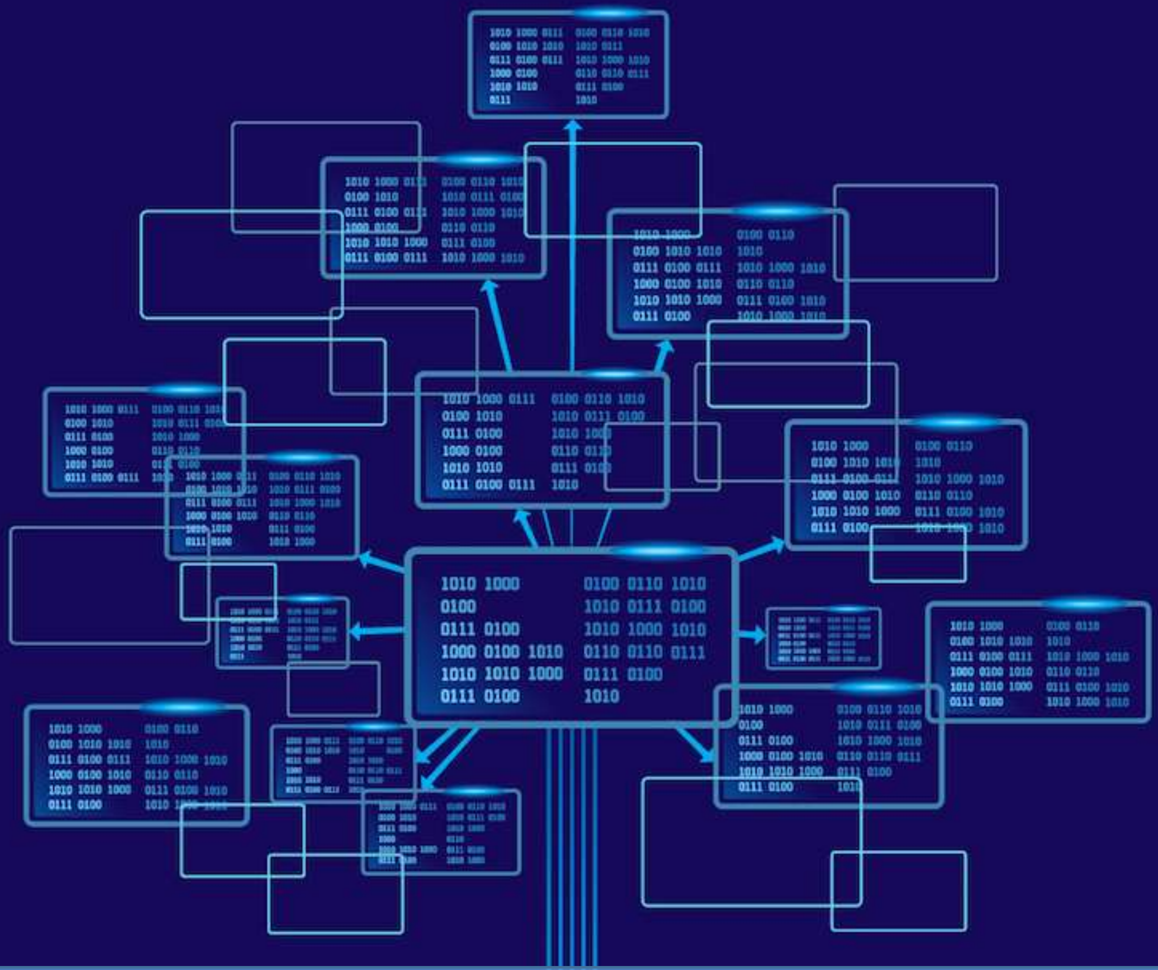
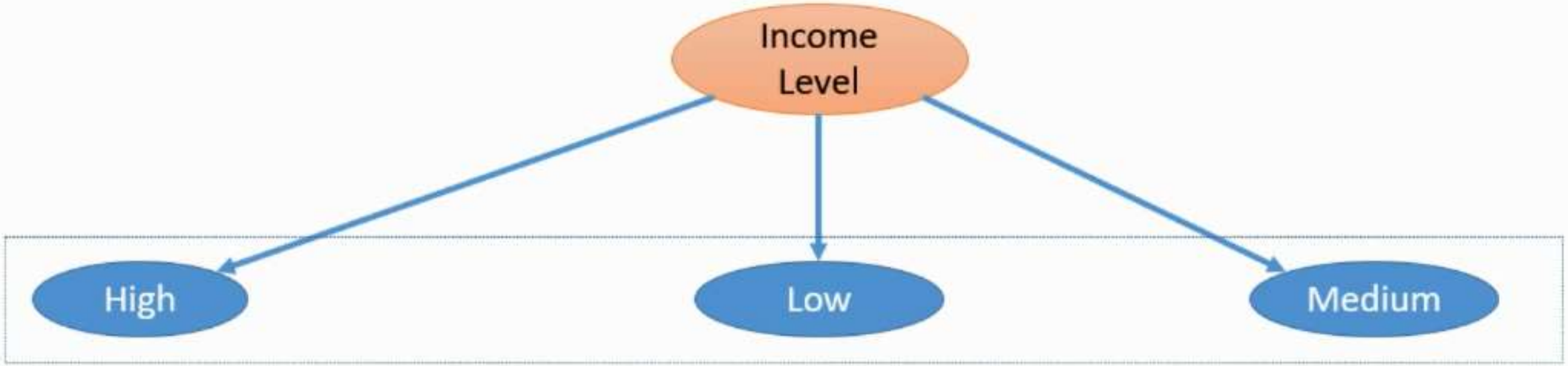


Decision Tree Algorithm

What is Decision Tree?

- Supervised learning method
- Decision support tool that uses a tree-like graph or model of decisions and their possible consequences
- Various variations such as Boosted Decision Tree, Random Forest
- Can be used for categorical as well as continuous variables

Loan ID	Income Level	Credit Score	Employment	Approved?
L1	Medium	Low	Self-Employed	No
L2	High	Low	Self-Employed	Yes
L3	High	High	Salaried	Yes
L4	Medium	Low	Salaried	Yes
L5	Low	High	Salaried	No
L6	Low	Low	Self-Employed	No
L7	High	Low	Salaried	Yes
L8	Medium	Low	Self-Employed	No
L9	High	High	Self-Employed	Yes
L10	Medium	High	Self-Employed	Yes
L11	High	Low	Salaried	Yes
L12	Medium	High	Salaried	Yes
L13	Medium	High	Self-Employed	Yes
L14	Low	Low	Self-Employed	No
L15	Low	High	Self-Employed	No
L16	**Medium**	**High**	**Salaried**	**???**



LID	IL	CS	ET	Status
L2	High	Low	SE	Yes
L3	High	High	Salaried	Yes
L7	High	Low	Salaried	Yes
L9	High	High	SE	Yes
L11	High	Low	Salaried	Yes

