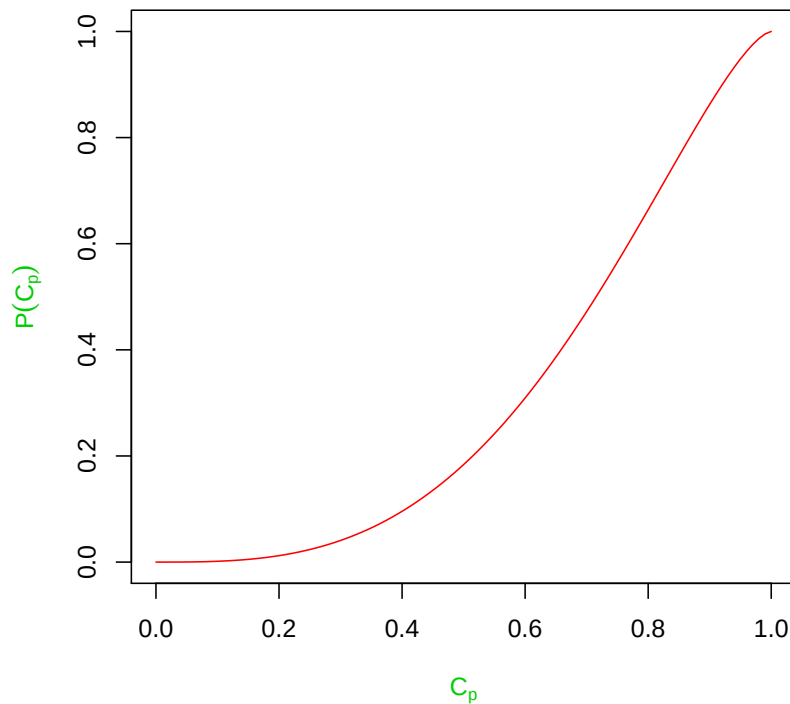
Statistical Model for WAT/EC Data

$$Y \sim \text{Bernoulli}(p(x))$$

- Independent variable X : C_p yield
- Response Y : pass/fail (1/0) of one EC test
- $p(x)$: passing probability of the wafer with C_p yield $X = x$
- $p(C_p)$: EC performance curve

Monotonicity

- *Passing rate should be positively associated with C_p yield.*
- *Need to develop a nonparametric regression estimator for GLM under the monotone constraint*



Related Works

- *Ramsay (1998)*

- *Imposed monotonicity in function estimation*

$$Y_i = f(x_i) + \epsilon_i, i = 1, \dots, n$$

by expressing the derivative of the original function as the exponential of a smooth function

$$g(x) = f'(x) = e^{\eta(x)}$$

- *Use B-spline approach*

- *Wang (2000)*

- *Extended the monotone regression method proposed by Ramsay (1998) to GLM models.*

Related Works (continued)

■ Zhang (2004)

- *Imposed monotonicity in function estimation by first estimating the derivative by **spline smoothing** and then truncated the negative part (for monotone increasing) of the spline estimate to zero*

$$\hat{g}_+(x) = \max\{\hat{g}(x), 0\}$$

- *Adopt **natural cubic spline** representation in Green and Silverman (1994) for efficient computation*

■ Gu (2002)

- *Smoothing spline smoother for responses from exponential families — GLM models*

Proposed Methodology — an Overview

- *Based on Zhang (2004), develop nonparametric monotone regression for GLM models*
- *By integrating the followings:*
 - *Penalized likelihood and quadratic approximation in Gu (2002)*
 - *Natural cubic spline estimation in Zhang (2004) and Green and Silverman (1994)*
 - *Modifying the natural cubic spline representation in Zhang (2004)*

Smoothing Splines

- *Nonparametric regression model*

$$Y_i = f(x_i) + \epsilon_i, i = 1, \dots, n$$

- *Minimizing the penalized least squares*

$$\frac{1}{n} \sum_{i=1}^n \{Y_i - f(x_i)\}^2 + \lambda \int_a^b [D^{(m)} f(x)]^2 dx$$

over the Sobolev Space

$W_2^m[a, b] = \{ f \mid f, f', \dots, f^{(m-1)} \text{ are absolutely continuous, } f^{(m)} \in L_2[a, b] \}$

where λ is the smoothing parameter controlling the tradeoff between the closeness to data and smoothness of the estimate

- *The solution is a natural spline of degree $2m - 1$ (popular cubic splines when $m = 2$).*

Penalized log-likelihood function

- Assume the response variables Y_i (with covariate x_i), $i = 1, \dots, n$, are i.i.d. samples from the exponential family with p.d.f.

$$f(y|x) = \exp\left\{\frac{y\eta(x) - b(\eta(x))}{a(\phi)} + c(y, \phi)\right\}$$

where $a(\cdot) > 0$, $b(\cdot)$, and $c(\cdot)$ are known functions, ϕ is a nuisance or scale parameter, $\eta(x)$ is the unknown function of interest depending on the covariate x

- Penalized log-likelihood function

$$-\frac{1}{n} \sum_{i=1}^n \{Y_i \eta(x_i) - b(\eta(x_i))\} + \frac{\lambda}{2} \int_a^b [D^{(m)} \eta(x)]^2 dx$$

where λ is a smoothing parameter.

Quadratic Approximation (Gu, 2002)

- The quadratic approximation of $-Y_i\eta(x_i) + b(\eta(x_i))$ at $\tilde{\eta}(x_i)$ is

$$\frac{1}{2}\tilde{\omega}_i\left\{\eta(x_i) - \tilde{\eta}(x_i) + \frac{\tilde{u}_i}{\tilde{w}_i}\right\}^2 + C_i$$

where $\tilde{u}_i = -Y_i + b'(\tilde{\eta}(x_i))$, $\tilde{\omega}_i = b''(\tilde{\eta}(x_i))$, and C_i is independent of $\eta(x_i)$.

- Let $\tilde{y}_i = \tilde{\eta}(x_i) - \frac{\tilde{u}_i}{\tilde{w}_i}$. Then the estimate of $\eta(\cdot)$ is obtained by recursively finding the minimizer (set to be the new $\tilde{\eta}(\cdot)$ in recursion) of the penalized weighted least squares functional

$$l = \frac{1}{n} \sum_{i=1}^n \tilde{\omega}_i \{\tilde{y}_i - \eta(x_i)\}^2 + \lambda \int_a^b [D^{(m)}\eta(x)]^2 dx$$

until convergence.

Estimation of $\eta(x)$

- Let $g(x) = \eta'(x)$ be a function in $W_2^2[a, b]$

$$W_2^2[a, b] = \{ f \mid f \text{ and } f' \text{ are absolutely continuous, } f'' \in L_2[a, b] \}$$

- For any x in $[a, b]$

$$\eta(x) = \eta(a) + \int_a^x g(u) du$$

- The penalized weighted least squares functional becomes

$$l = \frac{1}{n} \sum_{i=1}^n \tilde{\omega}_i \left\{ \tilde{y}_i - \eta(a) - \int_a^{x_i} g(x) dx \right\}^2 + \lambda \int_a^b [g''(x)]^2 dx$$

Natural Cubic Splines (Green and Silverman, 1994)

- Let $g(x)$ be a natural cubic spline on $[a, b]$. Let

$$\boldsymbol{\eta} = (\eta_1, \dots, \eta_n)^T, \boldsymbol{\gamma} = (\gamma_2, \dots, \gamma_{n-1})^T$$

$$\mathbf{g} = (g_1, \dots, g_n)^T$$

where $\gamma_i = g''(x_i)$ for $i = 2, \dots, n-1$ and $\gamma_1 = \gamma_n = 0$

- $\mathbf{g}$ and $\boldsymbol{\gamma}$ can specify a natural cubic spline if and only if

$$R\boldsymbol{\gamma} = Q^T \mathbf{g}.$$

- By Theorem 2.1 of Green and Silverman (1994)

$$\boldsymbol{\gamma} = R^{-1}Q^T \mathbf{g}$$

$$\int_a^b g''(x)^2 dx = \mathbf{g}^T K \mathbf{g}$$

where $K = QR^{-1}Q^T$

Construction of Q and R

■ $n = 6$

$$R = \begin{pmatrix} \frac{h_1+h_2}{3} & \frac{h_2}{6} & 0 & 0 \\ \frac{h_2}{6} & \frac{h_2+h_3}{3} & \frac{h_3}{6} & 0 \\ 0 & \frac{h_3}{6} & \frac{h_3+h_4}{3} & \frac{h_4}{6} \\ 0 & 0 & \frac{h_4}{6} & \frac{h_4+h_5}{3} \end{pmatrix}$$

$$Q = \begin{pmatrix} \frac{1}{h_1} & 0 & 0 & 0 \\ -\frac{1}{h_1} - \frac{1}{h_2} & \frac{1}{h_2} & 0 & 0 \\ \frac{1}{h_2} & -\frac{1}{h_2} - \frac{1}{h_3} & \frac{1}{h_3} & 0 \\ 0 & \frac{1}{h_3} & -\frac{1}{h_3} - \frac{1}{h_4} & \frac{1}{h_4} \\ 0 & 0 & \frac{1}{h_4} & -\frac{1}{h_4} - \frac{1}{h_5} \\ 0 & 0 & 0 & \frac{1}{h_5} \end{pmatrix}$$

■ $h_i = x_{i+1} - x_i$

Propositions

■ *Proposition 1: $\boldsymbol{\eta} = \eta(a) \cdot \mathbf{1}_n + M_{n,n}\mathbf{g}$*

where $M_{n,n} = \frac{1}{2}C_{n,n} - \frac{1}{24}D_{n,n-2}R_{n-2,n-2}^{-1}Q_{n-2,n}^T$

■ *Proposition 2: $M_{n,n}$ is invertible.*

Construction of C and D

■ $n = 6$

$$C = \begin{pmatrix} k_1 & k_2 & 0 & 0 & 0 & 0 \\ h_1 + k_1 & h_1 + k_2 & 0 & 0 & 0 & 0 \\ h_1 + k_1 & h_1 + h_2 + k_2 & h_2 & 0 & 0 & 0 \\ h_1 + k_1 & h_1 + h_2 + k_2 & h_2 + h_3 & h_3 & 0 & 0 \\ h_1 + k_1 & h_1 + h_2 + k_2 & h_2 + h_3 & h_3 + h_4 & h_4 & 0 \\ h_1 + k_1 & h_1 + h_2 + k_2 & h_2 + h_3 & h_3 + h_4 & h_4 + h_5 & h_5 \end{pmatrix}$$

$$D = \begin{pmatrix} k_3 & 0 & 0 & 0 \\ h_1^3 + k_3 & 0 & 0 & 0 \\ h_1^3 + h_2^3 + k_3 & h_2^3 & 0 & 0 \\ h_1^3 + h_2^3 + k_3 & h_2^3 + h_3^3 & h_3^3 & 0 \\ h_1^3 + h_2^3 + k_3 & h_2^3 + h_3^3 & h_3^3 + h_4^3 & h_4^3 \\ h_1^3 + h_2^3 + k_3 & h_2^3 + h_3^3 & h_3^3 + h_4^3 & h_4^3 + h_5^3 \end{pmatrix}$$

Where $k_1 = h_0 \left(2 + \frac{h_0}{h_1}\right)$, $k_2 = -\left(\frac{h_0^2}{h_1}\right)$, $k_3 = -2h_0^2 h_1$, and $h_0 = x_1 - a$.

