Ensemble Techniques :-

Bagging and Boosting

① Random Forest Classifier

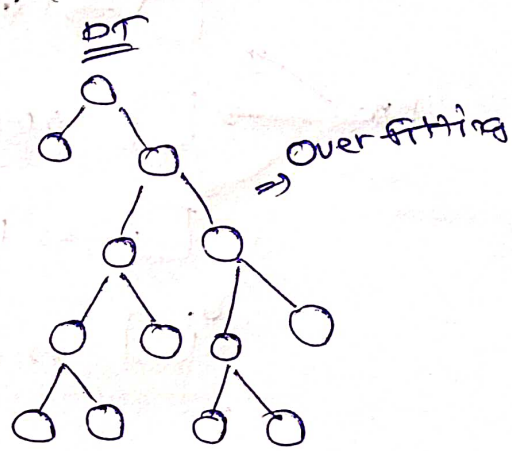
② Random Forest Regression

Decision Tree :-

* Problem is overfitting

  low bias

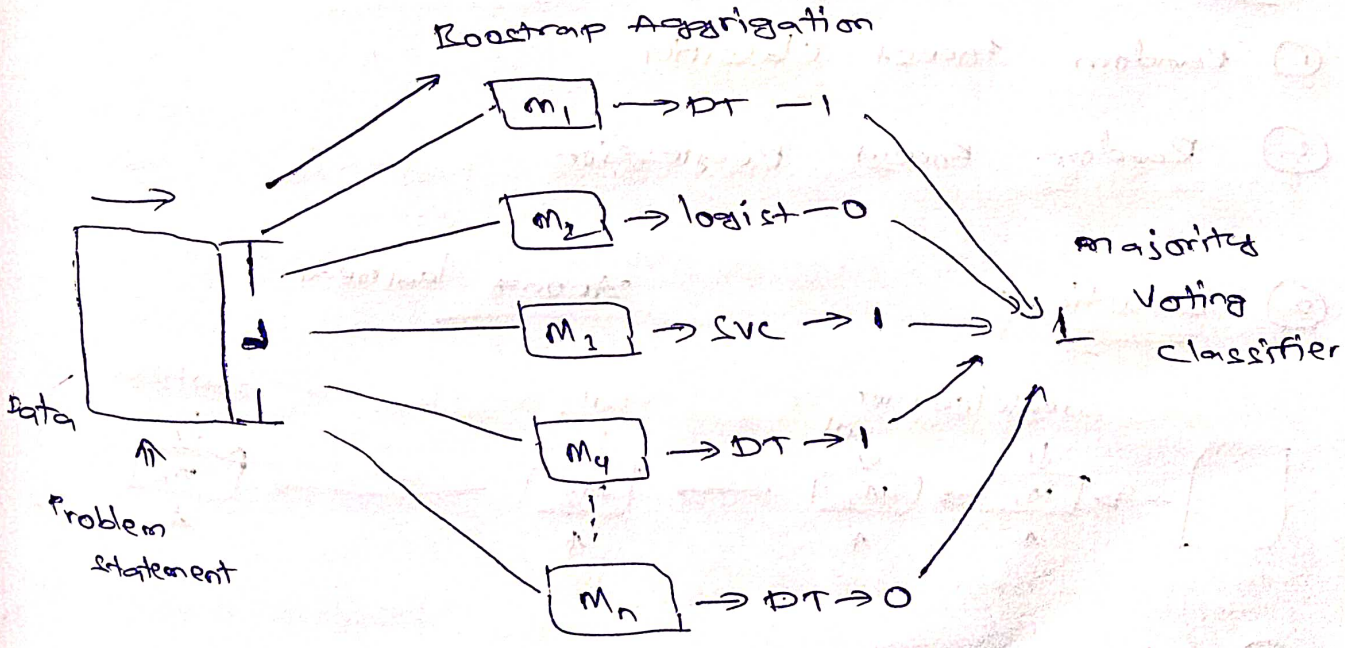
  high varience



Bagging and Boosting

① Bagging :-

* Bagging is used when our objective is to reduce the variance of decision tree

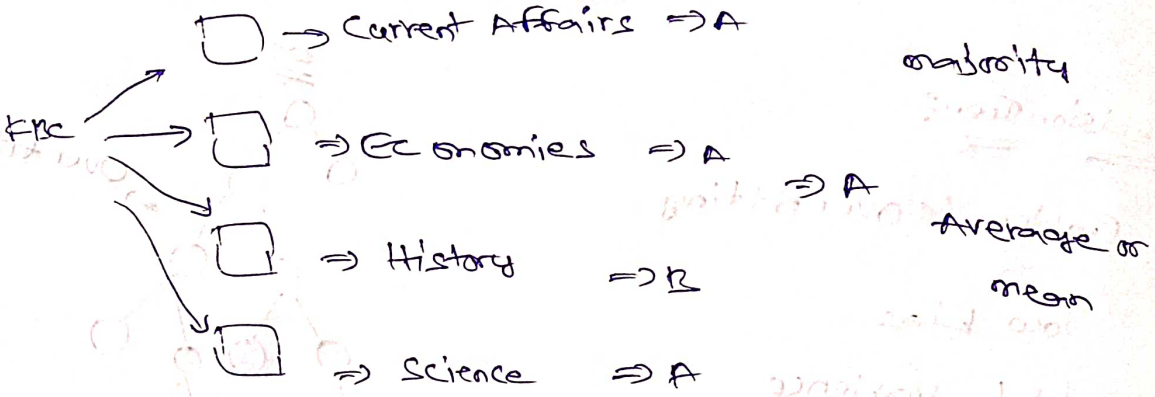


Example:

KBC : Kaun Banega Corepathi (Amitha bachan)

