$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots$ ① – Log / Linear

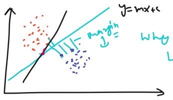
$Y = f(x)$ – NB / Knn – non-parametric model

Parameters v/s hyperparameter

- Parameters → Estimated / Calc. → during model building
- hyperparameter → settings of the model

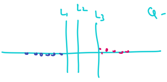
KNN –
- ✗ K = 3 – 3 neighbour – Accuracy – 70%
- ✓ K = 5 = 5 Neigh. – Acc. – 80% ✓
- ✗ K = 10 = 10 N. – Accuracy – 60%

SVM ⇒



Why green??
- ① getting 100% accuracy
- ②

Maximum margin classifier

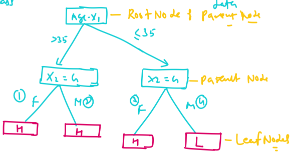


Q – which line is best separator??

Decision Tree ⇒ it comes up with tree on our dataset so that we can take decisions

Age X_1	Gender X_2	Y - Income class
30	F	H
35	M	L
30	M	H
40	F	H
37	M	?? H
35	M	L

Rule engine built on historical data

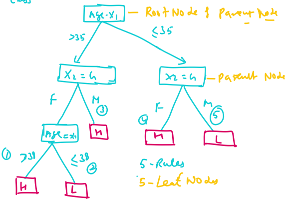


4 – Rules

Depth of the tree – 2

Split Methods – ??
- Gini Score
  - variable
  - Cutoff
- Entropy

class



5 – Rules

5 – Leaf Nodes

Depth of the tree – 3

Subtree – 3

Random forest ⇒

total – 52 cards

$P(R_1) = \frac{26}{52} = 0.5 = \frac{1}{2}$ → ①

$P(R_2|R_1) = \frac{25}{51}$ – without putting the card back to deck

I have put the card back to the deck

$$P(R_2|R_1) = \frac{26}{52} = 0.5$$

Apply Sampling – N = 1000

$1/1000 = n_1 \in N = 10\% = 100 = 900$ left

$1/1000 \quad n_2 \in N = 10\% = 100$

⋮

$1/1000 \quad n_{10} \in N = 10\% = 100$

Sampling with replacement
- Bootstrap Sampling

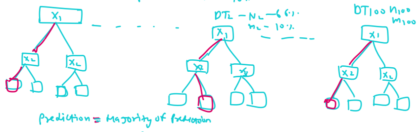
Decision Tree – When the dataset is small – they work well using h/p tuning
- they overfit – they are greedy algo.

N = 1MN] – DT – too much time to train – inefficient
(Xs) m = 100

RF ⇒ [N = 1MN, m = 100] – Random Sample from N – DT_{100}; Random Sample from M – DT_{100}
- Independent DT trees in parallel

N = 1MN
m = 100

DT_1 n_1 – Bootstrap Sample from N = 66%
m – Sampled variable – 10%



Prediction = Majority of prediction

DT_1 – H DT_3 = H
DT_2 = L DT_{100} = L

60% – DT – H
40% – DT – L

N = 10 total data points

DT_1	DT_2	DT_3
$10 = n_1 \in N$, m_1	$10 = n_2 \in N$, m_2	$10 = n_3 \in N$, m_3

P(single point) = 1/10

$P(2^{nd}\ p) = 1/10$

2/3 1/3

$10 = n_1 \sim 64\% = 6.4$ – 3.6 records (repeated)

$10 = n_2 \sim 64\% = 6.4$ – 3.6

$10 = n_3 \sim 64\% = 6.4$ – 3.6