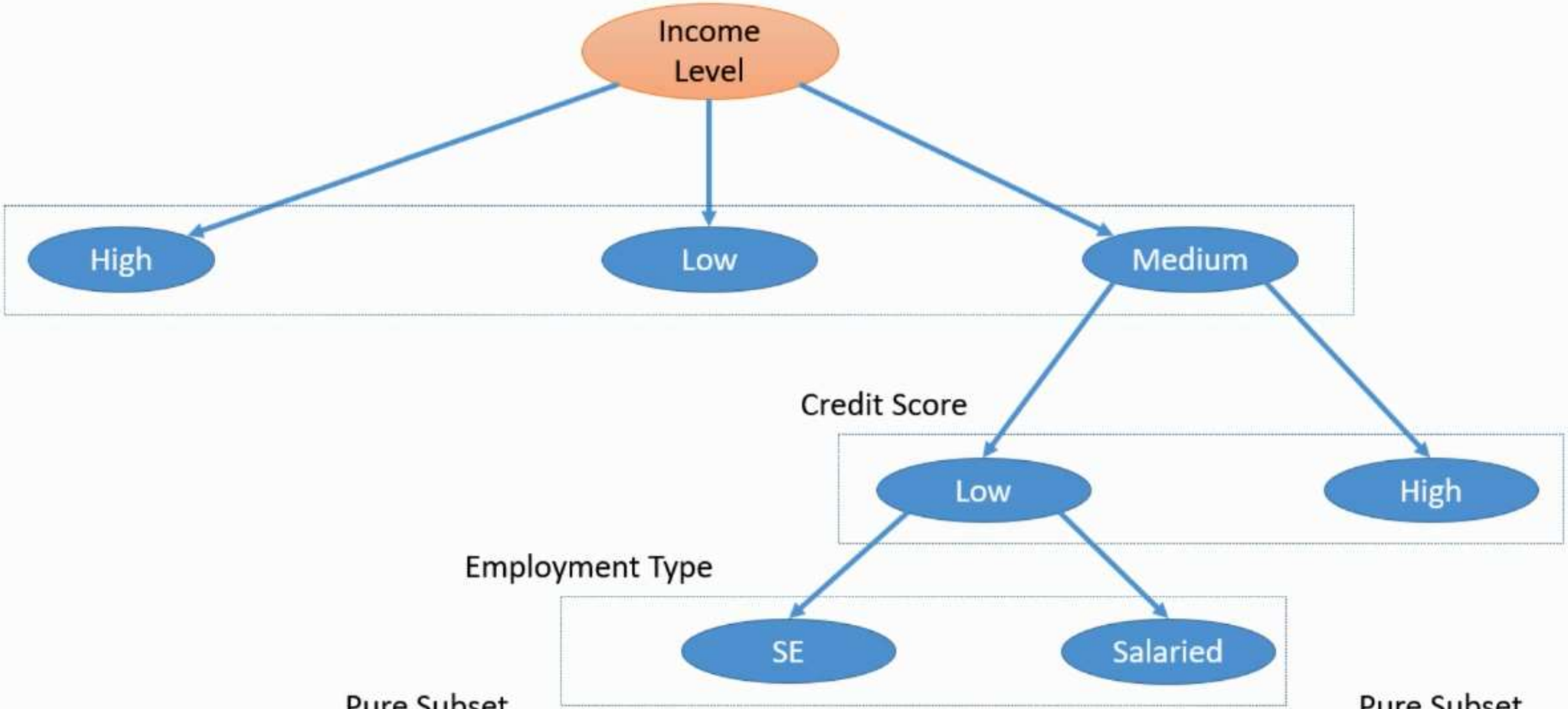
Pure Subset

LID	IL	CS	ET	Status
L5	Low	High	Salaried	No
L6	Low	Low	SE	No
L14	Low	Low	SE	No
L15	Low	High	SE	No

Pure Subset

LID	IL	CS	ET	Status
L1	Medium	Low	SE	No
L4	Medium	Low	Salaried	Yes
L8	Medium	Low	SE	No
L10	Medium	High	SE	Yes
L12	Medium	High	Salaried	Yes
L13	Medium	High	SE	Yes

Split Further



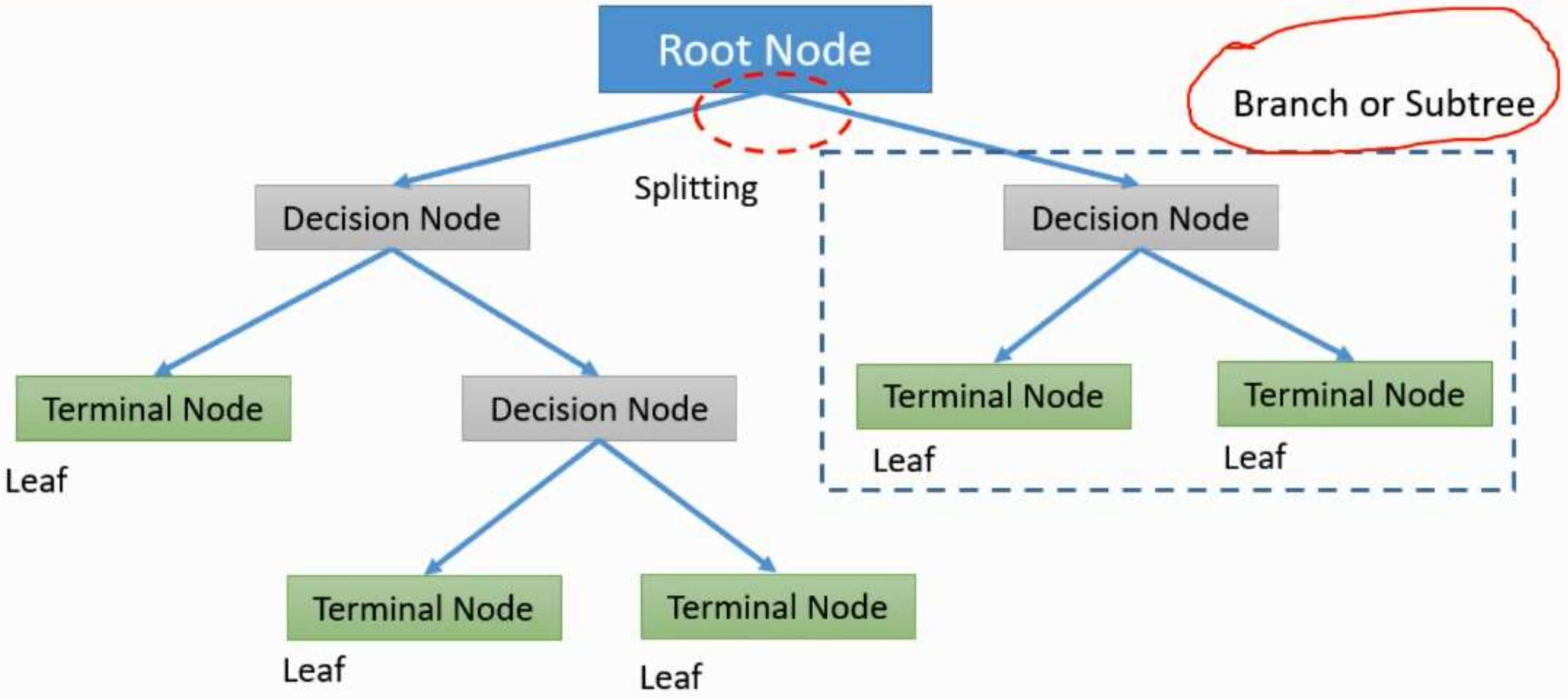
Pure Subset

LID	IL	CS	ET	Status
L1	Medium	Low	SE	No
L8	Medium	Low	SE	No

Pure Subset

LID	IL	CS	ET	Status
L4	Medium	Low	Salaried	Yes

Decision Tree Terms



Adult Income Dataset

age	wc	education	marital status	race	gender	hours per week	IncomeClass
38	Private	HS-grad	Divorced	White	Male	40	<=50K
28	Private	Bachelors	Married	Black	Female	40	<=50K
37	Private	Masters	Married	White	Female	40	<=50K
31	Private	Masters	Never-married	White	Female	50	>50K
42	Private	Bachelors	Married	White	Male	40	>50K

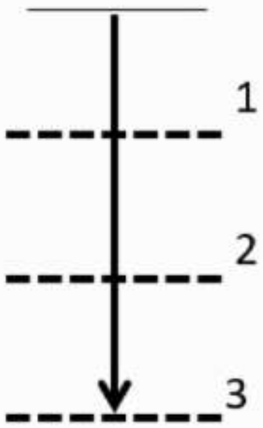
Prediction task is to determine whether a person makes over 50K a year.

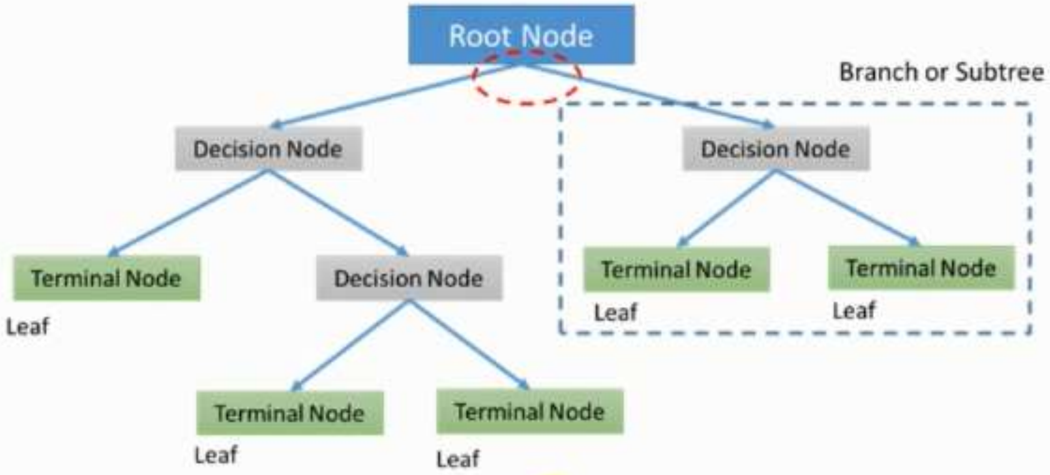
sklearn.tree.DecisionTreeClassifier – Parameters

