Bias-Variance Tradeoff

David Dalpiaz

STAT 432, Fall 2019

Announcements

- See Compass2g announcement
- See www.stat432.org
- Quiz 02
- Quiz 03
- Quiz 04 and Analysis 01 incoming

Statistical Learning

- Supervised Learning
  - **Regression**
    - Parametric
    - Non-Parametric
  - Classification
- Unsupervised Learning

Regression Setup

Given a random pair $(X, Y) \in \mathbb{R}^p \times \mathbb{R}$. We would like to “predict” Y with some function of X, say, $f(X)$.

Define the **squared error loss** of estimating Y using $f(X)$ as

$$L(Y, f(X)) \triangleq (Y - f(X))^2$$

We call the expected loss the **risk** of estimating Y using $f(X)$

$$R(Y, f(X)) \triangleq \mathbb{E}[L(Y, f(X))] = \mathbb{E}_{X,Y}[(Y - f(X))^2]$$

Minimizing Risk

After conditioning on X

$$\mathbb{E}_{X,Y}\left[(Y - f(X))^2\right] = \mathbb{E}_X \mathbb{E}_{Y|X}\left[(Y - f(X))^2 \mid X = x\right]$$

We see that the risk is minimzied by the conditional mean

$$f(x) = \mathbb{E}(Y \mid X = x)$$

We call this, the **regression function**.

Estimating f

Given data

$$\mathcal{D} = (x_i, y_i) \in \mathbb{R}^p \times \mathbb{R}$$

our goal is to find some $\hat{f}$ that is a good estimate of the regression function f.

Expected Prediction Error

$$\mathrm{EPE}\left(Y, \hat{f}(X)\right) \triangleq \mathbb{E}_{X,Y,\mathcal{D}}\left[\left(Y - \hat{f}(X)\right)^2\right]$$

Reducible and Irreducible Error

$$
\begin{aligned}
\mathrm{EPE}\left(Y, \hat{f}(x)\right) =& \mathbb{E}_{Y|X,\mathcal{D}}\left[\left(Y - \hat{f}(X)\right)^2 \mid X = x\right] \\
=& \mathbb{E}_{Y|X,\mathcal{D}}\left[\left(Y - \hat{f}(X)\right)^2 \mid X = x\right] \\
=& \underbrace{\mathbb{E}_{\mathcal{D}}\left[\left(f(x) - \hat{f}(x)\right)^2\right]}_{\text{reducible error}} + \underbrace{\mathbb{V}_{Y|X}\left[Y \mid X = x\right]}_{\text{irreducible error}}
\end{aligned}
$$

Bias and Variance

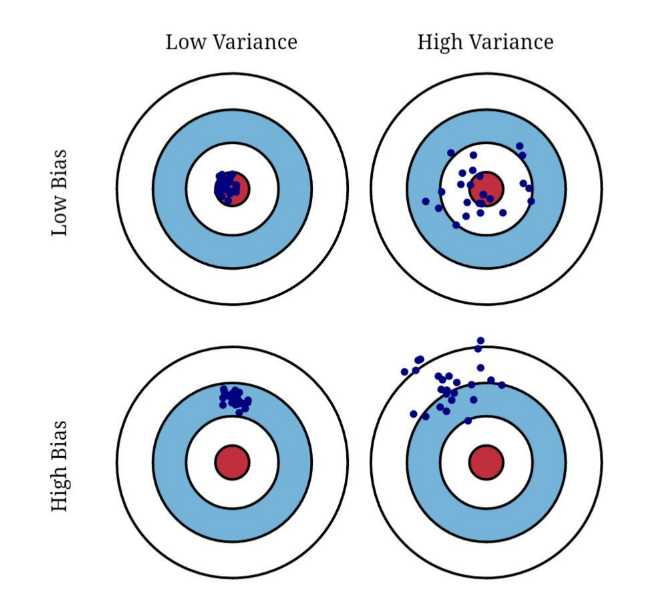
Recall the definition of the **bias** of an estimator.

