AdaBoost Machine Learning Algorithm.

If we create a Decision Tree to it's complete depth it leads to overfitting

Overfitting: More train accuracy → Low Bias
Less test accuracy → High variance

Random Forest:

We reduce overfitting using base learners
- low bias
- low variance

It is the process of Bagging

Boosting

$$\underbrace{\boxed{DT1} \rightarrow \boxed{DT2} \rightarrow \boxed{DT3}}_{\text{Weak Learners}} \rightarrow \text{Strong Learner}$$

Weak Learner → Have'nt learn much from traning dataset

Sequentially connected

AdaBoost:

Assigned weights to the weak learners

AdaBoost function will look like:

$f = \alpha_1(M_1) + \alpha_2(M_2) + \alpha_3(M_3) + \cdots + \alpha_n(M_n)$

where

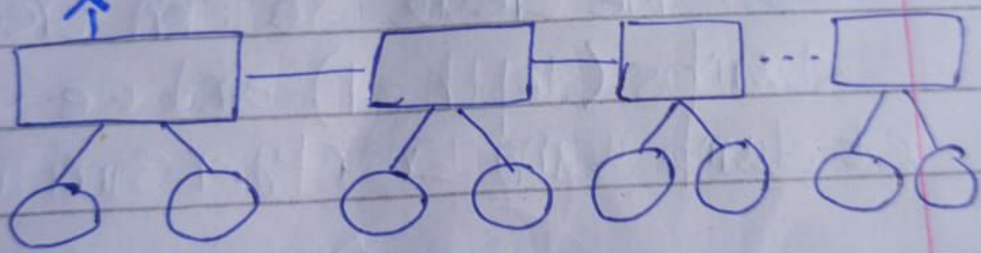
$M_1, \ldots M_n \Rightarrow$ Decision Tree Stumps

$\alpha_1, \alpha_2, \ldots \alpha_n \Rightarrow$ Weights

Decision Tree Stump

Contains decision tree whose depth is just 1

Decision Tree Stump (Weak learner)



It leads to

⇒ Underfitting

Less Train Accuracy → High Bias

More Test Accuracy ⇒ Low Variance

⇒ But in AdaBoost when we ensemble the Decision Stumps in serialize sequential order it leads to

