Partial Derivatives / Differentiation

04 April 2025 16:57

Background:

The sequence:

1. We started by classifying big fish & small fish in a dataset which had two features of these fishes & found that if we plot their "features" on a graph then what we need is to find a straight line that separates these dots.
2. Then we found many such lines & conclude that the line that has highest total distances from these points can act as best separator. So, we created a "Gain function" that represents total distances of points.
3. We then plotted this "Gain function" (or "Loss function" by changing sign of the gain function) and found that we are interested in finding the point on this graph which is either **maxima** (in case of Gain function) or **minima** (in case of Loss function) to get the best boundary (straight line)
4. Then we came to know that if we draw tangents at different points on that function curve then the differentiation of the function gives slopes of those tangents at those points. And we are looking for the points (minima or maxima) at which the slope of the tangent is zero (hence the differentiation is 0). For this, we studied differentiation.
5. But now what we observe is that the Gain or Loss function is not a function of only one variable (unknown) but it is having multiple variables. So now we want to learn how to differentiate a function that has multiple variable. These variables are nothing but components of w vector.

So, let's learn how to differentiate a function with more than one variable! **(Also known as partial differentiation)**

$$f(x,y) = 2x^3 - 4y^2$$

$$\frac{d}{dx} f(x,y) = \frac{d}{dx}(2x^3 - 4y^2)$$

$$= 6x^2 - 0$$

$$\frac{d}{dy} f(x,y) = \frac{d}{dy}(2x^3 - 4y^2)$$

$$= 0 - 8y$$

$$\boxed{\frac{\partial}{\partial x} f(x,y) = 6x^2}$$

$$\boxed{\frac{\partial}{\partial y} f(x,y) = -8y} \longrightarrow$$

Partial Diff

$$\text{Complete differentiation of } f(x,y) = \begin{bmatrix} 6x^2 \\ -8y \end{bmatrix}$$

The tangents (especially in higher dimensions) are also called **Gradient**s. Hence, we want to find the **Gradient** of our Loss/Gain function. Our Loss Function is given by:

$$L(\vec{w}, w_0) = - \sum_{i=1}^{n} \frac{\vec{w}^T \cdot x_i + w_0}{\|\vec{w}\|} \cdot y_i$$

$$\text{where } \vec{w} = [w_1 \quad w_2 \quad w_3 \quad \text{----} \quad w_d]$$

Therefore, we need to differentiate the Loss Function with respect to each member of $\vec{\omega}$ and with respect to ω_0 as well.

suppose $\frac{\partial L}{\partial \omega_0}$ represents partial differentiation of L w.r.t. ω_0,

$\frac{\partial L}{\partial \omega_1}$ " " " " " " ω_1,

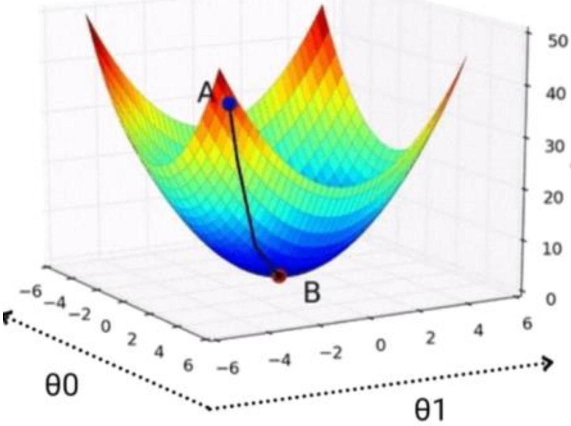
$\vdots$

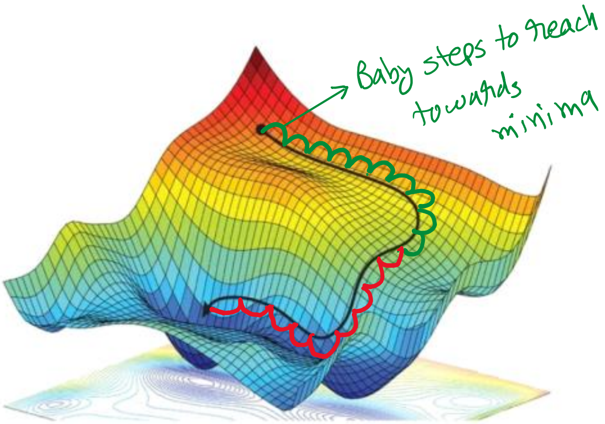
$\frac{\partial L}{\partial \omega_d}$ represent partial derivative of L w.r.t. ω_d

then the gradient (denoted by ∇L) is given by:

$$\boxed{\nabla L = \begin{bmatrix} \partial L / \partial \omega_0 \\ \partial L / \partial \omega_1 \\ \vdots \\ \partial L / \partial \omega_d \end{bmatrix}}$$

