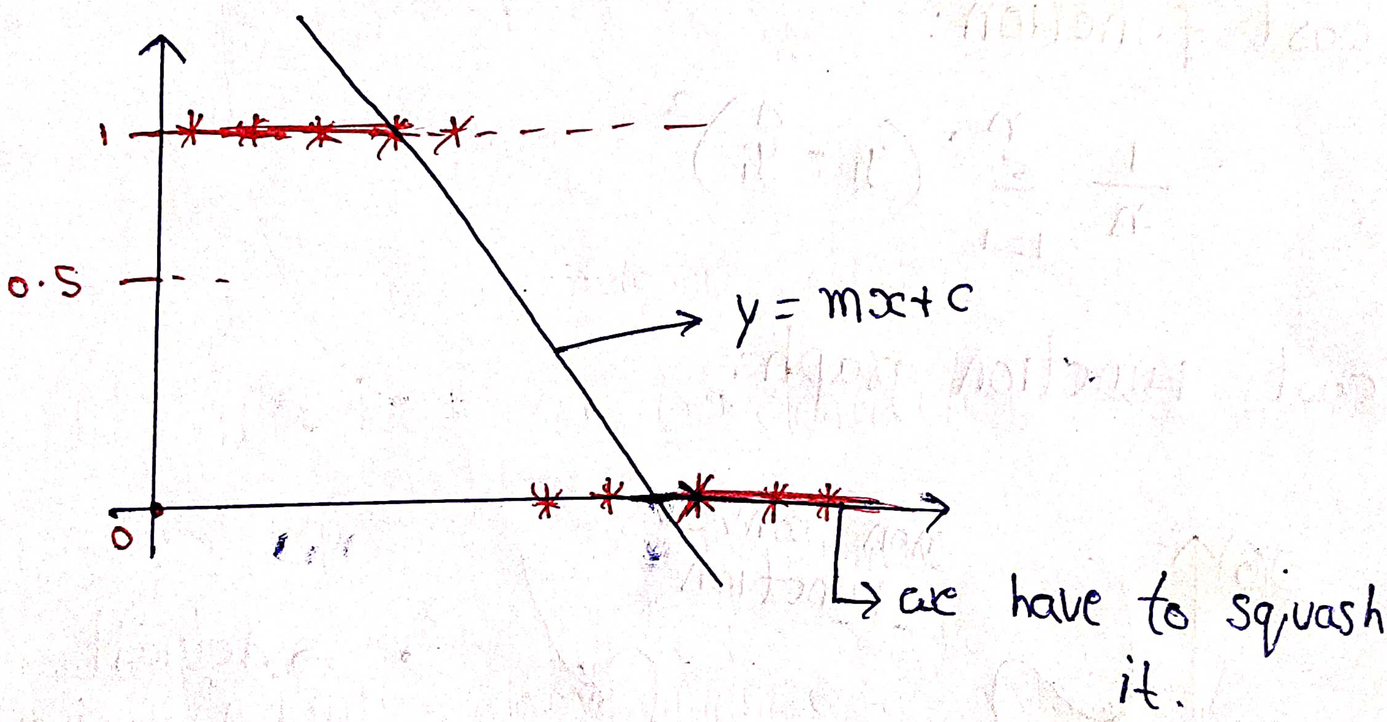
Logistic Regression:

* It is used for classification problems.

* It gives us best fit line, which will do the classification.



* we will use sigmoid activation function to convert $y = mx + c$ in between 0 and 1

sigmoid activation function $= \dfrac{1}{1+e^{-z}}$

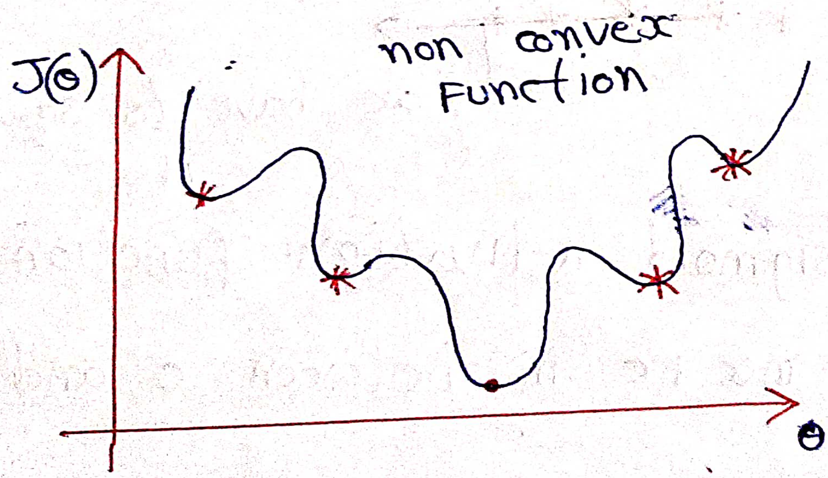
where $-z \Rightarrow y = mx + c$

So, the equation of best fit Line with sigmoid function becomes $\rightarrow \dfrac{1}{1+e^{-(mx+c)}}$

cost function:

