Decision Tree

Course Objectives

What is a decision tree model

How does a decision tree is being constructed?

Benefits of a decision tree model

Learning Outcomes

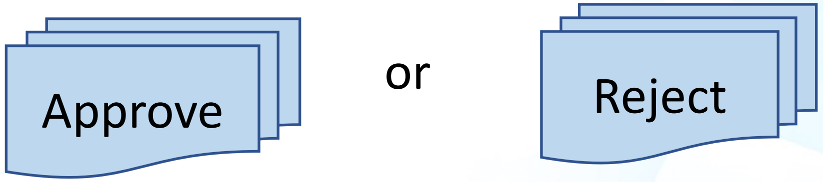
At the end of the course, you will be able to

- Understand what is a decision tree model
- Be familiar with how a decision tree is constructed
- Understand the benefits of a decision tree model

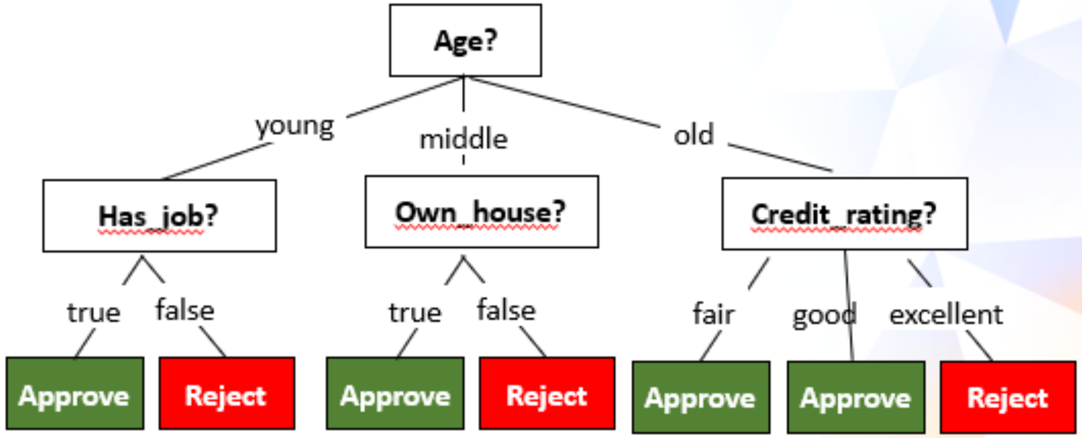
Classification Model – Decision Tree

- Another classification model

Loan Application



- As compared to other classification model such as KNN and Logistics Regression, it constructs a decision tree to assist in decision making



Example –
Bank Loan Application

Based on his profile, should I classify him as a potential loan defaulter?
If yes, I shall reject his application.

What is your age, job status, and credit history?
Do you own a house?

Bank Loan Example

ID	Age	Has_job	Own_house	Credit_rating	Outcome
1	young	false	false	fair	Reject
2	young	false	false	good	Reject
3	young	true	false	good	Approve
4	young	true	true	fair	Approve
5	young	false	false	fair	Reject
6	middle	false	false	fair	Reject
7	middle	false	false	good	Reject
8	middle	true	true	good	Approve
9	middle	false	true	excellent	Approve
10	middle	false	true	excellent	Approve
11	old	false	true	excellent	Approve
12	old	false	true	good	Approve
13	old	true	false	good	Approve
14	old	true	false	excellent	Approve
15	old	false	false	fair	Reject

features *label*

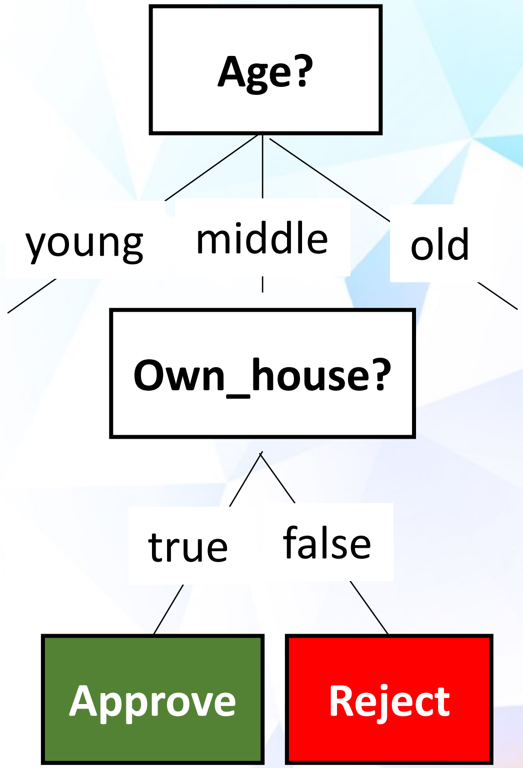
Outcome

Approve (non-defaulter)

