CSCI 5525 Machine Learning Fall 2019

Lecture 2: Linear Regression

Sep 5 2019

Lecturer: Steven Wu *Scribe: Steven Wu*

A curious manager

Suppose you work at a restaurant and you want to predict how much the customers tip. Let's say the restaurant is not very busy, and as a manager you have all the free time to record the following kind of data:

	total_bill	tip
0	16.99	1.01
1	10.34	1.66
2	21.01	3.50
3	23.68	3.31
4	24.59	3.61
5	25.29	4.71
6	8.77	2.00
7	26.88	3.12
8	15.04	1.96
9	14.78	3.23
10	10.27	1.71
11	35.26	5.00
12	15.42	1.57
13	18.43	3.00
14	14.83	3.02
15	21.58	3.92

Figure 1: A snapshot of the dataset "Gopher Express"

Perhaps the simplest prediction you could make is to predict every tip amount based on the mean estimate $\hat{\mu} = 2.99$. Can you do better? Well, you figure that you figure that customers often tip based on the size of the meals they had, so you decide to take advantage of this side information. Now you remember your first lecture in machine learning 5525 and realize that this is just a *supervised learning* problem: you are given data $(X_1, Y_1), \ldots, (X_n, Y_n)$ are drawn i.i.d. from the underlying distribution, and the range of X, denoted $\mathcal{X}$ and the range of Y, denoted $\mathcal{Y}$

are both $\mathbb{R}$. Now you would like to find a *predictor/prediction function* $\hat{f}: \mathcal{X} \to \mathcal{Y}$, so that in the future whenever you observe some new X, you can form a prediction $\hat{f}(X)$.

1 Linear regression

Let's use a linear model to predict Y with an *affine function*:

$$\hat{f}(X) = \mathbf{w}^\intercal \begin{pmatrix} X \\ 1 \end{pmatrix}$$

where $\mathbf{w} = \begin{pmatrix} w_1 \\ w_0 \end{pmatrix}$. Appending 1 makes this an affine function. To slightly abuse notation, we will just write x to denote $\begin{pmatrix} x \\ 1 \end{pmatrix}$, and view it as the input feature. More generally, x can be in $\mathbb{R}^d$.