krish → Data Science

UPSC → Diversified [Apti] [logical] [math] [Soci] [curr] [Hist]



Used Ensemble Techniques:

↓

Kaggle, Hacherrank

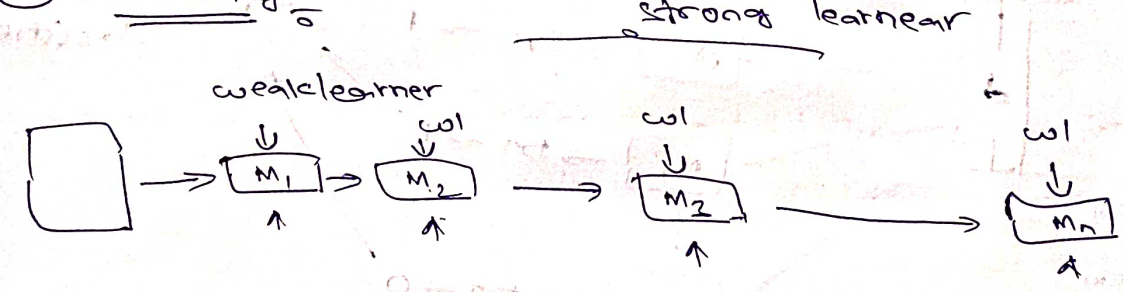
↓

Outperforming results

Bagging

1. Random Forest classifier
2. Random Forest Regression

2. Boosting:



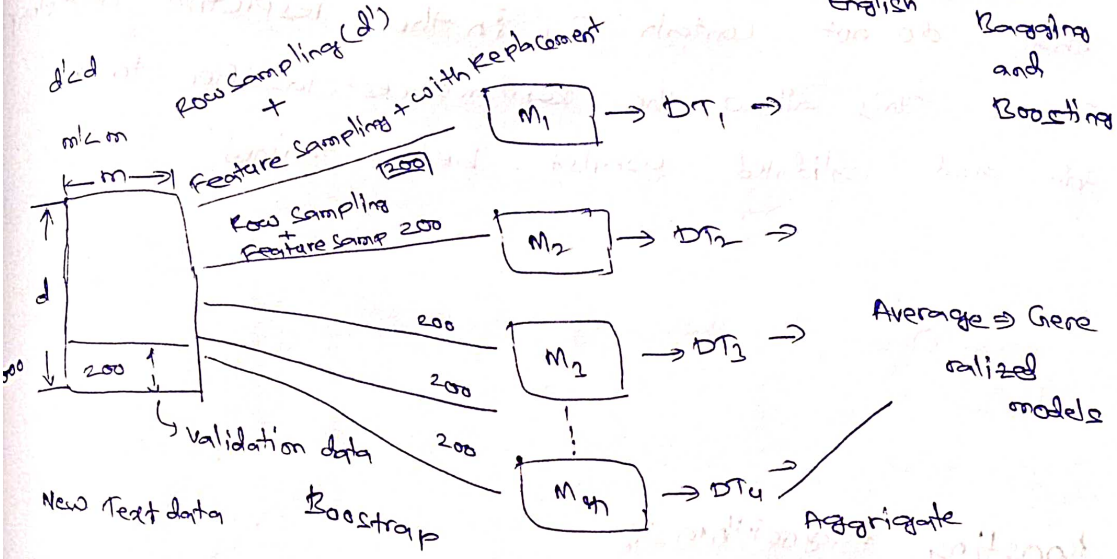
1. AdaBoost Regressor And classifier
2. Gradient Boost Regression and classifier
3. Xgboost Regression and classifier