Reject (defaulter)

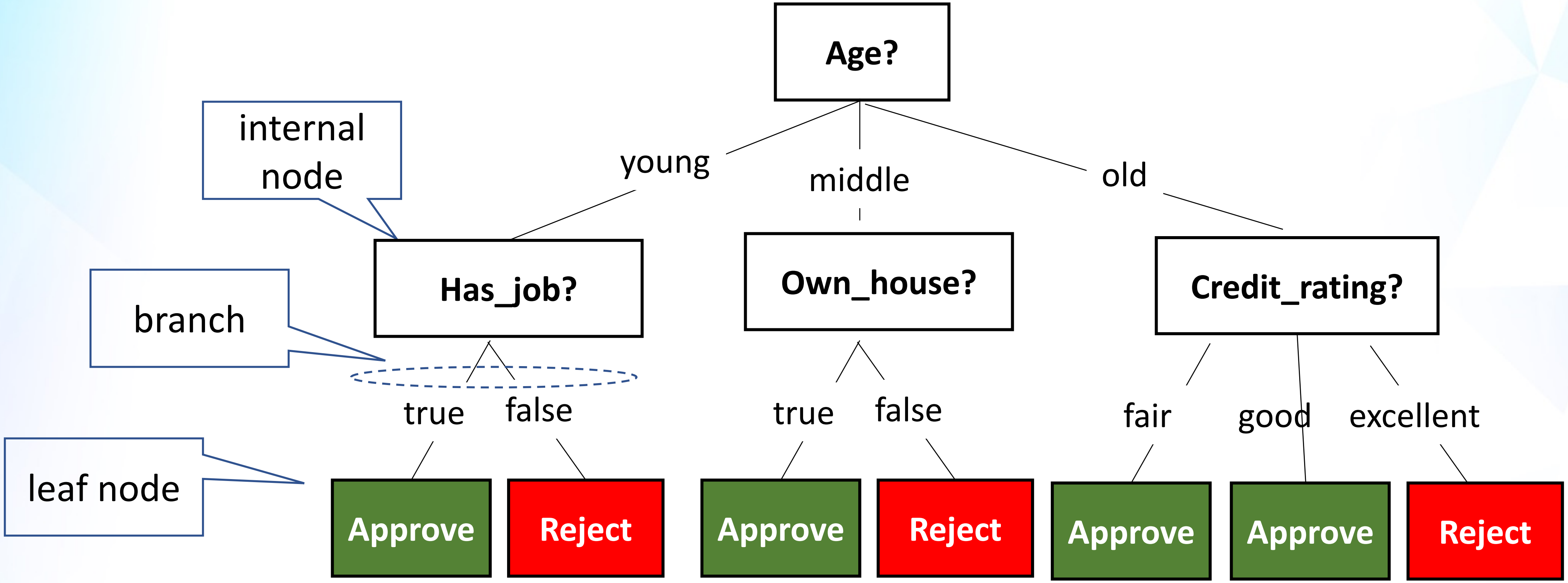
Finding patterns in data

ID	Age	Has_job	Own_house	Credit_rating	Outcome
1	young	false	false	fair	Reject
2	young	false	false	good	Reject
3	young	true	false	good	Approve
4	young	true	true	fair	Approve
5	young	false	false	fair	Reject
6	middle	false	false	fair	Reject
7	middle	false	false	good	Reject
8	middle	true	true	good	Approve
9	middle	false	true	excellent	Approve
10	middle	false	true	excellent	Approve
11	old	false	true	excellent	Approve
12	old	false	true	good	Approve
13	old	true	false	good	Approve
14	old	true	false	excellent	Approve
15	old	false	false	fair	Reject