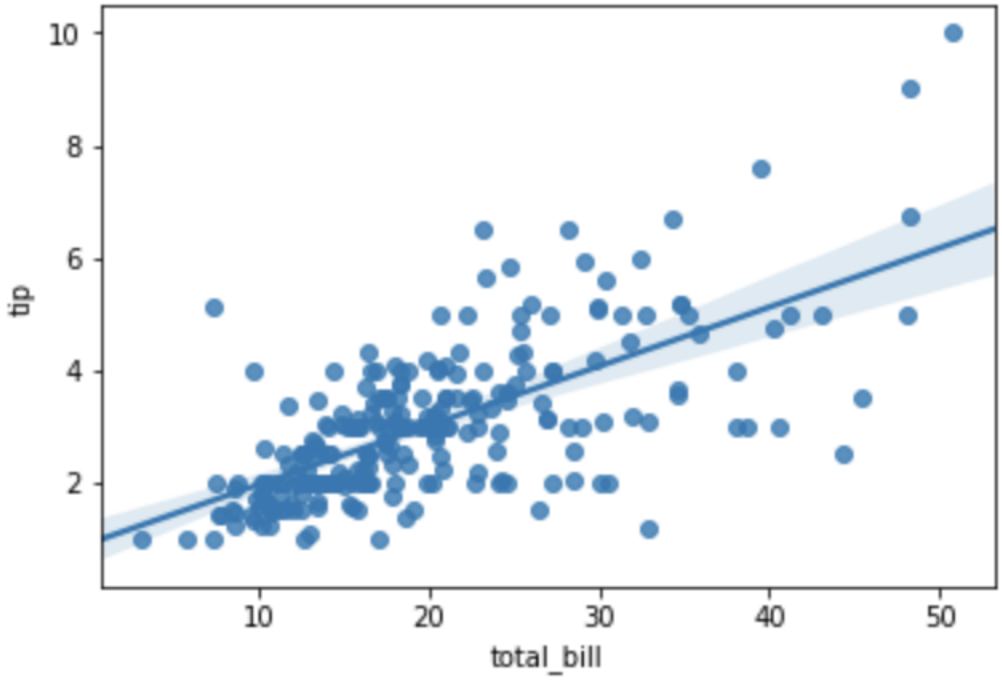

Figure 2: Fitting a affine function.

But which line should we choose?

2 ERM with Least Squares

Much of supervised learning follows the *empirical risk minimization (ERM)* approach. In the case of *least squares regression*:

- **Loss function:** the least square loss for prediction $\hat{y} = \hat{f}(x)$

$$\ell(y, \hat{y}) = \ell(y, \hat{y}) = (y - \hat{y})^2$$

  Sometimes we re-scale the loss by $1/2$.

- Goal: minimize least squares empirical risk

$$\hat{\mathcal{R}}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, \hat{f}(x_i)) = \frac{1}{n}\sum_{i=1}^{n} y_i - \hat{f}(x_i))^2$$

- In other words, we find a $\mathbf{w} \in \mathbb{R}^d$ (in the example $d = 2$) that minimizes $\hat{\mathcal{R}}(\mathbf{w})$

$$\arg\min_{w \in \mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} (y_i - \mathbf{w}^\intercal x_i)^2$$

More generally, we can apply the recipe of ERM as follows:

- Pick a family of models/predictors $\mathcal{F}$. (Linear models in this lecture.)
- Pick a loss function ℓ.
- Minimize the empirical risk over the model or equivalently the parameters.

In the course, we will see more examples like this, and will also learn about why ERM should work.

3 Least squares solution

Let's do some linear algebra and think about matrix forms. We can define the design matrix:

$$A = \begin{bmatrix} \leftarrow x_1^\intercal \rightarrow \\ \vdots \\ \leftarrow x_n^\intercal \rightarrow \end{bmatrix}$$

and response vector:

$$\mathbf{b} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

Then the empirical risk can be written as

$$\hat{\mathcal{R}}(w) = \frac{1}{n}\sum_{i=1}^{n} (y_i - \mathbf{w}^\intercal x_i)^2 = \frac{1}{n}\|Aw - b\|^2$$

Note that re-scaling the loss doesn't change the solution, so the least squares solution is given by

$$\hat{\mathbf{w}} \in \arg\min_{\mathbf{w}\in\mathbb{R}^d} \|A\mathbf{w} - b\|^2$$

From calculus, we learn that a necessary condition for $\mathbf{w}$ to be a minimizer of $\hat{\mathcal{R}}$ is that it needs to be a stationery point of the (re-scaled) risk function:

$$\nabla\hat{\mathcal{R}}(\mathbf{w}) = \mathbf{0}$$

This translates to the following condition

$$(A^\intercal A)\,\mathbf{w} - A^\intercal\mathbf{b} = \mathbf{0} \text{ or equivalently } (A^\intercal A)\,\mathbf{w} = A^\intercal\mathbf{b} \tag{1}$$

Note the solutions may not be unique.