Estimating $\mathbf{g}$ and $\eta(a)$

- Using the *back-fitting algorithm* to solve $\mathbf{g}$ and $\eta(a)$
- Set initial values of $\eta(a)$ and $\mathbf{g}$
- Fix $\mathbf{g}$, update $\eta(a)$ by minimizing $l(\eta(a), \mathbf{g})$
- Fix $\eta(a)$, update $\mathbf{g}$ by minimizing $l(\eta(a), \mathbf{g})$
- Iterate until convergence

Estimating $\eta(a)$ for given $\mathbf{g}$

- Given $\mathbf{g}$, obtain $\eta(a)$ by minimizing $l(\eta(a), \mathbf{g})$ via the Newton-Raphson method.
- That is, compute

$$\hat{\eta}^{new}(a) = \hat{\eta}^{old}(a) - \frac{l'(\hat{\eta}^{old}(a)|\mathbf{g})}{l''(\hat{\eta}^{old}(a)|\mathbf{g})}$$

iteratively until convergence.

Estimating g for given $\eta(a)$

- Given $\eta(a)$, the penalized weighted least squares

$$l = \mathbf{g}^T (M^T \tilde{W} M + n\lambda K) \mathbf{g} - 2\tilde{\mathbf{y}}^T \tilde{W} M \mathbf{g} + \tilde{\mathbf{y}}^T \tilde{W} \tilde{\mathbf{y}}$$

where

$$\tilde{W} = \text{diagonal}\{\tilde{\omega}_1, \dots, \tilde{\omega}_n\}$$

$$\tilde{\mathbf{y}} = (\tilde{y}_1 - \eta(a), \dots, \tilde{y}_n - \eta(a))^T$$

- Then the estimator $\hat{\mathbf{g}}$ that minimizes the penalized weighted least squares l is

$$\hat{\mathbf{g}} = (M^T \tilde{W} M + n\lambda K)^{-1} M^T \tilde{W} \tilde{\mathbf{y}}$$

Monotone Constraint

- Force $\hat{g}(x_i)$ to zero if $\hat{g}(x_i) \leq 0$. That is,

$$\hat{g}_+(x_i) = \max\{\hat{g}(x_i), 0\}$$

- The estimator of $\boldsymbol{\eta}$

$$\hat{\boldsymbol{\eta}} = \hat{\eta}(a) \cdot \mathbf{1}_n + M\hat{\mathbf{g}}_+$$

- $M\hat{\mathbf{g}}_+$ can be viewed as integrating the natural cubic spline that interpolates the points $\{(x_i, \hat{g}_+(x_i)), i = 1, \dots, n\}$.
- Denote the interpolating cubic spline by $\tilde{g}_+$

A Potential Problem and a Remedy

- When $\hat{g}_+(x_k) = \hat{g}_+(x_{k+1}) = 0$ for some k , it is then possible to have $\tilde{g}_+(x) < 0$ for some $x \in [x_k, x_{k+1}]$.

■ Then the monotonicity of $\hat{\eta}$ is not guaranteed.

- A remedy:

For $x_i \leq x \leq x_{i+1}$, $i = 1, \dots, n - 1$, let

$$\hat{\eta}_{mon}(x) = \hat{\eta}(a) + \sum_{j=1}^i \tau_j \int_{x_{j-1}}^{x_j} \tilde{g}_+(u) du + \tau_x \int_{x_i}^x \tilde{g}_+(u) du,$$

where τ_j is an indicator function defined as 1 if $\hat{\eta}_j > \hat{\eta}_{j-1}$ and 0 otherwise, and $\tau_x = 1$ when $\int_{x_i}^x \tilde{g}_+(u) du > 0$ and 0 otherwise.

Matrix Form

■ *Proposition 3* $\hat{\boldsymbol{\eta}}_{mon} = \hat{\eta}_a \cdot \mathbf{1}_n + SN\tilde{M}\bar{\mathbf{g}}$

where $\bar{\mathbf{g}} = \begin{pmatrix} \hat{\eta}_a \\ \hat{\mathbf{g}}_+ \end{pmatrix}$, $\tilde{M}$ is the $(n+1) \times (n+1)$ matrix given by

$$\tilde{M} = \begin{pmatrix} 1 & \mathbf{0}_n^T \\ \mathbf{1}_n & M \end{pmatrix};$$

N is the $n \times (n+1)$ matrix with elements n_{ij} , $1 \leq i, j \leq n$, given by

$n_{ii} = -1$, $n_{i,i+1} = 1$ for $1 \leq i \leq n$ and $n_{ij} = 0$ elsewhere; and

$S = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n)$ be the $n \times n$ matrix given by

$\mathbf{s}_i = (0_{i-1}^T, \tau_i \cdot \mathbf{1}_{n-i+1}^T)^T$ for $i = 1, \dots, n$.

Matrix Form (continued)

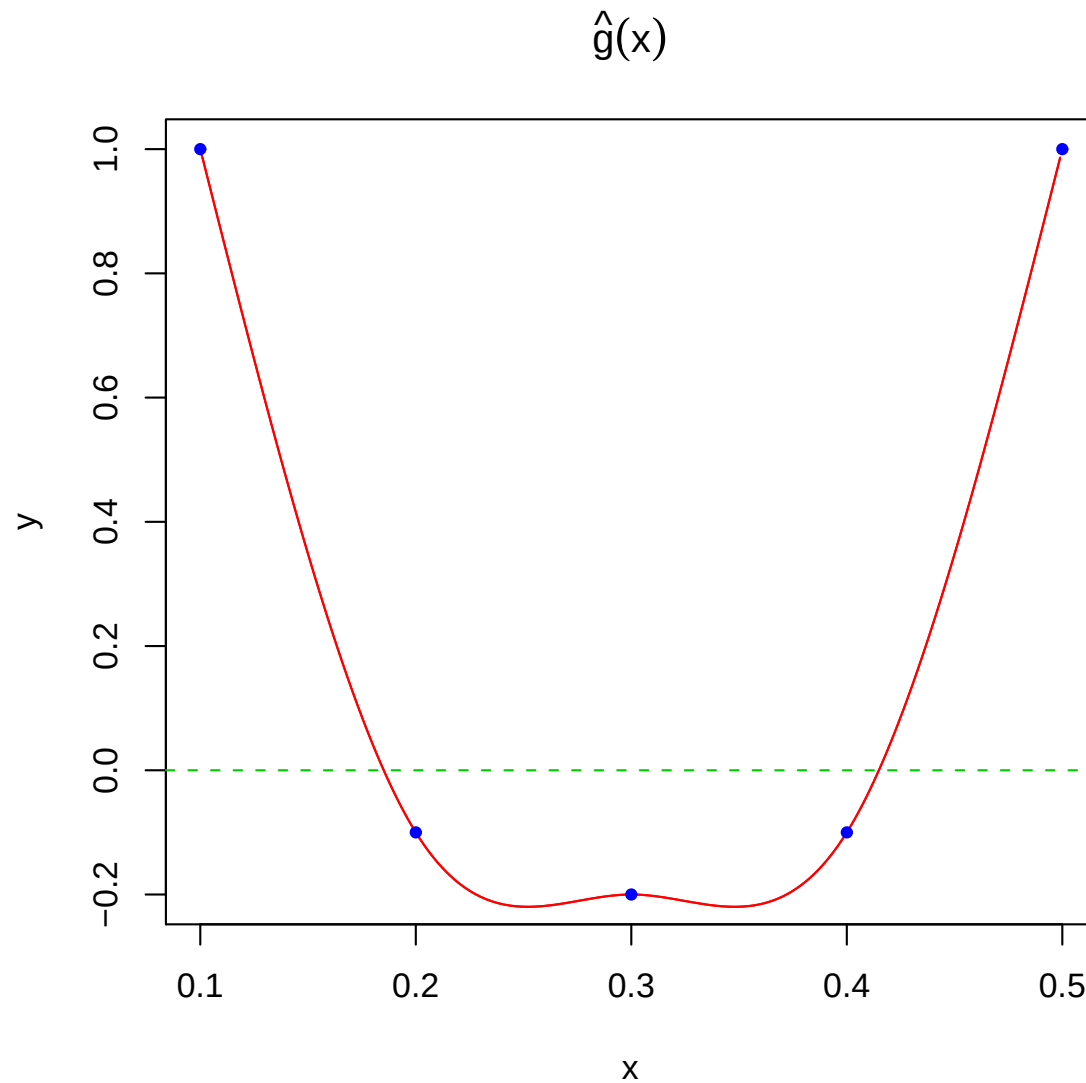
- The *effect of $\tilde{M}$ and N*

$$N\tilde{M}\bar{g} = N\tilde{M} \begin{pmatrix} \hat{\eta}_a \\ \hat{\mathbf{g}}_+ \end{pmatrix} = N \begin{pmatrix} \hat{\eta}_a \\ \hat{\eta}_1 \\ \vdots \\ \hat{\eta}_n \end{pmatrix} = \begin{pmatrix} \hat{\eta}_1 - \hat{\eta}_a \\ \hat{\eta}_2 - \hat{\eta}_1 \\ \vdots \\ \hat{\eta}_n - \hat{\eta}_{n-1} \end{pmatrix},$$

