Ensemble Technique

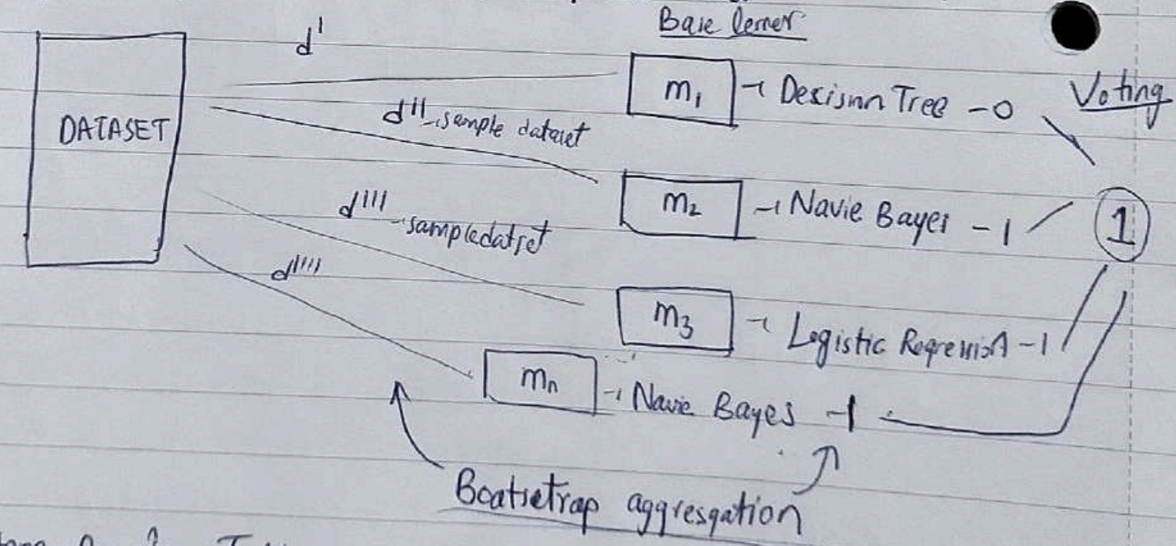
① What is ensemble?
Combining multiple models => TRAIN => PREDICTION

Two types

① Bagging
→ Random Forest Classifier and regressor

② Boasting
① AdaBoast
② Gradient Boast
③ Xgboast

Bagging Techineque: → we create multiple independent model d' = sample dataset.

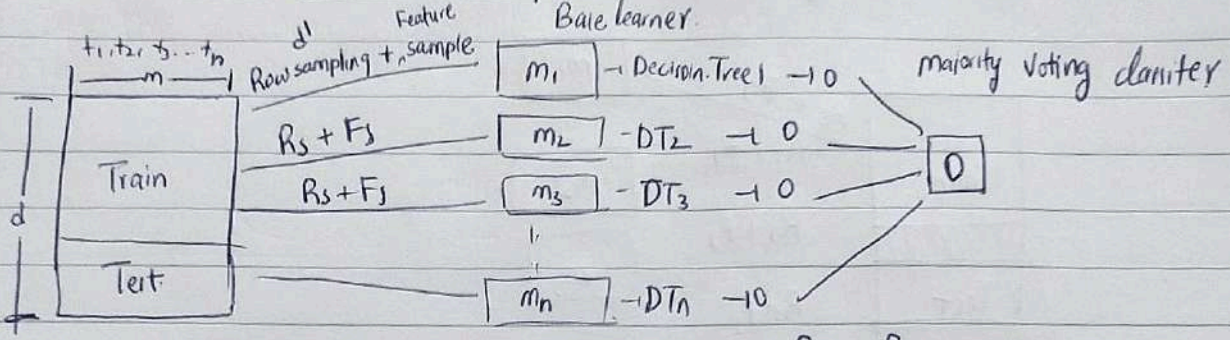


Custom Bagging Techinque

Regression { Avegre of all o/p → Prediction }

→ In Random Forest Classifier we specifically used Decision Tree.

