Supervised Learning & Statistical Decision Theory

DS 301

Iowa State University

Today's Agenda

- Please make sure you've filled out the Start of Semester Survey.

- Supervised learning setup.

- Statistical decision theory.

Most statistical learning problems fall broadly into one of two categories:

1. Supervised learning
2. Unsupervised learning

Supervised learning

This is the setting where you have labelled data:

$$(y_i, x_i), \quad i = 1, \ldots, n.$$

p predictors

- y_i is our response (outcome of interest), x_i's are our predictors.
- You have both **input** (X) and **output** (Y) values.
- Majority of machine learning problems/techniques fall into this category.

Regression vs. Classification

$(y_i, x_{i1}, x_{i2}, \ldots, x_{ip})$

- When the response is quantitative (i.e. continuous real number), we say this is a regression problem and we use a regression model.

- When the response is qualitative (i.e. categorical), we say this is a classification problem and we use a classification method.

Supervised learning setup

$y_i = f(x_{i1}, x_{i2}, \ldots, x_{ip}) + \varepsilon_i, \quad i = 1, \ldots, n$

$$Y = \underbrace{f}_{\text{signal}}(X) + \underbrace{\epsilon}_{\text{noise}}$$

$f(x) = \beta_0 + \beta_1 X_1$

$f(x) = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2$

$f(x) =$ we cannot write it out

- The function f captures the systematic relationship between X and Y.
- f is fixed and unknown.
- ϵ represents —? random noise (variation)
- Our goal: to estimate (learn) the function f, using a set of training data.

Why estimate f?

1. **Prediction:** predict Y from X.
   - In many settings, the predictors X are readily available, but the output Y cannot be easily obtained.
   - Example:
     - $X_1, \ldots, X_p$ are characteristics of a patient's blood sample from a lab.
     - Y is a patient's risk for a severe adverse reaction to a drug.

Why estimate f?

2. **Inference:** understand the association between Y and X.
    - Goal is not necessarily to make predictions but to help us *describe* the relationship between Y and X.
    - Example:
        - Which predictors are associated with the response? Can we identify those important predictors?
        - What is the relationship between the response and each predictor?
        - Is the relationship between the response and predictors linear or is it more complicated?

Regression models

General setup:

$$Y = \underline{f(X)} + \underline{\epsilon} \quad \text{(true model)}$$

- Y: quantitative response. (real number, continuous)
- $X_1, X_2, \ldots X_p$: p different predictors
- We assume that there is some $\underbrace{\text{relationship}}_{f(X)}$ between Y and $X = (X_1, X_2, \ldots, X_p)$:
- Our goal: to estimate (learn) the function f, using a set of training data:

$$\hat{Y} = \hat{f}(X) \leftarrow \text{(estimated model)}$$

where $\hat{f}$ represents our estimate for f and $\hat{Y}$ represents the resulting prediction for Y.

Accuracy of $\hat{Y}$

The accuracy of $\hat{Y}$ as a prediction for Y depends on two quantities: *reducible error* and *irreducible error*.

- $\hat{f}$ will not be a perfect estimate for f and this inaccuracy will introduce some error $\rightarrow$ **reducible error**.
  - We can potentially improve this error by using the most appropriate statistical learning technique to estimate f.

not measurable

- Y is also a function of ϵ $\rightarrow$ **irreducible error**.
  - By definition, ϵ cannot be predicted using X.
  - Therefore, the variability of ϵ will affect our predictions.
  - No matter what we do, we cannot the reduce the error introduced by ϵ.

Accuracy of $\hat{Y}$

The accuracy of $\hat{Y}$ as a prediction for Y depends on two quantities: *reducible error* and *irreducible error*.

Our focus will be estimating f with the aim of minimizing the **reducible error**.

Keep in mind that the irreducible error will always provide an upper bound on the accuracy of our prediction for Y.

This bound is almost always unknown in practice.

Statistical Decision Theory

Tradeoffs

Suppose we are considering two models:

- One is simpler, contains fewer predictors:

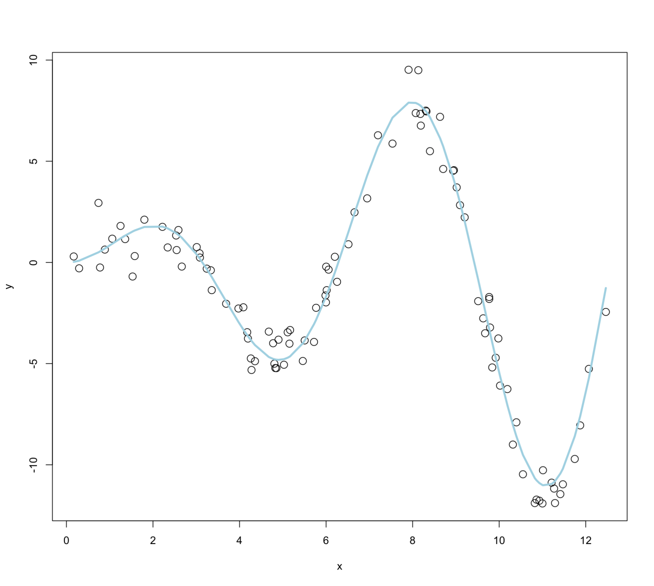
$$Y = \beta_1 X_1 + \beta_2 X_2 + \epsilon$$