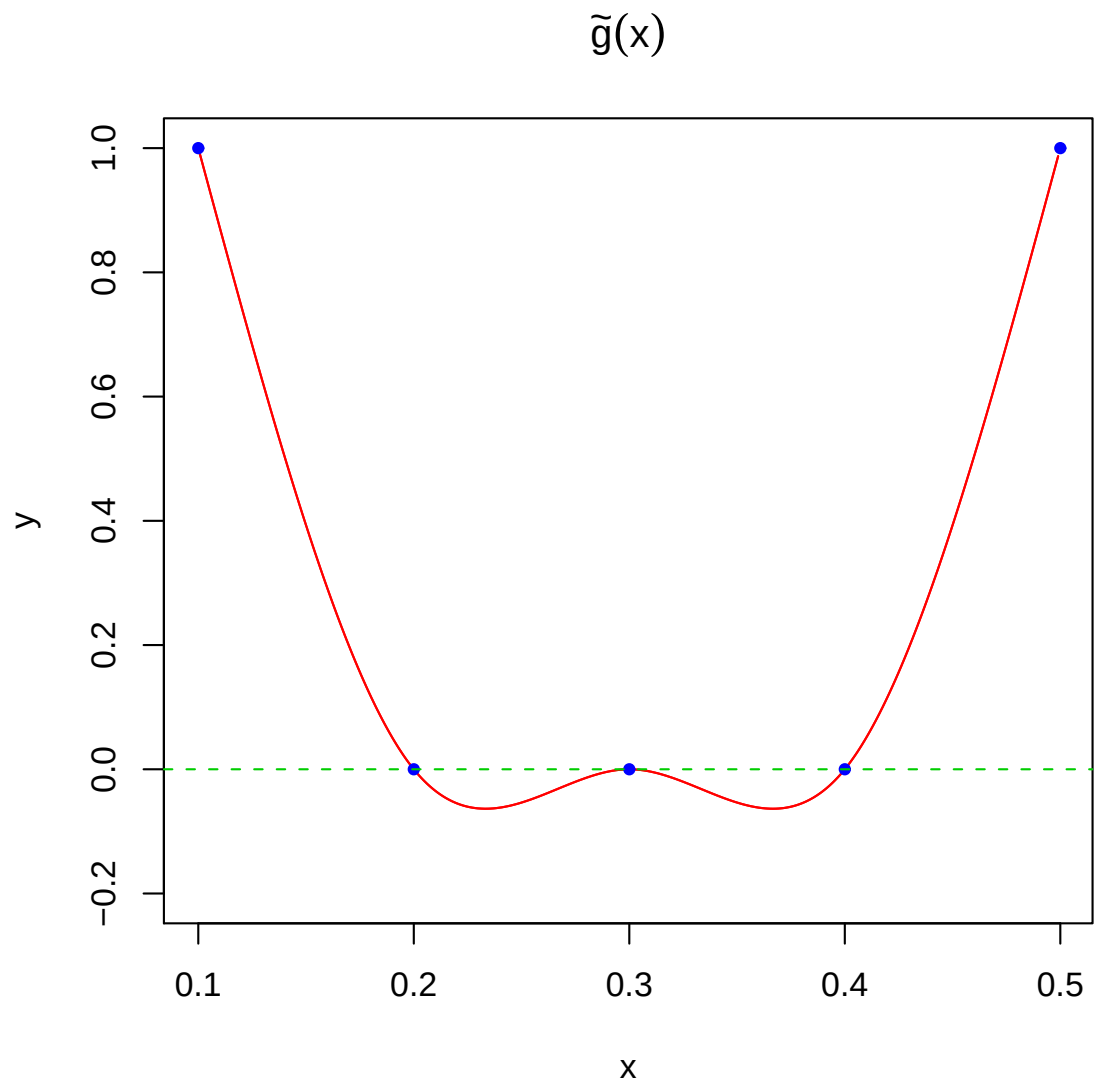
which computes the integral $\int_{x_i}^{x_{i+1}} \tilde{g}_+(u) du$ for each subinterval.

- The *effect of S* is to *accumulate* these integrals up to x_i but *skip* those integrals $\int_{x_i}^{x_{i+1}} \tilde{g}_+(u) du$ for which $\hat{\eta}_i \leq \hat{\eta}_{i-1}$.

