DS 3010

Spring 2026

Wenting Xu

IOWA STATE UNIVERSITY

Module 1: Multiple Linear Regression

Part2: MSE & Properties of Least Squares

Wenting Xu

Iowa State University

Copyright & Attribution

Some lecture slides and instructional materials in this course are adapted from the following sources:

- *An Introduction to Statistical Learning: With Applications in R (Second Edition)*
  Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani
  Springer, 2021

- Online course materials developed by Trevor Hastie, Robert Tibshirani, and collaborators.

Our objectives:

1. How do we evaluate how good our model is at prediction?
   use the test MSE

2. What are the theoretical properties of our model?

How do we evaluate how good our model is at prediction?

Measuring the Quality of Fit

Think back to our first in-class activity. How did we quantify how well our model is able to predict patient satisfaction?

We needed some way to measure how well our regression model's predicted values ($\hat{Y}$) (match the actual observed response (Y).

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

↳ RSS depends on n → if n is large, RSS will be large

↳ hard to compare datasets w/ different sizes

Mean squared error (MSE)

$$Y = f(X) + \epsilon$$

Problem: $f(x)$ is unknown.

Goal: Estimate $f(x)$ from the data: $\hat{f}(x)$.

We need some way to measure how well a regression model actually matches the observed data.

In the regression setting, the most commonly-used measure is the mean squared error (MSE), given by

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

Training MSE

Training data set is the data you used to build your model. The MSE evaluated on this data set is referred to as the training MSE.

Suppose $(x_i, y_i), i = 1, \ldots, n$ represents our training data. $\hat{f}$ is estimated from training data and then our training MSE is:

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

Training MSE

In general, we do not really care how well the method works training on the training data.

Rather, we are interested in the accuracy of the predictions that we obtain when we apply our model to previously *unseen test data.*

Test MSE

Test data set is some previously unseen data that were not used to train the model. Ithe MSE evaluated on the test set is referred to as the test MSE.

Suppose $(x_i', y_i'), i = 1, \ldots, n$ represents our test data. Then our test MSE is:

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

trained using traing data

Training MSE vs. Test MSE

Overfitting → fitting to the noise, not the general trend

- As we'll see in the next module, the training MSE and test MSE behave very differently.

- A model that results in a small training MSE will not necessarily result in a small test MSE.

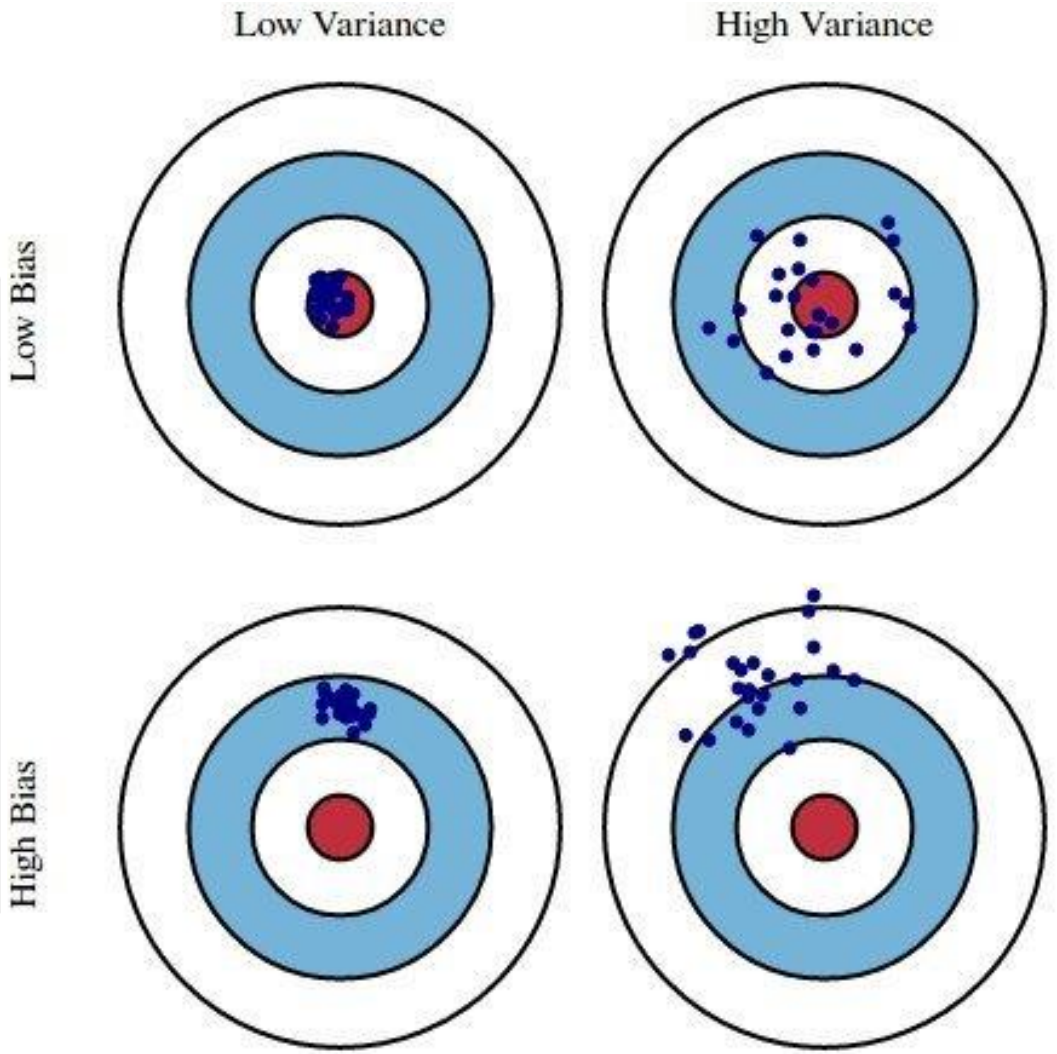
- Our goal in prediction is to select a method that minimizes the test MSE. Low training MSE does not imply low test MSE.

