STAT 824 hw 03

Cox-deBoor recursion, largest eigenvalue of a matrix, smoothing and penalized splines, Lindeberg CLT, least-squares splines

1. Use the Cox-deBoor recursion formula to find the quadratic B-spline function $N_{0,2}$ based on the knots $0, 1/3, 2/3, 1$.

We obtain the function

$$N_{0,2} = \begin{cases} (9/2)u^2, & 0 \leq u < 1/3 \\ 9u - 9u^2 - 3/2, & 1/3 \leq u < 2/3 \\ (9/2)(1-u)^2, & 2/3 \leq u < 1. \end{cases}$$

2. Let $\mathbf{A}$ be a $d \times d$ matrix such that $\mathbf{A} = \sum_{j=1}^d \lambda_j u_j u_j^T$, where $u_j^T u_k = 1$ if $j = k$ and $u_j^T u_k = 0$ if $j \neq k$ and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d > 0$. Note that any $\mathbf{x} \in \mathbb{R}^d$ can be represented as $\mathbf{x} = \sum_{j=1}^d c_j u_j$, since the eigenvectors of $\mathbf{A}$ form a basis for $\mathbb{R}^d$.

   (a) Show that for any $\mathbf{x} \in \mathbb{R}^d$, we have $\frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\|\mathbf{x}\|_2^2} \leq \lambda_1$.

   We have

$$\begin{aligned} \frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\|\mathbf{x}\|_2^2} &= \frac{\sum_{j=1}^d \lambda_j \mathbf{x}^T u_j u_j^T \mathbf{x}}{\|\sum_{k=1}^d c_k u_k\|_2^2} \\ &= \frac{\sum_{j=1}^d \lambda_j (\mathbf{x}^T u_j)^2}{\sum_{k=1}^d c_k u_k^T \sum_{l=1}^d c_l u_l} \\ &= \frac{\sum_{j=1}^d \lambda_j (\sum_{k=1}^d c_k u_k^T u_j)^2}{\sum_{k=1}^d c_k^2} \\ &= \frac{\sum_{j=1}^d \lambda_j c_j^2}{\sum_{k=1}^d c_k^2} \\ &\leq \max\{\lambda_1, \ldots, \lambda_d\} \\ &= \lambda_1. \end{aligned}$$

   (b) Show that for $x = a \cdot u_1$, $a \in \mathbb{R}$, we have $\frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\|\mathbf{x}\|_2^2} = \lambda_1$.

   Using the previous work, we have

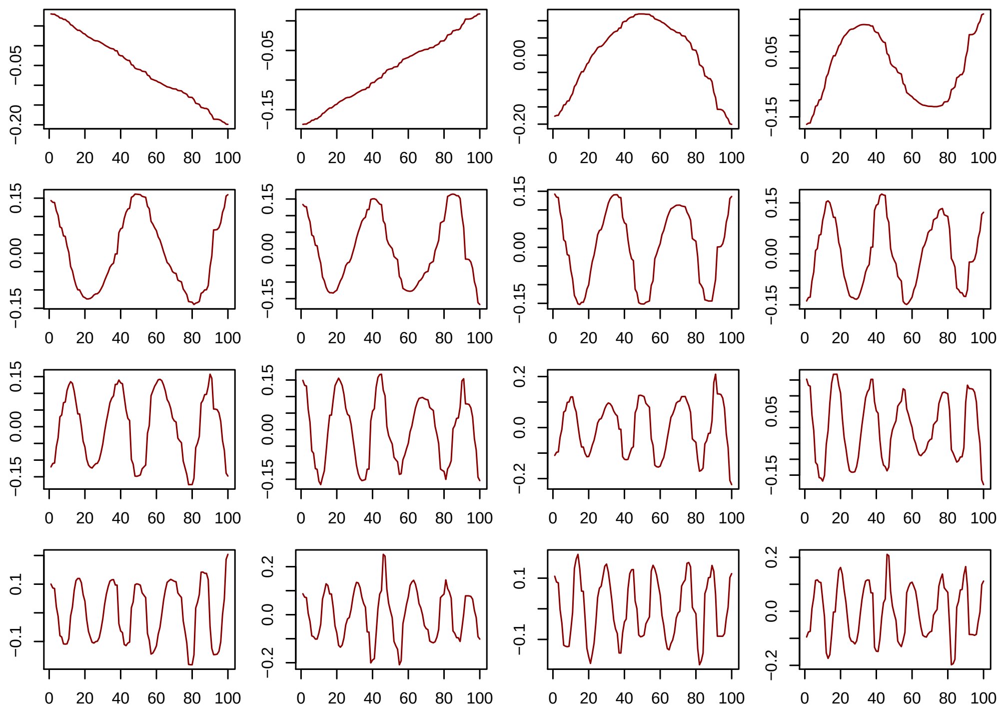
$$\frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\|\mathbf{x}\|_2^2} = \frac{\lambda_1 a^2}{a^2} = \lambda_1.$$

