https://satarora.com/w24-alt-tab.pdf

Why is my Machine Learning Model so Bad?

Alt-Tab W24: Sat Arora

A bit about me…

- Name: Sat Arora
- Program: 3B Computer Science
- Likes: Programming and Algorithms
- ~~On the lookout for F24 internships :D~~

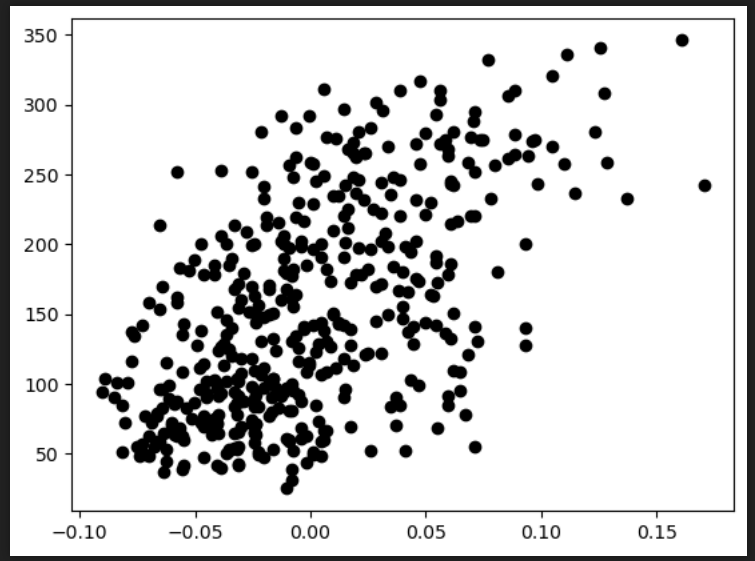
Wtf is machine learning?

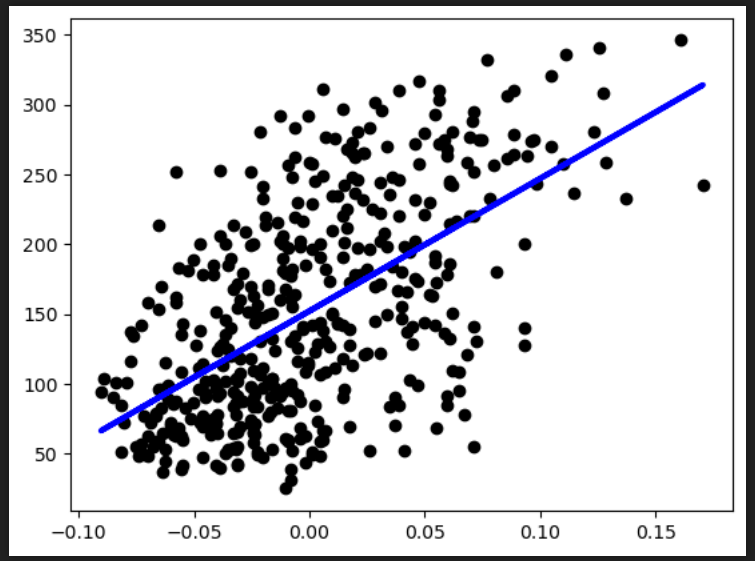
- "A branch of AI that focuses on imitating the way humans learn, gradually improving its accuracy" according to IBM: https://www.ibm.com/topics/machine-learning
- The current state of ML? Well, that one's subjective. Lots of $$$ into research though
- We will analyze a common problem: **given data, find a function that best fits it**

Why is this problem important?

- Recall our problem: **given data, find a function that best fits it**
- This type of problem is known as a **regression** problem
- It allows you to analyze the relationship between two or more variates of interest: if you get a good model, you can predict things just by plugging it in!

