CLEMSON UNIVERSITY

Lecture 6

Non-parametric Regression and Shrinkage Methods

Reading: Faraway 2014 Chapters 9.5-9.6 and 11.3-11.4; Faraway 2016 Chapters 14.1-14.2, 14.6; 15.2-15.3; ISLR 2021 Chapters 6.2 and 7.3-7.5, 7.7

STAT 8020 Statistical Methods II

Whitney Huang
Clemson University

Agenda

1. Non-parametric Regression
2. Ridge Regression
3. LASSO

Moving Away From Linear Regression

- We have mainly focused on linear regression so far

  Model: $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \overset{i.i.d.}{\sim} \mathrm{N}(0, \sigma^2)$

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y} \Rightarrow \hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}} = \underbrace{\boldsymbol{X}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}}_{\boldsymbol{H}\text{: "Hat" matrix}}\boldsymbol{y}$$

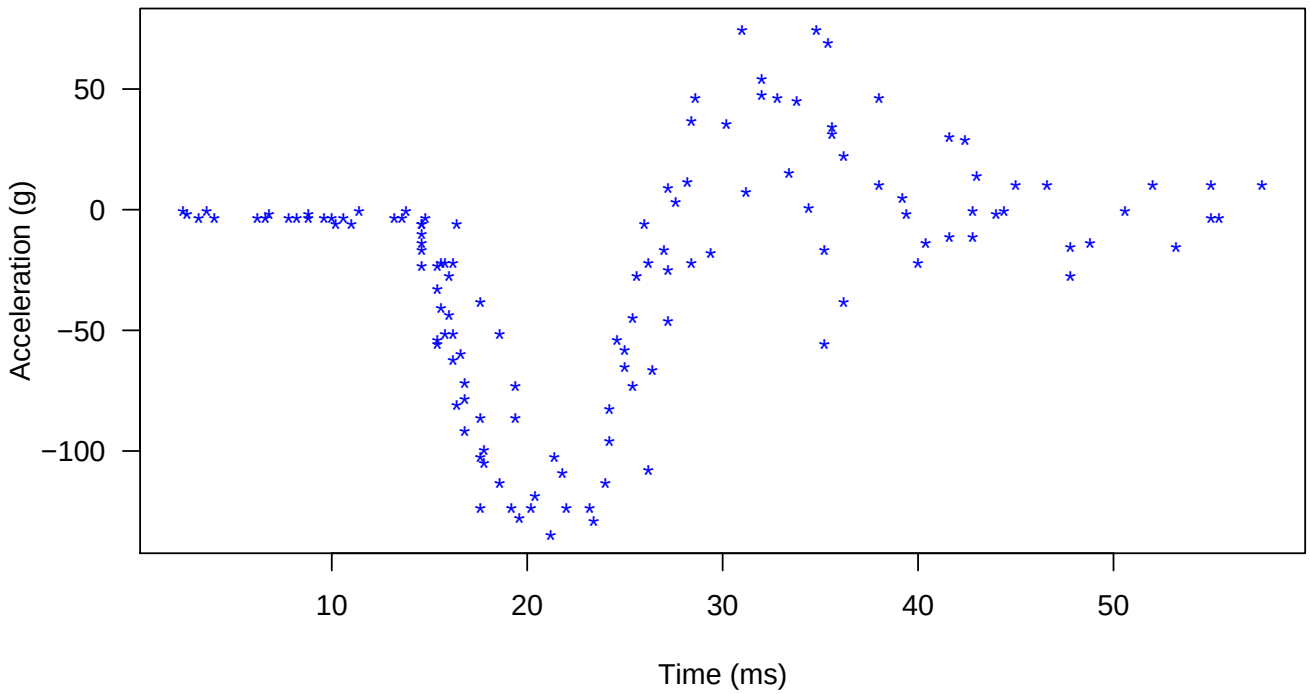
- Parametric: fixed form vs Nonparametric: learn from data (no pre-specified form)

  Model: $y = f(x) + \varepsilon \Rightarrow \mathrm{E}[y|x] = f(x) \Rightarrow$ estimate it flexibly from data

  - The (smooth) function $f(x)$ must be represented somehow
  - The degree of smoothness of $f(x)$ must be made controllable
  - Some means for estimating the most appropriate degree of smoothness from data is required

Data from a Simulated Motorcycle Accident [Silverman, 1985)]

This data set is taken from a simulated motor-cycle crash experiment in order to study the efficacy of crash helmets.



Question: How can we estimate the smooth function $f(x)$ without specifying its form?

