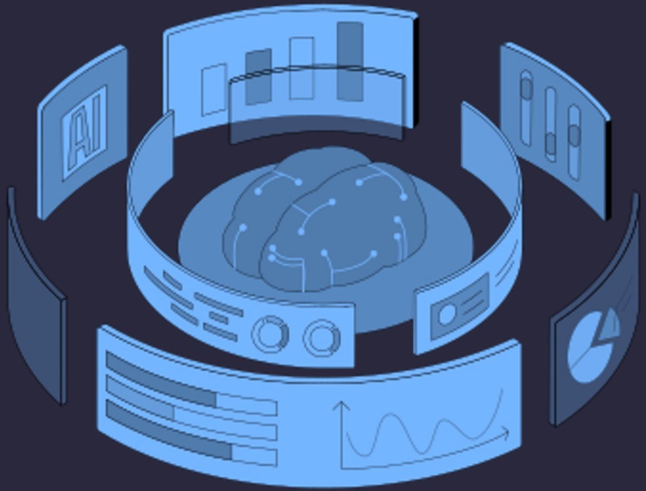
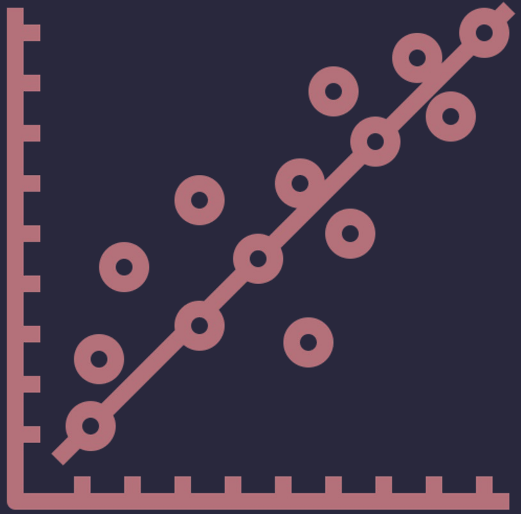
LEAP