↳ Xtream Gradient boost

Random Forest Classifier and Regression : Ensemble Technique

Ensemble → English

Technique → Bagging and Boosting



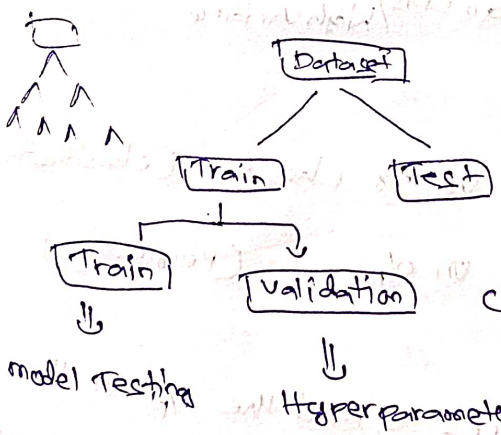
oob - score = True

DT1 → Overfitting → Preprocessing using Pre pruning & Post pruning

⇒ Cubersome Task

① Low Bias → Training

② Hig Variance → Test → low variance



model Testing

Validation → Hyperparameter tuning → score

(Cross validation (Grid search(CV))

oob :

Out of bag Evaluation

oob - score ⇒ 83%

↓

validation

Out of Bag Error

1 - oob - score = 1 - 83 = 0.17%

↓

{validation Error}

oob Errors

* the out of bag (OOB) error is the average error for each z_i calculated using predictions for the trees that do not contain z_i in their respective bootstrap sample. This allow the RandomForestClassifier to be fit and validated whilst being trained

Boosting Algorithms:-

* sequential weak learners
* Boosting tries to reduce bias.

① Adaboost

step

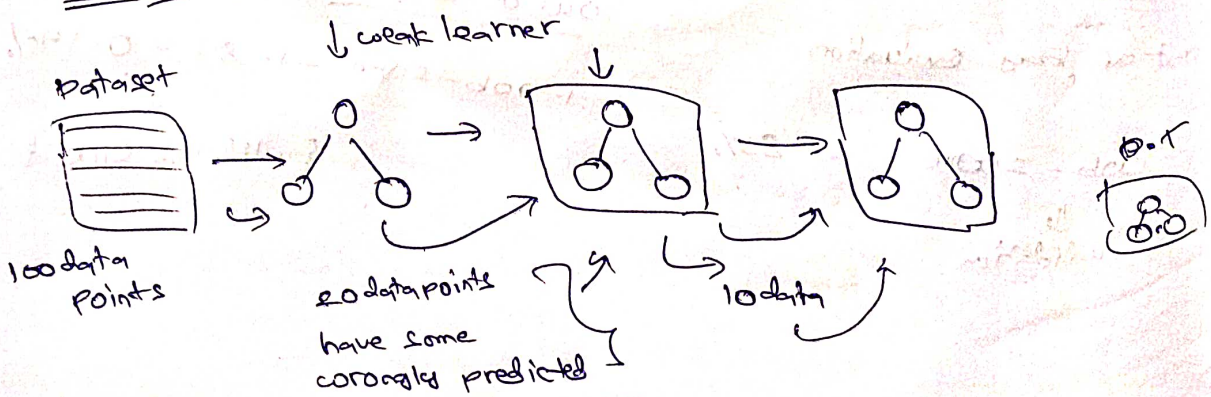
Boosting { low Bias ⇐ high Bias
low Variance ⇐ low Bias / high Variance

underfitting { Training data Acc↓↓ 4%
Test data Acc↑↑ 45%

Random Forest → majority voting classifier [classification]
Average of o/p [Regression]

Adaboost → weak learners → Add the o/p of the weak learners with some weights assigned to it

Adaboost :-



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

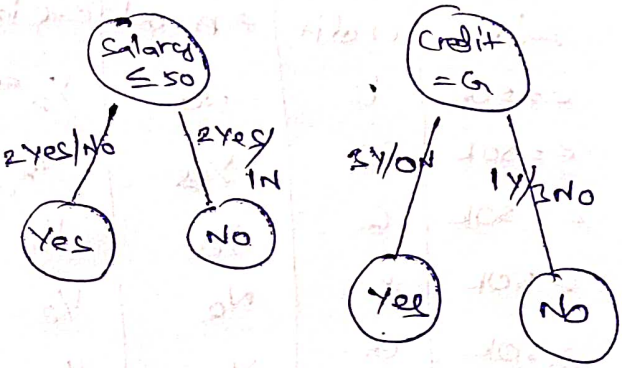
$M_1, M_2, M_3 \dots M_n \rightarrow$ weak learners $\rightarrow$