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

cost Function graph:



'.' → Actuall global minima

'*' → Local minimas

* To avoid this convex non convex Function we will use log loss Function.

$$\Rightarrow -y_i \log \hat{y}_i - (1-y_i) \log(1-\hat{y}_i)$$

If my y_i is 0

$$\Rightarrow -0(\log \hat{y}_i) - (1-0) \log(1-\hat{y}_i)$$

$$\boxed{\Rightarrow -\log(1-\hat{y}_i)}$$

If y_i is 1

$$\Rightarrow -1 \log \hat{y}_i - (1-1) \log(1-\hat{y}_i)$$

$$\boxed{\Rightarrow -\log \hat{y}_i}$$

convergence Algorithm

$$\theta_J = \theta_J - \alpha(\text{derivative})$$

Regularizations:

L2 Regularization:

* It is used to reduce the overfitting.

Loss Function becomes

$$\Rightarrow -y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i) + \lambda(\text{slope})^2$$

L1 Regularization:

* It is used for feature selection.

Loss Function becomes

$$\Rightarrow -y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i) \cancel{+\lambda(\text{slope})^2} + \lambda|\text{slope}|$$

Elastic Net:

* It is the combination of L_2 and L_1.

Loss Function becomes

$$\Rightarrow -y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i) + \lambda(\text{slope})^2 + \lambda|\text{slope}|$$

Evaluation metrics:

* classification Report:

* It shows us the releationship of numbers between Actual and predicted values.

		Actual 1	Actual 0
predicted	1	TP	FP
	0	FN	TN

→ The values in this diagnol are the Right predictions.

* Accuracy:

All the predictions of correct dived by correct + wrong predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

* precession:

* out of All the Actual values, how many were predicted correctly.

$\rightarrow \cancel{TP}$

$$\rightarrow \frac{TP}{TP+FP}$$

* Recall:

* out of all the predicted values, how many were correctly predicted

$$\rightarrow \frac{TP}{TP+FN}$$

* F1 Beta score:

$$(1+\beta^2)\frac{\text{precession} * \text{Recall}}{\text{precession} + \text{Recall}}$$

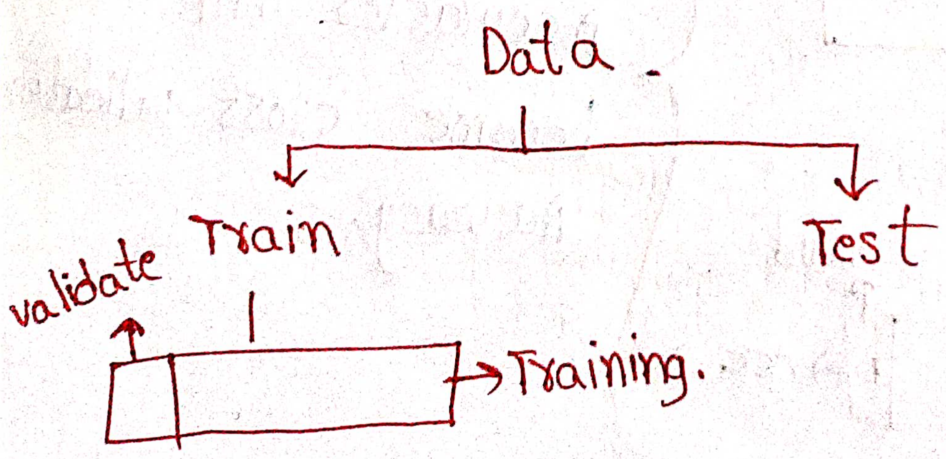
cross validation:

* After splitting the data into Training and Testing, we further split it into Different parts.

* within Training, we split the Data into Train and validate.

* By defining Train & validate ratios and How many checks we need to perform, we can get different accuracies for each of them.

* The mean of all the obtained accuracies is the accuracy of the cross validation.

