# Adaboost & Random Forest

(1)

## Ensemble Techniques:-

- BAGGING → Test data
- BOOSTING

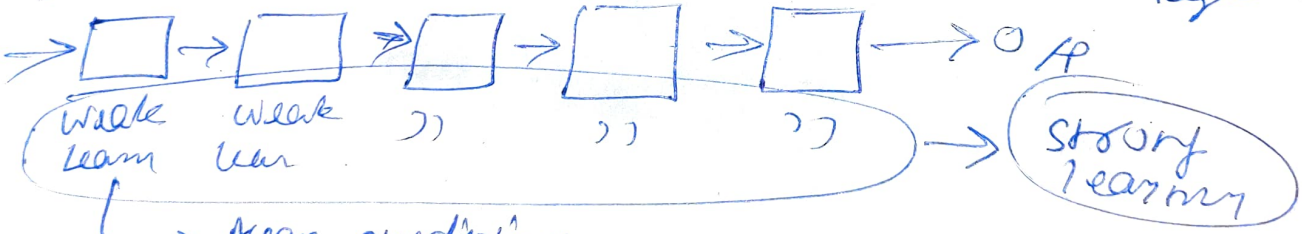

In Regression O/P mean will be taken

Bagging
① Random F Classifier
② Random F Regressor

## Boosting

(2)

Sequential models

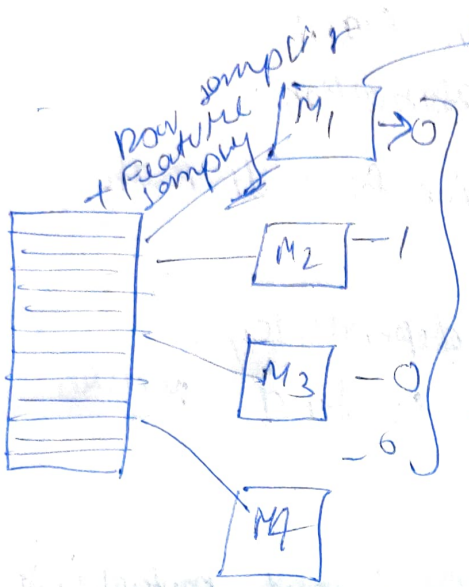


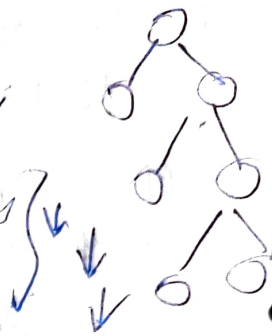

Mean prediction is very weak that's why we combined models.

① Adaboost
② Gradient
③ Xgboost

~~Topics Deep Search~~

DT



Low Bias
High var

Low Bias
Low var

Overfitting – Train more accu
Test less accu

Is normalization required in random forest or DT → No, because we are using split so skew or norm do not mat much necessary.

Is random forest impacted by outliers → Yes coz they hinders splitting process.

Is KNN impacted by outliers → Yes.

Boosting Technique → Solve in sequential

## (3) Adaboost

(Overall=1)

Information gain

| f1 | f2 | f3 | f4 | O/P | Weight |
|---|---|---|---|---|---|
| – | – | – | – | Yes | 1/7 |
| – | – | – | – | No | 1/7 |
| – | – | – | – | – | 1/7 |
| – | – | – | – | – | 1/7 |
| – | – | – | – | – | 1/7 |
| – | – | – | – | – | 1/7 |
| – | – | – | – | – | 1/7 |

7 rows

Stumps → f1 (Level) → f2, f3

Best feature we find from entropy

① Weak learner – 1-level-stumps

Wrong

Total error = 1/7

② Performance of stumps:

$$= \frac{1}{2}\log_e\left(\frac{1-TE}{TE}\right) = \frac{1}{2}\log_e\left(\frac{1-\frac{1}{7}}{\frac{1}{7}}\right) = 0.895.$$

# ~~Deep learning & NLP~~

(3) New sample weight = weight × $e^{-PS}$ (2)

