Automatic Model Structure Selection for Partially Linear Models

Yufeng Liu

University of North Carolina at Chapel Hill

http://www.unc.edu/∼yfliu

Joint work with Hao Helen Zhang (NCSU) and Guang Cheng (Purdue)

Outline

- Motivation
- Partially Linear Models
  - Structure Selection
  - Model Estimation
- New Regularization Method
  - Framework
  - Implementation and Computation Algorithm
  - Theoretical Properties
- Numerical Examples
- Summary

Various Regression Models

Given $(\boldsymbol{x}_i, y_i)_{i=1}^n$, $\boldsymbol{x} \in R^d$, a general regression model

$$y_i = f(\boldsymbol{x}_i) + \varepsilon_i, \quad E(\varepsilon_i) = 0, \text{Var}(\varepsilon_i) = \sigma^2.$$

- Linear model: $f(\boldsymbol{x}) = b + \sum_{j=1}^d \beta_j x_j$.
- Additive model: $f(\boldsymbol{x}) = b + \sum_{j=1}^d f_j(x_j)$, the form f_j is unspecified
- Partially linear model:

$$f(\boldsymbol{x}) = b + \sum_j \beta_j x_j + \sum_k f_k(x_k).$$

Partially linear models: flexible, highly interpretable.

Partially Linear Models

Widely used in longitudinal data analysis. Given the covarites $(X_1, \cdots, X_d, T)$, T is *time*,

$$Y = \boldsymbol{X}^T \boldsymbol{\beta} + f(T) + \varepsilon.$$

Estimation and inferences are typically based on the known model structure:

- kernel-smoothing approaches (Speckman 1998)
- smoothing splines (Engle et al. 1983; Heckman 1986; Wahba 1990; Green and Silverman 1994)
- penalized regression splines (Ruppert et al. 2003; Liang 2006)

Two Main Issues

Structure Selection Problem:

- Need to determine which covariates are linear and which are nonlinear

Model Estimation Problem:

- Fit the model and make inferences

Existing works typically

- tackle these two problems separately: first decide/guess the model structure, then estimate the model based on the stucture assumption

Structure Selection

In practice,

- assume the model structure is known (based on prior knowledge)
- make an educated guess: univariate analysis, marginal scatter plot (kind of heuristic)
- fit an additive model and test nonlinearity (difficult for large dimensional data)

Also, they usually allow only one nonlinear effect, for example, time

Our Goals

We propose a new class of frameworks which

- allow multiple nonlinear effects
- automatically detect which covariates are nonlinear and which ones are linear
- also detect which covariates are not useful for prediction
- simultaneously conduct structure selection and model estimation