Kernel Estimators

$$\hat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) y_i = \frac{1}{n}\sum_{i=1}^{n} w_i y_i, \quad w_i = \frac{1}{h} K\left(\frac{x - x_i}{h}\right)$$

Idea:

- Estimate $f(x)$ using **nearby observations**
- Closer points → **higher weight**
- Bandwidth h controls **smoothness**

Key components:

- Kernel $K(\cdot)$: smooth, integrates to 1 e.g., Epanechnikov kernel

$$K(x) = \begin{cases} \frac{3}{4}(1 - x^2), & |x| < 1 \\ 0, & \text{otherwise} \end{cases}$$

- Bandwidth h: smoothing parameter small $h \to$ rough; large $h \to$ oversmooth

Representing $f(x)$ via Basis Functions: Key Idea

Idea:

- Instead of estimating $f(x)$ directly,
- Represent it as a combination of simple functions:

$$f(x) = \sum_{j=1}^{J} \beta_j b_j(x)$$

Interpretation:

- Converts a **nonlinear problem** $\rightarrow$ **linear regression**
- Choice of basis functions controls **flexibility**

Common Choices of Basis Functions

- Examples:
  - Polynomials
  - Fourier series
  - Radial basis functions
- We focus on splines

Why splines?

- Global polynomials lack flexibility
- Splines provide **local flexibility** + **smoothness**

Cubic Splines: Flexible and Smooth

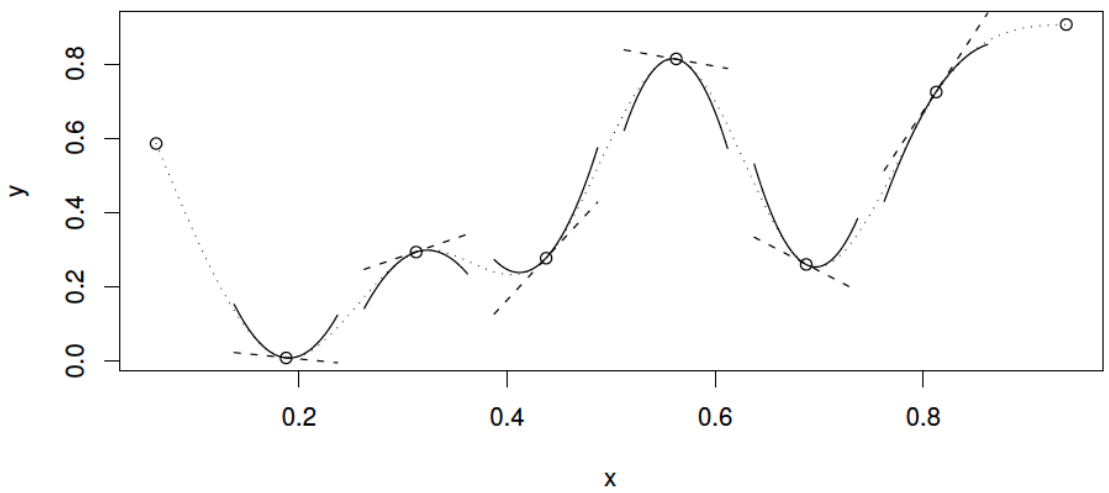


Figure 3.3 *A cubic spline is a curve constructed from sections of cubic polynomial joined together so that the curve is continuous up to second derivative. The spline shown (dotted curve) is made up of 7 sections of cubic. The points at which they are joined (○) (and the two end points) are known as the* knots *of the spline. Each section of cubic has different coefficients, but at the knots it will match its neighbouring sections in value and first two derivatives. Straight dashed lines show the gradients of the spline at the knots and the curved continuous lines are quadratics matching the first and second derivatives at the knots: these illustrate the continuity of first and second derivatives across the knots. This spline has zero second derivatives at the end knots: a* 'natural *spline'.*

Source: Simon Wood, *Generalized Additive Models*, p. 122, Fig. 3.3

Key idea:

- Piecewise polynomials joined smoothly at **knots**
- Flexible but still well-behaved

Regression Splines

- Choose knot locations $\{\xi_j\}_{j=1}^{J}$ to form basis $\boldsymbol{X}$
- Fit model using **linear regression tools**

Bias–Variance Tradeoff:

- Few knots → too rigid (high bias)
- Many knots → too wiggly (high variance)

Takeaway: Model flexibility is controlled by the number and placement of knots

Limitations of Regression Splines

- Not fully **nonparametric**:
  - Must choose number of knots J
  - Must choose knot locations $\{\xi_j\}_{j=1}^J$
- These choices can strongly affect the fit
- Selecting the "right" level of smoothness, determined by $\{\xi_j\}_{j=1}^J$, is **difficult**
- **Idea:** control smoothness via penalization [Green and Silverman 1993]

Controlling Smoothness via Penalization

Objective: Minimize

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int [f''(x)]^2 dx$$

Interpretation:

- First term: **fit to data**
- Second term: **penalty on curvature (wiggliness)**

Role of λ:

- $\lambda = 0 \rightarrow$ interpolates data (very wiggly)
- $\lambda \rightarrow \infty \rightarrow$ linear regression (very smooth)

λ controls the **bias–variance tradeoff** $\Rightarrow$ **selecting an appropriate λ is crucial**

Key Result: Smoothing Splines

Theorem: The function that minimizes

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int [f''(x)]^2 \, dx$$

is a natural cubic spline with knots at all observed x_i.