3. For the smoothing spline estimator

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

Green and Yandell (1985), [1], give details for computing the smoother matrix $\mathbf{S}$, which is the matrix such that $(\hat{m}_n^{\text{sspl}}(X_1), \dots, \hat{m}_n^{\text{sspl}}(X_n))^T = \mathbf{SY}$. Specifically, $\mathbf{S} = (\mathbf{I}_n + \lambda \mathbf{K})^{-1}$, with $\mathbf{K} = \mathbf{\Delta}^T \mathbf{C}^{-1} \mathbf{\Delta}$, where, for $h_i = X_{i+1} - X_i$ (assume that $X_1, \dots, X_n$ are sorted in increasing order), $\mathbf{\Delta}$ is a tridiagonal $(n-2) \times n$ matrix with $\Delta_{ii} = 1/h_i$, $\mathbf{\Delta}_{i,i+1} = -(1/h_i + 1/h_{i+1})$, $\mathbf{\Delta}_{i,i+2} = 1/h_{i+1}$, and $\mathbf{C}$ is a symmetric $(n-2) \times (n-2)$ tridiagonal matrix with $\mathbf{C}_{i-1,i} = \mathbf{C}_{i,i-1} = h_i/6$ and $C_{ii} = (h_i + h_{i+1})/3$.

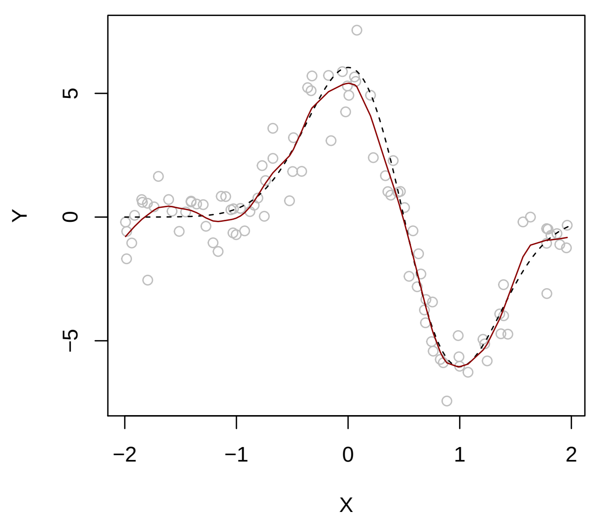
(a) Generate $X_1, \dots, X_n \overset{\text{ind}}{\sim} \text{Uniform}(-2, 2)$ for $n = 100$, compute the matrix $\mathbf{S}$, and then plot the first 16 eigenvectors. The plot should look something like this:



(b) Now generate $Y_i = m(X_i) + \varepsilon_i$, $i = 1, \dots, n$, where m is a function of your choosing. Then make a scatter plot of your $(X_1, Y_1), \dots, (X_n, Y_n)$ values with the true function overlaid. Then plot the values $\hat{m}_n^{\text{sspl}}(X_1), \dots, \hat{m}_n^{\text{sspl}}(X_n)$ against $X_1, \dots, X_n$, for some value of λ that makes the estimator look close to the true function.