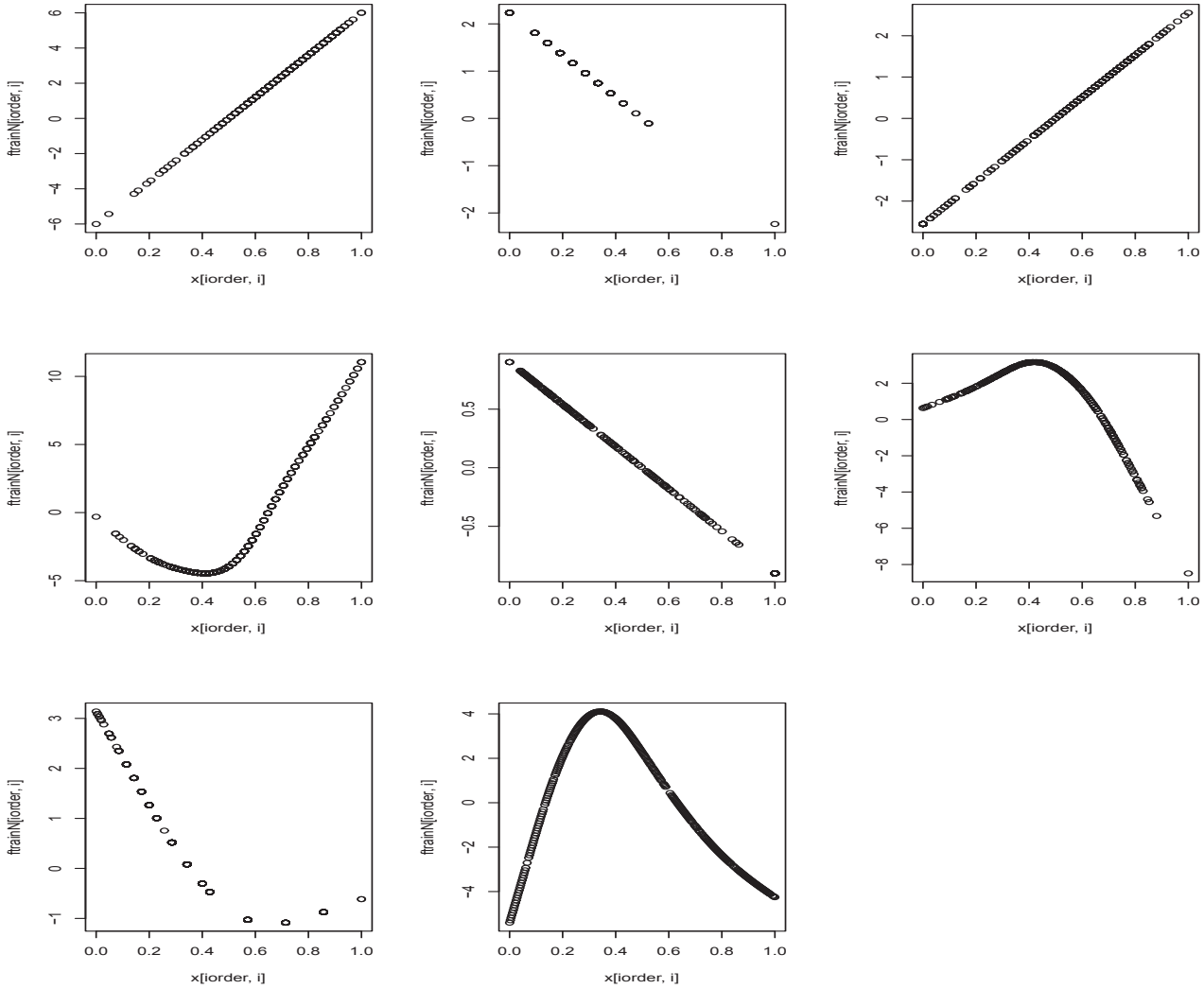
Ozone Concentration in Los Angeles Basin

Daily measurements of ozone concentration in the Los Angeles basin in 1976

$Y =$ Upland ozone concentration (ppm), Eight meterological quantities:

- vdht: Vandenberg 500 millibar height (m)
- wdsp: Wind speed (mph)
- hdmt: Humidity ($\%$)
- sbtp: Sandburg Air Base temperature (degree)
- ibht: Inversion base height (ft)
- dgpg: Dagget pressure gradient (mmHg)
- ibtp: Inversion base temperature (degree)
- vsty: Visibility (miles)

Ozone Concentration Data Marginal Plot



Basic Framework

Let f_j be the regression function of x_j. Hypothetically, we can decompose f_j as

$$f_j(x_j) = b_j + \beta_j x_j + h_j(x_j) \equiv f_j^L(x_j) + f_j^N(x_j)$$

- x_j is linear: if $\beta_j \neq 0$ and $h_j(x_j)$ is zero function
- x_j is nonlinear: if h_j is not a zero function
  - A "purely nonlinear" effect: $\beta_j = 0$ and $h_j \neq 0$
  - A "linear and nonlinear" effect: $\beta_j \neq 0$ and $h_j \neq 0$.

To assure identifiability, we enforce f_j^L and f_j^N to be orthogonal to each other in some sense.

Index Set Partition

Define the index set $I = \{1, \cdots, d\}$. Then

$$I = I_L \cup I_N \cup I_0,$$

where $I_N = I_{PN} \cup I_{LN}$ and

$$\begin{aligned}
I_L &= \{j : f_j^L \neq 0, f_j^N \equiv 0\}, && \text{(purely linear)} \\
I_{PN} &= \{j : f_j^L \equiv 0, f_j^N \neq 0\}, && \text{(purely nonlinear)} \\
I_{LN} &= \{j : f_j^L \neq 0, f_j^N \neq 0\}, && \text{(linear nonlinear)} \\
I_O &= \{j : f_j^L \equiv 0, f_j^N \equiv 0\}, && \text{(null effect)}.
\end{aligned}$$

Structured Model Representation

A structure representation for the true model is

$$
\begin{aligned}
f(\boldsymbol{x}) &= b + \sum_{j \in I_L} \beta_j x_j + \sum_{j \in I_{PN}} f_j^N(x_j) \\
&+ \sum_{j \in I_{LN}} (\beta_j x_j + f_j^N(x_j)) + \sum_{j \in I_0} 0(X_j).
\end{aligned}
$$

Two goals:

- need to determine three index subsets I_L, I_N, I_0.
- need to estimate all the function components.

