Forward Selection and Estimation in high dimensional single index models

A brief explanation

Ji Won Min Henry Grosse

ETH Zürich

Statistics Seminar - November 21, 2022

Table of Contents

Table of Contents

Recall : Single Index Model

Suppose that we have n i.i.d. observations (X_i, Y_i). A single index model is of the following form :

$$Y = g(X^\intercal \beta) + \epsilon \tag{1}$$

where

- $X \in \mathbb{R}^{pxn}$ is the predictors and $Y \in \mathbb{R}$ is the response
- $\mathbb{E}[\epsilon|X] = 0$ and $\mathbb{E}[\epsilon^2|X] < \infty$
- g is an unknown real-valued link function

Recall : High Dimensional Setting

In high dimensional data, the number of predictors p is relatively large, possibly larger than the sample size n.

The presence of too many variables may cause overfitting leading to poor prediction.
$\rightarrow$ It is crucial to select relevant variables from the predictive model.

We will combine both forward selection and penalization for variable selection and estimation for a high dimensional single index model with an unknown monotone increasing smooth link function under sparsity condition.

Problem Setting

We consider a high dimensional single index model with an unknown monotone increasing smooth link function g

$$Y|\mathbf{X} \sim P(\cdot, g(\mathbf{X}^\intercal \beta)) \tag{2}$$

where :

- $P(,\theta)$ is a stochastically increasing family with scalar parameter θ,
- g is an unknown, strictly increasing smooth link function, and
- $\|\beta\| = 1$

Special case : $Y = g(\mathbf{X}^\intercal \beta) + \epsilon$ where ϵ is a random variable with $\mathbb{E}[\epsilon] = 0$ and $Var(\epsilon) < +\infty$

Table of Contents

Idea

Observed data : $(\mathbf{X}_i, Y_i) \in (\mathcal{X}, \mathbb{R})$, $i = 1, 2, \cdots, n$, i.i.d.
Idea : Maximize the monotone association between Y and $\mathbf{X}^\intercal \beta$ under an ℓ_1-penalty

We will reduce high dimensional optimization problems to several one dimensional problems.

A natural estimator of β is proposed by maximizing the following Kendall's tau coefficient between Y and $\mathbf{X}^\top\beta$:

$$\tau_n(\beta) = \frac{1}{n(n-1)} \sum_{1 \le i_1 \neq i_2 \le n} \text{sign}(Y_{i_2} - Y_{i_1})\text{sign}(\boldsymbol{X}_{i_2}^\top \beta - \boldsymbol{X}_{i_1}^\top \beta) \qquad (3)$$

A natural estimator of β is proposed by maximizing the following Kendall's tau coefficient between Y and $\mathbf{X}^\top\beta$:

$$\tau_n(\beta) = \frac{1}{n(n-1)} \sum_{1 \le i_1 \neq i_2 \le n} \text{sign}(Y_{i_2} - Y_{i_1})\text{sign}(\boldsymbol{X}_{i_2}^\top \beta - \boldsymbol{X}_{i_1}^\top \beta) \qquad (3)$$

Since $\tau_n(\beta)$ is a step function and not continuous in β, we use a smoothed version of $\tau_n(\beta)$ given by :

$$\tau_n^\star(\beta) = \frac{1}{n(n-1)} \sum_{1 \le i_1 \neq i_2 \le n} \text{sign}(Y_{i_2} - Y_{i_1})\tanh\left(\frac{\boldsymbol{X}_{i_2}^\top \beta - \boldsymbol{X}_{i_1}^\top \beta}{h}\right) \qquad (4)$$

Algorithm

The following algorithm is called the **Smoothed Kendall's Iterative Maximizer Model Selector (SKIMMS)**.

For a given $\lambda > 0$ and $\epsilon > 0$:

Step 1. Maximize T_j over j where

$$T_j := \frac{1}{n(n-1)} \sum_{1 \le i_1 \neq i_2 \le n} \text{sign}(Y_{i_2} - Y_{i_1})\text{sign}(\mathbf{X}_{i_2,j} - \mathbf{X}_{i_1,j}) = \tau_n(\mathbf{e}_j) \quad (5)$$

and set $\hat{\beta}^{(1)} = \text{sign}(T_{j_{\max}})\mathbf{e}_{j_{\max}}$, where $\boldsymbol{e_j}$ is a unit vector with j-th entry 1.

Step 2. Suppose that $X_{j_1}, \cdots, X_{j_{k-1}}$ are already selected and current single index coefficient is $\hat{\beta}^{(k-1)}$. First, for all $j \notin \{j_1, \cdots, j_{k-1}\}$ compute

$$\hat{\beta}_j = \arg\max_{b \in \mathbb{R}} \left\{ \tau_n^\star \left(\hat{\beta}^{(k-1)} + b \cdot \mathbf{e}_j \right) - \lambda |b| \right\}$$

Step 2. Suppose that $X_{j_1}, \cdots, X_{j_{k-1}}$ are already selected and current single index coefficient is $\hat{\boldsymbol{\beta}}^{(k-1)}$. First, for all $j \notin \{j_1, \cdots, j_{k-1}\}$ compute

$$\hat{\beta}_j = \arg\max_{b \in \mathbb{R}} \left\{ \tau_n^\star \left(\hat{\beta}^{(k-1)} + b \cdot \mathbf{e}_j \right) - \lambda |b| \right\}$$

Then, let

$$j_k = \arg\max \left\{ \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j \right) : j \notin \{j_1, \cdots, j_{k-1}\} \right\}$$

Step 2. Suppose that $X_{j_1}, \cdots, X_{j_{k-1}}$ are already selected and current single index coefficient is $\hat{\boldsymbol{\beta}}^{(k-1)}$. First, for all $j \notin \{j_1, \cdots, j_{k-1}\}$ compute

$$\hat{\beta}_j = \arg\max_{b \in \mathbb{R}} \left\{ \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + b \cdot \mathbf{e}_j \right) - \lambda |b| \right\}$$

Then, let

$$j_k = \arg\max \left\{ \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j \right) : j \notin \{j_1, \cdots, j_{k-1}\} \right\}$$

If $\tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_{j_k} \mathbf{e}_{j_k} \right) - \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} \right) < \epsilon$, set $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}^{(k-1)}$ and STOP.

Step 2. Suppose that $X_{j_1}, \cdots, X_{j_{k-1}}$ are already selected and current single index coefficient is $\hat{\boldsymbol{\beta}}^{(k-1)}$. First, for all $j \notin \{j_1, \cdots, j_{k-1}\}$ compute

$$\hat{\beta}_j = \arg\max_{b \in \mathbb{R}} \left\{ \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + b \cdot \mathbf{e}_j \right) - \lambda |b| \right\}$$

Then, let

$$j_k = \arg\max \left\{ \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j \right) : j \notin \{j_1, \cdots, j_{k-1}\} \right\}$$

If $\tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_{j_k} \mathbf{e}_{j_k} \right) - \tau_n^\star \left(\hat{\boldsymbol{\beta}}^{(k-1)} \right) < \epsilon$, set $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}^{(k-1)}$ and STOP. Otherwise we set

$$\hat{\boldsymbol{\beta}}^{(k)} = \frac{\hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_{j_k} \mathbf{e}_{j_k}}{\left\| \hat{\boldsymbol{\beta}}^{(k-1)} + \hat{\beta}_{j_k} \mathbf{e}_{j_k} \right\|_2}$$

and repeat *Step 2*.

Algorithm

Step 3. Finally re-estimate β by maximizing $\tau_n^\star(\beta)$ over the selected variables from *Step 2*.

Finally we apply a simple isotonic regression method to estimate the link function $g(t)$...

Estimating the link function

1. Compute $Z_i = X_i^\top \hat{\beta}$, and sort $\{Z_1, \cdots, Z_n\}$ in increasing order to get $\{Z_{(1)}, \cdots, Z_{(n)}\}$. Sort the Y_i's in the corresponding order to get $\{Y_{(1)}, \cdots, Y_{(n)}\}$[1]
2. Run a pool-adjacent-violators (PAV) algorithm on the $Y_{(i)}$'s to get $\{Y^{(1)}, \cdots, Y^{(n)}\}$.

[1] $\{Y_{(1)}, \cdots, Y_{(n)}\}$ may not be in increasing order

Estimating the link function

1. Compute $Z_i = X_i^\top \hat{\beta}$, and sort $\{Z_1, \cdots, Z_n\}$ in increasing order to get $\{Z_{(1)}, \cdots, Z_{(n)}\}$. Sort the Y_i's in the corresponding order to get $\{Y_{(1)}, \cdots, Y_{(n)}\}$[1]
2. Run a pool-adjacent-violators (PAV) algorithm on the $Y_{(i)}$'s to get $\{Y^{(1)}, \cdots, Y^{(n)}\}$.
3. Link function estimate:

$$\hat{g}(t) = \frac{\sum_{j=1}^{n} K\left(\frac{t - Z_{(j)}}{b}\right) \cdot Y^{(j)}}{\sum_{j=1}^{n} K\left(\frac{t - Z_{(j)}}{b}\right)}$$

where $K(t)$ is a smooth, symmetric kernel function and b is chosen by GCV.

[1] $\{Y_{(1)}, \cdots, Y_{(n)}\}$ may not be in increasing order

Tuning parameters

An important parameter in the procedure above is λ. It is tuned using 5-fold cross validation.

A set of possible values of λ are chosen over a grid and $\tau_n(\beta)$ is computed on $1/5$ of the dataset using the estimate of β from the rest of the data.

λ that maximizes the 5-fold average of $\tau_n(\beta)$ is chosen.

Table of Contents

Simulation Studies

Paper Examples:

Example 1

$$Y_i = \arctan(2X_i^\top \beta) + 0.2\epsilon_i \ , \ \ i = 1, \cdots, n$$

$$\beta = (3, 5, 3, 5, 0, \cdots, 0)$$

Example 2

$$Y_i = \exp(X_i^\top \beta) + 0.5\epsilon_i \ , \ \ i = 1, \cdots, n$$

$$\beta = (3, 3, -4, -4, 5, 5, 0 \cdots, 0)$$

Example 3

$$Y_i = \exp(\arctan(2X_i^\top \beta)) + 0.2\epsilon_i \ , \ \ i = 1, \cdots, n$$

$$\beta = (3, 3, 3, -4, -4, -4, 5, 5, 5, 0, \cdots, 0)$$

Simulation Studies

- X_i is a p-dimensional predictor generated from a multivariate normal distribution $N(0, \Sigma)$,
- with $\sigma_{ii} = 1$ for $i = 1, \cdots, n$ and $\sigma_{ij} = \rho$ for $i \neq j$
- The noise ϵ_i is independent of the predictors, and is generated from a standard normal distribution "with 10% of the outliers following the Cauchy distribution".