What are the theoretical properties of our model?

Properties of least square estimators

- Remember in real applications, the true parameters $\beta_0, \beta_1, \ldots, \beta_p$ are unknown to us.
- Ideally, we hope our least square estimates $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ are close to the true values of $\beta_0, \beta_1, \ldots, \beta_p$.
- We can quantify how 'close' our estimates are to the truth using the following concepts:
    - Bias
    - Standard error

Population Variance	Sample Variance
$\sigma^2 = \frac{\sum_{i=1}^{N}(x_i - \mu)^2}{N}$	$s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$
σ^2 = population variance x_i = value of i^{th} element μ = population mean N = population size	s^2 = sample variance x_i = value of i^{th} element $\bar{x}$ = sample mean n = sample size



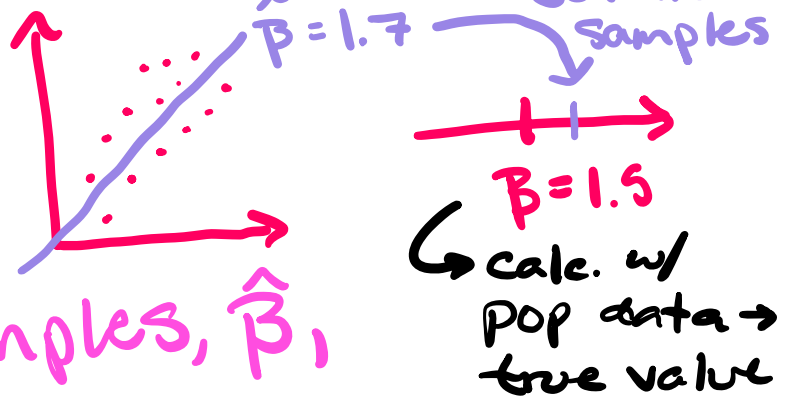
source: https://pbs.twimg.com/media/CpWDWuSW8AQUuCk.jpg

Properties of least square estimators

How good are our estimates?

Assume the linear approximation assumption holds

↳ the least square est. $\hat{\beta}$ are unbiased est. of the true parameter β



↳ If we can collect infinate samples, $\hat{\beta}$, the average of $\hat{\beta} = \beta$, $E(\hat{\beta}) = \beta$

↑ expection

Properties of least square estimators

$E(\hat{\beta}) = \beta$

↳ is $\hat{Y}$ an unbased estimate of Y

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 \cdot x_1 + \hat{\beta}_2 \cdot x_2 + \ldots + \beta_p \cdot x_p$$

$$E(\hat{Y}) = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \ldots + \beta_p \cdot x_p$$

unaccounted for error

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p \cdot x_p + \varepsilon$$

so $E(\hat{Y}) \neq Y$

Accuracy of β

- The unbiasedness property tell us that the average of our estimates from many many datasets will be very close to the true population parameter β.
- But we don't have access to many many datasets. For a particular data set, the single estimates $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ may be a substantial underestimate or overestimate of the true $\beta_0, \beta_1, \ldots, \beta_p$.
- How far off will our single estimate $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$ be?

Accuracy of β

standard error = $SE(\hat{\beta})$

↳ how far off a single estimate is from the truth

↳ tell us the average amount that $\hat{\beta}$ differs from β

↳ want $SE(\hat{\beta})$ = small

Accuracy of β

```
> summary(model)

Call:
lm(formula = satisf ~ age + severe + anxiety, data = patient)

Residuals:
     Min       1Q   Median       3Q      Max
-18.3524  -6.4230   0.5196   8.3715  17.1601

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 158.4913    18.1259   8.744 5.26e-11 ***
age          -1.1416     0.2148  -5.315 3.81e-06 ***
severe       -0.4420     0.4920  -0.898   0.3741
anxiety     -13.4702     7.0997  -1.897   0.0647 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

$\hat{\beta}$ $SE(\hat{\beta})$

Implementation in R

See R script: `leastsq properties.R`