Decision Tree

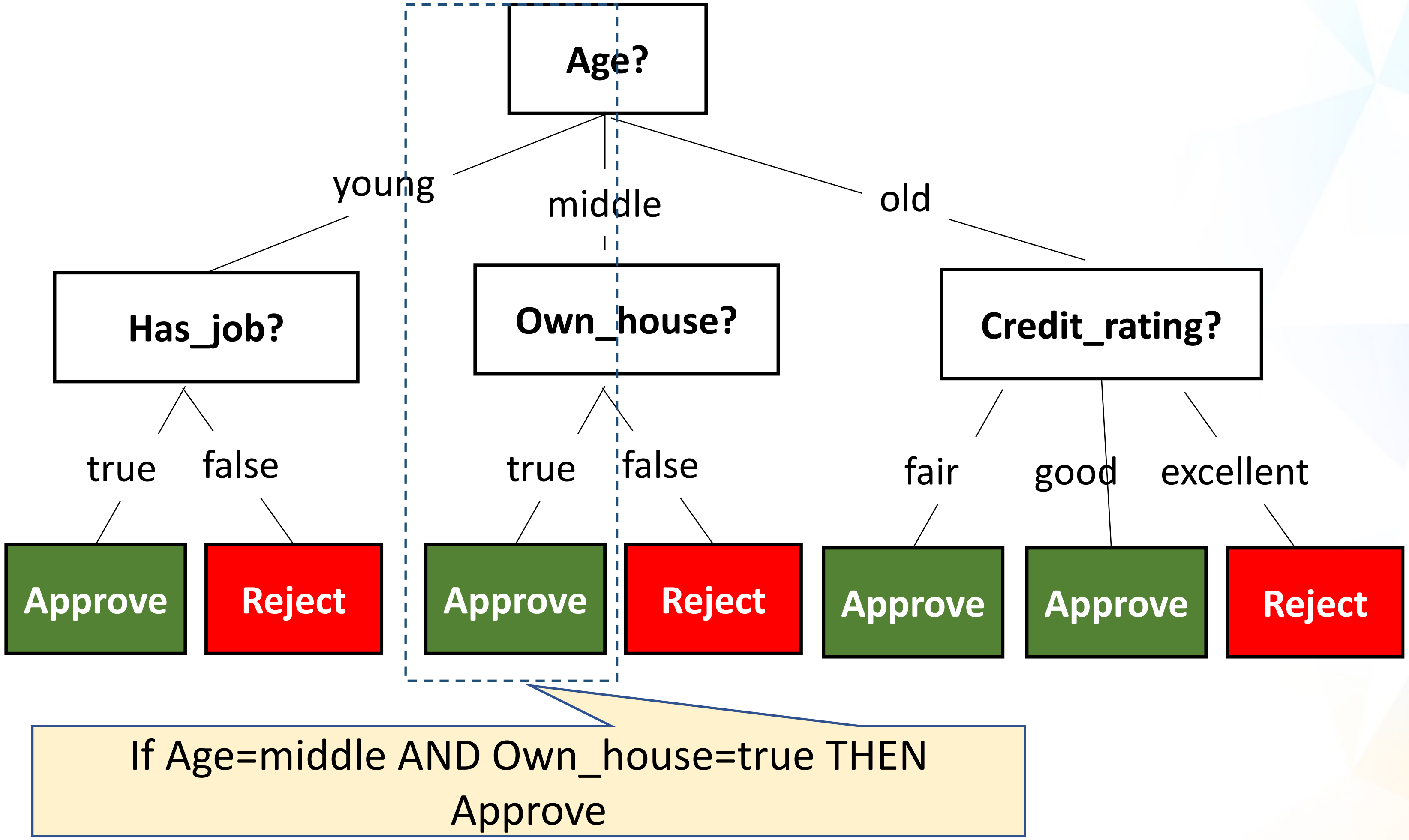
A decision tree is a flow-chart-like tree structure.