Paper Findings (Example 1)

		ρ	Method	FP	FN	PE
Example 1	$n = 100$	0.25	SKIMMS	4.10(2.23)	0(0)	0.06(0.01)
			LASSO	8.31(4.39)	0(0)	0.21(0.02)
		0.5	SKIMMS	3.10(2.02)	0.02(0.14)	0.06(0.02)
			LASSO	9.23(3.99)	0.09(0.32)	0.25(0.02)
	$n = 200$	0.25	SKIMMS	1.44(1.07)	0(0)	0.04(0.00)
			LASSO	7.98(4.12)	0(0)	0.18(0.01)
			SSIR	8.56(3.57)	2.75(1.28)	–
		0.50	SKIMMS	1.22(1.11)	0(0)	0.04(0.00)
			LASSO	8.98(3.31)	0(0)	0.23(0.01)
			SSIR	8.72(3.50)	2.97(1.21)	–

Paper Findings (Example 2)

		ρ	Method	FP	FN	PE
Example 2	$n = 100$	0.25	SKIMMS	9.16(3.71)	0.14(0.35)	1.25(0.60)
			LASSO	5.4(6.01)	1.51(1.92)	2.52(0.87)
		0.5	SKIMMS	7.39(3.12)	0.7(0.83)	1.12(0.51)
			LASSO	5.1(5.64)	2.31(1.95)	1.93(0.63)
	$n = 200$	0.25	SKIMMS	7.1(3.29)	0(0)	0.66(0.24)
			LASSO	3.92(4.05)	0.69(1.62)	2.16(0.87)
			SSIR	5.44(2.24)	4.74(1.31)	–
		0.50	SKIMMS	4.73(2.13)	0.04(0.20)	0.51(0.15)
			LASSO	4.59(5.93)	1.1(1.53)	1.65(0.52)
			SSIR	6.10(3.02)	4.21(1.39)	–

Paper Findings (Example 3)

		ρ	Method	FP	FN	PE
Example 3	$n = 100$	0.25	SKIMMS	6.64(3.06)	0.33(0.89)	0.15(0.18)
			LASSO	18.7(7.83)	0.03(0.17)	0.37(0.05)
		0.5	SKIMMS	5.97(2.72)	1.03(1.11)	0.20(0.16)
			LASSO	17.11(8.54)	0.35(0.86)	0.38(0.07)
	$n = 200$	0.25	SKIMMS	4.52(1.57)	0(0)	0.05(0.01)
			LASSO	18.73(8.37)	0(0)	0.28(0.02)
			SSIR	5.53(2.75)	7.57(1.47)	–
		0.50	SKIMMS	4.33(1.21)	0.14(0.35)	0.06(0.03)
			LASSO	18.67(7.07)	0(0)	0.28(0.02)
			SSIR	6.91(2.79)	6.89(1.60)	–

Visualization

SKIMMS:

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:

Visualization

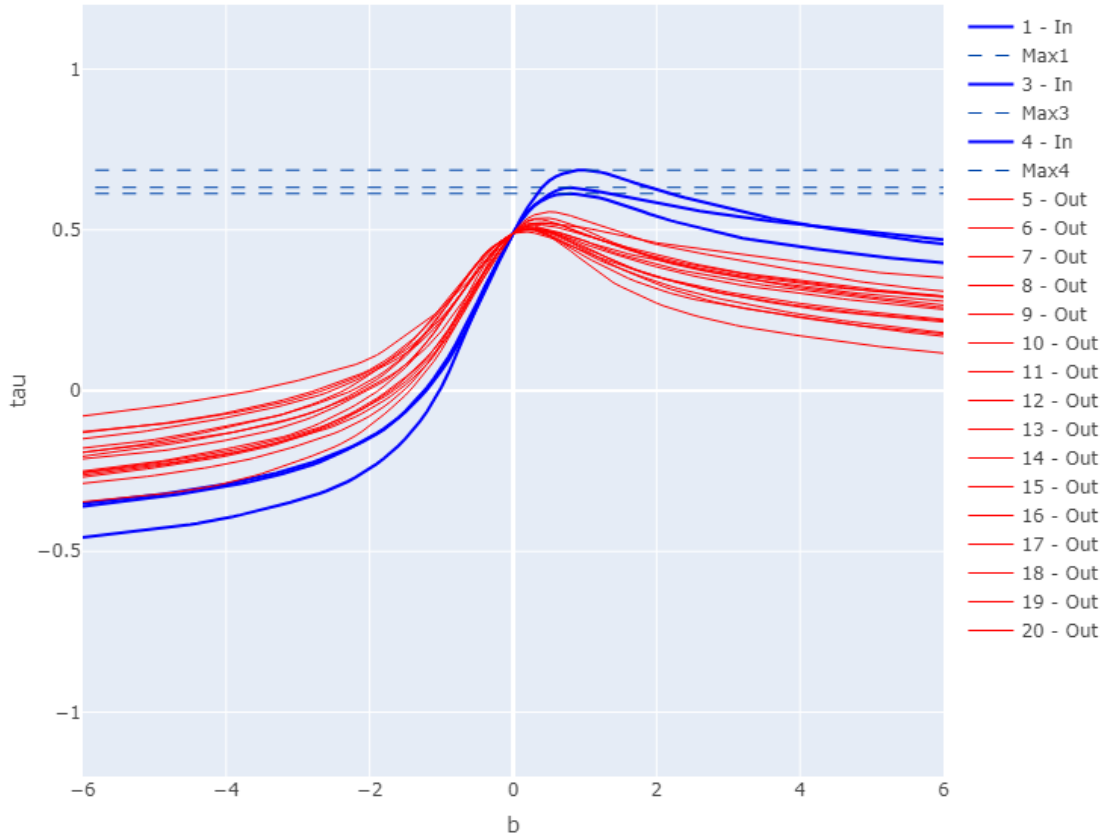
SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet

Visualization

SKIMMS:

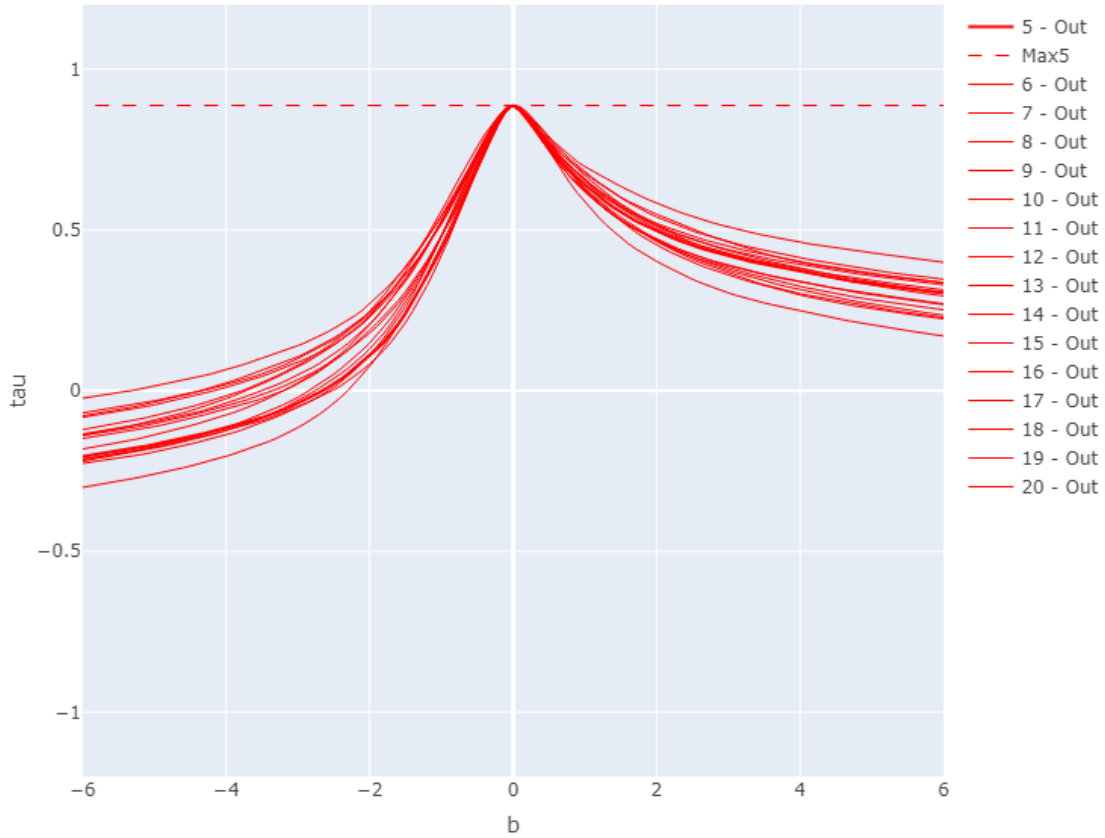
- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet



Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet



Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$
  $\implies$ R figure!

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$

Link Estimation:

- **Step 1.** Run PAVA on $\{Y_{(1)}, \cdots, Y_{(n)}\} \implies \{Y^{(1)}, \ldots, Y^{(n)}\}$

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$

Link Estimation:

- **Step 1.** Run PAVA on $\{Y_{(1)}, \cdots, Y_{(n)}\} \implies \{Y^{(1)}, \ldots, Y^{(n)}\}$
- **Step 2.** Compute $\hat{g}_b(t)$ from Kernel estimator

ETH

Visualization

SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$

Link Estimation:

- **Step 1.** Run PAVA on $\{Y_{(1)}, \cdots, Y_{(n)}\} \implies \{Y^{(1)}, \ldots, Y^{(n)}\}$
- **Step 2.** Compute $\hat{g}_b(t)$ from Kernel estimator
  $\implies$ R figure!

Visualization

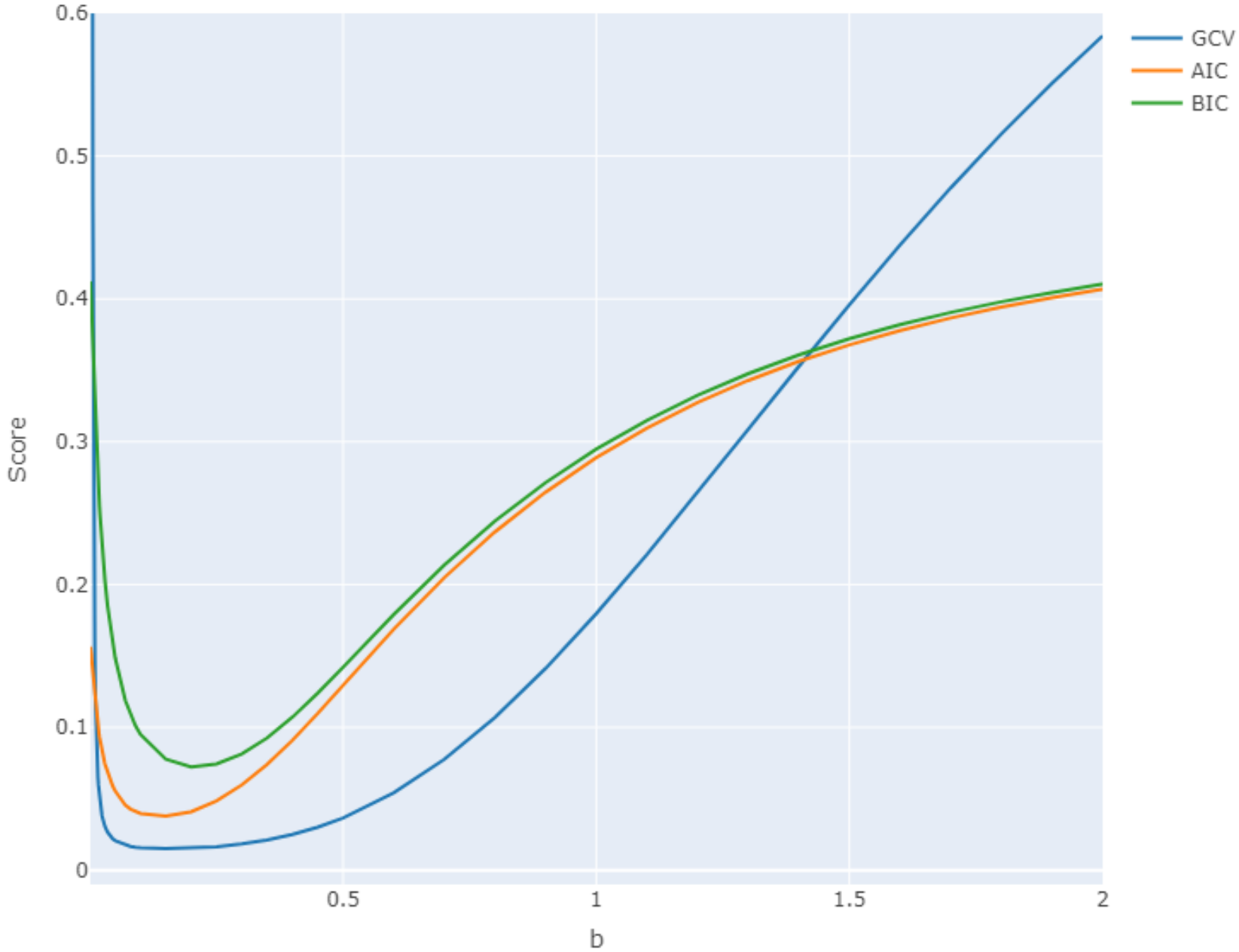
SKIMMS:

- **Step 1.** Find first variable to select and initialize β
- **Step 2.** Find new variable to select:
  - Compute $\hat{\beta}_j$ for all j not selected yet
  - Find j maximising $\tau_n^\star(\hat{\beta}^{(k-1)} + \hat{\beta}_j \mathbf{e}_j)$
  - If the increase in $\tau_n^\star$ is smaller than $\epsilon \implies$ stop!
  - Otherwise update $\hat{\beta}^{(k)}$ and repeat **Step 2.**
- **Step 3.** Re-estimate $\hat{\beta}$ by maximising $\tau_n^\star$

