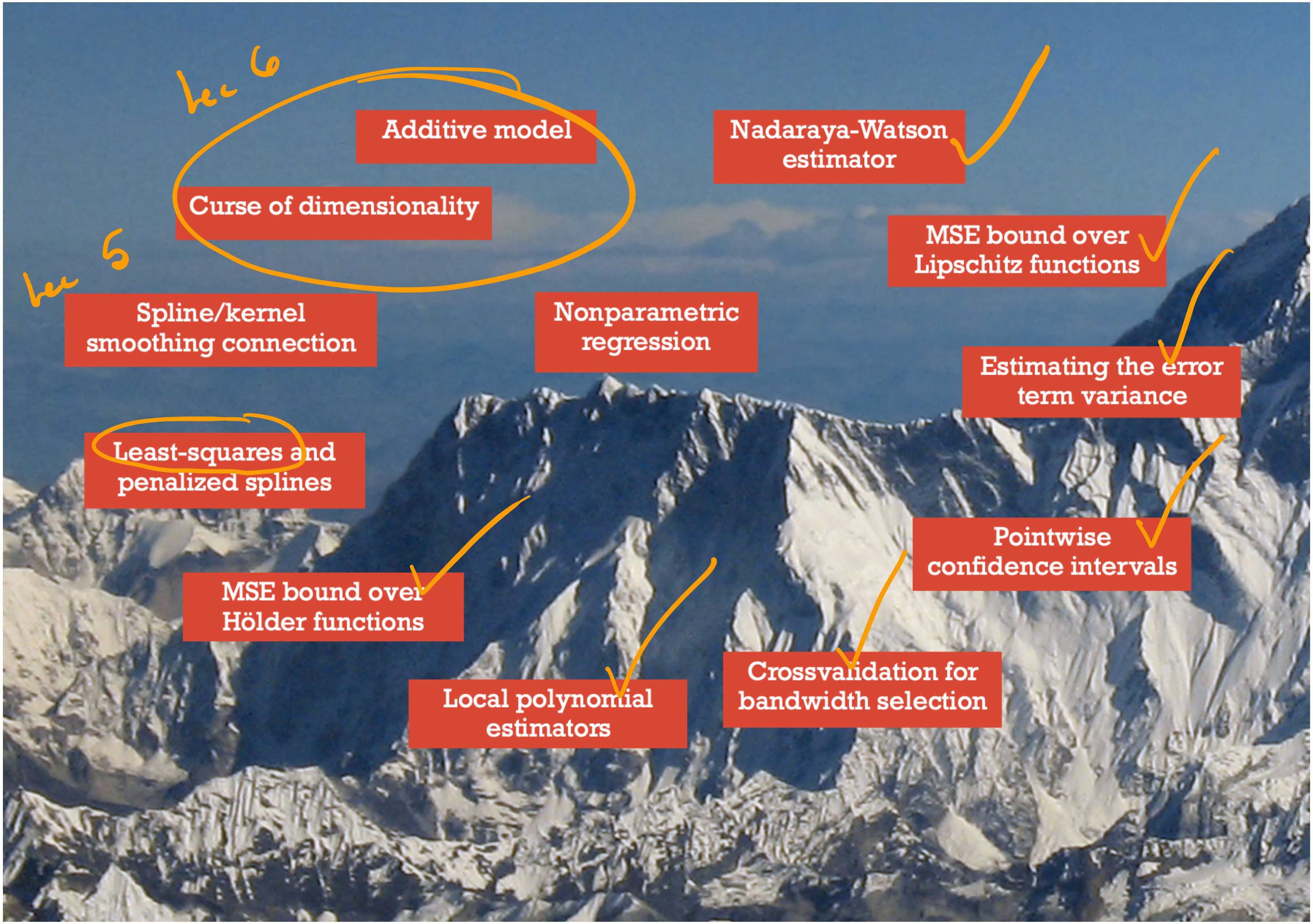
STAT 824 sp 2025 Lec 05 slides

Nonparametric regression: Splines

Karl Gregory

University of South Carolina

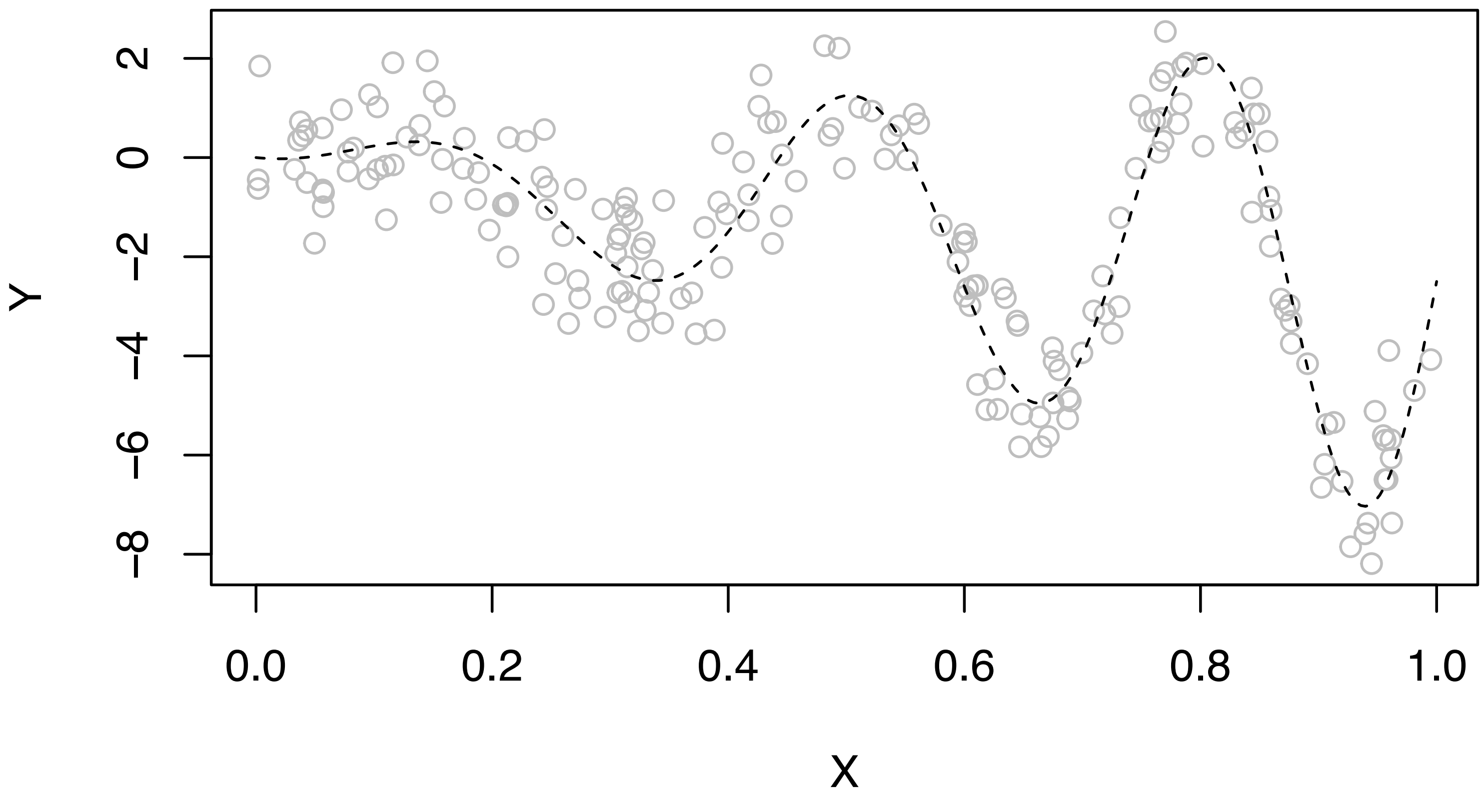
These slides are an instructional aid; their sole purpose is to display, during the lecture, definitions, plots, results, etc. which take too much time to write by hand on the blackboard. They are not intended to explain or expound on any material.


Additive model
Curse of dimensionality
Nadaraya-Watson estimator
MSE bound over Lipschitz functions
Spline/kernel smoothing connection
Nonparametric regression
Estimating the error term variance
Least-squares and penalized splines
Pointwise confidence intervals
MSE bound over Hölder functions
Local polynomial estimators
Crossvalidation for bandwidth selection
Lec 6
Lec 5

Let $(X_1, Y_1), \ldots, (X_n, Y_n)$ be indep. realizations of $(X, Y) \in [0,1] \times \mathbb{R}$, where

$$Y = m(X) + \varepsilon, \qquad \text{for some } m : [0,1] \to \mathbb{R},$$

where ε is independent of X with $\mathbb{E}\varepsilon = 0$ and $\mathbb{E}\varepsilon^2 = \sigma^2$.



The function plotted is $m(x) = 5x \cdot \sin(2\pi(1+x)^2) - (5/2)x$.

Sieve estimation

Given a set of building-block functions $b_1, \ldots, b_{d_n} : [0,1] \to \mathbb{R}$, assume

$$m(x) \approx \sum_{k=1}^{d_n} \alpha_k b_k(x) \quad \text{for some} \quad \alpha_1, \ldots, \alpha_{d_n}.$$

Estimate $\alpha_1, \ldots, \alpha_{d_n}$ with least squares to get $\hat{m}_n(x) = \sum_{k=1}^{d_n} \hat{\alpha}_k b_k(x)$.

- As $n \to \infty$, let $d_n \to \infty$ so that the approximation improves.
- Quality of approximation depends on
  1. the type and number of basis functions.
  2. the smoothness of the true function m.
- There will always be some approximation bias.

$$Y_i = m(x_i) + \varepsilon_i$$

$$m(x_i) \approx \sum_{k=1}^{d} \alpha_k b_k(x_i)$$

$$Y_i \approx \sum_{k=1}^{d} \alpha_k b_k(x_i) + \varepsilon_i$$

Write

$$\underset{n\times 1}{\underset{\sim}{Y}} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}, \quad \underset{n\times d}{B} = \begin{bmatrix} b_1(x_1) & \cdots & b_d(x_1) \\ \vdots & & \vdots \\ b_1(x_n) & & b_d(x_n) \end{bmatrix}, \quad \underset{\sim}{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{bmatrix} \quad \underset{\sim}{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

then

$$\underset{\sim}{Y} = B \underset{\sim}{\alpha} + \underset{\sim}{\varepsilon}$$

So estimate $\underset{\sim}{\alpha}$ as

$$\hat{\alpha} = \left(B^T B\right)^{-1} B^T \underset{\sim}{Y} = \left(\hat{\alpha}_1, \ldots, \hat{\alpha}_d\right)^T$$

Then

$$\hat{m}_n(x) = \sum_{k=1}^{d} \hat{\alpha}_k b_k(x).$$

A non-parametric least squares estimator

For a set of basis functions $b_1, \dots, b_{d_n} : [0,1] \to \mathbb{R}$, let

$$\mathcal{B}_n = \{m : m(x) = \textstyle\sum_{k=1}^{d_n} \alpha_k b_k(x), \alpha_1, \dots, \alpha_{d_n} \in \mathbb{R}\}.$$

Given $(X_1, Y_1), \dots, (X_n, Y_n)$, the least squares estimator of m in $\mathcal{B}_n$ is given by

$$\hat{m}_n = \operatorname*{argmin}_{g \in \mathcal{B}_n} \sum_{i=1}^{n} [Y_i - g(X_i)]^2.$$

Exercise: Let $\mathbf{Y} = (Y_1, \dots, Y_n)^T$ and define the matrix

$$\mathbf{B} = (b_k(X_i))_{1 \le i \le n, 1 \le k \le d_n}.$$

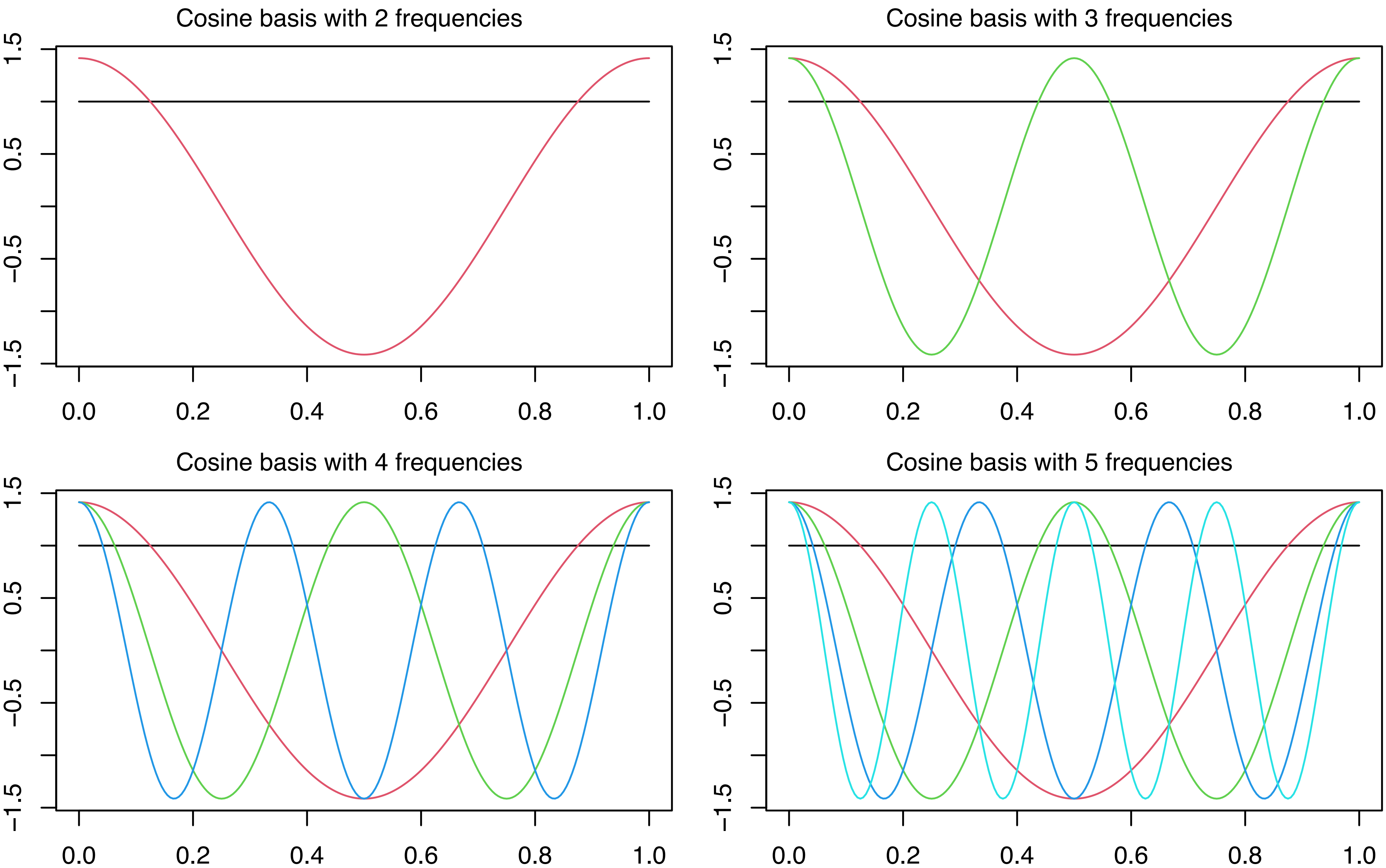
Show that $\hat{m}_n(x) = \mathbf{b}_x^T \hat{\alpha}$, where

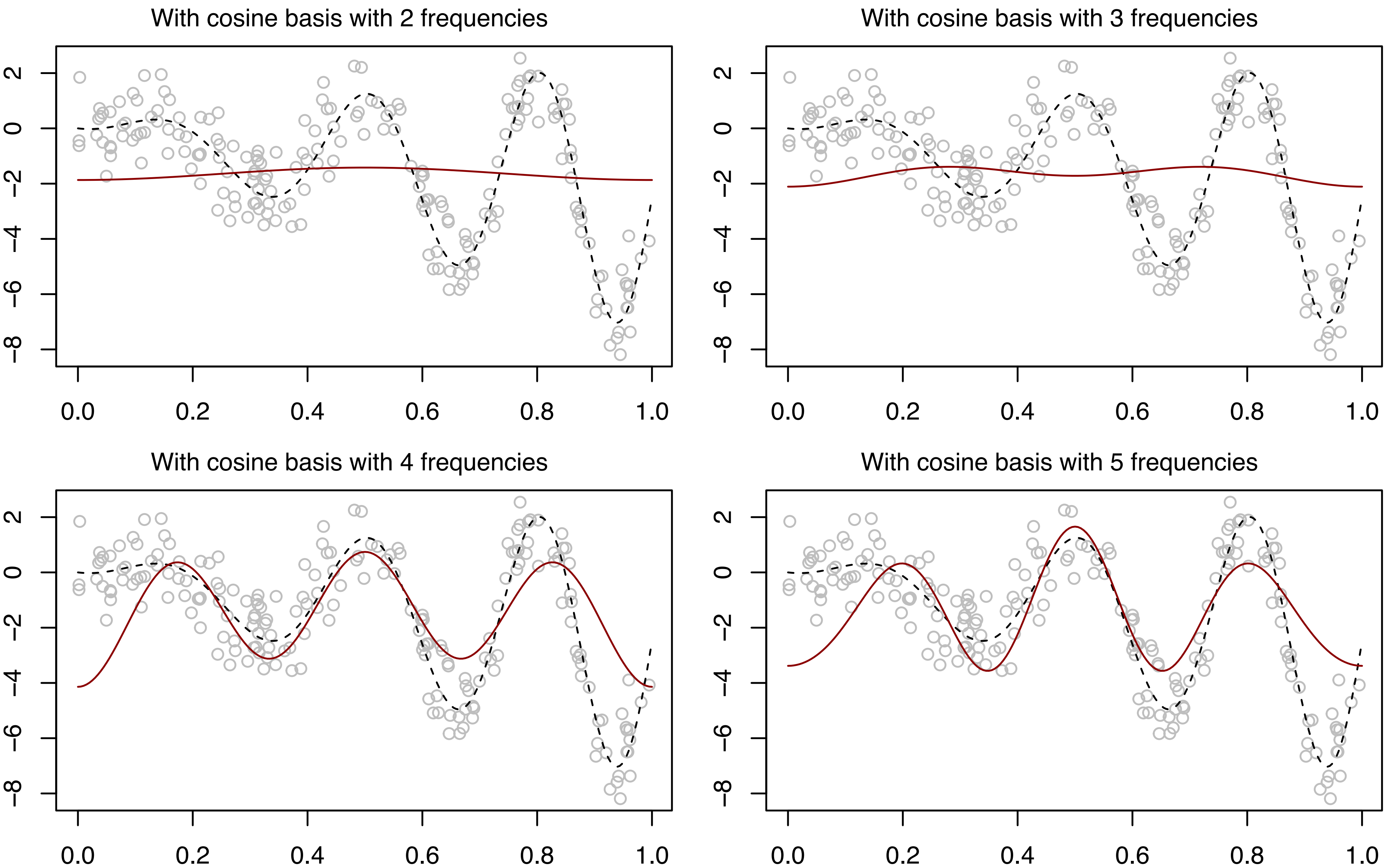
$$\hat{\alpha} = \left(\mathbf{B}^T\mathbf{B}\right)^{-1} \mathbf{B}^T\mathbf{Y} \quad \text{and} \quad \mathbf{b}_x = (b_1(x), \dots, b_{d_n}(x))^T.$$

Cosine basis

The cosine basis on $[0, 1]$ is the set of functions

$$b_1(x) = 1, \quad b_k = \sqrt{2}\cos(2\pi(k-1)x), \quad \text{for } k = 2, 3, \ldots$$


Cosine basis with 2 frequencies
Cosine basis with 3 frequencies
Cosine basis with 4 frequencies
Cosine basis with 5 frequencies
1.5
0.5
−0.5
−1.5
0.0
0.2
0.4
0.6
0.8
1.0


With cosine basis with 2 frequencies
With cosine basis with 3 frequencies
With cosine basis with 4 frequencies
With cosine basis with 5 frequencies
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