/LINEAR REGRESSION WORKSHOP



LEAP

/INTRODUCTION

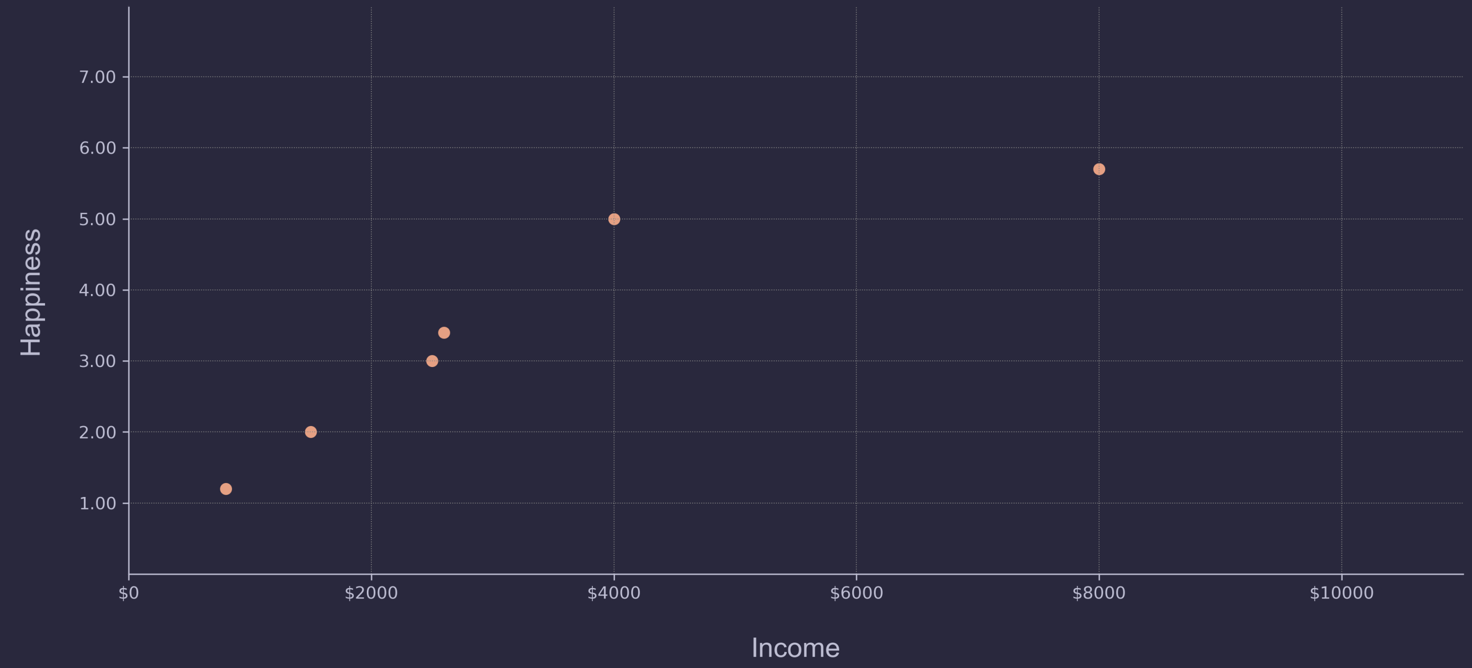


/WHAT

Linear Regression is a method used to *predict* the value of a variable based on the value of another variable.

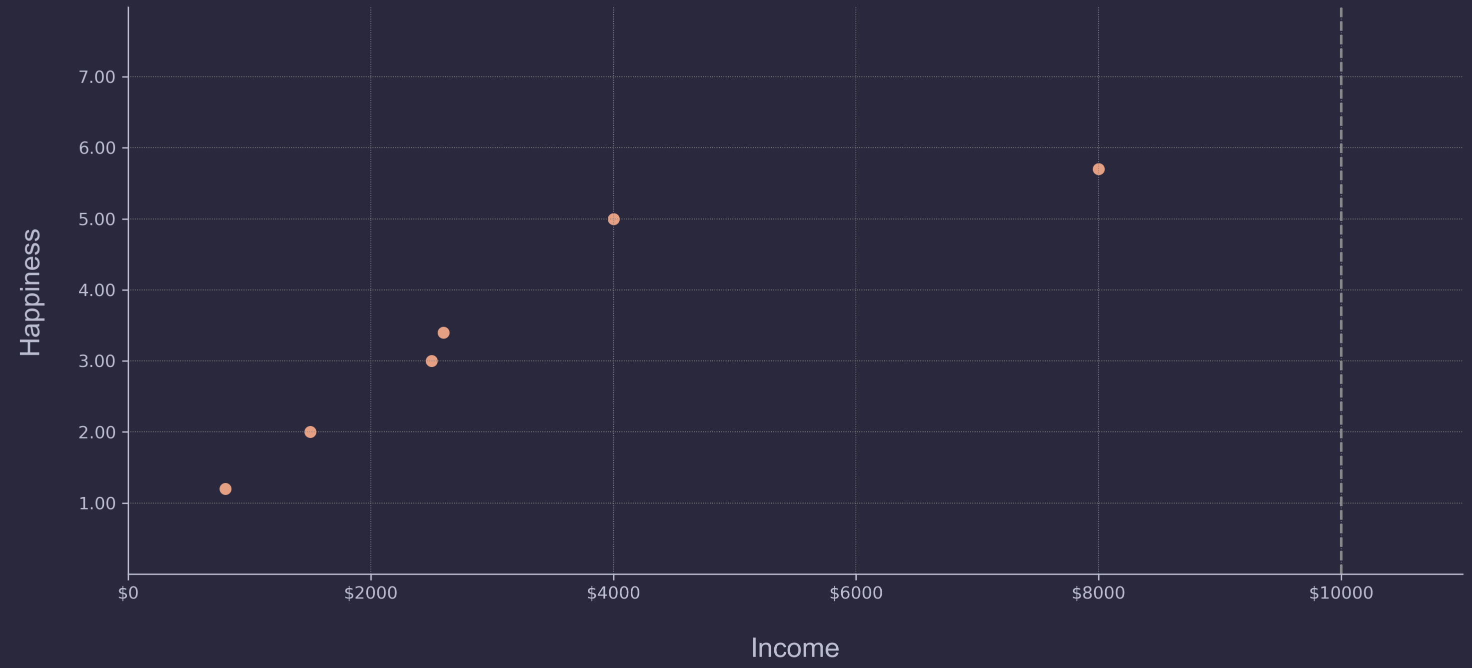
But what does this mean?

