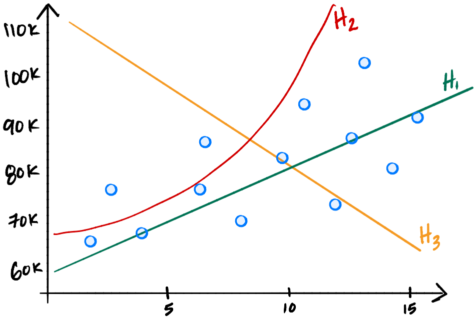


DSC 40A

Lecture 06

Least Squares Regression, pt. I

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, gender
- **Boolean**: knows Python?, had internship?

We'll start with just one feature (years of experience).

Today

- **Goal**: Predict salary from years of experience.
- How do we turn this into a math problem and solve it?

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

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

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

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

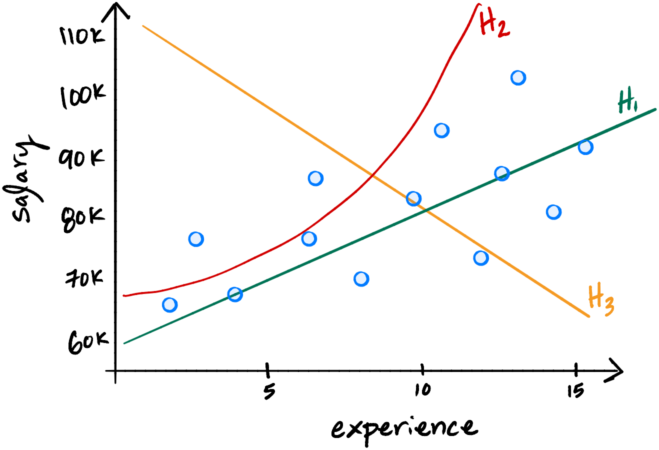
Comparing predictions

- 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:

$$\begin{matrix} (\text{Experience}_1, \text{Salary}_1) \\ (\text{Experience}_2, \text{Salary}_2) \\ \dots \\ (\text{Experience}_n, \text{Salary}_n) \end{matrix} \quad \rightarrow \quad \begin{matrix} (x_1, y_1) \\ (x_2, y_2) \\ \dots \\ (x_n, y_n) \end{matrix}$$

- 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:

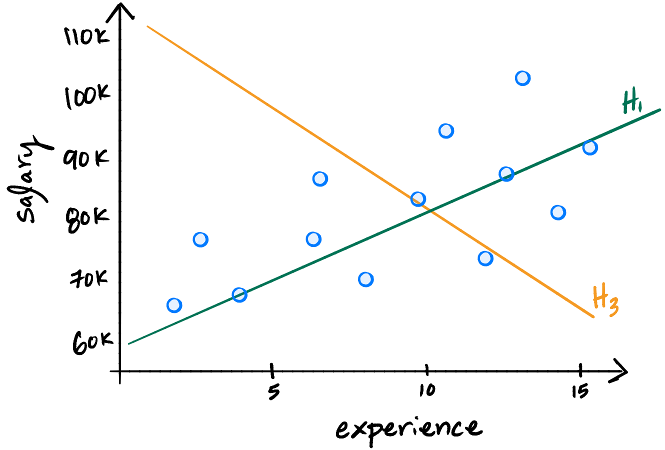
$$\left| H(x_i) - y_i \right|$$

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

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

- 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, find:

$$H^* = \arg\min_{H} \frac{1}{n} \sum_{i=1}^{n} \left| H(x_i) - y_i \right|$$

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, find:

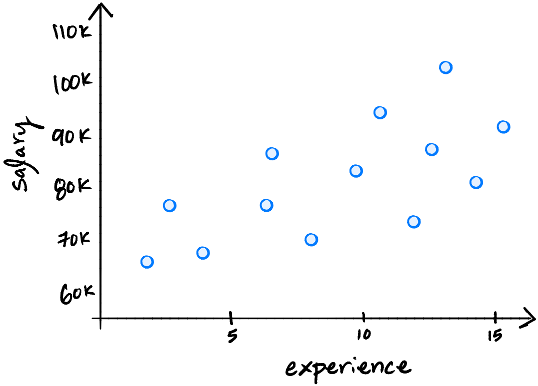
$$H^* = \arg\min_H \frac{1}{n} \sum_{i=1}^{n} \left| H(x_i) - y_i \right|$$

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

Discussion 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, find:

$$H^* = \underset{\text{linear } H}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left| H(x_i) - y_i \right|$$

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, find:

$$H^* = \underset{\text{linear } H}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left| H(x_i) - y_i \right|$$

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

Problem #2

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

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

- **Not differentiable!**
- What can we do?

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

Quantifying the error of a prediction rule H

- Instead of absolute error, use the **squared error** of a prediction:

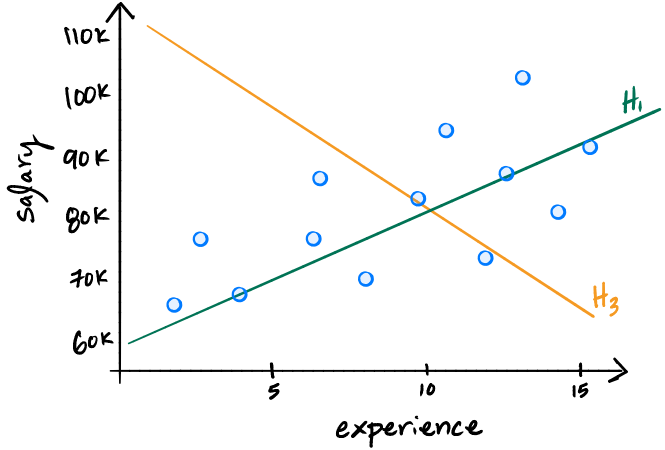
$$\left(H(x_i) - y_i\right)^2$$

- The **mean squared error** (MSE) of H:

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

- **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, find:

$$H^* = \underset{\text{linear } H}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left(H(x_i) - y_i \right)^2$$

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

Minimizing the MSE

- The MSE is a function of a function:

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

- But since H is linear, $H(x) = w_1 x + w_0$.

$$R_{\text{sq}}(w_1, w_0) = \frac{1}{n} \sum_{i=1}^{n} \left((w_1 x + w_0) - y_i \right)^2$$

- Now it's a function of w_1, w_0.

Updated Goal

- Find slope w_1 and intercept w_0 which minimize the MSE, $R_{\text{sq}}(w_1, w_0)$:

$$R_{\text{sq}}(w_1, w_0) = \frac{1}{n} \sum_{i=1}^{n} \left((w_1 x + w_0) - y_i \right)^2$$

- Strategy: multivariate calculus.

Recall: the gradient

- If $f(x, y)$ is a function of two variables, the **gradient** of f at the point (x_0, y_0) is a **vector** of **partial derivatives**:

$$\nabla f(x_0, y_0) = \begin{pmatrix} \frac{\partial f}{\partial x}(x_0) \\ \frac{\partial f}{\partial y}(y_0) \end{pmatrix}$$

- **Key Fact #1**: derivative : tangent line :: gradient : tangent plane

- **Key Fact #2**: points in direction of biggest increase

- **Key Fact #3**: if the gradient is zero at critical points.

Strategy

To minimize $R(w_1, w_0)$: compute the gradient, set equal to zero, solve.

$$R_{\mathrm{sq}}(w_1, w_0) = \frac{1}{n} \sum_{i=1}^{n} \left((w_1 x + w_0) - y_i \right)^2$$

$$\frac{\partial R_{\mathrm{sq}}}{\partial w_1} =$$

$$R_{\text{sq}}(w_1, w_0) = \frac{1}{n}\sum_{i=1}^{n}\left((w_1 x + w_0) - y_i\right)^2$$

$$\frac{\partial R_{\text{sq}}}{\partial w_0} =$$