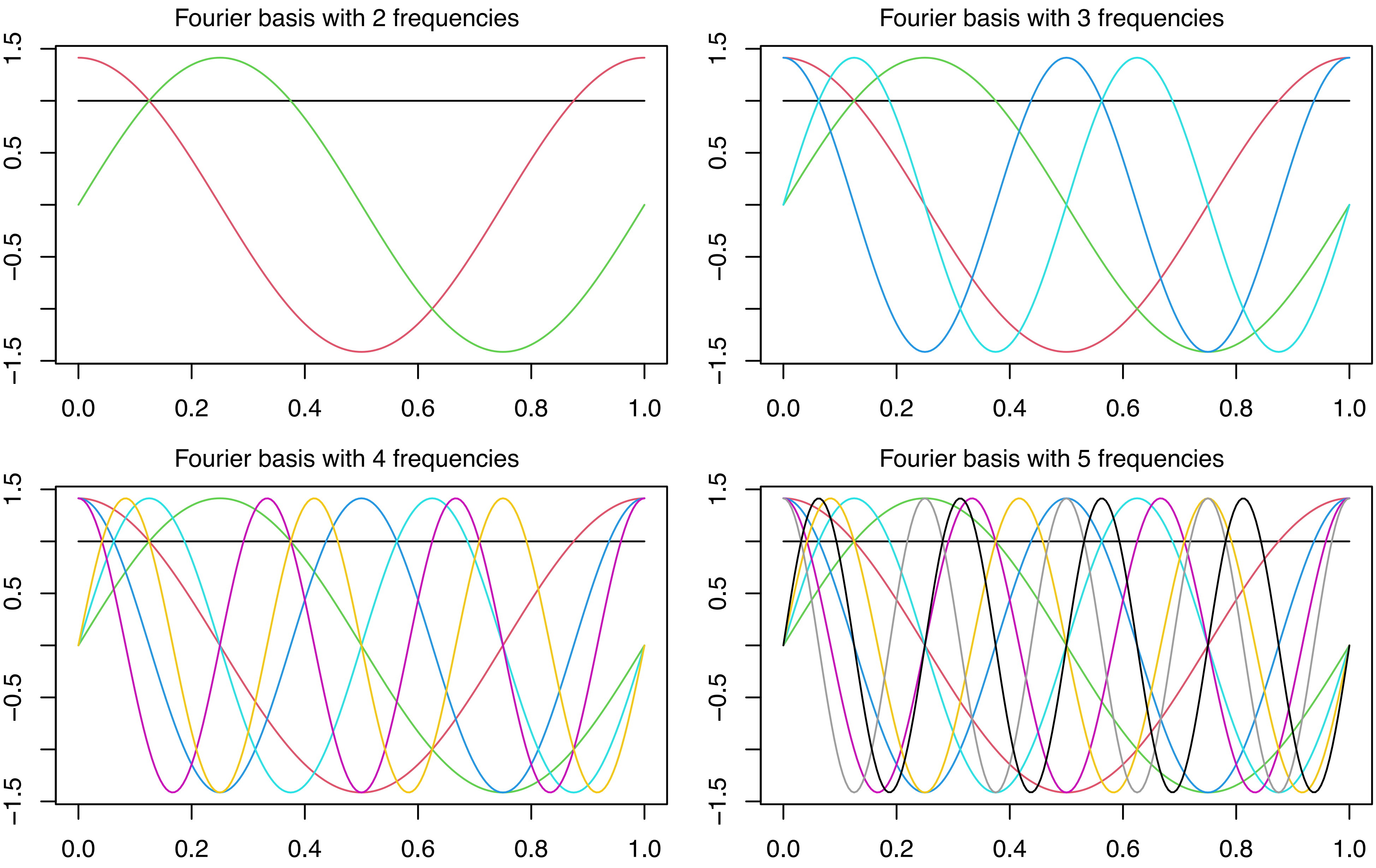
Fourier basis

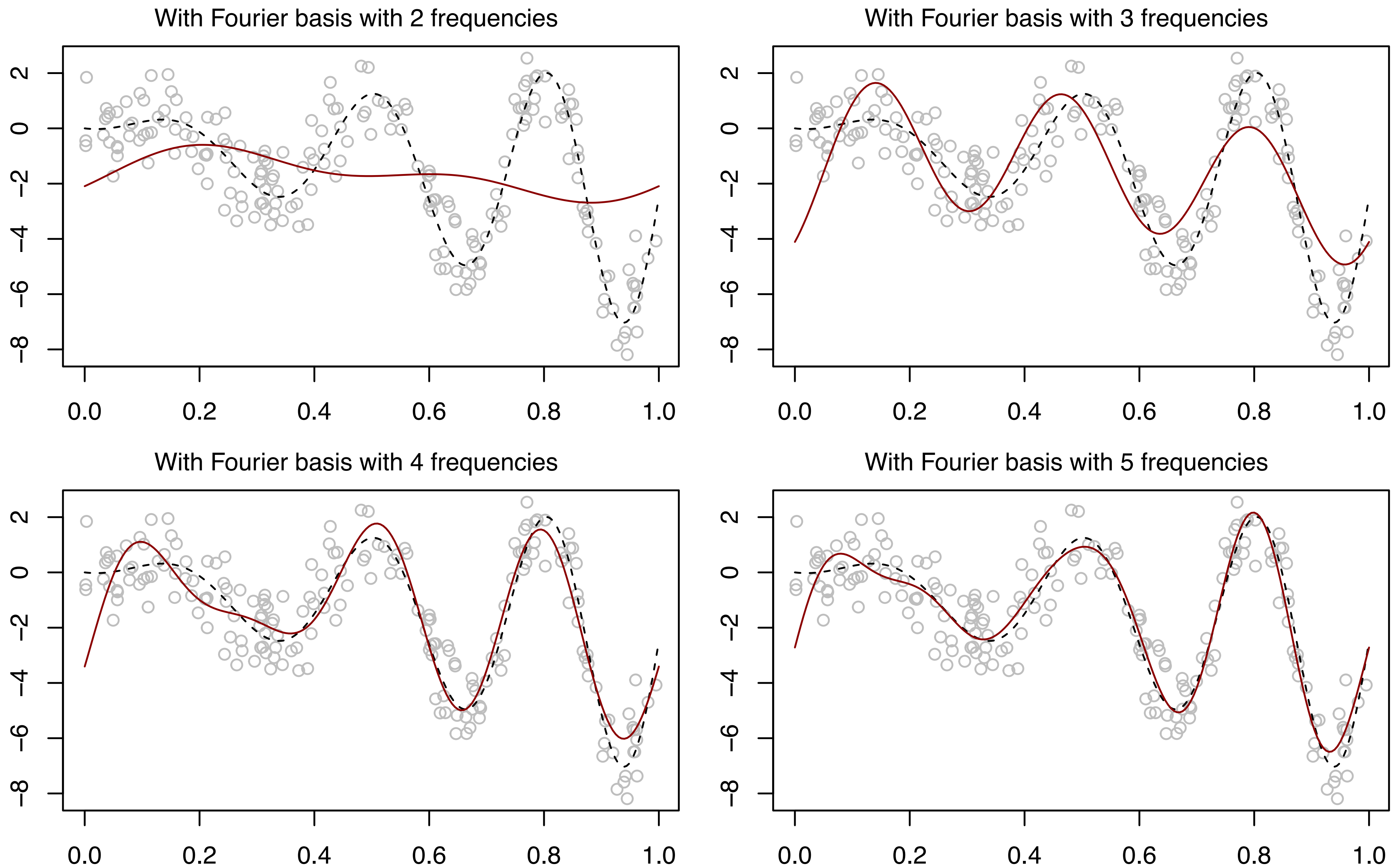
The Fourier basis on $[0, 1]$ is the set of functions given by

$$b_1(x) = 1, \quad b_{2j}(x) = \sqrt{2}\cos(2\pi jx), \quad b_{2j+1}(x) = \sqrt{2}\sin(2\pi jx)$$

for $j = 1, 2, \ldots$

$$\int_0^1 b_k(x)\, b_{k'}(x)\, dx = \begin{cases} 1 & k = k' \\ 0 & k \neq k' \end{cases}$$


Fourier basis with 2 frequencies
Fourier basis with 3 frequencies
Fourier basis with 4 frequencies
Fourier basis with 5 frequencies
1.5
0.5
−0.5
−1.5
0.0
0.2
0.4
0.6
0.8
1.0


With Fourier basis with 2 frequencies
With Fourier basis with 3 frequencies
With Fourier basis with 4 frequencies
With Fourier basis with 5 frequencies
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

Legendre polynomial basis

Given a set of knots $0 = u_0 < u_1 < \cdots < u_K = 1$, and an order r, set

$$b_{(j-1)(r+1)+\ell+1}(x) = P_\ell\Big(2\big(\frac{x - u_{j-1}}{u_j - u_{j-1}}\big) - 1\Big)$$

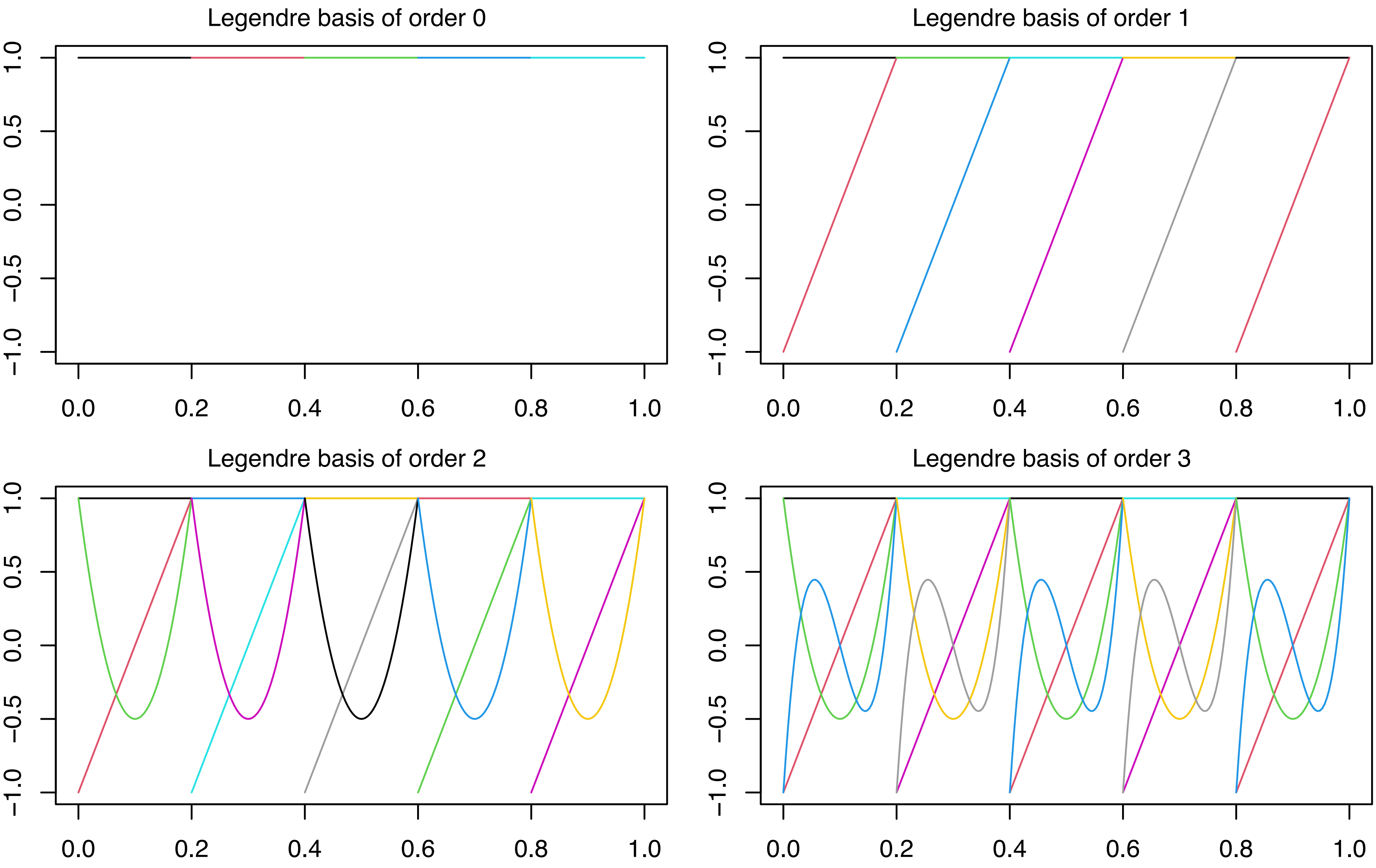
for $j = 1, \dots, K$ and $\ell = 0, 1, \dots, r$, where

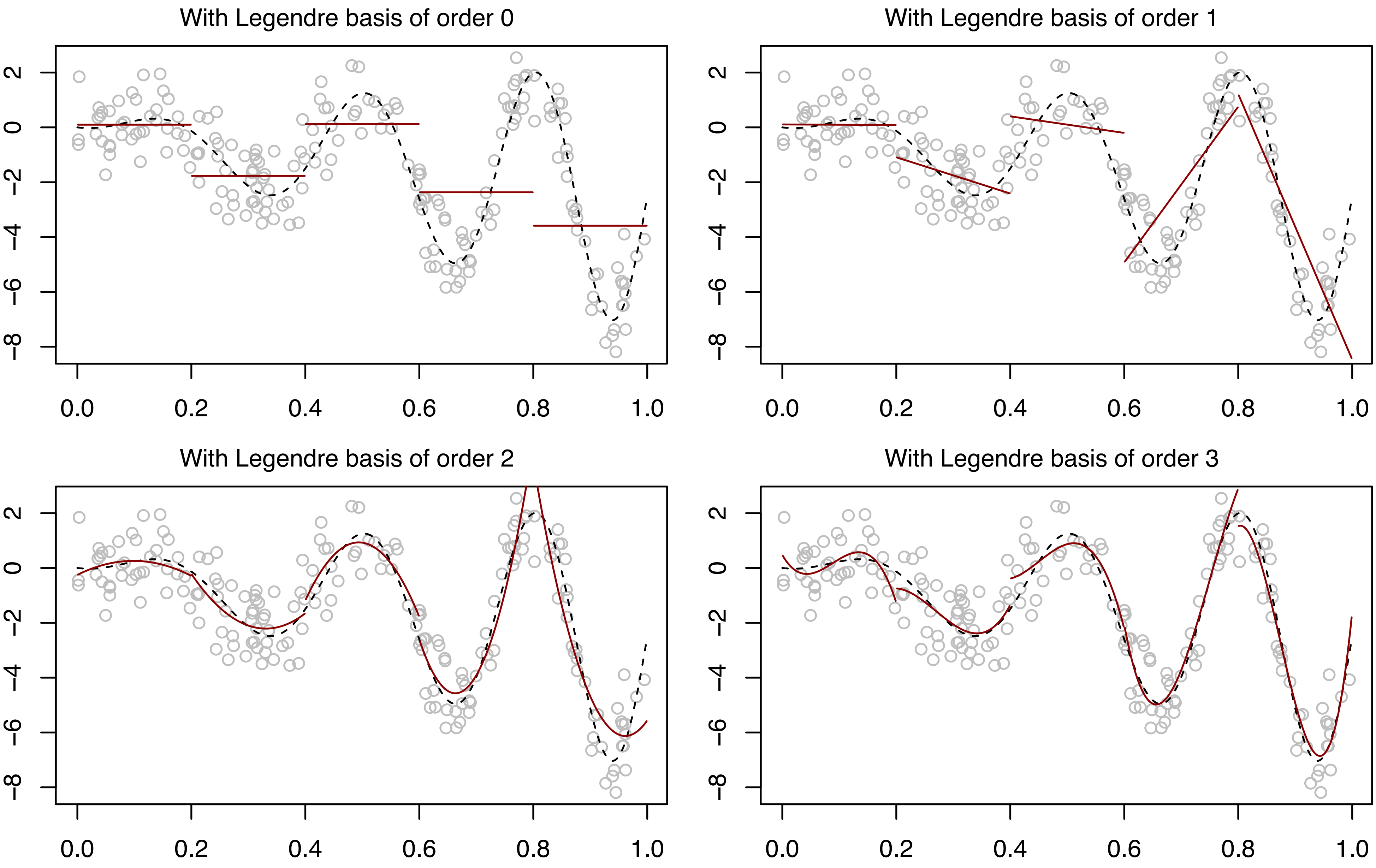
$$P_\ell(z) = \frac{1}{2^\ell \ell!}\frac{d^\ell}{dz^\ell}(z^2 - 1)^\ell, \quad z \in (-1, 1).$$

One obtains

$$P_0(z) = 1, \quad P_1(z) = z, \quad P_2(z) = \frac{1}{2}(3z^2 - 1), \quad P_3(z) = \frac{1}{2}(5z^3 - 3z).$$

One can make the number of knots K grow as $n \to \infty$.


Legendre basis of order 0
Legendre basis of order 1
Legendre basis of order 2
Legendre basis of order 3
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0


With Legendre basis of order 0
With Legendre basis of order 1
With Legendre basis of order 2
With Legendre basis of order 3
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

