* To decide which one should we look for between Sensitivity & Specificity

* If identifying +ve is more important to us, then we will select Algo that has high sensitivity

* If correctly identifying -ve is more important then we will select algo that has high specificity

Precision & Recall :=>

→ Used for Information Retrieval.

→ Google Search Engine =>

→ query fired

→ Have millions of Related Records

→ From these top 10 - 100 records are returned.

$$\text{Precision} = \frac{\text{Correctly predicted +ve}}{\text{Total +ve Predicted}} = \frac{TP}{TP + FP}$$

Range (0 — 1)

$$\text{Recall} = \frac{\text{Correctly Identified +ve}}{\text{Total Actual +ve}} = \frac{TP}{TP + FN}$$

(TPR)
(range 0 - 1)

* Ideally we want precision to be high (i.e 1) for a good classifier

(range 0 - 1)

* Ideally we want precision to be high (i.e 1) for a good classifier

$$\text{Precision} = 1 = \frac{TP}{TP + FP} = 1 \Rightarrow \text{when } \boxed{FP = 0}$$

* Ideally we want Recall to be very high (i.e 1) for a good classifier

$$\therefore \text{Recall} = 1 = \frac{TP}{TP + FN} \Rightarrow \text{when } \boxed{FN = 0}$$

So Ideally a good classifier has High Precision & Recall

But in Reality there is trade-off

* When we tweak (update) our model to increase one, then the other decreases.

Q) Explain

F1-Score

* In reality we need a metric that takes into account both precision and recall.

* F1 score is a metric that takes into account both precision & recall.

* F1 score is harmonic mean of Precision & Recall.

$$\text{F1-Score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Harmonic mean of two variables a & b

$$H = \frac{2}{\frac{1}{a} + \frac{1}{b}} = \frac{2ab}{a+b}$$

for n variables

$$H = \frac{n}{\frac{1}{n_1} + \frac{1}{n_2} + \cdots \frac{1}{n_n}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{n_i}}$$

If F1-Score = 1 ⇒ when Precision = 1, Recall = 1

* When Precision and Recall both are high then F1-Score is High

When to use F1-Score →

→ Accuracy is not a good metric to use when we have class imbalance.

Ex → Let say 99% of people visiting site are onlookers and not purchasing anything.

→ Suppose we have a model that predicts that 99% people visiting site are onlookers.

→ 1% Error is Acceptable

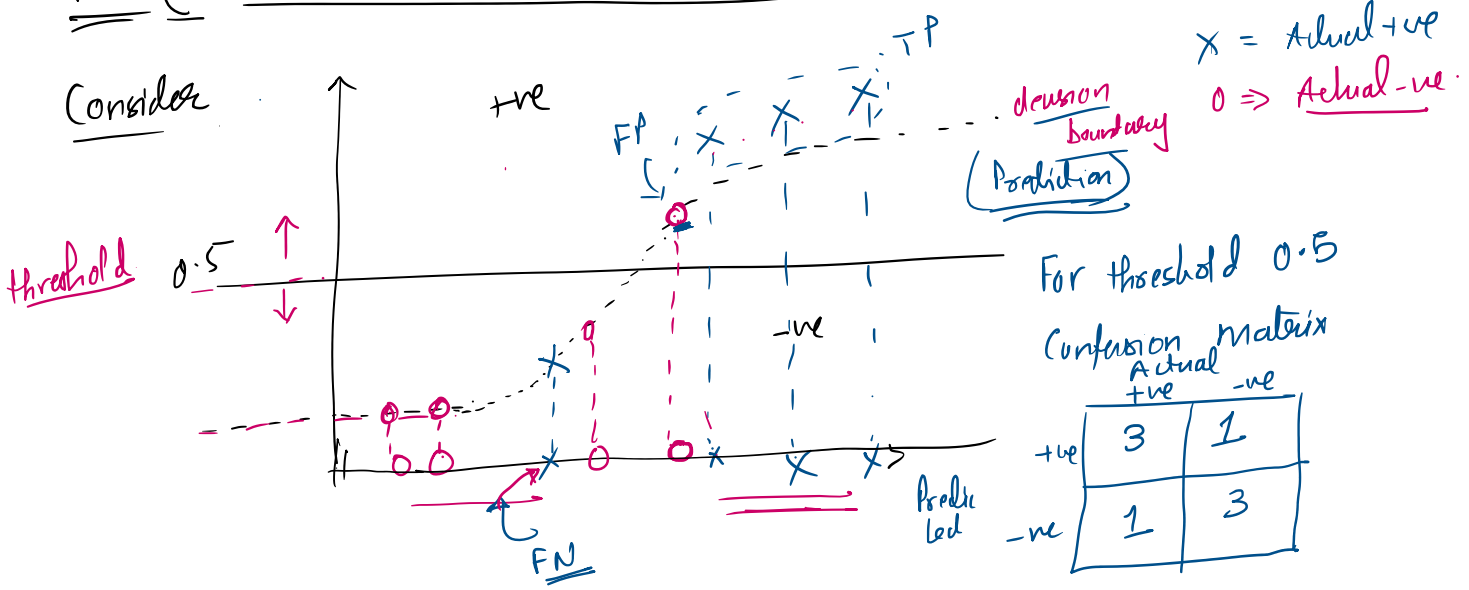
people visiting site are onlookers.