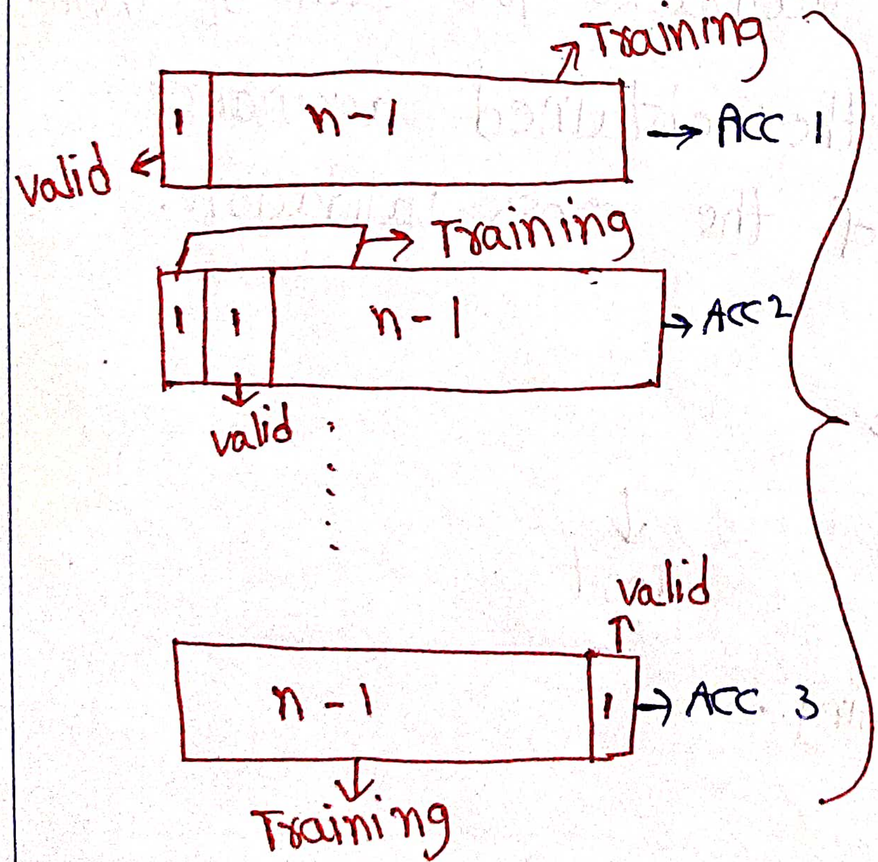


* Leave one out CV :

→ In this type of cross validation, one data of Training Dataset is given to valid and Remaining will be under Training.

→ In next iter, Next data will be in valid and remaining all under the Training.

→ This will continue till the end of All the Training Data's

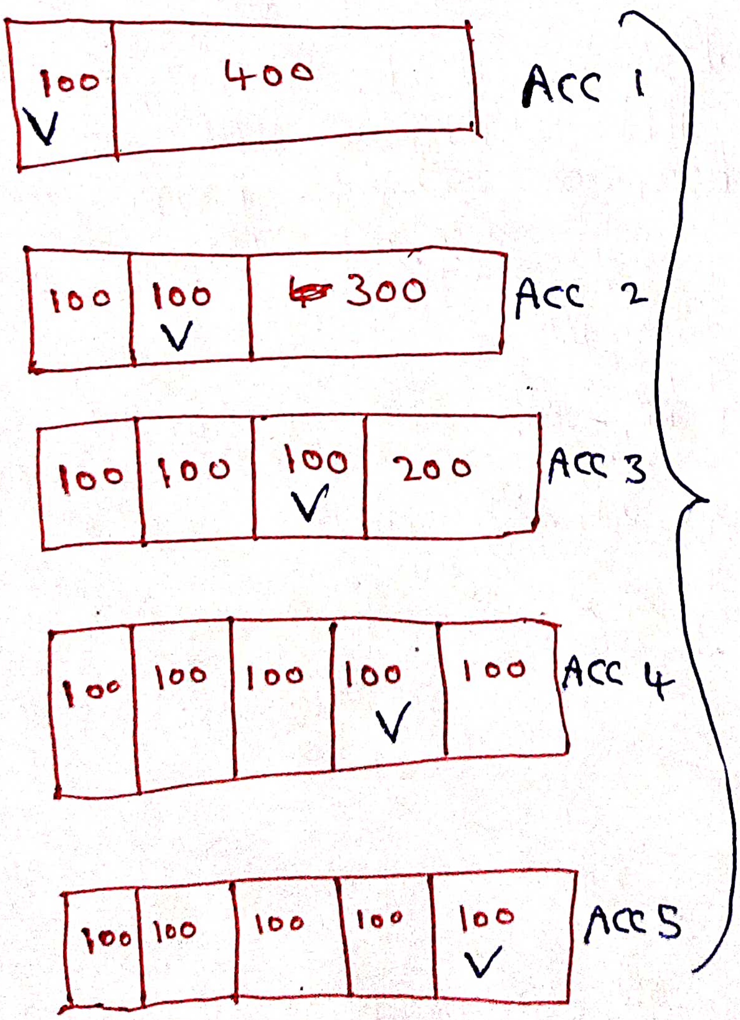


Average of All these Accuracies will become cross validation Accuracy.

K - Fold cross validation:

→ Instead of taking one to one data into the valid, we will take them in a group.

eg: let's consider we've a data of 500 if I want to divide it by 5, then



Average of All accuracies will become my cross validation Accuracy.

stratifed k-fold:

* It is used while data is imbalanced.
* In valid set It will take all classes in equal quantity