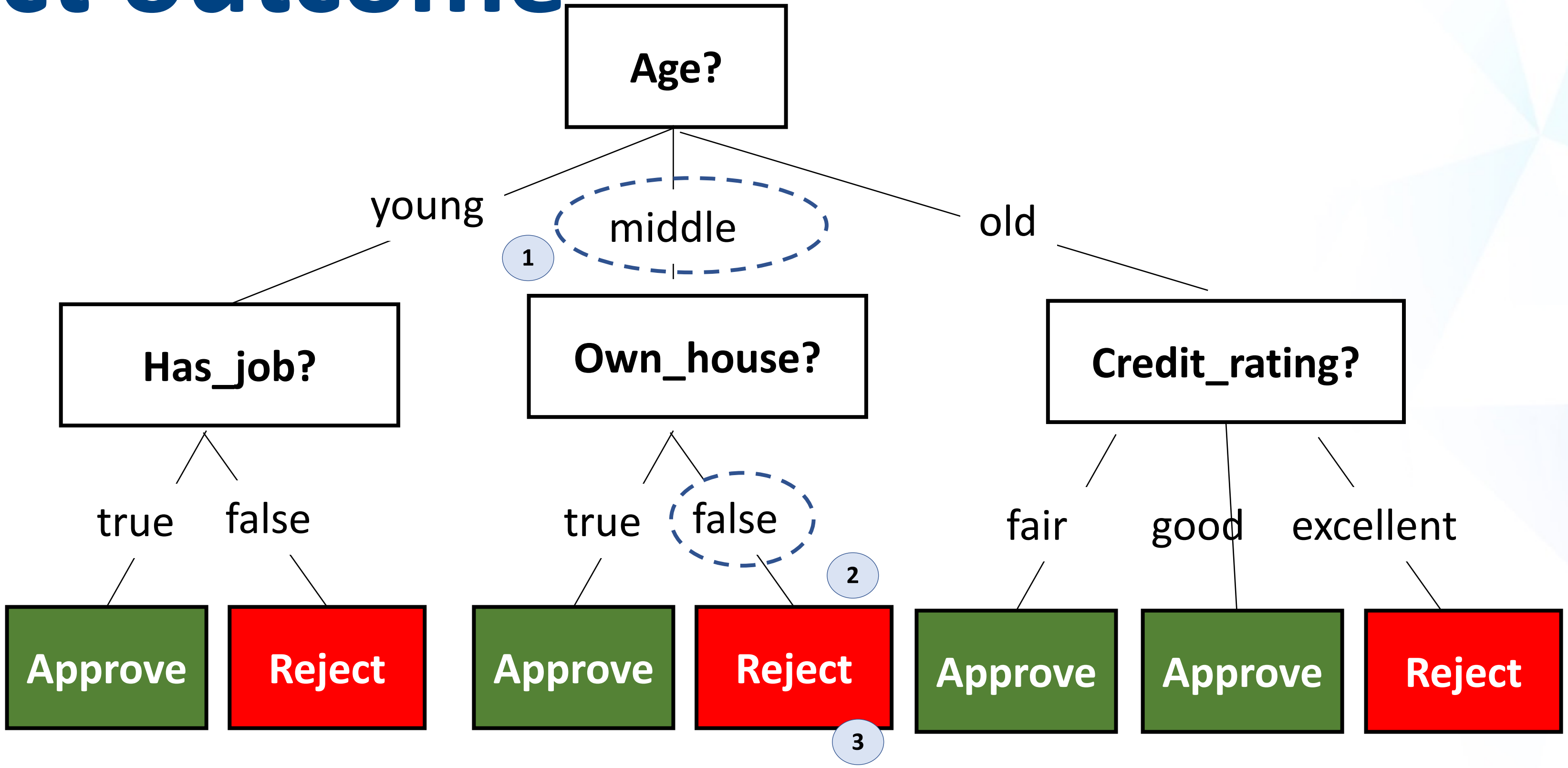
*Outcome: **Approve** (non defaulter) or **Rejecting** (defaulter) an applicant.*

Decision Tree Concepts

A path from root to a leaf node is a **conjunction** (“AND”) of attribute tests



Decision Trees – Predict outcome



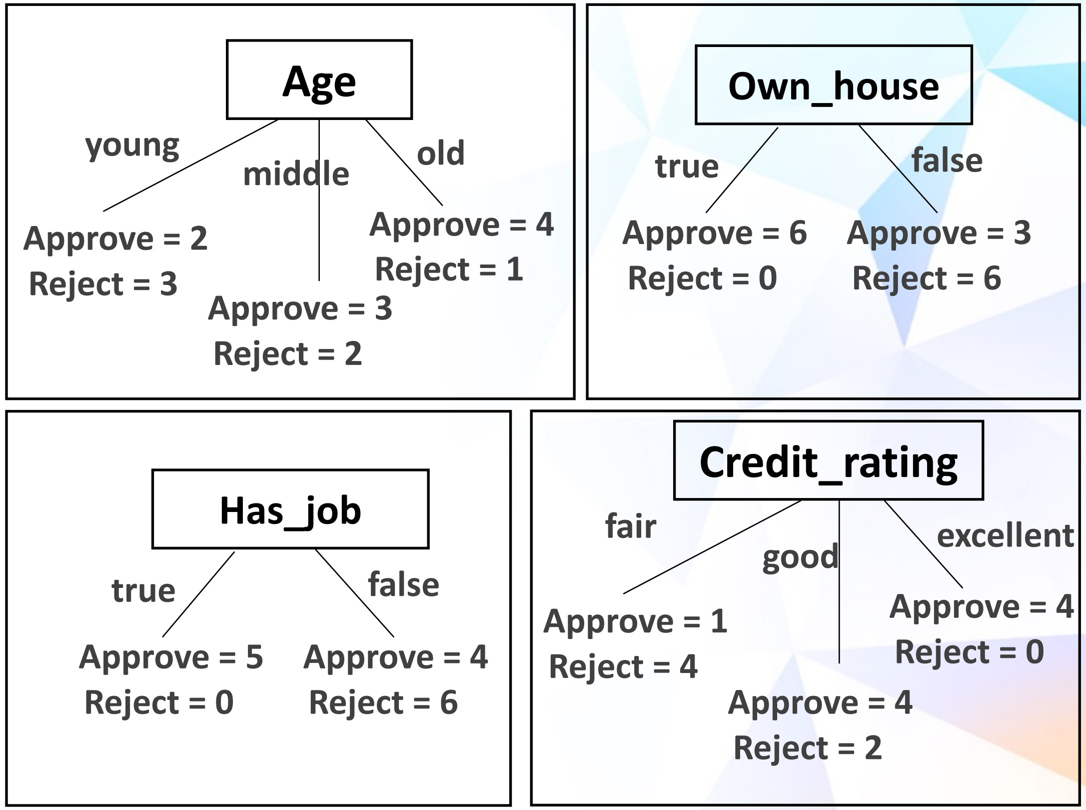
- Given a middle age person, but does not own a house, would the bank approve or reject his application?