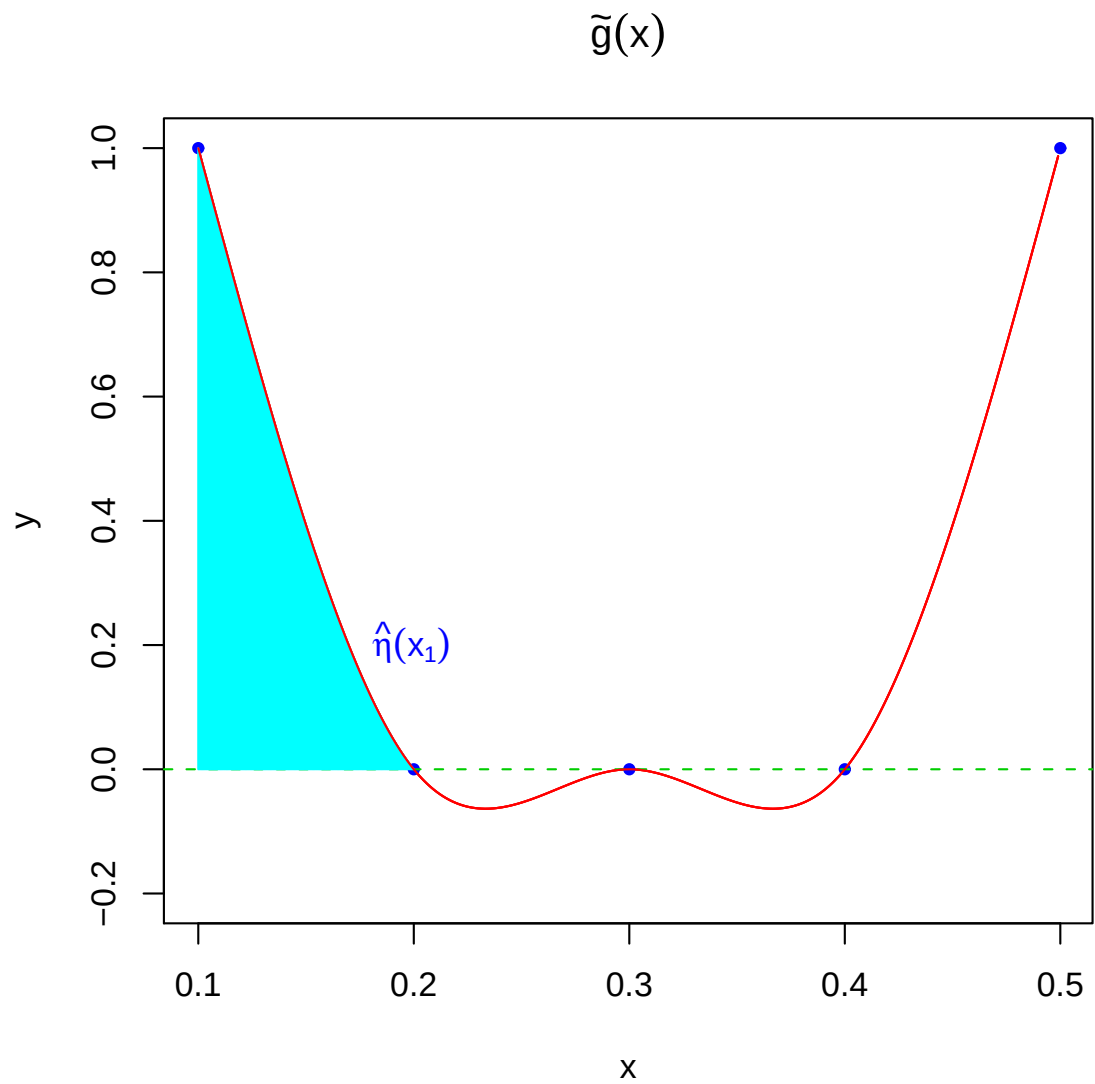
Illustration



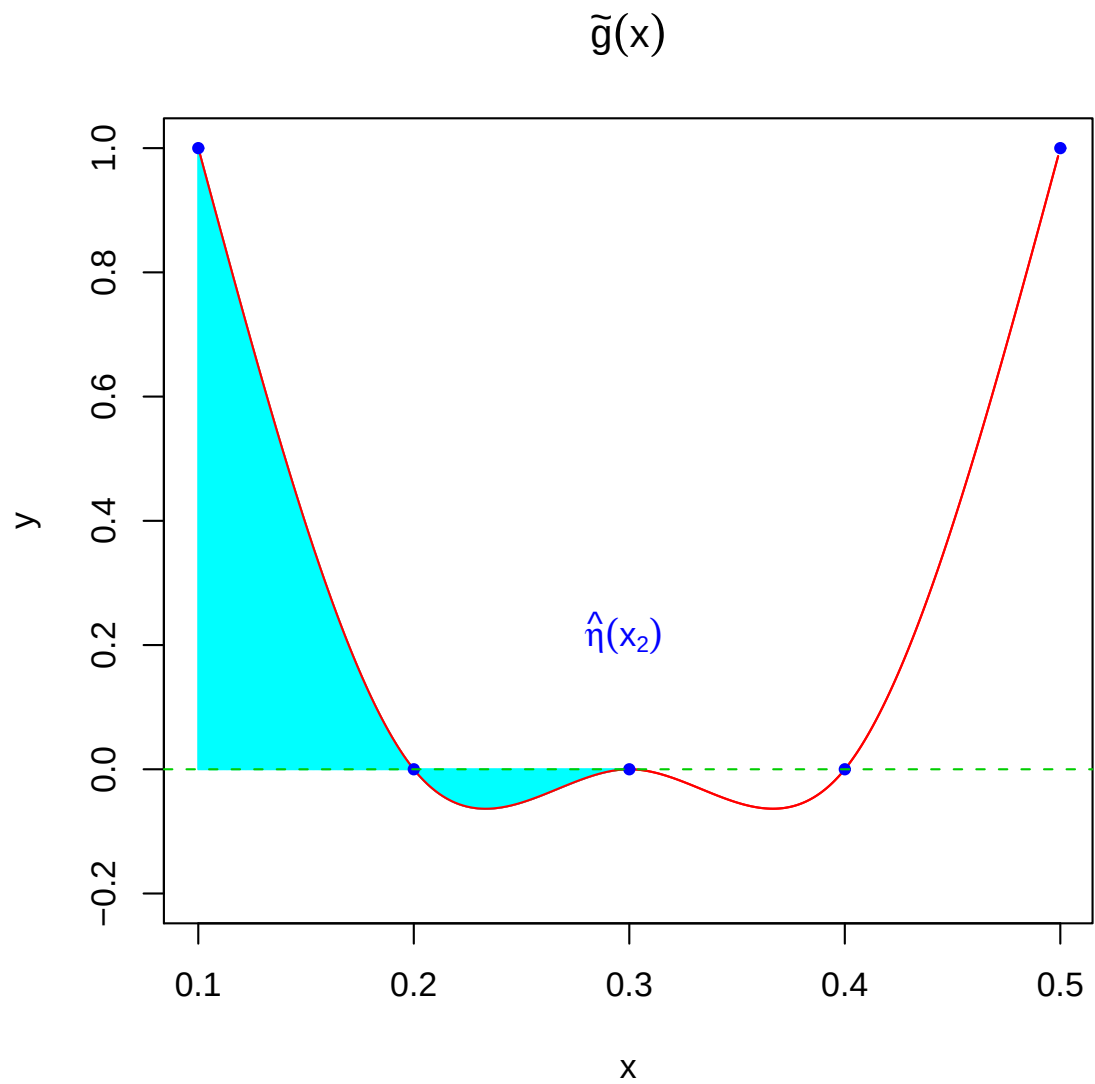
Illustration



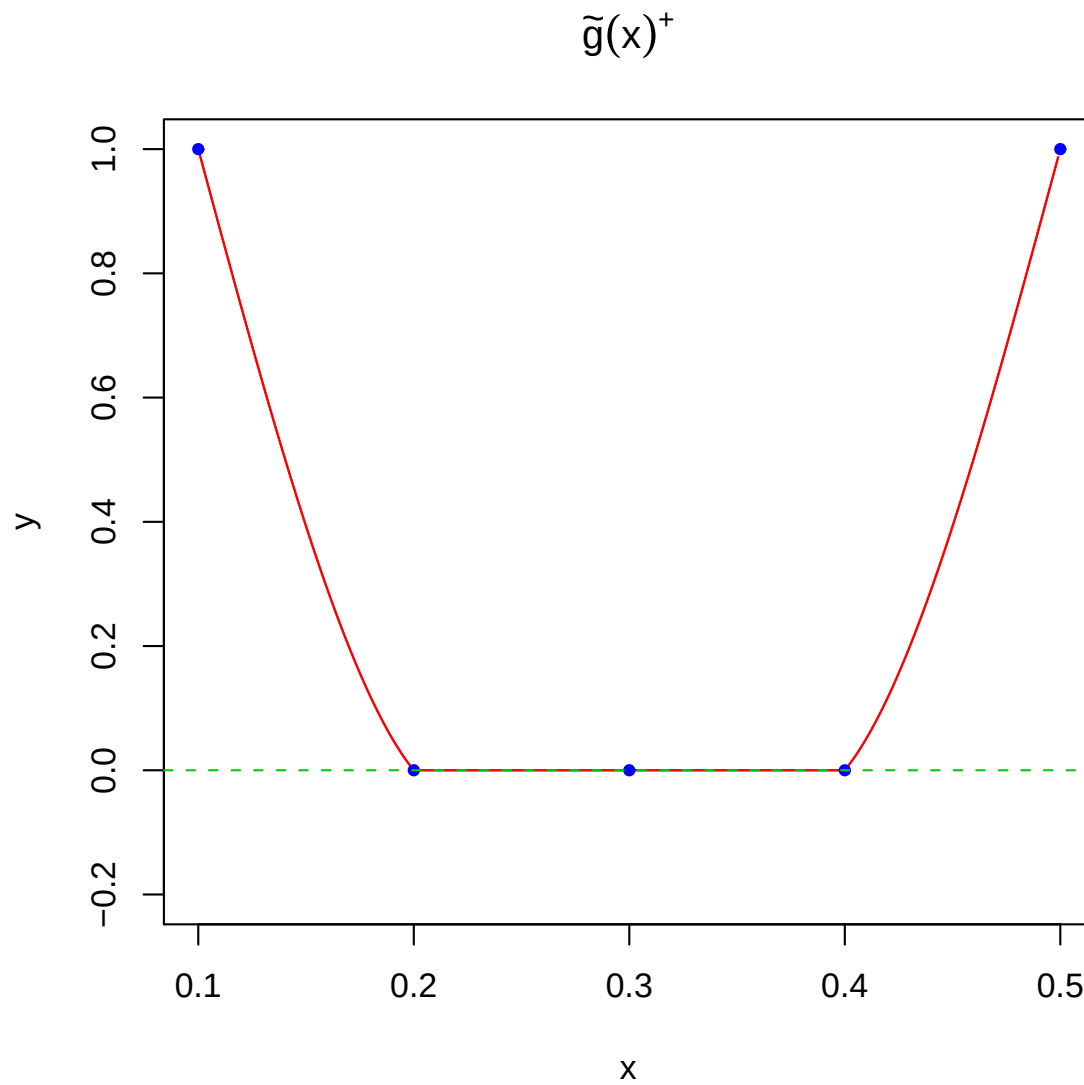
Illustration



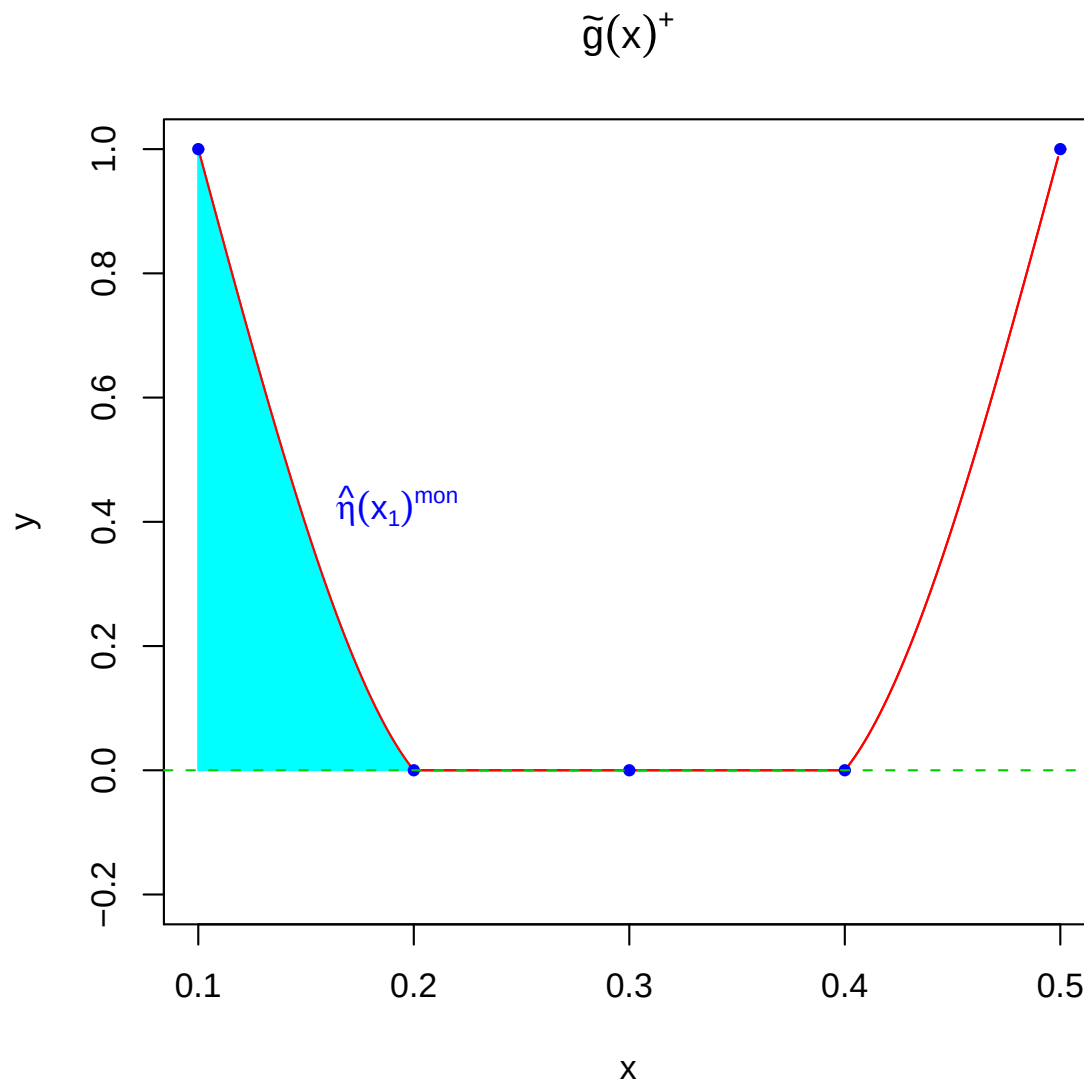
Illustration



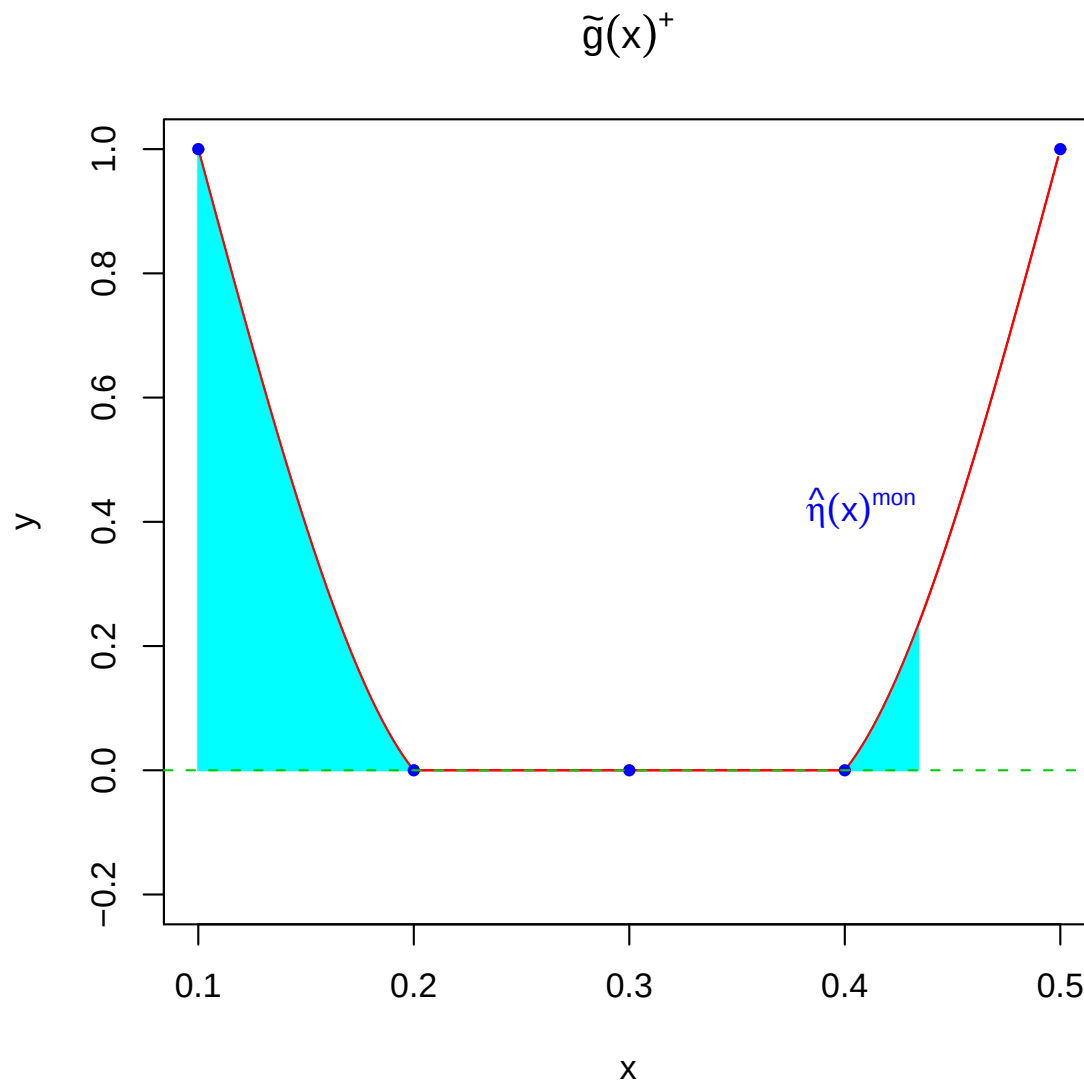
Illustration



Illustration



Illustration



Algorithm

- **Step 1** Calculate matrices C, D, Q, R , and K
- **Step 2** Set initial values $\hat{\mathbf{g}}^{(0)}$, $\hat{\eta}^{(0)}(a)$, and tolerance level T . Let $k = 0$.
- **Step 3** Fix $\hat{\mathbf{g}}^{(k)}$, update $\hat{\eta}(a)$ to $\hat{\eta}^{(k+1)}(a)$ by Newton-Ralphson method: compute

$$\hat{\eta}^{new}(a) = \hat{\eta}^{old}(a) - \frac{l'(\hat{\eta}^{old}(a)|\mathbf{g})}{l''(\hat{\eta}^{old}(a)|\mathbf{g})}$$

until convergence

- **Step 4** Fix $\hat{\eta}^{(k+1)}(a)$, update $\hat{\mathbf{g}}$ by

$$\hat{\mathbf{g}}^{(k+1)} = (M^T \tilde{W} M + n\lambda K)^{-1} M^T \tilde{W} \tilde{\mathbf{y}}$$