Reproducing kernel Hilbert Space (RKHS)

Estimate f_j in a rich function class $\mathcal{H}_j$ (RKHS)

- We use RKHS theory to formulate the regularization problem
- Second-order Sobolev Hilbert space

$$\mathcal{S}_2[0,1] = \{g : g, g' \text{absolutely cont.}, g'' \in \mathcal{L}_2[0,1]\},$$

equipped with the norm

$$\|g\|^2_{\mathcal{H}_j} = \left[\int_0^1 g(x)dx\right]^2 + \left[\int_0^1 g'(x)dx\right]^2 + \int_0^1 \left[g''(x)\right]^2 dx.$$

Orthogonal Decomposition

To distinguish linear and nonlinear effects, decompose $\mathcal{H}_j$ as

$$\mathcal{H}_j = \{1\} \oplus \mathcal{H}_{0j} \oplus \mathcal{H}_{1j},$$

- $\{1\}$ is the constant space,
- $\mathcal{H}_{0j} = \{g : g'' = 0\}$ is the linear function subspace
  - $\mathcal{H}_{0j} =$ span$\{x_j - 1/2\}$. Denote $k_1(x) = x - \frac{1}{2}$.
- $\mathcal{H}_{1j} = \{g : \int_0^1 g^{(\nu)} = 0, \nu = 0, 1; g'' \in \mathcal{L}_2[0, 1]\}$
  - $\mathcal{H}_{1j}$ is the nonlinear subspace, orthogonal to $\mathcal{H}_{0j}$.
  - $\mathcal{H}_{1j}$ is an RKHS with the reproducing kernel R_{1j} (Wahba, 1990).

Functional ANOVA Decomposition

We estimate f in the tensor sum of $\mathcal{H}_j$'s

$$\mathcal{H} = \bigoplus_{j=1}^{d} \mathcal{H}_j = \{1\} \bigoplus \oplus_{j=1}^{d} \mathcal{H}_{0j} \bigoplus \oplus_{j=1}^{d} \mathcal{H}_{1j}.$$

The corresponding function decomposition is

$$f(\mathbf{x}) = b + \sum_{j=1}^{d} \beta_j k_1(x_j) + \sum_{j=1}^{d} f_{1j}(x_j.)$$

$f_{1j} = \mathcal{P}_{1j}(f) \in \mathcal{H}_{1j}$. Here $\mathcal{P}_{1j}$ is the projection operator from $\mathcal{H}$ to $\mathcal{H}_{1j}$.

Regularization Problem

We propose to solve

$$\min_{f \in \mathcal{H}} \quad \sum_{i=1}^{n} [y_i - f(x_i)]^2 + \lambda_1 \sum_{j=1}^{d} w_{0j} |\beta_j|$$
$$+ \lambda_2 \sum_{j=1}^{d} w_{1j} \|\mathcal{P}_{1j}(f)\|_{\mathcal{H}_{1j}}$$

- $(\lambda_1, \lambda_2) > 0$ are tuning parameters.
- w_{0j}, w_{1j} are pre-determined weights

Adaptive Weights

Weights $w_{0j}, w_{1j} > 0$ are chosen such that: important components are penalized less than unimportant components

- adaptive LASSO penalty on linear parts (Tibshirani 1996, Zou 2006);
- adaptive COSSO penalty on nonlinear parts (Lin and Zhang, 2006; Zhang and Lin 2006)

Representer Theorem

The solution exists and lies in a finite dimensional space.

Lemma: Let $\widehat{f}$ be the minimizer, represented as

$$\widehat{f} = \widehat{b} + \sum_{j=1}^{d} \widehat{\beta}_j k_1(x_j) + \sum_{j=1}^{d} \widehat{f}_{1j}(x_j).$$

Then

$$\widehat{f}_{1j} \in \text{span}\{R_{1j}(x_i, \cdot), i = 1, \cdots, n\},$$

where $R_{1j}(\cdot, \cdot)$ is the reproducing kernel of the space $\mathcal{H}_{1j}$.

