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

<table>
<thead>
<tr>
<th>Wafer 1 ($C_p = 0.76$)</th>
<th>Test Item1</th>
<th>Test Item2</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wafer 2 ($C_p = 0.72$)</td>
<td>0</td>
<td>1</td>
<td>...</td>
</tr>
<tr>
<td>Wafer 3 ($C_p = 0.93$)</td>
<td>1</td>
<td>1</td>
<td>...</td>
</tr>
<tr>
<td>Wafer 4 ($C_p = 0.85$)</td>
<td>0</td>
<td>1</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

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

\[ Y \sim Bernoulli(p(x)) \]

- 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**

![Graphs showing monotonic relationship between $C_p$ and $P(C_p)$]
Related Works

- **Ramsay (1998)**
  - Imposed monotonicity in function estimation
    \[ Y_i = f(x_i) + \epsilon_i, \quad i = 1, \ldots, n \]
    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

  - 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**

\[ Y_i = f(x_i) + \epsilon_i, \ i = 1, \ldots, n \]

- **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

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

where \( \lambda \) 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 $x_i$), $i = 1, \ldots, n$, 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, $\phi$ 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 $\lambda$ is a smoothing parameter.
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

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

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

  \[ l = \frac{1}{n} \sum_{i=1}^{n} \tilde{\omega}_i \{ \tilde{y}_i - \eta(a) - \int_a^{x_i} g(x)dx \}^2 + \lambda \int_a^b [g''(x)]^2 dx \]
Natural Cubic Splines  
(Green and Silverman, 1994)

- Let $g(x)$ be a natural cubic spline on $[a, b]$. Let  
  $\eta = (\eta_1, \ldots, \eta_n)^T$, $\gamma = (\gamma_2, \ldots, \gamma_{n-1})^T$  
  $g = (g_1, \ldots, g_n)^T$  
  where $\gamma_i = g''(x_i)$ for $i = 2, \ldots, n - 1$ and $\gamma_1 = \gamma_n = 0$  

- $g$ and $\gamma$ can specify a natural cubic spline if and only if  
  $R\gamma = Q^Tg$.  

- By Theorem 2.1 of Green and Silverman (1994)  
  $$\gamma = R^{-1}Q^Tg$$  
  $$\int_a^b g''(x)^2 \, dx = g^T K g$$  
  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:** $\eta = \eta(a) \cdot 1_n + M_{n,n} g$

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

- *Using the back-fitting algorithm* to solve $g$ and $\eta(a)$

- *Set initial values of* $\eta(a)$ and $g$

- *Fix* $g$, *update* $\eta(a)$ by minimizing $l(\eta(a), g)$

- *Fix* $\eta(a)$, *update* $g$ by minimizing $l(\eta(a), g)$

- *Iterate until convergence*
Estimating $\eta(a)$ for given $g$

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

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

iteratively until convergence.
Estimating $g$ for given $\eta(a)$

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

$$
\begin{align*}
    l &= g^T(M^T\tilde{W}M + n\lambda K)g - 2\tilde{y}^T\tilde{W}Mg + \tilde{y}^T\tilde{W}\tilde{y} \\
    \text{where} \\
    \tilde{W} &= \text{diagonal}\{\tilde{\omega}_1, \ldots, \tilde{\omega}_n\} \\
    \tilde{y} &= (\tilde{y}_1 - \eta(a), \ldots, \tilde{y}_n - \eta(a))^T
\end{align*}
$$

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

$$
\hat{g} = (M^T\tilde{W}M + n\lambda K)^{-1}M^T\tilde{W}\tilde{y}
$$
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 $\eta$
  
  $$\hat{\eta} = \hat{\eta}(a) \cdot 1_n + M\hat{g}_+$$

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

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

  where $\tau_j$ 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**

\[ \hat{\eta}_{mon} = \hat{\eta}_a \cdot 1_n + SN\tilde{M}\bar{g} \]

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

\[ \tilde{M} = \begin{pmatrix} 1 & 0^T_n \\ 1_n & M \end{pmatrix}; \]

\( N \) 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 \( S = (s_1, s_2, \ldots, s_n) \) be the \( n \times n \) matrix given by

\[ s_i = (0^T_{i-1}, \tau_i \cdot 1^T_{n-i+1})^T \text{ for } i = 1, \ldots, n. \]
Matrix Form (continued)

- The effect of \( \tilde{M} \) and \( N \)

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

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

\[ \hat{g}(x) \]
Illustration

$\tilde{g}(x)$
Illustration

\[ \tilde{g}(x) \]

\[ \hat{\eta}(x_1) \]
Illustration

\[ \tilde{g}(x) \]

\[ \hat{\eta}(x_2) \]
\[ \tilde{g}(x)^+ \]

\[ \hat{\eta}(x_i)^{\text{mon}} \]
Illustration

\[ \hat{g}(x)^+ \]

\[ \hat{\eta}(x)^{\text{mon}} \]
Algorithm

- **Step 1** Calculate matrices $C, D, Q, R,$ and $K$

- **Step 2** Set initial values $\hat{g}(0), \hat{\eta}(0)(a),$ and tolerance level $T$. Let $k = 0$.