→ The model is 1% wrong, Generally 1% Error is Acceptable

→ But such model in this case is useless

→ In such case instead of accuracy, we will prefer F1-Score.

ROC [Reciever Operator characteristic]



For threshold 0.5

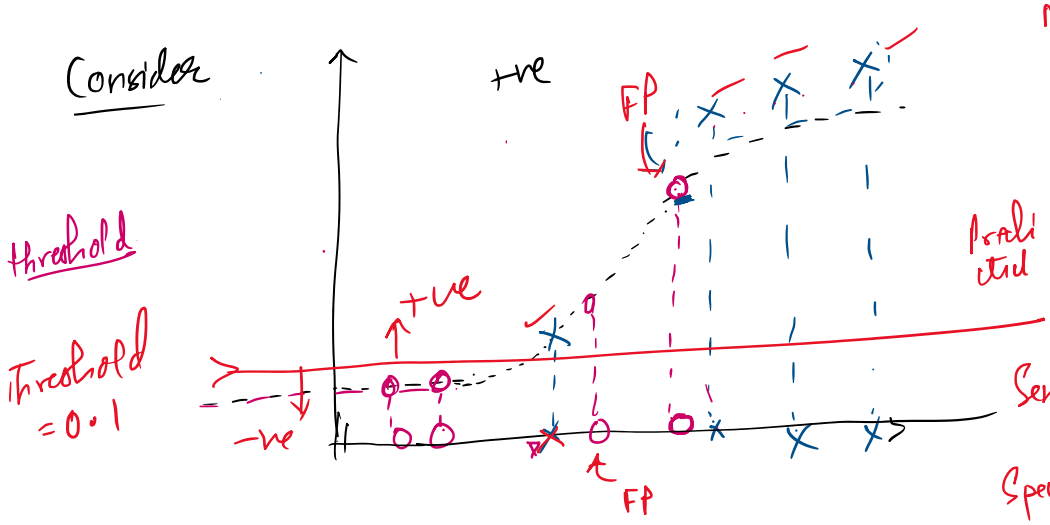
Confusion matrix

	Actual +ve	Actual -ve
+ve	3	1
-ve	1	3

$$\text{Sensitivity} = \frac{3}{4} \quad \text{TPR}$$

$$\text{Specificity} = \frac{3}{4} \quad \text{TNR}$$

$$\text{FPR} = 1 - \text{Specificity}$$



Now threshold = 0.1

Confusion matrix

	Actual +ve	Actual -ve
+ve	4	2
-ve	0	2

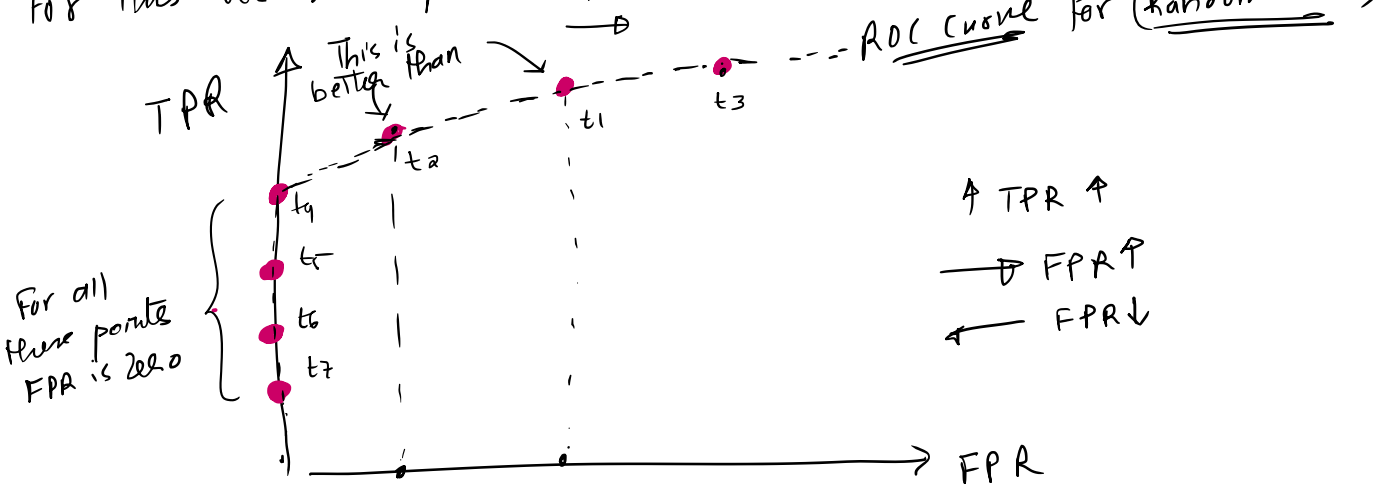
$$\text{Sensitivity} = \frac{4}{4} = 1$$

$$\text{Specificity} = \frac{2}{4} = 1/2$$

