Module 11 – The Normal Equations

DSC 40A, Summer 2023

Agenda

- Recap of Module 10.
- Minimizing mean squared error.
- Incorporating multiple features.

Recap of Module 10

Reframing regression using linear algebra

- Last time, we used linear algebra to reformulate our problem of fitting a linear prediction rule

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

- We defined a **design matrix** X, **parameter vector** $\vec{w}$, and **observation vector** $\vec{y}$ as follows:

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \dots & \dots \\ 1 & x_n \end{bmatrix}, \qquad \vec{w} = \begin{bmatrix} w_0 \\ w_1 \end{bmatrix}, \qquad \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix}$$

- Then we rewrote our prediction rule as a matrix-vector multiplication, defining the **hypothesis vector** $\vec{h}$ as

$$\vec{h} = X\vec{w}$$

Minimizing mean squared error

- With our new linear algebra formulation of regression, our mean squared error now looks like:

$$R_{sq}(\vec{w}) = \frac{1}{n}||\vec{y} - X\vec{w}||^2$$

- Today, we will minimize this function using calculus.
- We already saw a sneak peek of the result. The optimal parameter vector $\vec{w}^*$ is[1]

$$\vec{w}^* = (X^T X)^{-1} X^T \vec{y}$$

  - This gives the same w_0^* and w_1^* as our formulas from Module 6.

[1] assuming $X^T X$ is invertible

Minimizing mean squared error, again

Some key linear algebra facts

If A and B are matrices, and $\vec{u}, \vec{v}, \vec{w}, \vec{z}$ are vectors:

- $(A + B)^T = A^T + B^T$
- $(AB)^T = B^T A^T$
- $\vec{u} \cdot \vec{v} = \vec{v} \cdot \vec{u} = \vec{u}^T \vec{v} = \vec{v}^T \vec{u}$
- $\|\vec{u}\|^2 = \vec{u} \cdot \vec{u}$
- $(\vec{u} + \vec{v}) \cdot (\vec{w} + \vec{z}) = \vec{u} \cdot \vec{w} + \vec{u} \cdot \vec{z} + \vec{v} \cdot \vec{w} + \vec{v} \cdot \vec{z}$

Goal

- We want to minimize the mean squared error:

$$R_{\text{sq}}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

- Strategy: Calculus.

- **Problem:** This is a *function of a vector*. What does it even mean to take the derivative of $R_{\text{sq}}(\vec{w})$ with respect to a vector $\vec{w}$?

A function of a vector

- **Solution:** A function *of a vector* is really just a function *of multiple variables*, which are the components of the vector. In other words,

$$R_{\text{sq}}(\vec{w}) = R_{\text{sq}}(w_0, w_1, \dots, w_d)$$

  where $w_0, w_1, \dots, w_d$ are the entries of the vector $\vec{w}$.[2]

- We know how to deal with derivatives of multivariable functions: the gradient!

[2] In our case, $\vec{w}$ has just two components, w_0 and w_1. We'll be more general since we eventually want to use prediction rules with even more parameters.

The gradient with respect to a vector

- The **gradient of $R_{\text{sq}}(\vec{w})$ with respect to $\vec{w}$** is the vector of partial derivatives:

$$\nabla_{\vec{w}} R_{\text{sq}}(\vec{w}) = \frac{dR_{\text{sq}}}{d\vec{w}} = \begin{bmatrix} \frac{\partial R_{\text{sq}}}{\partial w_0} \\ \frac{\partial R_{\text{sq}}}{\partial w_1} \\ \vdots \\ \frac{\partial R_{\text{sq}}}{\partial w_d} \end{bmatrix}$$

where $w_0, w_1, \ldots, w_d$ are the entries of the vector $\vec{w}$.

Example gradient calculation

Example: Suppose $f(\vec{x}) = \vec{a} \cdot \vec{x}$, where $\vec{a}$ and $\vec{x}$ are vectors in $\mathbb{R}^n$.
What is $\frac{d}{d\vec{x}} f(\vec{x})$?

Goal

- We want to minimize the mean squared error:

$$R_{\text{sq}}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$$

- Strategy:
  1. Compute the gradient of $R_{\text{sq}}(\vec{w})$.
  2. Set it to zero and solve for $\vec{w}$.
     - The result is called $\vec{w}^*$.

- Let's start by rewriting the mean squared error in a way that will make it easier to compute its gradient.