$$\mathsf{bias}(\hat{\theta}) \triangleq \mathbb{E}\left[\hat{\theta}\right] - \theta$$

Also recall the definition of the **variance** of an estimator.

$$\mathbb{V}(\hat{\theta}) = \mathsf{var}(\hat{\theta}) \triangleq \mathbb{E}\left[\left(\hat{\theta} - \mathbb{E}\left[\hat{\theta}\right]\right)^2\right]$$

Bias and Variance



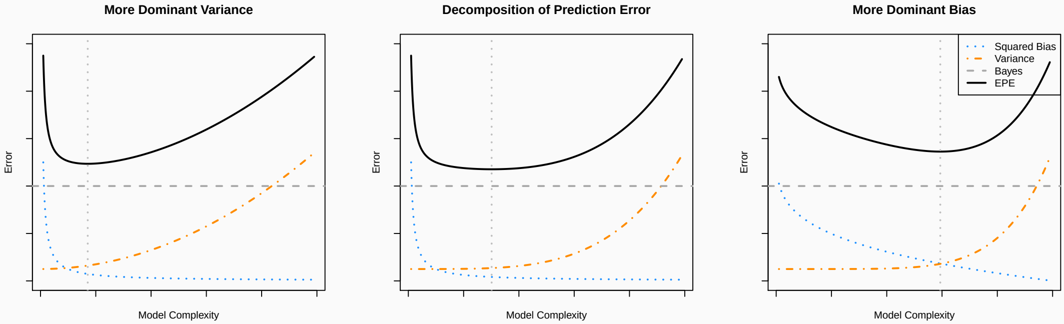
Bias-Variance Decomposition

$$
\begin{aligned}
\mathrm{MSE}\left(f(x), \hat{f}(x)\right) &= \mathbb{E}_{\mathcal{D}}\left[\left(f(x) - \hat{f}(x)\right)^2\right] \\
&= \underbrace{\left(f(x) - \mathbb{E}\left[\hat{f}(x)\right]\right)^2}_{\mathrm{bias}^2\left(\hat{f}(x)\right)} + \underbrace{\mathbb{E}\left[\left(\hat{f}(x) - \mathbb{E}\left[\hat{f}(x)\right]\right)^2\right]}_{\mathrm{var}\left(\hat{f}(x)\right)}
\end{aligned}
$$

Bias-Variance Decomposition

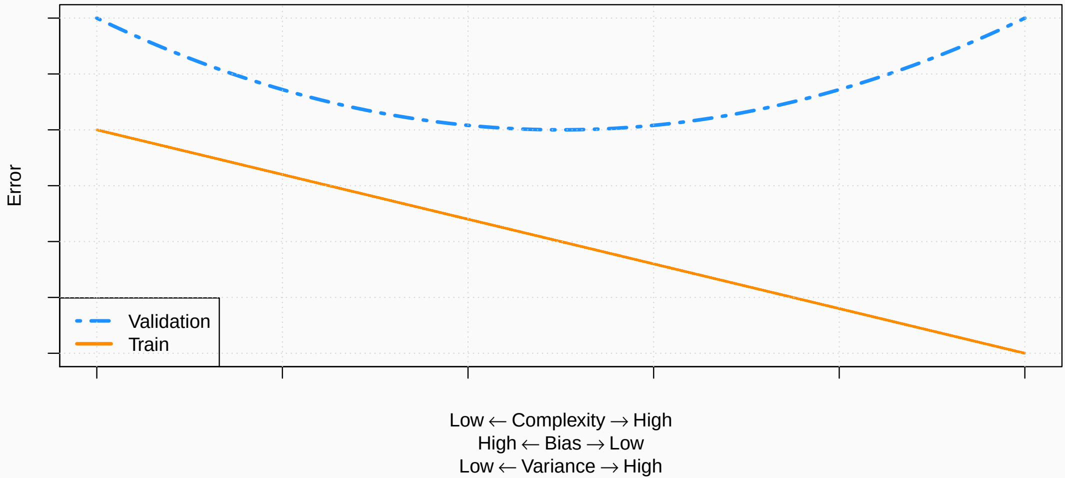
$$\text{MSE}\left(f(x), \hat{f}(x)\right) = \text{bias}^2\left(\hat{f}(x)\right) + \text{var}\left(\hat{f}(x)\right)$$