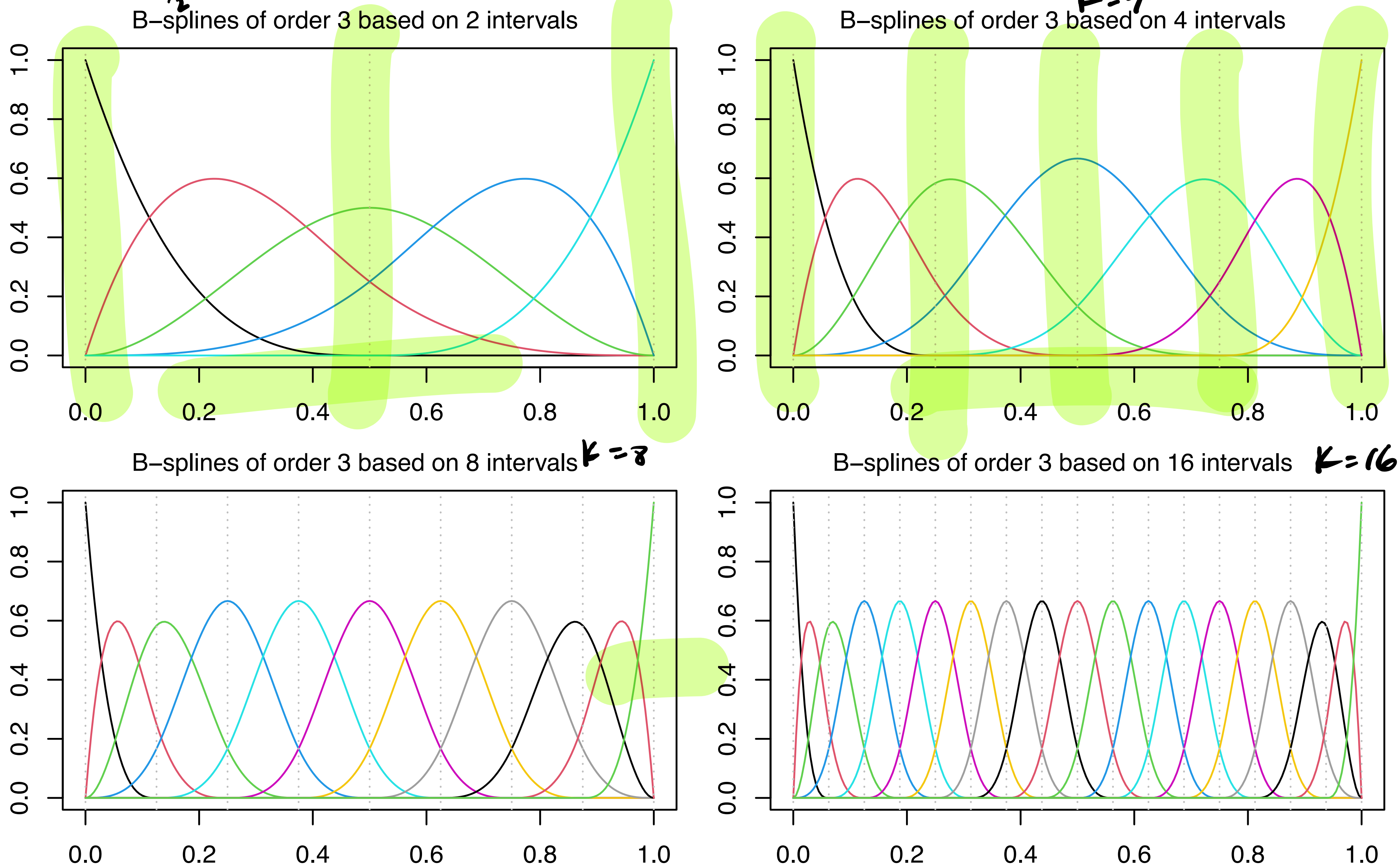
Cubic B-spline basis

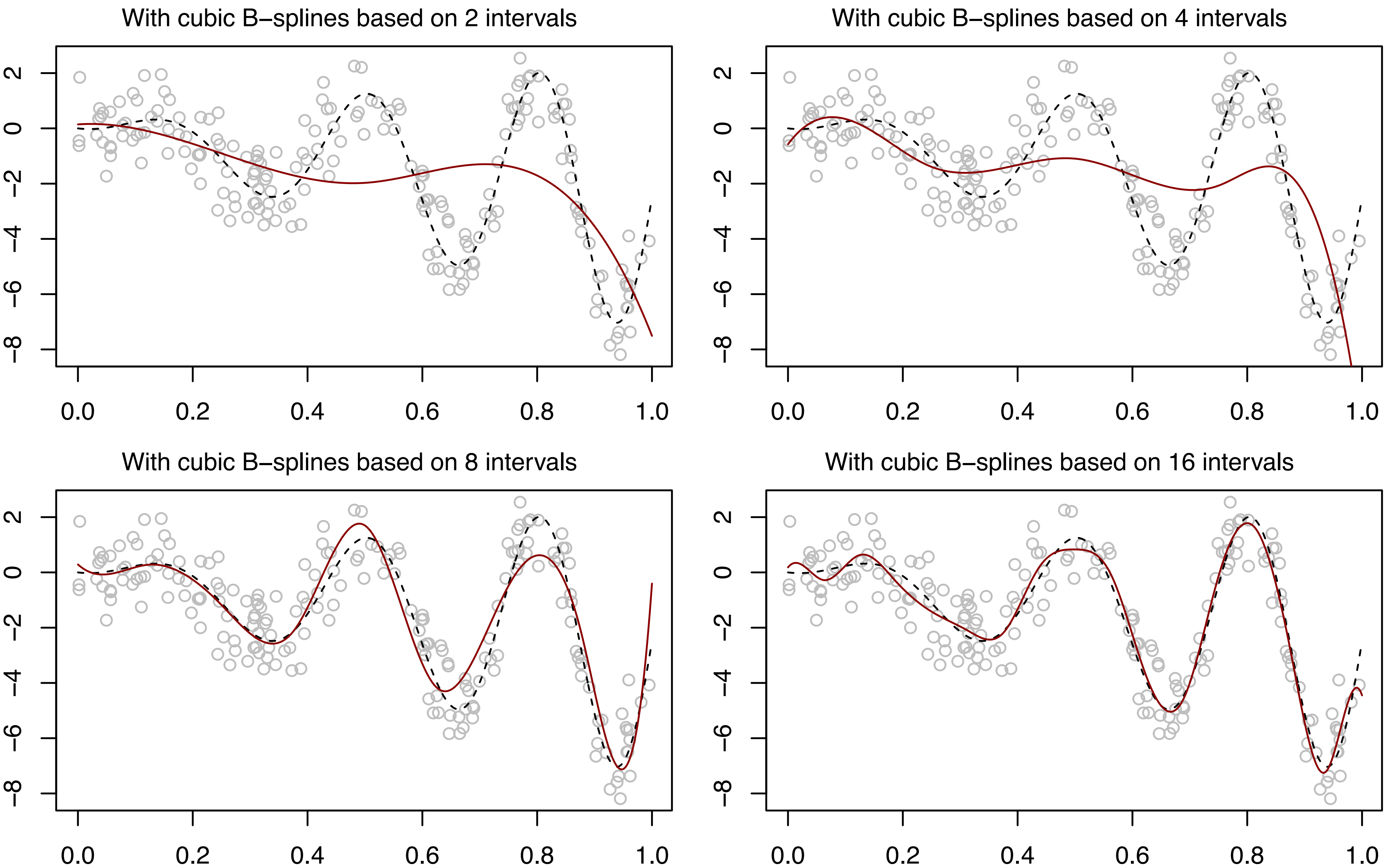
Will define shortly. . . look at some pictures first.

Make K intervals

$0 = u_0 < u_1 < u_2 = 1$ $\quad K = 2$ $\quad$ ($u_1 = 1/2$)

Knots: $0 = u_0 < u_1 < \dots < u_{K-1} < u_K = 1$




With cubic B−splines based on 2 intervals
With cubic B−splines based on 4 intervals
With cubic B−splines based on 8 intervals
With cubic B−splines based on 16 intervals
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

B-splines: Cox-deBoor recursion formula

r = order

For a non-decreasing set of knots $0 = u_0 \leq u_1 \leq \cdots \leq u_K = 1$, let

$$N_{k,0}(u) = \begin{cases} 1, & u_k \leq u < u_{k+1} \\ 0, & \text{otherwise} \end{cases} \quad \text{for } k = 0, \ldots, K-1,$$

r=0

and

$$N_{k,r}(u) = \frac{u - u_k}{u_{k+r} - u_k} N_{k,r-1}(u) + \frac{u_{k+r+1} - u}{u_{k+r+1} - u_{k+1}} N_{k+1,r-1}(u)$$

for $k = 0, \ldots, K - r - 1$. These functions are called *B-splines*.

Can compute row vector $\mathbf{N}_r(x) = (N_{0,r}(x), \ldots, N_{K-r-1,r}(x))$, $x \in [0, 1]$, with

```
splineDesign(knots = u, x = x, ord = r + 1)
```

Require `splines` package.

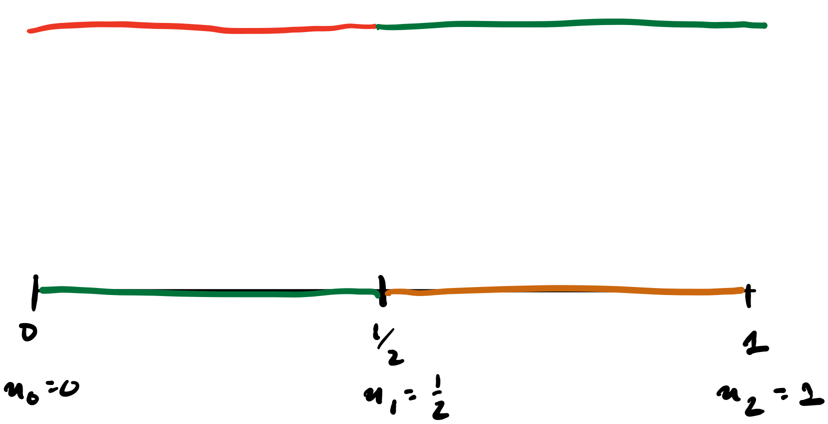
Exercise: Show construction of $N_{0,1}$ based on knots $(u_0, u_1, u_2) = (0, 1/2, 1)$.

For each k, r build $N_{k,r}(u)$.

$$N_{0,0}(u) = \begin{cases} 1 & u_0 \le u < u_1 \\ 0 & \text{o.w} \end{cases} = \begin{cases} 1 & 0 \le u < 1/2 \\ 0 & \text{o.w.} \end{cases}$$

$K = 2$

$$N_{1,0}(u) = \begin{cases} 1 & u_1 \le u < u_2 \\ 0 & \text{o.w.} \end{cases} = \begin{cases} 1 & 1/2 \le u < 1 \\ 0 & \text{o.w.} \end{cases}$$



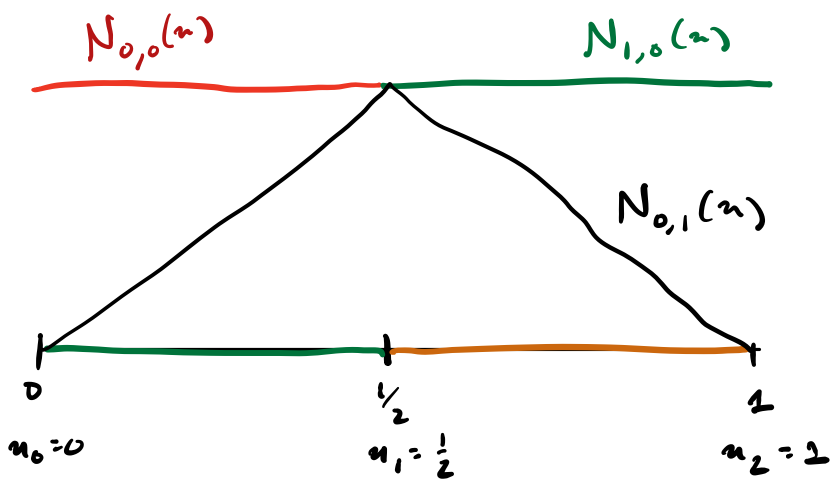
$$N_{k,r}(u) = \frac{u - u_k}{u_{k+r} - u_k} N_{k,r-1}(u) + \frac{u_{k+r+1} - u}{u_{k+r+1} - u_{k+1}} N_{k+1,r-1}(u)$$

$k=0,\ r=1$

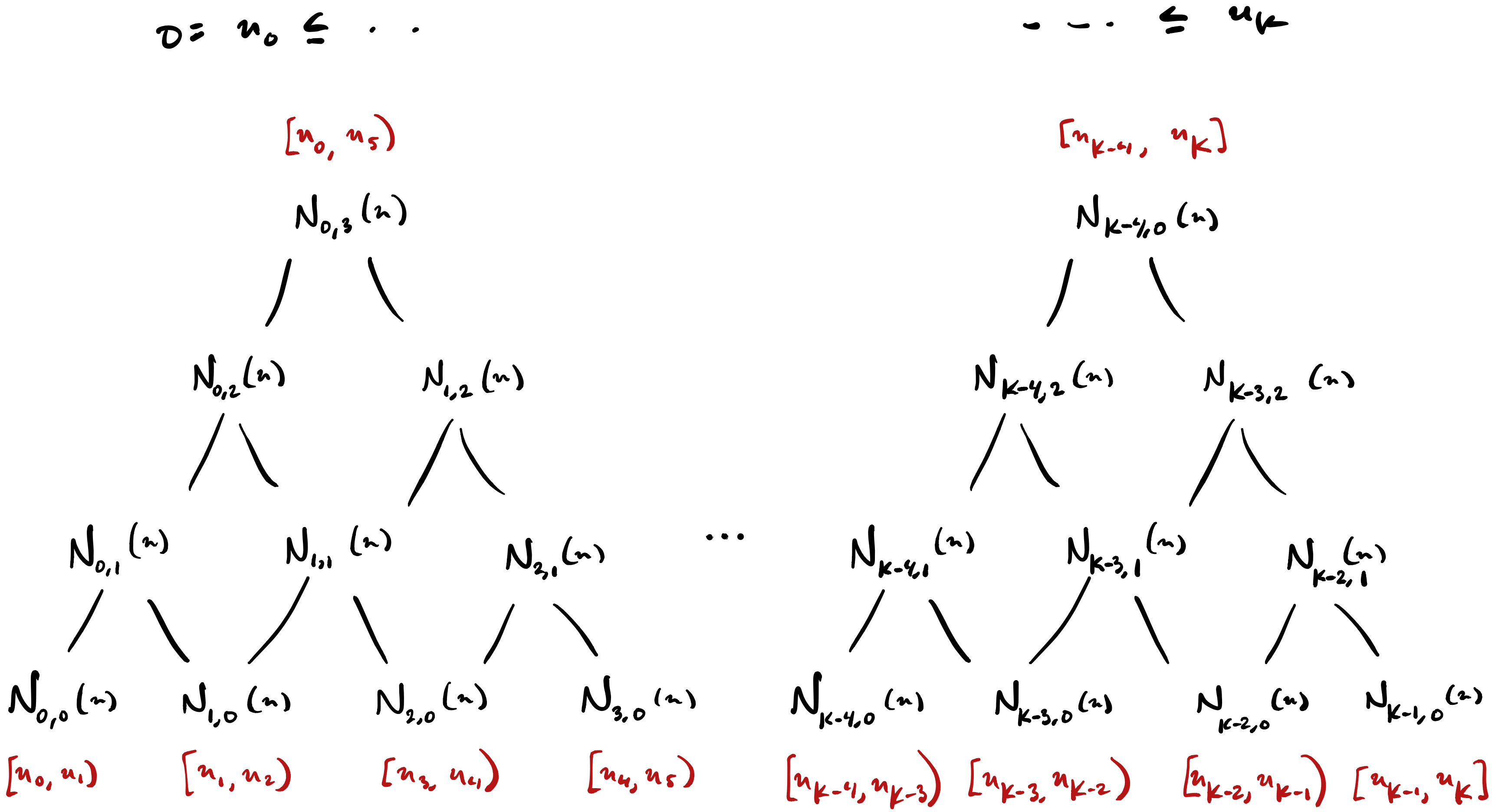
$$N_{0,1}(u) = \frac{u - u_0}{u_1 - u_0} N_{0,0}(u) + \frac{u_2 - u}{u_2 - u_1} N_{1,0}(u)$$

$$= \frac{x}{1/2 - 0} \mathbb{1}(0 \le x < 1/2) + \frac{1-x}{1 - 1/2} \mathbb{1}(1/2 \le x < 1)$$

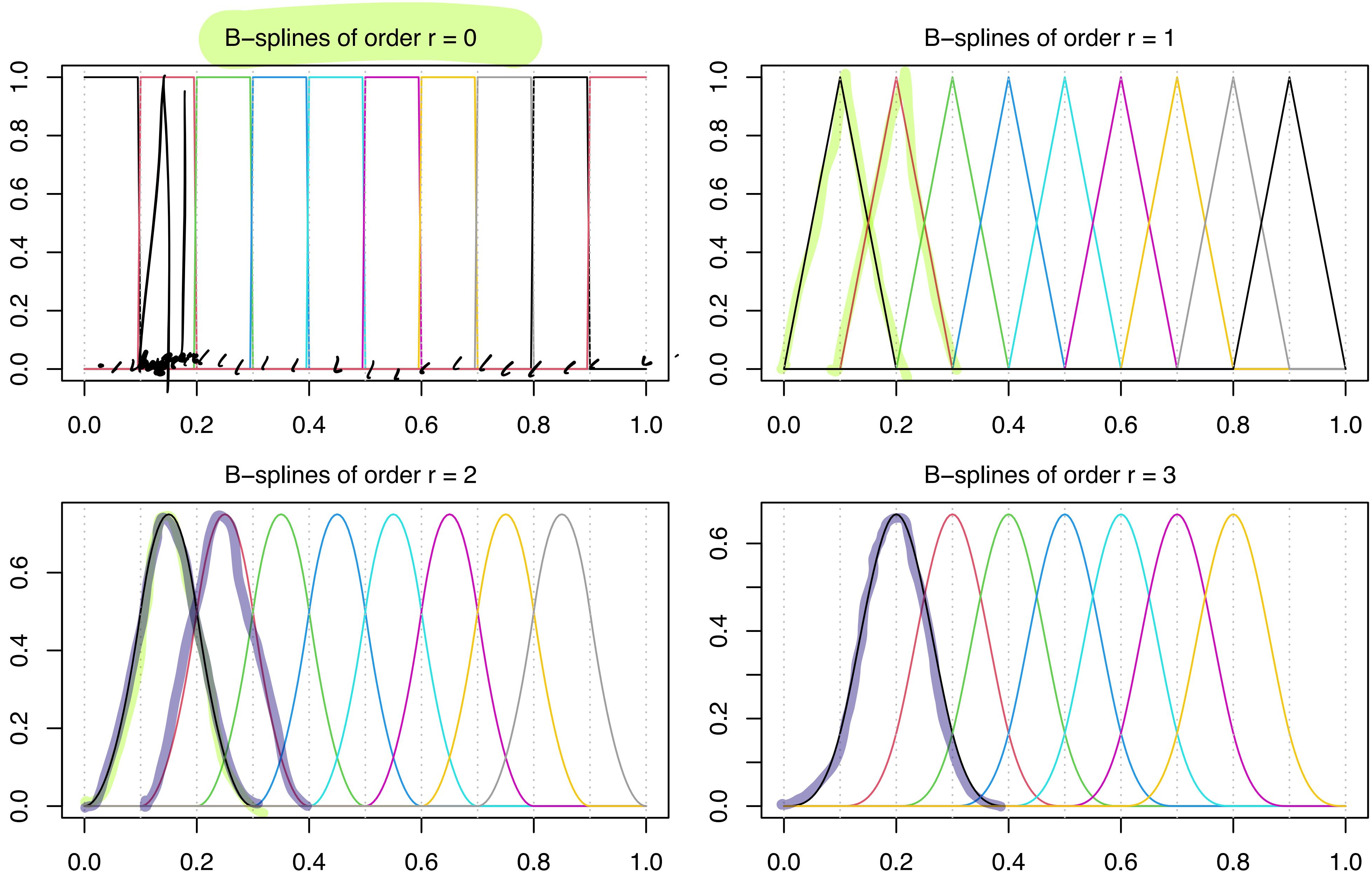
$$= 2x \, \mathbb{1}(0 \le x < \tfrac{1}{2}) + 2(1-x) \, \mathbb{1}(\tfrac{1}{2} \le x < 1)$$



The Cox-deBoor recursion has a structure like this:



Exercise: Show construction of $N_{0,1}$ based on knots $(u_0, u_1, u_2) = (0, 1/2, 1)$.


B−splines of order r = 0
B−splines of order r = 1
B−splines of order r = 2
B−splines of order r = 3
0.0
0.2
0.4
0.6
0.8
1.0

To handle boundary issues, a convention is to include the end knots $r+1$ times:

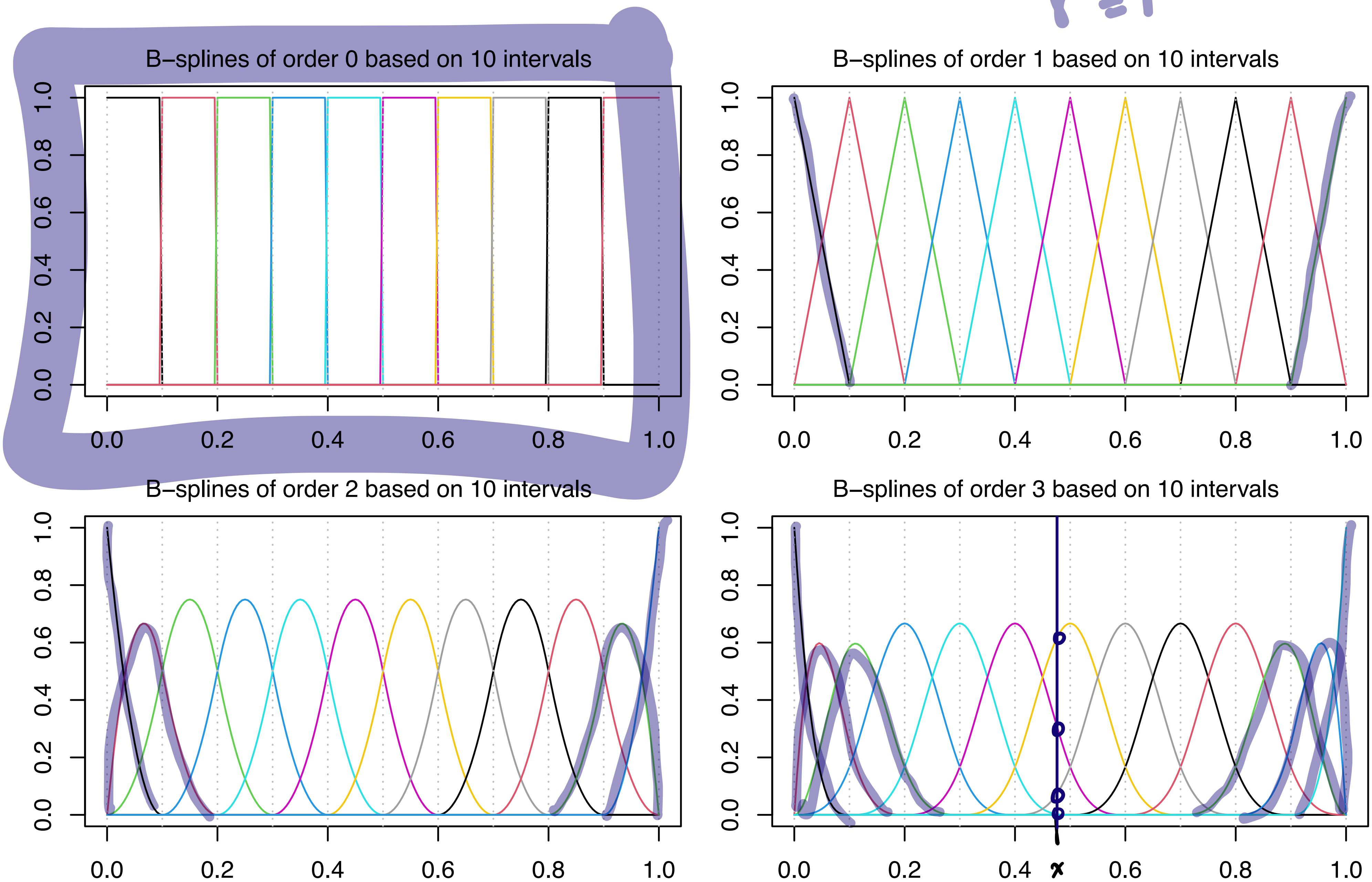
$$0 = u_{-r} = \cdots = u_0 < u_1 < \cdots < u_K = \cdots = u_{K+r}$$

This results in $K+r$ basis functions when $[0,1]$ is subdivided into K intervals.