Link Estimation:

- **Step 1.** Run PAVA on $\{Y_{(1)}, \cdots, Y_{(n)}\} \implies \{Y^{(1)}, \cdots, Y^{(n)}\}$
- **Step 2.** Compute $\hat{g}_b(t)$ from Kernel estimator
- **Step 3.** Take $\hat{g}(t)$ by optimizing GCV over $b > 0$

ETH


0.6
0.5
0.4
0.3
0.2
0.1
0
Score
0.5
1
1.5
2
b
GCV
AIC
BIC

Simulation Studies

Our Examples:[2]

Example 4

$$g(x) = x^3 \ , \ \beta = (2, -4, -4, 2, 3, 3, 0, \cdots, 0)$$

Example 5

$$g(x) = \sqrt[3]{x} \ , \ \beta = (2, -4, -4, 2, 3, 3, 0, \cdots, 0)$$

Example 6

$$g(x) = \sin(10x)/10 + x \ , \ \beta = (3, -4, -4, 5, 0 \cdots, 0)$$

Example 7

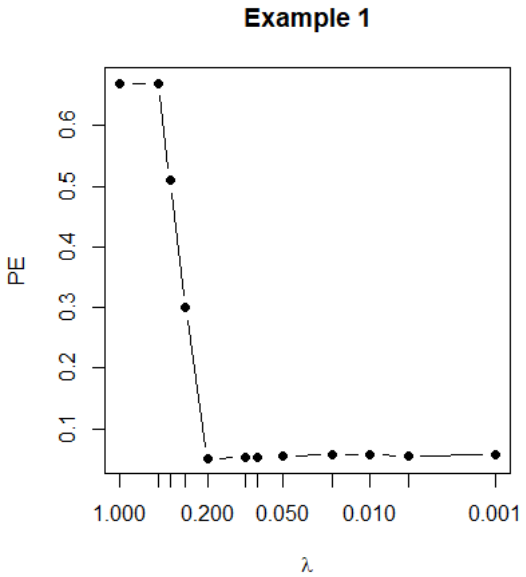
$$g(x) = 2(x + 0.5)^2 - 1 \ , \ \beta = (3, -4, -4, 5, 0, \cdots, 0)$$

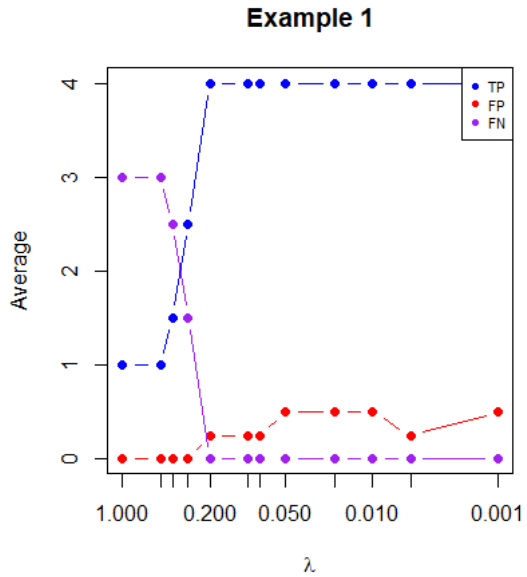
Example 8

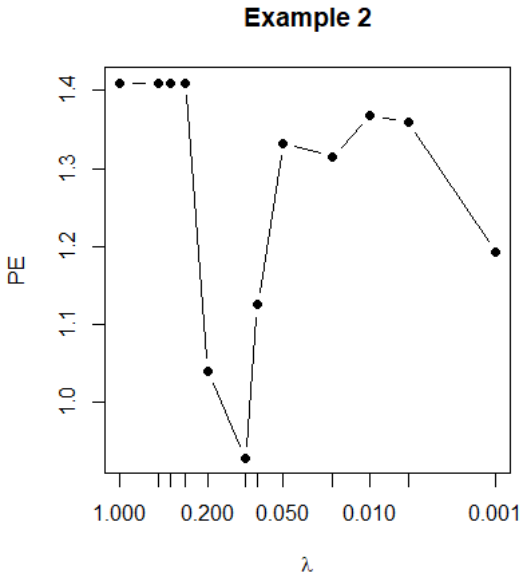
$$g = \text{piecewise-linear / continuous} \ , \ \beta = (3, -4, -4, 5, 0, \cdots, 0)$$