Equivalent Formulation

Define $\mathbf{y} = (y_1, \cdots, y_n)^\intercal$, $\mathbf{f} = (f(\boldsymbol{x}_1), \cdots, f(\boldsymbol{x}_n))^\intercal$, $\boldsymbol{\beta} = (b, \beta_1, \cdots, \beta_d)^\intercal$ and $\mathbf{c} = (c_1, ..., c_n)^\intercal$. Define $\mathbf{R}_{1j} = [[R_{1j}(x_i, x_{i'})]]$ and $\mathbf{R}_{\mathbf{w}_1,\boldsymbol{\theta}} = \sum_{j=1}^d w_{1j}^{-1}\theta_j \mathbf{R}_{1j}$. Let $\mathbf{T}$ be the linear design matrix. Then $\mathbf{f} = \mathbf{T}\boldsymbol{\beta} + \mathbf{R}_{\mathbf{w}_1,\boldsymbol{\theta}}\mathbf{c}$.

$$\min_{\mathbf{c},\boldsymbol{\beta},\boldsymbol{\theta}} (\mathbf{y} - \mathbf{T}\boldsymbol{\beta} - \mathbf{R}_{\mathbf{w}_1,\boldsymbol{\theta}}\mathbf{c})^\intercal(\mathbf{y} - \mathbf{T}\boldsymbol{\beta} - \mathbf{R}_{\mathbf{w}_1,\boldsymbol{\theta}}\mathbf{c})$$

$$+\lambda_1 \sum_{j=1}^d w_{0j}|\beta_j| + \tau_0 \mathbf{c}^\intercal \mathbf{R}_{\mathbf{w}_1,\boldsymbol{\theta}}\mathbf{c} + \lambda_2 \sum_{j=1}^d w_{1j}\theta_j,$$

where $\theta_j \geq 0$, $j = 1, ..., d$. Here τ_0 is a fixed constant.

Computational Algorithm

Minimize with respect to $(\boldsymbol{\theta}, \boldsymbol{\beta})$ and $\mathbf{c}$ alternatively.

- $\mathbf{c}$-step: Fixing $(\boldsymbol{\theta}, \boldsymbol{\beta})$ at $(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\beta}})$, solve

$$\min_{\mathbf{c}} (\mathbf{z} - \mathbf{R}_{\mathbf{w}_1, \widehat{\boldsymbol{\theta}}} \mathbf{c})^{\intercal} (\mathbf{z} - \mathbf{R}_{\widehat{\boldsymbol{\theta}}} \mathbf{c}) + \tau_0 \mathbf{c}^{\intercal} \mathbf{R}_{\mathbf{w}_1, \widehat{\boldsymbol{\theta}}} \mathbf{c}.$$

  where $\mathbf{z} = \mathbf{y} - \mathbf{T}\widehat{\boldsymbol{\beta}}$.

- $(\boldsymbol{\theta}, \boldsymbol{\beta})$-step: Define $\mathbf{l}_j = w_{1j}^{-1} \mathbf{R}_j \widehat{\mathbf{c}}$. Solve

$$\min_{\boldsymbol{\theta} \geq \mathbf{0}, \boldsymbol{\beta}} (\mathbf{y} - \mathbf{T}\boldsymbol{\beta} - \mathbf{L}\boldsymbol{\theta})^{\intercal} (\mathbf{y} - T\boldsymbol{\beta} - \mathbf{L}\boldsymbol{\theta}) + \lambda_1 \sum_{j=1}^{d} w_{0j} |\beta_j|$$
$$+ \tau_0 \widehat{\mathbf{c}}^{\intercal} \mathbf{L}\boldsymbol{\theta} + \lambda_2 \sum_{j=1}^{d} w_{1j} \theta_j,$$

  where $\mathbf{L} = (\mathbf{l}_1, \cdots, \mathbf{l}_d)$.

