Lecture 7: Smoothing Splines in Practice and Choosing λ

MATH5824 Generalised Linear and Additive Models

Richard P Mann

MATH5824 Generalised Linear and Additive Models

Reading

Course notes: Chapter 4, Section 4.5 and Chapter 5, Sections 5.1–5.2

`www.richardpmann.com/MATH5824`

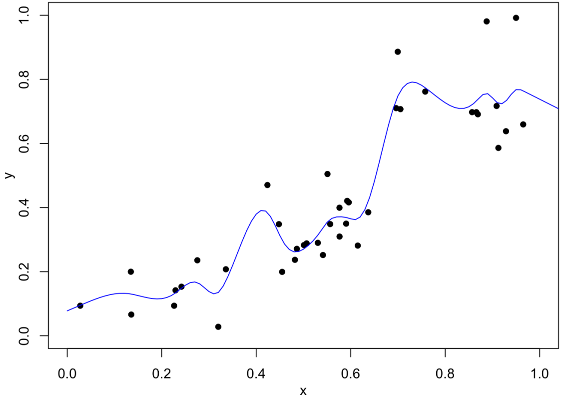
Smoothing Splines in R

The `smooth.spline()` function fits cubic smoothing splines:

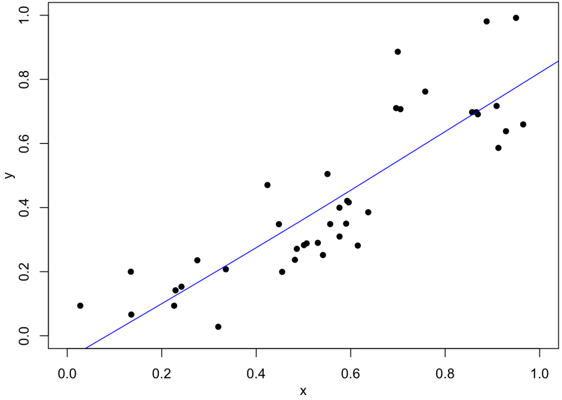
```
# Fit with specified lambda
fit1 <- smooth.spline(x, y, lambda = 0.00001)  # Nearly interpolating
fit2 <- smooth.spline(x, y, lambda = 1)         # Very smooth

# Plot
plot(x, y, pch = 19)
lines(fit1, col = "blue")
lines(fit2, col = "red")

# Predict at new locations
predict(fit1, x = c(2.5, 7.5))
```

Note: If `lambda` is not specified, the optimal value is selected by generalised cross-validation (discussed next lecture).

Example: Change-Point Data with Different λ



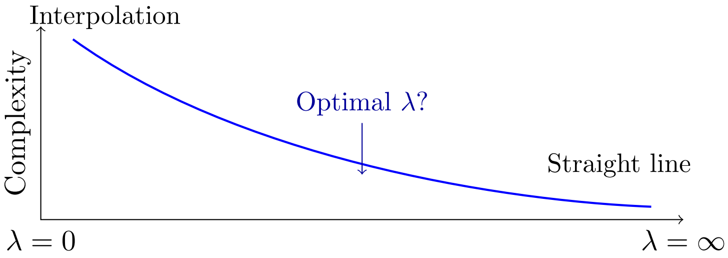
(a) Small $\lambda = 0.00001$



(b) Large $\lambda = 1$

With small λ, the spline captures the change-point but may track noise. With large λ, the spline is too smooth and misses the change-point entirely.

Effect of λ: Visual Summary



The fundamental question: How do we choose λ?

Strategy 1: Training/Test Split

Idea: Partition data indices into training set I_1 and test set I_2.

Procedure:

1. Fit the smoothing spline using only training data $\{(t_i, y_i) : i \in I_1\}$
2. Evaluate prediction quality on test data:

$$Q_{I_1:I_2}(\lambda) = \sum_{i \in I_2} \left(y_i - \hat{f}_{\lambda, I_1}(t_i) \right)^2$$

3. Choose λ minimising $Q_{I_1:I_2}(\lambda)$

Limitation: Wastes data — the test set is not used for fitting.

Strategy 2: Leave-One-Out Cross-Validation

Idea: Use each observation in turn as a single-point test set.

Ordinary Cross-Validation (OCV):

$$Q_{\mathrm{OCV}}(\lambda) = \frac{1}{n}\sum_{j=1}^{n}\Big(y_j - \hat{f}_{\lambda,-j}(t_j)\Big)^2$$

where $\hat{f}_{\lambda,-j}$ is the smoothing spline fitted to all data *except* observation j.

Choose λ to minimise $Q_{\mathrm{OCV}}(\lambda)$.

Apparent problem: Requires fitting n separate smoothing splines.

$\Rightarrow$ A computational trick avoids this (next lecture).

Key points:

- `smooth.spline()` in R fits cubic smoothing splines
- The smoothing parameter λ controls the trade-off between fit and smoothness
- Small λ: wiggly curve close to data
- Large λ: smooth curve (approaches a straight line)
- Training/test split: simple but wastes data
- Leave-one-out CV: uses all data, but apparently requires n fits

Next lecture: The smoothing matrix, effective degrees of freedom, and generalised cross-validation.