3. Mean absolute error :-

- absolute differences between actual and predicted value is calculated.
- $$MAE = \frac{\sum_{i=0}^{n} |Y - Y'|}{n}$$
- also known as L1 loss.

MTWTFS Date:

1 - Mean error :-

* Error of each training data is calculated and then mean value of these error is derived.
* errors can be both negative and positive. So they can cancel each other out during summation giving zero mean error of the model.

2- Mean Squared error :-

* Square of difference between the actual and predicted value is calculated.
* $MSE = \frac{\sum_{i=0}^{n} (Y-Y')^2}{n}$
* also Known as L2 Loss.

So what is regression Cost function?

As we discuss in previous class, regression models deal with predicting a continuous value for example salary of an employee price of car, loan prediction So the cost function used in that type of problems is regression cost function:-

$$Error = Y - Y'$$

Y = actual input.
Y' = predicted output

So mostly used regression Cost function are:-

1- Mean error.
2- Mean squared error.
3- Mean absolute error-

on varius datasets or training rounds.

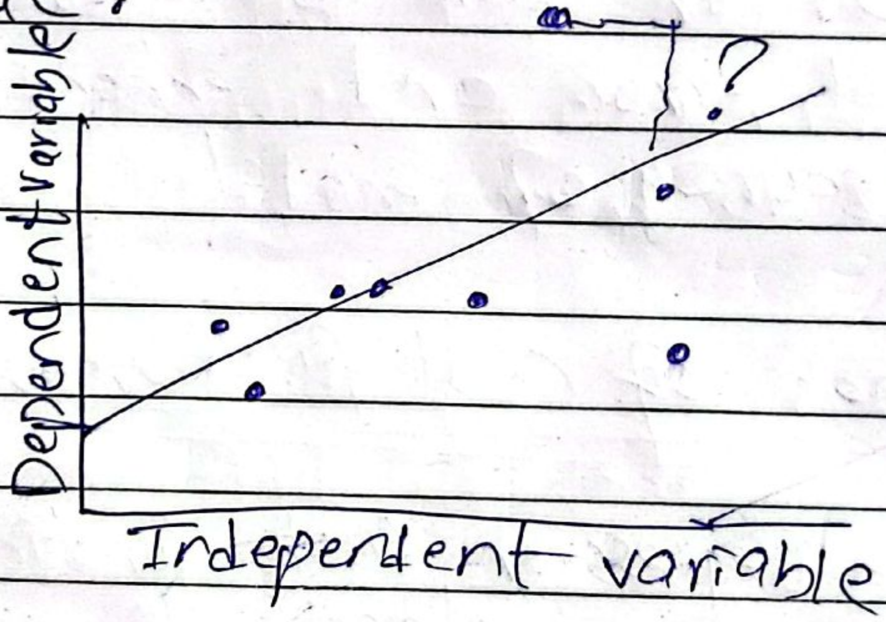
Improving guides:- objective is to reduce cost function. Through modifying the internal parameters of the model such as weights and biases, we can aim to minimize total error and enhance the accuracy of model.

Types of cost functions:

1- Regression cost function.
2- Binary classification cost function.
3- Multi class classificationcost function.

It takes both predicted outputs by the model and actual outputs and calculate how much wrong the model was in its prediction.

it outputs the higher number if predictions differs a lot from actual value.



Error calculation :- it determines the difference between the predicted outputs and the actual outputs.

Given one value :- This simplifies Comparing model's performance empty dots are cats & filled dots are dogs.

So there could be many classification. from the 3 graphs in blog the 3rd one is best because it does't misclassify any point. this is because line is almost exactly in between two groups. and not closer to any one of the groups.

So this is where cost function helps us reach the optimal solution.

So the appropriate defination of cost function :- "it is the technique of evaluating the performance of our algorith or model.

loss function:- for single training example error.

cost function :- Average of the loss function over an entire training dataset.