How to Choose Weights

Adaptive weights: w_{0j} and w_{1j}

- We construct the weights using some (good) initial estimates
- Traditional SS-ANOVA fit: replace the penalty with roughness penalty
- Denote the SS-ANOVA solution by $\widetilde{f}$ and

$$w_{0j} = |\tilde{\beta}_j|^{-\gamma_0}, \quad w_{1j} = \|\mathcal{P}_{1j}(\widetilde{f})\|_{\mathcal{L}_2}^{-\gamma_1}.$$

$\gamma_0, \gamma_1 > 0$.

Theoretical Results

Denote the solution by $\widehat{f}_{\lambda_1,\lambda_2}$. Let

$$\widehat{I}_L = \{j : \widehat{\beta}_j \neq 0, \widehat{f}_j^N \equiv 0\}, \widehat{I}_N = \{j : \widehat{f}_j^N \neq 0\},$$

$$\widehat{I}_0 = I \backslash (\widehat{I}_L \cup \widehat{I}_N).$$

Under some regularity conditions, we can

- establish the convergence rate of the function estimator.
- show that the procedure can identify the correct model structure asymptotically.

Convergence Rates

Under certain regularity conditions, we showed that

Theorem 1: If $\lambda_1, \lambda_2 \sim n^{-4/5}$, and $\gamma_0 \geq 3/2$, $\gamma_1 \geq 3/2$, then

- If f_0 is not constant, $\|\hat{f} - f_0\|_n = O_P(n^{-2/5})$.
- If f_0 is a constant, $\|\hat{f} - f_0\|_n = O_P(n^{-1/2})$.

Here $\|f\|_n^2 = \frac{1}{n}\sum_{i=1}^n f(x_i)^2$ and f_0 is the true function.

Selection Properties

Consider the second-order Sobolev space of periodic functions.

Theorem 2: Under certain regularity conditions, assume that

1. $n^{1/5}\lambda_1 w_{0j} \to \infty$ for $j \in I \backslash I_L$
2. $n^{3/20}\lambda_2 w_{1j} \to \infty$ for $j \in I \backslash I_N$,

then

(i) $\widehat{I}_L = I_L$,

(ii) $\widehat{I}_N = I_N$,

(iii) $\widehat{I}_O = I_O$

with probability tending to one as $n \to \infty$.

Comments

In order to achieve variable selection consistency and the nonparametric convergence rate simultaneously, we require that

$$\lambda_1, \lambda_2 \sim n^{-4/5}, \tag{1}$$

$$\gamma_0 > 3, \gamma_1 > 29/8 \tag{2}$$

by considering altogether Theorems 1 and 2.

Simulation Setting

Consider the following functions:

$$
\begin{aligned}
g_1(x) &= x \\
g_2(x) &= \cos(2\pi x) \\
g_3(x) &= \sin(2\pi x)/(2 - \sin(2\pi x)) \\
g_4(x) &= 0.1\sin(2\pi x) + 0.2\cos(2\pi x) + 0.3(\sin(2\pi x))^2 \\
&\quad +0.4(\cos(2\pi x))^3 + 0.5(\sin(2\pi x))^3 \\
g_5(x) &= (3x-1)^2.
\end{aligned}
$$

Attribute vector $\boldsymbol{X}$ uniformly from $[0,1]^d$

Example 1

- $Y = f(\boldsymbol{X}) + \varepsilon$ with $\varepsilon \sim N(0, \sigma^2)$
- $d = 10, 20, 40; n = 100.$
- True regression function

$$f(\boldsymbol{X}) = 3g_1(X_1) + 2g_2(X_2) + 2g_5(X_3)$$

  X_1 linear, X_2 purely nonlinear, X_3 L&N, $d - 3$ noise variables
- σ ranges $0.7 \sim 2$, giving different signal-to-noise ratios.
- We choose $\gamma_0 = 4$ and $\gamma_1 = 4$.
- Compare with additive models (SS-ANOVA implementation) for prediction performance.