Rewriting mean squared error

$$R_{\text{sq}}(\vec{w}) = \frac{1}{n}\|\vec{y} - X\vec{w}\|^2$$

Discussion Question

Which of the following is equivalent to $R_{\text{sq}}(\vec{w})$?

a) $\frac{1}{n}(\vec{y} - X\vec{w}) \cdot (X\vec{w} - \vec{y})$

b) $\frac{1}{n}\sqrt{(\vec{y} - X\vec{w}) \cdot (\vec{y} - X\vec{w})}$

c) $\frac{1}{n}(\vec{y} - X\vec{w})^T(\vec{y} - X\vec{w})$

d) $\frac{1}{n}(\vec{y} - X\vec{w})(\vec{y} - X\vec{w})^T$

Rewriting mean squared error

$R_{\text{sq}}(\vec{w}) = \frac{1}{n} \|\vec{y} - X\vec{w}\|^2$

Rewriting mean squared error

$R_{\text{sq}}(\vec{w}) =$

Compute the gradient

$$
\begin{aligned}
\frac{dR_{\text{sq}}}{d\vec{w}} &= \frac{d}{d\vec{w}}\left(\frac{1}{n}\left[\vec{y}\cdot\vec{y} - 2X^T\vec{y}\cdot\vec{w} + \vec{w}^TX^TX\vec{w}\right]\right) \\
&= \frac{1}{n}\left[\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) - \frac{d}{d\vec{w}}\left(2X^T\vec{y}\cdot\vec{w}\right) + \frac{d}{d\vec{w}}\left(\vec{w}^TX^TX\vec{w}\right)\right]
\end{aligned}
$$

Compute the gradient

$$
\begin{aligned}
\frac{dR_{\text{sq}}}{d\vec{w}} &= \frac{d}{d\vec{w}}\left(\frac{1}{n}\left[\vec{y}\cdot\vec{y} - 2X^T\vec{y}\cdot\vec{w} + \vec{w}^TX^TX\vec{w}\right]\right) \\
&= \frac{1}{n}\left[\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) - \frac{d}{d\vec{w}}\left(2X^T\vec{y}\cdot\vec{w}\right) + \frac{d}{d\vec{w}}\left(\vec{w}^TX^TX\vec{w}\right)\right]
\end{aligned}
$$

- $\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) = 0.$
    - Why? $\vec{y}$ is a constant with respect to $\vec{w}$.

Compute the gradient

$$\frac{dR_{\text{sq}}}{d\vec{w}} = \frac{d}{d\vec{w}}\left(\frac{1}{n}\left[\vec{y}\cdot\vec{y} - 2X^T\vec{y}\cdot\vec{w} + \vec{w}^T X^T X\vec{w}\right]\right)$$
$$= \frac{1}{n}\left[\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) - \frac{d}{d\vec{w}}\left(2X^T\vec{y}\cdot\vec{w}\right) + \frac{d}{d\vec{w}}\left(\vec{w}^T X^T X\vec{w}\right)\right]$$

- $\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) = 0.$
  - Why? $\vec{y}$ is a constant with respect to $\vec{w}$.
- $\frac{d}{d\vec{w}}\left(\vec{2}X^T\vec{y}\cdot\vec{w}\right) = 2X^T y.$
  - Why? We already showed $\frac{d}{d\vec{x}}\vec{a}\cdot\vec{x} = \vec{a}.$

Compute the gradient

$$\frac{dR_{\text{sq}}}{d\vec{w}} = \frac{d}{d\vec{w}}\left(\frac{1}{n}\left[\vec{y}\cdot\vec{y} - 2X^T\vec{y}\cdot\vec{w} + \vec{w}^TX^TX\vec{w}\right]\right)$$
$$= \frac{1}{n}\left[\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) - \frac{d}{d\vec{w}}\left(2X^T\vec{y}\cdot\vec{w}\right) + \frac{d}{d\vec{w}}\left(\vec{w}^TX^TX\vec{w}\right)\right]$$

- $\frac{d}{d\vec{w}}(\vec{y}\cdot\vec{y}) = 0.$
  - Why? $\vec{y}$ is a constant with respect to $\vec{w}$.
- $\frac{d}{d\vec{w}}\left(\vec{2}X^T\vec{y}\cdot\vec{w}\right) = 2X^Ty.$
  - Why? We already showed $\frac{d}{d\vec{x}}\vec{a}\cdot\vec{x} = \vec{a}$.
- $\frac{d}{d\vec{w}}\left(\vec{w}^TX^TX\vec{w}\right) = 2X^TX\vec{w}.$
  - Why? See Homework 2.

