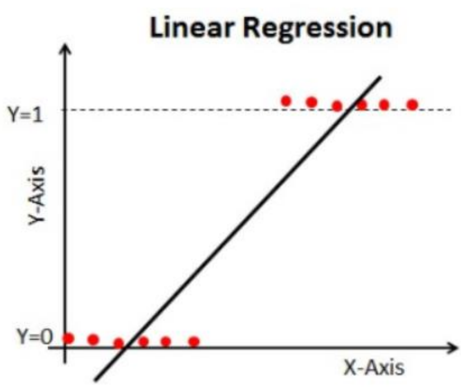
LOGISTIC REGRESSION

❖ Introduction to Logistic Regression:

- Logistic regression is a type of supervised machine learning algorithm.
- It can be used for classification and not for regression.
- It is actually a classifier but as it uses, the concept of linear regression or uses the linear model as it is called as Regression.
- And the word logistic came in picture as we use Log loss function.
- In logistic regression dependent variable always be in categorical form while independent variable can be in categorical or in continuous form.
- As it is a classifier, there are two types of classifier in Logistic Regression:
    1. **Binary Classifier**- Have two Outcomes (Binominal)
       Yes/no
       True/False
       Rain/No Rain
       Positive/negative, etc.
    2. **Multi-class Classifier**- Have more than two outcomes.
       High/medium/low>> (Ordinal)
       Apple/banana/Pineapple>>>(Multinominal)

❖ Why do we go for Logistic Regression?

If we use Linear Regression in our classification problem, we will get a best-fit line w r to our independent variable's data points like this:



Problem with a linear line is, when we extend this line, we will be having values greater than 1 and less than 0, which do not make any sense in classification problem.

It will make a model interpretation a challenge. That is where `Logistic Regression` comes in.

-We need a function to transform this straight line in such a way that values will be between 0 and 1.

Equation for straight line in linear regression is Y=mx+c.

So here comes a Sigmoid Function which convert or transform this linear line into **'S' shaped curve,** which have values in between 0 and 1 only.

LOGISTIC REGRESSION

Sigmoid= 1/ 1+ e^-(mx+c) or   1/1+e^-Y

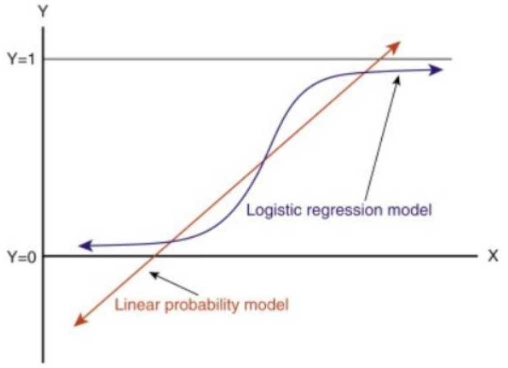
Where,

Y= independent variable

x= dependent variable

e = Euler's constant =2.178

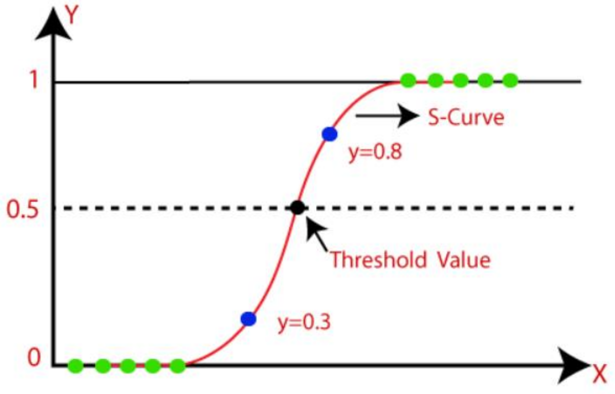
❖ **Cost Function** quantifies the error between predicted values and expected values.



Another thing that will change with this transformation is **Cost Function.**

For linear regression we use, MSE as a cost function. But for Logistic regression we use **Log Loss** as a cost function.

- Instead of a best fit line or a regression line we fit a **'S' shaped curve** in logistic regression. Which is known as **logistic /Sigmoid function.**



❖ **Logistic Function/Sigmoid Function:**

- Sigmoid function simply tries to convert independent variable into an expression of probability that ranges in between 0 and 1 w. r. to dependent variable.
- Sigmoid curve has that property that it will always remain in between 0 and 1.
- It converts numerical values to probability functions.