/EXAMPLE 01



7.00
6.00
5.00
4.00
3.00
2.00
1.00
Happiness
$0
$2000
$4000
$6000
$8000
$10000
Income

/EXAMPLE 01



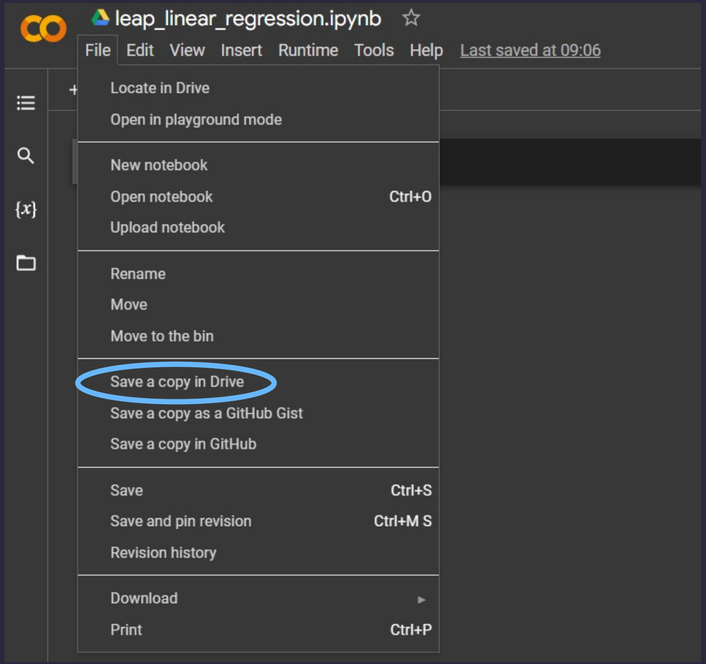
7.00
6.00
5.00
4.00
3.00
2.00
1.00
Happiness
$0
$2000
$4000
$6000
$8000
$10000
Income

/EXAMPLE 01

7.52
7.00
6.00
5.00
4.00
3.00
2.00
1.00
Happiness
$0
$2000
$4000
$6000
$8000
$10000
Income

/LET THE PREDICTIONS BEGIN!

1. Go to **leap-ai.tech**
2. Open the Colab file



LEAP

/The what?

To create a model to predict the price of your uber drive for a particular distance.

/DATA COLLECTION

Let’s predict Uber prices

/DATA COLLECTION

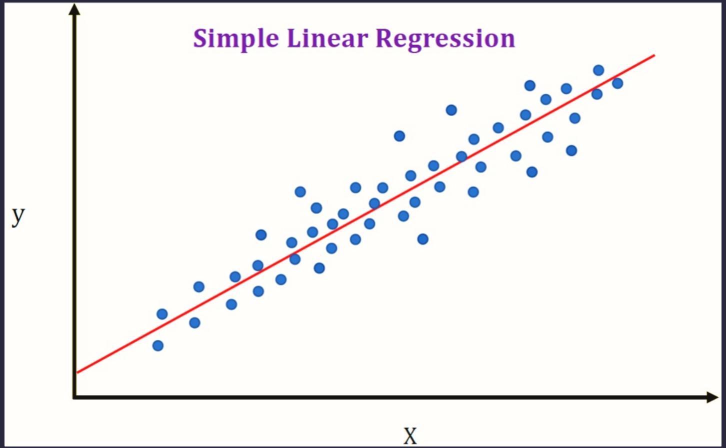
/place	/distance (km)	/price (inr)
Roadies Koffeehouz, Sector 82		
ISB Mohali		
Bestech Square Mall		
Mohali Golf Range		
Elante Mall		
Old Pal Dhaba, Sector 28		

LEAP

/DATA COLLECTION

Let's visualize the prices!

`Please open your notebooks 📒.`



/The how?

Linear Regression: we will apply *linear regression algorithms* on the data collected just a few moments ago to make predictions.

LEAP

/The idea, aim and solution

The idea behind Linear Regression is that you plot a relationship graph between *distance* and *price* and then find the line of best fit.

So, then the aim is to **plot a line**!

High school mathematics says that the equation of a line is:

/The idea, aim and solution

The idea behind Linear Regression is that you plot a relationship graph between *distance* and *price* and then find the line of best fit.

So, then the aim is to **plot a line**!