- **Step 5** If $\left| \frac{\hat{\eta}^{(k+1)}(a) - \hat{\eta}^{(k)}(a)}{\hat{\eta}^{(k)}(a)} \right| < T$, stop iteration, set

$$\hat{\mathbf{g}}_+^{(k+1)} = \max\{\hat{\mathbf{g}}^{(k+1)}, 0\}, \text{ and output}$$

$$\hat{\eta}_{mon} = \hat{\eta}^{(k+1)}(a) \mathbf{1}_n + SN \tilde{M} \bar{\mathbf{g}}^{(k+1)}.$$

Otherwise, increment k by 1 and return to Step 3.

Simulation 1

- *Generate*

$$y_i \sim \text{Bernoulli}(1, p(x_i)), x_i \sim U(0, 1)$$

for $i = 1, \dots, 200$.

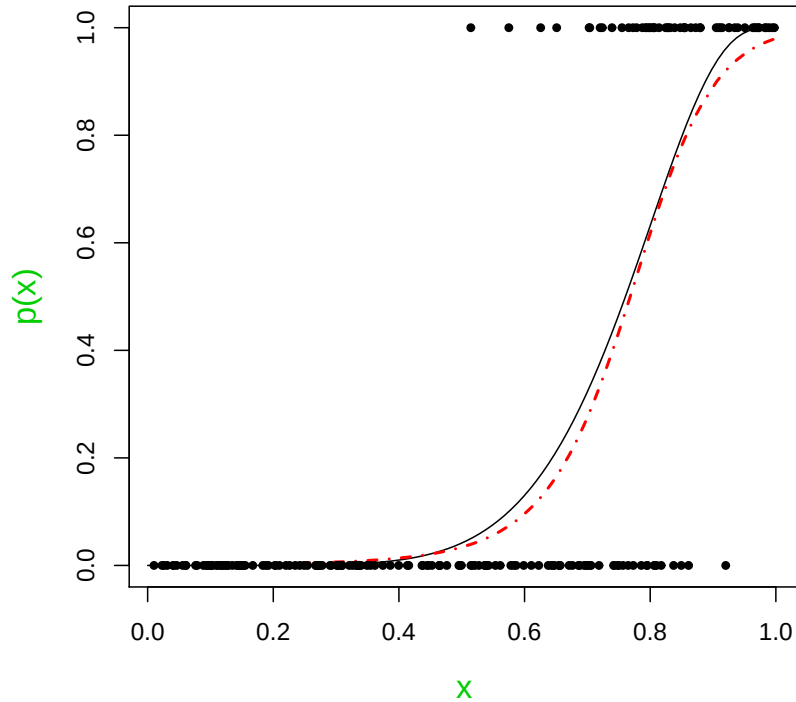
- *Use the function*

$$p(x) = 1 - (1 - x^{4.5})^{2.5}, x \in (0, 1)$$

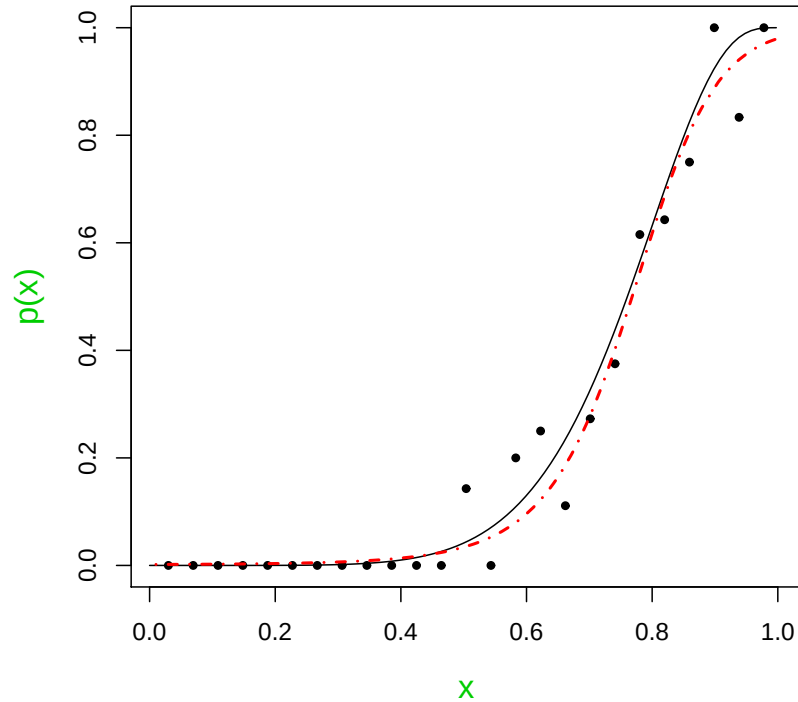
- $\lambda = 5 \times 10^{-5}$ by the GCV method.

Bernoulli Data

Bernoulli



Bernoulli



Dots in the left panel are observations (x_i, y_i) and in the right panel are binned data points $(m_j, \bar{y}_j)$,

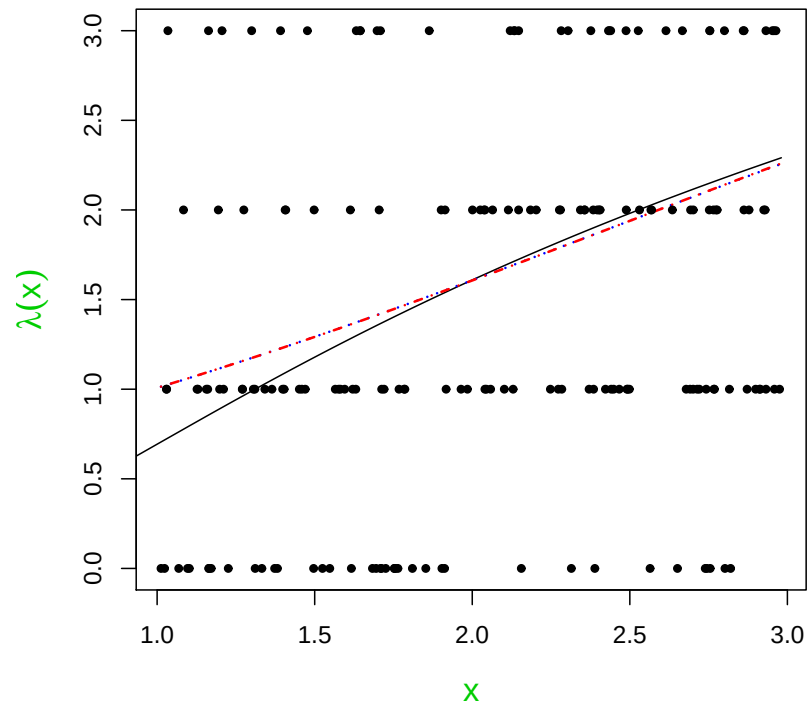
where m_j = midpoint and $\bar{y}_j$ = average of y_i 's in the bin interval j