LOGISTIC REGRESSION

- In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.
- **Summery: The sigmoid equation fixes the range of the curve from 0-1. while the log loss decides how much curve the sigmoid graph should have.**

❖ **Assumptions of Logistic Regression**:

- The dependent variable must be categorical in nature.
- The independent variable should not have multi-collinearity. (same like linear regression) before.
- Data size should be large. (observations)
- Linearity between independent var. and logit/ log of odd function……. log(p)/1-p. after.

❖ **Log Loss Function/Cost Function:**

- Log Loss is the most important classification metric based on probabilities.
- Log Loss is the negative average of the log of corrected predicted probabilities for each instance.
- Logistic regression also uses the Gradient Descent Algorithm at the back end to find the BFL.
- Then with the help of Sigmoid Function, it transforms the line to the "S" shaped curve.
- And using m & c, we can find out which sigmoid surface we got is the best one.
- To find out the best Sigmoid surface from infinite possibilities, it uses a function called **Log Loss Function**.
- It's hard to interpret raw log-loss values, but log-loss is still a good metric for comparing models. For any given problem, **a lower log loss value means better predictions.**
- For a **good binary Classification model**, the value of **log loss should be near to 0**
- Log Loss function is also called as **Binary cross entropy.**

❖ **In short, there are three steps to find Log Loss**:

LOGISTIC REGRESSION

1. To find corrected probabilities. (see the example)

    - It is the probability that a particular observation belongs to its original class. (https://www.analyticsvidhya.com/blog/2021/03/binary-cross-entropy-log-loss-for-binary-classification/)
    - If predicted value does not fit in actual class,

2. Take a log of corrected probabilities.

3. Take the negative average of the values we get in the 2nd step.

$$-\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i))$$

Here Yi represents the actual class and log(p(yi)is the probability of that class.

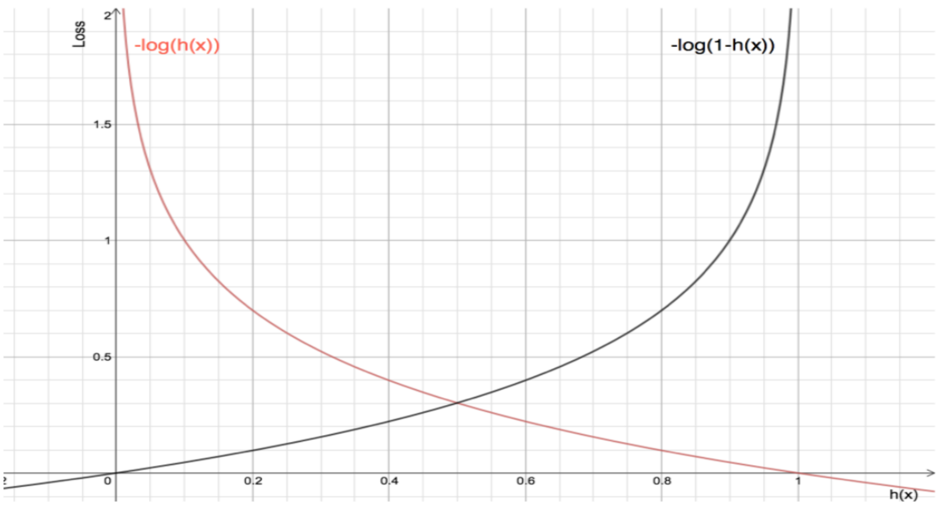
- p(yi) is the probability of 1.
- 1-p(yi) is the probability of 0

When actual class is 1 ---> Only first part of formula will exist

i.e. yi.log(p(yi))

When actual class is 0 ---> Only second part of formula will exist

i.e. (1-yi).log(1-p(yi))



LOGISTIC REGRESSION

- The Red line represents 1 class. As we can see, when the predicted probability (x-axis) is close to 1, the loss is less and when the predicted probability is close to 0, loss approaches infinity.

- The Black line represents 0 class. As we can see, when the predicted probability (x-axis) is close to 0, the loss is less and when the predicted probability is close to 1, loss approaches infinity.