Examples of Loss functions in 3 dimensions:




Baby steps to reach towards minima

How will we take these baby steps? Means, what will be the formula to move to the next hyperplane from the current one?

$$\omega_{n+1} = \omega_n - \text{"baby step value"} \cdot \text{gradient}$$

→

These "baby steps" is called "Learning Rate" in ML and denoted by h (eta). Hence the formula to find next values of w becomes:

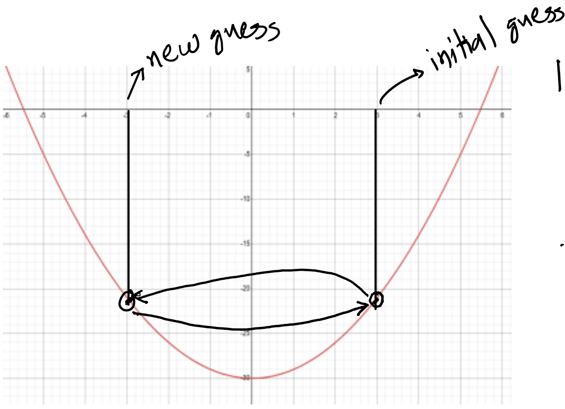
$$\boxed{\vec{w}_{i+1} = w_i - \eta \cdot \nabla L}$$

What should be the size of these "Learning Rate"? What happens if it is very large? What happens if they are too small? Let's visualize this in 2D to avoid complication.

Let $f(x) = x^2 - 30$ be the Loss Function

$\therefore \nabla L = 2x$ & let's take $\eta = 1$

$\therefore w_{i+1} = w_i - \eta \nabla L$ equation will become : $\boxed{x_{i+1} = x_i - 2x_i}$



If we take initial guess $x_i = 3$

$x_{i+1} = x_i - 2(x_i)$

$\therefore x_{i+1} = 3 - 2(3) \Rightarrow \boxed{x_{i+1} = -3}$

Taking -3 as new x_i,

$x_{i+1} = (-3) - 2(-3) \Rightarrow \boxed{x_{i+1} = +3}$

This proved to be a very big value of learning rate & we could also understand what can possibly happen if we take a big value of learning rate.

Case - 2: What if the learning rate is too small? Let's take h = 0.01

x_{i+1} = xi - η ∇L
$x_{i+1} = x_i$ - 0.01*2x
$x_{i+1} = x_i$ - 0.02x

Let the initial guess x_i = 3
x_{i+1} = 3 - 0.02*3 = 2.94
x_{i+1} = 2.94 - 0.02*2.94 = 2.8812
x_{i+1} = 2.8812 - 0.02*2.8812 = 2.8236
This will take very long to reach to the actual minima.

Therefore, to decide a correct value of Learning Rate (η) is a very crucial thing. And, it is in our hands. **Any parameter that we can change manually and that affects our algorithm is known as a Hyperparameter and hence, η is one of such Hyperparameters.**

The example we have been discussing since the beginning of ML i.e. fish sorting problem is an example of **classification problems**. Means, we want "classify" the fish as big or small. To understand the problem-solving approach for this type of problems, we need to first understand what are **Regression Problems** & the approach to solve them.

Hence for a while, let's put the fish-sorting problem aside & discuss a new type of problem that is predicting price of a house.

Regression Problems:

Unlike to classification problems where we have fixed number of outcomes (categories) to classify the incoming item, in regression problems the number of outcomes is infinite.

Examples of classification problems:

1. Fish sorting problem (categories: big fish/small fish)
2. Spam filter that determines whether the incoming mail is spam or not spam
3. Fraudulent transaction detection (categories: fraud/genuine)

Examples of regression problems:

1. House price/car price prediction
2. Weather forecasting
3. Cyclone intensity/route prediction
4. Stock price prediction

Linear Regression:

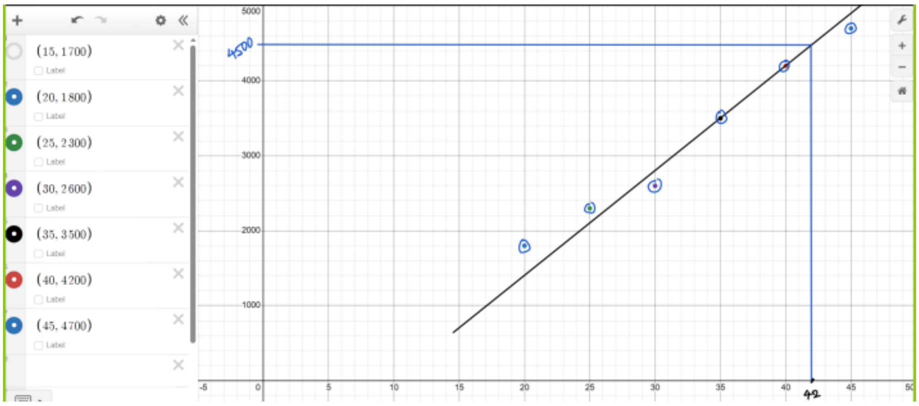
Example: I have observed that the electricity bill of my home is directly proportional to the average temperature of the city. Some of the observed data is as under:

Avg.Temp	Ele. Bill
15	1500
20	2000
25	2500
30	3000
35	3500
40	4000
45	4500

From the above data, what would be my last month's electricity bill if the avg. temperature of the city was 42 last month?
Ans: 4200

This is called Linear Regression at the easiest way I can put.
Let's consider a slightly real-world modification to the above data:

Avg.Temp.	Ele. Bill
15	1700
20	1800
25	2300
30	2600
35	3500
40	4200
45	4700



○ (15, 1700)
□ Label

● (20, 1800)
□ Label

● (25, 2300)
□ Label

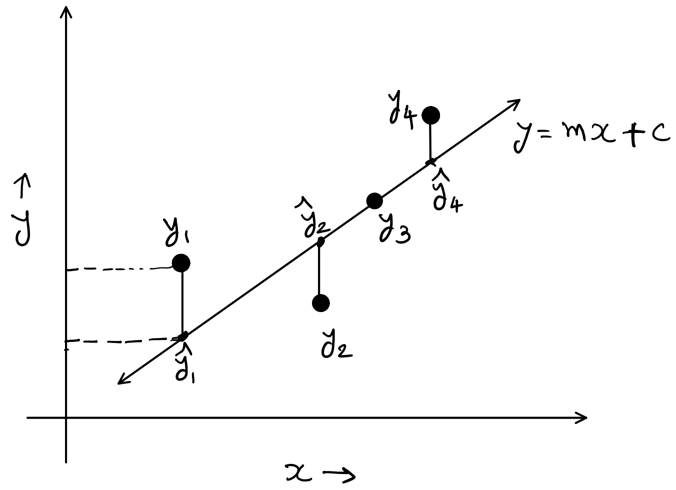
● (30, 2600)
□ Label

● (35, 3500)
□ Label

● (40, 4200)
□ Label

● (45, 4700)
□ Label

Now suppose graph of a dataset is as under:



Here "x" is the feature (like avg. temperature of the city) an "y" is our target variable (like electricity bill).

And suppose the "best" line we could get is given by:
y = mx + c

Hence we will predict our target variable by putting the given value of "x" into this equation.
Let's call our predicted value of y as $\hat{y}_i$
And the actual value of y as y_i

∴ The error in $y_1 = y_1 - \hat{y}_1 > 0$. Error in $y_2 = y_2 - \hat{y}_2 < 0$

This way, the +ve & -ve errors will cancel out each other and even if our is far from every datapoints but still the total error become 0. So we need such an equation for error that will always return +ve error. There are two ways to do this:

① Taking absolute of error (modulus)

② " square of error

We will choose 2nd option because modulus is not differentiable at 0.

∴ First we will square the errors & then to calculate total error we will take sum of these squares.

$$\therefore \text{Total errors} = (y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + \dots + (y_n - \hat{y}_n)^2$$

$$= \sum (y_i - \hat{y}_i)^2$$

$$L = \sum_{i=1}^{n} (y_i - (m \cdot x_i + c))^2 \rightarrow \text{This is our Loss Function}$$

If we want to find average total error

$$L = \frac{1}{n} \sum (y_i - (mx_i + c))^2$$

Now, we differentiate the Loss Function to find the gradient ∇L. In this equation, we want to find 'm' & 'c'. In other words, they are our unknowns. ∴ We will differentiate the Loss function partially w.r.t. them one by one.

$$\frac{\partial L}{\partial m} = \frac{1}{n} \frac{d}{dm} \sum (y_i - (mx_i + c))^2$$

$$= \frac{1}{n} \sum 2(y_i - (mx_i + c)) . \frac{d}{dm} (y_i - (mx_i + c))$$

$$= \frac{2}{n} \sum (y_i - (mx_i + c)) (0 - x_i - 0)$$

$$\frac{\partial L}{\partial m} = \frac{-2}{n} \sum (y_i - (mx_i + c)) \cdot x_i$$

$$\boxed{\frac{\partial L}{\partial c} = \frac{1}{n} \frac{d}{dc} \sum (y_i - (mx_i + c))^2}$$

$$= \frac{1}{n} \sum 2(y_i - (mx_i + c)) . \frac{d}{dc} (y_i - (mx_i + c))$$

$$= \frac{2}{n} \sum (y_i - (mx_i + c)) \cdot (0 - 0 - 1)$$

$$\boxed{\frac{\partial L}{\partial c} = \frac{-2}{n} \sum (y_i - (mx_i + c))}$$