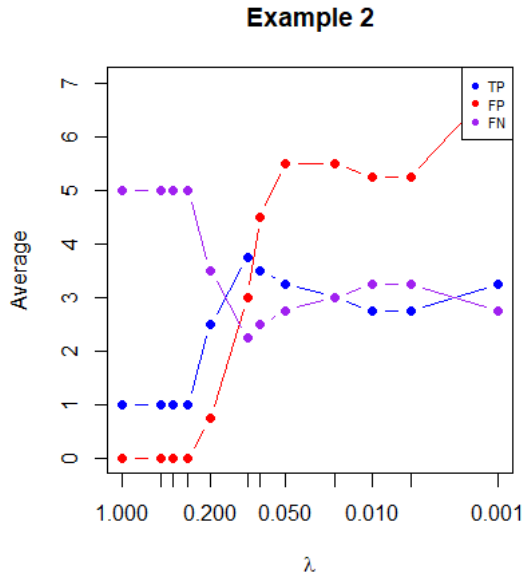
[2] up to a constant

Our Findings

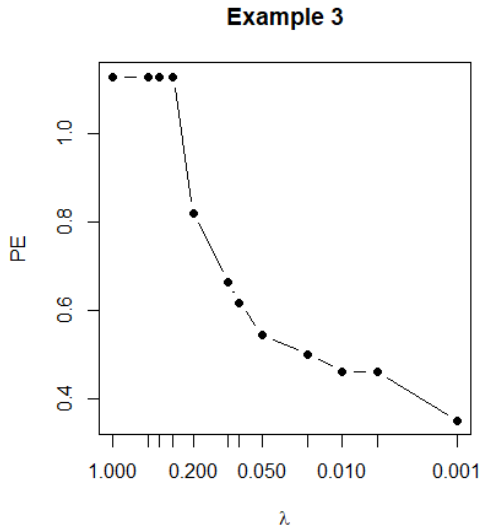


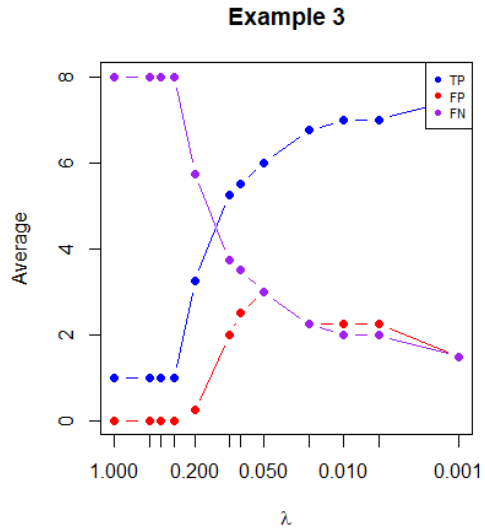


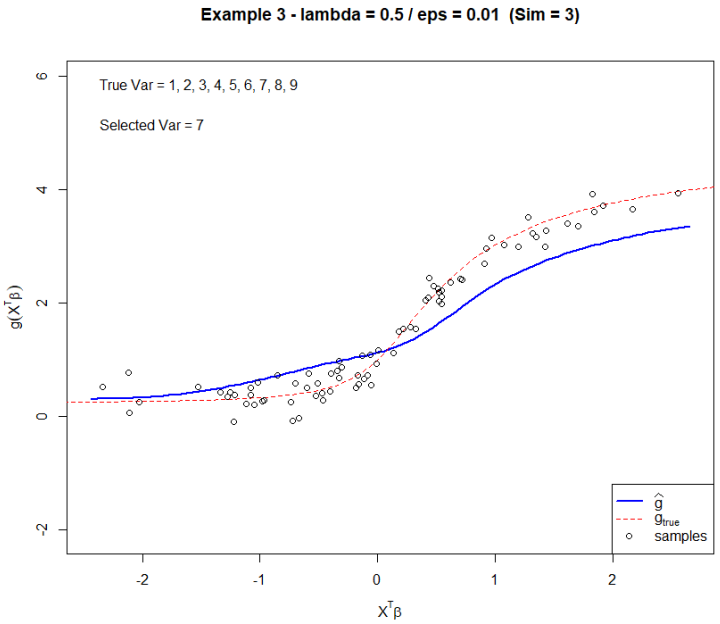




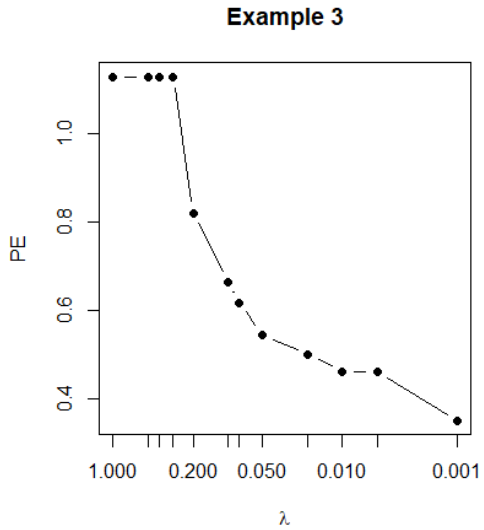
Our Findings

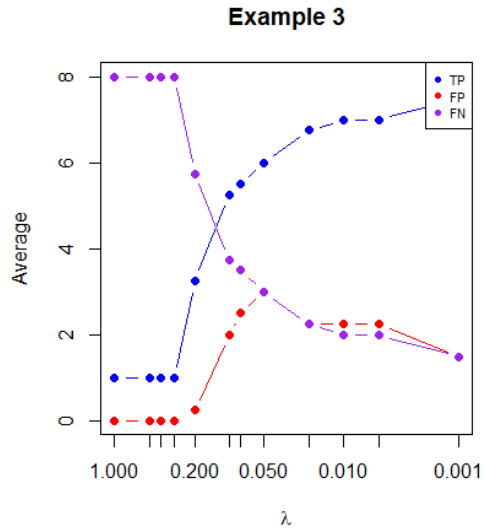


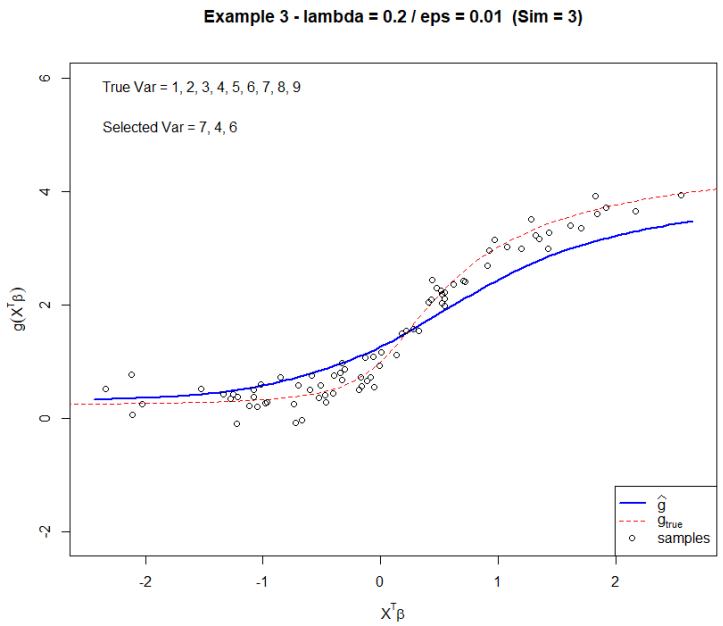




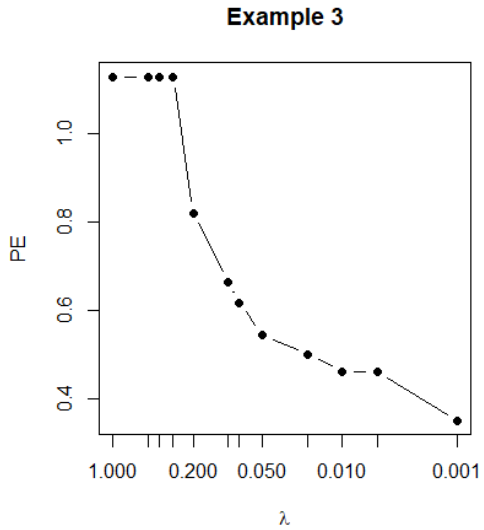
Our Findings

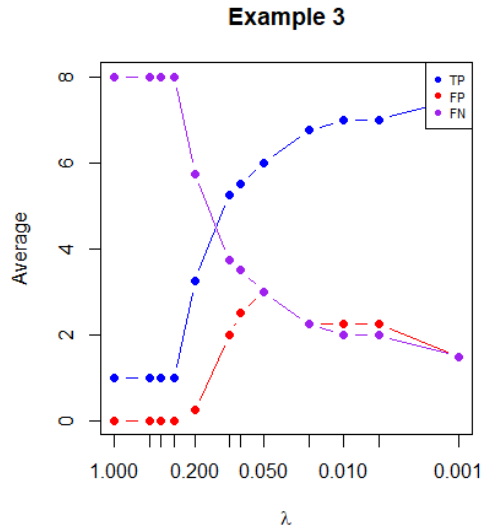


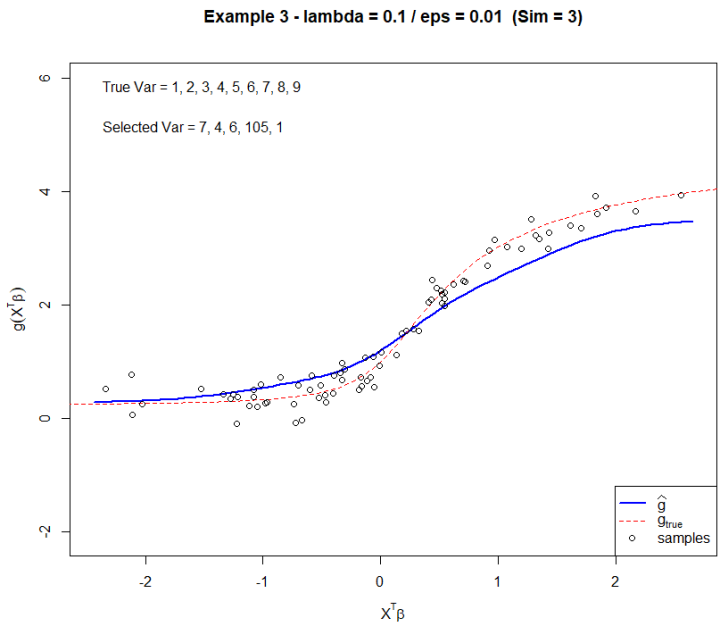




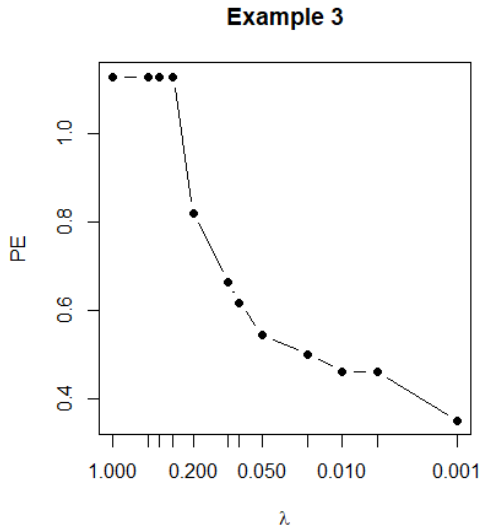
Our Findings

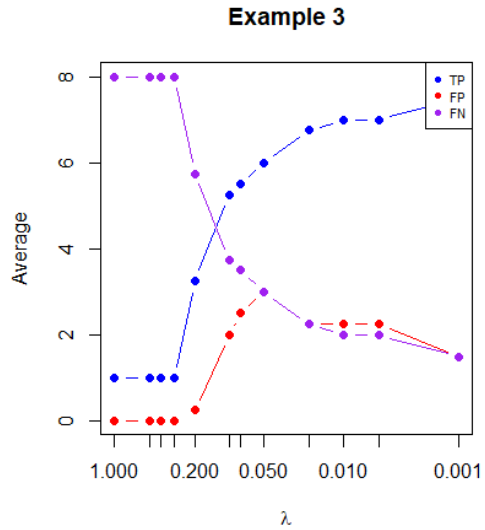


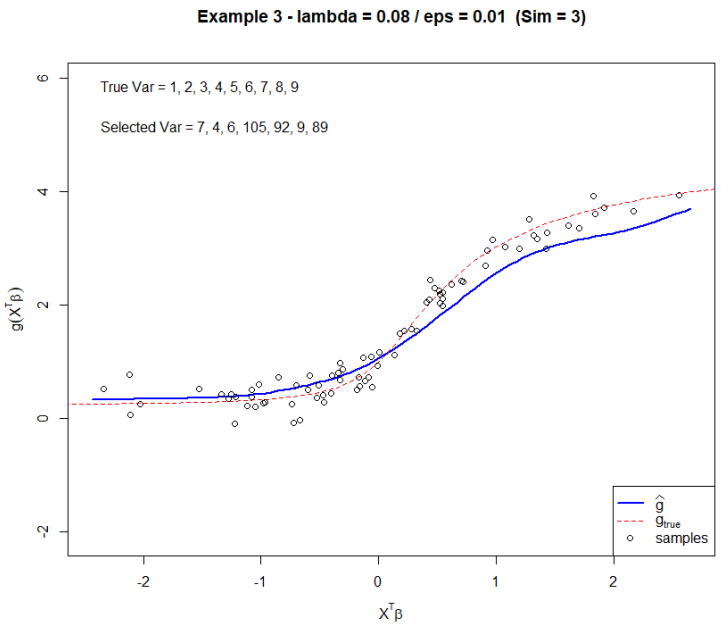




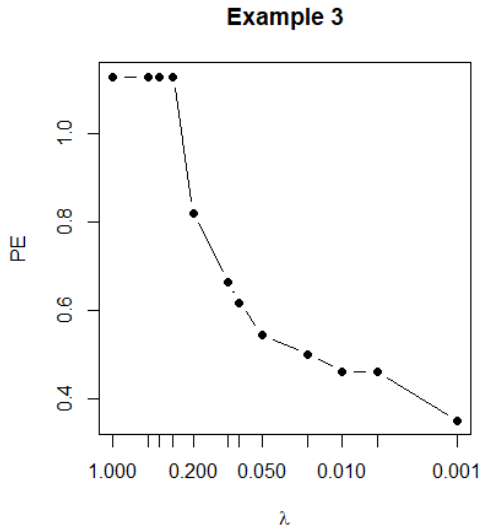
Our Findings

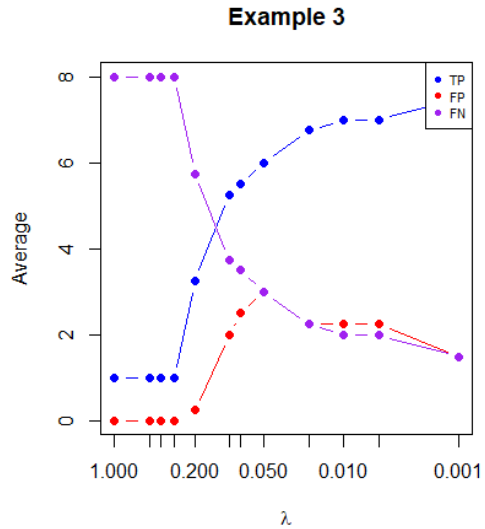


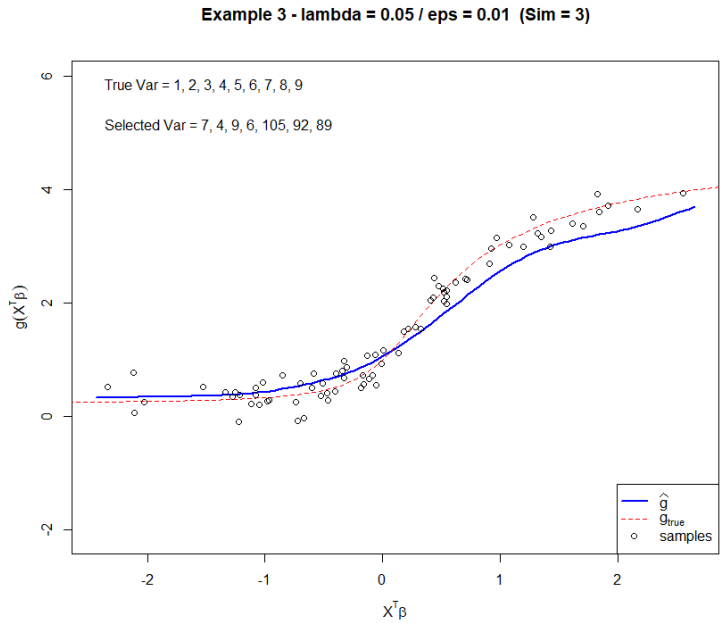




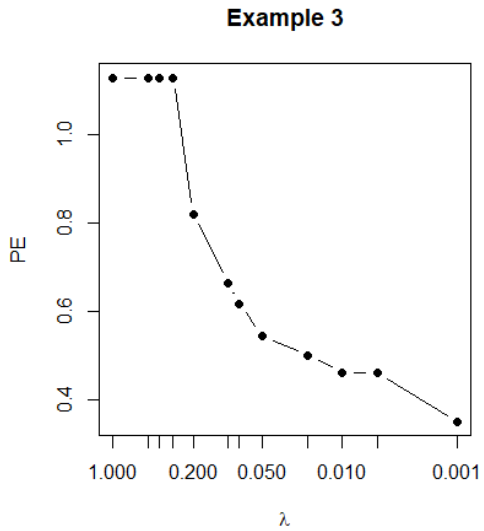
Our Findings

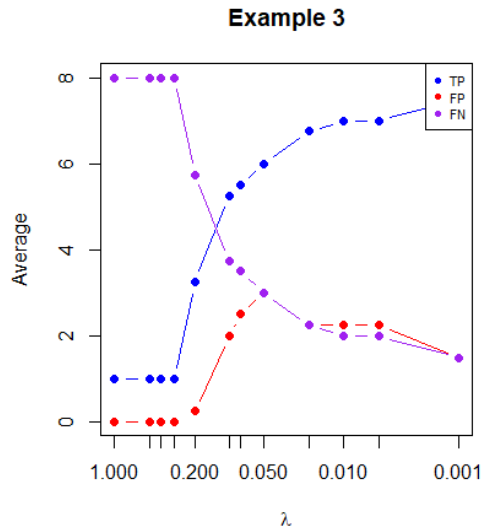


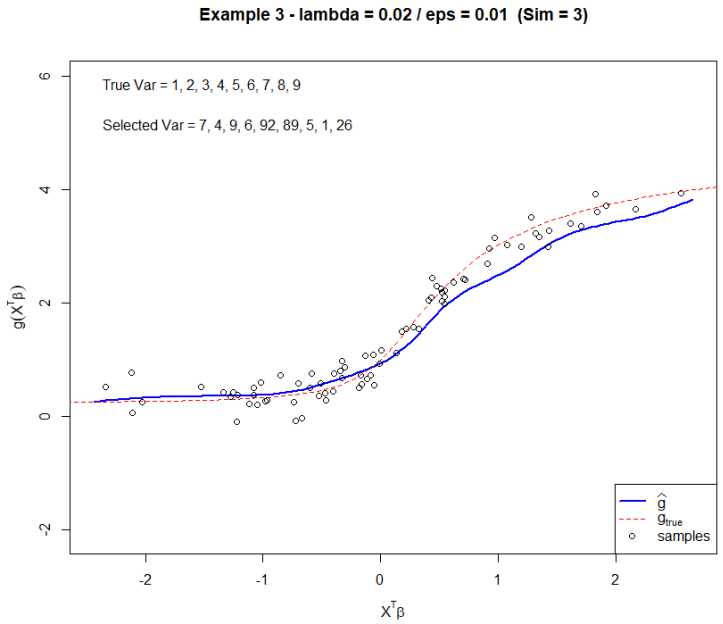




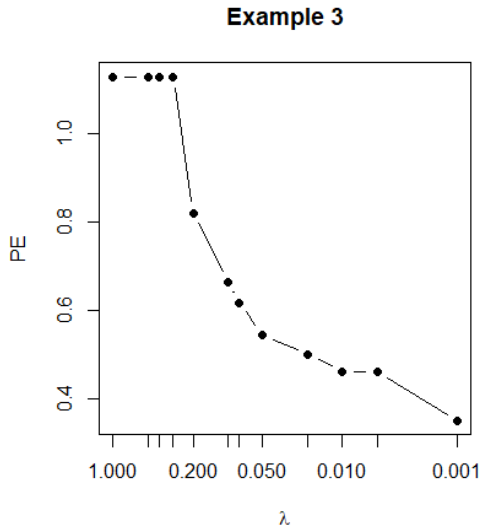
Our Findings

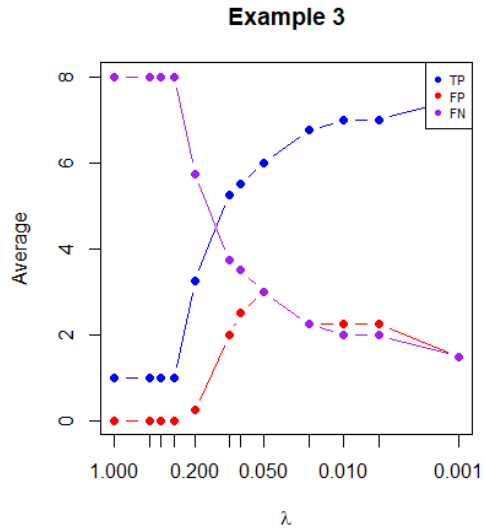


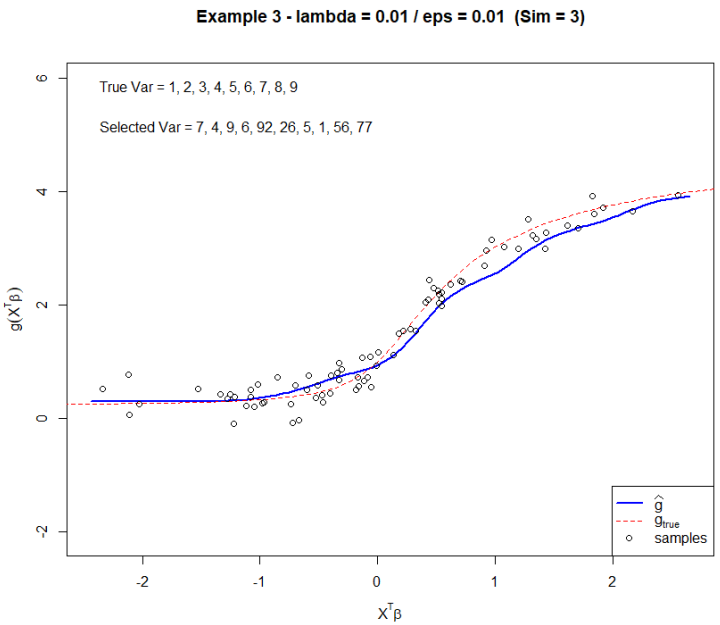




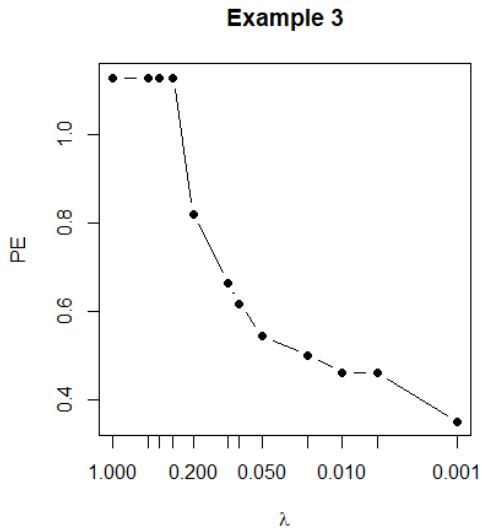
Our Findings

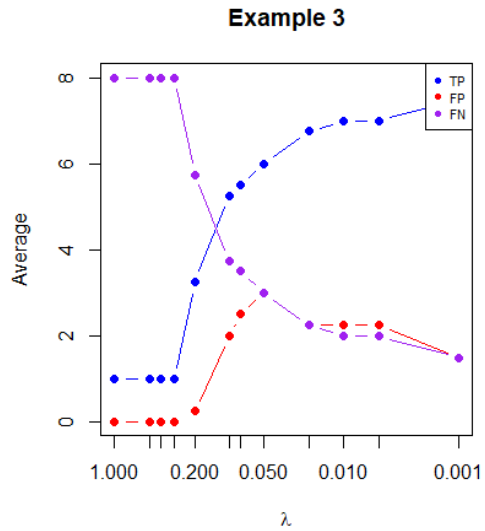