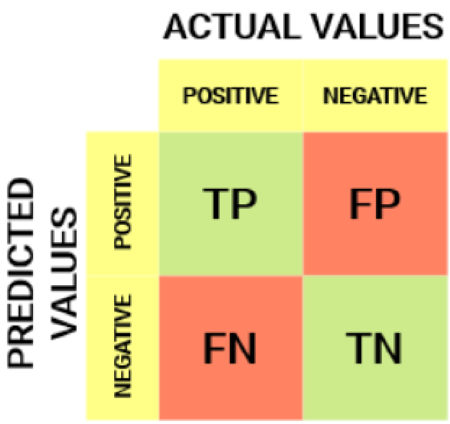
❖ Metrics Of Classification:

- Accuracy`
- Confusion matrix
- Log-loss
- Precision and Recall
- F1-Scores
- Receiver operating characteristic (ROC) curve
- Area under curve (AUC) ("curve" corresponds to the ROC curve)

❖ Accuracy

Accuracy simply measures how often the classifier makes the correct prediction. It’s the ratio between the number of correct predictions and the total number of predictions (the number of test data points).

- **TP (1,1)-** The predicted value matches the actual value. When person have cancer and report is also positive.
- **TN (0,0)-** The predicted value matches the actual value. When person don’t have cancer and reports are also negative.



FP (0,1)- This is also called Type-I error. When person don’t have cancer still report is positive

FN (1,0)- This is also called Type-II error. When person have cancer but reports are negative.

Above figure is a confusion matrix model for binary class classifier.

❖ Confusion Matrix:

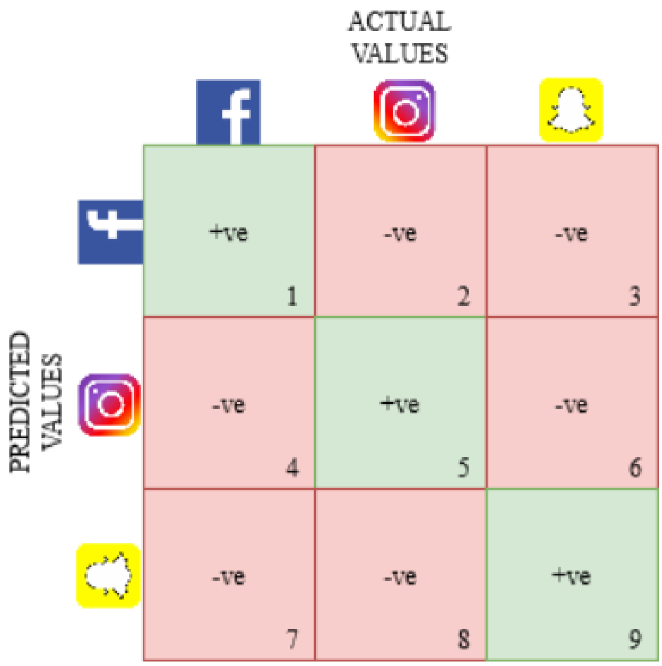
LOGISTIC REGRESSION

A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes. The matrix compares the actual target values with those predicted by the machine learning model. It gives visualization of performance of an algorithm.

Accuracy = Number of correct predictions / Total number of predictions.

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

For every good model, FN and FP should be as less as possible.



This is an image of 3x3 confusion matrix.

Recall:

Recall tells us how many of the actual positive cases we were able to predict correctly with our model. When **FN** is very important, we go for recall. Recall is a useful metric in cases where False Negative trumps False Positive.

Recall is also called as TPR or Sensitivity. What proportion of positive class got correctly classified by classifier.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall is important in medical cases.

We use recall to reduce FN.

- **Precision**:

LOGISTIC REGRESSION

Precision tells us how many of the correctly predicted cases actually turned out to be positive. Precision is a useful metric in cases where False Positive is a higher concern than False Negatives. E.g spam folder.

$$\text{Precision} = \frac{TP}{TP + FP}$$

We use precision to reduce FP.

F1-Score:

In practice, when we try to increase the precision of our model, the recall goes down, and vice-versa. The F1-score captures both the trends in a single value:

It is defined as the harmonic mean of the model's precision and recall.

$$\text{F1 Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

When both FN and FP are important in model or both precision or recall are imp in model, F1-score com in picture.

F-Beta Score:

B = 0.5 >> FP is imp >> Precision >> f0.5-score

B = 2 >> FN is imp >> Recall >> f2-score

B = 1 >> FP & FP >> f1-score

- **ROC/AUC Curve:**

  ROC- Receiver Operating Characteristics

  AUC- Area Under Curve

  For ROC, the terms TPR and FPR are in consideration i.e True Positive Rate, False Positive Rate.