Regression example





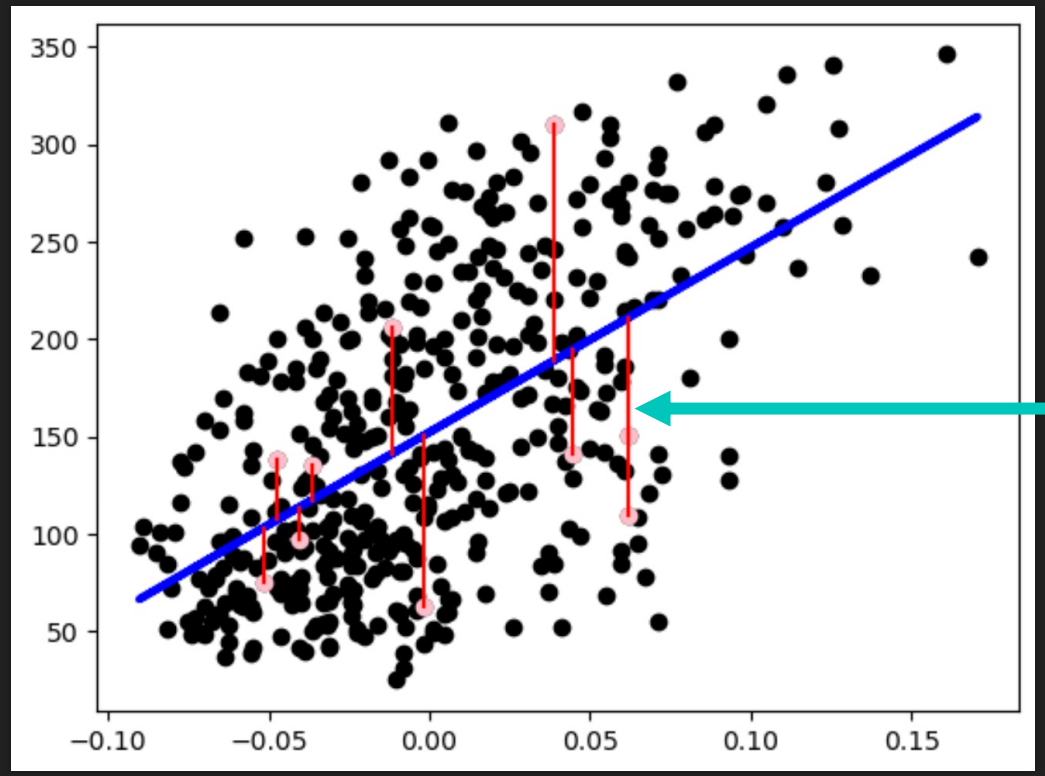
What actually happened just now?

- With a **linear model**, the process of fitting is called **linear regression**
- Using the dataset points $\mathrm{P} = (x_i, y_i), i = 1, \ldots, n$, the objective of linear regression is to find a function f^* such that it gives the minimum value of

$$\min_f \sum_i (f(x_i) - y_i)^2$$

  which is the **sum of squares of residuals** (aka sum of squares of differences)
- There are other types of regression – but this is by far the most common and is implemented this way in libraries if you don't specify anything

Visualizing residuals



A **residual** is the "error" or distance between the predicted value and the actual value.

Notice that our line is not perfectly accurate!

- A perfect regression model would have loss 0 – aka every point is on the function you create
- Is that always possible?
- **No! Thus there may always be some error!**
  - Can you think of a general idea of when this occurs? We will revisit this.
- Recall our original goal: Find f^* that is the exact value of $\min_f \sum_{i=1}^{n} (f(x_i) - y_i)^2$, **and this process without specifying its "linear" is just "regression"**
- **We will transform this** into a common "loss" function called **Mean Squared Error (MSE):**

$$f^* = \min_f \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2$$

```
    print("The mean squared error is: ", mean_squared_error(diabetesY, regr.predict(diabetesX)))
[33] ✓ 0.0s
...  The mean squared error is:  3890.456585461273
```

Assumption for our dataset

- **IMPORTANT ASSUMPTION:** Assume our dataset values come from some distribution D.

Now to learn more about errors in our predictions!

- Recall again f^* is our regression function. We show this is optimal for MSE!
- First note that as $f^* = \min_f \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2$, thus:
  - f^* is the minimum value of the expected squared error for ALL possible functions f
  - This is also the definition of **conditional expectation**, which for MSE is $E\left[\left(Y - f(X)\right)^2 \middle| X = x\right]$
  - Thus, we have that the regression function f^* satisfies $f^*(x) = E[Y|X = x]$ for any input x
- Thus, for some arbitrary function f that we may learn, we have that

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2 = \frac{1}{n}\sum_{i=1}^{n}([f(x_i) - f^*(x)] + [f^*(x_i) - y_i])^2$$

$$= \frac{1}{n}\left(\sum_{i=1}^{n}\left(f(x_i) - f^*(x_i)\right)^2 + \sum_{i=1}^{n}(f^*(x_i) - y_i)^2 + 2\sum_{i=1}^{n}(f(x_i) - f^*(x_i))(f^*(x_i) - y_i)\right)$$

- We show the third term is 0 ☺

Showing the product term is 0

- We see that

$$\frac{1}{n}\sum_{i=1}^{n}(f(x_i)-f^*(x_i))(f^*(x_i)-y_i) = E\left[\left(f(x)-f^*(x)\right)(f^*(x)-y)\right]$$

  where E is indicating that we are taking the expectation over the dataset.