- max_depth
- min_samples_split
- min_samples_leaf
- max_leaf_nodes
- splitter
- max_features
- criterion
- min_impurity_decrease

sklearn.tree.DecisionTreeClassifier – Parameters

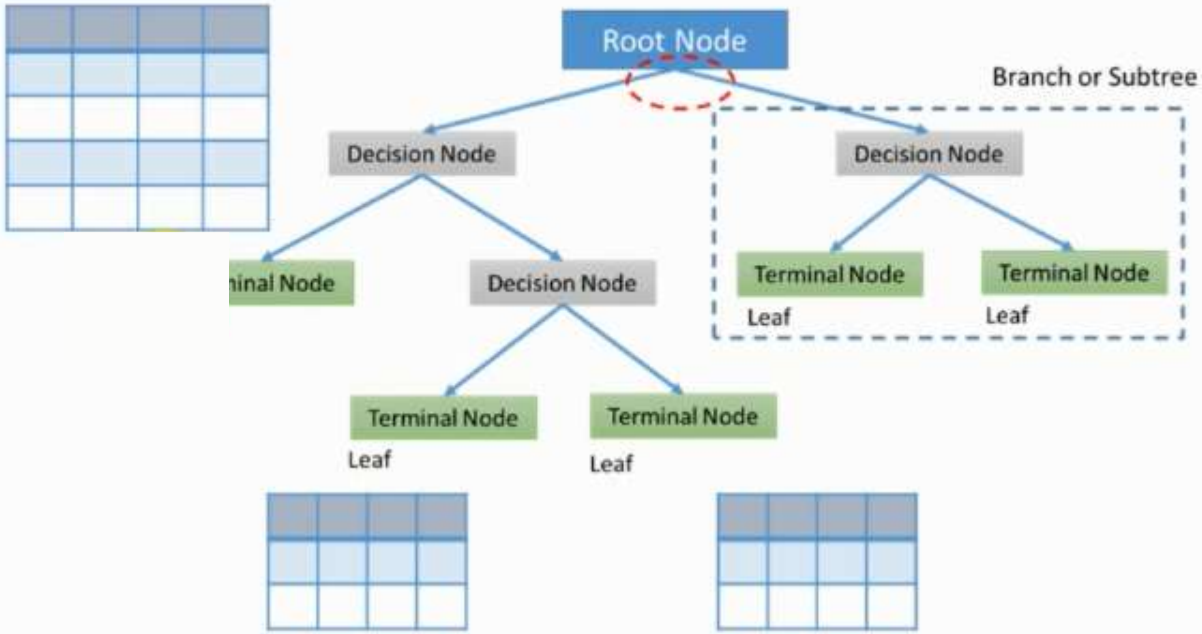
- **max_depth – max depth of the tree**
- min_samples_split
- min_samples_leaf
- max_leaf_nodes





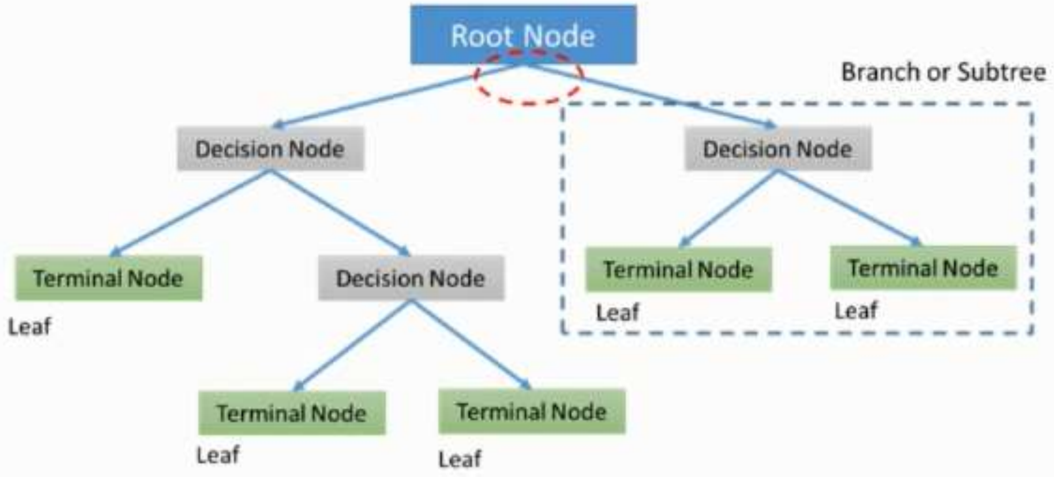
sklearn.tree.DecisionTreeClassifier – Parameters

- max_depth
- **min_samples_split – Min Samples required for the split**
- min_samples_leaf
- max_leaf_nodes



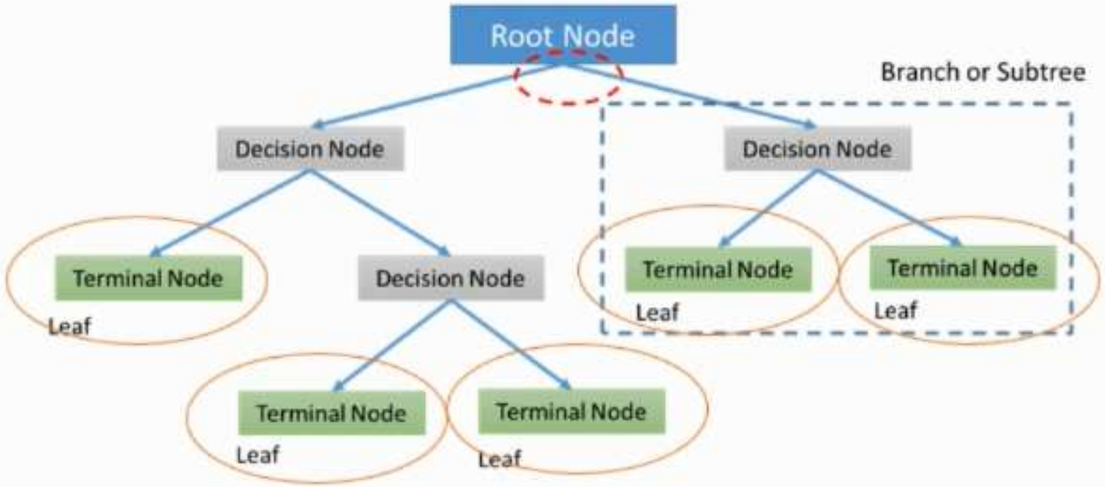
sklearn.tree.DecisionTreeClassifier – Parameters

- max_depth
- min_samples_split
- **min_samples_leaf - Min samples required at the leaf**
- max_leaf_nodes



sklearn.tree.DecisionTreeClassifier – Parameters

- max_depth
- min_samples_split
- min_samples_leaf
- **max_leaf_nodes - max number of leaf nodes**



sklearn.tree.DecisionTreeClassifier – Parameters

- max_depth
- min_sam[illegible]split
- [illegible]ples_leaf
- max_leaf_nodes

- splitter
- max_features

- criterion
- min_impurity_decrease

Loan ID	Income Level	Credit Score	Employment	Gender	Approve
L1	Medium	Low	Self-Employed	Male	No
L2	High	High	Self-Employed	Male	Yes
L3	High	High	Salaried	Female	Yes
L4	Medium	Low	Salaried	Male	Yes
L5	Low	Low	Salaried	Female	No
L6	Low	High	Self-Employed	Male	No
L7	High	High	Salaried	Male	Yes
L8	Medium	Low	Self-Employed	Female	No
L9	High	High	Self-Employed	Female	Yes
L10	Medium	High	Self-Employed	Female	Yes
L11	High	Low	Salaried	Male	No
L12	Medium	High	Salaried	Female	Yes
L13	Medium	High	Self-Employed	Male	Yes
L14	Low	Low	Self-Employed	Male	No
L15	Low	Low	Self-Employed	Female	No
L16	High	Low	Salaried	Female	No

Gender

Male: Yes = 4 No = 4

Gender	Approved?
Male	No
Male	Yes
Male	Yes
Male	No
Male	Yes
Male	No
Male	Yes
Male	No

50% 50%

Female: Yes = 4 No = 4

Gender	Approved?
Female	Yes
Female	No
Female	No
Female	Yes
Female	Yes
Female	Yes
Female	No
Female	No