= $\frac{1}{7} \times e^{-0.895}$ = 0.5

Incorrect record = Weight × $e^{PS}$ = $\frac{1}{7} \times e^{0.895}$ = 0.349

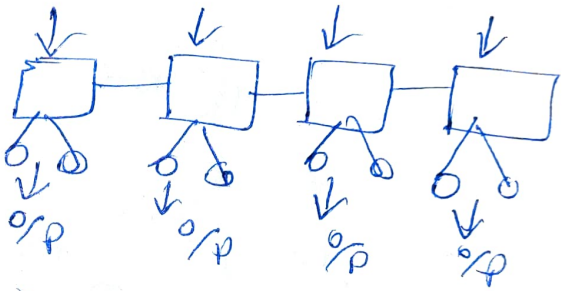
New weight — Is it = 1.

0.05 ÷ 0.649
0.05
0.05
0.05
0.05 → 0.349
0.05
0.05
0.05
0.649

Normalised weight:
0.07
0.07
0.07
0.537
0.07
0.07
0.07

Buckets:
[0 — 0.07]
[0.07 — 0.14]
[0.14 — 0.21]
[0.7 — 0.749]

0.649




Regression = Avg.
Classification = Majority.

~~(4) Xgboost~~

## Black box vs White Box Model

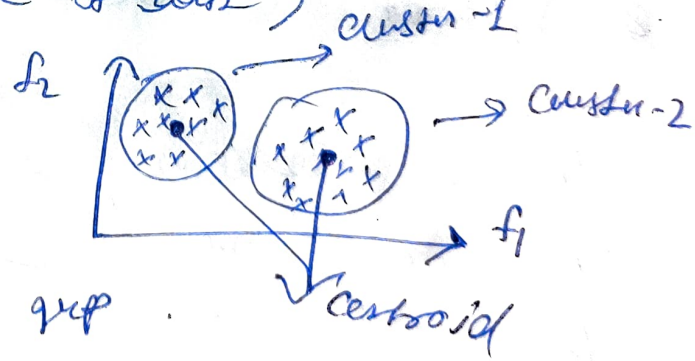
Linear Regression → White → We can visualize
Random forest → Black
Decision Tree → White
ANN → Black box.

## Day-6

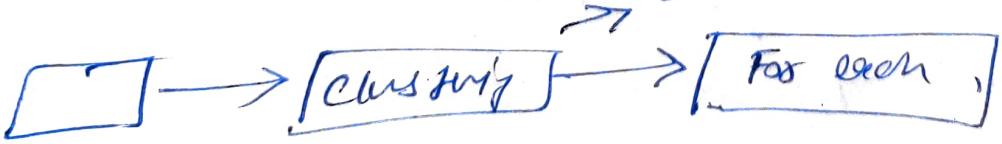
Unsupervised ML → Don't have specific o/p.

Clusters ↓ {similar kind of data}

1. K-means clustering
2. Hierarchical clustering
3. Silhouette score
4. DBScan clustering

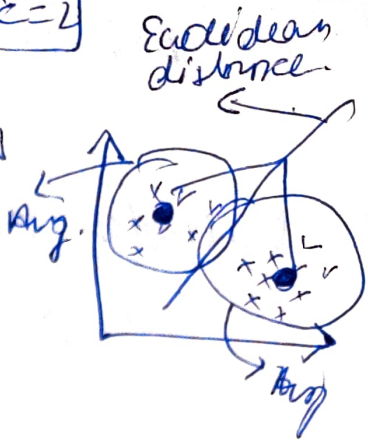



Cluster Ensemble Technique

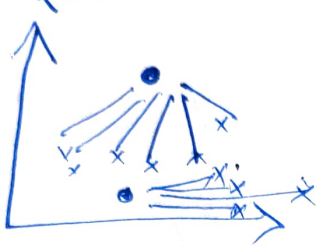


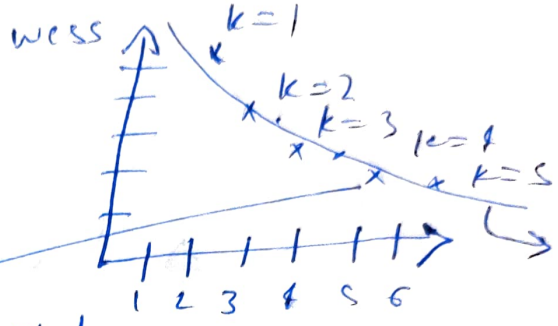
① We try k values ⇒ Suitable k=2
↳ Centroids.