Estimation Results for $d = 10$

		Additive		New	
d	σ	Train error	Test error	Train error	Test error
10	0.7	0.10(0.03)	0.60(0.05)	0.05(0.02)	0.53(0.02)
	1	0.19(0.06)	1.21(0.10)	0.10(0.04)	1.08(0.056)
	1.4	0.36(0.12)	2.35(0.19)	0.23(0.13)	2.16(0.18)
	1.6	0.46(0.15)	3.06(0.24)	0.34(0.24)	2.88(0.34)
	2	0.71(0.23)	4.76(0.38)	0.68(0.50)	4.70(0.67)

Variable Selection Results for $d = 10$

d	σ	corrlin	corrnon	corrlnn	corr0
oracle		1	1	1	7
10	0.7	1.00(0.00)	0.85(0.36)	1.00(0.00)	6.28(1.56)
	1	0.96(0.20)	0.85(0.36)	1.00(0.00)	6.16(1.59)
	1.4	0.96(0.20)	0.80(0.40)	1.00(0.00)	5.80(1.51)
	1.6	0.98(0.14)	0.71(0.46)	0.97(0.17)	5.32(1.60)
	2	0.98(0.14)	0.71(0.46)	0.97(0.17)	4.65(1.65)

Estimation Results for $d = 20$

		Additive		New	
d	σ	Train error	Test error	Train error	Test error
20	0.7	0.16(0.05)	0.73(0.09)	0.05(0.02)	0.54(0.03)
	1	0.32(0.09)	1.46(0.17)	0.11(0.05)	1.11(0.07)
	1.4	0.61(0.18)	2.81(0.31)	0.30(0.20)	2.30(0.28)
	1.6	0.79(0.23)	3.65(0.40)	0.47(0.34)	3.11(0.47)
	2	1.21(0.37)	5.66(0.63)	0.98(0.68)	5.18(0.92)

Variable Selection Results for $d = 20$

d	σ	corrlin	corrnon	corrlnn	corr0
oracle		1	1	1	17
20	0.7	0.92(0.27)	0.90(0.30)	1.00(0.00)	16.28(2.12)
	1	0.92(0.27)	0.91(0.29)	1.00(0.00)	16.06(1.88)
	1.4	0.94(0.24)	0.80(0.40)	1.00(0.00)	14.14(2.91)
	1.6	0.95(0.22)	0.76(0.43)	0.99(0.10)	13.39(2.86)
	2	0.98(0.14)	0.61(0.49)	0.97(0.17)	11.67(2.71)

Estimation Results for $d = 40$

		Additive		New	
d	σ	Train error	Test error	Train error	Test error
40	0.7	0.27(0.05)	1.09(0.19)	0.05(0.02)	0.58(0.03)
	1	0.53(0.10)	2.18(0.38)	0.14(0.07)	1.21(0.11)
	1.4	1.02(0.20)	4.19(0.70)	0.68(0.55)	3.06(0.92)
	1.6	1.33(0.26)	5.45(0.95)	0.89(0.58)	4.07(1.10)
	2	2.07(0.42)	8.48(1.45)	1.89(0.96)	7.38(2.05)

Variable Selection Results for $d = 40$

d	σ	corrlin	corrnon	corrlnn	corr0
oracle		1	1	1	37
40	0.7	0.99(0.10)	0.93(0.26)	1.00(0.00)	36.36(1.28)
	1	1.00(0.00)	0.91(0.29)	1.00(0.00)	35.09(1.67)
	1.4	0.99(0.10)	0.78(0.42)	0.97(0.17)	30.42(2.57)
	1.6	0.99(0.10)	0.73(0.45)	0.96(0.20)	28.06(2.95)
	2	0.97(0.17)	0.56(0.50)	0.81(0.39)	23.83(3.41)

Example 2

- True regression function

$$
\begin{aligned}
f(\boldsymbol{X}) \quad = \quad & 3g_1(X_1) - 4g_1(X_2) + 2g_1(X_3) + 2g_2(X_4) \\
& +3g_3(X_5) + (5g_4(X_6) + 2g_1(X_6)) \\
& +2g_5(X_7) + \epsilon,
\end{aligned}
$$

  Linear: X_1, X_2, X_3, Purely Nonlinear: X_4, X_5, Linear & Nonlinear: X_6, X_7.

- $d = 20, 40; n = 100.$
- $d - 7$ noise variables
- σ ranges $0.7 \sim 2$.