Random Forest Classifier & Regressor



m = features → $d' < d$ $m < n$

New Test Data

R_s → Row sampling

F_s → Feature sampling

{if regression → Average of all the model o/p}

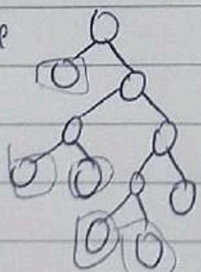
Note

Classification → majority Voting Classifier

Regressor → Average o/p of all the models

① Why Should we use Random Forest instead of DT?

Decision Tree



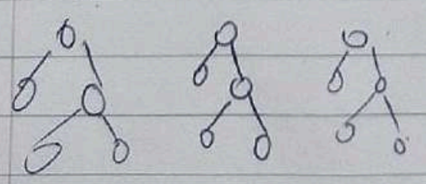
→ Overfitting

Traning Acc↑↑ → Low Bias

Test Acc↓↓ → High variance

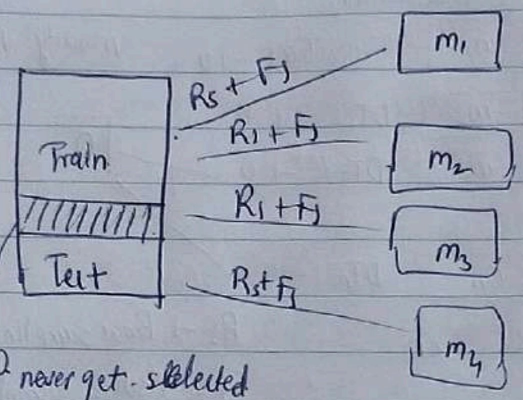
Generalized Model

Low Bias, Low variance



all will be expert in something

Out of Bag score



never get selected

$\boxed{\text{oob\_score=True}}$

so. out of Bag Data

⇓

Validationdata → Performace and accuracy of RamdomG ⇒ oob score → 85%, 90%, 75%

Boasting {sequentially connected}

$\boxed{} \rightarrow \boxed{DT_1} \rightarrow \boxed{DT_2} \dots \boxed{DT_3} \Rightarrow$ Strong Learner

└ Weak leaner

Weak learner : Havan't leaint much from the Traning Dataset.

Random Forest : Majority voting classifier
average of (O/P)

AdaBoast : Assign weigth to the weak learner.

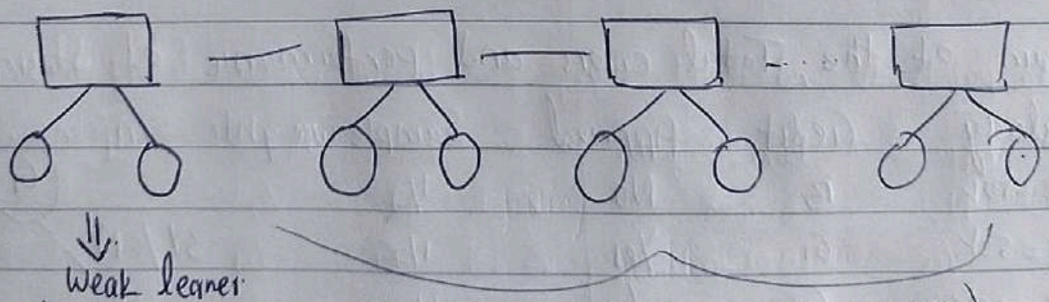
$$f = \alpha_1 (m_1) + \alpha_2 (m_2) + \alpha_3 (m_3) + \dots \alpha_n (m_n)$$

→ classification
→ regression.

$\{\alpha_1 \alpha_2 \alpha_3 \dots \alpha_n\} \Rightarrow$ weights.

Decision
tump

epth = 1



⇓
Weak leaner
└ underfitting
  └ Traning Acc ↓ 40%
  └ Test Acc ↑ 45%

{ High Bias
  low variance

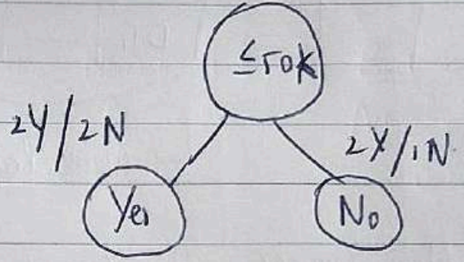
Decision Treestump
⇓

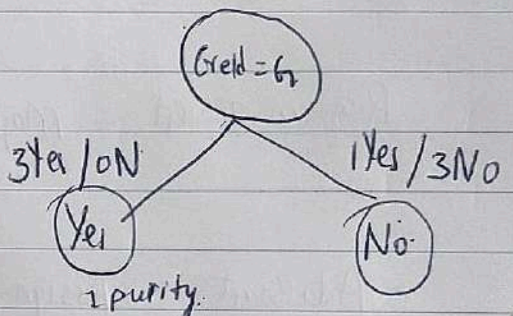
$\Rightarrow$ { low Bias
  High varience }

Adaboost Classifier Maths Indepth Intuition

(1).