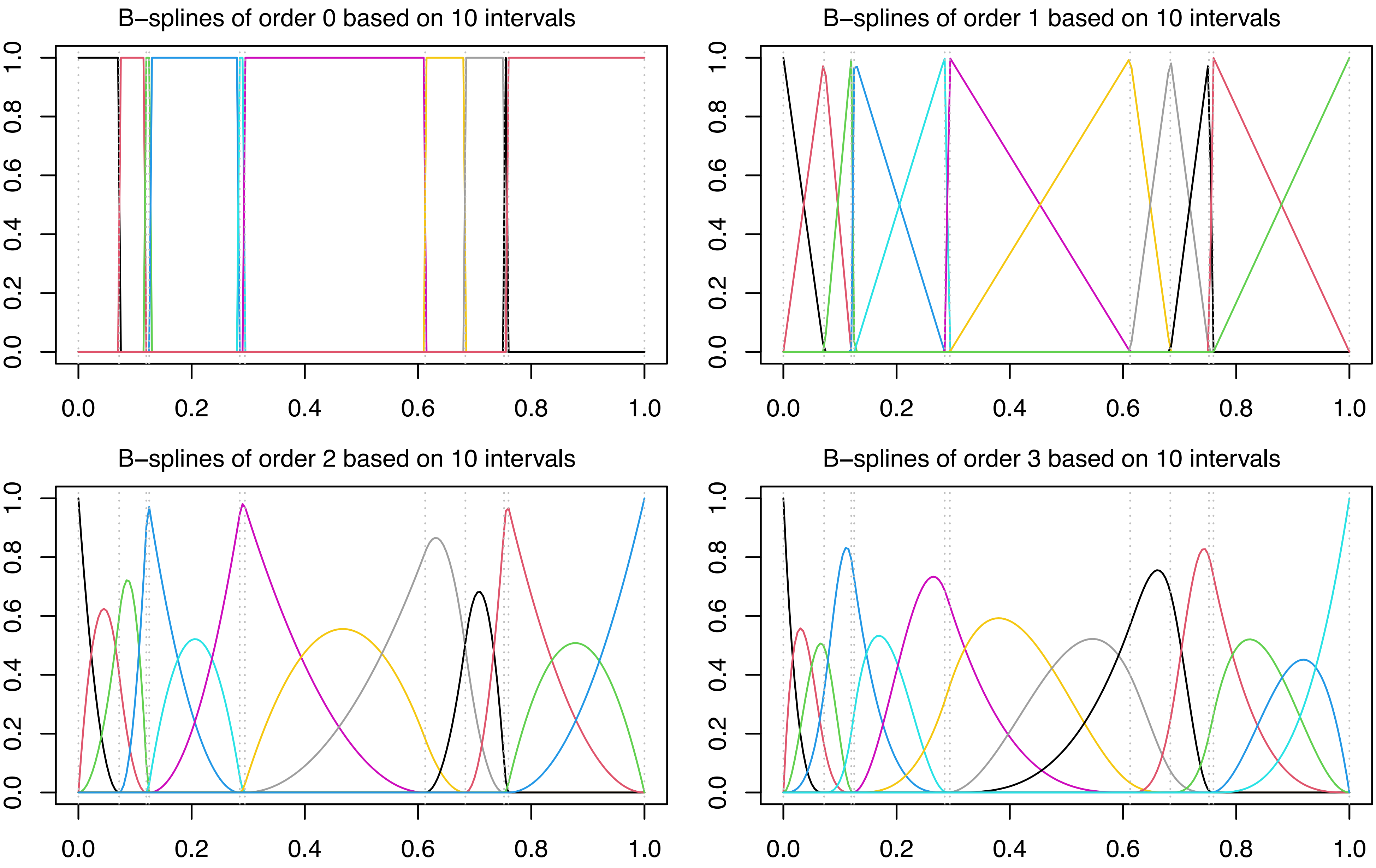
Exercise: Make beautiful plots of B-spline functions of order $r = 0,1,2,3$ in R

1. with equally spaced knots.
2. with unequally spaced knots.

r = 1



$\underset{\sim}{b}_x = (b_1(x), \ldots, b_d(x))^T$ ← has most entries = 0
Entries sum to 1!


B−splines of order 0 based on 10 intervals
B−splines of order 1 based on 10 intervals
B−splines of order 2 based on 10 intervals
B−splines of order 3 based on 10 intervals
0.0
0.2
0.4
0.6
0.8
1.0

library(splines)

Replicating boundary knots r times results in $d_n = K_n + r$ basis functions.

For $X_1, \dots, X_n$, we can obtain the $n \times d_n$ design matrix $\mathbf{B}$ with

`splineDesign(knots = u, x = X, ord = r + 1)`,

where `X` is a vector containing the values $X_1, \dots, X_n$.

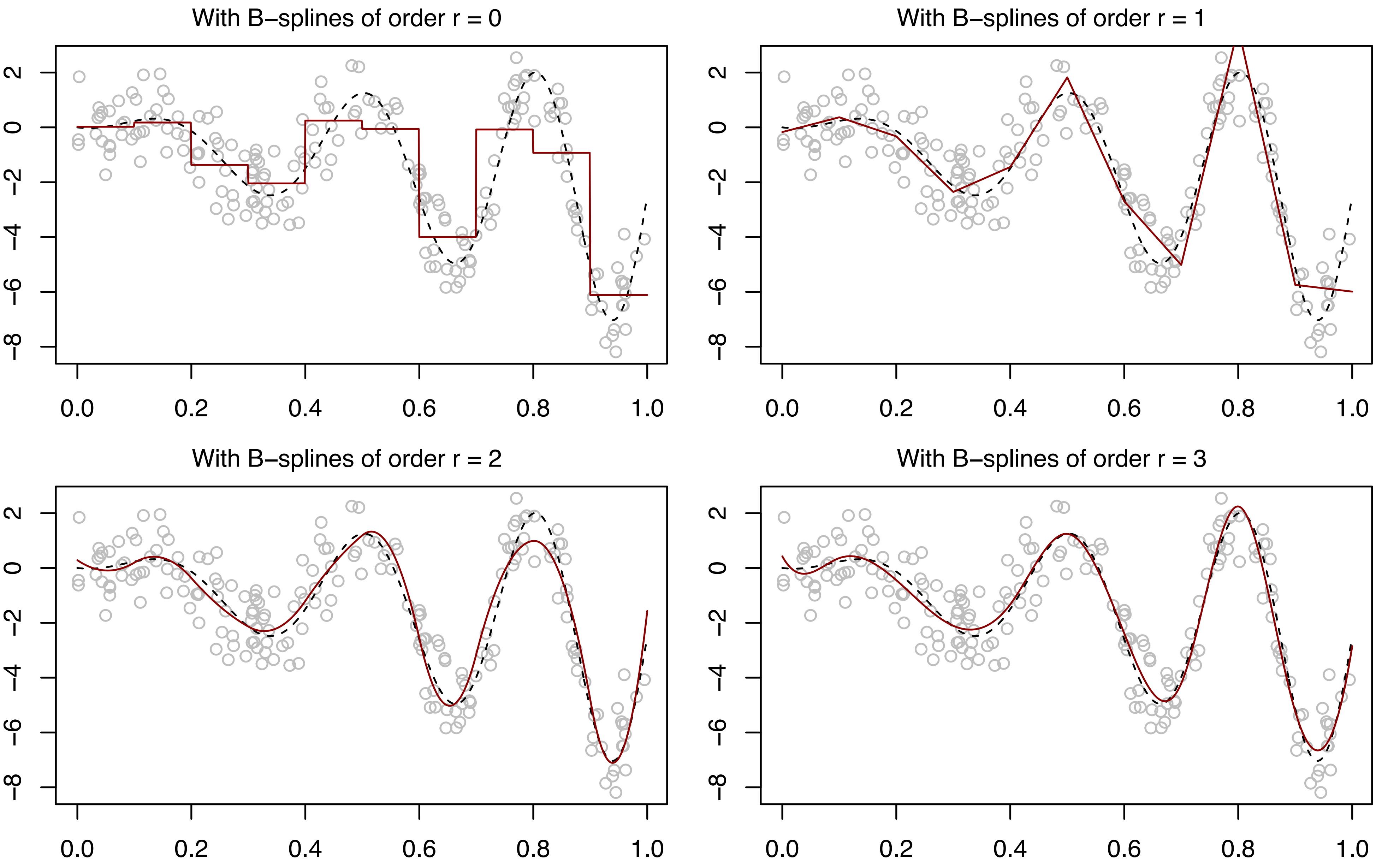
Note that (with the replicated boundary knots) the rows of $\mathbf{B}$ always sum to 1.

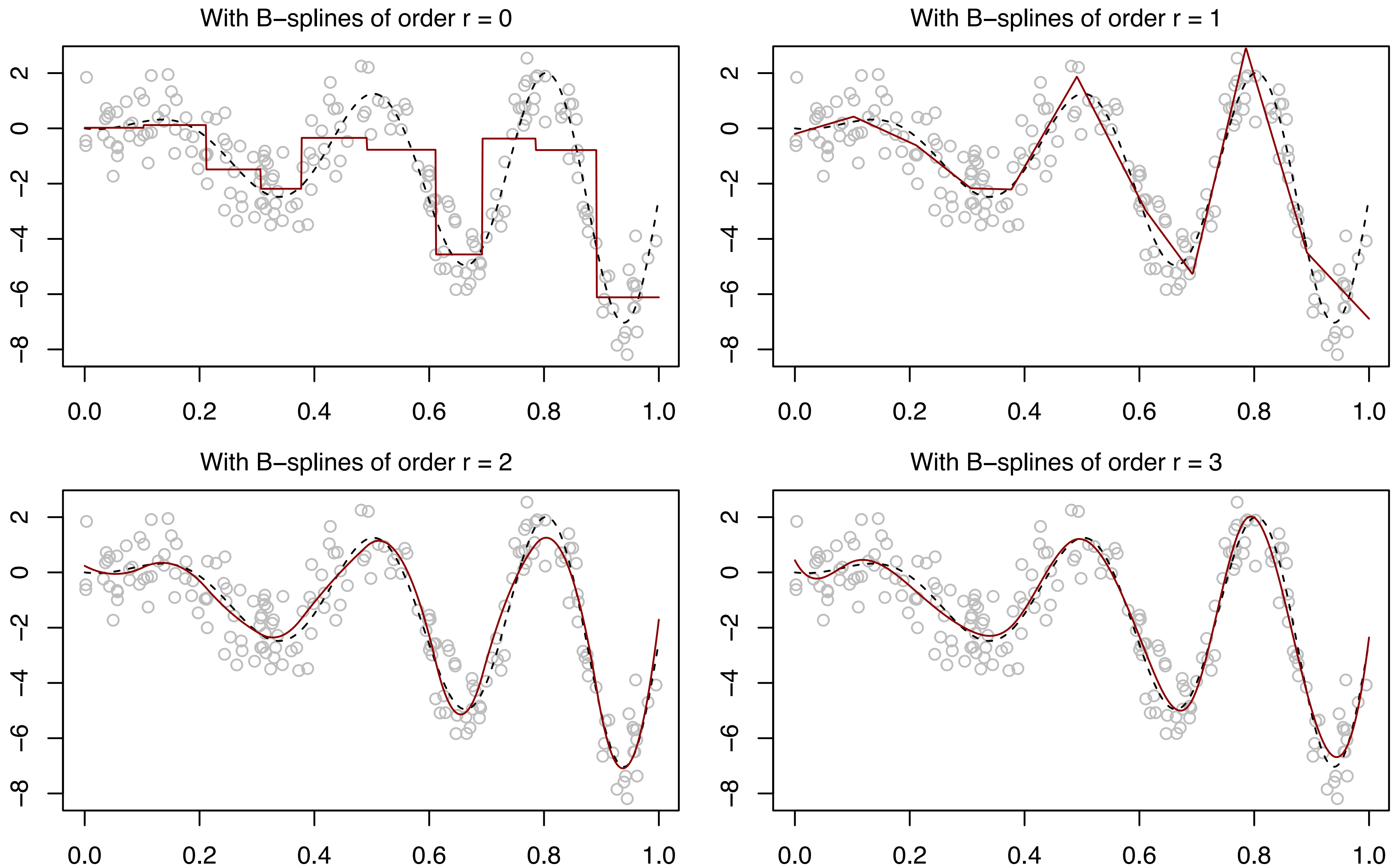
$$B = (b_k(x_i))_{1 \le i \le n,\, 1 \le k \le d}$$

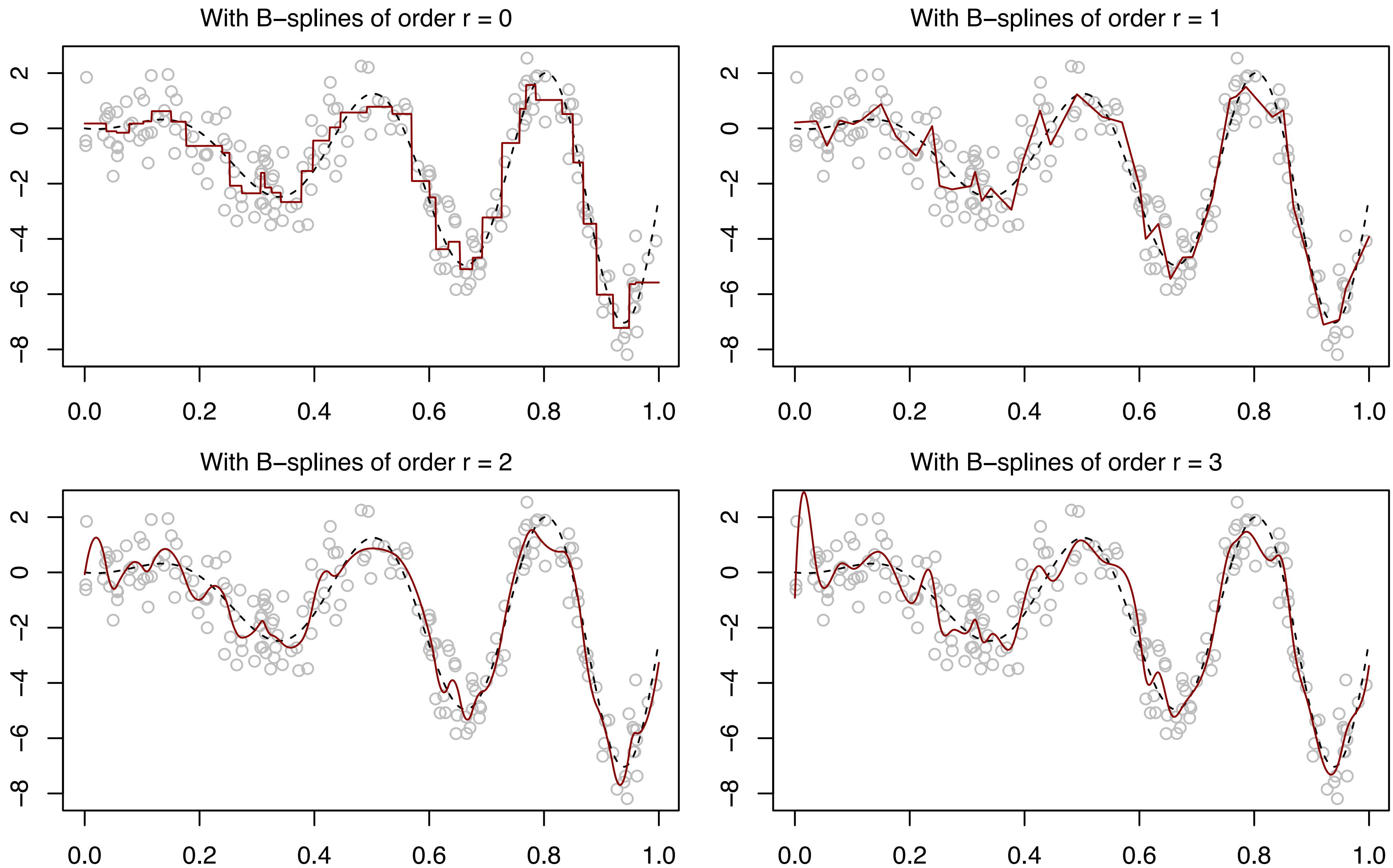
$$\hat{m}(x) = \underset{\sim x}{b}^T \underset{\sim}{\hat{\alpha}} = \sum_{k=1}^{d} \hat{\alpha}_k b_k(x) \qquad \hat{\alpha} = (B^T B)^{-1} B^T \underset{\sim}{Y}, \quad \underset{\sim x}{b} = (b_1(x), \dots, b_d(x))^T$$

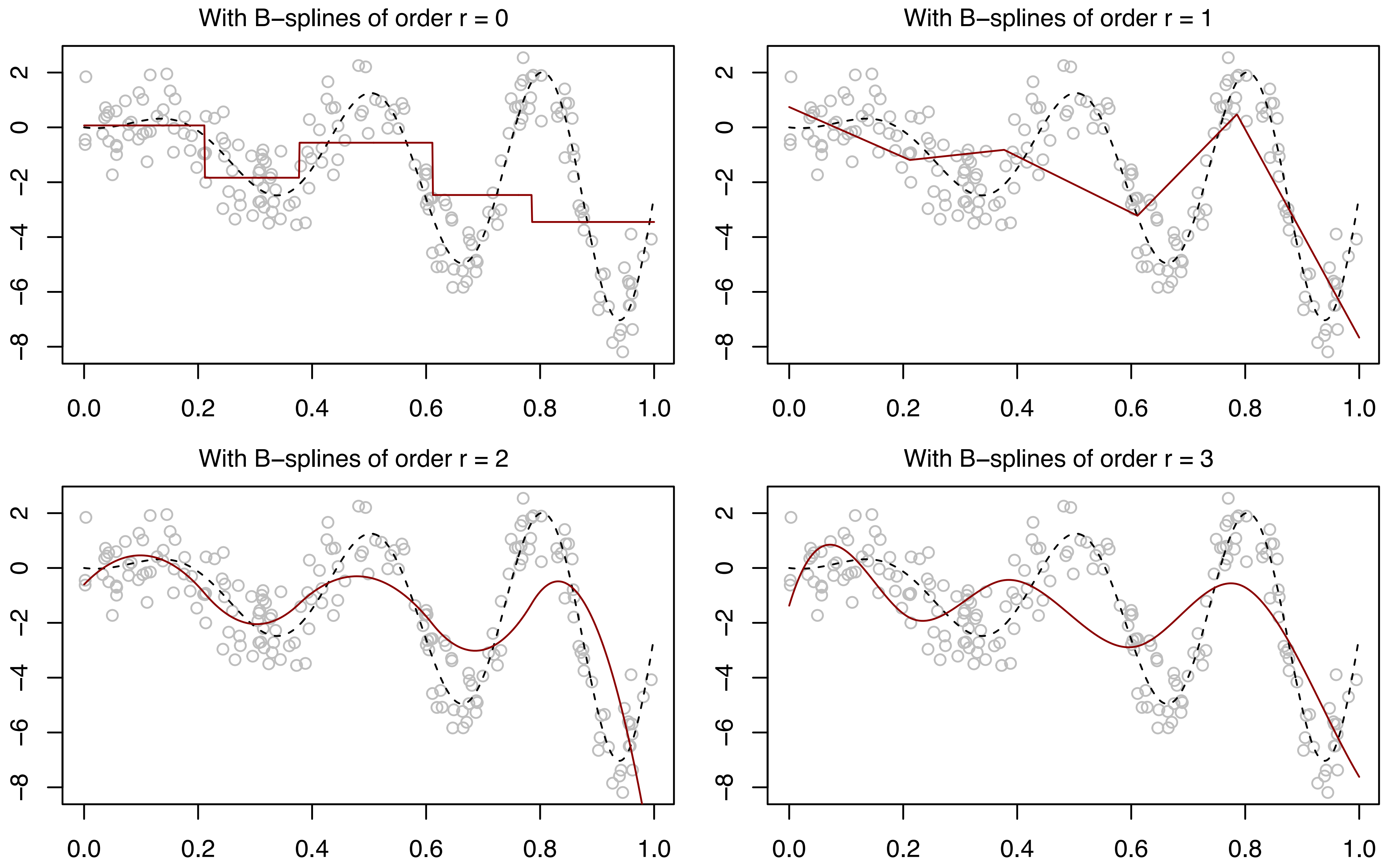
Exercise:

1. For $n = 200$, generate data $Y_i = m(X_i) + \varepsilon_i$ with
   - $X_1, \dots, X_n \overset{\text{ind}}{\sim} \text{Uniform}(0, 1)$, indep. of $\varepsilon_1, \dots, \varepsilon_n \overset{\text{ind}}{\sim} \text{Normal}(0, 1)$
   - $m(x) = 5x \cdot \sin(2\pi(1 + x)^2) - (5/2)x$
2. Plot $\hat{m}_{n,r}^{\text{spl}}$ under $K_n = 10$ for $r = 0, 1, 2, 3$ with
   - knots equally spaced in $[0, 1]$
   - knots at equally space quantiles of $X_1, \dots, X_n$
3. Try different values of K_n.