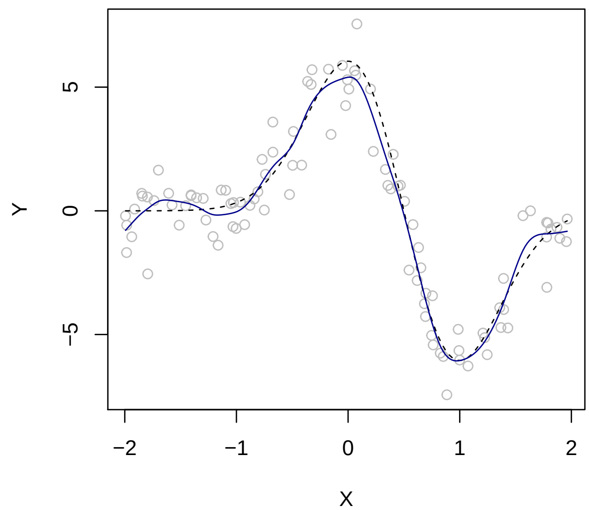
```
m <- function(x){ - 50 * (x - 1/2) * dnorm(2*(x - 1/2))}
n <- 100
X <- sort(runif(n,-2,2))
Y <- m(X) + rnorm(n)

# Construct the smoother matrix for smoothing splines:
```

```
h <- diff(X)
Delta1 <- cbind(diag(1/h[-c(n-1,n)]),matrix(0,n-2,2))
Delta2 <- cbind(matrix(0,n-2,1),-diag(1/h[-c(n-1,n)]+1/h[-c(1,n)]),matrix(0,n-2,1))
Delta3 <- cbind(matrix(0,n-2,2),diag(1/h[-c(1,n)]))
Delta <- Delta1 + Delta2 + Delta3

Cdiag <- diag( (h[-(n-1)] + h[-1])/3 )
C0 <- diag( h[-(n-1)]/6 )
C1 <- rbind(C0[-1,],rep(0,n-2))
C2 <- cbind(C0[,-1],rep(0,n-2))
C <- Cdiag + C1 + C2

K <- t(Delta) %*% solve(C) %*% Delta
lambda <- .2/n
S <- solve((diag(n) + lambda * K))
Y.hat <- S %*% Y

plot(Y~X,
     col = "gray",
     ylim = range(m(x),Y))
x <- seq(-2,2,length = 199)
lines(m(x)~x, lty = 2)
lines(Y.hat ~ X, col = rgb(.545,0,0))
```



(c) On the same data, fit a penalized spline estimator with the same λ value and some fairly large number of knots (you can choose). Compare the fit of the smoothing splines and the penalized splines estimator.

The plot of my penalized splines estimator looks like this (I chose $K_n = 40$):



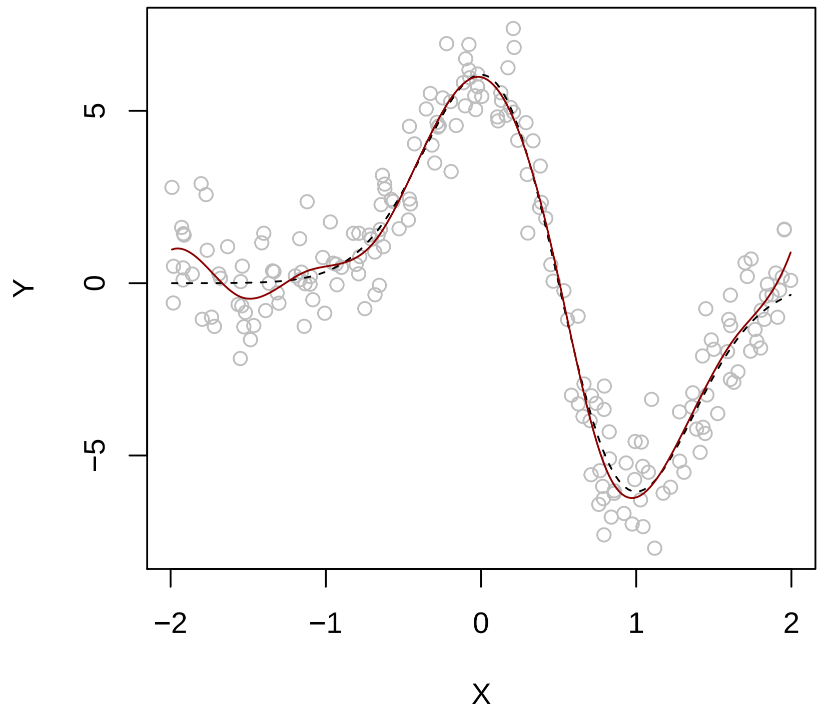
This is very nearly identical to the smoothing splines estimator.