- However, we know that $f^*(x) = E[Y|X=x]$, thus this expectation can be written as

$$E_X E_{Y|X}\left[\left(f(x)-f^*(x)\right)(f^*(x)-y)\right] = E_X\left[\left(f(x)-f^*(x)\right)\left(f^*(x)-E_{Y|X}(y)\right)\right]$$
$$= E_X\left[\left(f(x)-f^*(x)\right)(f^*(x)-f^*(x))\right] = 0$$

- Thus, we get

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i)-f^*(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i)-y_i)^2$$

- The second term here indicates the loss from our "most" ideal function – the regression function!

What did we learn so far?

- Thus, the **ideal** function is always the function learned from **regression**, but there is always some error.
- Some sources of error?
  - Points that are too close in x axis but are far apart in y axis
  - Or even points that have the same x value but different y values!
  - Thus this loss is **uncontrollable** when learning a function
- What can we do?
  - Simply get as close to the regression function as possible :sunglasses:
  - Smh this don't work like Discord

We are so back

- In the formula $MSE = \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - f^*(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$, **the first term is minimizable!**
- Let $\bar{f}(x)$ be the expectation of our learned $f(x)$ if we chose **different datasets**. In other words, of all possible datasets **from the same underlying distribution (the true distribution, which we obviously don't know)**
  - Thus, $\bar{f}(x) = E_D[f(x)]$ where f is the function we learn.
- Then, we expand the first term as

$$\frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - f^*(x_i)\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\left[f(x_i) - \bar{f}(x_i)\right] + \left[\bar{f}(x_i) - f^*(x_i)\right]\right)^2$$

$$= \frac{1}{n}\left(\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)\right]\right)^2 + 2\sum_{i=1}^{n}(f(x_i) - \bar{f}(x_i))(\bar{f}(x_i) - f^*(x_i))\Bigg)$$

- Can we cancel out the third term… again?

I swear this is the last math-y proof…

- Notice that

$$\frac{1}{n}\left(\sum_{i=1}^{n}(f(x_i)-\bar{f}(x_i))(\bar{f}(x_i)-f^*(x_i))\right)$$

  can be written as $E\left[(f(x)-\bar{f}(x))(\bar{f}(x)-f^*(x))\right]$ where we once again take the expectation over our dataset.

- However, recalling $\bar{f}(x)=E_D[f(x)]$, we have the expectation is equivalent to

$$E_{D,X}\left[(f(x)-\bar{f}(x))(\bar{f}(x)-f^*(x))\right]=E_X\left[E_D\left[\left(f(x)-\bar{f}(x)\right)\left(\bar{f}(x)-f^*(x)\right)\right]\right]$$

$$=E_X\left[\left(E_D[f(x)]-\bar{f}(x)\right)\left(\bar{f}(x)-f^*(x)\right)\right]=E_X\left[\left(\bar{f}(x)-\bar{f}(x)\right)\left(\bar{f}(x)-f^*(x)\right)\right]=0$$

Sooo… what did we get?

- We end up with

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)\right]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

  as our final formula!
- Ok, let's rearrange these values to write these nicer later

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)\right]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

- **Recap:**
  - f: Function that we learn
  - $\bar{f}$: Expectation of function output if we get functions from many **samples** of the underlying distribution **behind our sample itself**
  - f^*: The function that is learned when doing regression (linear regression if just a linear model) – aka the function that depicts $E[Y|X = x]$ for all x

Looking at the Three Terms again:

- Function:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

- The first term:

$$\frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)]\right)^2$$

represents the model's **(square) bias**: this comes from the assumptions in the *learning algorithm* – i.e., the simpler you make your model, the more variance you will have!

Looking at the Three Terms again:

- Function:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

- The second term:

$$\frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2$$

represents the model's **variance**: how far away is **our** model compared to the expected model if we had repeated sampling of the dataset's sample distribution? It represents "how unlucky" you got with your dataset compared to the distribution the dataset is taken from

Looking at the Three Terms again:

- Function:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

- The third term:

$$\frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$$

represents the model's **noise**: things that you can't control when you try to create a function. Remember, we wanted the dataset to come from some distribution – but this suggests that there is noise between that distribution and the points themselves.

How does model complexity affect the three terms?

- Bias: The larger the model, the less bias there is. This is because you can be "potentially" closer to the regression model.
- Variance: The larger the model, the more variance you *can* have because there is more room for deviation from the expected model.
- Noise: I think I've said it enough times – you tell me, what is it?

What did we just do?

- $MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\bar{f}(x_i) - f^*(x_i)]\right)^2 + \frac{1}{n}\sum_{i=1}^{n}\left(f(x_i) - \bar{f}(x_i)\right)^2 + \frac{1}{n}\sum_{i=1}^{n}(f^*(x_i) - y_i)^2$
- This is known as the **bias-variance decomposition.**
- Models will always have a sense of inaccuracy, but the bias and variance can be toyed around with.
- Try it out for yourself!

Thanks for watching!

Feel free to take another look at this at https://satarora.com/w24-alt-tab.pdf