With B–splines of order r = 0
With B–splines of order r = 1
With B–splines of order r = 2
With B–splines of order r = 3
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0


With B–splines of order r = 0
With B–splines of order r = 1
With B–splines of order r = 2
With B–splines of order r = 3
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0


With B–splines of order r = 0
With B–splines of order r = 1
With B–splines of order r = 2
With B–splines of order r = 3
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0


With B–splines of order r = 0
With B–splines of order r = 1
With B–splines of order r = 2
With B–splines of order r = 3
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

Splines: see Stone (1985) [4]

For $K_n > 0$ an integer let $I_{nk} = [(k-1)/K_n, k/K_n)$ for $k = 1, \dots, K_n - 1$ and $I_{nK_n} = [(K_n - 1)/K_n, 1]$. Then for $r \geq 1$, define the set of functions

$$\mathcal{M}_{n,r} = \left\{ m : [0,1] \to \mathbb{R} : \begin{array}{l} m \text{ is a polynomial of degree } r \text{ or less on} \\ \text{each interval } I_1, \dots, I_{nK_n}, \text{ and } m \text{ is } r-1 \text{ times} \\ \text{continuously differentiable on } [0,1] \end{array} \right\}.$$

Moreover, let

$$\mathcal{M}_{n,0} = \{m : [0,1] \to \mathbb{R} : m \text{ is piecewise constant on } I_1, \dots, I_{nK_n}\}$$

- Fns in $\mathcal{M}_{n,1}$, $\mathcal{M}_{n,2}$, and $\mathcal{M}_{n,3}$ are called *linear, quadratic, and cubic splines*.
- Values j/K_n, $j = 0, \dots, K_n$ are called *knots*. Can choose knots differently.
- B-splines of order r defined over the same knots form a basis for $\mathcal{M}_{n,r}$.
- Functions in these spaces can nicely approximate functions in Hölder classes.

For a function $g : \mathcal{T} \to \mathbb{R}$, we write $\|g\|_\infty = \sup_{x \in \mathcal{T}} |g(x)|$.

Key result from deBoor (1968) [1]

For each $m \in \mathcal{H}(\beta, L)$ on $[0, 1]$, there exists a function $m^{\text{spl}}_{n,r} \in \mathcal{M}_{n,r}$, where $r \geq \beta - 1$ such that

$$\|m - m^{\text{spl}}_{n,r}\|_\infty \leq C \cdot K_n^{-\beta}$$

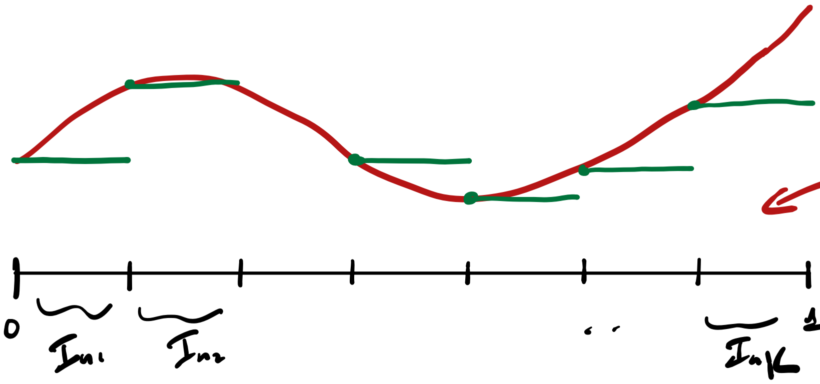
for some constant $C > 0$. As K_n grows, the spline approx to m gets better.

Idea is to let $K_n \to \infty$ as $n \to \infty$, so that this approximation error goes to zero.

Exercise: For $m \in \text{Lipschitz}(L)$ on $[0, 1]$, show that $\exists\ m^{\text{spl}}_{n,0} \in \mathcal{M}_{n,0}$ such that

$$\sup_{x \in [0,1]} |m(x) - m^{\text{spl}}_{n,0}(x)| \leq \frac{L}{K_n}.$$

pcwise constant over $\left[\frac{k-1}{K}, \frac{k}{K}\right) \ldots$



Each interval has width $\frac{1}{K_n}$

Take $m_{n,0}^{spl}(x) = m\left(\frac{k-1}{K_n}\right)$ if $x \in I_{nk}$

Then

$$\sup_{x \in [0,1]} \left| m(x) - m_{n,0}^{spl}(x) \right| = \max_{1 \le k \le K_n} \sup_{x \in I_{nk}} \left| m(x) - m_{n,0}^{spl}(x) \right|$$

$$= \max_{1 \le k \le K_n} \sup_{x \in I_{nk}} \underbrace{\left| m(x) - m\left(\frac{k-1}{K_n}\right) \right|}$$

$$\le L \left| x - \frac{k-1}{K_n} \right| \le \frac{L}{K_n}$$

$$\le \frac{L}{K_n}$$

We now define the order r least-squares splines estimator of m as

$$\hat{m}_{n,r}^{\text{spl}} = \underset{g \in \mathcal{M}_{n,r}}{\text{argmin}} \sum_{i=1}^{n} [Y_i - g(X_i)]^2.$$

$g(x) = \sum_{k=1}^{d} \alpha_k b_k(x)$ for some α_k

Bound on MSE $\hat{m}_{n,r}^{\text{spl}}(x_0)$

If $m \in \mathcal{H}(\beta, L)$ on $[0,1]$, then for $r \geq \beta - 1$, we have

$$\text{MSE}\, \hat{m}_{n,r}^{\text{spl}}(x_0) \leq C \cdot \left(K_n^{-2\beta} + \frac{K_n}{n} \right)$$

for all $x_0 \in [0,1]$ for large enough n, provided (C1), (C2), and (C3) hold.

We will study the conditions (C1), (C2), and (C3) later on.

Exercise:

1. Find the value of K_n which minimizes the MSE bound. $K_n = \text{const.}\ n^{\frac{1}{2\beta+1}}$
2. Give the minimum bound over choices of K_n. $\text{MSE}\ \hat{m}_{n,r}^{\text{spl}}(x) \leq \text{const.}\ n^{-\frac{2\beta}{2\beta+1}}$
3. Anything interesting about this?

$$Y_i = m(x_i) + \varepsilon_i, \quad i = 1, \ldots, n, \qquad \varepsilon_1, \ldots, \varepsilon_n \overset{iid}{\sim} N(0, \sigma^2)$$

(x_i fixed)

$$\hat{m}^{spl}_{n,r}(x) = \sum_{k=1}^{d} \hat{\alpha}_k b_k(x) = \underset{\sim}{b}_x^T \hat{\underset{\sim}{\alpha}}, \qquad \underset{\sim}{b}_x = (b_1(x), \ldots, b_d(x))^T$$

$$\hat{\underset{\sim}{\alpha}} = (B^T B)^{-1} B^T \underset{\sim}{Y}$$

So

$$\hat{m}^{spl}_{n,r}(x) = \underset{\sim}{b}_x^T (B^T B)^{-1} B^T \underset{\sim}{Y}, \qquad \mathrm{Cov}(\underset{\sim}{Y}) = \sigma^2 I_n$$

$$\underset{\sim}{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$$

Then

$$\mathrm{Var}\left(\hat{m}^{spl}_{n,r}(x)\right) = \sigma^2 \underset{\sim}{b}_x^T (B^T B)^{-1} \underset{\sim}{b}_x$$

$$= \frac{\sigma^2}{n} \underset{\sim}{b}_x^T \underset{\sim}{b}_x \frac{\underset{\sim}{b}_x^T \left(\frac{1}{n} B^T B\right)^{-1} \underset{\sim}{b}_x}{\underset{\sim}{b}_x^T \underset{\sim}{b}_x}$$

$$\left(\underset{\sim}{b}_x^T \underset{\sim}{b}_x = \sum_{k=1}^{d} b_k^2(x) \le 1 \quad \text{since } \|\underset{\sim}{b}_x\|_1 = 1 \right)$$

$$\le \frac{\sigma^2}{n} (1) \, \lambda_{max}\left(\left(\frac{1}{n} B^T B\right)^{-1}\right)$$

$$= \frac{\sigma^2}{n} \frac{1}{\lambda_{min}\left(\frac{1}{n} B^T B\right)}$$

$$\le \frac{\sigma^2}{n} \frac{1}{c_1 \cdot \frac{1}{K_n}}$$

$$= \frac{K_n}{n} \frac{\sigma^2}{c_1}.$$

For a p.d. matrix A

$$\sup_{\underset{\sim}{x}} \frac{\underset{\sim}{x}^T A \underset{\sim}{x}}{\underset{\sim}{x}^T \underset{\sim}{x}} = \lambda_{max}(A)$$

If A has eigenvalues $\lambda_1 \le \cdots \le \lambda_d$

Then A^{-1} has eigenvalues $\frac{1}{\lambda_1} \ge \cdots \ge \frac{1}{\lambda_d}$

$$\text{(C1)}\ K_n^{-1} \cdot c_1 \le \lambda_{\min}\left(n^{-1}\mathbf{B}^T\mathbf{B}\right) \le \lambda_{\max}\left(n^{-1}\mathbf{B}^T\mathbf{B}\right) \le C_1 \cdot K_n^{-1}$$

Bias

$$\mathbb{E}\, \hat{m}^{spl}_{n,r}(x) - m(x) = \mathbb{E}\left(\underset{\sim x}{b}^T (B^TB)^{-1} B^T \underset{\sim}{Y} \right) - m(x)$$

$$= \underset{\sim x}{b}^T (B^TB)^{-1} B^T \underset{\sim}{m} - m(x), \quad \underset{n\times 1}{\underset{\sim}{m}} = \begin{bmatrix} m(X_1) \\ \vdots \\ m(X_n) \end{bmatrix} \leftarrow \text{at "design points"}$$

$$= \underset{\sim x}{b}^T (B^TB)^{-1} B^T \underset{\sim}{m} - m^{spl}_{n,r}(x) + m^{spl}_{n,r}(x) - m(x)$$

where $m^{spl}_{n,r}$ is a function in $\mathcal{M}_{n,r}$ such that $\| m^{spl}_{n,r} - m \|_\infty \le \text{const.} \frac{1}{K^\beta}$

$m^{spl}_{n,r}(x) = \underset{\sim x}{b}^T \underset{\sim}{\alpha}^*$ for some $\underset{\sim}{\alpha}^*$.

$$= \underset{\sim x}{b}^T (B^TB)^{-1} B^T B \underset{\sim}{\alpha}^*$$

$$\underset{n\times 1}{\underset{\sim}{m}^{spl}_{n,r}} = \begin{bmatrix} m^{spl}_{n,r}(X_1) \\ \vdots \\ m^{spl}_{n,r}(X_n) \end{bmatrix}$$

$$= \underset{\sim x}{b}^T (B^TB)^{-1} B^T \left(\underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right) + m^{spl}_{n,r}(x) - m(x)$$

$$\left| \mathbb{E}\, \hat{m}^{spl}_{n,r}(x) - m(x) \right| \le \left| \underset{\sim x}{b}^T (B^TB)^{-1} B^T \left(\underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right) \right| + \underbrace{\left| m^{spl}_{n,r}(x) - m(x) \right|}_{\le \text{const.} \frac{1}{K_n^\beta}}$$

$$\Big| \underbrace{\underbrace{\underset{\sim x}{b}^T}_{(d\times 1)^T} \underbrace{(B^TB)^{-1}}_{d\times d}}_{1\times d} \underbrace{\underbrace{B^T}_{(n\times d)^T} \underbrace{\left(\underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right)}_{n\times 1}}_{d\times 1} \Big| \le \left\| (B^TB)^{-1} \underset{\sim x}{b} \right\|_1 \left\| B^T \left(\underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right) \right\|_\infty$$

$$= \left\| \left(\tfrac{1}{n} B^TB\right)^{-1} \underset{\sim x}{b} \right\|_1 \left\| \tfrac{1}{n} B^T \left(\underset{\sim}{m} - m^{spl}_{n,r} \right) \right\|_\infty$$

$$u, v \in \mathbb{R}^n \Rightarrow |u^T v| = \left|\sum_{i=1}^n u_i v_i\right| \le \sum_{i=1}^n |u_i||v_i| \le \max_{1\le i\le n} |v_i| \cdot \sum_{i=1}^n |u_i| = \|u\|_1 \cdot \|v\|_\infty$$

$$\|u\|_1 \quad \|v\|_\infty$$

$$\|\underset{d\times d}{A}\, \underset{d\times 1}{x}\|_1 = \left\| \begin{bmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{d1} & \cdots & a_{dd} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \right\|_1 \le \|h_x\|_1 \left\| \left(\frac{1}{n} B^T B\right)^{-1} \right\|_\infty \left\| \frac{1}{n} B^T \left(m - m_{n,r}^{spl}\right) \right\|_\infty$$

$\|h_x\|_1$: 1

$\|\cdot\|_\infty$ of a matrix is max abs row sum.

$$= \left\| x_1 \begin{bmatrix} a_{11} \\ \vdots \\ a_{d1} \end{bmatrix} + \cdots + x_d \begin{bmatrix} a_{1d} \\ \vdots \\ a_{dd} \end{bmatrix} \right\|_1$$

$$\le \sum_{k=1}^d |x_k| \left\| \begin{bmatrix} a_{k1} \\ \vdots \\ a_{kd} \end{bmatrix} \right\|_1$$

$$\le \left(\max_{1\le k\le d} \left\| \begin{bmatrix} a_{k1} \\ \vdots \\ a_{kd} \end{bmatrix} \right\|_1 \right) \|x\|_1$$

(max abs column sum)

(C2) $\left\| \left(n^{-1}\mathbf{B}^T\mathbf{B}\right)^{-1} \right\|_\infty \le C_2 \cdot K_n$

(C3) $\left\| n^{-1}\mathbf{B}^T(\mathbf{m} - \mathbf{m}_{n,r}^{\text{spl}}) \right\|_\infty \le C_3 \cdot K_n^{-1-\beta}$,

$$\le C_2 \cdot K_n \cdot C_3 \cdot K_n^{-1-\beta}$$

$$= \text{const.} * K_n^{-\beta}$$

Conditions for bounding MSE $\hat{m}^{\text{spl}}_{n,r}(x_0)$; see Zhou (1998) [7]

Let $m \in \mathcal{H}(\beta, L)$ on $[0,1]$ and let $m^{\text{spl}}_{n,r} \in \mathcal{M}_{n,r}$ satisfy $\|m - m^{\text{spl}}_{n,r}\|_\infty \leq C \cdot K_n^{-\beta}$.

Let $X_1, \dots, X_n \in [0,1]$ be deterministic such that for large enough n,

(C1) $K_n^{-1} \cdot c_1 \leq \lambda_{\min}\left(n^{-1}\mathbf{B}^T\mathbf{B}\right) \leq \lambda_{\max}\left(n^{-1}\mathbf{B}^T\mathbf{B}\right) \leq C_1 \cdot K_n^{-1}$

(C2) $\left\| \left(n^{-1}\mathbf{B}^T\mathbf{B}\right)^{-1} \right\|_\infty \leq C_2 \cdot K_n$

(C3) $\left\| n^{-1}\mathbf{B}^T(\mathbf{m} - \mathbf{m}^{\text{spl}}_{n,r}) \right\|_\infty \leq C_3 \cdot K_n^{-1-\beta}$,

where

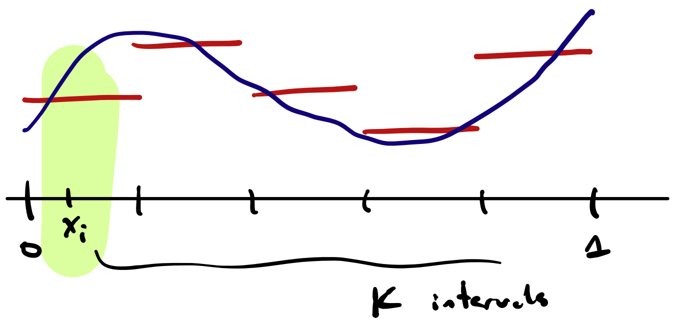
$$\mathbf{m} = (m(X_1), \dots, m(X_n))^T \quad \text{and} \quad \mathbf{m}^{\text{spl}}_{n,r} = (m^{\text{spl}}_{n,r}(X_1), \dots, m^{\text{spl}}_{n,r}(X_n))^T.$$

Exercise:

1. Use above to get bounds on the bias and variance of $\hat{m}^{\text{spl}}_{n,r}(x_0)$.
2. Consider (C1), (C2), and (C3) in the case of $\beta = 1$, $r = 0$.

Let $\beta = 1$ (Lipschitz)

Let $r = 0$ (piecewise constant estimator)



We use $b_1(x), \dots, b_K(x)$, $b_k(x) = \mathbb{1}(x \in I_{nk})$, $k = 1, \dots, K$

$$B = \begin{bmatrix} b_1(x_1) & \cdots & b_K(x_1) \\ \vdots & & \vdots \\ b_1(x_n) & \cdots & b_K(x_n) \end{bmatrix}$$

(C.1) $$\frac{1}{n} B^T B = \begin{bmatrix} \frac{1}{n}\sum_{i=1}^{n} b_1(x_i) b_1(x_i) & \cdots & \frac{1}{n}\sum_{i=1}^{n} b_1(x_i) b_K(x_i) \\ \vdots & \ddots & \vdots \\ \frac{1}{n}\sum_{i=1}^{n} b_K(x_i) b_1(x_i) & \cdots & \frac{1}{n}\sum_{i=1}^{n} b_K(x_i) b_K(x_i) \end{bmatrix}$$

$b_1(x_i)\, b_K(x_i) = 0$

$$= \begin{bmatrix} \frac{\#\{x_i \in I_{n1}\}}{n} & & \\ & \ddots & \\ & & \frac{\#\{x_i \in I_{nk}\}}{n} \end{bmatrix}$$

diagonal $\longrightarrow$

$$\lambda_{min}\left(\tfrac{1}{n} B^T B\right) = \min_{1 \le k \le K} \left\{ \frac{\#\{x_i \in I_{nk}\}}{n} \right\} \geq \text{const.}\ \frac{1}{K_n}$$

Makes sense to assume this?