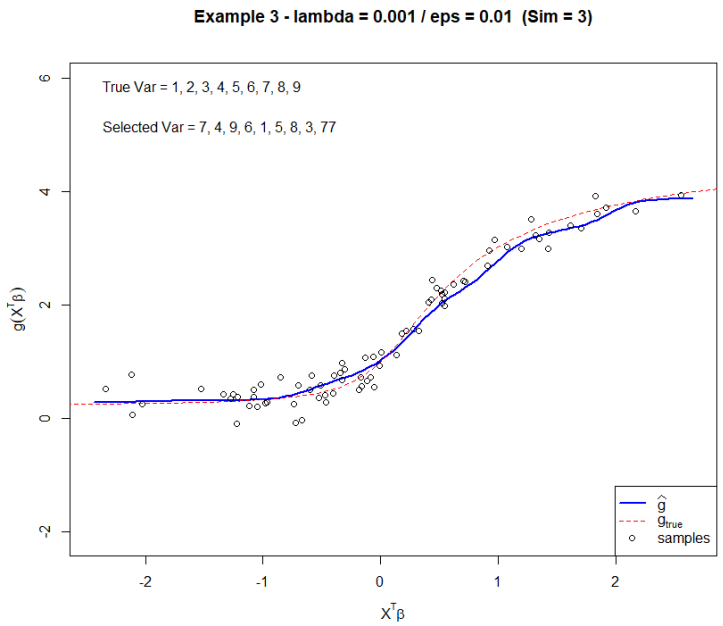




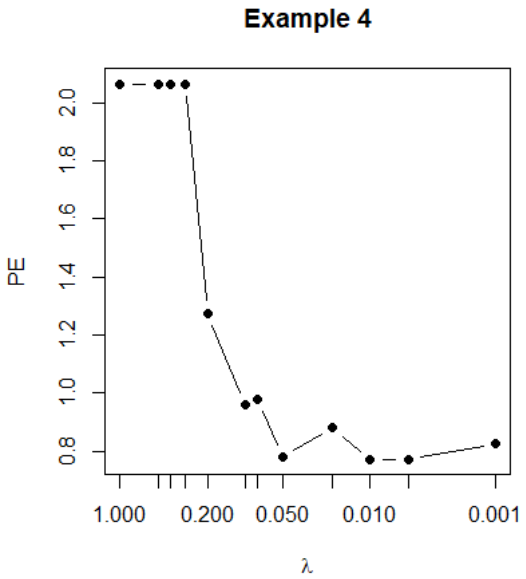
Our Findings

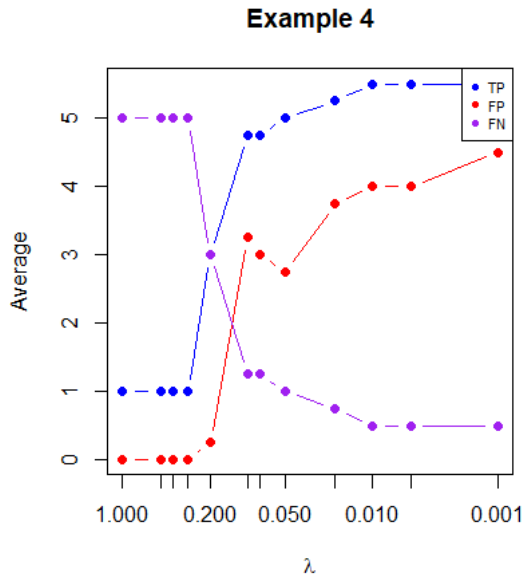


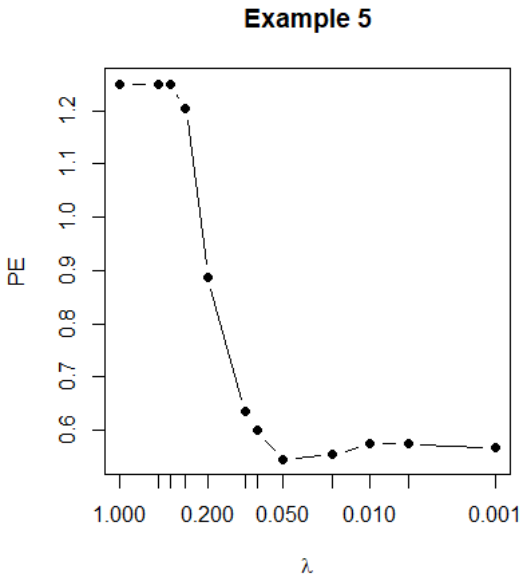


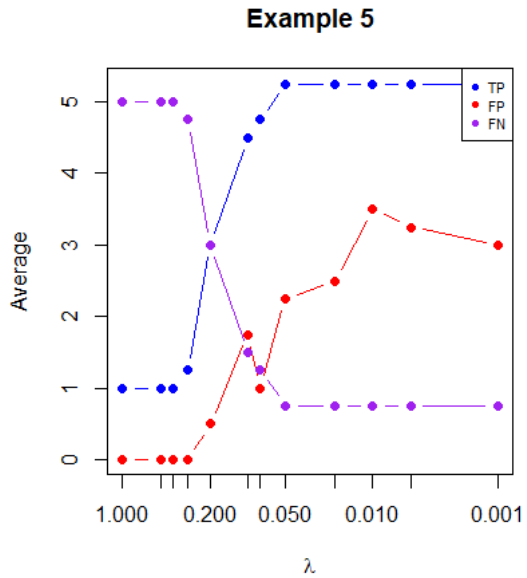


Our Findings

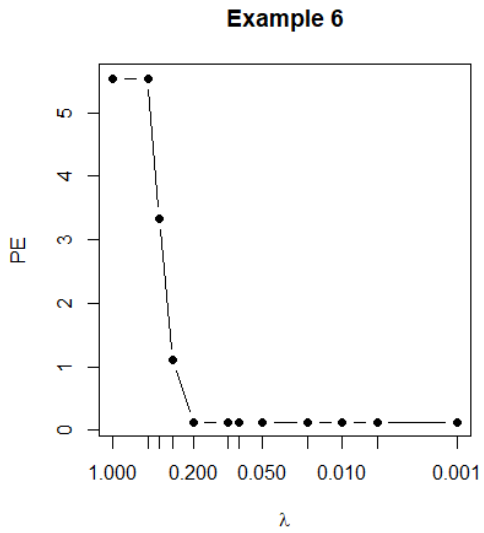


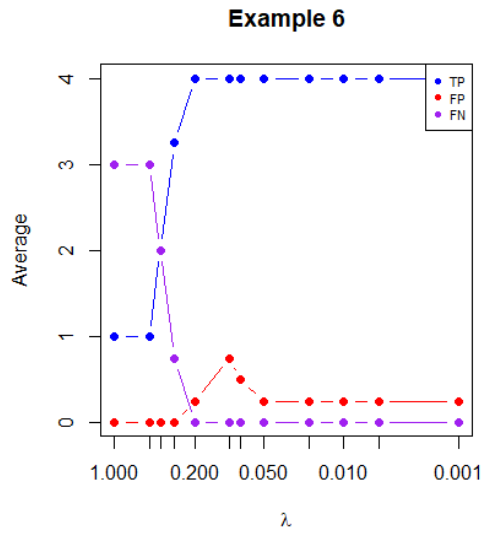


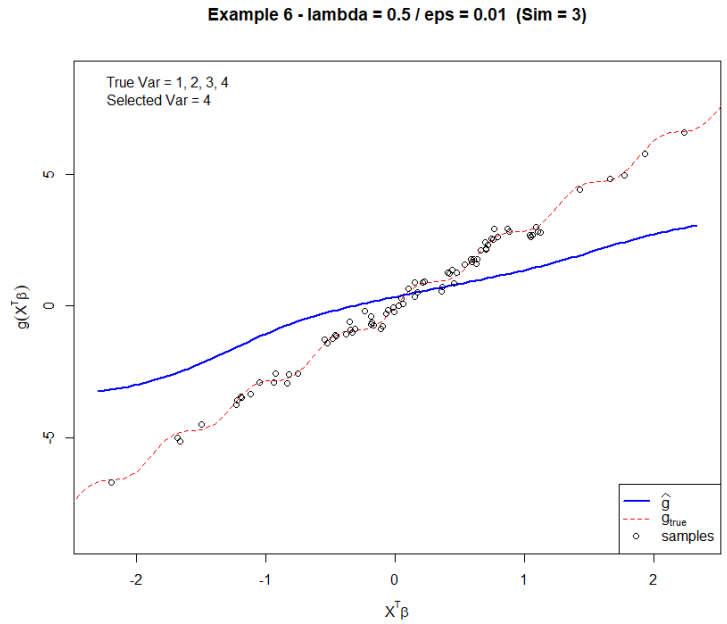




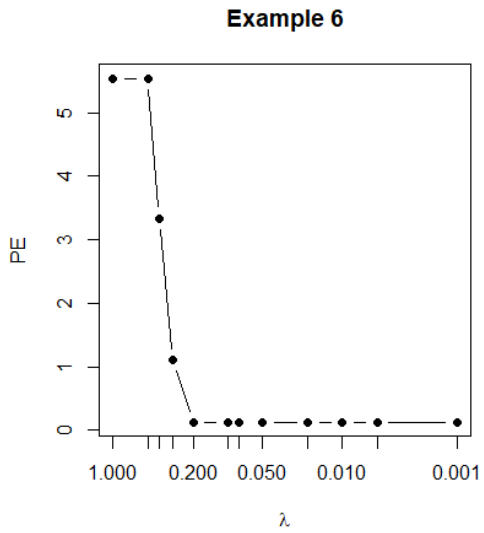
Our Findings

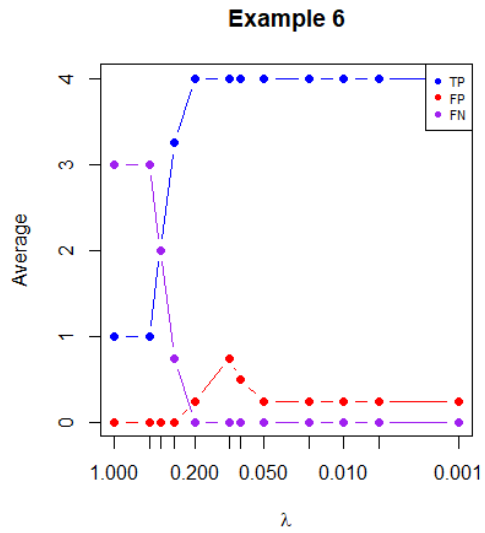


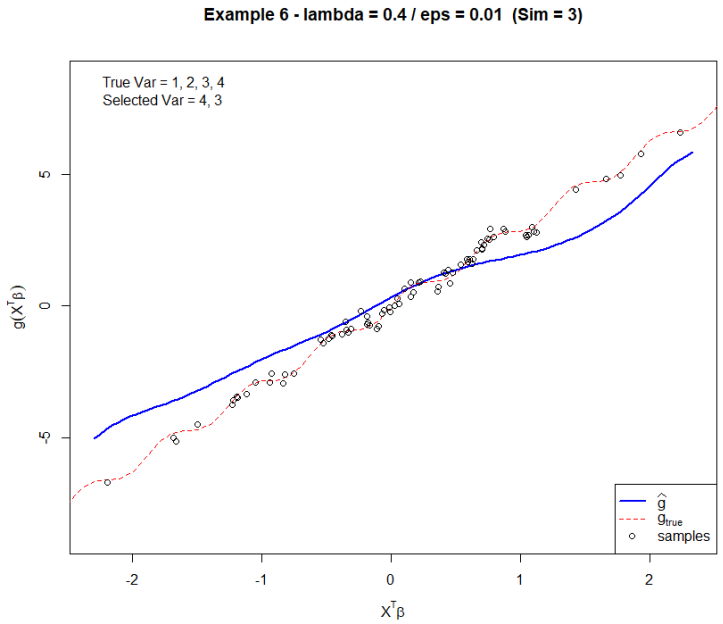




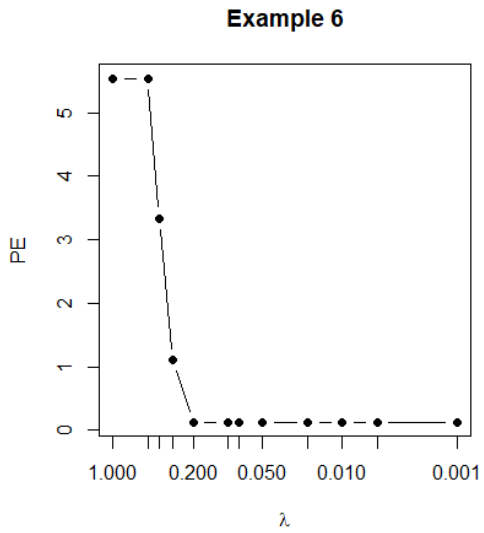
Our Findings

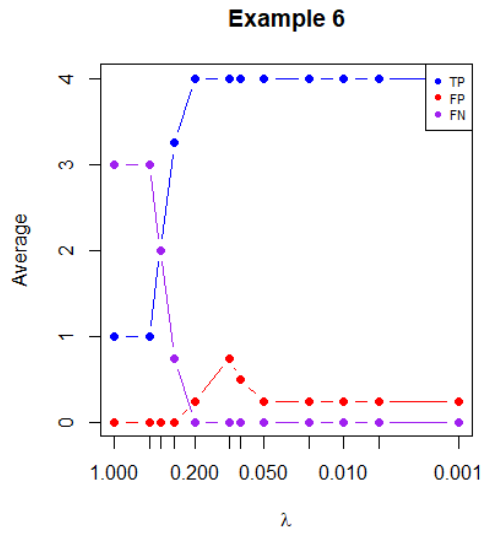


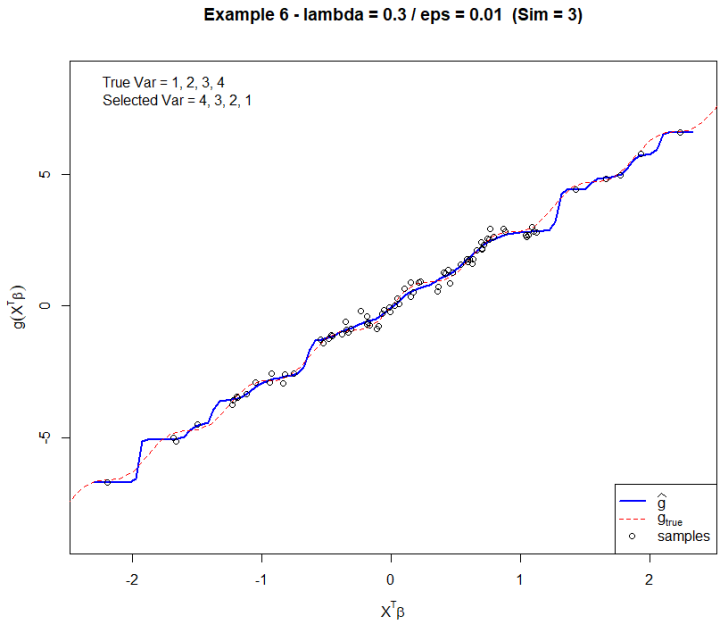




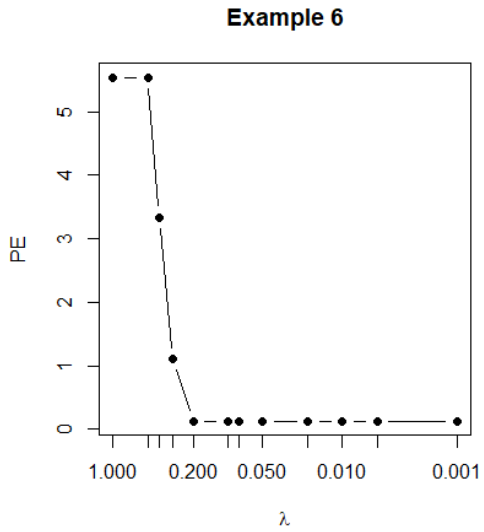
Our Findings

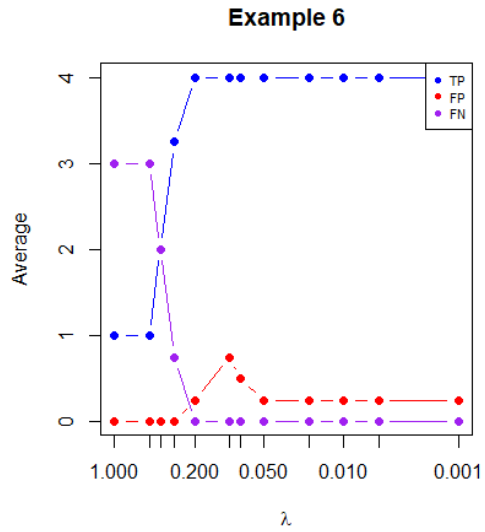


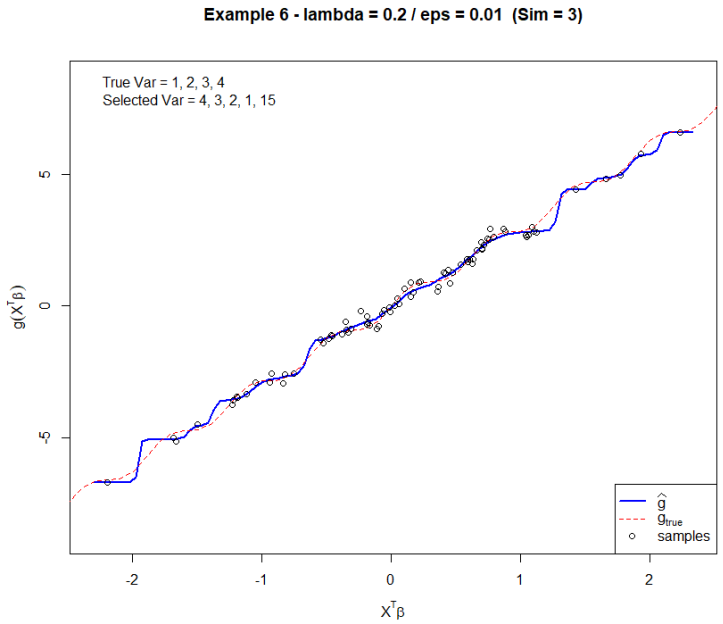




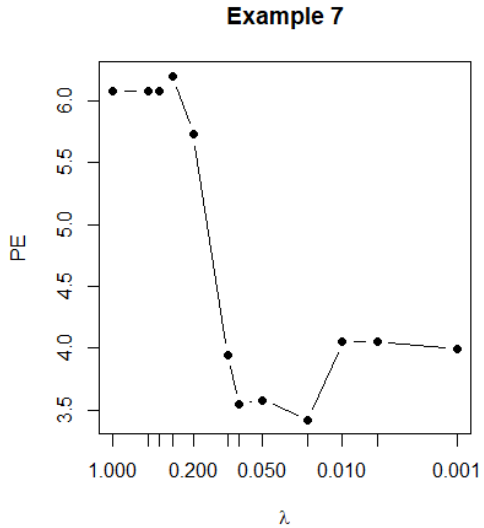
Our Findings

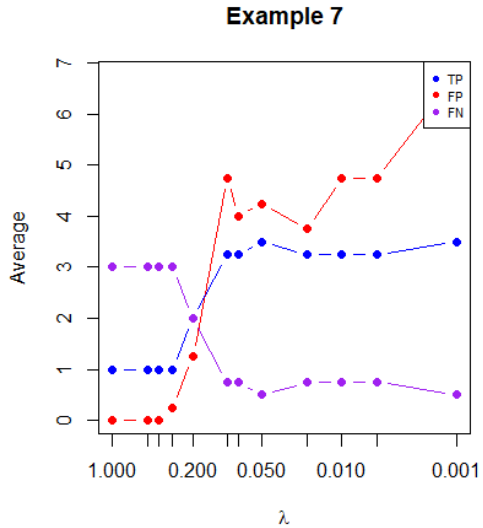


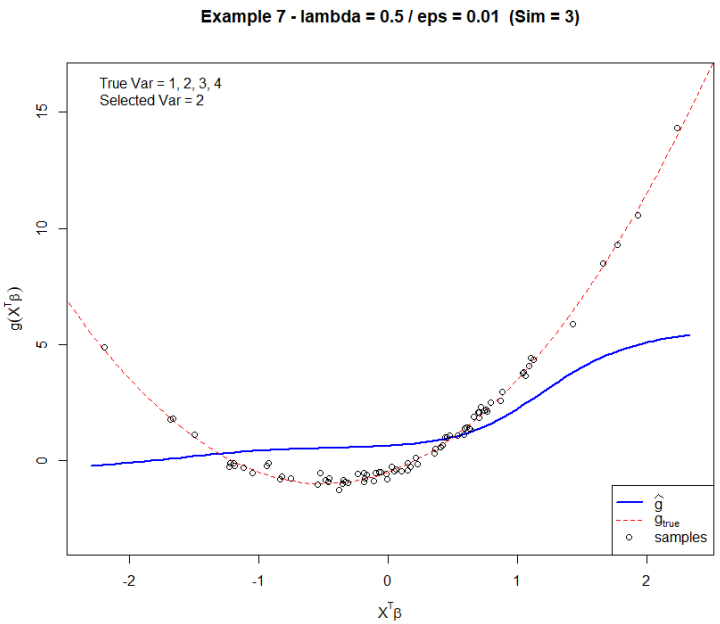




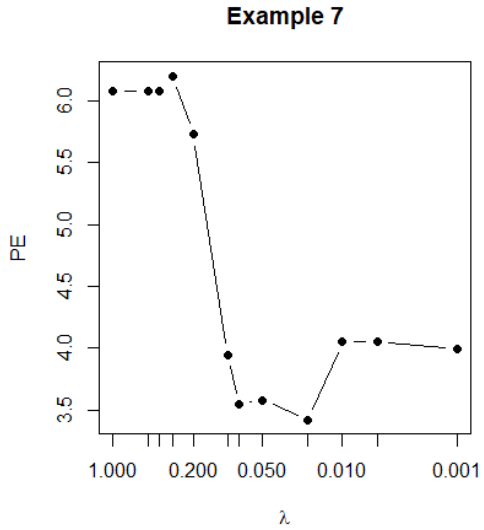
Our Findings

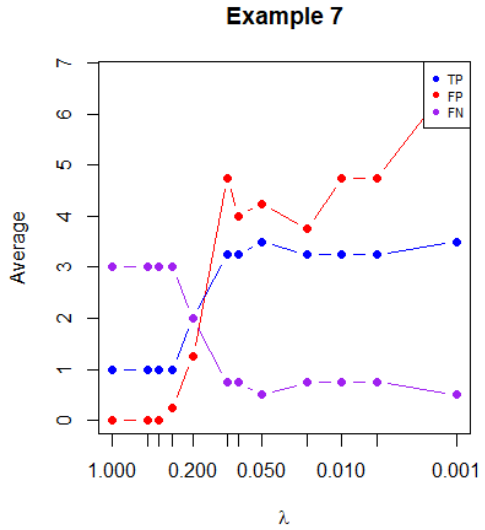


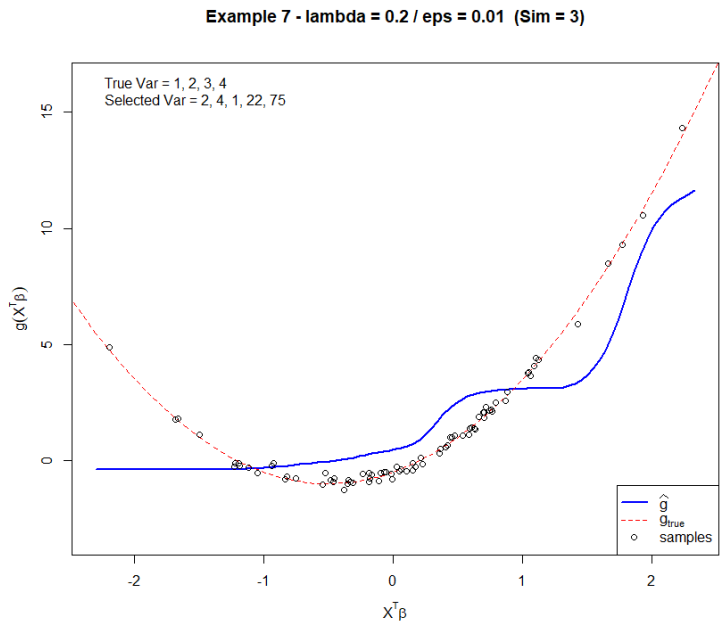




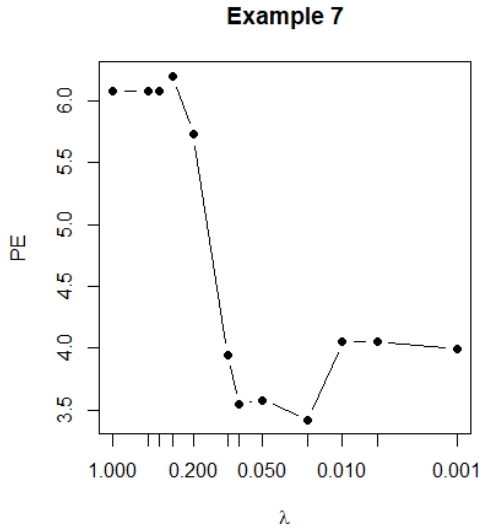
Our Findings

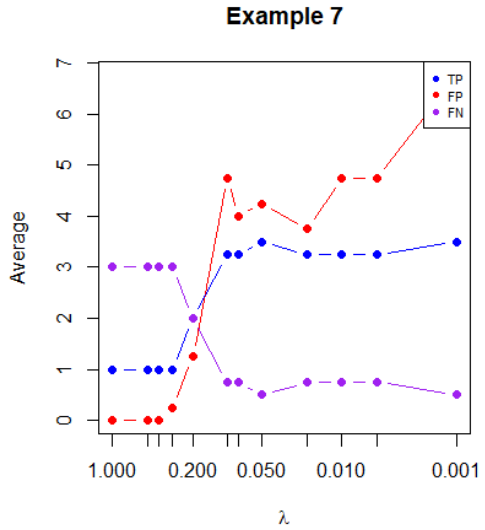