Outcome is reject!

Many Possible Split

Age	Has_job	Own_house	Credit_rating	Outcome
young	false	false	fair	Reject
young	false	false	good	Reject
young	true	false	good	Approve
young	true	true	fair	Approve
young	false	false	fair	Reject
middle	false	false	fair	Reject
middle	false	false	good	Reject
middle	true	true	good	Approve
middle	false	true	excellent	Approve
middle	false	true	excellent	Approve
old	false	true	excellent	Approve
old	false	true	good	Approve
old	true	false	good	Approve
old	true	false	excellent	Approve
old	false	false	fair	Reject

Many possible ways to split the same data!



We could start the with root node as Age, Own_house, Has_job or Credit_rating.

The Smallest Tree

Which is the best attribute to be chosen as the root node?

- The one which yields the smallest tree

A popular technique:

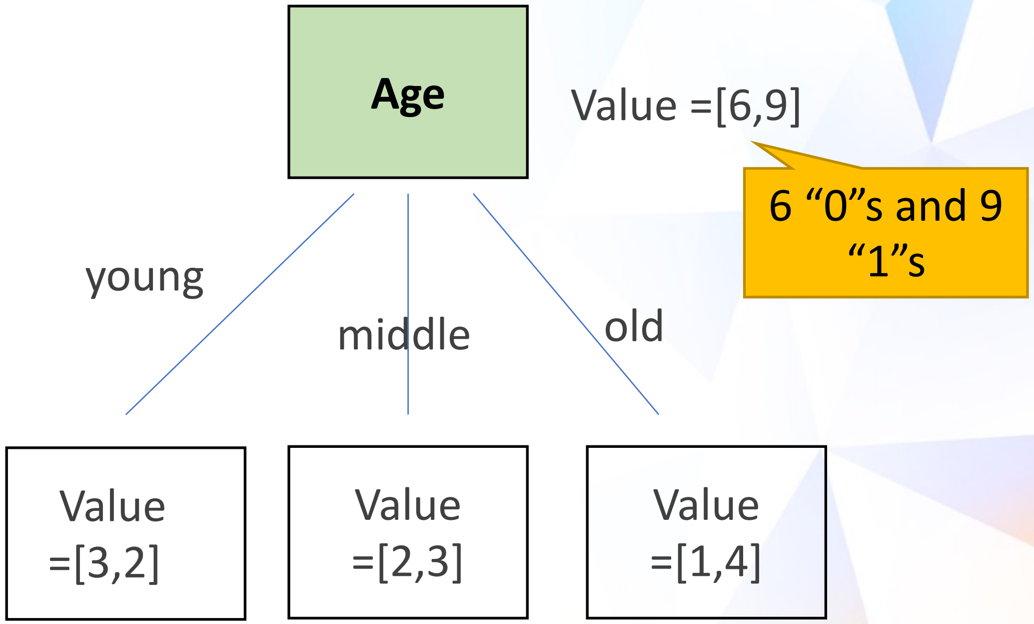
- *Gini index*

Example

Age	Has_job	Own_ house	Credit_ rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0

"0" : Reject (defaulter)
"1": Approve (non-defaulter)

