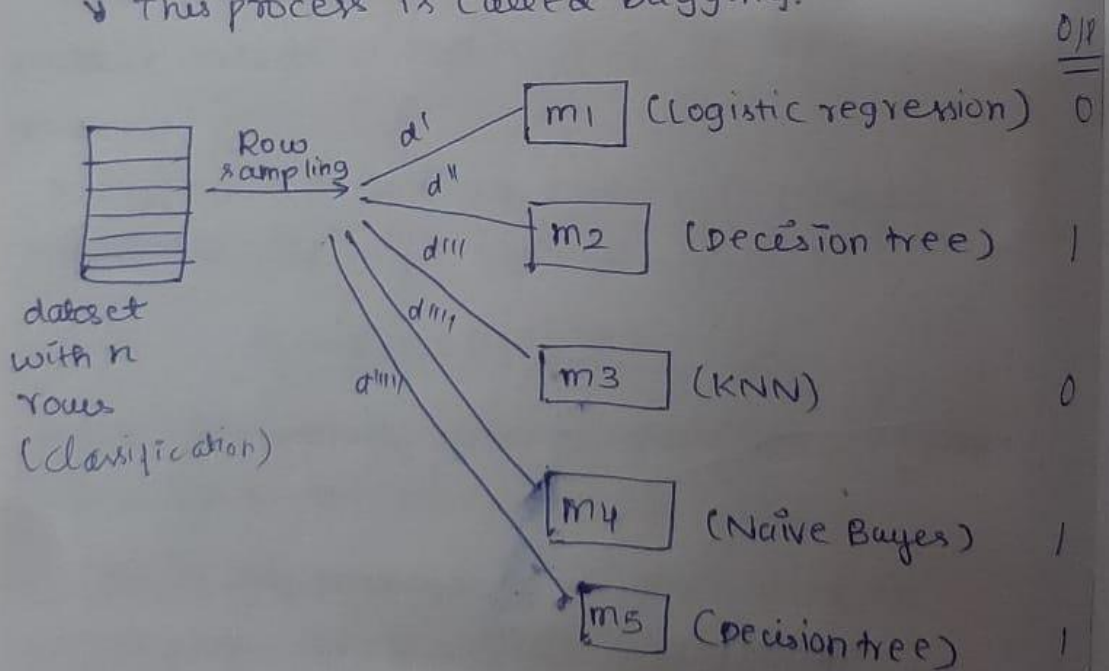
Bagging and boosting:-

* Ensemble techniques or methods basically means the combination of different models together.

* For example a dataset has 1000 records, row sampling will done, ~~and so~~ splitted as subsets and parsed to all the models and get the output.

* In classification problem, the final O/P selected on the basis of voting, and in case of regression the mean of all the O/P were taken.

* This process is called bagging.



* The models can repeat.

* The O/P were (0,1,0,1,1) as the majority 1 will be taken as final

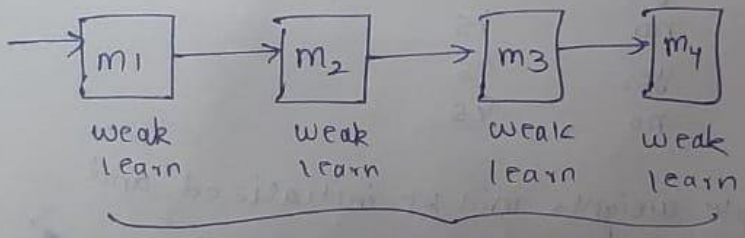
* this bagging process will contain 100 to 200 models.

* In case of regression, mean of all the models will be taker.

* This ~~is~~ ~~ads~~ process of Bagging is also called Bootstrap aggregating.

Boosting

* It is the sequential set of all the models combined together and each models will be a weak learners and at ~~at~~ final all the weak learners combined to become a strong learner and will give good O/P.



* weak learners basically means it gives bad training accuracy, the records which gave wrongly predicted O/P will be given to next model along with other features.