- **Step 3** Fix $\hat{g}^{(k)}$, update $\hat{\eta}(a)$ to $\hat{\eta}^{(k+1)}(a)$ by Newton-Raphson method: compute

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

until convergence

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

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

- **Step 5** If $\left| \frac{\hat{\eta}^{(k+1)}(a) - \hat{\eta}^{(k)}(a)}{\hat{\eta}^{(k)}(a)} \right| < T$, stop iteration, set $\hat{g}_{+}^{(k+1)} = \max\{\hat{g}^{(k+1)}, 0\}$, and output

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

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

- Generate
  \[ y_i \sim \text{Bernoulli}(1, p(x_i)), \quad x_i \sim U(0, 1) \]
  for \( i = 1, \ldots, 200 \).

- Use the function
  \[ p(x) = 1 - (1 - x^{4.5})^{2.5}, \quad x \in (0, 1) \]

- \( \lambda = 5 \times 10^{-5} \) by the GCV method.
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 $i = 1, \ldots, 200$.

- Use the mean functions
  - $\lambda(x) = \log(x^2 + 1)$ for $1 \leq x \leq 3$

- Choose $\lambda_{GCV}$ and $\lambda_{\text{opt}}$ (optimal in the sense of the averaged squared error (ASE))

- $\text{ASE}(\hat{\lambda}(x)) = \frac{1}{n} \sum_{i=1}^{n} (\hat{\lambda}(x_i) - \lambda(x))^2$

- Compare ASE of $\lambda_{GCV}$ and $\lambda_{\text{opt}}$
Dash-dot line is with $\lambda_{opt}$, $ASE(\lambda_{opt}) = 0.01190$.

Dot line is with $\lambda_{GCV}$, $ASE(\lambda_{GCV}) = 0.01189$
Constrained vs. Unconstrained

- **Simulation:**
  - Generate $y_i \sim \text{Bernoulli}(1, p(x_i))$, $x_i \sim U(0, 1)$ for $i = 1, \ldots, n$.
  - 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

$\alpha = 1.98, \beta = 28$
$\alpha = 6.28, \beta = 17.67$
\[ \alpha = 6.9, \beta = 1.1 \]
### ASE: Unconstrained ($\hat{p}_g$) vs. Monotone ($\hat{p}_m$)

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Median</th>
<th>Mean</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 50$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 1.98$</td>
<td>2.3</td>
<td>4.9</td>
<td>7.0</td>
<td>9.3</td>
</tr>
<tr>
<td>$\beta = 28$</td>
<td>2.3</td>
<td>4.9</td>
<td>6.8</td>
<td>9.1</td>
</tr>
<tr>
<td>$n = 100$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.28$</td>
<td>1.6</td>
<td>2.8</td>
<td>4.0</td>
<td>5.1</td>
</tr>
<tr>
<td>$\beta = 17.67$</td>
<td>1.5</td>
<td>2.7</td>
<td>3.7</td>
<td>4.9</td>
</tr>
<tr>
<td>$n = 200$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.9$</td>
<td>1.0</td>
<td>1.7</td>
<td>2.2</td>
<td>2.8</td>
</tr>
<tr>
<td>$\beta = 1.1$</td>
<td>0.83</td>
<td>1.5</td>
<td>2.0</td>
<td>2.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Median</th>
<th>Mean</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 100$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 1.98$</td>
<td>1.2</td>
<td>2.9</td>
<td>4.6</td>
<td>6.1</td>
</tr>
<tr>
<td>$\beta = 28$</td>
<td>1.2</td>
<td>2.9</td>
<td>4.6</td>
<td>6.1</td>
</tr>
<tr>
<td>$n = 100$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.28$</td>
<td>0.71</td>
<td>1.7</td>
<td>2.7</td>
<td>3.6</td>
</tr>
<tr>
<td>$\beta = 17.67$</td>
<td>0.7</td>
<td>1.6</td>
<td>2.5</td>
<td>3.4</td>
</tr>
<tr>
<td>$n = 200$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.9$</td>
<td>0.43</td>
<td>0.97</td>
<td>1.4</td>
<td>1.9</td>
</tr>
<tr>
<td>$\beta = 1.1$</td>
<td>0.39</td>
<td>0.92</td>
<td>1.3</td>
<td>1.8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Median</th>
<th>Mean</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n = 200$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 1.98$</td>
<td>1.4</td>
<td>3.1</td>
<td>4.8</td>
<td>6.3</td>
</tr>
<tr>
<td>$\beta = 28$</td>
<td>1.8</td>
<td>3.2</td>
<td>5.6</td>
<td>6.9</td>
</tr>
<tr>
<td>$n = 100$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.28$</td>
<td>0.91</td>
<td>2.0</td>
<td>2.9</td>
<td>3.8</td>
</tr>
<tr>
<td>$\beta = 17.67$</td>
<td>1.1</td>
<td>2.2</td>
<td>3.0</td>
<td>4.0</td>
</tr>
<tr>
<td>$n = 200$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\alpha = 6.9$</td>
<td>0.54</td>
<td>1.1</td>
<td>1.6</td>
<td>2.1</td>
</tr>
<tr>
<td>$\beta = 1.1$</td>
<td>0.59</td>
<td>1.2</td>
<td>1.6</td>
<td>1.2</td>
</tr>
</tbody>
</table>