<u>If we build a decision tree with Age as the root node</u>



3 "0"s and 2 "1"s when we branch by "young: age

Determine the Split with Gini Index

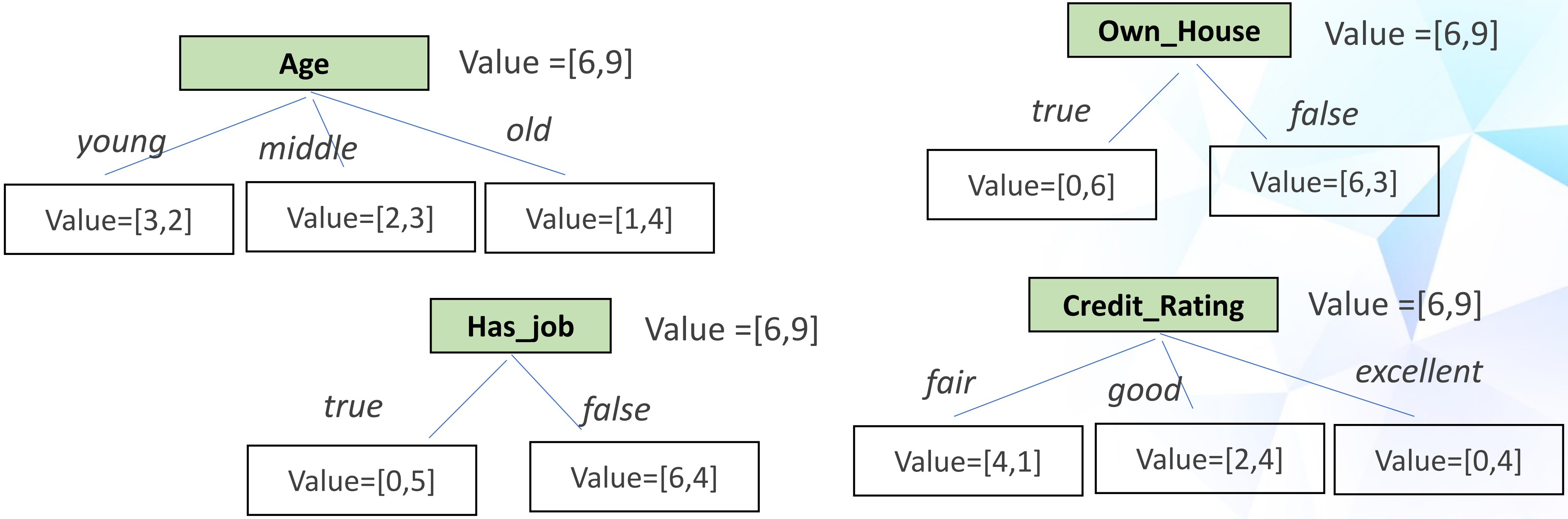
For each possible split, compute Gini index of the nodes

⇩

For each possible split, compute the Gini split value.

⇩

Choose the split with the <u>smallest</u> Gini split value.



Possible Split	Gini Split Value
Age	
Own_House	
Has_Job	
Credit_Rating	

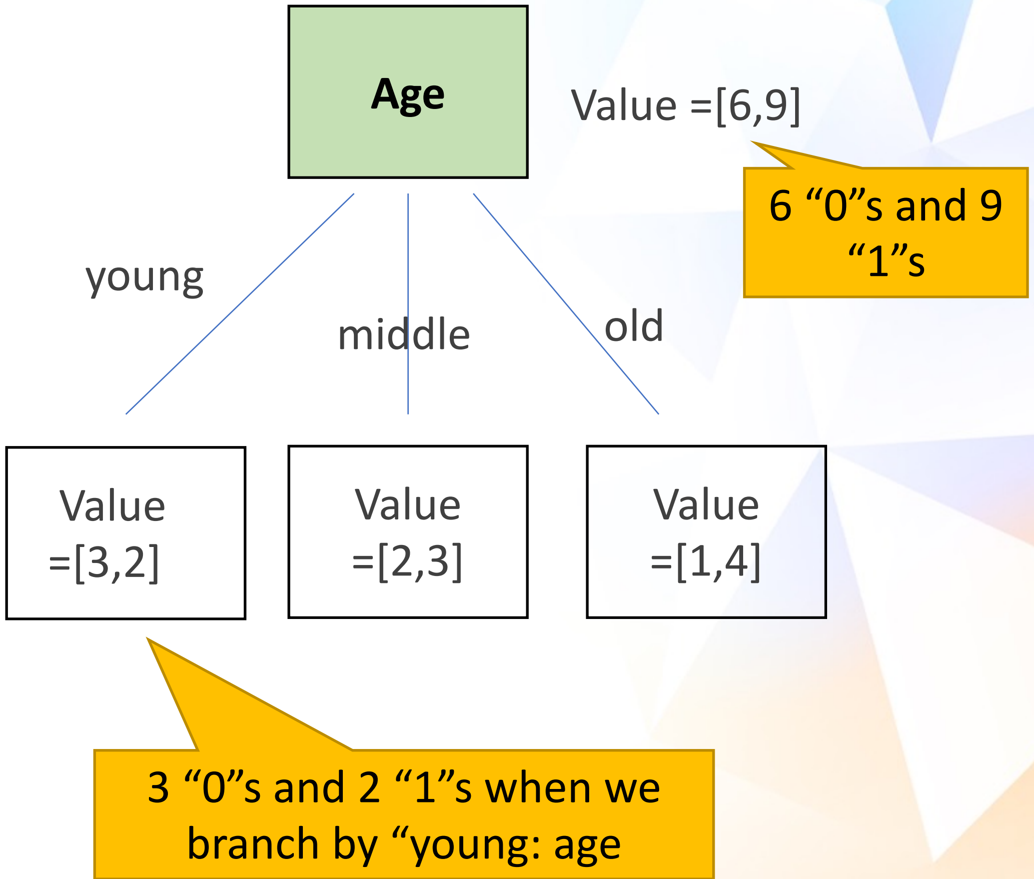
Which split has the lowest split value?