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





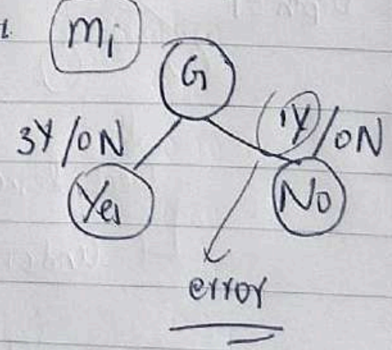
Which Stump should be select first?

Entropy (or) Gini Impurity

$H(s) = -p_+ \log_2 p_+ - p_- \log_2 p_-$ (or) Gini Impurity

(2) Sum of the Total errors and performance of Stump

Salary	Credit	Approval	Sample weights
<=50k	B	No	1/7
<=50k	G	Yes	1/7
<=50k	G	Yes	1/7
>50k	B	No	1/7
>50k	G	Yes	1/7
>50k	N	Yes	1/7
<=50k	N	No	1/7



Sum. of all the
Total error $= \frac{1}{7}$

(2) Performance of stump : $\frac{1}{2} \ln \left[\frac{1 - TE}{TE} \right]$

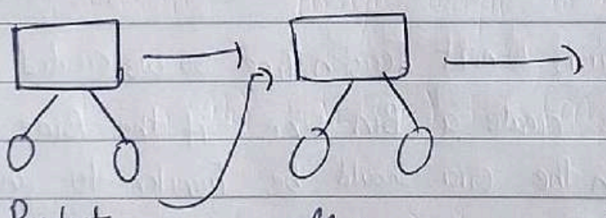
$$= \frac{1}{2} \ln \left[\frac{1 - 1/7}{1/7} \right]$$

$$= \frac{1}{2} \ln [6] = 0.896$$

The performance of stump = 0.896.

$$f = \alpha_1 (m_1) + \alpha_2 (m_2) + \ldots \alpha_n (m_n)$$

$$\boxed{\alpha_1 = 0.896.}$$



Predict
some wrong
record

all the wrong records
will pass to each other

(3) Update the weights for correctly and Incorrectly classified points

Salary	Credit	Approval	Sample	Updated weight
<=50K	B	No	1/7	0.058
<=50k	G	Yes	1/7	0.058
<=50k	G	Yes	1/7	0.058
>50k	B	No	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

For correct classified points

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

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

$= 0.058$

For incorrect classified point

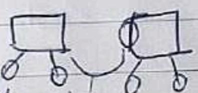
$= \text{weight} * e^{\text{Performance of stump}}$

$= \frac{1}{7} * e^{(0.896)} = 0.349$

(4) Namizeing Weights In Adaboost and assigning Bins

Salary	Credit	Approval	Updatat wts	Normalized weight	Bins $\frac{0.058}{0.697}$
<=50k	B	No	0.058	0.08	0-0.08
<=50k	G	Yes	0.058	0.08	0.08-0.16
<=50k	G	Yes	0.058	0.08	0.16-0.24
~~>=~~ 50k	B	No	0.058	0.08	0.24-0.32
>50k	G	Yes	0.058	0.08	0.32-0.40
>50k	N	Yes	0.349	0.50	$\boxed{0.40-0.90}$
<=50k	N	No	0.058	0.08	0.90-0.98
			0.697	≈1	

Huge range



we have to send all the errors to next decision Tree so we created weigths the namilized it. then create a Bin size. if the Bine size is more the error is there then the error should be Tranfer to another. Decision Tree