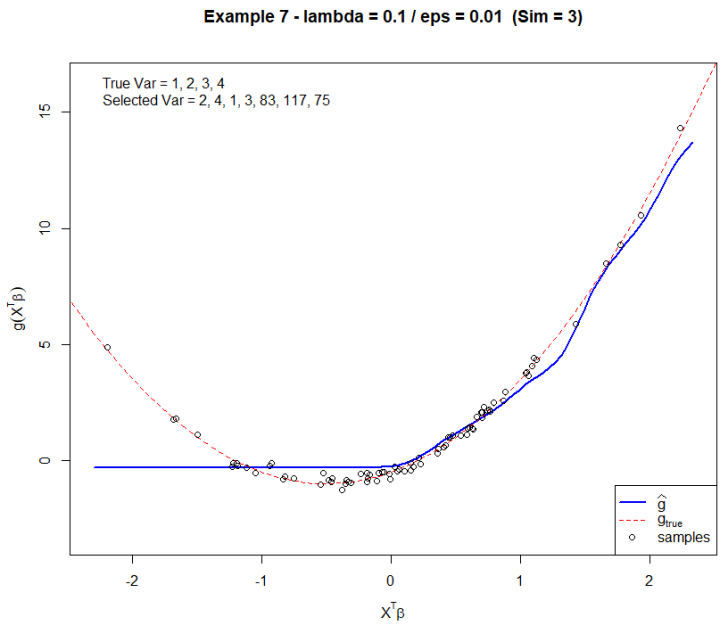




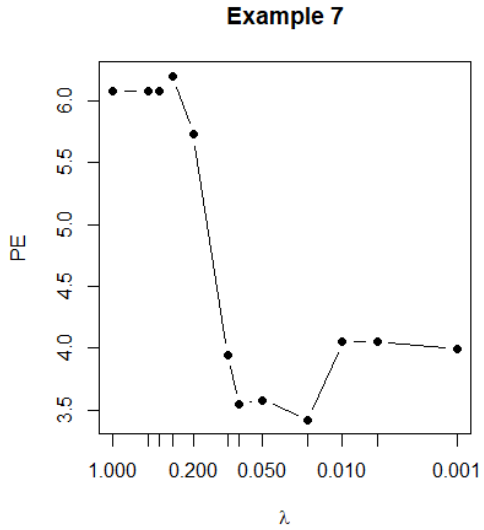
Our Findings

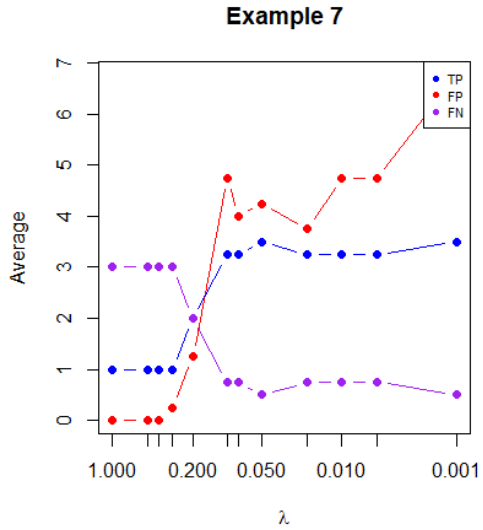


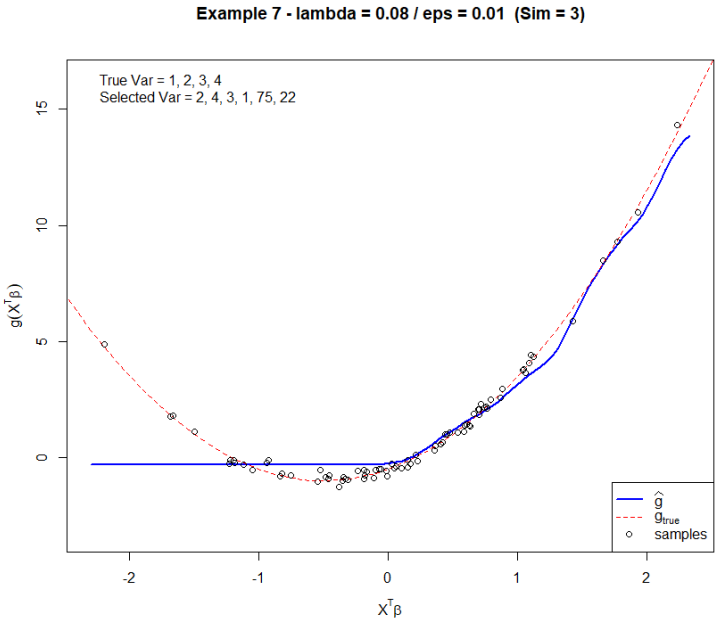




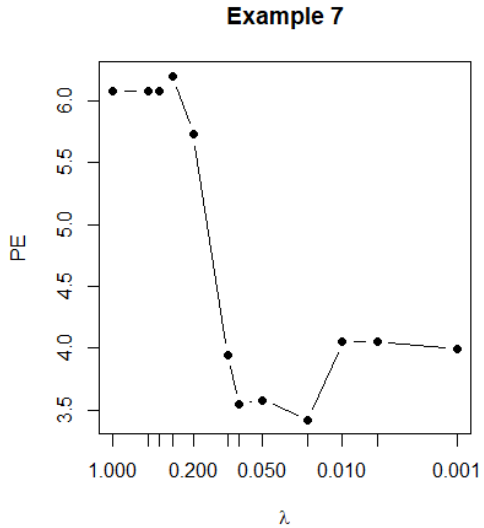
Our Findings

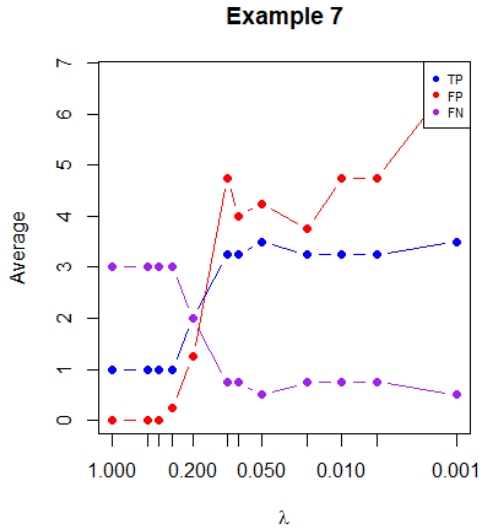






Our Findings





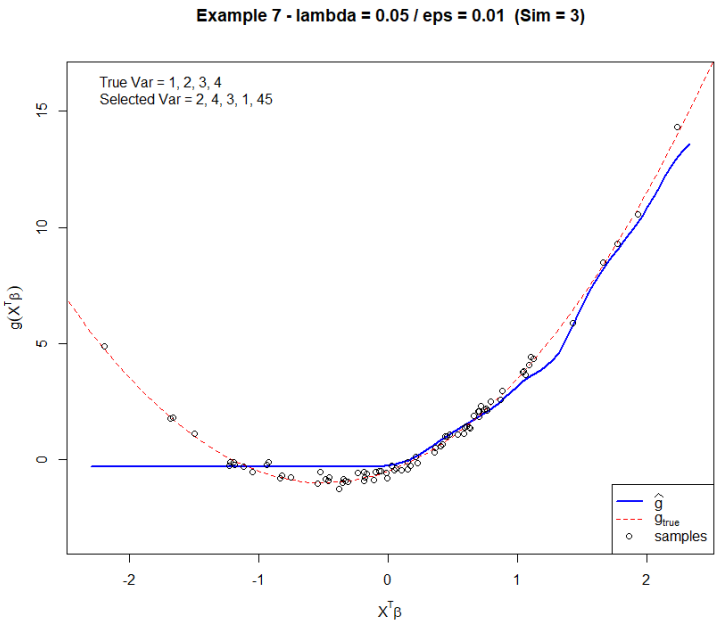


Table of Contents

Boston Housing Data

Boston housing data is used to illustrate the proposed procedure. The dataset has 506 samples and 14 variables.

It consists of several potential predictors of house prices and the response **medv** which is the median-value of owner-occupied homes.

Variable Selection

1. crim = per capita crime rate by town.
2. zn = proportion of residential land zoned for lots over 25,000 sq.ft.
3. indus = proportion of non-retail business acres per town.
4. chas = Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
5. nox = nitrogen oxides concentration (parts per 10 million).
6. rm = average number of rooms per dwelling.
7. age = proportion of owner-occupied units built prior to 1940.
8. dis = weighted mean of distances to five Boston employment centres.