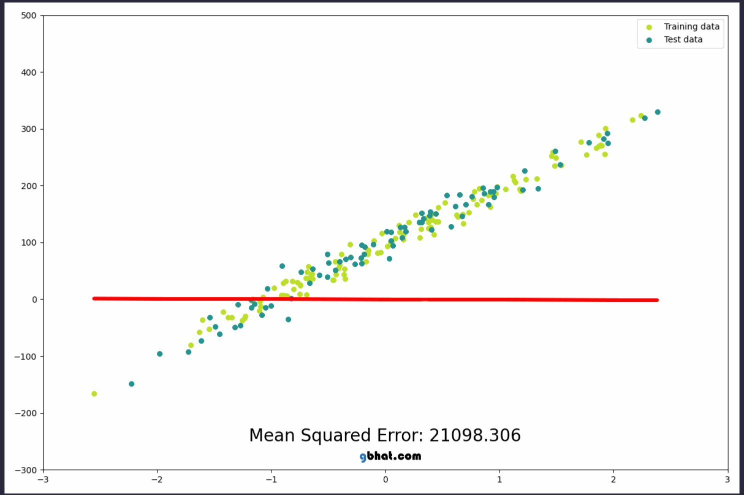
High school mathematics says that the equation of a line is:

$$y = mx + c$$

Where:

y = *price*
x = *distance*
c = *minimum price for a ride*

LEAP



But what's affecting the line?

$$y = mx + c$$

- The slope: m
- The intercept: c

in this case, c is minimum cost per ride (bias)

So, the aim is to find the right values of slope (m) and intercept (c).

But how do we evaluate?

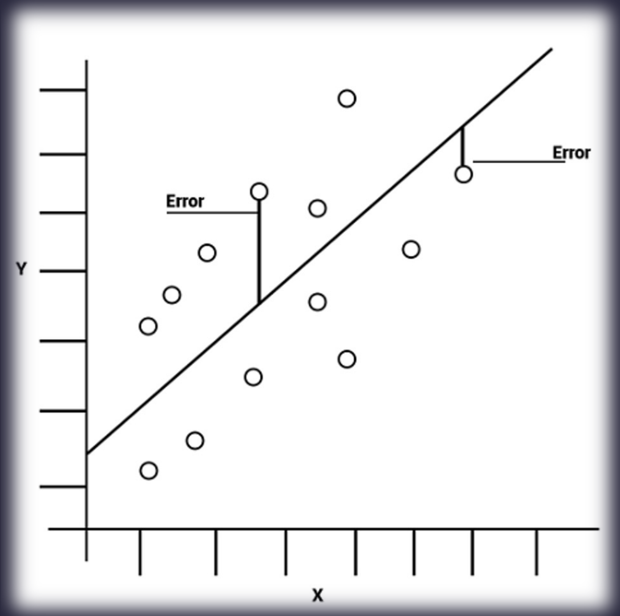
- using the ***Loss Function***

LEAP

/Loss Function- The eye of ML models

/WHAT IS LOSS?

At its core, a loss function is a measure of how **"good"** your prediction is compared to actual value.



/Loss Function- The eye of ML models

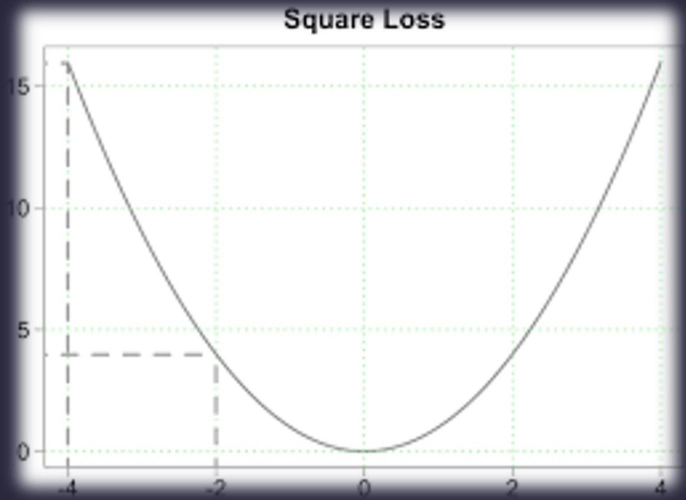
Squared loss: a popular loss function

$$(y_i - y_{predicted_i})^2$$

Where:

y_i => actual value from data

$y_{predicted_i}$ => predicted value from our model



The Loss function is Uber checking how **close** their estimated ride cost is to the actual cost for a **single trip**. If it's not very close, the loss is high, and they might need to adjust the price prediction for that one ride.