(5) Select data points to send Next Stump

Salary	Credit	Approval	Bin assingemnt
<=50k	B	No	0-0.08
<=50k	G	Yes	0.08-0.16
<=50k	G	Yes	0.16-0.24
>=50k	B	No	0.24-0.32
>50k	G	Yes	0.32-0.40
>50k	N	Yes	0.40-0.90
<=50k	N	No	0.90-0.98

(1) Inutive process selecting random value b/w 0 and 1

S	Credit	Appral	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

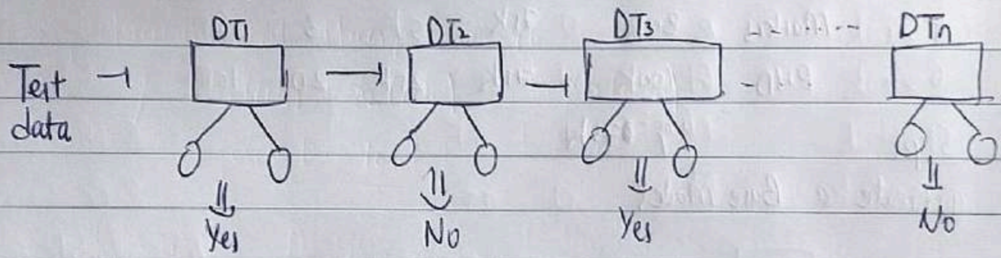
⑥ Then records will be sent to next DT stump

⑦ Final Predictionrion.

α = error

m = model.

test data ($\leq$ 50k, G)



$\alpha_1 = 0.896$. $\alpha_2 = 0.650$ $\alpha_3 = 0.24$ $\alpha_n = -0.30$.

$$t = \alpha_1(m_1) + \alpha_2(m_2) + \alpha_3(m_3) + \ldots \alpha_n(m_4)$$

$$= 0.896\,(\text{Yes}) + 0.650\,(\text{No}) + 0.24\,(\text{Yes}) - 0.320\,(\text{No})$$

$$= \boxed{1.136}\,(\text{Yes}) + 0.350\,(\text{No})$$

0.896
0.240

O/p = Yes

Performance of say Yes = 1.136

Performance to say No = 0.350.

Gradient Boosting Algorithum

1. Regression
2. Classification

R_1 = Residual

Regression Dataset

Exp	Degree	Salary y	$\hat{y}$	$(y-\hat{y})$ R_1	R_2	$\hat{y}$	R_3	R_4
2	BE	50k	75K	-25k	-23	72.7	-22.7	
3	Masters	70k	75K	-5k	-3	74.7	-4.7	
5	Masters	80k	75k	5K	3	85		
6	PHD	100k	75k	25k	20	105		
		avg = 75k						

Steps

① Create a Base model

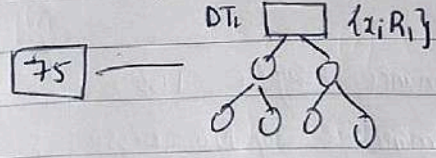
$\boxed{75} \rightarrow$ average $= \frac{50+70+80+100}{4} = 75$.

② Compute Residual, error

③ Construct a Decision Tree

Consider input x_i and o/p R_1

New record →



Predicted o/p ⇒ $75 + (-23) = 52$ // {overfitting}

Predicted o/p ⇒ $75 + \alpha(DT_1) = 75.(0.1)(-23)$

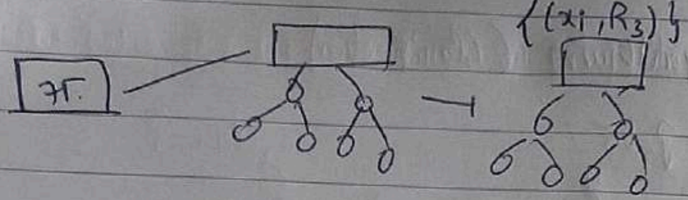
α = learning (0 to 1)

$= 75 - 2.3$

$= 72.7$ //

$\Rightarrow 75 + 0.1(-3)$

$\Rightarrow 75 - 0.3 = 74.7$



$F(x) = \alpha_0 h_0(x) + \alpha_1 (h_1(x)) + \alpha_2 (h(x) + \alpha_3 (h(x)) \ldots \alpha_n (h(x))$.