Bias-Variance Decomposition



Expected Test Error

Error versus Model Complexity



Simulation Study, Regression Function

We will illustrate these decompositions, most importantly the bias-variance tradeoff, through simulation. Suppose we would like to train a model to learn the true regression function function

$$f(x) = x^2$$

```
f = function(x) {
  x ^ 2
}
```

Simulation Study, Regression Function

More specifically, we'd like to predict an observation, Y, given that $X = x$ by using $\hat{f}(x)$ where

$$\mathbb{E}[Y \mid X = x] = f(x) = x^2$$

and

$$\mathbb{V}[Y \mid X = x] = \sigma^2.$$

Simulation Study, Data Generating Process

To carry out a concrete simulation example, we need to fully specify the **data generating process**. We do so with the following `R` code.

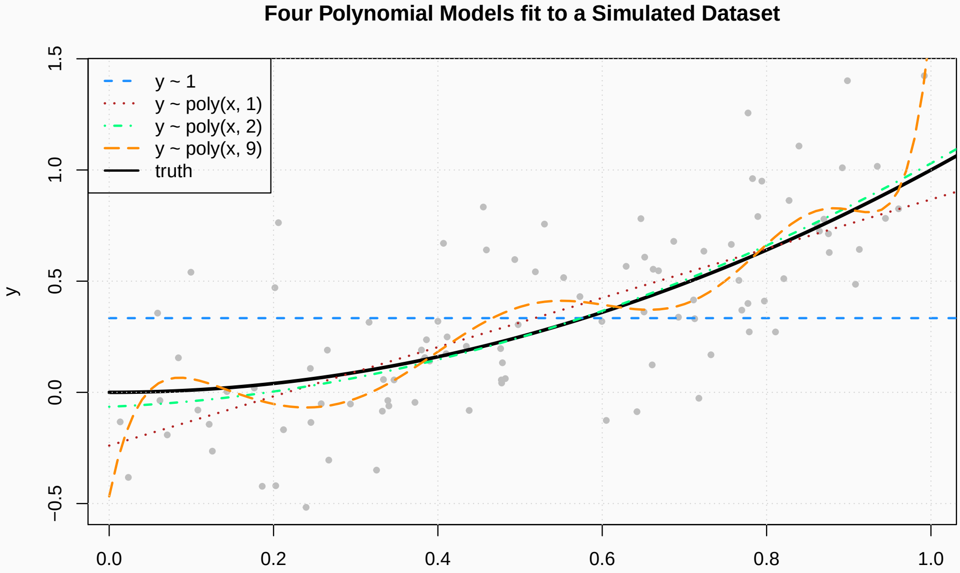
```r
gen_sim_data = function(f, sample_size = 100) {
  x = runif(n = sample_size, min = 0, max = 1)
  y = rnorm(n = sample_size, mean = f(x), sd = 0.3)
  data.frame(x, y)
}
```