Simulation 2

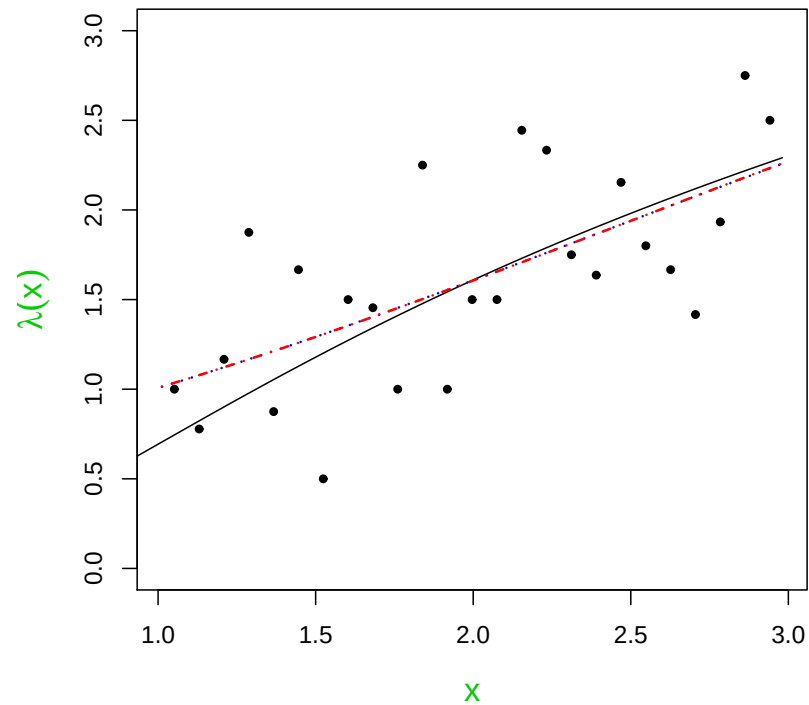
- Generate $y_i \sim \text{Poisson}(\lambda(x_i))$, $x_i \sim U(1, 3)$
for $i = 1, \dots, 200$.
- Use the mean functions
 - $\lambda(x) = \log(x^2 + 1)$ for $1 \leq x \leq 3$
- Choose λ_{GCV} and λ_{opt} (optimal in the sense of the averaged squared error (ASE))
- $ASE(\hat{\lambda}(x)) = \frac{1}{n} \sum_{i=1}^n (\hat{\lambda}(x_i) - \lambda(x))^2$
- Compare ASE of λ_{GCV} and λ_{opt}

Poisson Data

Poisson



Poisson



Dash-dot line is with λ_{opt} , $ASE(\lambda_{opt}) = 0.01190$.

Dot line is with λ_{GCV} , $ASE(\lambda_{GCV}) = 0.01189$

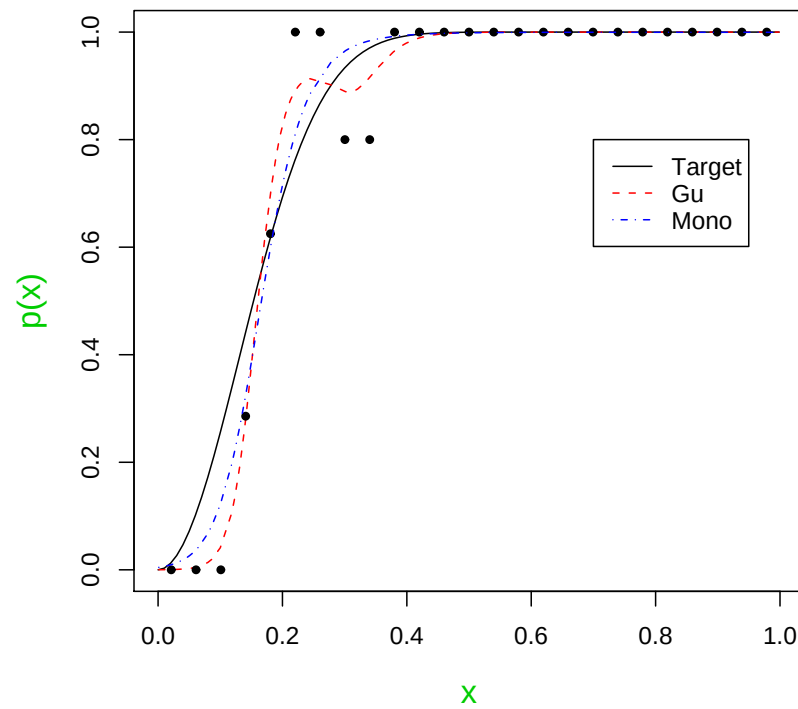
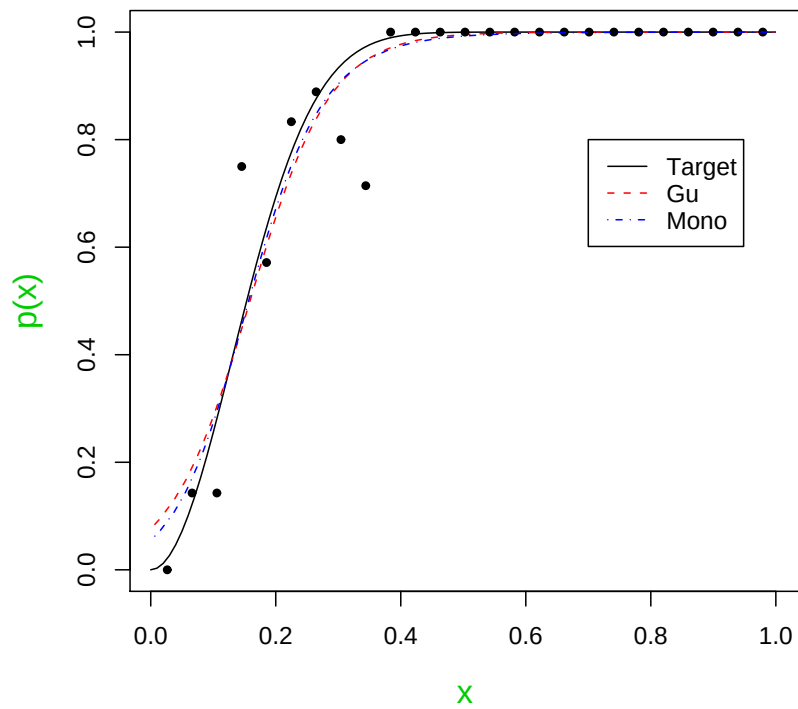
Constrained vs. Unconstrained

■ *Simulation:*

- *Generate $y_i \sim \text{Bernoulli}(1, p(x_i))$, $x_i \sim U(0, 1)$ for $i = 1, \dots, n$.*
- *Use the mean function $p(x) = 1 - (1 - x^\alpha)^\beta$*
- *Let $n = 50, 100, 200$*
- *For each sample size n , consider $(\alpha, \beta) = (1.98, 28)$, $(6.28, 17.67)$ and $(6.9, 1.1)$.*
- *Repeat 10000 times and calculate ASE for each replicate.*
- *Let $\hat{p}_g$ be the unconstrained estimator in Gu (2002) and $\hat{p}_m$ be the estimator with monotone constraint*

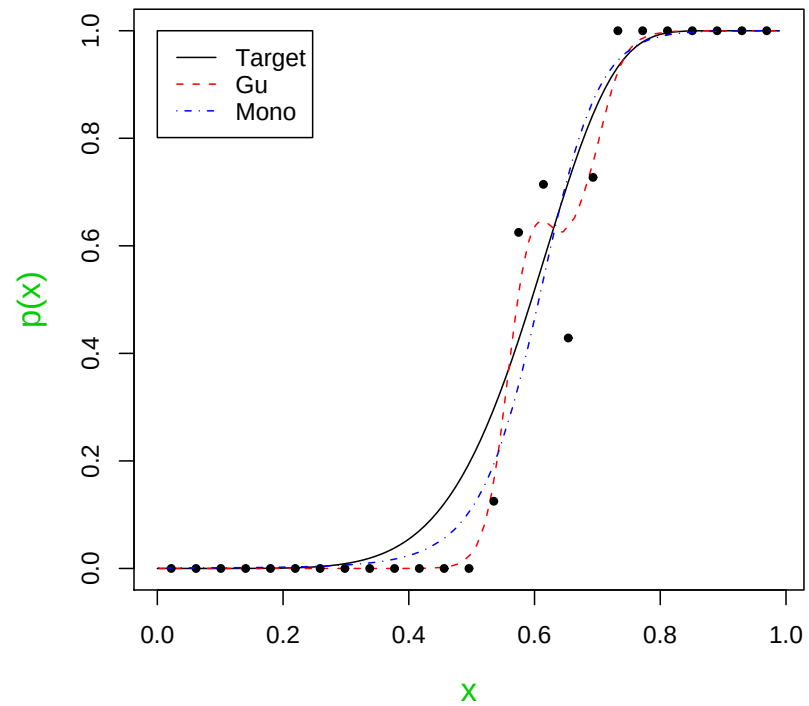
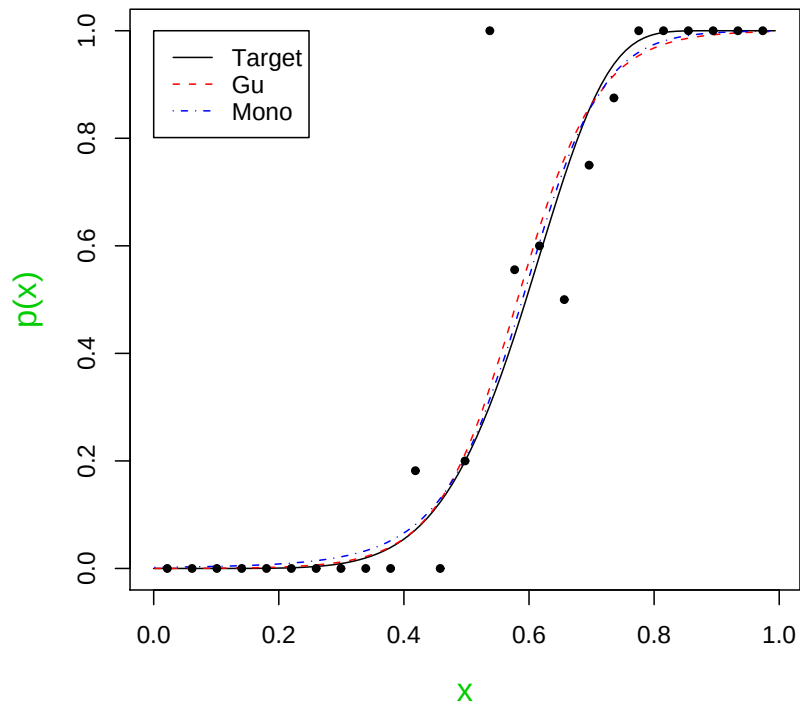
Illustration