Learning Rate $\alpha = 0.1$

$$\boxed{F(x) = \sum_{i=0}^{n} \alpha_i h_i(x)}$$

Final Function

Grandiant Boarting

Xg.Boost ML Algorithum (Classification)

$\boxed{\hat{y}=0.5}$

Dataset

Salary	Credit	Approved (y)	R_1	$\hat{y}$	R_2 (App $-\hat{y}$)
<=50k	B	0	-0.5	0.52	-0.48
<=50k	G	1	0.5	0.58.	0.42
<=50k	G	1	0.5	-	
>50k	B	0	-0.5	-	
>50k	G	1	0.5	-	
>50k	N	1	0.5	-	
<=50k	N	0	-0.5	-	

Steps

1. Construct a base Model
2. Construct a decision Tree with root.
3. Calculate Similarity $= \dfrac{\Sigma (\text{Residual})^2}{\Sigma Pr\,(1-Pr)+\lambda}$
4. Calculate Gain

① Base [ ] — [Salary] SW=0.14

$Pr=0.5$ ($\hat{y}$)

$\le 50k$: [-0.5, 0.5, 0.5, -0.5] Similary w=0

$>50k$: [-0.5, 0.5, 0.5] Simal w=0.33

Similarity weigth (LC) $= \dfrac{\Sigma (\text{Residual})^2}{\Sigma Pr\,(1-Pr)+\lambda}$ ← λ hyperparamet. equ.

$$= \frac{\left[(\cancel{-0.5}+\cancel{0.5}+\cancel{0.5}-\cancel{0.5})^2\right]}{\left[0.5(1-0.5)+0.5(1-0.5)+0.5(1-0.5)+0.5(1-0.5)\right]} \quad \boxed{=0.}$$

Rigth child

$$\text{Simaily (RC)} = \frac{(-0.5+0.5+0.5)^2}{0.75} = \frac{0.25}{0.75} = \frac{1}{3}$$

$\boxed{=0.33}$

Gain = Leftchild + rightchild − root

$= 0 + 0.33 - 0.14$

$= 0.21$

o/p = 0 → √Salary

≤50k → [−0.5, 0.5, 0.5, 0.5]

>50k → [−0.5, 0.5, 0.5]

Sim = 0

Simi wei = 0.33

CREDIT: B → [−0.5], sim = 1; G, N → [0.5 0.5 0.5], SW = 0.33

Credit: B, G; N

$$(L.C) = \frac{(-0.5)^2}{0.5(1+0.5)} = \frac{0.25}{0.25} = 1$$

$$\text{Simy of } (R.C) = \frac{0.25}{0.75} = 0.33$$

$$\text{Gain} = 1 + 0.33 - 0 = 1.33$$

Test data

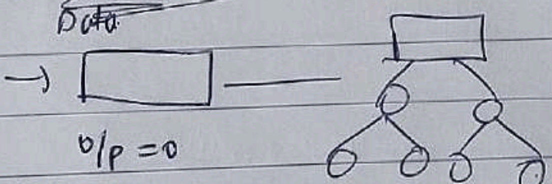
$$\log(\text{odds}) = \log\left(\frac{p}{1-p}\right)$$

$$\log(\text{odds}) = \log\left(\frac{0.5}{0.5}\right)$$

$$= 0$$

Final o/p

~~New Test Data = σ(0~~



o/p = 0

New data $= \sigma\,(0 + \alpha(1))$ ← simplarity weigth $\quad \alpha = 0.1$

sigmoid activation / logistic function

$$= \sigma\,(0 + (0.1)(1))$$

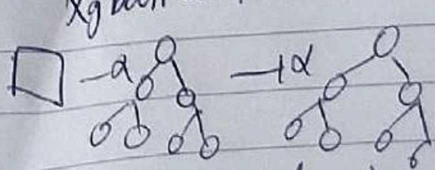
$$= \frac{1}{1+e^{-0.1}} = 0.52$$

Secondrecord $= \sigma\,(0 + \alpha(0.33))$

$$= \sigma\,(0 + 0.1(0.33))$$

$$= \frac{1}{1+e^{-0.033}} = 0.58.$$

Xg boost clasifier



o/p = σ (Base learner)

$op = \sigma(\text{Base learner} + \alpha_1 (DT_1) + \alpha_2 (DT_2) + \alpha_3 (DT_3))$