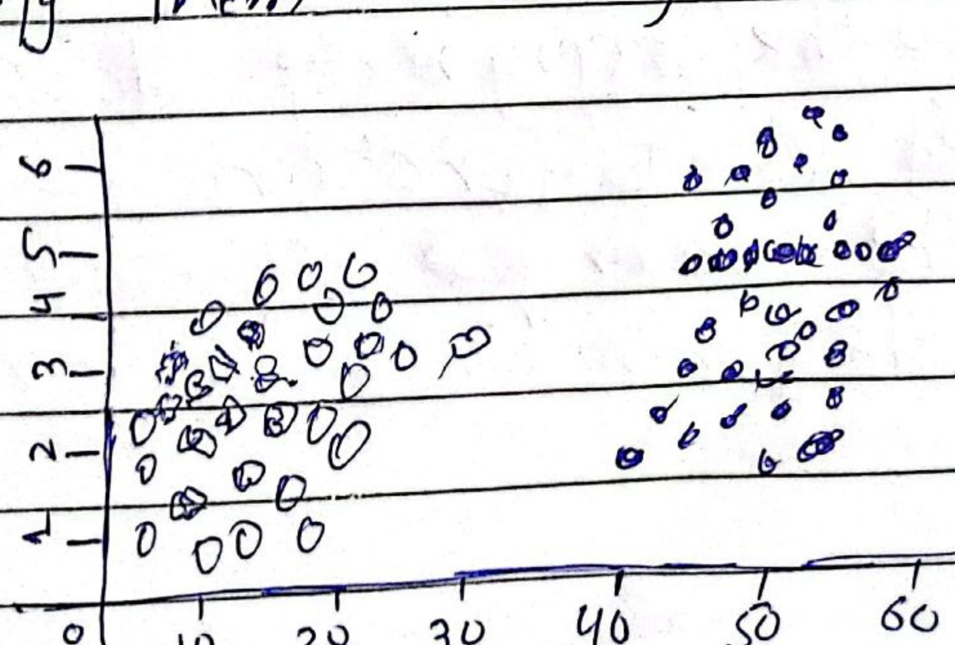
Why?

- where we wish to classify data.

for example :-

Suppose we have heights and weights of some cats & dogs.

So use these 2 features to classify them correctly :-



Cost function is No Rocket Science !

1 – Importance of cost function
2- How they help to evalute, how well a model performs
3- Different types of cost function

What ?

* Cost function also reffered as a loss function or objective function
* it quantifies the difference between predicted and actual values
* Cost function guides the model towards optimal predictions by measuring its accuracy against the training data.

step 5 :- Computing Cost over a range of value of Beta

This's ~~Sqr~~ step actually finding the MSE using a range of beta value from 0.1 to 1.5 with the increment of 0.01.

After that convert data into datafram

step 6 :- This code visualize the cost with respect to beta.

So from this gradient Descent approach helps us to find the best fit line

Step 3 :- Plotting the data :-

```
plt.scatter(data.experience, data.salary,
    color ='red', label = 'datapoints')
plt.xlim(1, 4.5)
plt.ylim (1, 7)
plt.xlabel('experience')
plt.ylabel('salary')
plt.legand()
```

step 4 :- plotting the varius lines using β and b

Take β=0.1 and b=1.1
MSE = 2.69665.
Take β =1.5 and b=1.1
MSE = 6.4029166666666
Take β=0.8 and b=1.1
MSE= 0.3366000000000023

Convert this dataset into pandas Dataframe.
So lets first import necessary libraries :-

Step 1

```
import matplotlib.pyplut as plt
 import pandas as pd
from sklearn.metrics import mean_squared
_error as mse.
```

Step2 :- Creating sample Data

```
experience =[1.2, 1.5, 1.9, 2.2, 2.4, 2.5, 2.8,
         3.1, 3.3, 3.7, 4.2, 4.4].
salary =[1.7, 2.4, 2.3, 3.1, 3.7, 4.2, 4.4,
         6.1, 5.4, 5.7, 6.4, 6.2].
 data = pd. Dataframe({
       "salary" : salary,
        "experience" : experience
  })
 data.head().
```

between X and Y in the best possible manner and once we have the slope and intercept of the line which gives the least error, we can use that line to predict Y.

Any line can be represented as slope(B) and intercept(b) So

$$Y = \beta X + b$$

what is the error of various values of β and b the How do we find the most optimal values of these parameters.

In the problem given in video there is a dataset for experience and salary and we have to first

Cost function:-

while dealing with linear regression we can have multiple lines for different vallues of slopes & intercepts. But the main question is that is which of those lines actually represents the right relationship between X and Y so we use Mean Squared Error

MSE = Cost function

So Actually mean squared error is a cost function.

So what is mean squared error. the answer is simple as that mean squared error is the sum of the squared differences between the prediction value and true value. And the output is single number representing Cost. So the Line with minimum MSE represents the relationship