* Consider for logistic Regression, where we identify a threshold point and prepare confusion matrix and calculate Sensitivity & Specificity.

and calculate Sensitivity & Specificity

* If threshold changes then the confusion matrix and accordingly the Sensitivity and Specificity changes.
* We can have many such thresholds bet^n 0 → 1
* We want to analyze the performance at diff threshold and want to identify the best of it.
* For this we will plot TPR and FPR for diff threshold.



→ From the above ROC Curve for Random Forest model, it will help us to find the best threshold

Q) Explain

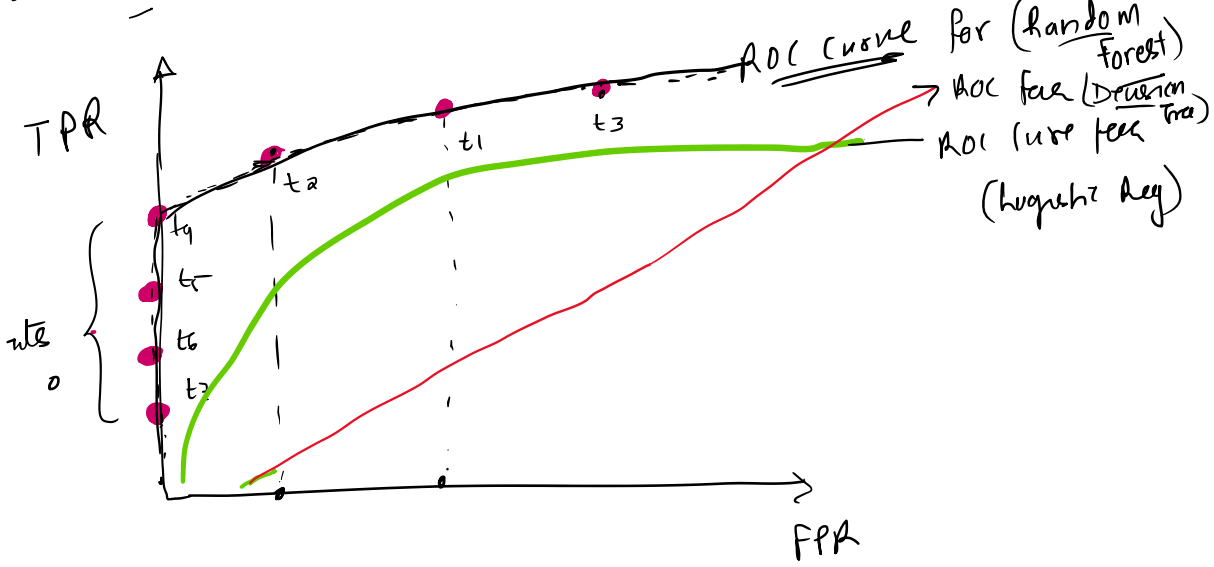
AUC [Area Under Curve] → It is a method to compare ROC for more than one method and will help to judge which one is better.