Compute the gradient

$$\begin{aligned}
\frac{dR_{\text{sq}}}{d\vec{w}} &= \frac{d}{d\vec{w}}\left(\frac{1}{n}\left[\vec{y}\cdot\vec{y} - 2X^T\vec{y}\cdot\vec{w} + \vec{w}^TX^TX\vec{w}\right]\right) \\
&= \frac{1}{n}\left[\frac{d}{d\vec{w}}\left(\vec{y}\cdot\vec{y}\right) - \frac{d}{d\vec{w}}\left(2X^T\vec{y}\cdot\vec{w}\right) + \frac{d}{d\vec{w}}\left(\vec{w}^TX^TX\vec{w}\right)\right]
\end{aligned}$$

The normal equations

- To minimize $R_{\text{sq}}(\vec{w})$, set gradient to zero and solve for $\vec{w}$:

$$-2X^T\vec{y} + 2X^TX\vec{w} = 0$$
$$\implies X^TX\vec{w} = X^T\vec{y}$$

- This is a system of equations in matrix form, called the **normal equations**.

- If X^TX is invertible, the solution is

$$\vec{w}^* = (X^TX)^{-1}X^T\vec{y}$$

- This is equivalent to the formulas for w_0^* and w_1^* we saw before!
  - Benefit – this can be easily extended to more complex prediction rules.

Incorporating multiple features

Incorporating multiple features

- How do we predict salary given **multiple** features?
- We believe salary is a function of experience *and* GPA.
- In other words, we believe there is a function H so that:

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

- Recall: H is a **prediction rule**.
- **Our goal**: find a good prediction rule, H.

Example prediction rules

$$H_1(\text{experience}, \text{GPA}) = \$2,000 \times (\text{experience}) + \$40{,}000 \times \frac{\text{GPA}}{4.0}$$

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

$$H_3(\text{experience}, \text{GPA}) = \cos(\text{experience}) + \sin(\text{GPA})$$

Linear prediction rules

- We'll restrict ourselves to **linear** prediction rules:

$$H(\text{experience}, \text{GPA}) = w_0 + w_1(\text{experience}) + w_2(\text{GPA})$$

- As before, we can solve the **normal equations** to find w_0^*, w_1^*, and w_2^*. All we need to do is change the design matrix X.

- Linear regression with multiple features is called **multiple linear regression**.

Geometric interpretation

Question: The prediction rule

$$H(\text{experience}) = w_0 + w_1(\text{experience})$$

looks like a line in 2D.

1. How many dimensions do we need to graph

$$H(\text{experience}, \text{GPA}) = w_0 + w_1(\text{experience}) + w_2(\text{GPA})$$

2. What is the shape of the prediction rule?

Example dataset

- For each of n people, collect each feature, plus salary:

Person #	Experience	GPA	Salary
1	3	3.7	85,000
2	6	3.3	95,000
3	10	3.1	105,000

- We represent each person with a **feature vector**:

$$\vec{x}_1 = \begin{bmatrix} 3 \\ 3.7 \end{bmatrix}, \qquad \vec{x}_2 = \begin{bmatrix} 6 \\ 3.3 \end{bmatrix}, \qquad \vec{x}_3 = \begin{bmatrix} 10 \\ 3.1 \end{bmatrix}$$

The hypothesis vector

- When our prediction rule is

$$H(\text{experience}, \text{GPA}) = w_0 + w_1(\text{experience}) + w_2(\text{GPA}),$$

the hypothesis vector $\vec{h} \in \mathbb{R}^n$ can be written

$$\vec{h} = \begin{bmatrix} H(\text{experience}_1, \text{GPA}_1) \\ H(\text{experience}_2, \text{GPA}_2) \\ \dots \\ H(\text{experience}_n, \text{GPA}_n) \end{bmatrix} = \begin{bmatrix} 1 & \text{experience}_1 & \text{GPA}_1 \\ 1 & \text{experience}_2 & \text{GPA}_2 \\ \dots & \dots & \dots \\ 1 & \text{experience}_n & \text{GPA}_n \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix}$$

Finding the optimal parameters

- To find the best parameter vector, $\vec{w}^*$, we can use the design matrix and observation vector

$$X = \begin{bmatrix} 1 & \text{experience}_1 & \text{GPA}_1 \\ 1 & \text{experience}_2 & \text{GPA}_2 \\ \dots & \dots & \dots \\ 1 & \text{experience}_n & \text{GPA}_n \end{bmatrix}, \vec{y} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix}$$

and solve the **normal equations**

$$X^T X \vec{w}^* = X^T \vec{y}$$

- Notice that the rows of the design matrix are the (transposed) feature vectors, with an additional 1 in front.