$$\alpha = 1.98, \beta = 28$$



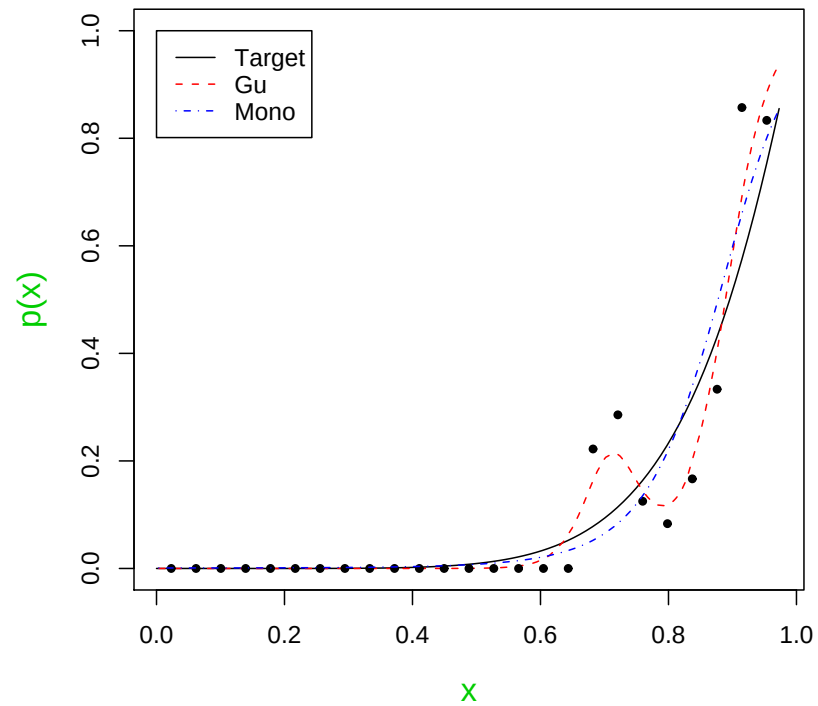
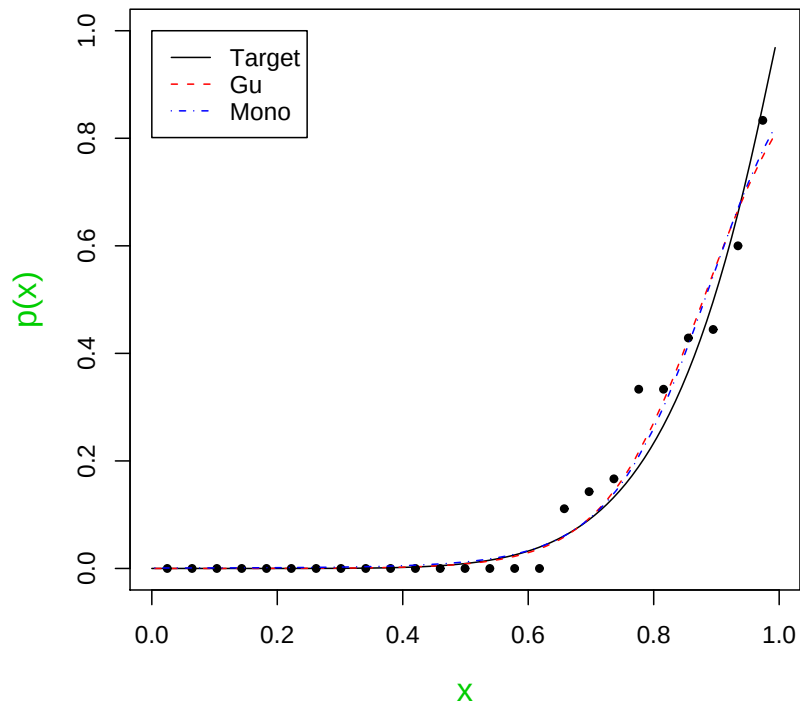
Illustration

$$\alpha = 6.28, \beta = 17.67$$



Illustration

$$\alpha = 6.9, \beta = 1.1$$



ASE: Unconstrained ($\hat{p}_g$) vs. Monotone ($\hat{p}_m$)

			Q1	Median	Mean	Q3
$\alpha = 1.98$ $\beta = 28$	$n = 50$	$\hat{p}_g$	2.3	4.9	7.0	9.3
		$\hat{p}_m$	2.3	4.9	6.8	9.1
	$n = 100$	$\hat{p}_g$	1.6	2.8	4	5.1
		$\hat{p}_m$	1.5	2.7	3.7	4.9
	$n = 200$	$\hat{p}_g$	1.0	1.7	2.2	2.8
		$\hat{p}_m$	0.83	1.5	2.0	2.6
$\alpha = 6.28$ $\beta = 17.67$	$n = 50$	$\hat{p}_g$	1.2	2.9	4.6	6.1
		$\hat{p}_m$	1.2	2.9	4.6	6.1
	$n = 100$	$\hat{p}_g$	0.71	1.7	2.7	3.6
		$\hat{p}_m$	0.7	1.6	2.5	3.4
	$n = 200$	$\hat{p}_g$	0.43	0.97	1.4	1.9
		$\hat{p}_m$	0.39	0.92	1.3	1.8
$\alpha = 6.9$ $\beta = 1.1$	$n = 50$	$\hat{p}_g$	1.4	3.1	4.8	6.3
		$\hat{p}_m$	1.8	3.2	5.6	6.9
	$n = 100$	$\hat{p}_g$	0.91	2.0	2.9	3.8
		$\hat{p}_m$	1.1	2.2	3.0	4.0
	$n = 200$	$\hat{p}_g$	0.54	1.1	1.6	2.1
		$\hat{p}_m$	0.59	1.2	1.6	1.2

($\times 10^{-3}$)

Monotone Proportion of $\hat{p}_g$

The percentage of monotone cases in 10000 repeats of $\hat{p}_g$

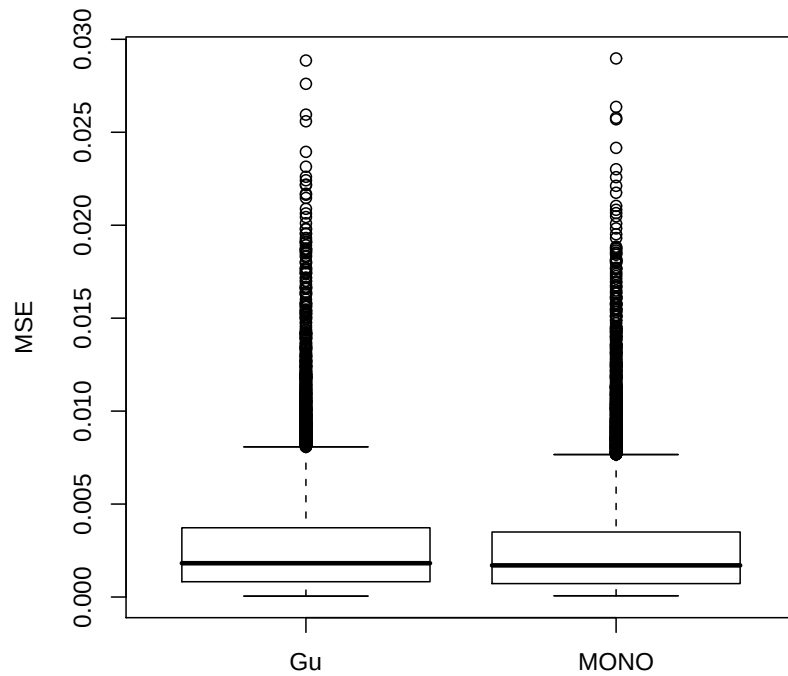
		n=50	n=100	n=200
$\hat{p}_g$	$\alpha = 1.98$ $\beta = 28$	77.89%	91.38%	96.2%
	$\alpha = 6.28$ $\beta = 17.67$	79.43%	93.46%	97.27%
	$\alpha = 6.9$ $\beta = 1.1$	85.54%	93.36%	94.67%