($\times 10^{-3}$)
## Monotone Proportion of $\hat{p}_g$

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

<table>
<thead>
<tr>
<th>$\hat{p}_g$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>n=50</th>
<th>n=100</th>
<th>n=200</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.98</td>
<td>28</td>
<td>77.89%</td>
<td>91.38%</td>
<td>96.2%</td>
</tr>
<tr>
<td></td>
<td>6.28</td>
<td>17.67</td>
<td>79.43%</td>
<td>93.46%</td>
<td>97.27%</td>
</tr>
<tr>
<td></td>
<td>6.9</td>
<td>1.1</td>
<td>85.54%</td>
<td>93.36%</td>
<td>94.67%</td>
</tr>
</tbody>
</table>
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 \((\alpha, \beta)\), 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
$$p(x) = 1 - (1 - x^{6.28})^{17.67}$$

\[ \hat{p}_g \text{ is monotone} \quad \hat{p}_g \text{ is non-monotone} \]
\[ p(x) = 1 - (1 - x^{6.9})^{1.1} \]

\( \hat{p}_g \) is monotone

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

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Median</th>
<th>Mean</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha = 1.98$</td>
<td>$\hat{p}_g$</td>
<td>5.4</td>
<td>8.1</td>
<td>9.2</td>
</tr>
<tr>
<td>$\beta = 28$</td>
<td>$\hat{p}_m$</td>
<td>0.8</td>
<td>1.8</td>
<td>2.7</td>
</tr>
<tr>
<td>$\alpha = 6.28$</td>
<td>$\hat{p}_g$</td>
<td>5.0</td>
<td>7.4</td>
<td>8.4</td>
</tr>
<tr>
<td>$\beta = 17.67$</td>
<td>$\hat{p}_m$</td>
<td>0.8</td>
<td>1.8</td>
<td>2.8</td>
</tr>
<tr>
<td>$\alpha = 6.9$</td>
<td>$\hat{p}_g$</td>
<td>4.1</td>
<td>6.4</td>
<td>7.5</td>
</tr>
<tr>
<td>$\beta = 1.1$</td>
<td>$\hat{p}_m$</td>
<td>1.1</td>
<td>2.3</td>
<td>3.3</td>
</tr>
</tbody>
</table>
ASEs of 10000 monotone $\hat{p}_g \times 10^{-3}$

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>$\hat{p}_g$</th>
<th>Q1</th>
<th>Median</th>
<th>Mean</th>
<th>Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.98</td>
<td>28</td>
<td>0.8</td>
<td>1.8</td>
<td>2.8</td>
<td>3.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\hat{p}_m$</td>
<td>0.7</td>
<td>1.7</td>
<td>2.6</td>
<td>3.5</td>
</tr>
<tr>
<td>6.28</td>
<td>17.67</td>
<td>0.7</td>
<td>1.6</td>
<td>2.5</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\hat{p}_m$</td>
<td>0.7</td>
<td>1.7</td>
<td>2.6</td>
<td>3.5</td>
</tr>
<tr>
<td>6.9</td>
<td>1.1</td>
<td>1.3</td>
<td>2.5</td>
<td>3.4</td>
<td>4.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\hat{p}_m$</td>
<td>1.1</td>
<td>2.2</td>
<td>3.1</td>
<td>4.1</td>
</tr>
</tbody>
</table>
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.