However, the cost function is a generic view that looks at all rides over time and sees if Uber's predictions are accurate for **all trips on average**. If they are incorrect often, the cost is high, and they might need to improve their prediction system.

/WHAT IS THE COST FUNCTION?

`How is my model performing?` *`Is it not working?`*

`Cost Function = Average loss of the model`

$$\text{Cost} = \frac{1}{2n}\sum_{i=1}^{n} loss_i$$

/COST FUNCTION

But, $loss_i = (y_i - y_{predicted_i})^2$

$$\text{Cost} = \frac{1}{2n}\sum_{i=1}^{n}(y_i - y_{predicted_i})^2$$

n - size of data
y_i - actual value from data
$y_{predicted_i}$ - predicted value from our model

/COST FUNCTION (contd.)

Let $\boldsymbol{m} = 2$ *and* $\boldsymbol{c} = 1$

x_i	y_i	$y_{predicted} = mx_i + c$
1	6	2 x 1 + 1 = 3
2	8	2 x 2 + 1 = 5
3	10	2 x 3 + 1 = 7
4	14	2 x 4 + 1 = 9

$$\text{Cost} = \frac{1}{4\text{X}2}\left(3^2 + 3^2 + 3^2 + 5^2\right)$$

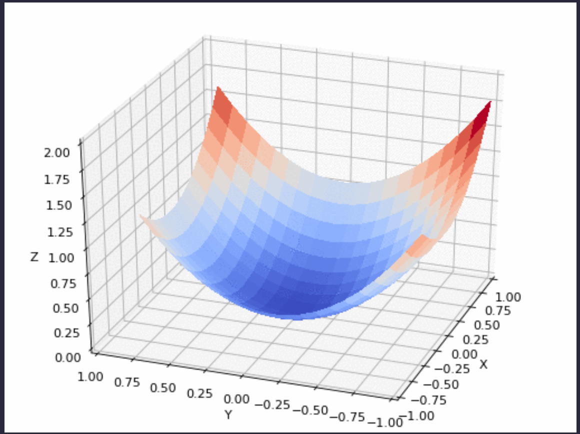
$$= \frac{1}{8}(52)$$

$$= 6.5$$

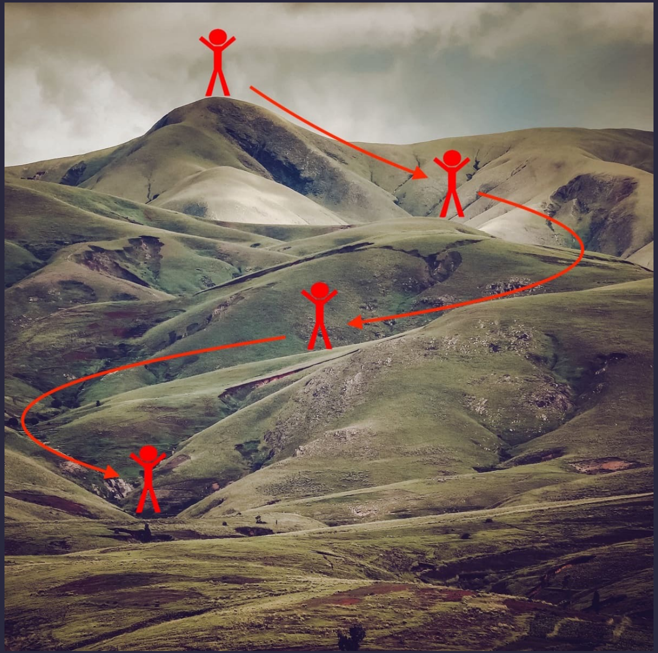
Cost will be different for different w and b!