* Boosting as some techiniques:
  i) Adaboost
  ii) Gradient boost
  iii) Xgboost (extreme gradient)

Adaboost :-

* It uses only decision trees for all the weak learner.

* ~~Boosting~~

* The features will be selected according to the information gain & entropy.

* Each weak learner will be called as decision stumps which means, decision tree with one level.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
Yes	Yes	205	Yes	1/8
No	Yes	180	Yes	1/8
Yes	No	210	Yes	1/8
Yes	Yes	167	Yes	1/8
No	Yes	156	No	1/8
No	Yes	125	No	1/8
Yes	No	168	No	1/8
Yes	Yes	172	No	1/8

This is the sample dataset which is a classification problem where we want to find whether patient has heart disease or not.

Initially the sample weights will be assigned to all the features by the formula **1/(total no of records)**

Then according to the high information gain the first feature will be selected as the first stump.

For example if the weight column has high info gain then it is selected as the first stump.

Now our task it to update the sample weights

For that we want to calculate the total error of the stump means adding the sample weights of incorrectly classified record.

As per the above example 4th record is incorrectly classified remaining all were correct.

So

Total error=1/8

Now we want to calculate amount of say for the dataset that is performance of the data

$$\text{Amount of Say} = \frac{1}{2}\log\left(\frac{1 - \text{Total Error}}{\text{Total Error}}\right)$$

Now we want to update the sample weights

New sample weight formula for correctly classified records

$$\text{New Sample Weight} = \text{sample weight} \times e^{-\text{amount of say}}$$

New sample weight formula for incorrectly classified records

$$\text{New Sample Weight} = \text{sample weight} \times e^{\text{amount of say}}$$

now we can update the table with new weights.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	New Weight
Yes	Yes	205	Yes	1/8	0.05
No	Yes	180	Yes	1/8	0.05
Yes	No	210	Yes	1/8	0.05
Yes	Yes	167	Yes	1/8	0.33
No	Yes	156	No	1/8	0.05
No	Yes	125	No	1/8	0.05
Yes	No	168	No	1/8	0.05
Yes	Yes	172	No	1/8	0.05

But the condition is if we add all the sample weights it must equal 1, but in above case its not so we want to normalise it

It can be done by adding all the values and dividing all the record's new sample weight with that sum. Then we will get normalised weight.

Right now, if you add up the **New Sample Weights**, you get **0.68**.

So divide all the values with 0.68 and we will get normalised weights whose sum equals to 1 with some + or – errors.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	New Weight	Norm. Weight
Yes	Yes	205	Yes	1/8	0.05	0.07
No	Yes	180	Yes	1/8	0.05	0.07
Yes	No	210	Yes	1/8	0.05	0.07
Yes	Yes	167	Yes	1/8	0.33	0.49
No	Yes	156	No	1/8	0.05	0.07
No	Yes	125	No	1/8	0.05	0.07
Yes	No	168	No	1/8	0.05	0.07
Yes	Yes	172	No	1/8	0.05	0.07

Now we don't need the new weight column we need only normalised weight column and it also set the range for selection of records to create next stump.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight	
Yes	Yes	205	Yes	0.07	**0 - 0.07**
No	Yes	180	Yes	0.07	**0.07 - 0.14**
Yes	No	210	Yes	0.07	**0.14 - 0.21**
Yes	Yes	167	Yes	0.49	**0.21 - 0.70**
No	Yes	156	No	0.07	**0.70 - 0.77**
No	Yes	125	No	0.07	**0.77 - 0.84**
Yes	No	168	No	0.07	**0.84 - 0.91**
Yes	Yes	172	No	0.07	**0.91 - 0.98**

The main purpose and the overall idea behind all these process is to find out which records is incorrectly classifying the outputs and sending it to the next stump if it gets more importance then is will be corrected.