Implications:

- No need to choose knot locations explicitly
- Smoothness is controlled entirely by λ

Result: Smoothing splines (Wahba, 1990)

Smoothing Splines: Basis Representation

- Represent regression function as:

$$f(x) = \sum_{j=1}^{n} N_j(x)\beta_j,$$

where Design matrix: $\boldsymbol{N}_{ij} = N_j(x_i)$

- Objective function:

$$(\boldsymbol{y} - \boldsymbol{N}\boldsymbol{\beta})^T(\boldsymbol{y} - \boldsymbol{N}\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}^T\boldsymbol{\Omega}\boldsymbol{\beta},$$

where $\boldsymbol{\Omega}_{jk} = \int N_j''(x)N_k''(x)\,dx$

- The minimizer:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{N}^T\boldsymbol{N} + \lambda\boldsymbol{\Omega})^{-1}\boldsymbol{N}^T\boldsymbol{y}$$

Smoothing Splines are Linear Smoothers

Fitted values:

$$\hat{\boldsymbol{y}} = \hat{f}(\boldsymbol{x}) = \boldsymbol{N}\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{N} + \lambda\boldsymbol{\Omega}\right)^{-1}\boldsymbol{N}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{L}_{\lambda}\boldsymbol{y}$$

where

$$\boldsymbol{L}_{\lambda} = \boldsymbol{N}\left(\boldsymbol{N}^{\mathrm{T}}\boldsymbol{N} + \lambda\boldsymbol{\Omega}\right)^{-1}\boldsymbol{N}^{\mathrm{T}}$$

is the **smoothing matrix**

Key insight:

- $\hat{\boldsymbol{y}}$ is a **linear function of the data** $\boldsymbol{y}$
- **Degrees of freedom:**

$$\mathrm{df} = \mathrm{tr}(\boldsymbol{L}_{\lambda})$$

measures **model flexibility**

Choosing λ: Cross-Validation

Idea:

- Leave one observation out
- Predict it using the remaining data
- Choose λ that minimizes prediction error

Efficient CV formula:

$$\text{CV}(\lambda) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_{-i})^2 = \frac{1}{n}\sum_{i=1}^{n}\frac{(y_i - \hat{y}_i)^2}{(1 - L_{\lambda,ii})^2}$$

Generalized CV (GCV):