② Initialize k no. of centroids
③ Find avg of those




Elbow Method (k value)

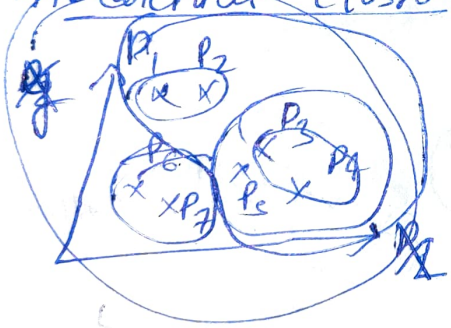


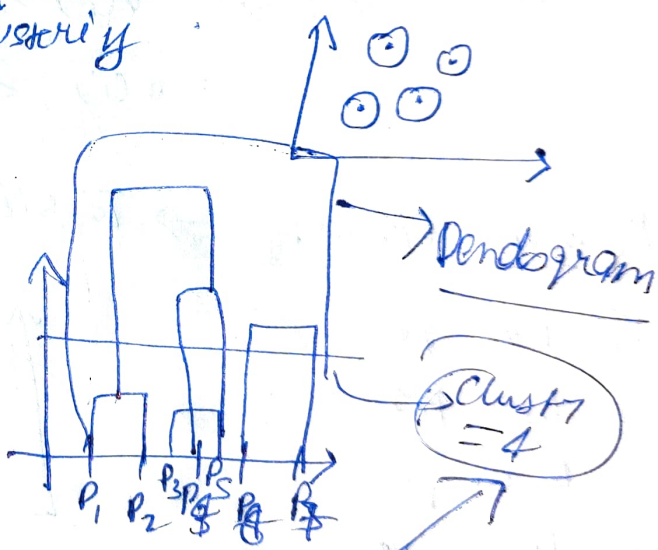



Here we find k value where mean is abrupt change.

All are the funda of k-mean clustering

② Hierarchical clustering




You need to find out longest vertical line that has no horizontal line passing through it.

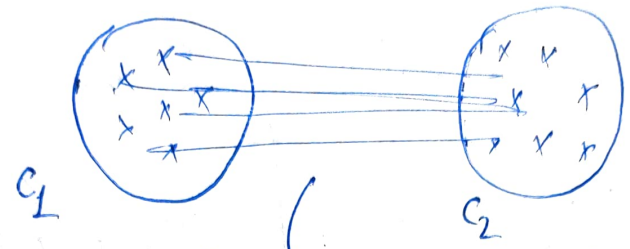
Max line is taken by → K-Mean or Hierarchical
↳ Coz it will keep of creating lots of dendogram.

Dataset is small → Hierarchical

" " large → K-Mean.

## Validate clustering models (- to +) → more good clustering model

↳ Silhouette (clustering).



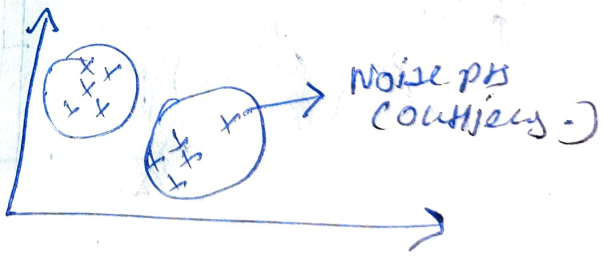

⇒ Average will be calculated.