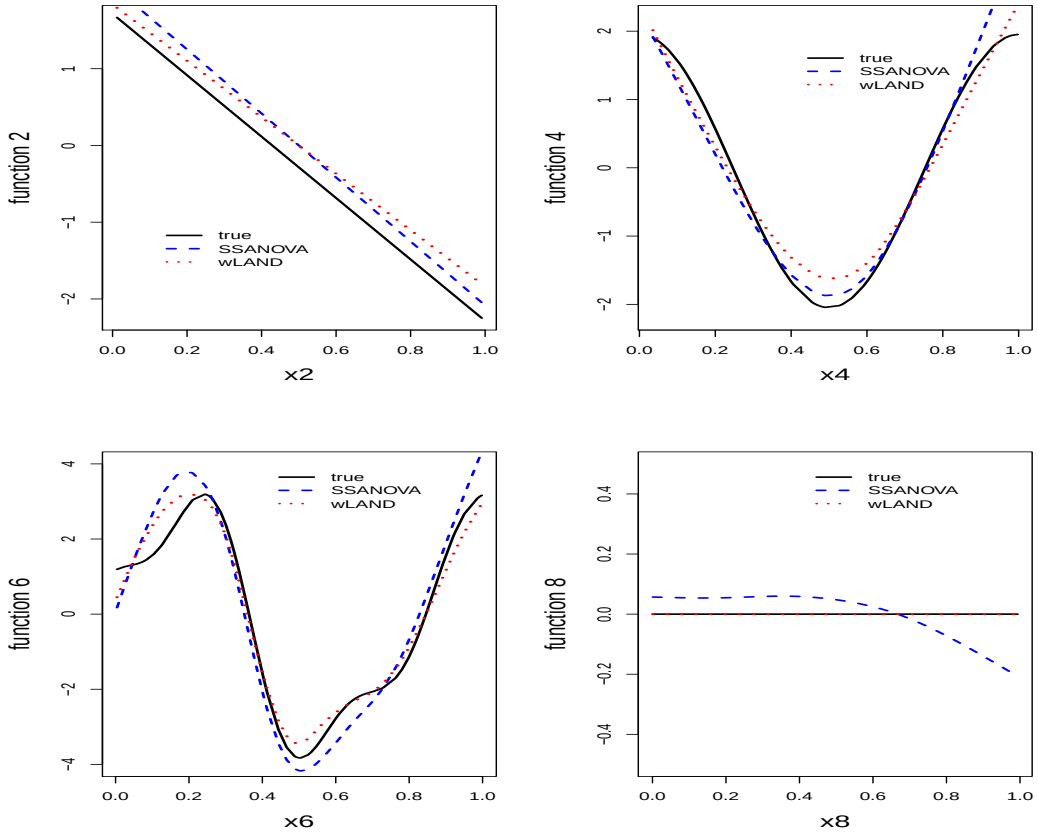
Estimation Results for Example 2

		Additive		New	
d	σ	Train error	Test error	Train error	Test error
20	0.7	0.29 (0.07)	0.90 (0.12)	0.29 (0.06)	0.68 (0.08)
	1	0.61 (0.14)	1.75 (0.21)	0.60 (0.14)	1.43 (0.25)
	1.5	1.40 (0.34)	3.83 (0.57)	1.48(0.42)	3.46 (0.70)
	2	2.55 (0.62)	6.73 (0.91)	2.85 (1.27)	6.51 (1.48)
40	0.7	0.41 (0.22)	2.19 (1.06)	0.61 (0.35)	1.21 (0.66)
	1	0.73 (0.29)	3.80 (1.14)	1.15 (0.49)	2.38 (1.07)
	1.5	1.39 (0.43)	7.05(1.68)	2.23 (0.71)	5.10 (1.49)
	2	2.22 (0.65)	11.43 (2.67)	3.75 (2.34)	8.86 (2.54)