Simulation Study, Models

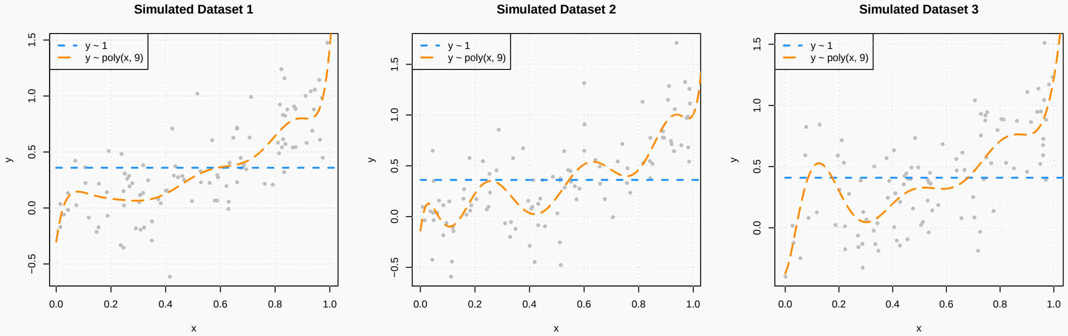
Using this setup, we will generate datasets, $\mathcal{D}$, with a sample size $n = 100$ and fit four models.

$$
\begin{aligned}
\texttt{predict(fit0, x)} &= \hat{f}_0(x) = \hat{\beta}_0 \\
\texttt{predict(fit1, x)} &= \hat{f}_1(x) = \hat{\beta}_0 + \hat{\beta}_1 x \\
\texttt{predict(fit2, x)} &= \hat{f}_2(x) = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x^2 \\
\texttt{predict(fit9, x)} &= \hat{f}_9(x) = \hat{\beta}_0 + \hat{\beta}_1 x + \hat{\beta}_2 x^2 + \ldots + \hat{\beta}_9 x^9
\end{aligned}
$$

Simulation Study, Trained Models



Simulation Study, Repeated Training

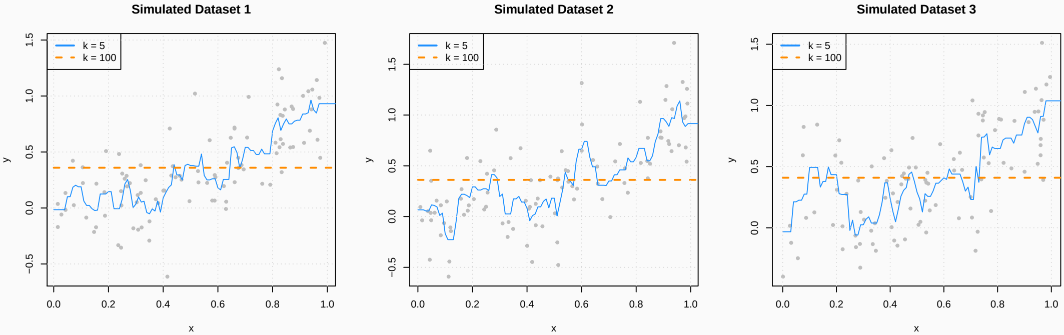


Simulated Dataset 1

Simulated Dataset 2

Simulated Dataset 3

Simulation Study, KNN



Simulated Dataset 1

Simulated Dataset 2

Simulated Dataset 3

Simulation Study, Setup

```r
set.seed(1)
n_sims = 250
n_models = 4
x = data.frame(x = 0.90)
predictions = matrix(0, nrow = n_sims, ncol = n_models)
```