50% 50%

Highly Impure

Credit History

Low: Yes = 1 No = 7

Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	No
Low	No
Low	No
Low	No
Low	No

12% 88%

High: Yes = 7 No = 1

Credit Score	Approved?
High	Yes
High	Yes
High	No
High	Yes
High	Yes
High	Yes
High	Yes
High	Yes

88% 12%

Less Impure

Gender

Male | Female

Yes = 4 No = 4

Gender	Approved?
Male	No
Male	Yes
Male	Yes
Male	No
Male	Yes
Male	No
Male	Yes
Male	No

50%	50%

Yes = 4 No = 4

Gender	Approved?
Female	Yes
Female	No
[illegible]	No
[illegible]	Yes
[illegible]	Yes
[illegible]	Yes
Female	No
Female	No

50%	50%

Highly Impure

Credit History

Low | High

Yes = 1 No = 7

Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	[illegible]
Low	[illegible]
Low	No
Low	No
Low	No

12%	88%

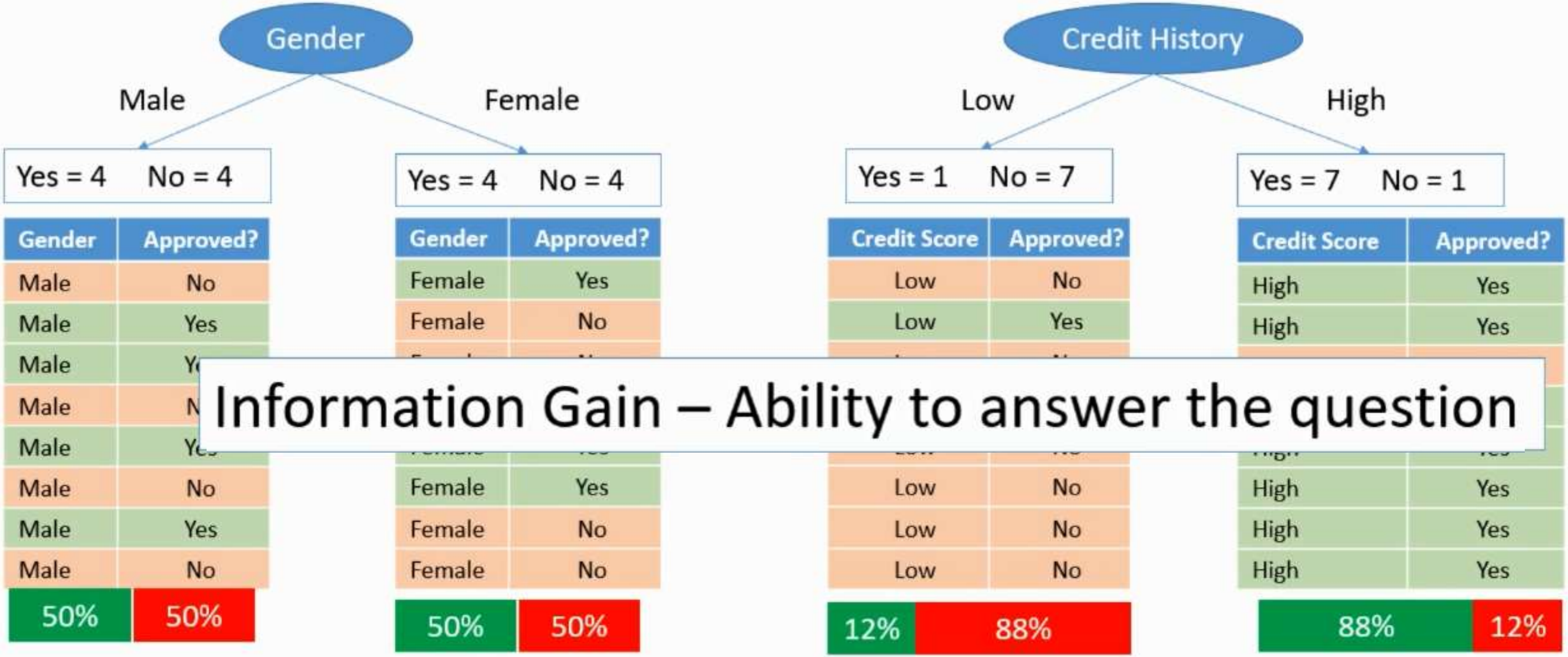
Yes = 7 No = 1

Credit Score	Approved?
High	Yes
High	Yes
High	No
[illegible]	Yes
[illegible]	Yes
[illegible]	Yes
High	Yes
High	Yes

88%	12%

Less Impure

Information Gain – Ability to answer the question



Gender

Male: Yes = 4 No = 4

Gender	Approved?
Male	No
Male	Yes
Male	[illegible]
Male	[illegible]
Male	[illegible]
Male	No
Male	Yes
Male	No

50% 50%

Female: Yes = 4 No = 4

Gender	Approved?
Female	Yes
Female	No
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
Female	Yes
Female	No
Female	No

50% 50%

Best Split

Credit History

Low: Yes = 1 No = 7

Credit Score	Approved?
Low	No
Low	Yes
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
Low	No
Low	No
Low	No

12% 88%

High: Yes = 7 No = 1

Credit Score	Approved?
High	Yes
High	Yes
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
High	Yes
High	Yes
High	Yes

88% 12%

Information Gain – Ability to answer the question

Best Split → Credit History

Low

Yes = 1 No = 7

Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	No
Low	No
Low	No
Low	No
Low	No

12% 88%

High

Yes = 7 No = 1

Credit Score	Approved?
High	Yes
High	Yes
High	No
High	Yes
High	Yes
High	Yes
High	Yes
High	Yes

88% 12%

Parameters of Decision Tree relation to splitting

- **splitter – Split strategy for Best feature or Random feature**

- max_features

Best Split → Credit History

Low | High

Yes = 1 No = 7 | Yes = 7 No = 1

Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	No
Low	No
Low	No
Low	No
Low	No

12% 88%

Credit Score	Approved?
High	Yes
High	Yes
High	No
High	Yes
High	Yes
High	Yes
High	Yes
High	Yes

88% 12%

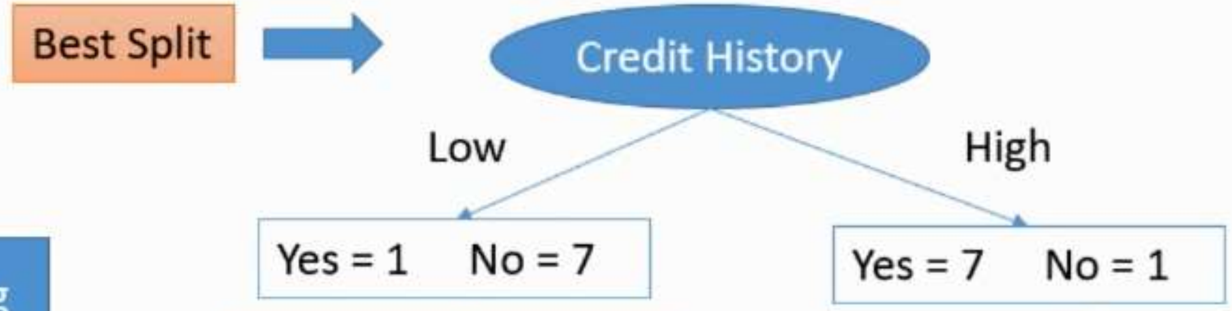
Parameters of Decision Tree relation to splitting

- splitter – Split strategy for Best feature or Random feature

- **max_features – Number of features to search before Best splitter is found**

Parameters of Decision Tree relation to splitting

- splitter – Split strategy for Best feature or Random feature
- max_features – Number of features to search before Best splitter is found



Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	No
Low	No
Low	No
Low	No
Low	No

12% 88%

Credit Score	Approved?
High	Yes
High	Yes
High	No
High	Yes
High	Yes
High	Yes
High	Yes
High	Yes

88% 12%

How to decide which Feature has the Best Split?

Best Split → Credit History

What should be the criterion?