Given $\hat{p}_g$ monotone vs. non-monotone

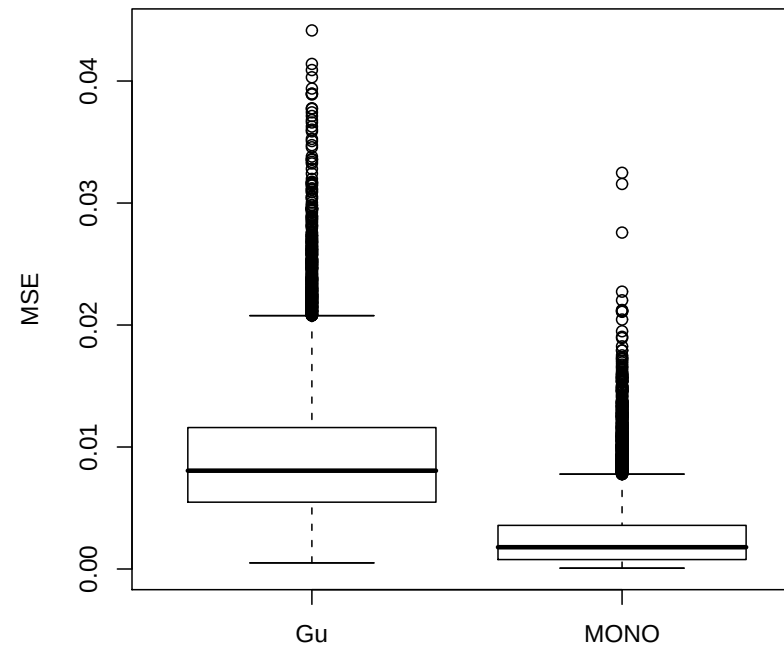
- *Given the sample size $n=100$*
- *Consider three sets of parameters: $(\alpha, \beta)=(1.98, 28)$, $(6.28, 17.67)$ and $(6.9, 1.1)$*
- *For each set of parameters (α, β) , repeat the procedure until the number of monotone $\hat{p}_g$ is equal to 10000*
- *Same for non-monotone $\hat{p}_g$*

$$p(x) = 1 - (1 - x^{1.98})^{28}$$

$\hat{p}_g$ is monotone

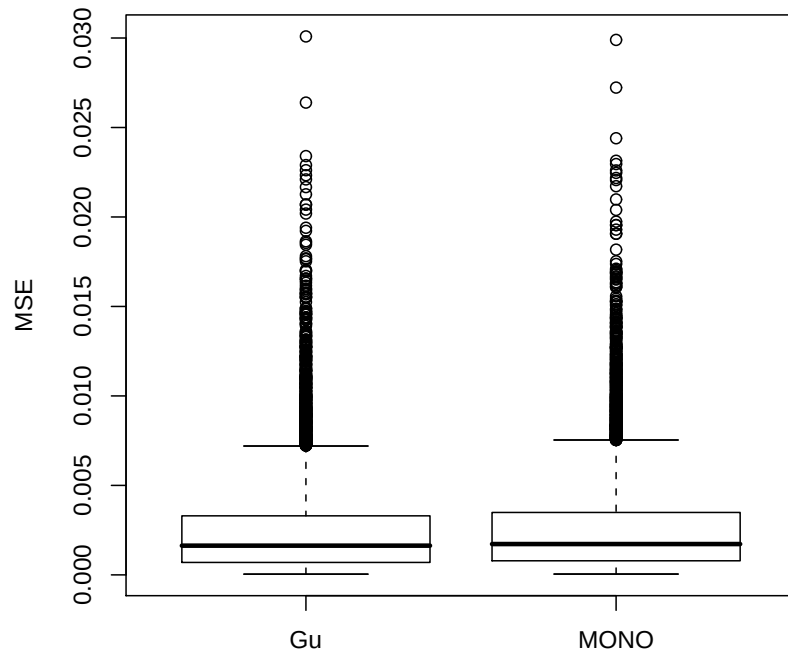


$\hat{p}_g$ is non-monotone

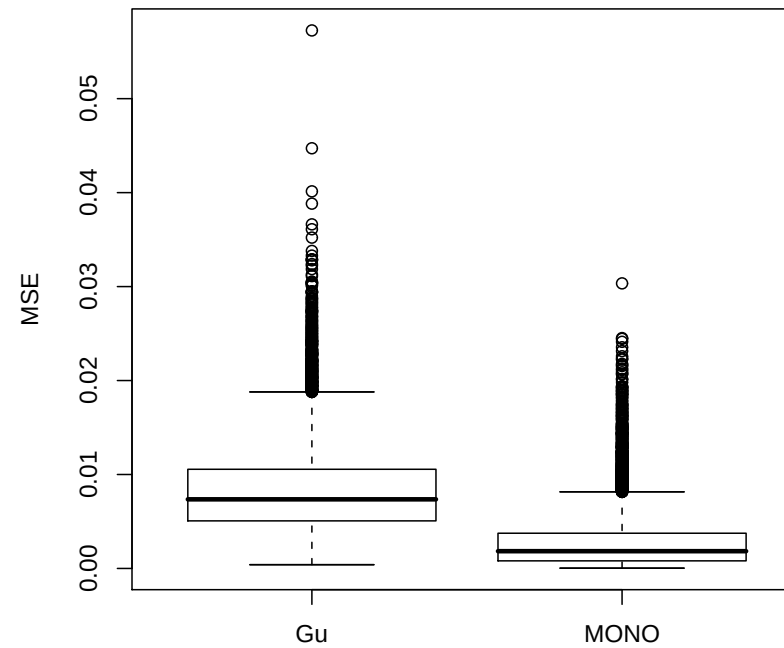


$$p(x) = 1 - (1 - x^{6.28})^{17.67}$$

$\hat{p}_g$ is monotone

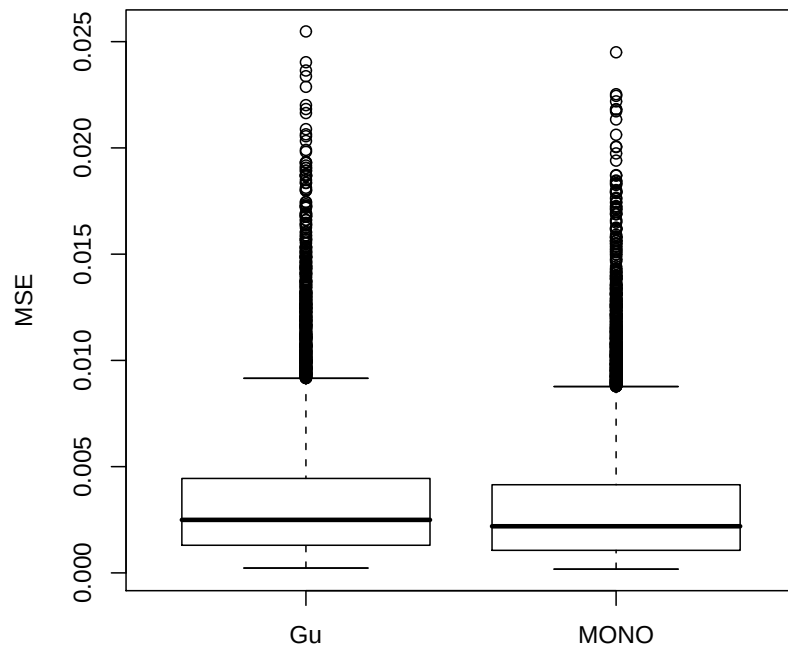


$\hat{p}_g$ is non-monotone

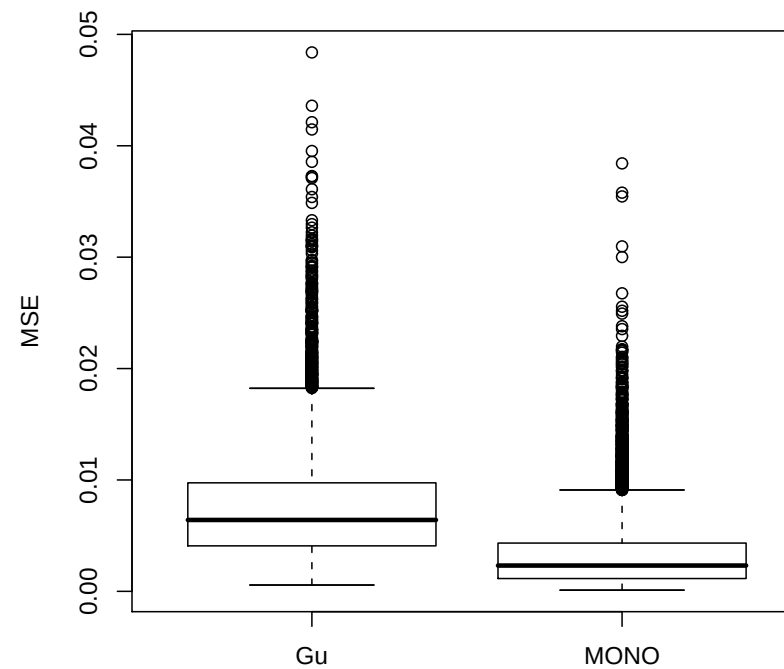


$$p(x) = 1 - (1 - x^{6.9})^{1.1}$$

$\hat{p}_g$ is monotone



$\hat{p}_g$ is non-monotone



ASEs of 10000 non-monotone $\hat{p}_g$ ($\times 10^{-3}$)

		Q1	Median	Mean	Q3
$\alpha = 1.98$ $\beta = 28$	$\hat{p}_g$	5.4	8.1	9.2	11.5
	$\hat{p}_m$	0.8	1.8	2.7	3.6
$\alpha = 6.28$ $\beta = 17.67$	$\hat{p}_g$	5.0	7.4	8.4	10.5
	$\hat{p}_m$	0.8	1.8	2.8	3.7
$\alpha = 6.9$ $\beta = 1.1$	$\hat{p}_g$	4.1	6.4	7.5	9.7
	$\hat{p}_m$	1.1	2.3	3.3	4.3

ASEs of 10000 monotone $\hat{p}_g$ ($\times 10^{-3}$)

		Q1	Median	Mean	Q3
$\alpha = 1.98$ $\beta = 28$	$\hat{p}_g$	0.8	1.8	2.8	3.7
	$\hat{p}_m$	0.7	1.7	2.6	3.5
$\alpha = 6.28$ $\beta = 17.67$	$\hat{p}_g$	0.7	1.6	2.5	3.3
	$\hat{p}_m$	0.7	1.7	2.6	3.5
$\alpha = 6.9$ $\beta = 1.1$	$\hat{p}_g$	1.3	2.5	3.4	4.4
	$\hat{p}_m$	1.1	2.2	3.1	4.1

Conclusions

- *Motivated by the WAT-EC problem, we develop a nonparametric monotone smoother based on smoothing splines for analyzing responses from exponential families.*
- *An algorithm with implementation details is provided. Computation is efficient because we utilize the characteristics of the natural cubic splines.*
- *The simulation results demonstrate that the proposed method performs well in the regression models with both the Bernoulli and Poisson responses.*
- *The proposed monotone estimator outperforms the unconstrained smoother when the latter produces non-monotone estimates, while retaining about the same performance when the latter produces monotone estimates.*

Thanks for your attention.