For that finding purpose only the sample weights were assigned, we can able to see that for correctly classified records in updating process the sample weights will reduce when compared to older weights.

But in case of incorrectly classified record the weight will increase compared to old one. So these records with more weight will emphasized more or considered more for the next stump.

Now we want to create new set of dataset for second stump.

Now any random number will be selected between 0 and 1, now we can clearly say that the wrongly classified record will have high probability of getting selected because it has highest range 0.21 – 0.70. If that random number is within any of the range the respective record will be selected for successive dataset.

Chest Pain	Blocked Arteries	Patient Weight	Heart Disease	Sample Weight
No	Yes	156	No	1/8
Yes	Yes	167	Yes	1/8
No	Yes	125	No	1/8
Yes	Yes	167	Yes	1/8
Yes	Yes	167	Yes	1/8
Yes	Yes	172	No	1/8
Yes	Yes	205	Yes	1/8
Yes	Yes	167	Yes	1/8

After selecting for n records the weights will be re assigned as beginning for this new dataset.

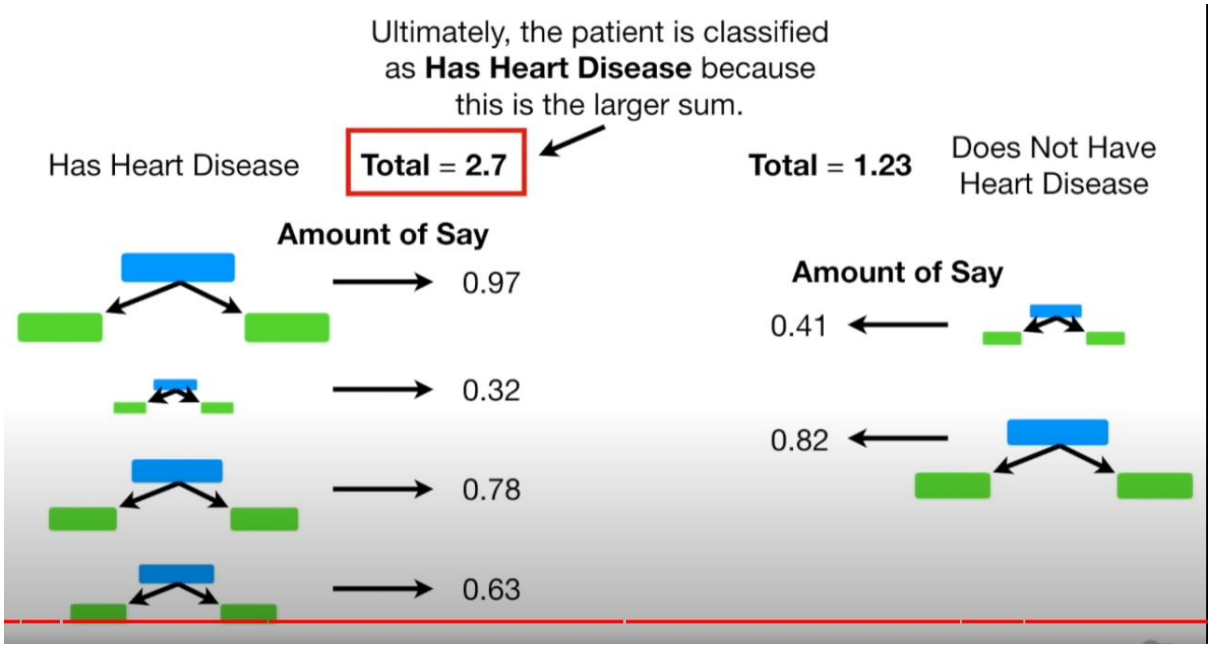
Now with this dataset again with high info gain feature will be selected and threshold will be fixed for that feature for the successive stumps.

This process will go on.

This is how error in first stump affect the successive stumps.

Finally combined together form a strong learner.

Finally the amount of say will be added



So which group of stumps has higher sum of amount of say will classify the output.