9. rad = index of accessibility to radial highways.
10. tax = full-value property-tax rate per \$10,000.
11. ptratio = pupil-teacher ratio by town.
12. black = $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.
13. lstat = lower status of the population (percent).

Common Variables SKIMMS/LASSO (8)

1. crim = per capita crime rate by town.

3. indus = proportion of non-retail business acres per town.
4. chas = Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).

6. rm = average number of rooms per dwelling.

8. dis = weighted mean of distances to five Boston employment centres.

11. ptratio = pupil-teacher ratio by town.
12. black = $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.
13. lstat = lower status of the population (percent).

SKIMMS Selection (9)

1. crim = per capita crime rate by town.

3. indus = proportion of non-retail business acres per town.
4. chas = Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).

6. rm = average number of rooms per dwelling.

8. dis = weighted mean of distances to five Boston employment centres.
9. rad = index of accessibility to radial highways.

11. ptratio = pupil-teacher ratio by town.
12. black = $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.
13. lstat = lower status of the population (percent).

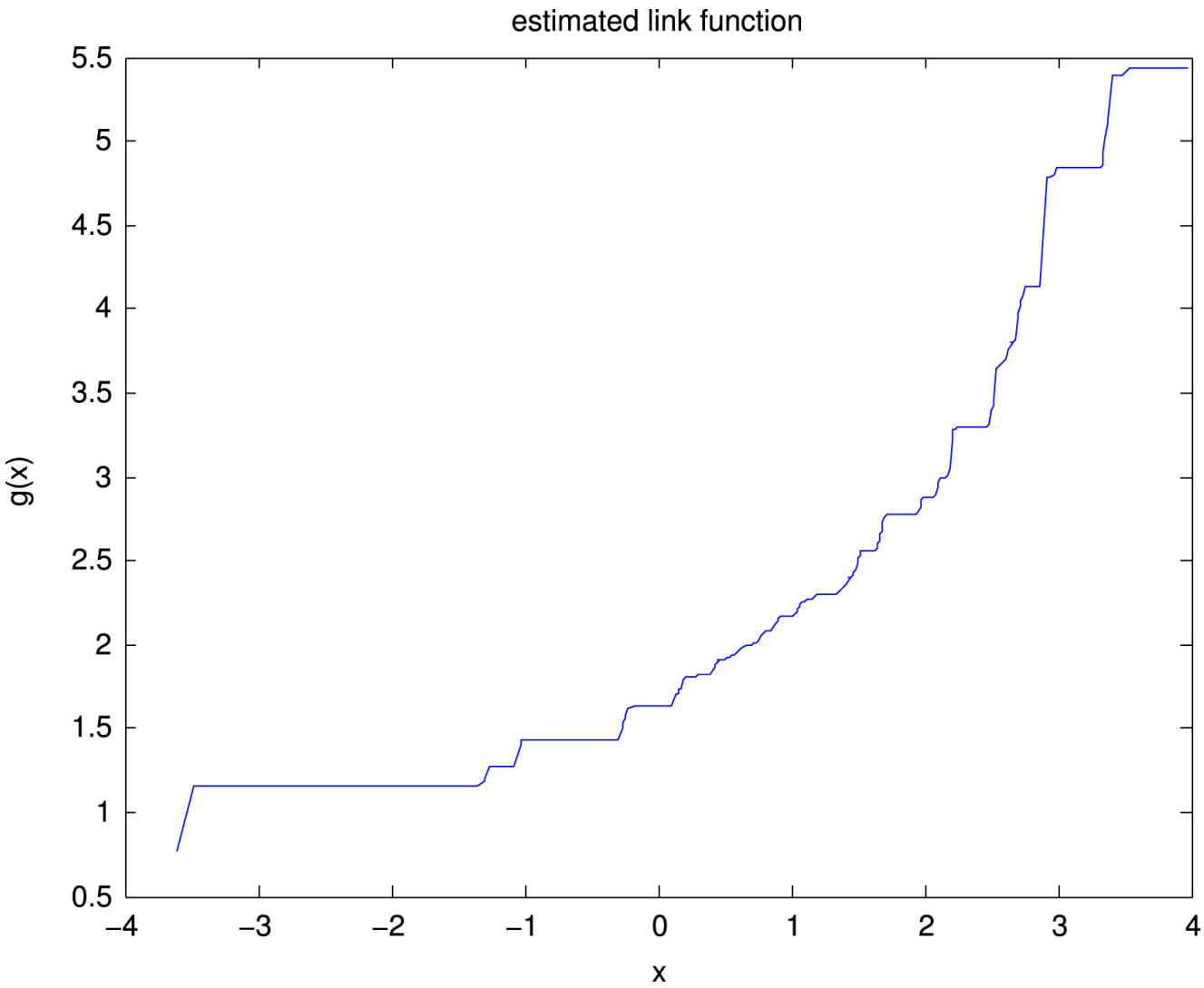
LASSO Selection (10)

1. crim = per capita crime rate by town.
2. zn = proportion of residential land zoned for lots over 25,000 sq.ft.
3. indus = proportion of non-retail business acres per town.
4. chas = Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
5. nox = nitrogen oxides concentration (parts per 10 million).
6. rm = average number of rooms per dwelling.

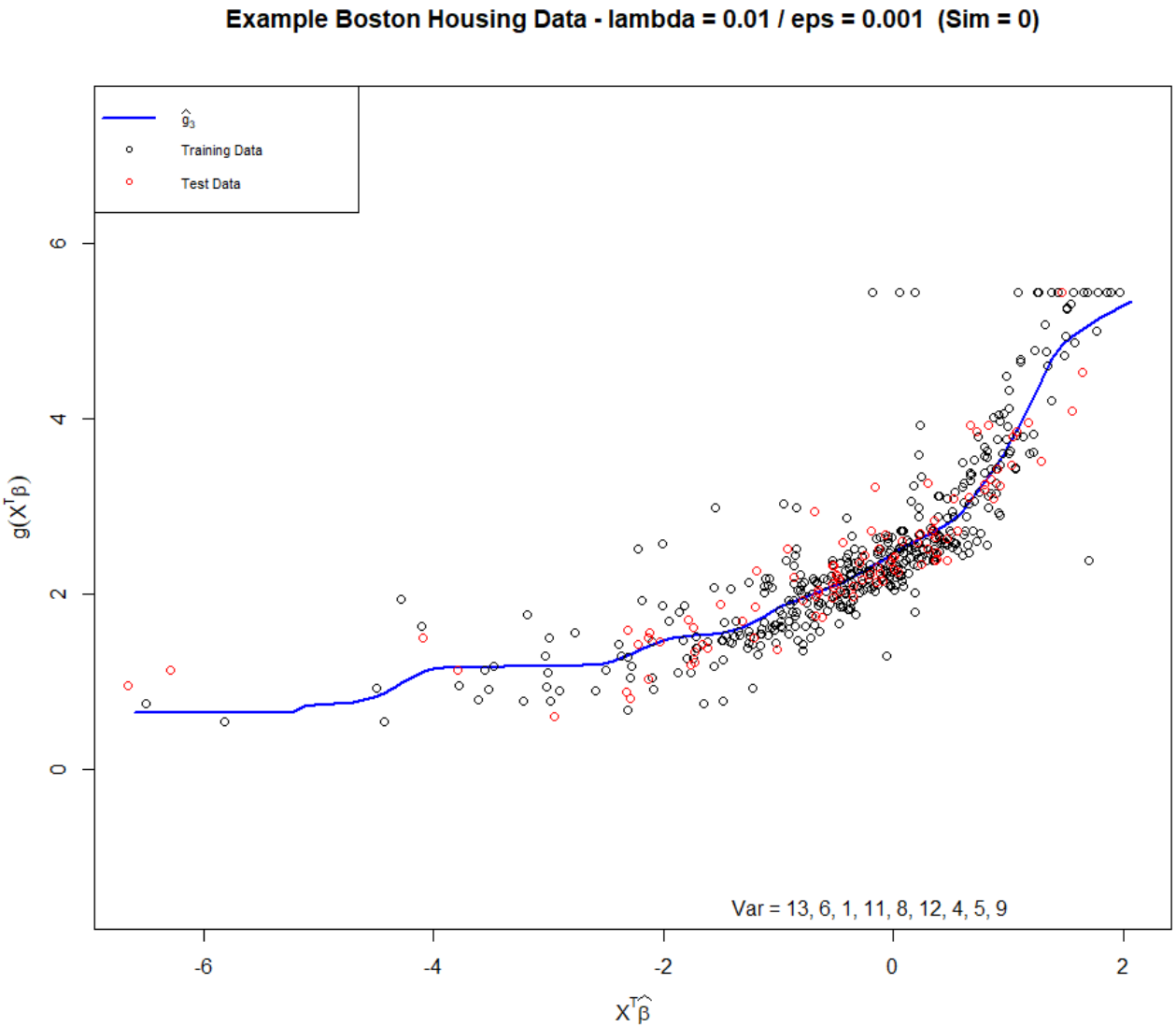
8. dis = weighted mean of distances to five Boston employment centres.

11. ptratio = pupil-teacher ratio by town.
12. black = $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town.
13. lstat = lower status of the population (percent).

Paper Estimations



Our Estimations



Summary

S moothed

K endall

I terative

M aximizer

M odel

S elector

Bibliography

[1] Shikai Luo, Subhashis Ghosal, **Forward Selection and Estimation in high dimensional single index models** Statistical Methodology, Volume 33, 2016, Pages 172-179, ISSN 1572-3127, https://doi.org/10.1016/j.stamet.2016.09.002.

Thank you ! :)