  TPR/Recall/Sensitivity - What proportion positive class got correctly classified by classifier. **(TP/TP+FN)**

  A higher TPR and a lower FNR

LOGISTIC REGRESSION

FNR- What proportion the positive values got incorrectly classified by the classifier. **(FN/TP+FN)**

FPR/ (1- Specificity)-What proportion the negative values are incorrectly classified. **(FP/FP+TN)**

TNR/Specificity- What proportion the negative values are correctly classified. **(TN/TN+FP)**

A higher TNR and a lower FPR

- **AUC/ROC** curve helps use to visualize how well our machine learning classifier is performing.
- Although it is used only for binary classification problems, but we can extend it to use for multi-class classification problems as well by OVR i.e. One Verses Rest technique.
- **clf = OneVsRestClassifier(LogisticRegression())**

❖ **Probability of Predictions**:

- A machine learning classification model can be used to predict the actual class of the data point directly or predict its probability of belonging to different classes.
- We can determine our own threshold to interpret the result of the classifier.
- This is sometimes more prudent than just building a completely new model!
- Setting different thresholds for classifying positive class for data points will change the Sensitivity and Specificity of the model. And one of these thresholds will probably give a better result than the others.
- depending on whether we want to lower the number of FNs or FPs.

LOGISTIC REGRESSION

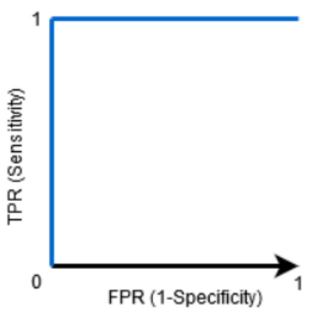
ID	Actual	Prediction Probability	>0.6	>0.7	> 0.8	Metric
1	0	0.98	1	1	1	
2	1	0.67	1	0	0	
3	1	0.58	0	0	0	
4	0	0.78	1	1	0	
5	1	0.85	1	1	1	
6	0	0.86	1	1	1	
7	0	0.79	1	1	0	
8	0	0.89	1	1	1	
9	1	0.82	1	1	1	
10	0	0.86	1	1	1	
			0.75	0.5	0.5	TPR
			1	1	0.66	FPR
			0	0	0.33	TNR
			0.25	0.5	0.5	FNR

The metrics change with the changing threshold values. We can generate different confusion matrices and compare the various metrics. But that would not be a prudent thing to do.

The AUC-ROC curve solves just that problem!

The **Receiver Operator Characteristic (ROC)** curve is an evaluation metric for binary classification problems. It is a probability curve that plots the **TPR** against **FPR** at various threshold values and essentially **separates the AUC/'signal' from the 'noise'(FN…n FP)**.

The **Area Under the Curve (AUC)** is the measure of the ability of a classifier to distinguish between classes and is used as a summary of the ROC curve.



Higher the AUC, ……….**better the performance of the model** at distinguishing between the positive and negative classes.

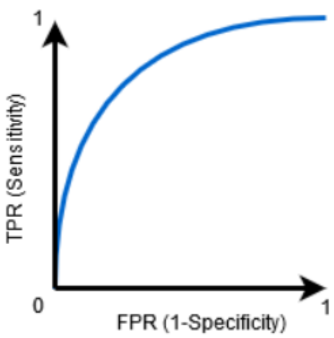
When AUC = 1, then the classifier is able to perfectly distinguish between all the Positive and the Negative class points correctly. If, however, the AUC had been 0, then the classifier would be predicting all Negatives as Positives, and all Positives as Negatives.

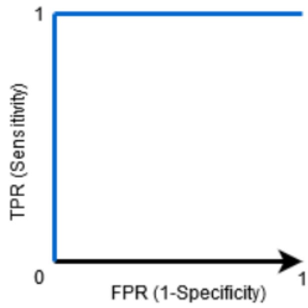
In ROC curve:

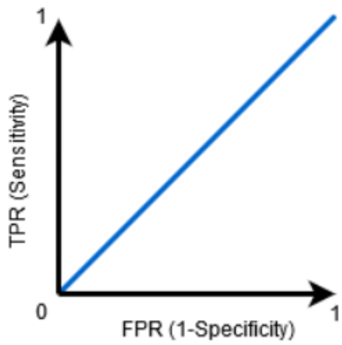
Higher X-axis value means higher number of **FP** than TN.

Higher Y-axis value mean higher number of **TP** than FN.

LOGISTIC REGRESSION







When 0.5>AUC>1 **>>When AUC=1<<** **When AUC=0.5**

When **TPR** and **TNR** are **high**, or 1 then our **model is the best** model.