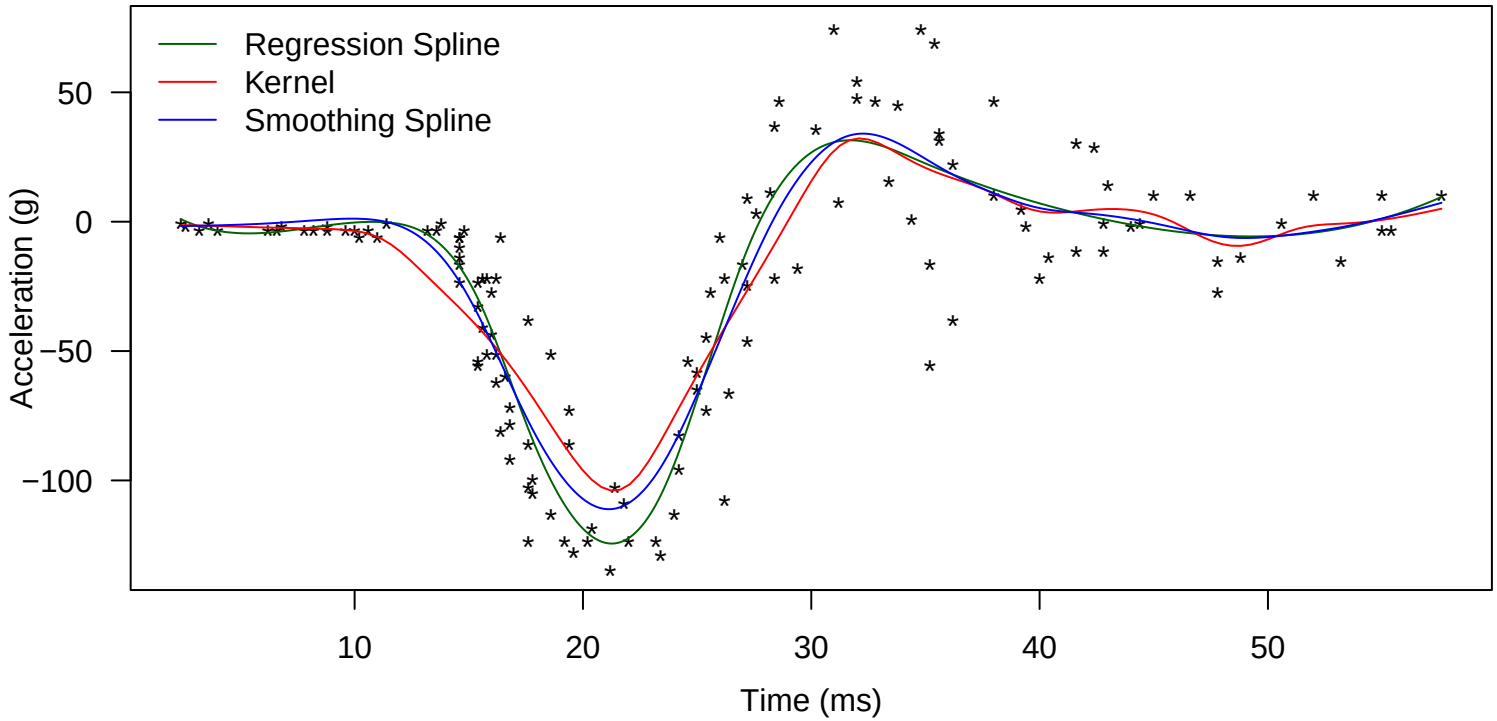
$$\text{GCV}(\lambda) = \frac{1}{n}\frac{\sum(y_i - \hat{y}_i)^2}{\left(1 - \frac{\text{tr}(\boldsymbol{L}_\lambda)}{n}\right)^2}$$

Non-parametric Regression Fits for Motorcycle Data

Regression Spline: 10 degrees of freedom quantile knot

Smoothing Spline: the amount of smoothness is estimated from the data by GCV

Kernel Regression: K: Epanechnikov kernel and $h = 5$



Generalized Additive Models for Multiple Predictors

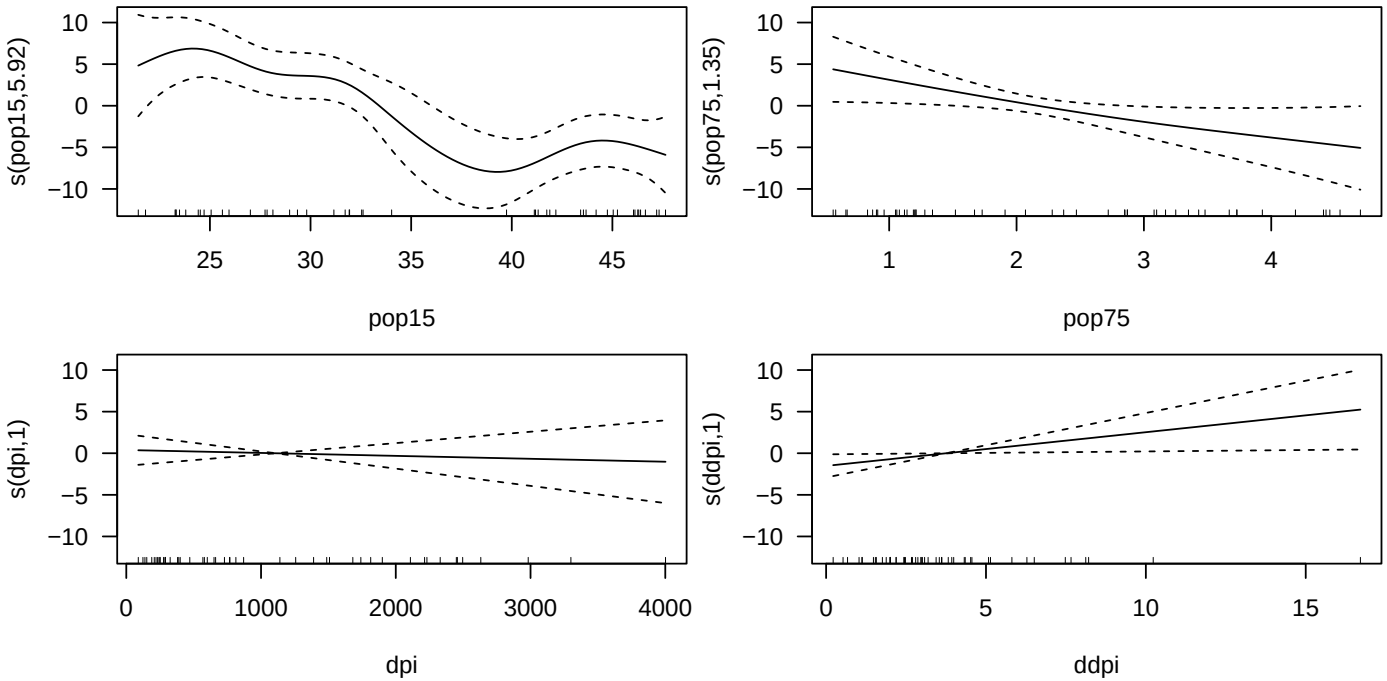
Problem: Full nonparametric model:

$$y = f(x_1, \ldots, x_p) + \varepsilon$$

Challenge: suffers from the "curse of dimensionality"
One solution: Generalized Additive Model (GAM)

$$y = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_p(x_p) + \varepsilon$$

- Each predictor has its own smooth function
- Additive structure reduces complexity



Shrinkage Methods: Motivation

$$y = \beta_0 + \sum_{j=1}^{p-1} \beta_j x_j + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$$

Question: What if p is large?

- **Collinearity** → unstable estimates
- **Overfitting** → poor prediction

Idea: shrink coefficients toward 0 to improve stability and prediction

Methods: Ridge and LASSO

Ridge Regression [Hoerl & Kennard, 1970]

Idea: penalize large coefficients

- Estimate:

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_j \beta_j x_{ij})^2 + \lambda \sum_j \beta_j^2$$

- Closed-form solution:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \boldsymbol{X} + \lambda \boldsymbol{I})^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

- Works well when predictors are **highly correlated**

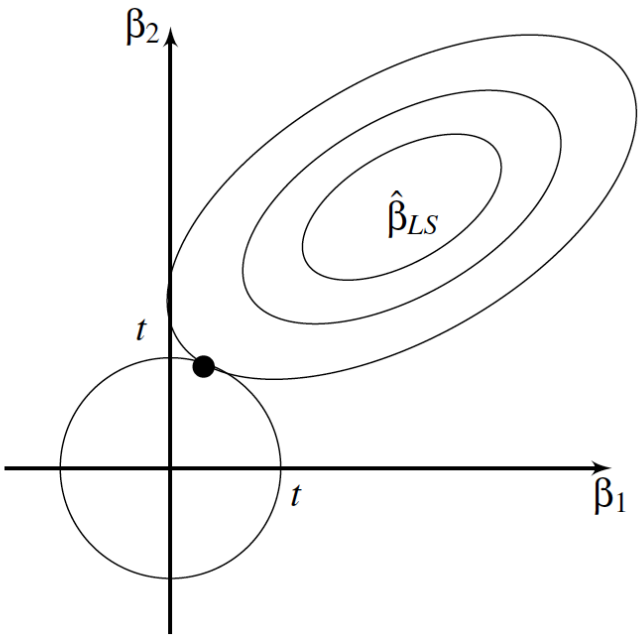