Example

Age	Has_job	Own_ house	Credit_ rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0

“0” : Reject (defaulter)
“1”: Approve (non-defaulter)

<u>Starting with Age as the root node</u>

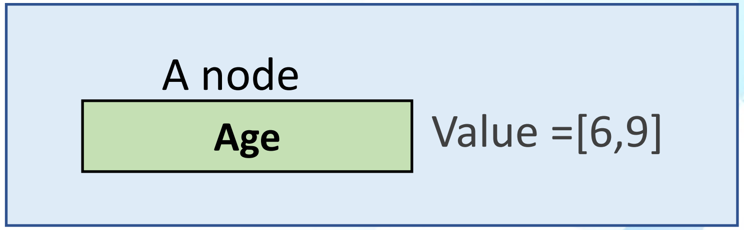


Gini Index and Gini Split

- Gini Index

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

where *p(j | t)* is the relative frequency of class j at node t

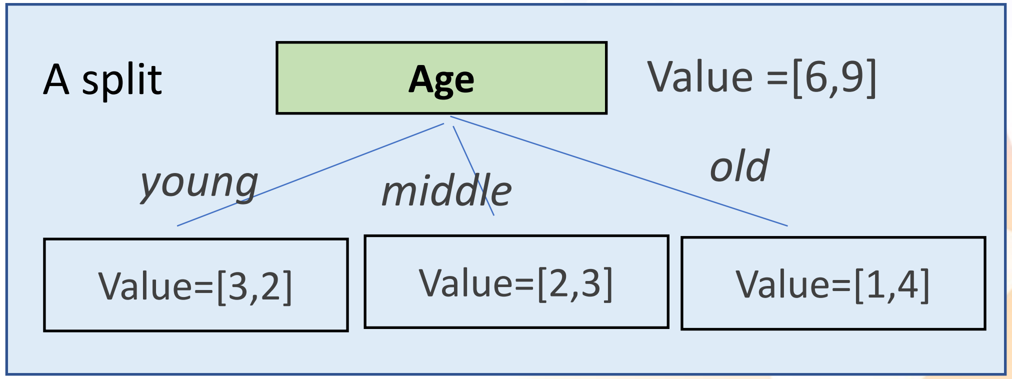


- Gini Split
  - When a node p is split into k partitions (children), the <u>quality of split</u> is computed

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

where, n_i = number of records at child i,

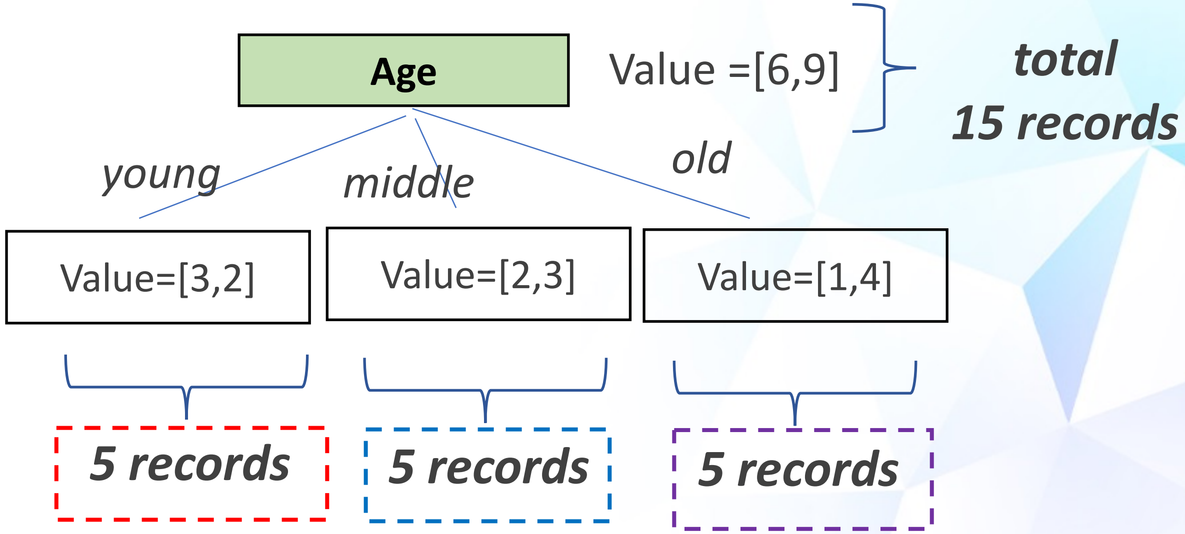
n = number of records at node p.