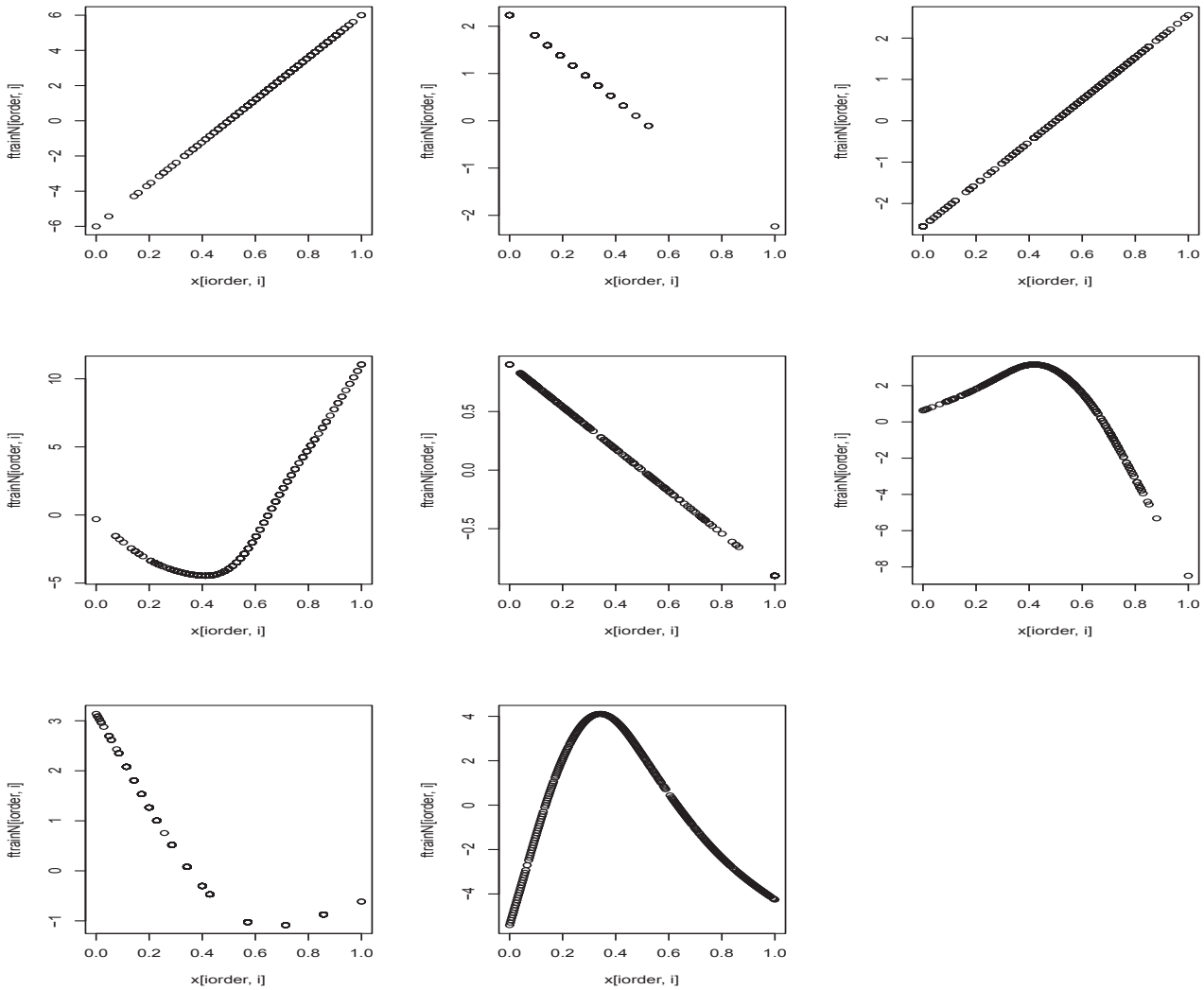
Variable Selection Results for Example 2

d	σ	corrlin	corrnon	corrlnn	corr0
oracle		3	2	2	$d-7$
20	0.7	3.00(0.00)	1.76(0.47)	1.98(0.14)	12.40(1.89)
	1	3.00(0.00)	1.65(0.61)	1.95(0.22)	11.88(2.61)
	1.5	2.87(0.34)	1.53(0.61)	1.89(0.31)	11.63(2.30)
	2	2.71(0.57)	1.23(0.68)	1.66(0.52)	10.51(2.53)
40	0.7	3.00(0.00)	1.58(0.52)	1.95(0.22)	32.07(1.58)
	1	2.94(0.24)	1.35(0.56)	1.75(0.46)	31.21(2.28)
	1.5	2.80(0.40)	1.11(0.58)	1.43(0.61)	29.48(2.58)
	2	2.55(0.77)	0.83(0.59)	1.20(0.77)	27.42(3.15)

Fitted Functions for Example 2



Results for Ozone Concentration Data



Summary

We develop a new regularization method which

- can distinguish linear from nonlinear covariates
- can distinguish important from uninformative variables
- have nice theoretical properties

Future work

- Further improve computation efficiency and tuning process