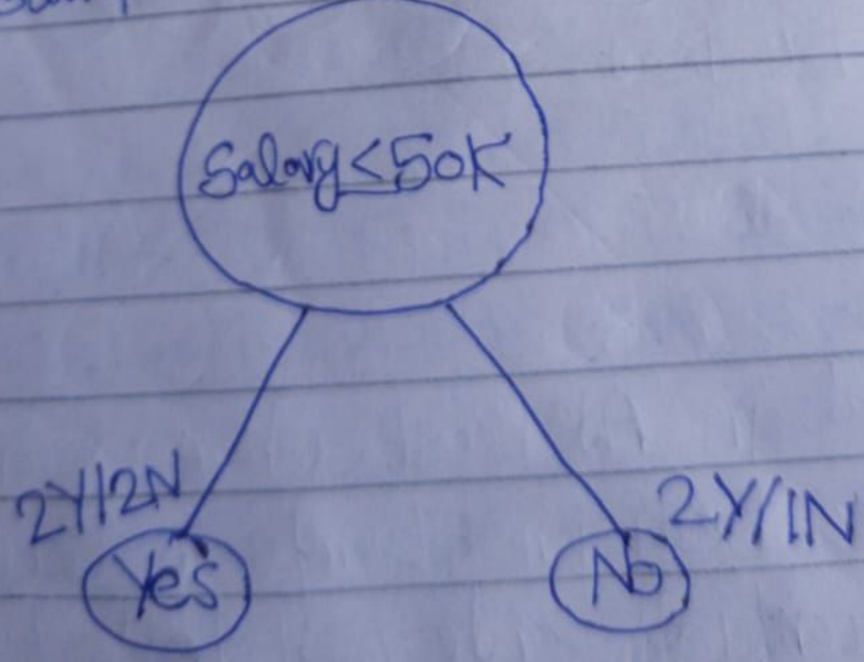
Low Bias

High Variance

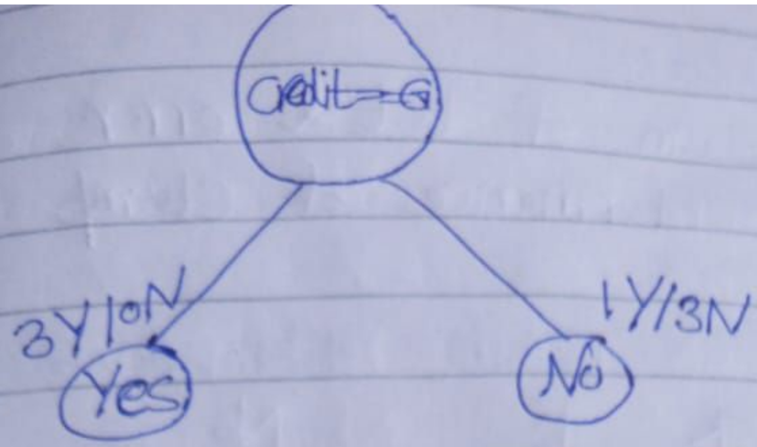
AdaBoost Classifier Maths
Indepth Intuition:

Salary	Credit	Approval
<=50K	B	No
<=50K	G	Yes
<=50K	G	Yes
>50K	B	No
>50K	G	Yes
>50K	N	Yes
<=50K	N	No

Step1: We create decision tree stumps (depth=1) and we select best stump using Entropy or Ginni Impurity.



Decision Tree Stump1



Decision Tree Stump 02

- Now we can create multiple decision tree stumps like this.
- Now we have to decide which decision tree stump need to be selected first. For this we will use **Entropy** or **Ginni Impurity** and we will select lesser impure split means less value of entropy and ginni impurity

Entropy $\Rightarrow -P_+\log_2 P_+ - P_-\log_2 P_-$

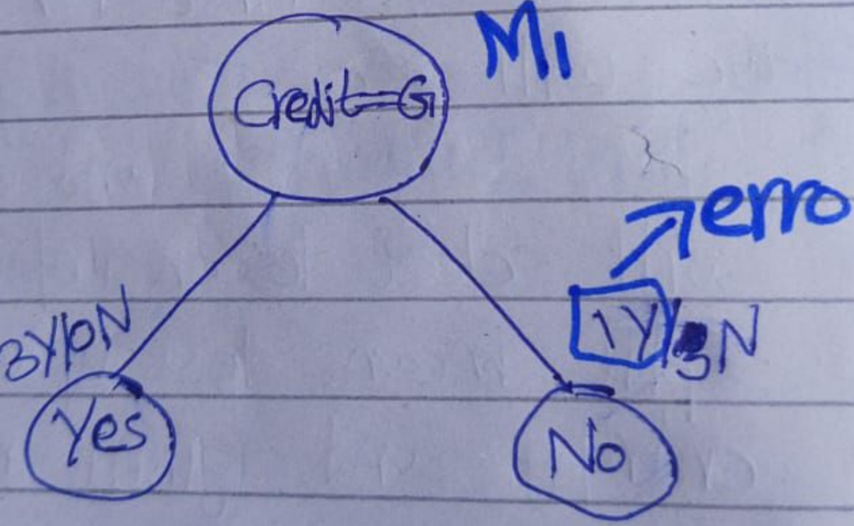
Ginni Impurity $\Rightarrow 1 - \sum_{i=1}^{n} (P)^2$

So **Stump 2** got selected

Step: 02
Sum of total errors and performance of stump

Salary	Credit	Approval	Sample Weights	
<=50K	B	No	$\frac{1}{7}$	
<=50K	G	Yes	$\frac{1}{7}$	
<=50K	G	Yes	$\frac{1}{7}$	
>50K	B	No	$\frac{1}{7}$	
>50K	G	Yes	$\frac{1}{7}$	
>50K	N	Yes	$\frac{1}{7}$	→ error
<=50K	N	No	$\frac{1}{7}$	

M_1



Sum of total error = Sum of sample weights of wrong records

(ii) $\boxed{\text{Sum of total error} = \frac{1}{7}}$

$$\text{Performance of stump} = \frac{1}{2} \ln\left[\frac{1 - \text{Total Error}}{\text{Total Error}}\right]$$

$$= \frac{1}{2} \ln\left[\frac{1 - \frac{1}{7}}{\frac{1}{7}}\right]$$

$\boxed{\text{Preformance of stump} = 0.896}$

The function for AdaBoost is

$f = \alpha_1 (M_1) + \alpha_2 M_2 + \cdots + \alpha_n (M_n)$

So actually performance of stump is α_1

$\boxed{\alpha_1 = 0.896} \Rightarrow$ **Weight**

Step: 03

Update the weights for correctly and incorrectly classified points

* We need to ensure that wrongly predicted record should pass to next decision tree stump