→ Here will find Area Under ROC for each method.

→ ROC that has maximum Area, the corresponding method will be best.

ROC Curve for (Random Forest)

→ ROC that has maximum Area, the corresponding



From above ROC Curves, we find that there is maximum area under ROC Curve for Random forest, hence the method Random forest will be the best method.

Q) Explain Kappa Statistic?

* Kappa Statistic or Cohen's Kappa is statistical measure of inter-rater reliability for categorical variable.

* It is used when two/more raters apply a criteria based on a tool to assess whether or not some condition occur.

Ex Let say Two doctors rates whether or not each of 20 patients has diabetes based on symptoms

* If two raters uses same criteria on same target to evaluate and then their agreement is very high then we will have evidence of reliable rating

* If their agreement is not very high then →

→ Either criterion tool is not useful
→ or raters are not trained enough.

* Kappa statistics corred for chance agreement and not percent agreement

Ex

Evalor B \ Evalor A	Yes	NO
Yes	35	20
NO	15	40

⇒ 35 times both agreed - said Yes
⇒ 40 times both agreed → said NO
⇒ 20 time A said NO but B said Yes
⇒ 15 time A said Yes but B said NO.

⇒ 20 time A said NO but B said YES

⇒ 15 time A said Yes but B said NO.

Cohen Suggested following Statistics. ⇒

value ≤ 0	⇒	No agreement
0.01 → 0.20	⇒	as none to slight
0.21 → 0.40	⇒	as fair
0.41 → 0.60	⇒	as moderate
0.60 — 0.80	⇒	as Substantial
0.81 — 1.00	⇒	perfect agreement.

* Rather than Calculating the percentage of items, the raters agreed on Cohen's Kappa attempts to account the fact that raters may happen to agree on some items purely by chance

Ex Two Curators asked to rate 70 paintings.

Curator C1 \ Curator C2	Yes	NO
Yes	25	10
NO	15	20

Step (1) Calculate Relative Agreement bet^n Curators.

$$P_o = \frac{\text{Both Said Yes + Both Said NO}}{\text{Total}} = \frac{25+20}{70} = 0.6429$$

Step 2 > Calculate hypothetical probabilities of chance Agreement bet^n Curators.

$$P(\text{Yes}) = C_1(\text{Yes}) * C_2(\text{Yes}) = \frac{(25+10)}{70} * \frac{(25+15)}{70} = 0.2857$$

$$P(Yes) = \frac{C_1(Yes)}{\text{Total Rus}} * \frac{C_2(Yes)}{\text{Total Rus}} = \frac{(25+10)}{70} * \frac{(25+15)}{70} = 0.2857$$

$$P(No) = \frac{C_1(No)}{\text{Total Rus}} * \frac{C_2(No)}{\text{Total Rus}} = \frac{(15+20)}{70} * \frac{(10+20)}{70} = 0.214285$$

$$P_e = P(Yes)+P(No) = 0.2857 + 0.214285 \approx 0.5$$

Calculate Kohen's Kappa $= K = \frac{P_o - P_e}{1 - P_e} = \frac{(0.6429 - 0.5)}{1 - 0.5} = 0.2857$

It is in range $0.21 \rightarrow 0.40$ so the agreement bet^n two Cuthor is fair

Module - 3

Ensemble learning →

3.1 Understand Ensembles

K-fold Cross Validation

→ Boosting

Stumping (AdaBoost)

XGBoost

3.2 → Bagging

Subagging

Random Forest

Comparison with Boosting

Different ways to Combine Classifiers

* Ensemble. ->

* In ML, ensemble is a model that combines the prediction from two or more models.

