

Lecture 6 – Simple Linear Regression



DSC 40A, Fall 2021 @ UC San Diego

Suraj Rampure, with help from **many others**



Freya Holmér

@FreyaHolmer

btw these large scary math symbols
are just for-loops

Summation
(capital sigma)

$$\sum_{n=0}^4 3n$$

```
sum = 0;  
for( n=0; n<=4; n++ )  
    sum += 3*n;
```

Product
(capital pi)

$$\prod_{n=1}^4 2n$$

```
prod = 1;  
for( n=1; n<=4; n++ )  
    prod *= 2*n;
```

7:51 PM · 11 Sep 21 · [Twitter Web App](#)

Announcements

- ▶ Homework 3 will be released after lecture, will be due on **Monday 10/18 at 11:59pm**. Will be shorter than usual.
- ▶ Groupwork 3 will be released after lecture, will be due on **Thursday 10/14 at 11:59pm**.
- ▶ **DISCUSSION SECTION ON WEDNESDAY WILL BE IN-PERSON!**
 - ▶ Wednesday, 6-6:50PM, Center Hall 113.
- ▶ Homework 1, Groupwork 1, and Groupwork 2 grades are released on Gradescope.
- ▶ Midterm is **Thursday, 10/21 during class time**.
 - ▶ **Review Session:** Tuesday 10/19, 5-8PM, PCYNH 109.
 - ▶ See <https://dsc40a.com/resources>.

Agenda

- ▶ Recap of gradient descent.
- ▶ Prediction rules.
- ▶ Minimizing mean squared error, again.

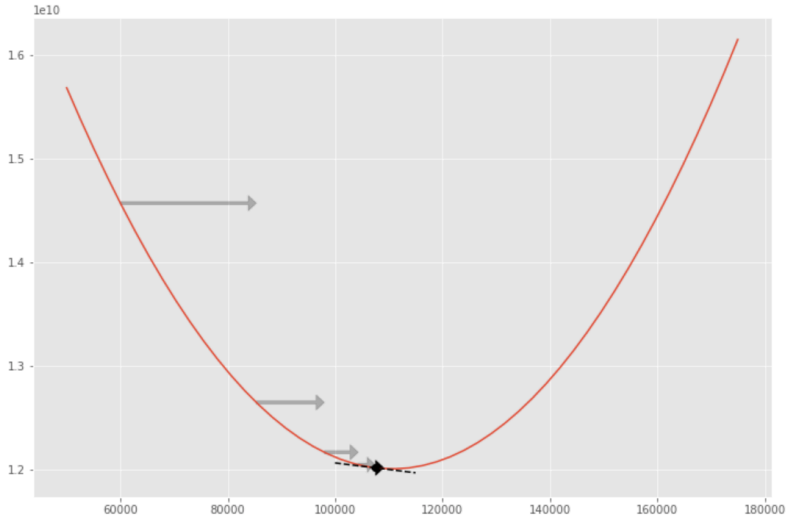
Recap: gradient descent

Gradient descent

- ▶ The goal of gradient descent is to minimize a function $R(h)$.
- ▶ Gradient descent starts off with an initial guess h_0 of where the minimizing input to $R(h)$ is, and on each step tries to get closer to the minimizing input h^* by moving opposite the direction of the slope:

$$h_i = h_{i-1} - \alpha \cdot \frac{dR}{dh}(h_{i-1})$$

- ▶ α is known as the learning rate, or step size. It controls how much we update our guesses by on each iteration.
- ▶ Gradient descent terminates once the guesses h_i and h_{i-1} stop changing much.



See Lecture 5's supplemental notebook for animations.

When does gradient descent work?

- ▶ A function f is convex if, for any two inputs a and b , the line segment connecting the two points $(a, f(a))$ and $(b, f(b))$ does not go below the function f .
 - ▶ $R_{abs}(h) = \frac{1}{n} \sum_{i=1}^n |y_i - h|$: convex.
 - ▶ $R_{sq}(h) = \frac{1}{n} \sum_{i=1}^n (y_i - h)^2$: convex.
 - ▶ $R_{ucsd}(h) = \frac{1}{n} \sum_{i=1}^n [1 - e^{-(y_i - h)^2 / \sigma^2}]$: not convex.
- ▶ **Theorem:** If $R(h)$ is convex and differentiable then gradient descent converges to a **global minimum** of R given an appropriate step size.

Prediction rules

How do we predict someone's salary?

After collecting salary data, we...

1. Choose a loss function.
 2. Find the best prediction by minimizing empirical risk.
- ▶ So far, we've been predicting future salaries without using any information about the individual (e.g. GPA, years of experience, number of LinkedIn connections).
 - ▶ **New focus:** How do we incorporate this information into our prediction-making process?