- The other is more complicated, it contains more predictors and is more flexible:

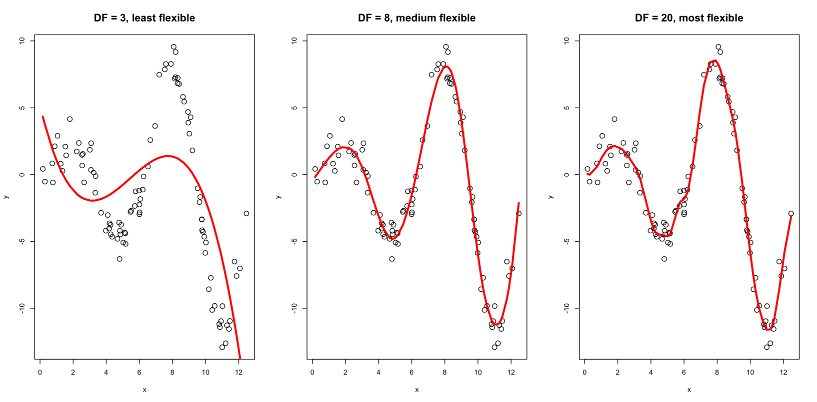
$$Y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_2^2 + \beta_6 X_3^2 + \epsilon.$$

Is there ever a reason to use a simpler, more restrictive model over a complex, more flexible model?

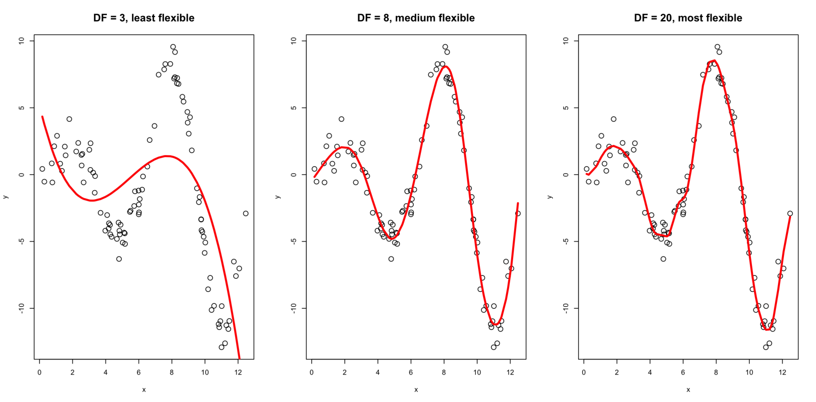
Motivating example



Motivating example



Training error



- least flexible model's training error: 16.92441
- medium flexible model's training error: 0.8542847
- most flexible model's ~~prediction~~ error: 0.6513902
  training

Suppose we have new observations...

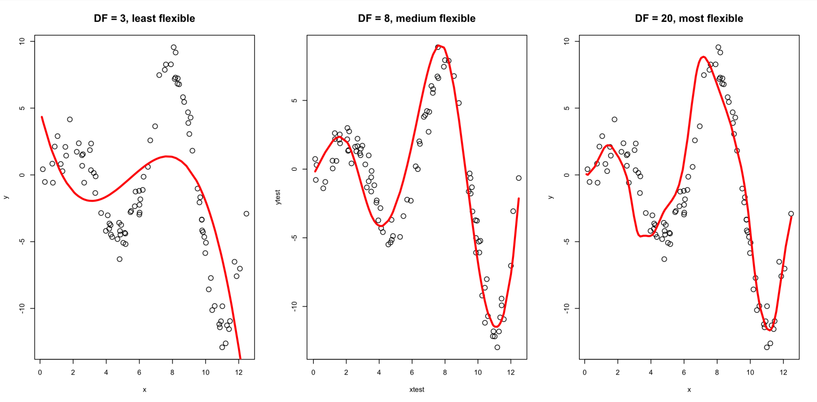
We want to use the models we just built ('trained') to help us make predictions on new data.

test set.

We can then evaluate how well each model's predictions actually match the observed data. We refer to this as **test error**.

(test set)

Test error



- least flexible model's test error: 16.71268
- medium flexible model's test error: 3.579566
- most flexible model's test error: 5.47645

small training error $\not\to$ small test error (overfitting)

Mean squared error

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

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

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

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.**

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

$\hat{f}$ estimated from training set

(x_i, y_i), $i=1,\dots,n$
training set

y	x_1
10	4
8	2

$\hat{f}(x) = 3 + 2x$

$\hat{f}(x)$ $(\hat{y})$
11
7

$\frac{(11-10)^2 + (7-8)^2}{2} = \boxed{}$: training MSE.

Test MSE

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

trained $\hat{f}$

$$\text{test MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i' - \hat{f}(x_i'))^2.$$

(x_i', y_i'), $i = 1, \ldots, m$. test set

$\left(\hat{f}(x) = 3 + 2x \right)$

Tradeoff

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