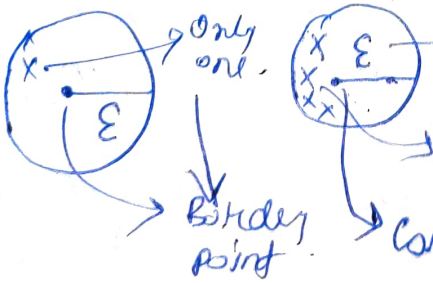
$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}, \text{ if } |C_i| > 1$$

Good cluster → $a(i) >> b(i)$ or $a(i) << b(i)$

## DB Scan Clustering → To skip outliers.

1. Min pts
2. Core pts
3. Border pts
4. Noise pts.



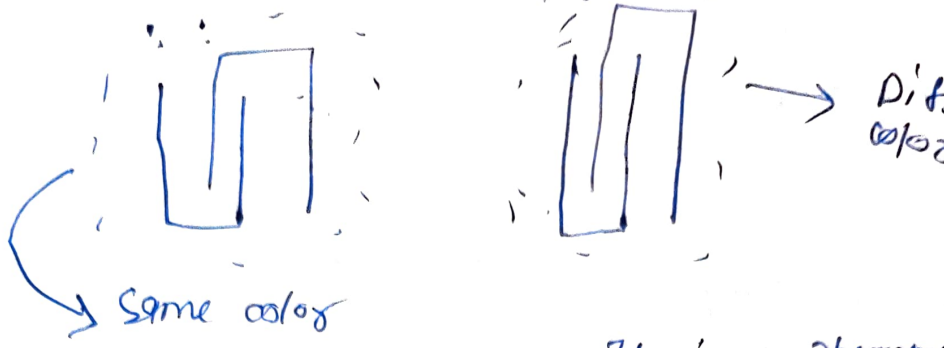

ε → Epsilon - Radius of the circle

Min pts = Hyperparameter

→ If no core or border pts then noise point will be neglected.

# K-Means Vs Hierarchial

Algo 3/56




## Bias & Variance

→ It is a phenomenon that skews the result of algorithm in favour or against the idea.

Train = 90%
Test = 10%
→ Overfitting
↓
Low Bias
High Variance.

Train data → Good – High Bias
→ Bad – Low Bias.

## Variance

Refers to the change in model when we'd different portions of the training or test data.

Test data → Good – Low Var.
→ Bad – High Var.

## SVM, SVR & Xgboost

① Xgboost Classifier

Extreme Gradient Boosting

Base Model → Pr = 0.5

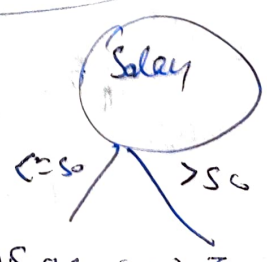
Dataset

| Salary | Condit | Approval | Residual |
|---|---|---|---|
| <=50 | B | 0 | -0.5 |
| <=50 | G | 1 | 0.5 |
| <=50 | G | 1 | 0.5 |
| >50 | B | 0 | -0.5 |
| >50 | G | 1 | 0.5 |
| >50 | N | 1 | 0.5 |
| <=50 | N | 0 | -0.5 |

① Calculate a Binary Decision Tree using the feature.

② Calculate Similarity weight

$$= \frac{\Sigma(\text{Residual})^2}{\Sigma(Pr(1-Pr)+\lambda)}$$




[-0.5, 0.5, 0.5, -0.5] [-0.5, 0.5, 0.5]

$$Sim_w = \frac{[-0.5+0.5+0.5-0.5)2}{0.5(1-0.5)+0.5(1-0.5)+0.5(1-0.5)(0.5)(1-0.5)} = 0$$

$Sim_w = \frac{0.25}{1.75} = \frac{1}{7} = 0.14$

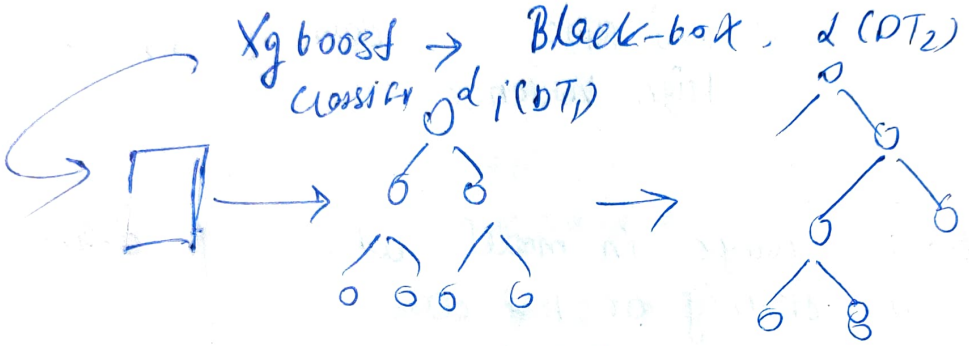
③ Information Gain

$= 0 + 0.3 - 0.14$

$= 0.19$

We basically compare on basis of information gain.