Claim 3.1. *The condition in equation 1 is also a sufficient condition for optimality.*

Proof. Consider any $\mathbf{w}'$.

$$\begin{aligned}\|A\mathbf{w}' - \mathbf{b}\|^2 &= \|A\mathbf{w}' - A\mathbf{w} + A\mathbf{w} - \mathbf{b}\|^2 \\ &= \|A\mathbf{w}' - A\mathbf{w}\|^2 + 2(A\mathbf{w}' - A\mathbf{w})^\intercal(A\mathbf{w} - \mathbf{b}) + \|A\mathbf{w} - \mathbf{b}\|^2\end{aligned}$$

Observe that

$$(A\mathbf{w}' - A\mathbf{w})^\intercal(A\mathbf{w} - \mathbf{b}) = (w' - w)^\intercal A^\intercal(A\mathbf{w} - \mathbf{b}) = (w' - w)^\intercal(A^\intercal A\mathbf{w} - A^\intercal\mathbf{b}) = 0,$$

where the last equality follows from the condition in (1). It follows that

$$\|A\mathbf{w}' - \mathbf{b}\|^2 = \underbrace{\|A\mathbf{w}' - A\mathbf{w}\|^2}_{\geq 0} + \|A\mathbf{w} - \mathbf{b}\|^2$$

This means any $\mathbf{w}'$ cannot have smaller loss. □

We can also prove the claim above with convexity.

The matrix $A^\intercal A$ is sometimes called the *covariance matrix*. Now if we are lucky and the covariance matrix $A^\intercal A$ is invertible, then the least squares solution is simply the following

$$\mathbf{w}^* = (A^\intercal A)^{-1}(A\mathbf{b}).$$

So how do we compute a solution for (1) when the covariance matrix is not invertible?

4 Singular Value Decomposition

We will first recall how *singular value decomposition (SVD)* works, *and also fix a mistake I had in class*. Given any matrix $M \in \mathbb{R}^{m \times n}$, we want to factorize the matrix as $M = USV^\intercal$, where

- r is the rank of the matrix M;
- $U \in \mathbb{R}^{m \times r}$ is orthonormal, that is $U^\intercal U = I_r$;
- $V \in \mathbb{R}^{n \times r}$ is orthonormal, that is $V^\intercal V = I_r$;
- $S \in \mathbb{R}^{r \times r}$ is a diagonal matrix $\text{diag}(s_1, \ldots, s_r)$.

We could also express the factorization as a sum

$$M = \sum_{i=1}^{r} s_i u_i v_i^\intercal$$

where each u_i is a column vector for U and each v_i is a column vector for V. Note that $\{u_i\}$ spans the column space of M and $\{v_i\}$ spans the row space of M.

This allows us to define the *(Moore-Penrose) pseudoinverse*

$$M^+ = \sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\intercal.$$

Basically, we take the inverse of the singular values and reverse the positions of the v_i and u_i within each term.

Note that if M is the full-zero matrix, then M^+ is also just all zeros.

Now let's return to the problem of least squares regression, where would like to fine a solution for (1). Now consider $\mathbf{w}^* = A^+ b$.

Claim 4.1. *The vector* $\mathbf{w}^*$ *satisfies* (1).

Proof. We can derive the following:

$$A^\intercal A w = \left(\sum_{i=1}^{r} s_i v_i u_i^\intercal \right) \left(\sum_{i=1}^{r} s_i u_i v_i^\intercal \right) \left(\sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\intercal \right) \mathbf{b} \tag{2}$$

$$= \left(\sum_{i=1}^{r} s_i v_i u_i^\intercal \right) \left(\sum_{i=1}^{r} u_i v_i^\intercal v_i u_i^\intercal \right) \mathbf{b} \tag{3}$$

$$= \left(\sum_{i=1}^{r} s_i v_i u_i^\intercal \right) \left(\sum_{i=1}^{r} u_i u_i^\intercal \right) \mathbf{b} \tag{4}$$

$$= \left(\sum_{i=1}^{r} s_i v_i u_i^\intercal \right) \mathbf{b} \tag{5}$$

$$= A^\intercal \mathbf{b} \tag{6}$$

where the (4) follows because $v_i^\intercal v_j = 0$ whenever $i \neq j$ and $v_i^\intercal v_i = 1$, and (5) follows from the fact that $u_i^\intercal u_j = 0$ whenever $i \neq j$ and $u_i^\intercal u_i = 1$. $\square$