λ controls **shrinkage strength**

Ridge Regression: Geometric View

Ridge regression can also be solved by choosing β to minimize

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p-1}\beta_j x_{ij})^2$$

subject to $\sum_{j=1}^{p}\beta_j^2 \leq t^2$

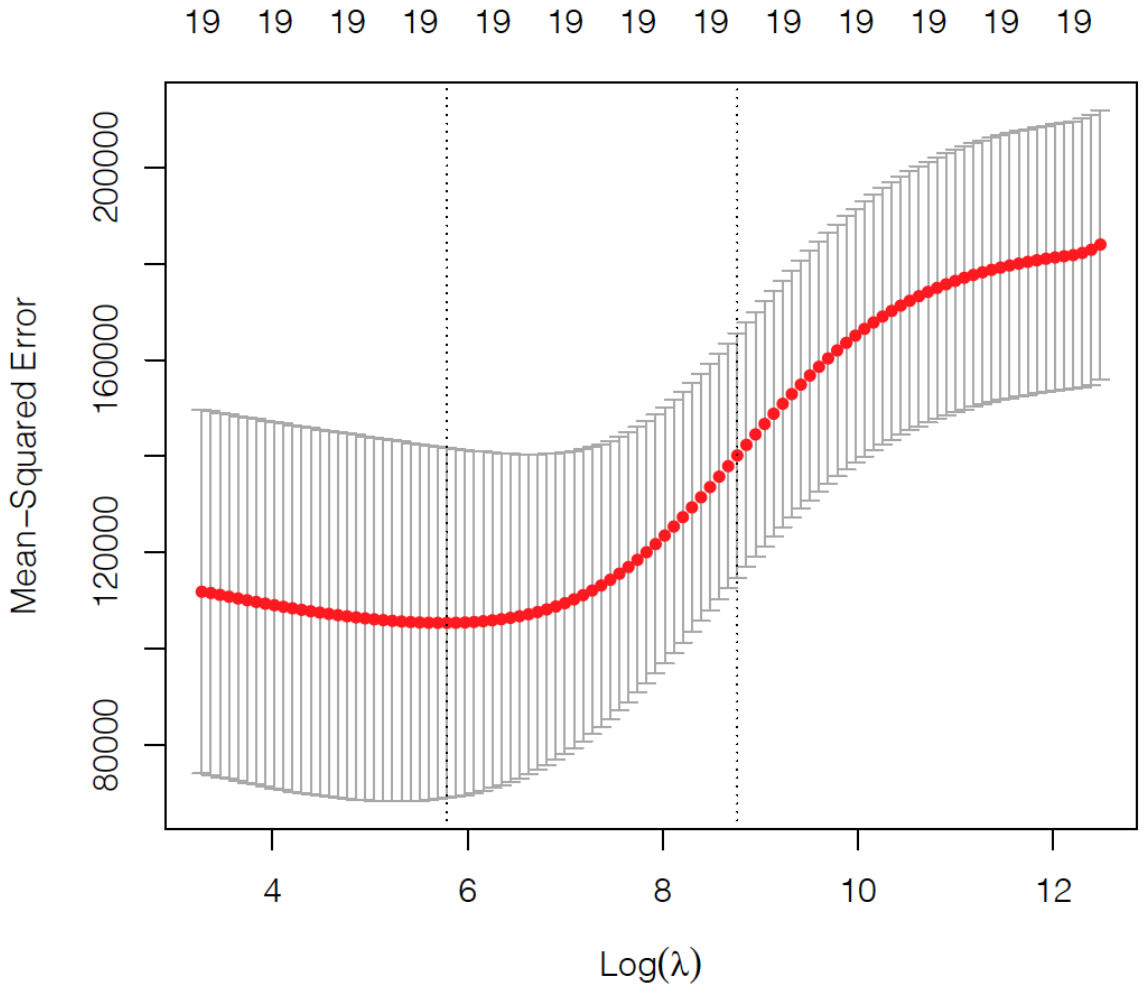


Source: p. 175, Fig. 11.9 *Linear Models with* R, Faraway, 2014

Takeaway:

- Circular constraint → **shrinks coefficients smoothly**
- Rarely sets coefficients exactly to zero
- Keeps **all predictors** in the model

Choosing λ via Cross-Validation



Takeaway:

- Choose λ minimizing CV error
- Left: overfit (small λ)
- Right: oversmooth (large λ)

Least Absolute Shrinkage and Selection Operator (LASSO)
Tibshirani, 1996

Idea: assume **sparsity** (few important predictors)

- Estimate:

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \beta_0 - \sum_j \beta_j x_{ij})^2 + \lambda \sum_j |\beta_j|$$

- No closed-form solution $\Rightarrow$ requires numerical optimization (e.g., coordinate descent)
- Produces **sparse models**: some $\hat{\beta}_j = 0$

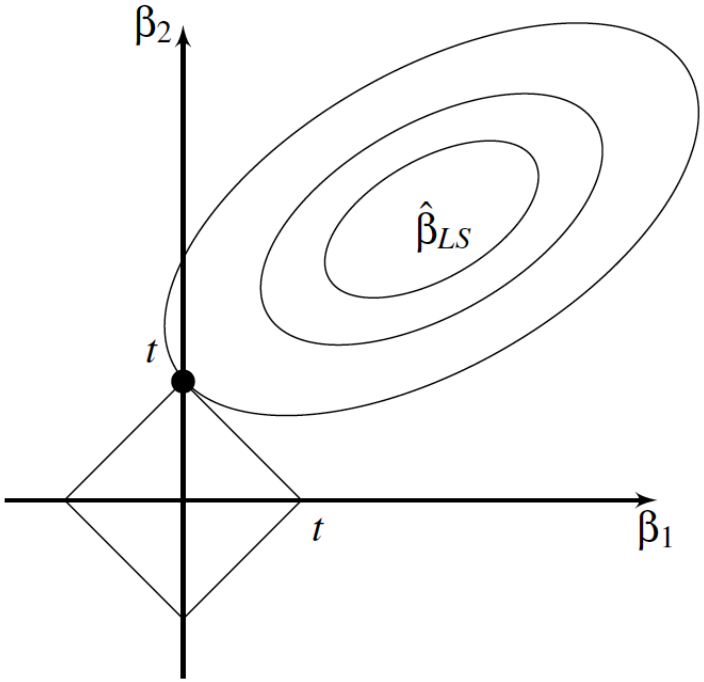
LASSO = **shrinkage + variable selection**