Credit History

Low

Yes = 1 No = 7

Credit Score	Approved?
Low	No
Low	Yes
Low	No
Low	No
Low	No
Low	No
Low	No
Low	No

12% 88%

High

Yes = 7 No = 1

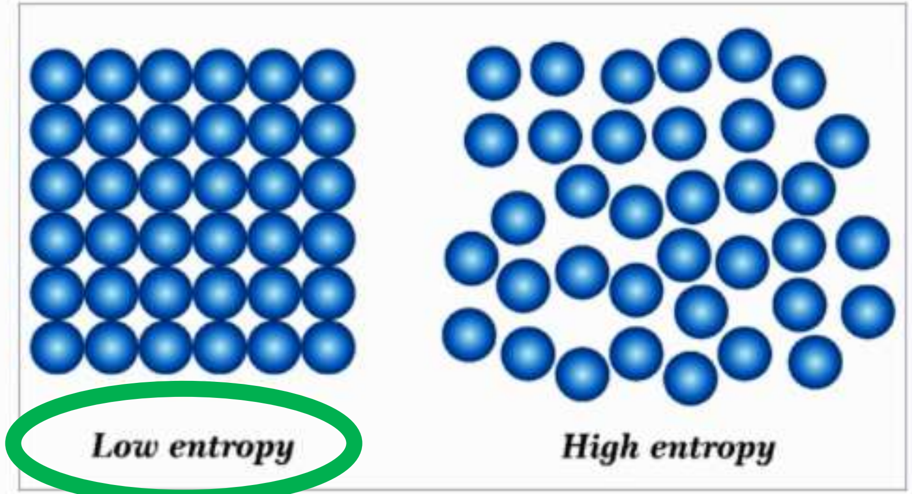
Credit Score	Approved?
High	Yes
High	Yes
High	No
High	Yes
High	Yes
High	Yes
High	Yes
High	Yes

88% 12%

How to decide which Feature has the Best Split?

Entropy

What should be the **criterion**?



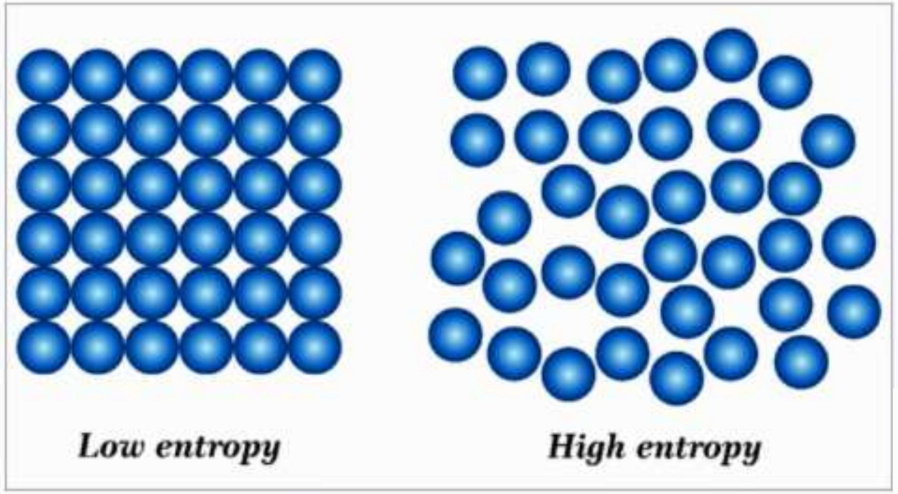
Our aim is to get lower entropy values

High entropy means impure data

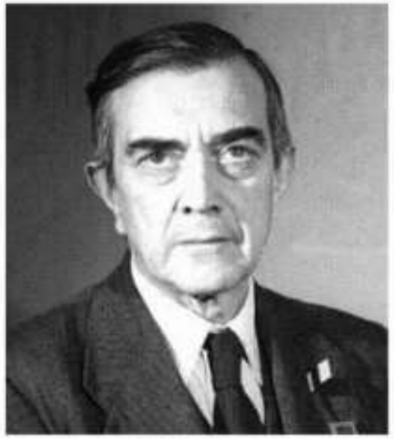
How to decide which Feature has the Best Split?

What should be the **criterion**?