4. For each $n \geq 1$, let $Y_i = x_i\beta + \varepsilon_i$, $i = 1, \dots, n$, where $\varepsilon_1, \dots, \varepsilon_n$ are iid with $\mathbb{E}\varepsilon_1 = 0$ and $\operatorname{Var}\varepsilon_1 = \sigma^2 < \infty$ and $x_1, \dots, x_n$ are deterministic, and let $\hat{\beta}_n = \sum_{i=1}^n x_i Y_i / \sum_{i=1}^n x_i^2$. Use the corollary to the Lindeberg Central Limit Theorem given in Lec 04 to show that
$$\frac{\max_{1\leq i\leq n}|x_i|}{\sqrt{\sum_{i=1}^n x_i^2}} \to 0 \text{ as } n \to \infty$$
implies $\sqrt{n}(n^{-1}\sum_{i=1}^n x_i^2)^{1/2}(\hat{\beta}_n - \beta)/\sigma \to N(0,1)$ in distribution as $n \to \infty$.

We find that we may write
$$\sqrt{n}(n^{-1}\sum_{i=1}^n x_i^2)^{1/2}(\hat{\beta}_n - \beta)/\sigma = \frac{\sum_{i=1}^n x_i(\varepsilon_i/\sigma)}{\sqrt{\sum_{i=1}^n x_i^2}},$$
to which the corollary to the Lindeberg Central Limit Theorem directly applies.

5. For a sample size of $n = 200$, generate $Y_i = m(X_i) + \varepsilon_i$, for $i = 1, \dots, n$, where $\varepsilon_1, \dots, \varepsilon_n \overset{\text{ind}}{\sim} \text{Normal}(0,1)$, $X_1, \dots, X_n \overset{\text{ind}}{\sim} \text{Uniform}(-2, 2)$, and with $m(x) = -50(x - 1/2)\phi(2(x - 1/2))$.
   (a) Construct an estimate of m with a least squares splines estimator using cubic B splines basis functions; choose some number K_n of intervals into which to subdivide the range of the covariate values, and position the knots at equally spaced quantiles of $X_1, \dots, X_n$. Plot your estimator of m as well as the true function on a scatterplot of the (X, Y) values.

The plot should look something like this (I used $K_n = 10$):



(b) The number of intervals K_n into which we break the range of the covariate values plays an important role in least-squares splines estimation. Run a simulation: On each of 500 simulated data sets, build a 95% confidence interval for $m(x_0)$ at the point $x_0 = 0$ based on your least-squares splines estimator under $K_n = 1, \dots, 15$. So for each data set you will have 15 confidence intervals. Record for each choice of K_n the proportion of times the confidence interval contained the true value of $m(x_0)$ as well as the average width of the confidence intervals across the 500 data sets. Arrange your results in a table like the one below (this is the table I got, so your numbers should be fairly close to these):

K_n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
coverage	0.00	0.00	0.00	0.24	0.05	0.72	0.65	0.87	0.93	0.94	0.96	0.95	0.95	0.95	0.94
average width	0.40	0.53	0.50	0.62	0.59	0.71	0.69	0.79	0.80	0.87	0.90	0.95	0.98	1.03	1.06

(c) Why does the average width keep getting wider as K_n increases?

Intuitively, when we make K_n larger we introduce more "parameters" to the model—more coefficients to estimate, so the variability will be larger. Also, recall that the variance of the least-squares splines estimator is like a constant times K_n/n, so as K_n increases, the variance of the estimator increases, which is reflected in wider confidence intervals.

(d) Why does the coverage start out too low and then stabilize around 0.95 as K_n increases?

We are seeing the bias vanish as K_n increases. Recall that the bias of the least-squares splines estimator is like a constant times $K_n^{-\beta}$, where $\beta > 0$ describes the smoothness of the function. So as K_n increases, the bias decreases, and the confidence interval centers itself at a height closer to that of the true function.

References

[1] Peter J Green and Brian S Yandell. Semi-parametric generalized linear models. In *Generalized linear models*, pages 44–55. Springer, 1985.