$\alpha_1, \alpha_2, \alpha_3 \dots \alpha_n \rightarrow$ weights

$\Rightarrow$ total (7)

Salary	Credit	Approval	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

① We created DT stump by selecting the best one

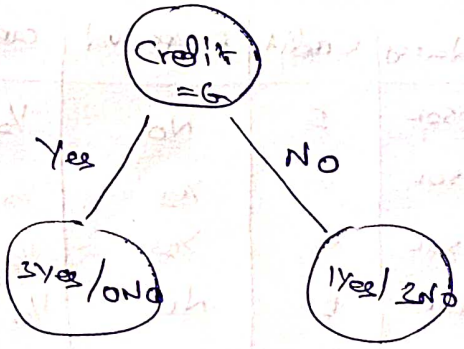


Entropy or Gini

$$H(S) = -p_+ \log_2 p_+ \qquad -p_- \log_2 p_-$$

② Salary Credit Approval weight { B = Bad, G = Good, N = Normal }

<=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



③. Calculate the Total Error (Add the weights of wrong data points)

$$T.E = \frac{1}{7}$$

③ Performance of stump $= \frac{1}{2}\ln\left(\frac{1-TE}{TE}\right) = \frac{1}{2}\ln[6] \approx 0.89$

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

$\alpha_1 = 0.896$

④ Update the weight for Correctly and Incorrectly data Points

Salary	Credit	Approval	weight	Update weight
<=50K	B	No	$1/7 = 0.14$	0.58
<=50K	G	Yes	1/7	0.58
<=50K	G	Yes	1/7	0.58
>50K	B	No	1/7	0.58
>50K	G	Yes	1/7	0.58
>50K	N	Yes	1/7	0.349
<=50K	N	No	1/7	0.58

For Correctly Classified points

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

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

$= 0.058$

For Incorrect Classified Point

$\Rightarrow$ weight $\times e^{\text{Performance}}$

$\Rightarrow 1/7 * e^{(0.896)}$

$\Rightarrow 0.349$

⑤ Normalized weights and Assigned Bins :-

Salary	Credit	Approval	weights	Updated weights	Normalized weights	Bins Assigned
<=50K	B	No	$1/7 = 0.14$	0.058	0.08	0 – 0.08
<=50K	G	Yes	1/7	0.058	0.08	0.08 – 0.16
<=50K	G	Yes	1/7	0.058	0.08	0.16 – 0.24
>50K	B	No	1/7	0.058	0.08	0.24 – 0.32
>50K	G	Yes	1/7	0.058	0.08	0.32 – 0.40
>50K	N	Yes	1/7	0.349	0.50	0.40 – 0.90
<=50K	N	No	1/7	0.058	0.08	0.90 – 1
				0.697	≈ 1	

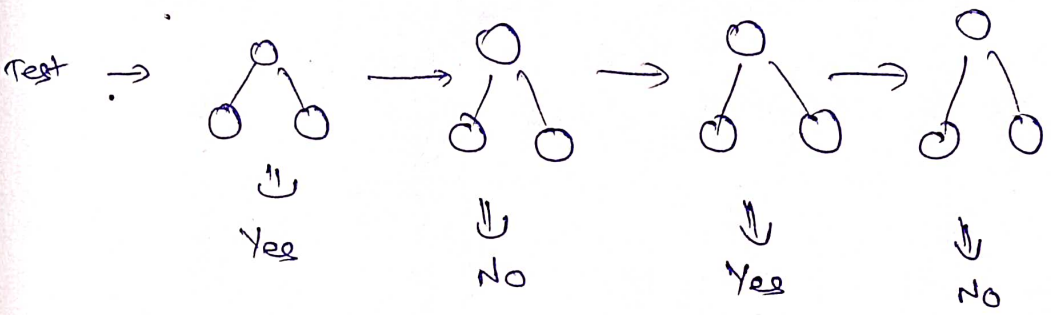
0-1 → 0.09

Random (0 to 1)

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

(*) Final Prediction

Test (≤50K, G)



↳ $\alpha_1 = 0.896$ $\alpha_2 = 0.650$ $\alpha_3 = 0.36$ $\alpha_4 = 0.20$

$f = L_1(M_1) + L_2(M_2) + L_3(M_3) + L_4(M_4)$

$= 0.896(\text{Yes}) + 0.650(\text{NO}) + 0.18(\text{Yes}) + 0.20(\text{NO})$

$= 1.2(\text{Yes}) + 0.85(\text{NO})$

↓

Yes