STA 35C: Statistical Data Science III

Lecture 7: Assessing Model Accuracy

Dogyoon Song

Spring 2025, UC Davis

Agenda

Quick review: The regression framework

Assessing a regression model:

- Training & test MSE
- The bias-variance tradeoff

Hints on Homework 1

Recap: Regression

Regression = Supervised learning with a quantitative response Y

- Given data $(x_1, y_1), \ldots, (x_n, y_n)$, we estimate f so that $Y \approx f(X)$
- *Prediction:* For x_{new}, predict $\hat{y}_{\text{new}} = \hat{f}(x_{\text{new}})$
- *Inference:* Learn relationships among X and Y

(Simple) linear regression:

- Assume $f(X) = \beta_0 + \beta_1 X$; estimate β_0, β_1 by least squares
- *Assessment:*
  - Prediction fit: RSS or $R^2 = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$
  - Inference: confidence intervals, hypothesis tests via RSE
- Extensions: multiple predictors, nonlinear terms, qualitative predictors
- Pitfalls: invalid linear model assumptions, outliers/high-leverage points, collinearity

Today's focus: We've learned to *build* regression models; let's see how to *evaluate* them

Mean squared error (MSE)

Motivation: Given a model $\hat{f}$, we need a metric to gauge how well $\hat{f}$ predicts Y

Why?

- To evaluate the current model's accuracy
- To select among multiple candidate models

Mean squared error (MSE):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

In linear regression, MSE corresponds to the residual sum of squares (RSS)

- Minimizing MSE $\Leftrightarrow$ minimizing RSS (least squares)

Training MSE vs. test MSE

Training MSE uses the same data that built the model:

$$\mathrm{MSE}_{\mathrm{train}} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

However, we truly care about *future* performance on *unseen* test data

- Hypothetically, if we had a set of *new* test points $(x_j^{\mathrm{test}}, y_j^{\mathrm{test}})$:

$$\mathrm{MSE}_{\mathrm{test}} = \frac{1}{n_{\mathrm{test}}}\sum_{j=1}^{n_{\mathrm{test}}}(y_j^{\mathrm{test}} - \hat{f}(x_j^{\mathrm{test}}))^2.$$

Ideally, we might want to learn a model by minimizing test MSE directly, but...

The challenge in practice

Reality: We usually do *not* have a separate test set

- Minimizing test MSE is impossible
- Thus, we typically end up minimizing training MSE instead

However, low training MSE $\not\Rightarrow$ low test MSE

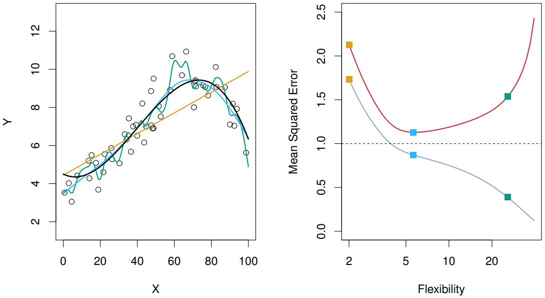

Figure: As model flexibility grows, training MSE usually decreases, but test MSE can increase [JWHT21, Figure 2.9]

The bias-variance tradeoff

Question: Why does the U-shape in test error occur?

Test MSE can be decomposed into $\text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$

$$\begin{aligned}\mathbb{E}\big(y_0 - \hat{f}(x_0)\big)^2 &= \Big(\mathbb{E}[\hat{f}(x_0)] - f(x_0)\Big)^2 + \mathbb{E}\Big[\big(\hat{f}(x_0) - \mathbb{E}[\hat{f}(x_0)]\big)^2\Big] + \mathrm{Var}(\epsilon) \\ &= \underbrace{\mathrm{Bias}^2\big(\hat{f}(x_0)\big)}_{\text{model mismatch}} + \underbrace{\mathrm{Var}\big(\hat{f}(x_0)\big)}_{\text{sensitivity to data}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{irreducible}}\end{aligned}$$

As model flexibility increases:

- Bias tends to *decrease*
- Variance tends to *increase*

Takeaway: An optimal model should balance bias *and* variance for the lowest test error

Interpreting bias and variance

Bias:

- Systematic error due to an overly simplistic model class
- Example: A linear model used when the true f is highly nonlinear

Variance:

- How much $\hat{f}$ would fluctuate given a different training sample
- Complex and flexible models can vary greatly from one sample to another

Sweet spot:

- Balanced complexity—neither too simple nor too complex
- Techniques like cross-validation (future lecture) can help find it

Summary of the bias-variance tradeoff

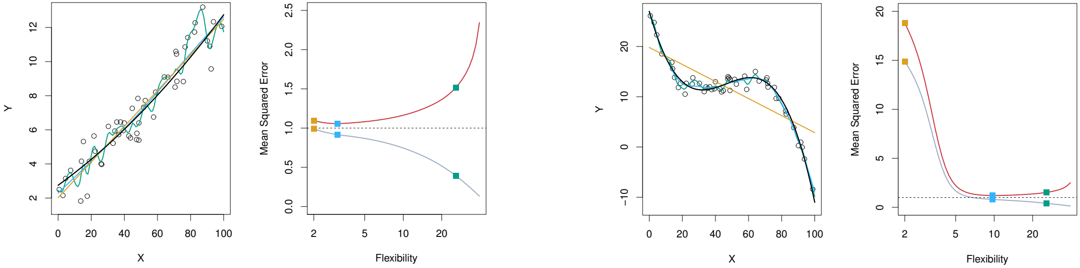

Figure: Illustrations of bias-variance in different settings [JWHT21, Figures 2.10 & 2.11].

Bottom line:

- Overly simple models $\Rightarrow$ high bias, low variance
- Overly complex models $\Rightarrow$ low bias, high variance
- **Optimal model complexity** finds a sweet spot balancing both
- This idea applies broadly to many supervised learning methods

Wrap-up

Summary of today's lecture:

- Assessing regression model accuracy via MSE
- Training vs. test MSE
- The bias-variance tradeoff

Recap of regression:

- Regression problem; setup and the objectives
- Linear regression
  - Model & interpretation of regression coefficients
  - Parameter estimation & inference
  - Assessing model fit using RSS and R^2
  - Extensions: multiple predictors, nonlinear terms, qualitative predictors
- Model assessment: importance of test performance, and the bias-variance tradeoff

Next lecture: Classification

Homework 1: What each problem is asking

Problem 1: Probability basics

- Recognizing that probabilities lie in $[0, 1]$ and sum to 1
- Visualizing a distribution
- Computing expectation and variance

Problem 2: Bayes' theorem and basic learning

- Estimating p_{true} from coin flips via Bayes' theorem
- Treating our "guess" as a random variable and updating its assigned probabilities using data
- Understanding how learning occurs and its sensitivity to p_{true} and the initial guess

Problem 3: Simple linear regression

- Confirming the formula for least squares estimates
- Practicing basic computations on a simple dataset
- Running linear regression in R to see it in action

Problem 4: Model assessment

- Comparing training vs. test error as model complexity changes
- Exploring the bias-variance tradeoff and its subtle nuances

References

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani.
An Introduction to Statistical Learning: with Applications in R, volume 112 of *Springer Texts in Statistics*.
Springer, New York, NY, 2nd edition, 2021.