* The models that contributes to Ensemble are known as ensemble members.

* The members may or may not be trained on same training data and they may be of same type or different type.

* It is very powerful method to improve the performance of the model

* It is technique that uses group of weak learners in order to create a strong and aggregated learner.

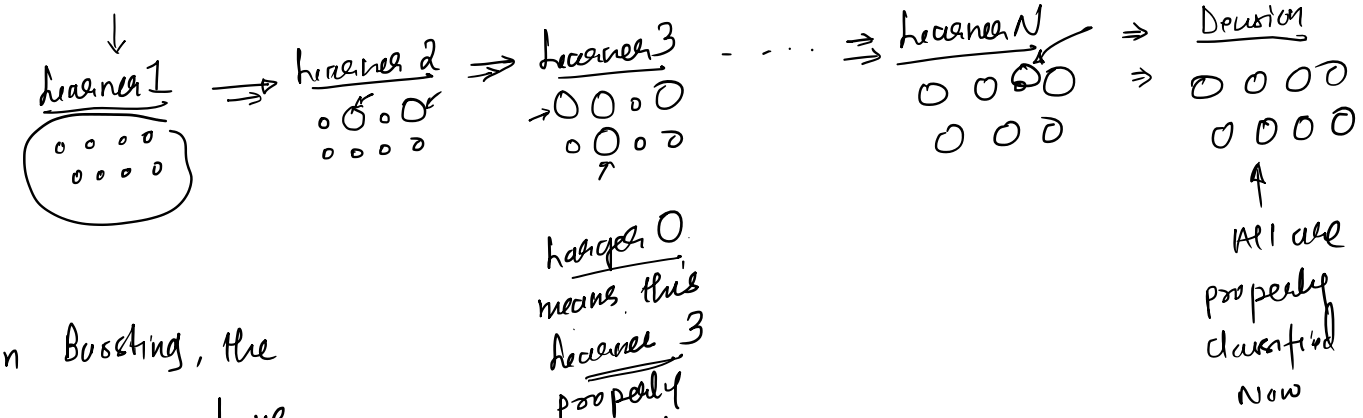
-> The Ensemble technique helps to reduce the Variance (By Bagging) and Bias (By Boosting) and thus helps in improving the predictions.

-> Boosting model: * It falls inside family of Ensemble method.

* It consist of filtering or weighting the data that is used to train team of weak learners, so that the new learner can give more weight on sample that is poorly classified by previous learner.

-> In Boosting the learners are trained

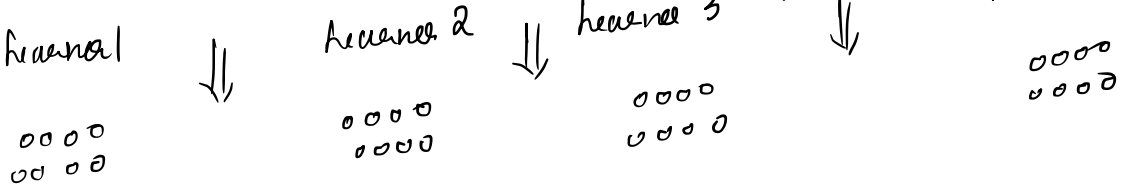
Sequentially.



Learner 0 means this Learner 3 properly classifieds the poorly classified result of learner 2

* In Boosting, the learners can have diff importance or weight (represented by diff sizes of circles).

Bagging:



⇒ In Bagging the weak learners are trained in parallel using randomness

⇒ All learners have same weights.