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?

Think of features as columns in a DataFrame (i.e. table).

	YearsExperience	Age	FormalEducation	Salary
0	6.37	28.39	Master's degree (MA, MS, M.Eng., MBA, etc.)	120000.0
1	0.35	25.78	Some college/university study without earning ...	120000.0
2	4.05	31.04	Bachelor's degree (BA, BS, B.Eng., etc.)	70000.0
3	18.48	38.78	Bachelor's degree (BA, BS, B.Eng., etc.)	185000.0
4	4.95	33.45	Master's degree (MA, MS, M.Eng., MBA, etc.)	125000.0

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.
- ▶ In other words, we think that there is a function H such 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 .

Possible prediction rules

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

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

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

- ▶ These are all valid prediction rules.
- ▶ Some are better than others.

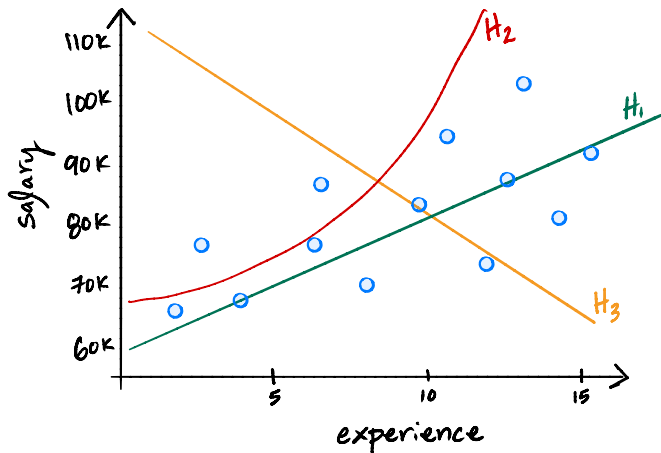
Comparing predictions

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

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

- ▶ See which rule works better on data.

Example

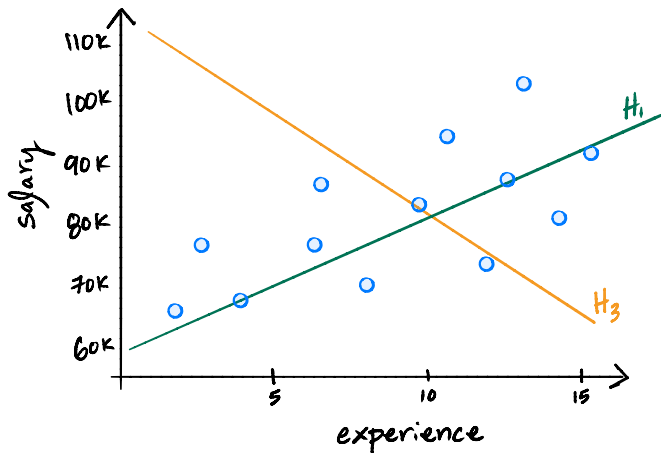


Quantifying the quality of a prediction rule H

- ▶ Our prediction for person i 's salary is $H(x_i)$.
- ▶ As before, we'll use a **loss function** to quantify the quality of our predictions.
 - ▶ Absolute loss: $|y_i - H(x_i)|$.
 - ▶ Squared loss: $(y_i - H(x_i))^2$.
- ▶ We'll use squared loss, since it's differentiable.
- ▶ Using squared loss, the **empirical risk** (mean squared error) of the prediction rule H is:

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

Mean squared error



Finding the best prediction rule

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

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

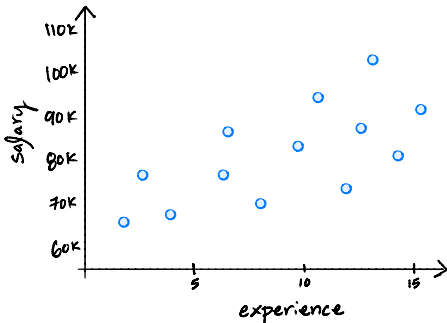
- ▶ There's a problem.

Discussion Question

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

a) Yes b) No

To answer, go to [menti.com](https://www.menti.com) and enter the code 88515429.



Problem

- ▶ We can make mean squared 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: $H(x) = w_0 + w_1 x$.
 - ▶ Quadratic: $H(x) = w_0 + w_1 x_1 + w_2 x^2$.
 - ▶ 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 $\mathbb{R} \rightarrow \mathbb{R}$, find the function H^* with the smallest mean squared error.
 - ▶ Linear functions are of the form $H(x) = w_0 + w_1 x$.
 - ▶ They are defined by a slope (w_1) and intercept (w_0).
- ▶ That is, H^* should be the linear function that minimizes

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

- ▶ This problem is called **least squares regression**.
 - ▶ “Simple linear regression” refers to linear regression with a single predictor variable.