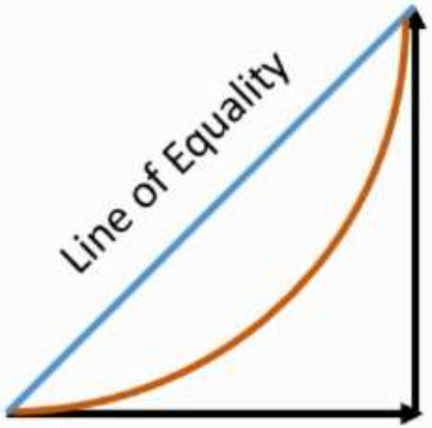
Entropy



Gini

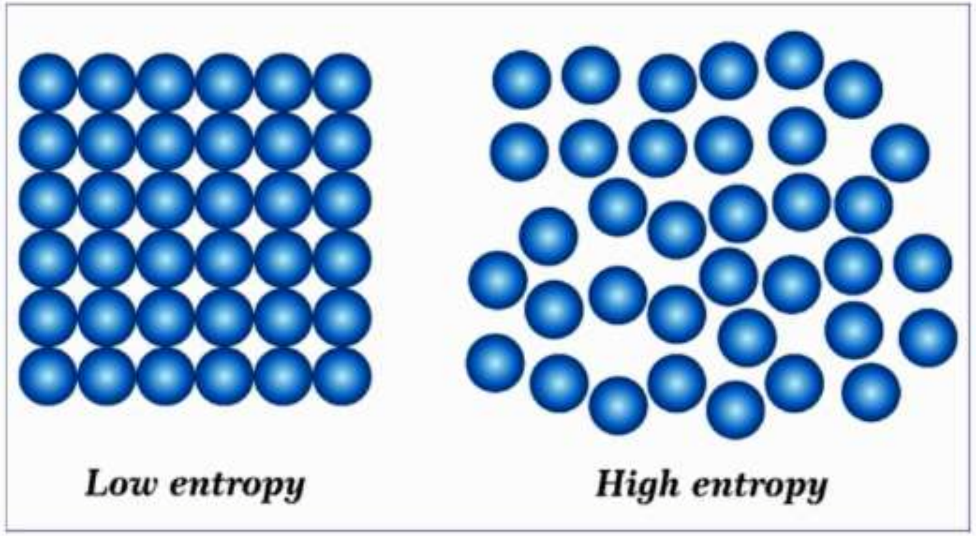


Corrado Gini

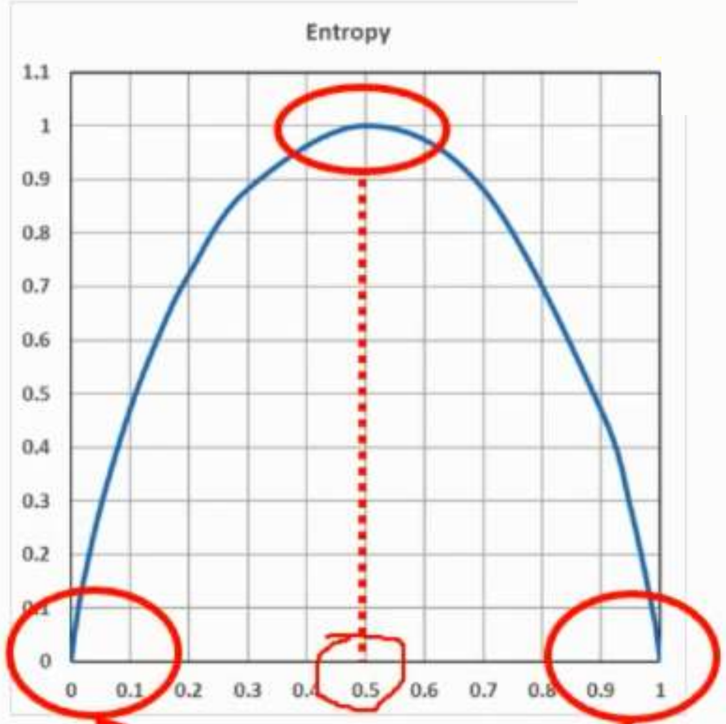


Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$



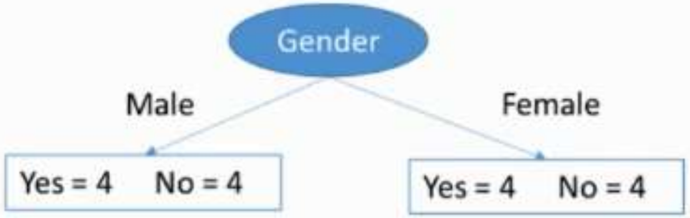
Highly Impure Classes

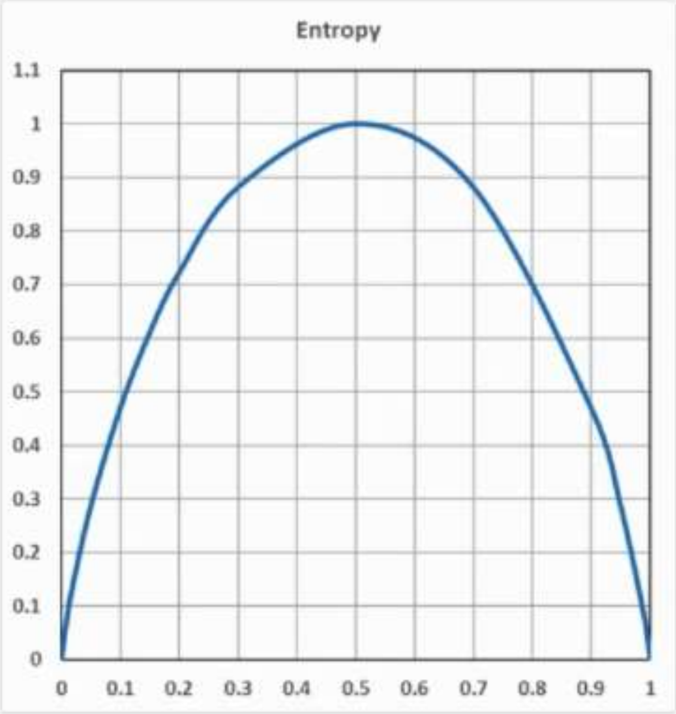


Pure Classes

Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$

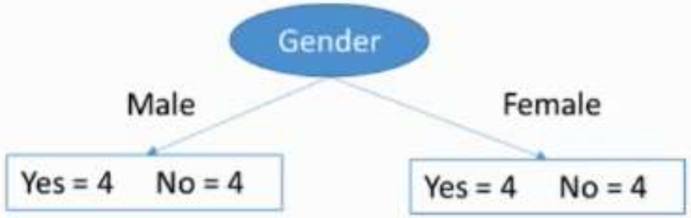




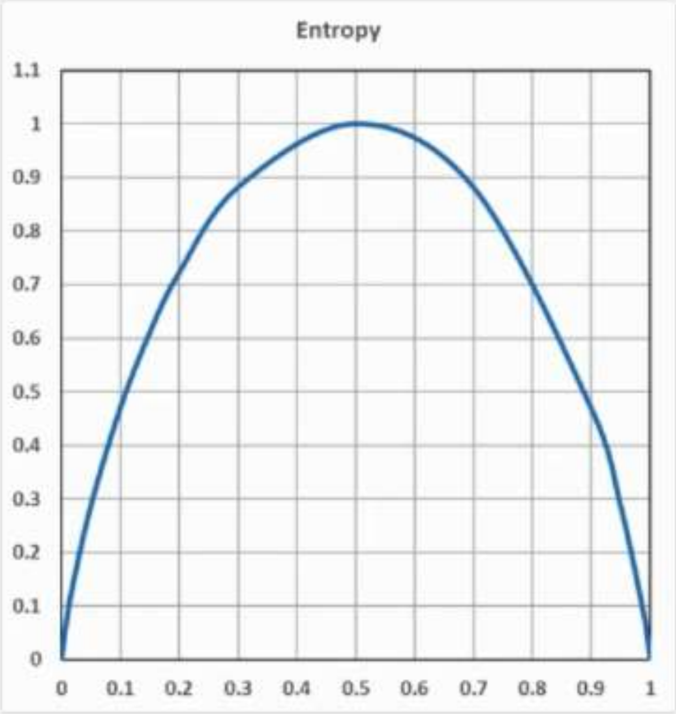
$$p_{Yes} = \frac{4}{4+4} = 0.5 \qquad p_{No} = \frac{4}{4+4} = 0.5$$

Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$

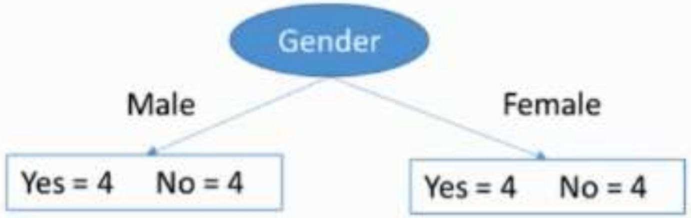


$$S = -1 * (0.5*\log_2 0.5 + 0.5*\log_2 0.5)$$



Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$

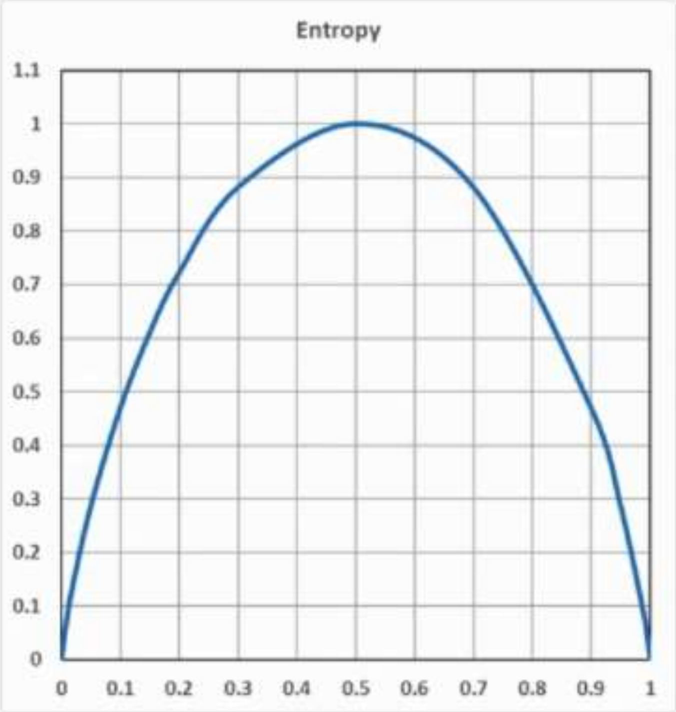


$S = -1 * (0.5*\log_2 0.5 + 0.5*\log_2 0.5)$

$= -1 * (0.5 * (-1) + 0.5 * (-1))$

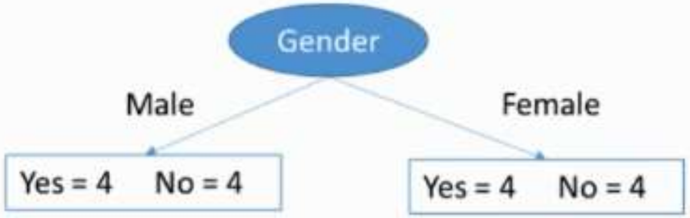
$= 0.5 + 0.5$

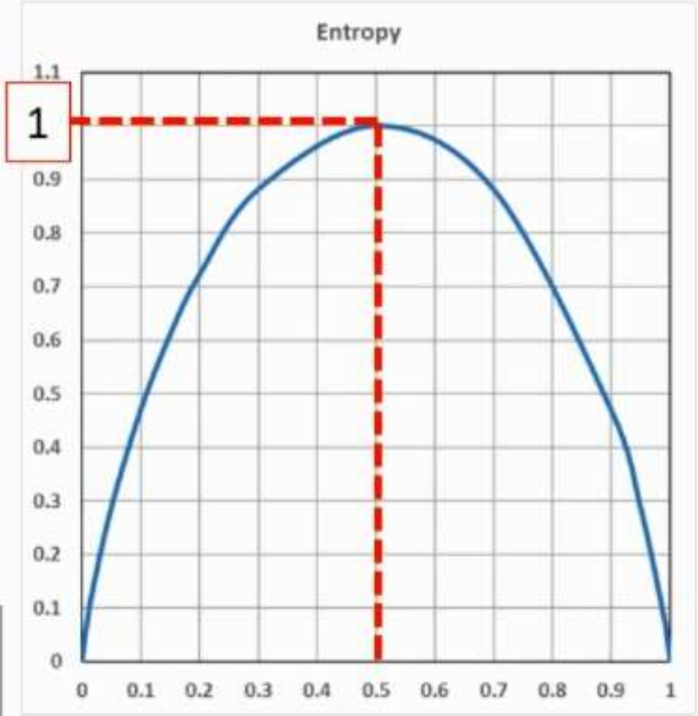
$= 1$



Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$



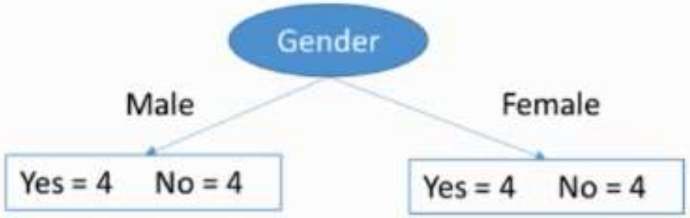


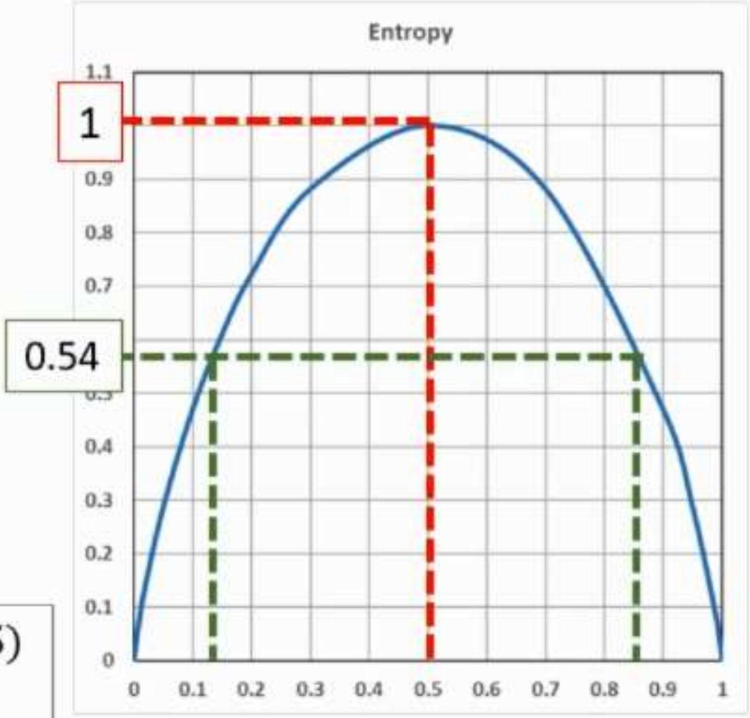
$$
\begin{aligned}
S &= -1 * (0.5 * \log_2 0.5 + 0.5 * \log_2 0.5) \\
&= -1 * (0.5 * (-1) + 0.5 * (-1)) \\
&= 0.5 + 0.5 \\
&= 1
\end{aligned}
$$

$$
\begin{aligned}
S &= -1 * (0.125 * \log_2 0.125 + 0.875 * \log_2 0.875) \\
&= -1 * (0.125 * (-3) + 0.875 * (-0.1926)) \\
&= -1 * (-0.375 + (-0.1685)) \\
&= 0.5435
\end{aligned}
$$

Entropy - Measure of Impurity

$$Entropy = -1 * \sum_{i=1}^{n} p_i \log_2 p_i$$





$$S = -1 * (0.5*\log_2 0.5 + 0.5*\log_2 0.5)$$
$$= -1 * (0.5 * (-1) + 0.5 * (-1))$$
$$= 0.5 + 0.5$$
$$= 1$$

$$S = -1 * (0.125*\log_2 0.125 + 0.875*\log_2 0.875)$$
$$= -1 * (0.125 * (-3) + 0.875 * (-0.1926))$$
$$= -1 * (-0.375 + (-0.1685))$$
$$= 0.5435$$

Gini

$$Gini = 1 - \sum_{i=1}^{n} p_i^2$$

Gender

Male: Yes = 4 No = 4

Female: Yes = 4 No = 4

$$\begin{aligned} Gini &= 1 - (0.5^2 + 0.5^2) \\ &= 1 - (0.25 + 0.25) \\ &= 0.5 \end{aligned}$$

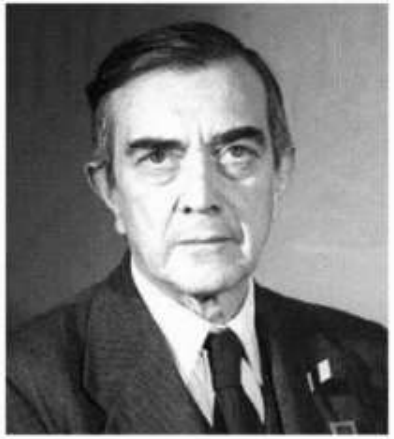
Credit History

Low: Yes = 1 No = 7

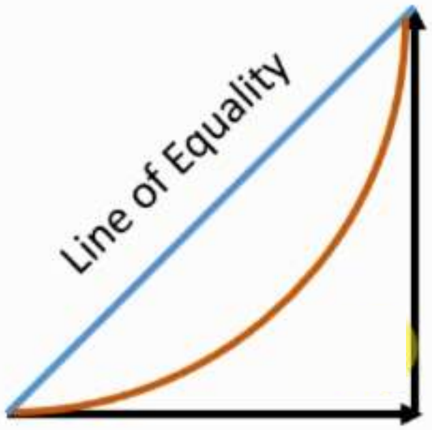
High: Yes = 7 No = 1

$$\begin{aligned} Gini &= 1 - (0.125^2 + 0.875^2) \\ &= 1 - (0.015 + 0.765) \\ &= 0.22 \end{aligned}$$

Less value of Gini is better



Corrado Gini



Demo: Create ML model using Decision tree to predict if customer will buy the product.

Demo: Create ML model using Decision tree to predict the income.

Ensemble Learning / Random Forest

Everyday Ensemble Learning

Decision?

Is this price fair?

Construction Quality?

Appreciation of price?

Location appropriate?

Neighbourhood?

Decision?

Broker or real estate portal to check fair price, price appreciation

Friend or colleague who stays nearby or stayed in the neighbourhood

Inspection by an architect for quality checks and structural defects.

Decision?

Is this price fair?

Appreciation of price?

Construction Quality?

Location appropriate?

Neighbourhood?

Decision?

Is this price fair?

Appreciation of price?

Construction Quality?

Majority

Weighted Average

Location appropriate?

Neighbourhood?

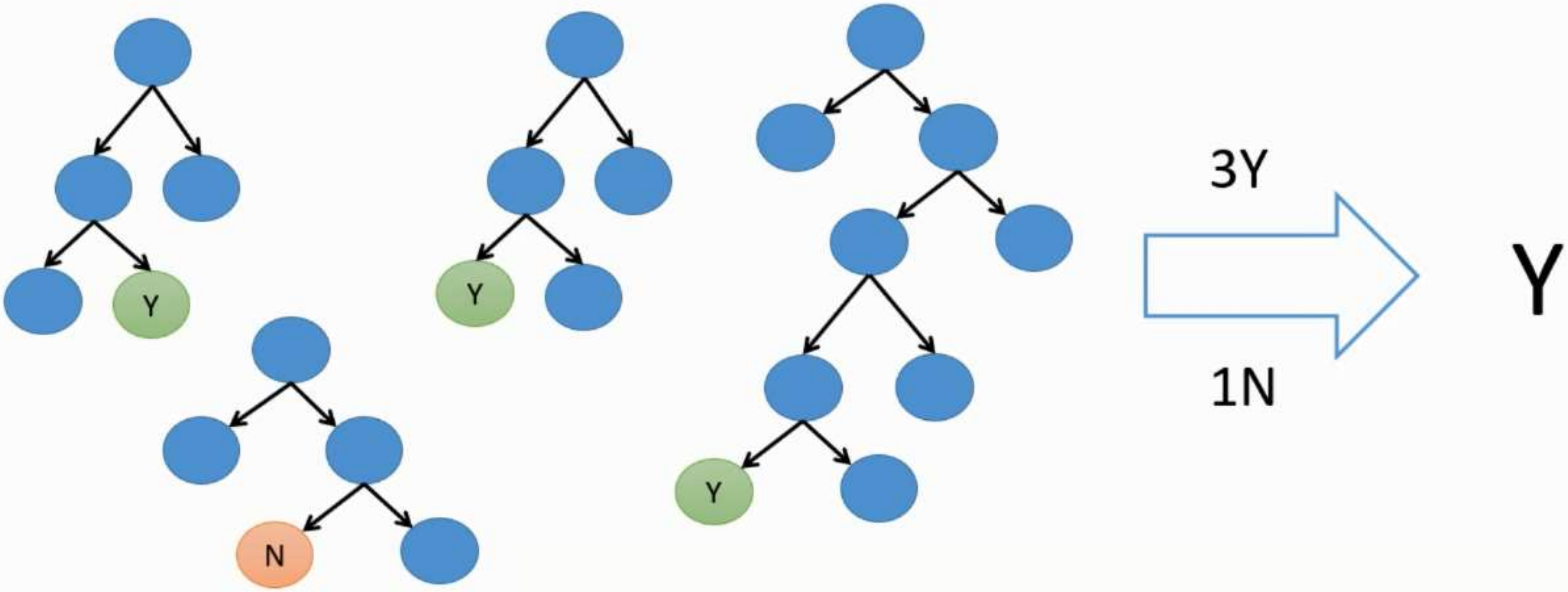
Ensemble Learning

- All algorithms have errors
- Collective wisdom is higher than the individual intelligence
- Generate a group of base learners and combined result gives higher accuracy
- Different base learners can use different,
  - Parameters
  - Sequence
  - Training sets etc
- Two major Ensemble Learning Methods
  - Bagging
  - Boosting