LASSO: Geometric View

LASSO can also be solved by choosing β to minimize

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p-1}\beta_j x_{ij})^2$$

subject to $\sum_{j=1}^{p}|\beta_j| \leq t$

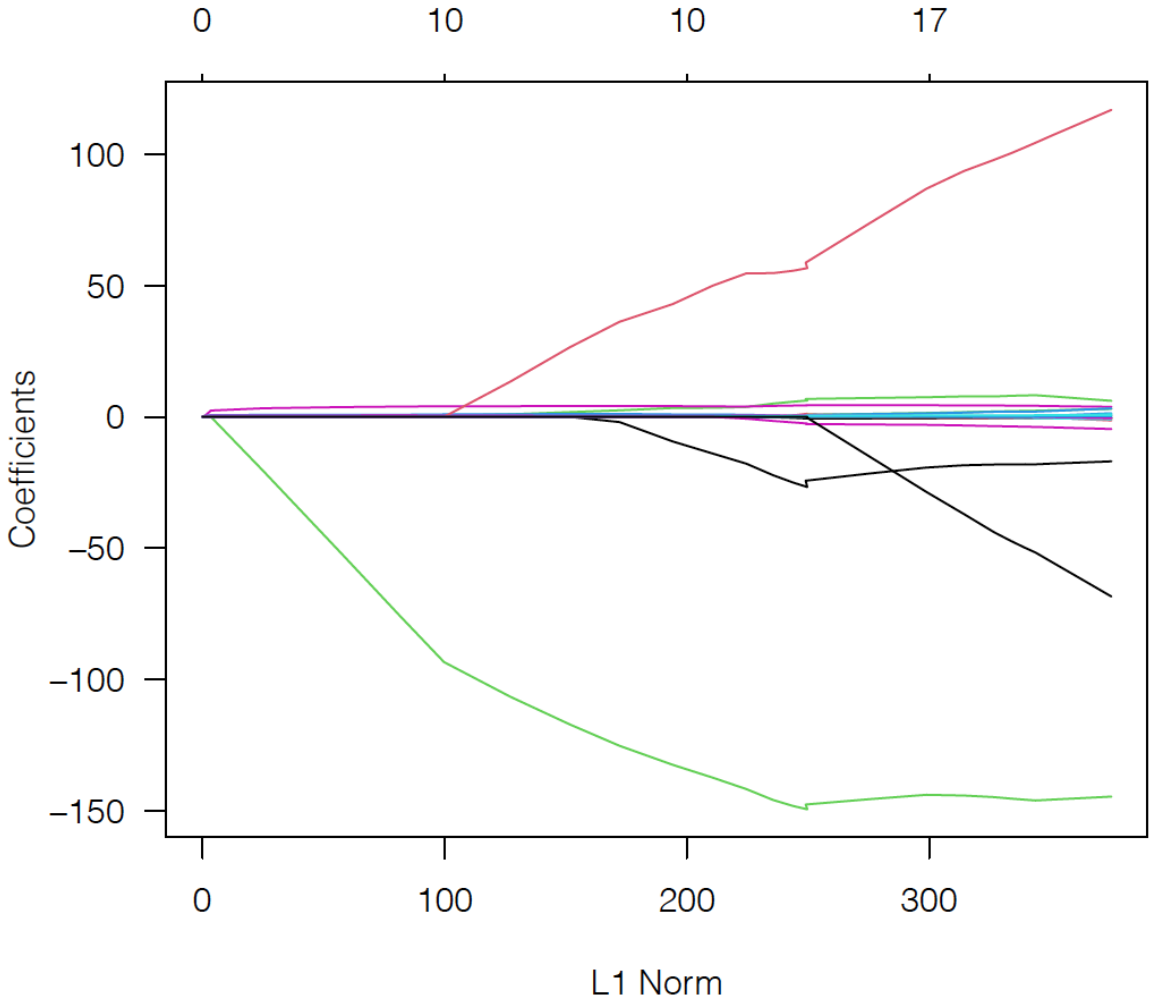


Takeaway:

- Diamond constraint → **corners**
- Corners → coefficients hit **exactly zero**
- Enables **variable selection**