Minimizing mean squared error for the linear prediction rule

Minimizing the mean squared error

- ▶ The MSE is a function R_{sq} of a function H .

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

- ▶ But since H is linear, we know $H(x_i) = w_0 + w_1 x_i$.

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

- ▶ Now R_{sq} is a function of w_0 and w_1 .
- ▶ We call w_0 and w_1 **parameters**.
 - ▶ Parameters define our prediction rule.

Updated goal

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

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

- Strategy: multivariable 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**:

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

- ▶ **Key Fact #1:** The derivative is to the tangent line as the gradient is to the tangent plane.
- ▶ **Key Fact #2:** The gradient points in the direction of the biggest increase.
- ▶ **Key Fact #3:** The gradient is zero at critical points.

Strategy

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

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

Discussion Question

Choose the expression that equals $\frac{\partial R_{sq}}{\partial w_0}$.

- a) $\frac{1}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i))$
- b) $-\frac{1}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i))$
- c) $-\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i)) x_i$
- d) $-\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i))$

Go to [menti.com](https://www.menti.com) and enter the code 8851 5429.

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

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

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

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

Strategy

$$-\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i)) = 0 \qquad -\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i)) x_i = 0$$

1. Solve for w_0 in first equation.
 - The result becomes w_0^* , since it is the “best intercept”.
2. Plug w_0^* into second equation, solve for w_1 .
 - The result becomes w_1^* , since it is the “best slope”.

Solve for w_0^*

$$-\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i)) = 0$$

Solve for w_1^*

$$-\frac{2}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i)) x_i = 0$$

Least squares solutions

- We've found that the values w_0^* and w_1^* that minimize the function $R_{sq}(w_0, w_1) = \frac{1}{n} \sum_{i=1}^n (y_i - (w_0 + w_1 x_i))^2$ are

$$w_1^* = \frac{\sum_{i=1}^n (y_i - \bar{y})x_i}{\sum_{i=1}^n (x_i - \bar{x})x_i} \qquad w_0^* = \bar{y} - w_1^* \bar{x}$$

where

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \qquad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

- Let's re-write the slope w_1^* to be a bit more symmetric.

Key fact

The **sum of deviations from the mean** for any dataset is 0.

$$\sum_{i=1}^n (x_i - \bar{x}) = 0 \qquad \sum_{i=1}^n (y_i - \bar{y}) = 0$$

Proof:

Equivalent formula for w_1^*

Claim

$$w_1^* = \frac{\sum_{i=1}^n (y_i - \bar{y})x_i}{\sum_{i=1}^n (x_i - \bar{x})x_i} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Proof:

Least squares solutions

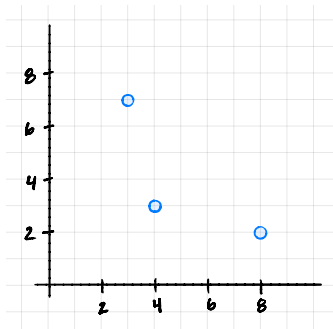
- ▶ The **least squares solutions** for the slope w_1^* and intercept w_0^* are:

$$w_1^* = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \qquad w_0^* = \bar{y} - w_1^* \bar{x}$$

- ▶ We also say that w_0^* and w_1^* are **optimal parameters**.
- ▶ To make predictions about the future, we use the prediction rule

$$H^*(x) = w_0^* + w_1^* x$$

Example



$$\bar{x} =$$

$$\bar{y} =$$

$$w_1^* = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} =$$

$$w_0^* = \bar{y} - w_1^* \bar{x} =$$

x_i	y_i	$(x_i - \bar{x})$	$(y_i - \bar{y})$	$(x_i - \bar{x})(y_i - \bar{y})$	$(x_i - \bar{x})^2$
3	7				
4	3				
8	2				

Summary

Summary, next time

- ▶ We introduced the linear prediction rule, $H(x) = w_0 + w_1 x$.
- ▶ To determine the best choice of slope (w_1) and intercept (w_0), we chose the squared loss function $(y_i - H(x_i))^2$ and minimized empirical risk $R_{sq}(w_0, w_1)$:

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

- ▶ After solving for w_0^* and w_1^* through partial differentiation, we have a prediction rule $H^*(x) = w_0^* + w_1^* x$ that we can use to make predictions about the future.
- ▶ **Next time:** Revisiting correlation from DSC 10. Revisiting gradient descent. Introducing a linear algebraic formulation of linear regression.