Salary	credit	Approval	Sample weights	Updated weights
<=50K	B	No	$\frac{1}{7}$	0.058
<=50K	G	Yes	1/7	0.058
<=50K	G	Yes	1/7	0.058
>50K	B	No	$\frac{1}{7}$	0.058
>50K	G	Yes	1/7	0.058
>50K	N	Yes	1/7	0.349
<=50K	N	No	1/7	0.058
				total = 0.697

error / wrongly classified

Weight Updation

For wrongly classified points:

$$= \text{weight} * e^{\text{performance of stump}}$$

$$= \frac{1}{7} * e^{0.896}$$

$$= \boxed{0.349}$$

For correct classified points

$= weight * e^{-(performance\ of\ stump)}$

$= \frac{1}{7} \times e^{-(0.896)}$

$= \boxed{0.058}$

⇒ You can see that in updation the weights of correctly classified points decrease where wrongly classified points increase so that next model (weak learner) can pick up wrong ones

Step:04

Normalized Weights Computation and Assigning Bins

- You can see that the total sum of our updated weights is not 1 so we have to normalize it. Currently the total is 0.647 so we have to divide every weight with it to normal

Salary	credit	Approval	Updated Weights	Normalized Weights
<=50K	B	No	0.058	$\frac{0.058}{0.677} = 0.08$
<=50K	G	Yes	0.058	0.08
<=50K	G	Yes	0.058	0.08
>50K	B	No	0.058	0.08
>50K	G	Yes	0.058	0.08
>50K	N	Yes	0.349	0.50
<=50K	N	No	0.058	0.08

error/wrongly predicted

Total=1

Bins Assignment

0 — 0.08
0.08 — 0.16
0.16 — 0.24
0.24 — 0.32
0.32 — 0.40
0.40 — 0.90
0.90 — 0.98

Step: 05

Select datapoints to send to next stump

⇛ Iterative process selecting random values between 0 and 1

Salary	Credit	Approval	Random
>50K	N	Yes	0.50
<=50K	G	Yes	0.10
>50K	N	Yes	0.60
>50K	N	Yes	0.75
<=50K	G	Yes	0.24
>50K	B	No	0.32
>50K	N	Yes	0.87

The marked one were wrongly predicted.

What we did:

We selected random values b/w 0 and 1 7times see that in which bucket that value is coming and added that record.

- As bucket of wrongly predicted point is greater so it is selected most of the time.

Step:06

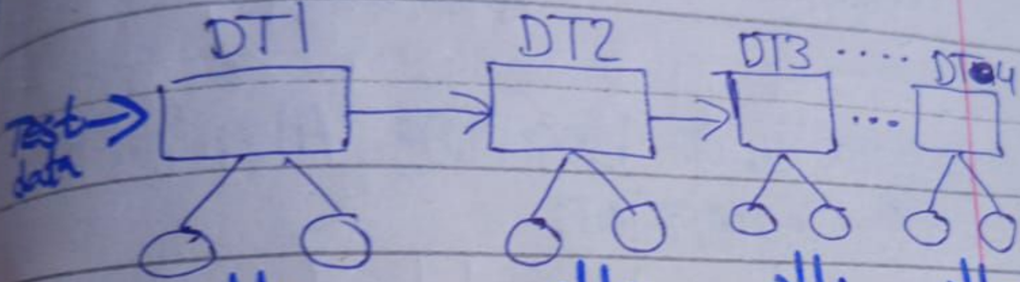
These records will be sent to next Decision Tree Stump.

Salary	Credit	Approval
<=50K	G	Yes
>50K	N	Yes
>50K	N	Yes
<=50K	G	Yes
>50K	B	No
>50K	N	Yes

- Then we will assign weights and repeat all stumps

⇒ Final Prediction for Adaboost

Test dat (salary ≤ 50K, credit = G)



Yes — No — Yes — No

$\alpha_1 = 0.896$ $\alpha_2 = 0.650$ $\alpha_3 = 0.24$ $\alpha_4 = 0.30$

$f = \alpha_1 (M_1) + \alpha_2 (M_2) + \alpha_3 (M_3) + \alpha_4 (M_4)$

$f = (0.896)(Yes) + (0.650)(No) + (0.24)(Yes)$
$\quad - (0.30)(No)$

Now add alphas of Yes and also alphas of No

$f = 1.136 (Yes) + 0.350 (No)$

As ~~weig~~ Performance of say (Yes) > Performance of say (No)

So, **Yes** will be the output.

⇒ In case of AdaBoost Regression we select the stump on the basis of Mean Squared Error (MSE) instead of Entropy.

And the prediction will be

$$\boxed{y(x) = \frac{\alpha_1(M_1) + \alpha_2(M_2) + \alpha_3(M_3) + \alpha_4(M_4)}{\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4}}$$