If $X_1, \ldots, X_n \overset{ind}{\sim} U(0,1)$, then $\mathbb{E}\, \frac{\#\{X_i \in I_{nk}\}}{n} = \frac{1}{K_n}$

(C.2)

$$\left\| \left(\tfrac{1}{n} B^T B\right)^{-1} \right\|_\infty = \left\| \begin{bmatrix} \frac{n}{\#\{X_i \in I_{n1}\}} & & \\ & \ddots & \\ & & \frac{n}{\#\{X_i \in I_{nk}\}} \end{bmatrix} \right\|_\infty$$

$$= \max_{1 \le k \le K} \left\{ \frac{n}{\#\{X_i \in I_{nk}\}} \right\}$$

$$= \frac{1}{\min_{1 \le k \le K} \left\{ \frac{\#\{X_i \in I_{nk}\}}{n} \right\}}$$

$$\lesssim \widetilde{\text{const.}} \quad K_n$$

(C.3) $$\left\| \frac{1}{n} B^T \left(\underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right) \right\|_\infty$$

$$= \left\| \frac{1}{n} \underbrace{\underset{K \times n}{\begin{bmatrix} b_1(X_1) & \cdots & b_1(X_n) \\ & & \\ b_K(X_1) & \cdots & b_K(X_n) \end{bmatrix}} \underset{n \times 1}{\begin{pmatrix} m(X_1) - m^{spl}_{n,r}(X_1) \\ \vdots \\ m(X_n) - m^{spl}_{n,r}(X_n) \end{pmatrix}}}_{K \times 1} \right\|_\infty$$

$$= \left\| \frac{1}{n} \sum_{i=1}^{n} \begin{bmatrix} b_1(X_i) \\ \vdots \\ b_K(X_i) \end{bmatrix} \left(m(X_i) - m^{spl}_{n,r}(X_i) \right) \right\|_\infty$$

$$\leq \max_{1 \leq k \leq K} \left| \frac{1}{n} \sum_{i=1}^{n} b_k(X_i) \underbrace{\left(m(X_i) - m^{spl}_{n,r}(X_i) \right)} \right|$$

$$\sum_{i=1}^{n} a_i b_i \leq \|\underset{\sim}{a}\|_1 \, \|\underset{\sim}{b}\|_\infty$$

$$\leq \frac{1}{n} \left(\max_{1 \leq k \leq K} \sum_{i=1}^{n} |b_k(X_i)| \right) \underbrace{\left\| \underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right\|_\infty}_{\leq C \cdot K^{-\beta}}$$

$$= \left(\max_{1 \leq k \leq K} \frac{\# \{ X_i \in I_{nk} \}}{n} \right) \left\| \underset{\sim}{m} - \underset{\sim}{m}^{spl}_{n,r} \right\|_\infty$$

$$\underbrace{\qquad}_{\text{const.}\ \frac{1}{K_n}} \qquad \underbrace{\qquad}_{C \cdot \frac{1}{K_n^{\beta}}}$$

$$\leq \quad \text{const.} \quad K_n^{-1-\beta} \quad .$$

Let $(X_1, Y_1), \dots, (X_n, Y_n)$ be indep. realizations of $(X, Y) \in [0, 1] \times \mathbb{R}$, where

$$Y = m(X) + \varepsilon, \qquad \text{for some } m : [0,1] \to \mathbb{R},$$

where ε is independent of X with $\mathbb{E}\varepsilon = 0$ and $\mathbb{E}\varepsilon^2 = \sigma^2$.

Define the class of functions (which is a Sobolev space)

$$\mathcal{W}_2 = \left\{ g : [0,1] \to \mathbb{R} : g' \text{ is continuous, } \int_0^1 [g''(x)]^2 dx < \infty \right\}.$$

Note that $\mathcal{H}(2, L)$ on $[0, 1]$ is contained in $\mathcal{W}_2$.

Smoothing spline estimator

The estimator

$$\hat{m}_n^{\text{sspl}} = \underset{g \in \mathcal{W}_2}{\operatorname{argmin}} \sum_{i=1}^{n} [Y_i - g(X_i)]^2 + \lambda \int_0^1 [g''(x)]^2 dx,$$

for $\lambda > 0$ is called the *smoothing spline estimator* of m.

How do we search among all functions belonging to $\mathcal{W}_2$?

Beautiful result: $\hat{m}_n^{\text{sspl}}$ is a natural cubic spline with knots $u_i = X_i$, $i = 1, \ldots, n$.

Natural cubic splines are cubic splines constrained to be linear beyond end knots.

Can bound MSE by $C \cdot n^{-4/5}$ when $\lambda = c \cdot n^{1/5}$. See Grace Wahba's book, [6].

Sets of basis functions for natural cubic splines lack nice properties of B-splines.

Since we love B-splines, we often consider (instead of smoothing splines) this:

Penalized spline estimator

Let $\mathcal{M}_{n,3}$ be the space of cubic splines on the knots

$$u_{-3} = u_{-2} = u_{-1} = u_0 < u_1 < \cdots < u_{K_n} = u_{K_n+1} = u_{K_n+2} = u_{K_n+3}.$$

Large K_n

Then

$$\hat{m}_n^{\text{pspl}} = \underset{g \in \mathcal{M}_{n,3}}{\operatorname{argmin}} \sum_{i=1}^{n} [Y_i - g(X_i)]^2 + \lambda \int_0^1 [g''(x)]^2 dx,$$

for $\lambda > 0$ is the *penalized spline estimator* of m. Nice reference is [2].

Idea is to choose K_n very large and then tune wiggliness by choosing λ.

When K_n is very large, $\hat{m}_n^{\text{pspl}}$ is practically identical to $\hat{m}_n^{\text{sspl}}$.

Exercise: Give a representation of $\hat{m}_n^{\text{pspl}}(x_0)$ in matrices given a basis for $\mathcal{M}_{n,3}$.

$$= \underset{g \in \mathcal{M}_{n,3}}{\operatorname{argmin}} \sum_{i=1}^{n} [Y_i - g(X_i)]^2 + \lambda \int_0^1 [g''(x)]^2 dx,$$

For $g \in \mathcal{M}_{n,3}$ there are coefficients $\alpha_1, \dots, \alpha_d$ such that

$$g(x) = \sum_{k=1}^{d} \alpha_k b_k(x), \quad b_1(x), \dots, b_d(x)$$

are basis functions.

For $g \in \mathcal{M}_{n,3}$

$$\sum_{i=1}^{n} \left(Y_i - g(X_i) \right)^2 = \sum_{i=1}^{n} \left(Y_i - \sum_{k=1}^{d} \alpha_k b_k(X_i) \right)^2$$

$$= \left\| \underset{n \times 1}{\underset{\sim}{Y}} - \underset{n \times d}{B} \underset{d \times 1}{\underset{\sim}{\alpha}} \right\|^2, \quad B = \begin{bmatrix} b_1(X_1) & \cdots & b_d(X_1) \\ \vdots & & \vdots \\ b_1(X_n) & \cdots & b_d(X_n) \end{bmatrix}$$

LS splines:

$$\hat{\underset{\sim}{\alpha}}^{LS} = (B^T B)^{-1} B^T \underset{\sim}{Y}$$

$$\Rightarrow \quad \hat{m}^{spl}_{n,r}(x) = \sum_{k=1}^{d} \hat{\alpha}^{LS}_k b_k(x) = \underset{\sim}{b}_x^T \hat{\underset{\sim}{\alpha}}^{LS},$$

$$\underset{\sim}{b}_x = (b_1(x), \dots, b_d(x))^T$$

Add penalty:

$$f''(x) = \frac{d^2}{dx^2} \sum_{k=1}^{d} \alpha_k b_k(x) = \sum_{k=1}^{d} \alpha_k b_k''(x)$$

$$\lambda \int_0^1 [f''(x)]^2 dx = \lambda \int_0^1 \left[\sum_{k=1}^{d} \alpha_k b_k''(x) \right]^2 dx$$

$$= \lambda \int_0^1 \sum_{k=1}^{d} \sum_{k'=1}^{d} \alpha_k \alpha_{k'} b_k''(x) b_{k'}''(x) \, dx$$

$$= \lambda \sum_{k=1}^{d} \sum_{k'=1}^{d} \alpha_k \alpha_{k'} \int_0^1 b_k''(x) b_{k'}''(x) \, dx$$

$$= \lambda \, \underset{\sim}{\alpha}^T \underbrace{\left(\left(\int_0^1 b_k''(x) b_{k'}''(x) \, dx \right)_{1 \le k, k' \le d} \right)}_{\Omega} \underset{\sim}{\alpha}$$

$$= \lambda \, \underset{\sim}{\alpha}^T \Omega \underset{\sim}{\alpha}$$

$$\hat{\underset{\sim}{\alpha}} = \underset{\underset{\sim}{\alpha}}{\operatorname{argmin}} \, \|\underset{\sim}{Y} - B \underset{\sim}{\alpha}\|^2 + \lambda \, \underset{\sim}{\alpha}^T \Omega \underset{\sim}{\alpha}$$

$$\frac{\partial}{\partial \underset{\sim}{\alpha}} \left(\|\underset{\sim}{Y} - B \underset{\sim}{\alpha}\|^2 + \lambda \, \underset{\sim}{\alpha}^T \Omega \underset{\sim}{\alpha} \right) \overset{\text{set}}{=} \underset{\sim}{0}$$

$$\iff -2 B^T (Y - B \underset{\sim}{\alpha}) + \lambda \, 2 \, \Omega \alpha = \underset{\sim}{0}$$

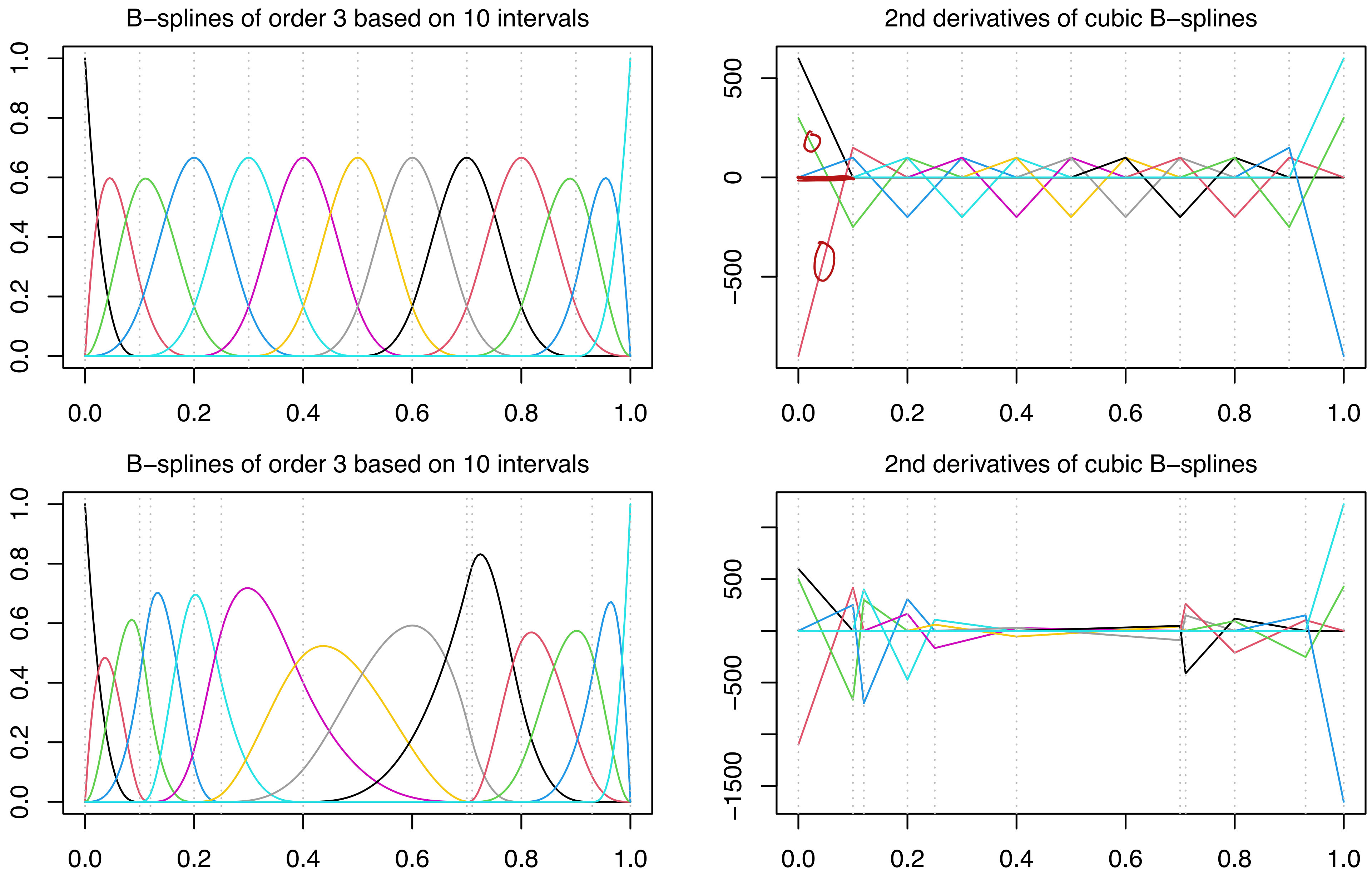
$$\iff (B^T B + \lambda \Omega) \underset{\sim}{\alpha} = B^T \underset{\sim}{Y}$$

$$\Leftrightarrow \quad \hat{\underset{\sim}{\alpha}} = \left(B^T B + \lambda \Omega \right)^{-1} B^T \underset{\sim}{y} .$$

We can obtain the row vector $\mathbf{b}''(x) = (b_1''(x), \ldots, b_d''(x))$ with

```
splineDesign(knots=knots,x=x,ord=4,derivs=rep(2,K+1))
```

where `knots` is the complete set of knots $u_{-3}, \ldots, u_{K+3}$.


B–splines of order 3 based on 10 intervals
2nd derivatives of cubic B–splines
B–splines of order 3 based on 10 intervals
2nd derivatives of cubic B–splines

Computation of $\boldsymbol{\Omega}$

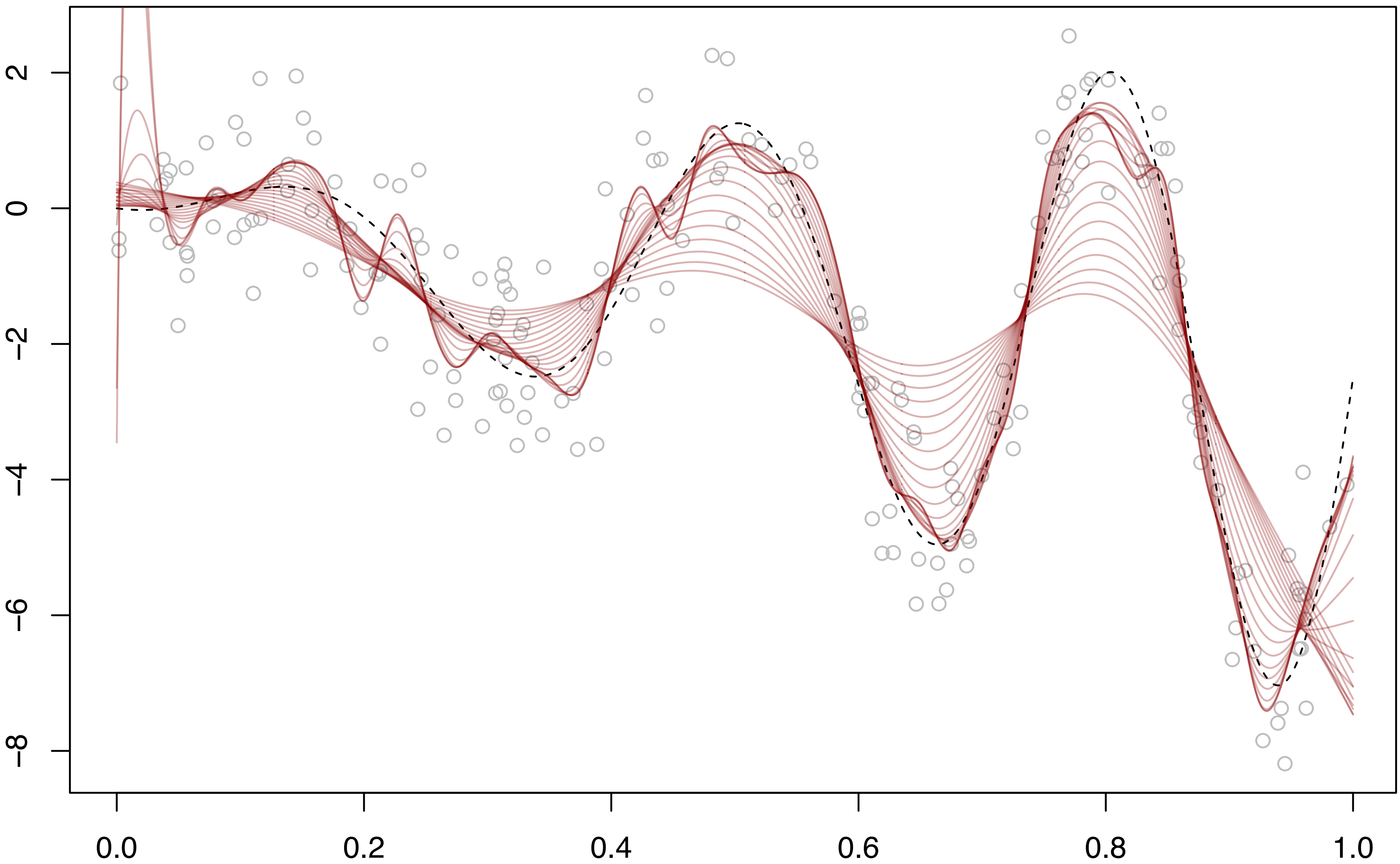
For cubic B-splines basis functions $b_1, \dots, b_{d_n}$, $d_n = K_n + 3$, based on knots $u_{-3} = u_{-2} = u_{-1} = u_0 < u_1 < \cdots < u_{K_n} = u_{K_n+1} = u_{K_n+2} = u_{K_n+3}$, we have

$$\int_0^1 b_\ell''(x) b_j''(x) dx = \sum_{k=0}^{K-1} (u_{k+1} - u_k) \Big[\frac{1}{2}(b_j''(u_k) b_\ell''(u_{k+1}) + b_j''(u_{k+1}) b_\ell''(u_k)) + \frac{1}{3}(b_\ell''(u_{k+1}) - b_\ell''(u_k))(b_j''(u_{k+1}) - b_j''(u_k)) \Big]$$

for each $1 \leq j, \ell \leq d_n$.

We can derive the above using the fact that each b_ℓ'' is piecewise linear.

Exercise: Demonstrate fitting the penalized splines estimator.


2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0

$\underset{n\times 1}{\hat{\underset{\sim}{m}}{}_n^{pspl}} = \begin{bmatrix} \hat{m}_n^{pspl}(X_1) \\ \vdots \\ \hat{m}_n^{pspl}(X_n) \end{bmatrix} = \mathbf{B}\hat{\underset{\sim}{\alpha}} = \underbrace{\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda\Omega)^{-1}\mathbf{B}^T}_{S}\underset{\sim}{Y} = S\underset{\sim}{Y}$

"Fitted values"

$\hat{\underset{\sim}{\alpha}} = (\mathbf{B}^T\mathbf{B} + \lambda\Omega)^{-1}\mathbf{B}^T\underset{\sim}{Y}$

"smoother" matrix.

Note that we may write

$$(\hat{m}_n^{\text{pspl}}(X_1), \ldots, \hat{m}_n^{\text{pspl}}(X_n))^T = \mathbf{SY},$$

$Y = X\beta + \underset{\sim}{\varepsilon}$

$\hat{\beta} = (X^TX)^{-1}X^T\underset{\sim}{Y}$

$\hat{Y} = X\hat{\underset{\sim}{\beta}}$

where $\mathbf{S} = \mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda\mathbf{\Omega})^{-1}\mathbf{B}^T$, with

$$\mathbf{B} = (b_\ell(X_i))_{1\leq i\leq n, 1\leq \ell\leq d_n} \quad \text{and} \quad \mathbf{\Omega} = \left(\int_0^1 b_\ell''(x)b_j''(x)dx\right)_{1\leq \ell, j\leq d}.$$

The $n \times n$ matrix $\mathbf{S}$ is called a *smoother matrix*.