Bagging

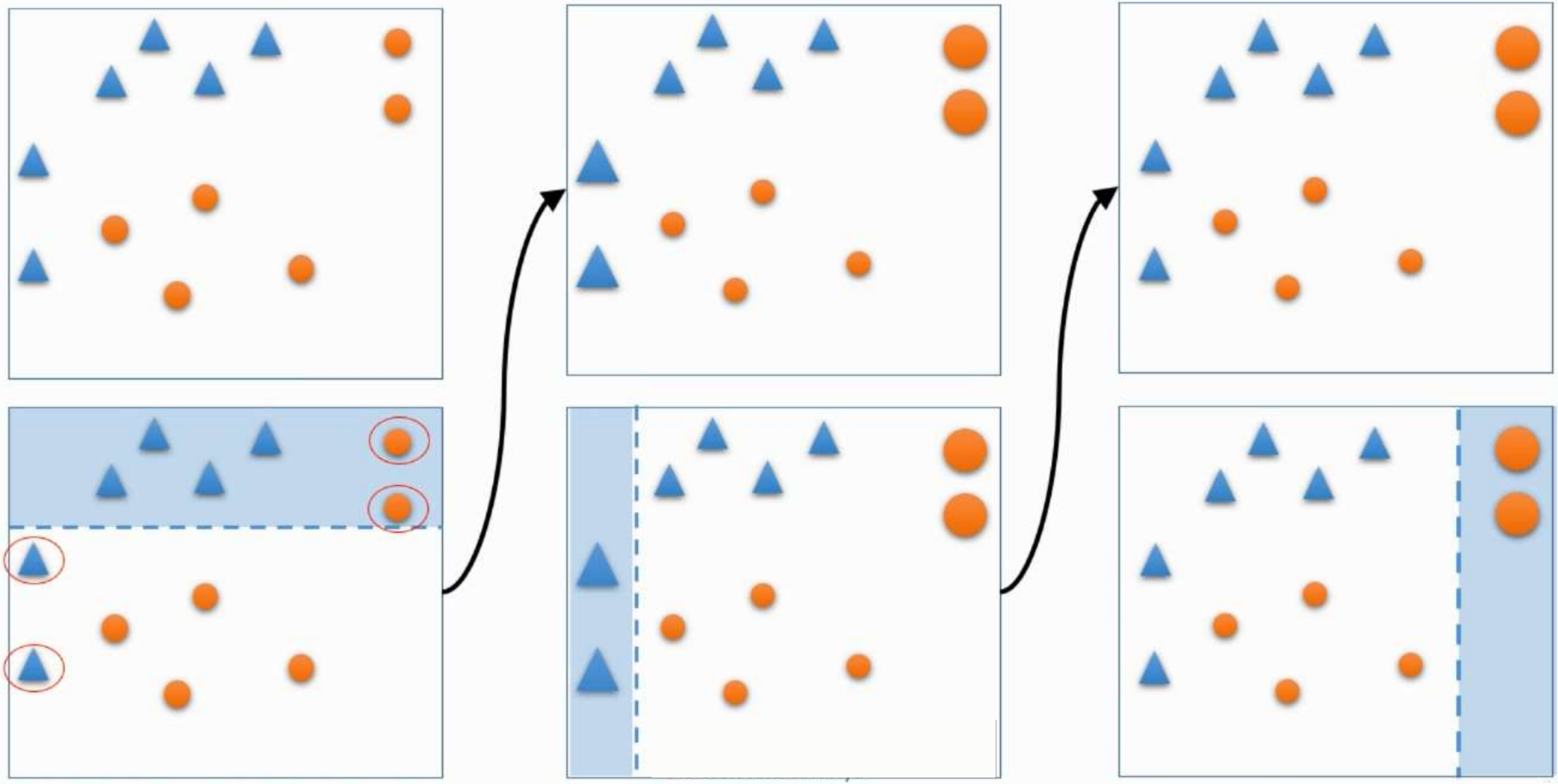
- Various models are built in parallel

- All models vote to give the final prediction

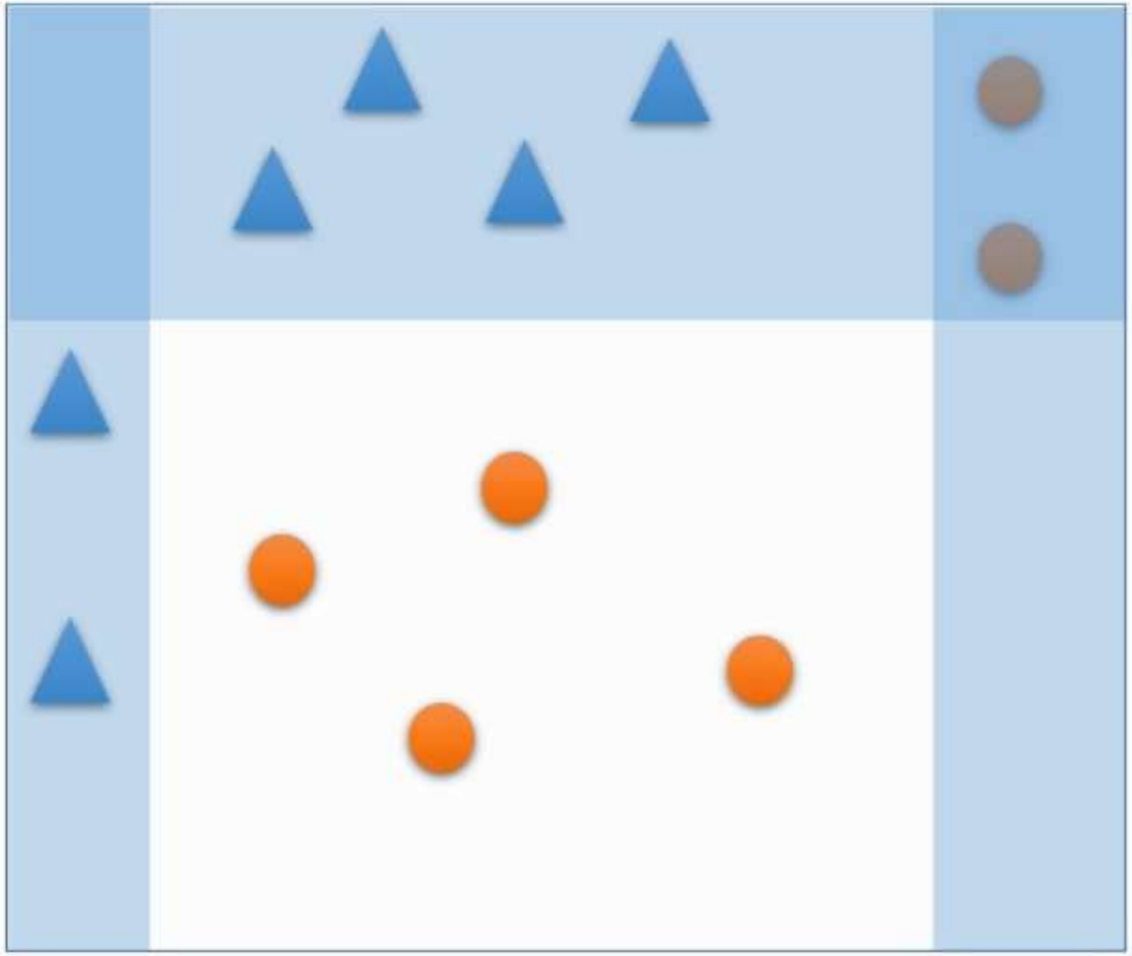


Boosting

- Train the Decision Tree in a sequence
- Learn from the previous tree by focussing on incorrect observations
- Build new model with higher weight for incorrect observations from previous sequence



Boosted Model



Demo: Create ML model using Random Forest to predict if customer will buy the product.

Demo: Create ML model using Random Forest to predict the income.