/GRADIENT DESCENT

Optimization go *brrrr*

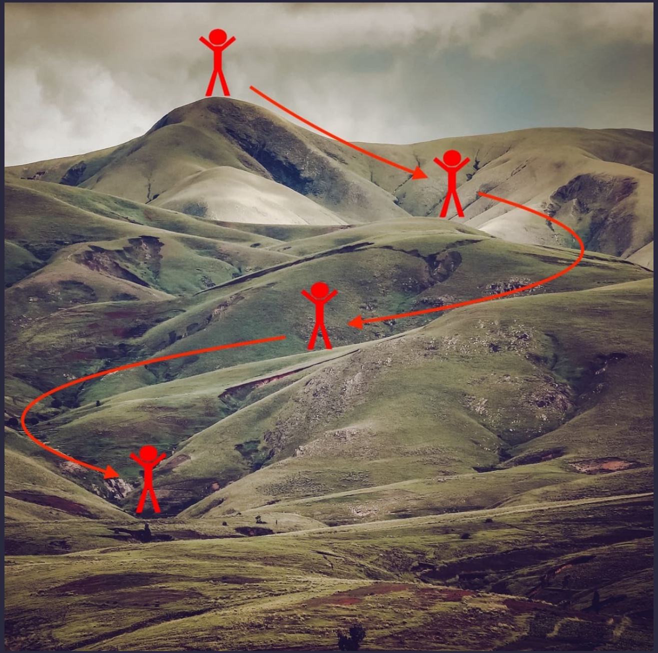


LEAP



/LETS IMAGINE!

- Imagine you are a Hiker at the top of a mountain.
- But it's night-time...
- How do you get to the bottom?



- Imagine you are a Hiker at the top of a mountain.
- But it's night-time...
- How do you get to the bottom?

- One Step at a time, that's how.
- Just see what direction appears to be *decreasing*.
- Take a step in that direction.
- Repeat

/RELEVANCE?

/POSITION

Think of the position like our weights and biases

/ALTITUDE

...the altitude as our cost function

/BOTTOM

...and the bottom as the point where our cost is the lowest.

/f(u)

Consider a function f(u)

$$\overset{\text{new position}}{u_{n+1}} = \underset{\text{current position}}{u_n} - \overbrace{\underset{\text{step size}}{\eta} \frac{df}{du}}^{\text{step}} \rightarrow \text{derivative}$$

Show visualizer

/f(u, v)

Now, what if we have a function in two variables, $f(u,v)$

$$u_{n+1} = u_n - \eta \frac{\partial f}{\partial u}$$

$$v_{n+1} = v_n - \eta \frac{\partial f}{\partial v}$$

/MINIMIZING THE COST FUNCTION

We need to find w and b that minimize our cost function $J(w, b)$

$$w_{n+1} = w_n - \eta \frac{\partial J}{\partial w}$$

$$b_{n+1} = b_n - \eta \frac{\partial J}{\partial b}$$

/CALCULATION OF THE DERIVATIVES

$$J(\boldsymbol{w}, \boldsymbol{b}) = \frac{1}{2n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{x}_i) - \boldsymbol{y}_i)^2$$

$$\frac{\partial J}{\partial w} = \frac{1}{2n}\sum_{i=1}^{n}\frac{\partial}{\partial w}(\boldsymbol{h}(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{x}_i) - \boldsymbol{y}_i)^2$$

(Taking the derivative w.r.t. w)

$$\Rightarrow \frac{\partial J}{\partial w} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{x}_i) - \boldsymbol{y}_i)\frac{\partial}{\partial w}(wx_i + b - y_i)$$

$$\Rightarrow \frac{\partial J}{\partial w} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{x}_i) - \boldsymbol{y}_i)x_i$$

/CALCULATION OF THE DERIVATIVES

$$J(\boldsymbol{w},\boldsymbol{b}) = \frac{1}{2n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w},\boldsymbol{b},\boldsymbol{x}_i) - \boldsymbol{y}_i)^2$$

$$\frac{\partial J}{\partial b} = \frac{1}{2n}\sum_{i=1}^{n}\frac{\partial}{\partial b}(\boldsymbol{h}(\boldsymbol{w},\boldsymbol{b},\boldsymbol{x}_i) - \boldsymbol{y}_i)^2$$

(Taking the derivative w.r.t. b)

$$\Rightarrow \frac{\partial J}{\partial b} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w},\boldsymbol{b},\boldsymbol{x}_i) - \boldsymbol{y}_i)\frac{\partial}{\partial b}(wx_i + b - y_i)$$

$$\Rightarrow \frac{\partial J}{\partial b} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{h}(\boldsymbol{w},\boldsymbol{b},\boldsymbol{x}_i) - \boldsymbol{y}_i)$$

/QUIZ TIME!

Go to joinmyquiz.com

Top 5 will get prizes 😋

<THANKS!>

Check **post-workshop content** in the notebook