$$O/P \rightarrow \left[ \sigma \left[ 0 + \alpha_1(DT_1) + \alpha_2(DT_2) + \alpha_3(DT_3) + \cdots \alpha_q(DT_q) + \cdots \alpha_n(DT_n) \right] \right]$$

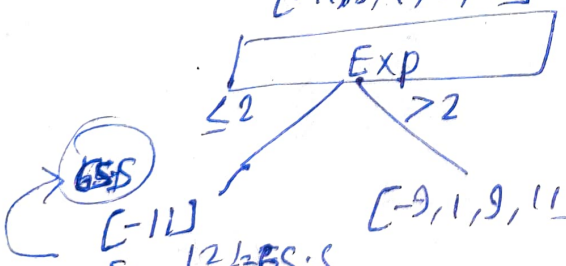



Xg-boost Regressor

| Exp | Crap | Salary → O/P | $R_1$ |
|---|---|---|---|
| 2 | M | 40k | -11k |
| 2.5 | Y | 48k | -9 |
| 3 | N | 52k | 1 |
| 4 | N | 60k | 9 |
| 4.5 | Y | 62k | 11 |

→ [Base model] → 51k

[-11, -9, 1, 9, 11] → Sm = 1/6.




$= \frac{12}{1+1} = 6.5$

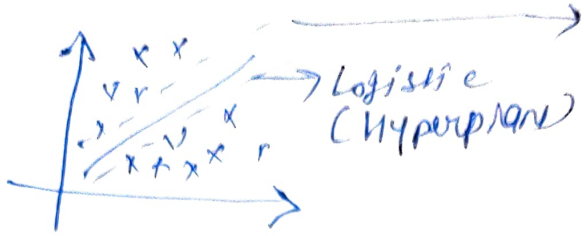
$= \frac{(-9+1+9+11)^2}{4+1}$

Sim weight $= \frac{\Sigma(\text{Residual})^2}{\text{No. of residuals}+1} = \frac{144}{5} = 28.8$

Inform gain $= 6.5 + 28.8 - \frac{1}{6}$

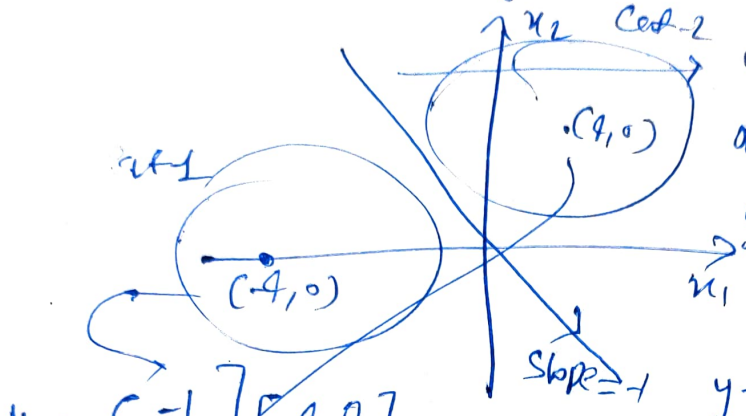
$= 98.34$

SVM → Similar like logistic regression




→ Marginal plane

→ In SVM we are going to increase marginal plane.




$y = mx + c$ → Slope, Intercept

$ax + by + c = 0$ → Eq of st line

$y = -\frac{c}{b} - \frac{ax}{b}$

$m = -a/b$, $c = -c/b$

$y = mx + c$

$y = w_1x + w_2x_2 + \dots - w_nx_n$

$\boxed{y = w^T x + b}$

$y = \begin{bmatrix} -1 \\ 0 \end{bmatrix} [-4, 0]$

$= 4 \Rightarrow$ +ve value

$y = \begin{bmatrix} -1 \\ 0 \end{bmatrix} [4, 4] = -4 + 0 = -4$

How marginal plane is created :—