Source: p. 175, Fig. 11.9 *Linear Models with* R, Faraway, 2014

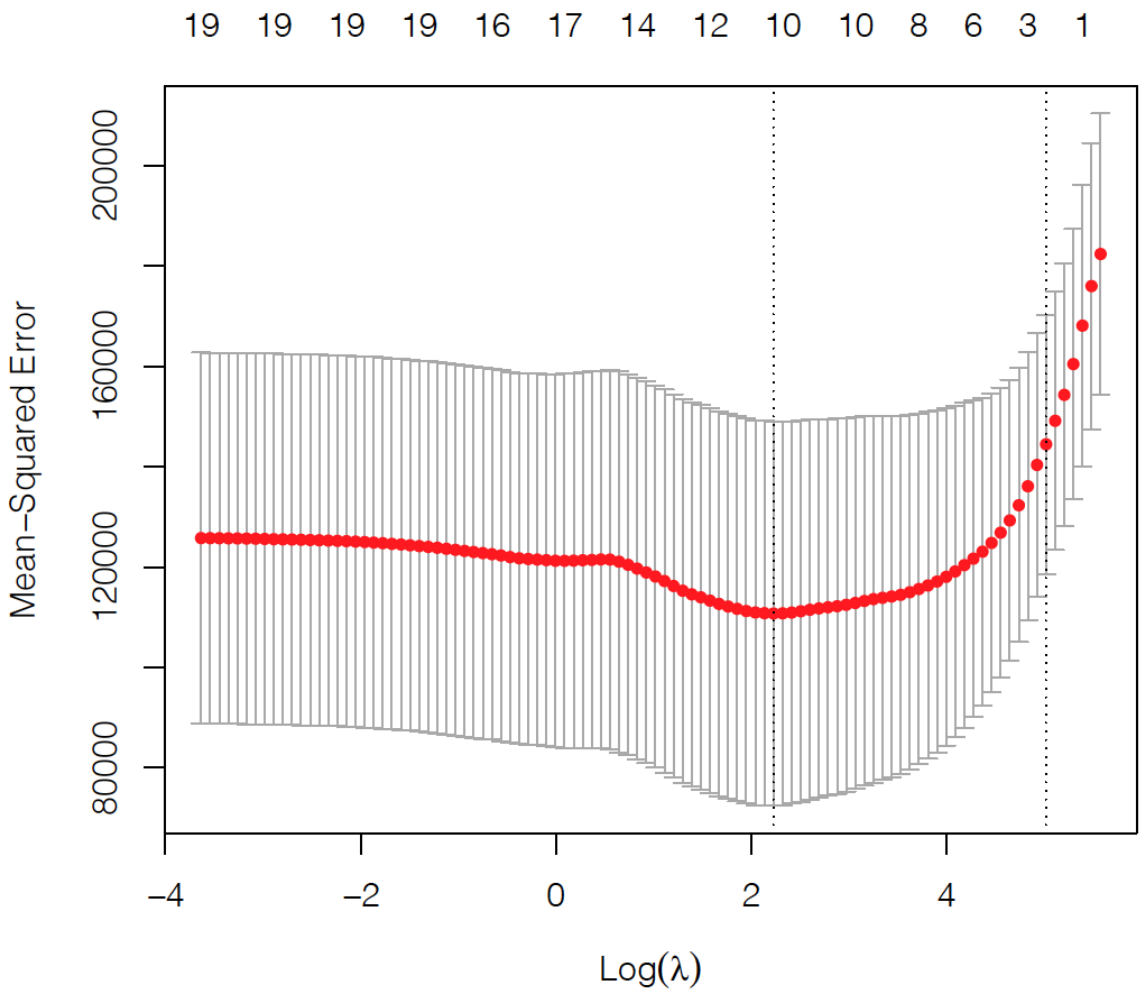
LASSO Path



Takeaway:

- Left: strong penalty → all coefficients ≈ 0
- Right: weaker penalty → more variables enter
- LASSO builds the model **progressively**

Selecting λ (LASSO)



Takeaway: Choose λ minimizing CV error

Ridge vs LASSO

- **Ridge:**
  - Shrinks coefficients
  - Keeps all predictors
- **LASSO:**
  - Shrinks coefficients
  - Sets some to zero → selection

Ridge = **stability**
LASSO = **sparsity**

Summary

These slides cover:

- Non-Parametric Regression
- Ridge Regression
- LASSO

R functions to know:

- **Non-Parametric Regression**: `ksmooth` (kernel regression); `bs` (regression splines); `sreg` in the `fields` package (smoothing splines); `gam` (generalized additive models)
- **Ridge Regression/LASSO**: `glmnet` and `cv.glmnet` in the `glmnet` package