Notation for multiple linear regression

- We will need to keep track of multiple[3] features for every individual in our data set.

- As before, subscripts distinguish between individuals in our data set. We have n individuals (or **training examples**).

- Superscripts distinguish between features.[4] We have d features.
  - experience = $x^{(1)}$
  - GPA = $x^{(2)}$

---

[3] In practice, we might use hundreds or even thousands of features.
[4] Think of them as new variable names, such as new letters.

Augmented feature vectors

- The **augmented feature vector** $\text{Aug}(\vec{x})$ is the vector obtained by adding a 1 to the front of feature vector $\vec{x}$:

$$\vec{x} = \begin{bmatrix} x^{(1)} \\ x^{(2)} \\ \vdots \\ x^{(d)} \end{bmatrix} \qquad \text{Aug}(\vec{x}) = \begin{bmatrix} 1 \\ x^{(1)} \\ x^{(2)} \\ \vdots \\ x^{(d)} \end{bmatrix} \qquad \vec{w} = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix}$$

- Then, our prediction rule is

$$\begin{aligned} H(\vec{x}) &= w_0 + w_1 x^{(1)} + w_2 x^{(2)} + \ldots + w_d x^{(d)} \\ &= \vec{w} \cdot \text{Aug}(\vec{x}) \end{aligned}$$

The general problem

- We have n data points (or **training examples**): $(\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n)$ where each $\vec{x}_i$ is a feature vector of d features:

$$\vec{x}_i = \begin{bmatrix} x_i^{(1)} \\ x_i^{(2)} \\ \dots \\ x_i^{(d)} \end{bmatrix}$$

- We want to find a good linear prediction rule:

$$\begin{aligned} H(\vec{x}) &= w_0 + w_1 x^{(1)} + w_2 x^{(2)} + \dots + w_d x^{(d)} \\ &= \vec{w} \cdot \text{Aug}(\vec{x}) \end{aligned}$$

The general solution

- Use design matrix

$$X = \begin{bmatrix} 1 & x_1^{(1)} & x_1^{(2)} & \dots & x_1^{(d)} \\ 1 & x_2^{(1)} & x_2^{(2)} & \dots & x_2^{(d)} \\ \dots & \dots & \dots & & \dots \\ 1 & x_n^{(1)} & x_n^{(2)} & \dots & x_n^{(d)} \end{bmatrix} = \begin{bmatrix} \mathrm{Aug}(\vec{x}_1)^T \\ \mathrm{Aug}(\vec{x}_2)^T \\ \dots \\ \mathrm{Aug}(\vec{x}_n)^T \end{bmatrix}$$

and observation vector to solve the **normal equations**

$$X^T X \vec{w}^* = X^T \vec{y}$$

to find the optimal parameter vector.

Interpreting the parameters

- With d features, $\vec{w}$ has $d + 1$ entries.
- w_0 is the **bias**, also known as the **intercept**.
- $w_1, \dots, w_d$ each give the **weight**, i.e. **coefficient**, of a feature.

$$H(\vec{x}) = w_0 + w_1 x^{(1)} + \dots + w_d x^{(d)}$$

- The sign of w_i tells us about the relationship between ith feature and the output of our prediction rule.

Summary

Summary

- We minimized the mean squared error for the prediction rule $H(x) = w_0 + w_1 x$, which was

$$R_{sq}(\vec{w}) = \frac{1}{n} ||\vec{y} - X\vec{w}||^2$$

- We found that the minimizing $\vec{w}$ satisfies the **normal equations**, $X^T X \vec{w} = X^T \vec{y}$.
  - If $X^T X$ is invertible, the solution is:

$$\vec{w}^* = (X^T X)^{-1} X^T \vec{y}$$

- These same normal equations can be used to solve the **multiple linear regression** problem, where we use multiple features to predict an outcome. We simply need to adjust the design matrix X.