DSC 40A

Theoretical Foundations of Data Science I

In This Video

How can we make more informed predictions based on attributes of the individuals in our data set?

Recommended Reading

Course Notes: Chapter 2, Section 1

How do we predict someone's salary?

- Gather salary data, find prediction that minimizes risk.
- So far, we haven't used any information about the person.
- How do we incorporate, e.g., years of experience into our prediction?

Features

A **feature** is an attribute – a piece of information.

- **Numerical**: age, height, years of experience
- **Categorical**: college, city, education level
- **Boolean**: knows Python?, had internship?

Variables

- The features, x, that we base our predictions on are called **predictor variables**.
- The quantity, y, that we're trying to predict based on these features is called the **response variable**.
- We'll start by predicting salary based on years of experience.

Prediction Rules

- We believe that salary is a function of experience.
- I.e., there is a function H so that:

$$\text{salary} \approx H(\text{years of experience})$$

- H is called a **hypothesis function** or **prediction rule**.
- **Our goal**: find a good prediction rule, H.

Example Prediction Rules

$$H_1(\text{years of experience}) = \$50{,}000 + \$2,000 \times (\text{years of experience})$$

$$H_2(\text{years of experience}) = \$60{,}000 \times 1.05^{(\text{years of experience})}$$

$$H_3(\text{years of experience}) = \$100{,}000 - \$5{,}000 \times (\text{years of experience})$$

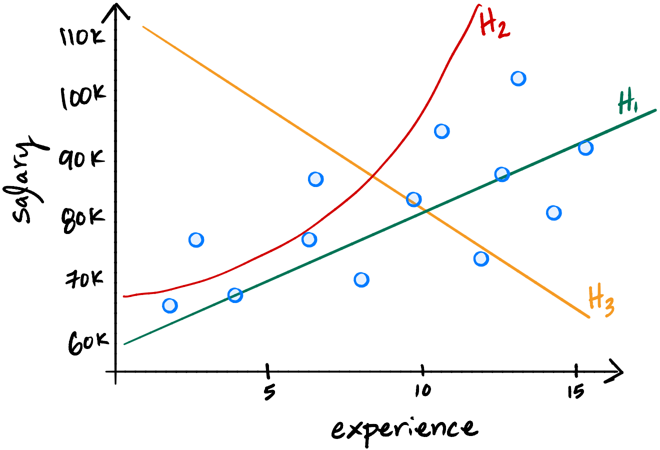
Comparing predictions

- How do we know which is best: H_1, H_2, H_3?
- We gather data from n people. Let x_i be experience, y_i be salary:

$$\begin{array}{ccc} (\text{Experience}_1, \text{Salary}_1) & & (x_1, y_1) \\ (\text{Experience}_2, \text{Salary}_2) & \rightarrow & (x_2, y_2) \\ \ldots & & \ldots \\ (\text{Experience}_n, \text{Salary}_n) & & (x_n, y_n) \end{array}$$

- See which rule works better on data.

Example



Quantifying the error of a prediction rule H

- Our prediction for person i's salary is $H(x_i)$
- The **absolute error** in this prediction:

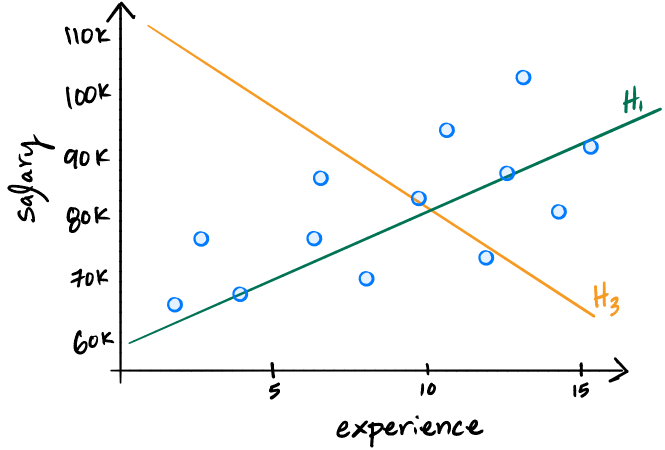
$$|H(x_i) - y_i|$$

- The **mean absolute error** of H:

$$R_{\text{abs}}(H) = \frac{1}{n}\sum_{i=1}^{n} |H(x_i) - y_i|$$

- Smaller the mean absolute error, the **better** the prediction rule.

Mean Absolute Error



Finding the best prediction rule

- **Goal:** out of all functions $\mathbb{R} \to \mathbb{R}$, find the function H^* with the smallest mean absolute error.