Simulation Study, Running Simulations

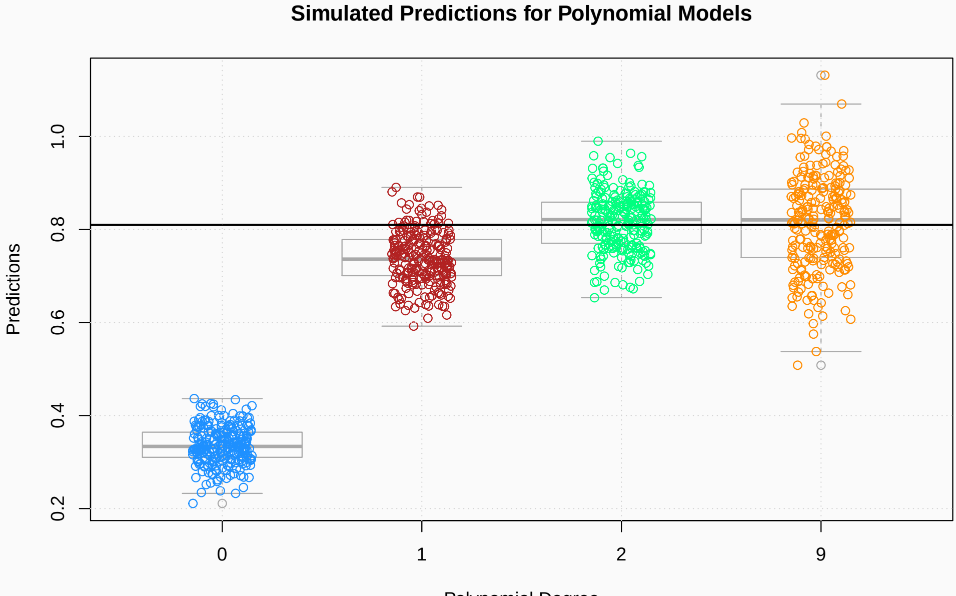
```
for(sim in 1:n_sims) {

  sim_data = gen_sim_data(f = f)

  # fit models
  fit_0 = lm(y ~ 1,                   data = sim_data)
  fit_1 = lm(y ~ poly(x, degree = 1), data = sim_data)
  fit_2 = lm(y ~ poly(x, degree = 2), data = sim_data)
  fit_9 = lm(y ~ poly(x, degree = 9), data = sim_data)

  # get predictions
  predictions[sim, 1] = predict(fit_0, x)
  predictions[sim, 2] = predict(fit_1, x)
  predictions[sim, 3] = predict(fit_2, x)
  predictions[sim, 4] = predict(fit_9, x)
}
```

Simulation Study, Results



Bias-Variance Tradeoff

- As complexity *increases*, **bias decreases**.
- As complexity *increases*, **variance increases**.

Simulation Study, Quantities of Interest

$$\mathsf{MSE}\left(f(0.90), \hat{f}_k(0.90)\right) = \underbrace{\left(\mathbb{E}\left[\hat{f}_k(0.90)\right] - f(0.90)\right)^2}_{\mathsf{bias}^2\left(\hat{f}_k(0.90)\right)}$$

$$+ \underbrace{\mathbb{E}\left[\left(\hat{f}_k(0.90) - \mathbb{E}\left[\hat{f}_k(0.90)\right]\right)^2\right]}_{\mathsf{var}\left(\hat{f}_k(0.90)\right)}$$

Estimation Using Simulation

$$\widehat{\text{MSE}}\left(f(0.90), \hat{f}_k(0.90)\right) = \frac{1}{n_{\texttt{sims}}} \sum_{i=1}^{n_{\texttt{sims}}} \left(f(0.90) - \hat{f}_k(0.90)\right)^2$$

$$\widehat{\text{bias}}\left(\hat{f}(0.90)\right) = \frac{1}{n_{\texttt{sims}}} \sum_{i=1}^{n_{\texttt{sims}}} \left(\hat{f}_k(0.90)\right) - f(0.90)$$

$$\widehat{\text{var}}\left(\hat{f}(0.90)\right) = \frac{1}{n_{\texttt{sims}}} \sum_{i=1}^{n_{\texttt{sims}}} \left(\hat{f}_k(0.90) - \frac{1}{n_{\texttt{sims}}} \sum_{i=1}^{n_{\texttt{sims}}} \hat{f}_k(0.90)\right)^2$$

Simulation Study, Results

Degree	Mean Squared Error	Bias Squared	Variance
0	0.22643	0.22476	0.00167
1	0.00829	0.00508	0.00322
2	0.00387	0.00005	0.00381
9	0.01019	0.00002	0.01017

If Time

- Note that, $\hat{f}_9(x)$ is unbiased
- Some live coding