If $\lambda = 0$, $S = B(B^TB)^{-1}B^T$

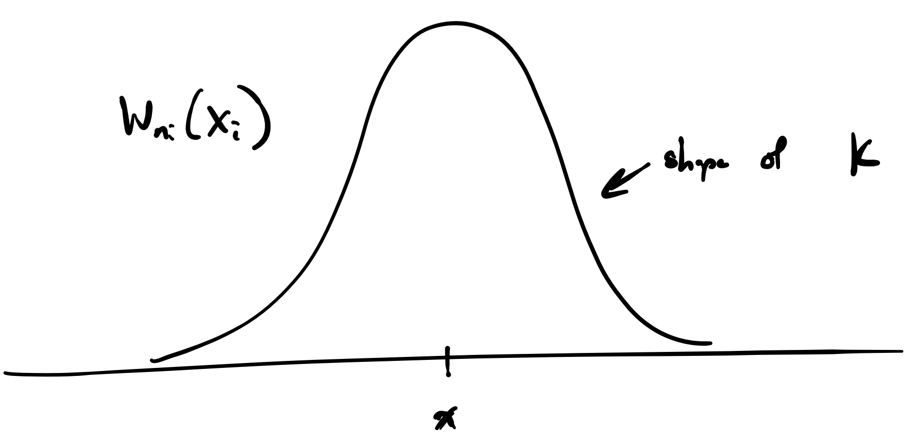
```
# set number of knots, penality parameter
K <- 10
lam <- .001

# place knots in [0,1], compute design matrix of basis function evaluations
u <- (0:K)/K
urep <- c(rep(0,3),u,rep(1,3))  # replicate boundary knots for cubic splines
B <- spline.des(knots = urep, x = X, ord = 4)$design # X contains x1,...,xn

# obtain 2nd derivatives
ddBu <- spline.des(knots = urep,
                   x = u,
                   derivs = rep(2,length(u)),
                   ord = 4)$design

# compute the matrix Omega
Om <- matrix(NA,K+3,K+3)
for(l in 1:(K+3))
  for(j in 1:(K+3)){

    tk <- ddBu[-(K+1),l]
    gk <- ddBu[-(K+1),j]
    tkp1 <- ddBu[-1,l]
    gkp1 <- ddBu[-1,j]
    Om[l,j] <- sum((u[-1]-u[-(K+1)])*((1/2)*(gk*tkp1+gkp1*tk)+(1/3)*(tkp1-tk)*(gkp1-gk)))

  }

# Compute smoother matrix
S <- B %*% solve(t(B) %*% B + lam * Om) %*% t(B)
```

Recall writing

$$\hat{m}_n(x) = \sum_{i=1}^{n} W_{ni}(x)\, Y_i$$

$$\begin{bmatrix} \hat{m}_n(X_1) \\ \vdots \\ \hat{m}_n(X_n) \end{bmatrix} = \underbrace{\begin{bmatrix} W_{n1}(X_1) & \cdots & W_{nn}(X_1) \\ \vdots & & \vdots \\ W_{n1}(X_n) & \cdots & W_{nn}(X_n) \end{bmatrix}}_{S} \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}$$

N-W: $$W_{ni}(x) = \frac{K\left(\frac{X_i - x}{h}\right)}{\sum_{i'=1}^{n} K\left(\frac{X_{i'} - x}{h}\right)}$$



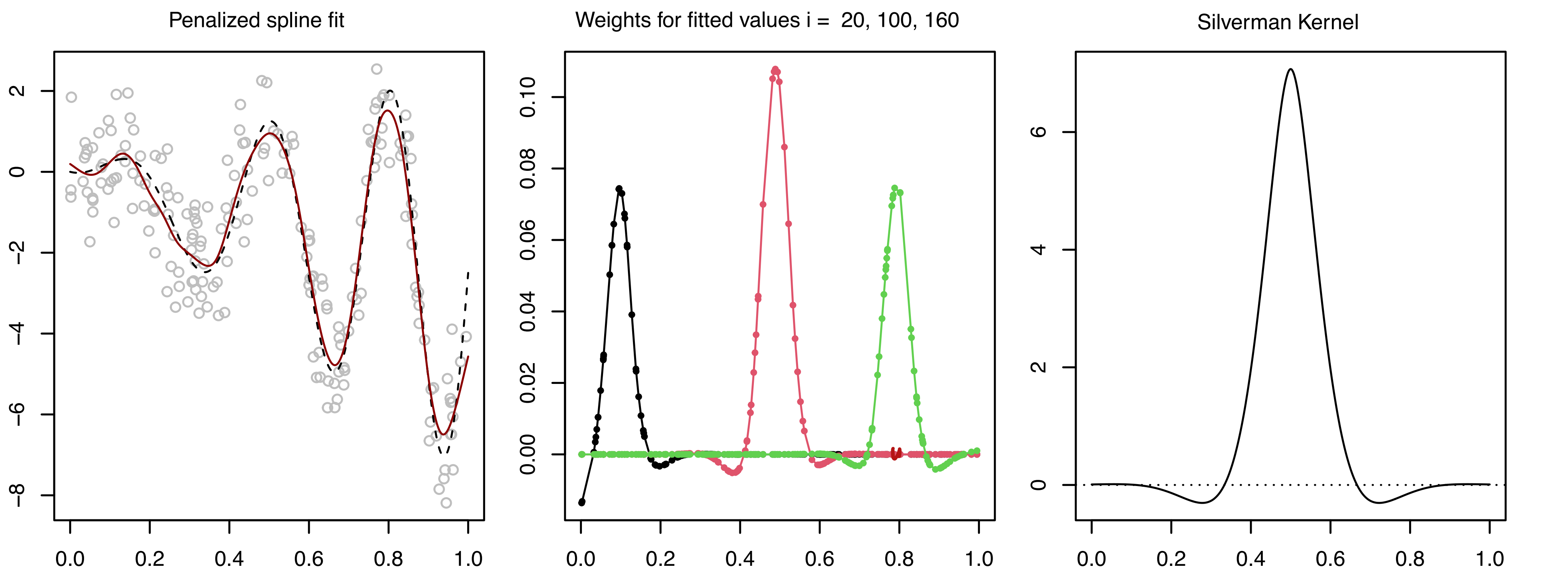
Linear estimator and smoother matrix

A *linear estimator* is any estimator $\hat{m}_n$ of the form

$$\hat{m}_n(x) = \sum_{i=1}^n W_{ni}(x) Y_i \quad \text{for some weights} \quad W_{n1}(x), \dots, W_{nn}(x).$$

The *smoother matrix* associated with $\hat{m}_n$ is the matrix $\mathbf{S} = (W_{ni}(X_{i'}))_{1 \le i, i' \le n}$.

Exercise: Plot some rows of a penalized spline smoother matrix $\mathbf{S}$. Discuss.


Penalized spline fit
Weights for fitted values i = 20, 100, 160
Silverman Kernel
0.0
0.2
0.4
0.6
0.8
1.0

Silverman's kernel, [3]

For $X_1, \ldots, X_n$ having the density f on $[0,1]$, the smoothing spline estimator is asymptotically equivalent to N-W under K and local bandwidth $h(x)$ given by

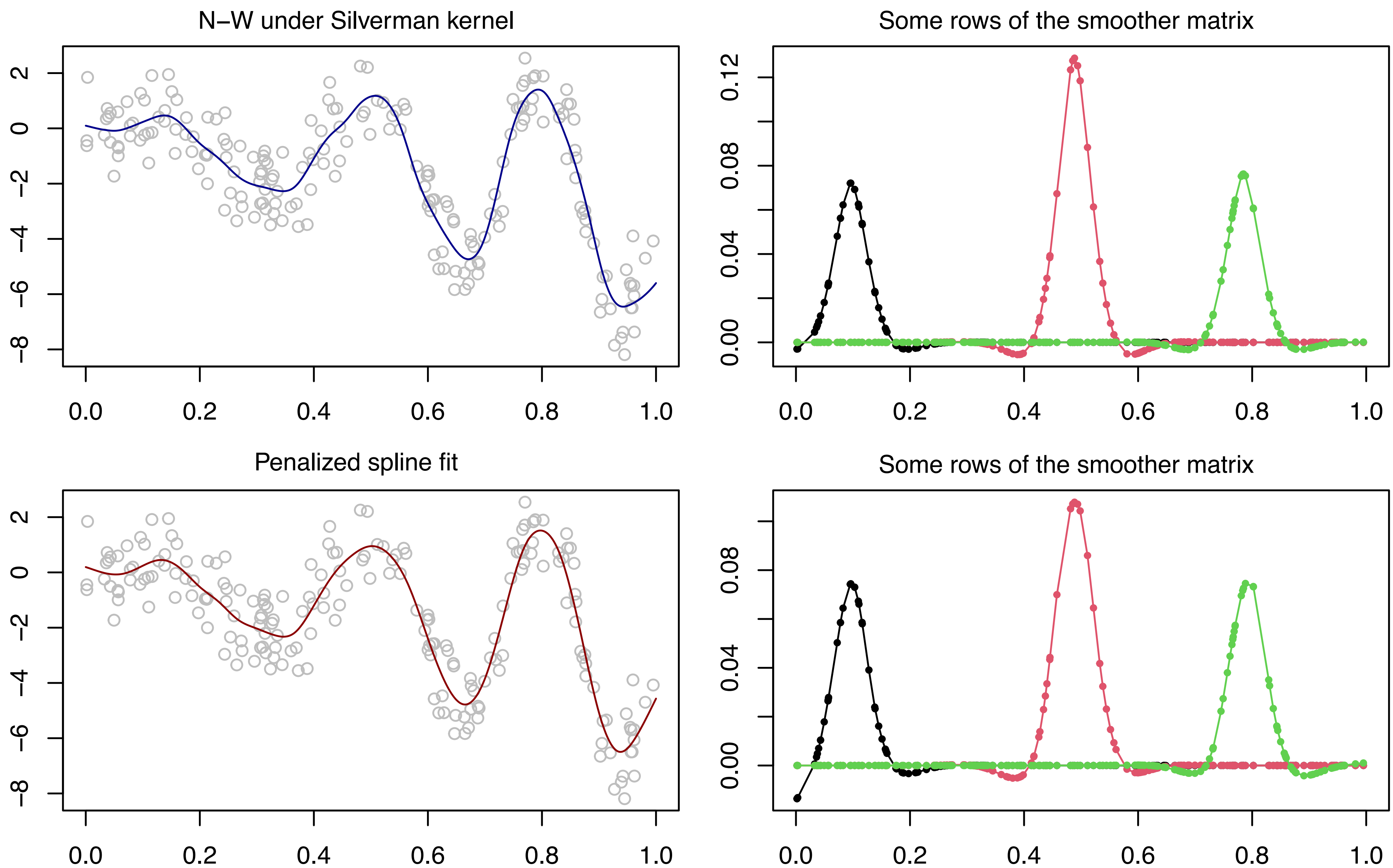
$$K(u) = \frac{1}{2} e^{-|u|/\sqrt{2}} \cdot \sin(|u|/\sqrt{2} + \pi/4) \quad \text{and} \quad h(x) = (n^{-1}\lambda/f(x))^{1/4}.$$

The previous slide shows a plot of the Silverman kernel function.

Exercise: Make some plots comparing the rows of the smoother matrices

1. of a penalized spline estimator.
2. of the N-W estimator under Silverman's kernel with bandwidth $h = (\lambda/n)^{1/4}$.

For the exercise, generate $X_1, \ldots, X_n \overset{\text{ind}}{\sim} \text{Uniform}(0,1)$.


N–W under Silverman kernel
Some rows of the smoother matrix
Penalized spline fit
Some rows of the smoother matrix
2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0
0.12
0.08
0.04
0.00

Smoother Matrix

$$\hat{\underset{\sim}{m}} = S \underset{\sim}{Y}$$

diag, eigenvalues.

Eigendecomposition: $S = U \Lambda U^T$ $\quad U^T U = I$

eigenvectors

$$\hat{\underset{\sim}{m}} = S \underset{\sim}{Y} = U \Lambda U^T Y$$

$$= U \Lambda \underbrace{(U^T U)^{-1} U^T Y}_{\text{look like LS coefficients}}$$

Consider the eigendecomposition of the smoother matrix

$$\mathbf{S} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T, \quad \mathbf{U}\mathbf{U}^T = \mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{I}_n, \quad \boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \dots, \lambda_n),$$

where $\lambda_1 \geq \lambda_2 \geq \dots \lambda_n \geq 0$ are the eigenvalues.

We see that the vector of fitted values can be written as

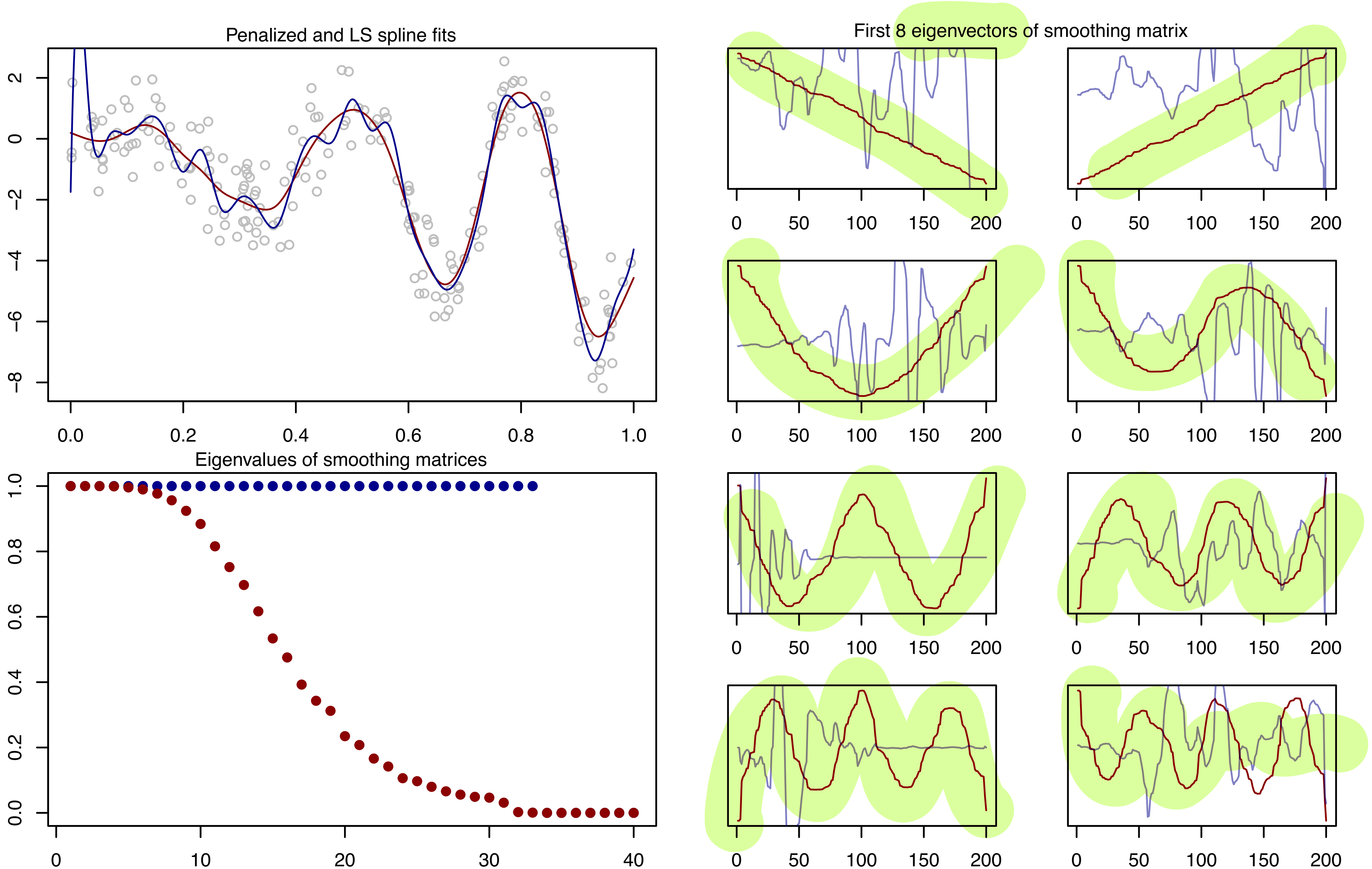
$$(\hat{m}_n^{\text{pspl}}(X_1), \dots, \hat{m}_n^{\text{pspl}}(X_n))^T = \mathbf{S}\mathbf{Y} = \mathbf{U}\boldsymbol{\Lambda}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{Y}.$$

Discuss: Can we learn anything from this?

Exercise: Inspect eigenvectors/eigenvalues of the smoother matrices

1. $\mathbf{S} = \mathbf{B}^T(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$
2. $\mathbf{S} = \mathbf{B}^T(\mathbf{B}^T\mathbf{B} + \lambda\boldsymbol{\Omega})^{-1}\mathbf{B}^T$

Make plots and discuss.



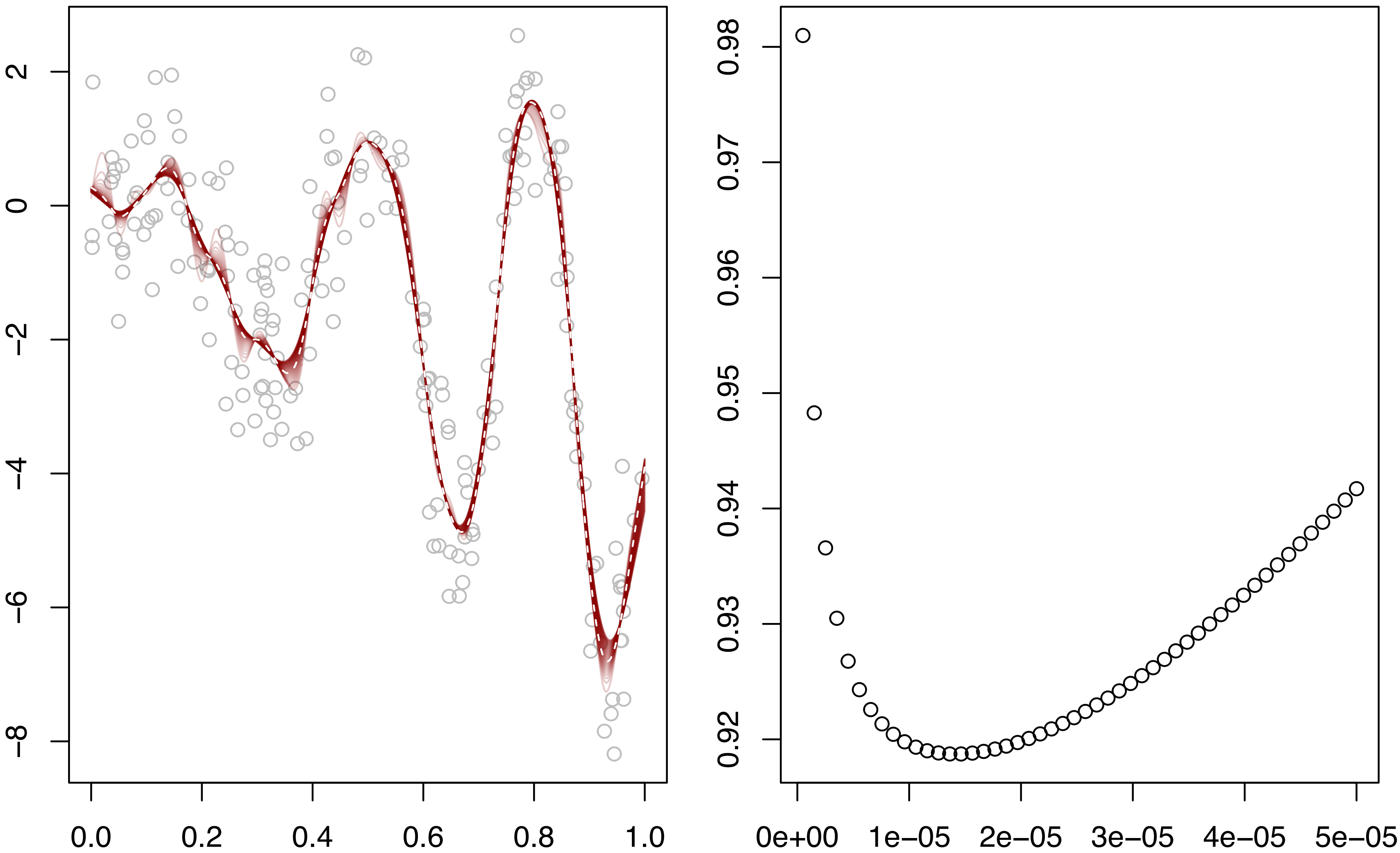
Prove: If $\underset{n\times d}{\mathbf{B}}$ full-rank, $\mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$ has exactly d nonzero eigenvals, all $= 1$.