- That is, H^* should be the function that minimizes

$$R(H) = \frac{1}{n} \sum_{i=1}^{n} |H(x_i) - y_i|$$

Finding the best prediction rule

- **Goal:** out of all functions $\mathbb{R} \to \mathbb{R}$, find the function H^* with the smallest mean absolute error.
- That is, H^* should be the function that minimizes

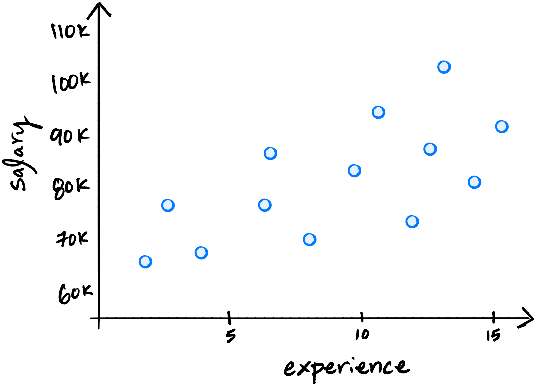
$$R(H) = \frac{1}{n} \sum_{i=1}^{n} |H(x_i) - y_i|$$

- **There are two problems with this.**

Question

Given the data below, is there a prediction rule H which has **zero** mean absolute error?

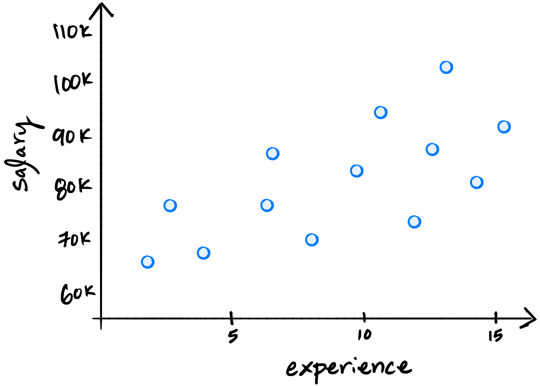
a) yes b) no



Question

Given the data below, is there a prediction rule H which has **zero** mean absolute error?

a) yes   b) no



Problem #1

- We can make mean absolute error very small, even zero!
- But the function will be weird.
- This is called **overfitting**.
- Remember our real goal: make good predictions on data **we haven't seen**.

Solution

- Don't allow H to be just any function.
- Require that it has a certain form.
- Examples:
  - Linear: $H(x) = w_1 x + w_0$
  - Quadratic: $H(x) = w_2 x^2 + w_1 x + w_0$
  - Exponential: $H(x) = w_0 e^{w_1 x}$
  - Constant: $H(x) = w_0$

Finding the best linear prediction rule

- **Goal:** out of all linear functions $\mathbb{R} \to \mathbb{R}$, find the function H^* with the smallest mean absolute error.

- That is, H^* should be the linear function that minimizes

$$R(H) = \frac{1}{n} \sum_{i=1}^{n} |H(x_i) - y_i|$$

Finding the best linear prediction rule

- **Goal:** out of all linear functions $\mathbb{R} \to \mathbb{R}$, find the function H^* with the smallest mean absolute error.

- That is, H^* should be the linear function that minimizes

$$R(H) = \frac{1}{n} \sum_{i=1}^{n} |H(x_i) - y_i|$$

- **There is still a problem with this.**

Problem #2

- It is hard to minimize the mean absolute error:[1]

$$\frac{1}{n}\sum_{i=1}^{n}|H(x_i) - y_i|$$

- **Not differentiable!**

- What can we do?

[1] Though it can be done with linear programming.

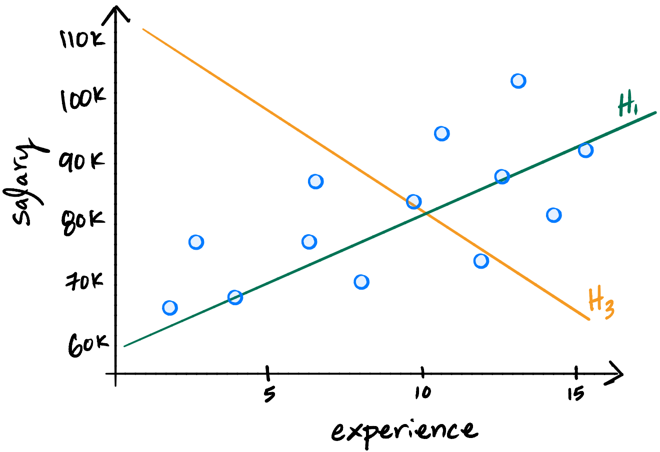
Quantifying the error of a prediction rule H

- Use the **mean squared error** (MSE) instead:

$$R_{\text{sq}}(H) = \frac{1}{n}\sum_{i=1}^{n}(H(x_i) - y_i)^2$$

- **Is differentiable!**

Mean Squared Error



Our Goal

- Out of all **linear** functions $\mathbb{R} \to \mathbb{R}$, find the function H^* with the smallest **mean squared error**.

- That is, H^* should be the linear function that minimizes

$$R_{\text{sq}}(H) = \frac{1}{n} \sum_{i=1}^{n} (H(x_i) - y_i)^2$$

- This problem is called **least squares regression**.

- **Next Time:** We find the linear prediction rule H^* that minimizes the mean squared error.