Since $\hat{m}_n^{\text{pspl}}$ is a linear estimator, we have

$$\mathrm{CV}_n(h) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{m}_{n,-i}^{\text{pspl}}(X_i))^2 = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{Y_i - \hat{m}_n^{\text{pspl}}(X_i)}{1 - S_{ii}}\right]^2,$$

where S_{ii} is the element i on the diagonal of the smoother matrix $\mathbf{S}$.

$W_{ni}(x_i)$


2
0
−2
−4
−6
−8
0.0
0.2
0.4
0.6
0.8
1.0
0.98
0.97
0.96
0.95
0.94
0.93
0.92
0e+00
1e−05
2e−05
3e−05
4e−05
5e−05

Suppose $X_i = i/n$, $i = 1, \ldots, n$, for now, and let $\boldsymbol{\mu} = (m(X_1), \ldots, m(X_n))^T$.

Trend filtering estimator (good reference is Tibshirani's paper, [5])

The *trend filtering estimator* of μ is given by

$$\hat{\boldsymbol{\mu}} = \underset{\mathbf{u} \in \mathbb{R}^n}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{u}\|_2^2 + \lambda \|\mathbf{D}^{(k+1)}\mathbf{u}\|_1,$$

where

$$D^{(1)} = \begin{bmatrix} -1 & 1 & 0 & \ldots & 0 & 0 \\ 0 & -1 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & -1 & 1 \end{bmatrix}$$

and $D^{(k+1)} = D^{(1)} D^{(k)}$, $k \geq 1$, adjusting the dimension of $D^{(1)}$ as needed.

To estimate m at any $x \in [0, 1]$, we can just linearly interpolate $\boldsymbol{\mu}$.

Accommodate unequally spaced inputs with a modification to $\mathbf{D}^{(k+1)}$. See [5].

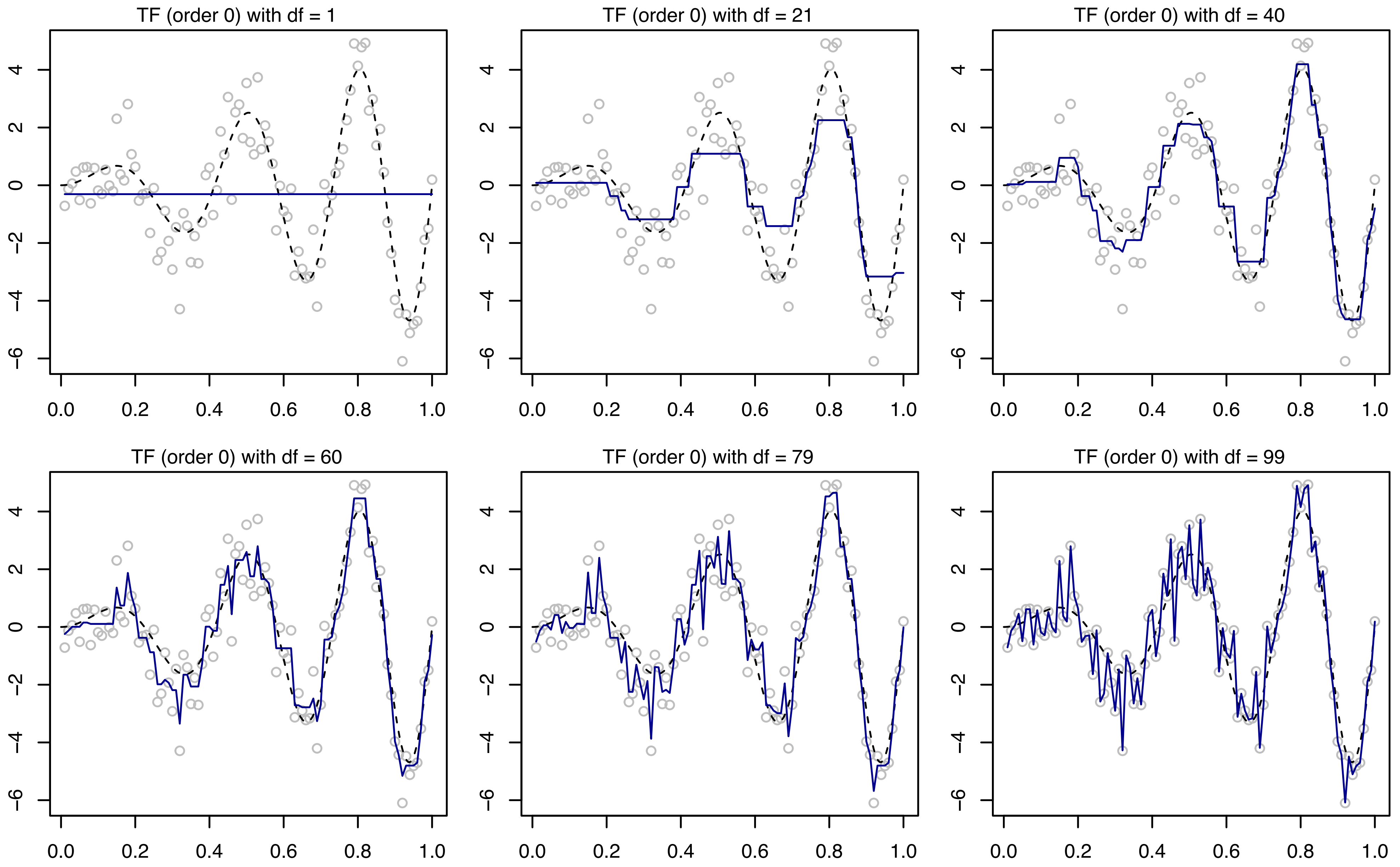
Exercise: Study the penalties $\lambda\|\mathbf{D}^{(1)}\mathbf{u}\|_1$, $\lambda\|\mathbf{D}^{(2)}\mathbf{u}\|_1$, $\lambda\|\mathbf{D}^{(3)}\mathbf{u}\|_1$, and $\lambda\|\mathbf{D}^{(4)}\mathbf{u}\|_1$ and consider their effects on $\hat{\boldsymbol{\mu}}$.

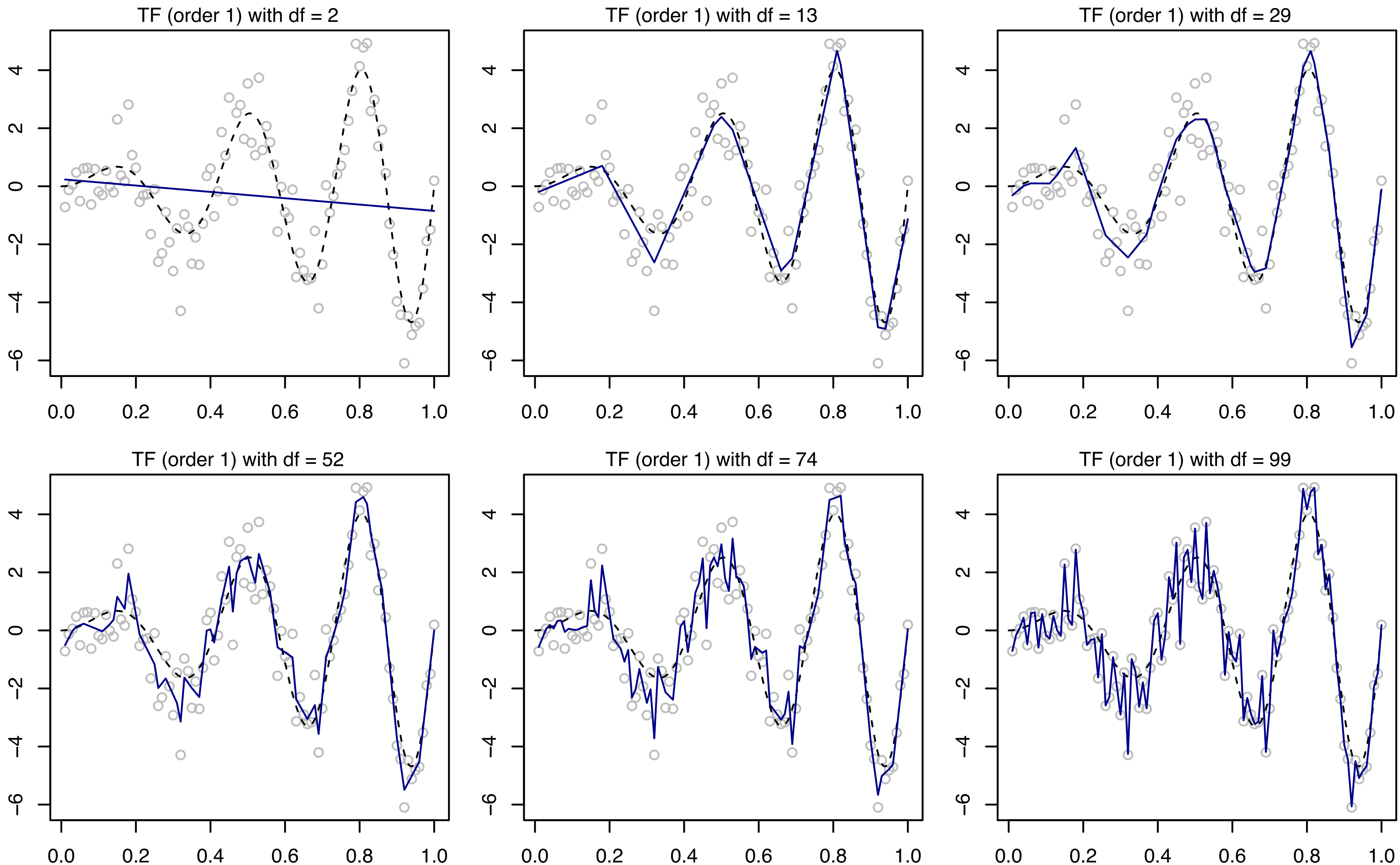
The following is known as a *generalized lasso* minimization problem:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \|\mathbf{D}\beta\|_1.$$

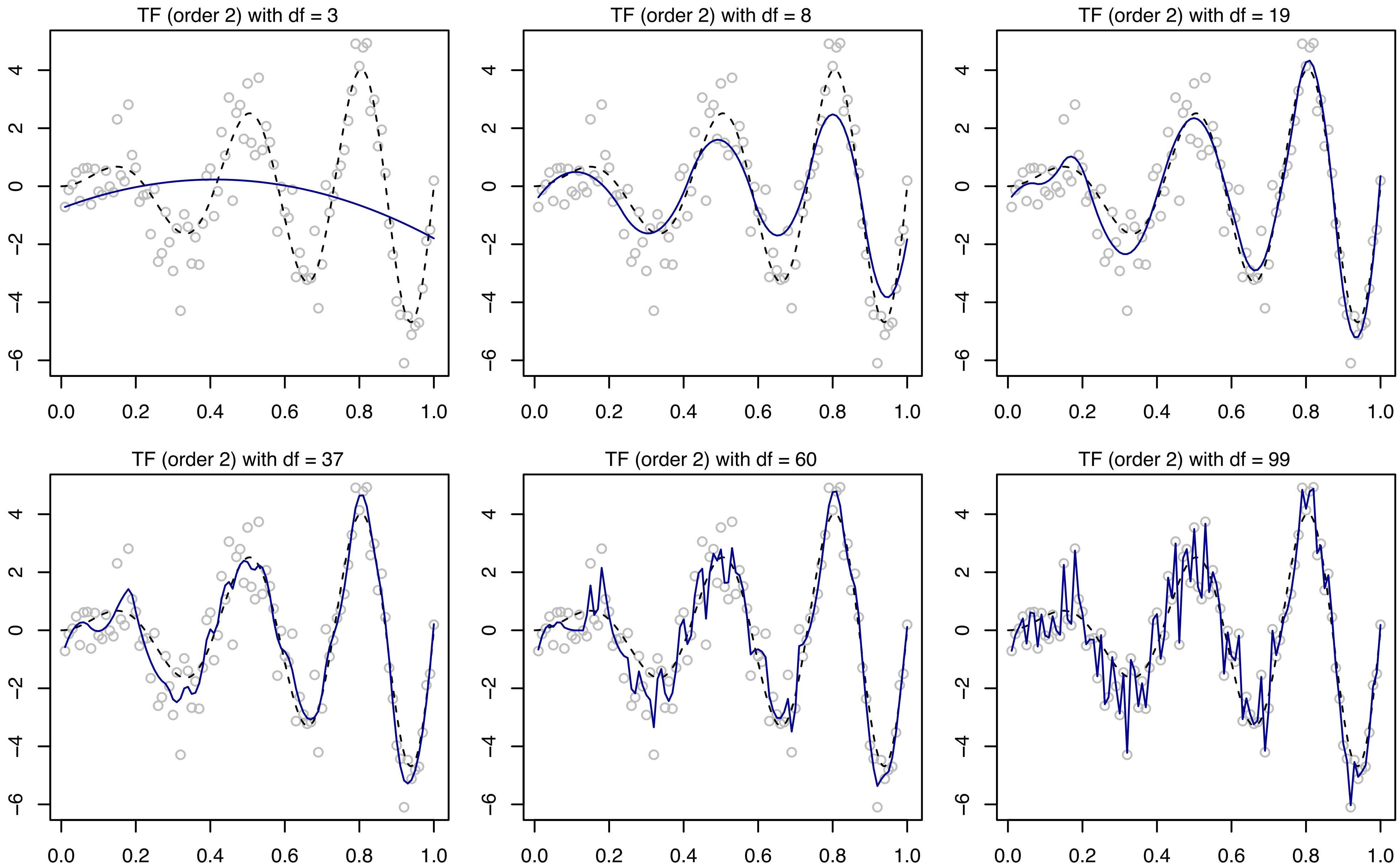
It can be solved with the `genlasso` package.

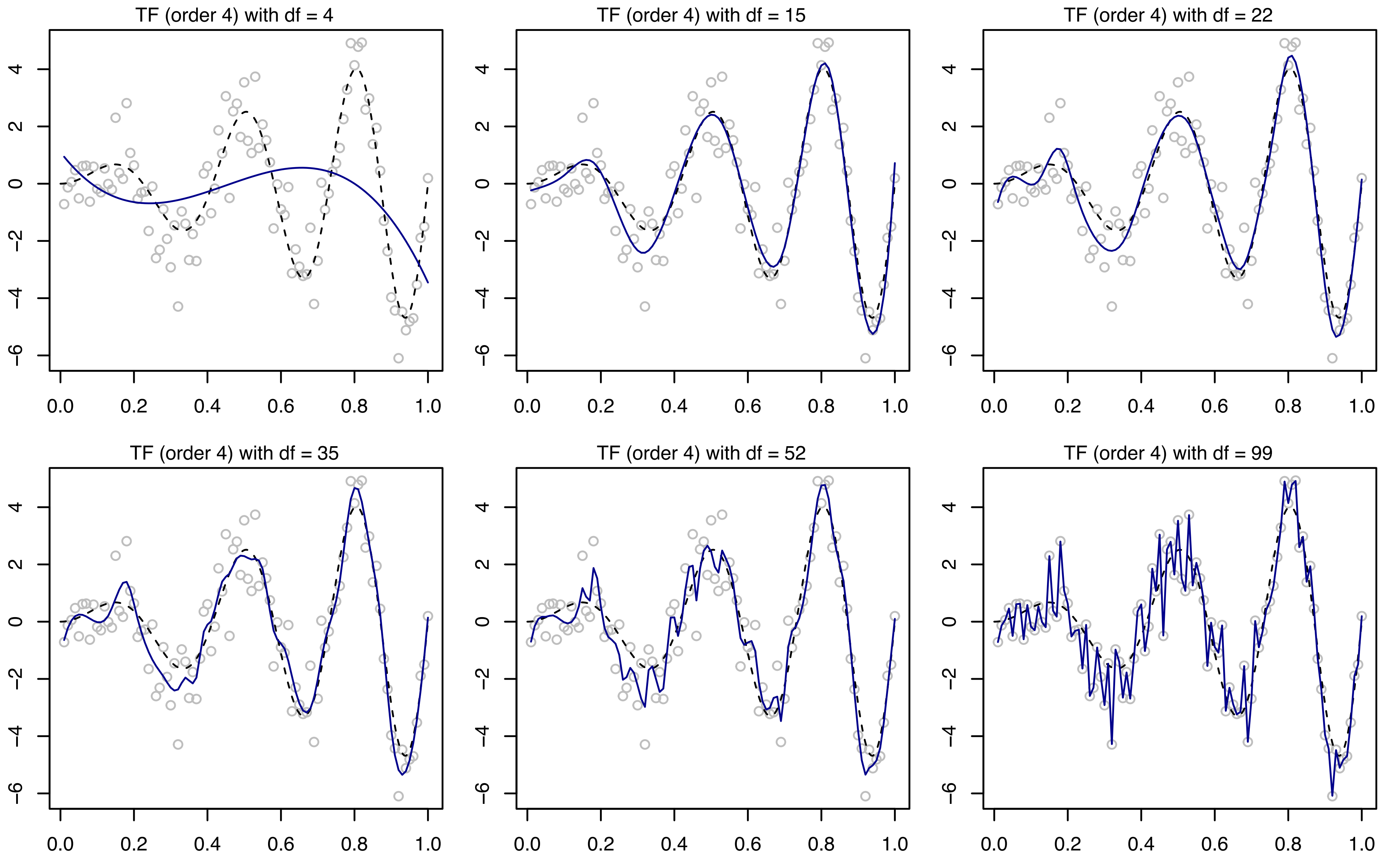
Exercise: Fit the trend filtering estimator on some data for $k = 0, 1, 2, 3$ and plot.


TF (order 0) with df = 1
TF (order 0) with df = 21
TF (order 0) with df = 40
TF (order 0) with df = 60
TF (order 0) with df = 79
TF (order 0) with df = 99



TF (order 1) with df = 2
TF (order 1) with df = 13
TF (order 1) with df = 29
TF (order 1) with df = 52
TF (order 1) with df = 74
TF (order 1) with df = 99
4
2
0
−2
−4
−6
0.0
0.2
0.4
0.6
0.8
1.0


TF (order 2) with df = 3
TF (order 2) with df = 8
TF (order 2) with df = 19
TF (order 2) with df = 37
TF (order 2) with df = 60
TF (order 2) with df = 99
−6
−4
−2
0
2
4
0.0
0.2
0.4
0.6
0.8
1.0


TF (order 4) with df = 4
TF (order 4) with df = 15
TF (order 4) with df = 22
TF (order 4) with df = 35
TF (order 4) with df = 52
TF (order 4) with df = 99
−6
−4
−2
0
2
4
0.0
0.2
0.4
0.6
0.8
1.0

Carl De Boor.
On uniform approximation by splines.
J. Approx. Theory, 1(1):219–235, 1968.

David Ruppert, Matt P Wand, and Raymond J Carroll.
Semiparametric regression.
Number 12. Cambridge university press, 2003.

Bernard W Silverman.
Spline smoothing: the equivalent variable kernel method.
The Annals of Statistics, pages 898–916, 1984.

Charles J Stone.
Additive regression and other nonparametric models.
The Annals of Statistics, pages 689–705, 1985.

Ryan J Tibshirani et al.
Adaptive piecewise polynomial estimation via trend filtering.
The Annals of statistics, 42(1):285–323, 2014.

Grace Wahba.
Spline models for observational data.
SIAM, 1990.