Gini Index and Split Value (Age)

Compute Gini Index

- for the node and the branches



$$Gini(Age) = 1 - \left(\frac{6}{15}\right)^2 - \left(\frac{9}{15}\right)^2 = 0.48$$

$$Gini(Y) = 1 - \left(\frac{3}{5}\right)^2 - \left(\frac{2}{5}\right)^2 = \boxed{0.48}$$

$$Gini(M) = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2 = \boxed{0.48}$$

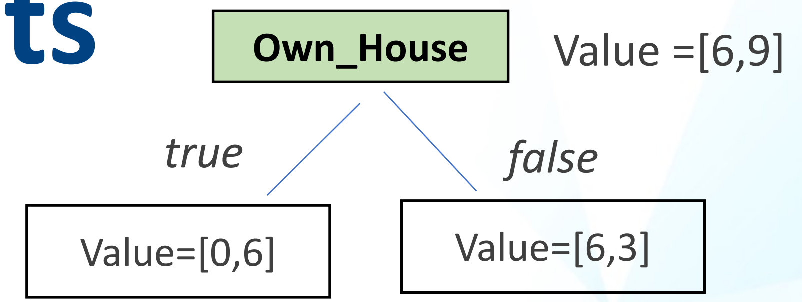
$$Gini(O) = 1 - \left(\frac{1}{5}\right)^2 - \left(\frac{4}{5}\right)^2 = \boxed{0.32}$$

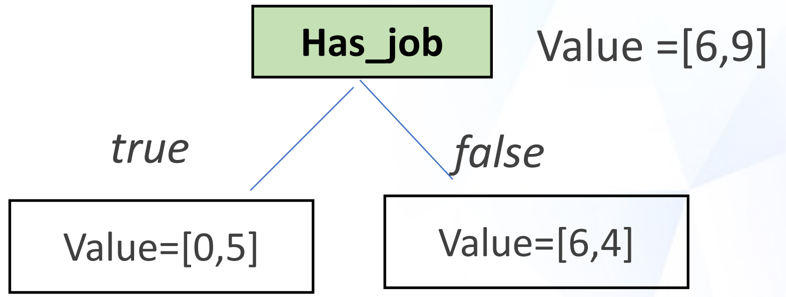
Compute Gini Split

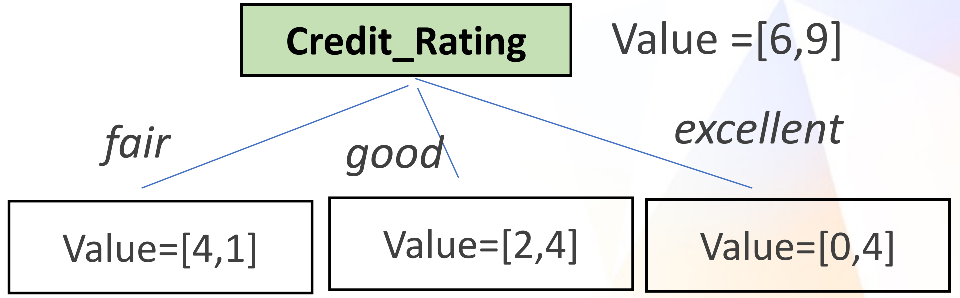
$$\boldsymbol{Gini_{split}(Age)} = \boxed{\left(\frac{5}{15}\right)0.48} + \boxed{\left(\frac{5}{15}\right)0.48} + \boxed{\left(\frac{5}{15}\right)0.32} = \boxed{\mathbf{0.43}}$$

Example – Different Roots

Age	Has_job	Own_house	Credit_rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0



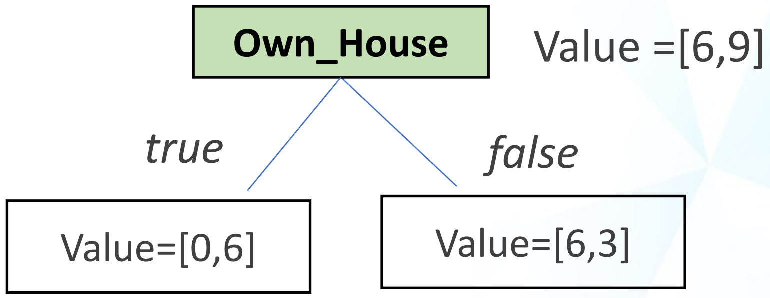




Gini Index and Split Value (Own_House)

Compute Gini Index

- for the node and the branches



$$Gini(Own\_House) = 1 - \left(\frac{6}{15}\right)^2 - \left(\frac{9}{15}\right)^2 = 0.48$$

$$Gini(T) = 1 - \left(\frac{6}{6}\right)^2 - \left(\frac{0}{6}\right)^2 = 0$$

$$Gini(F) = 1 - \left(\frac{3}{9}\right)^2 - \left(\frac{6}{9}\right)^2 = 0.45$$

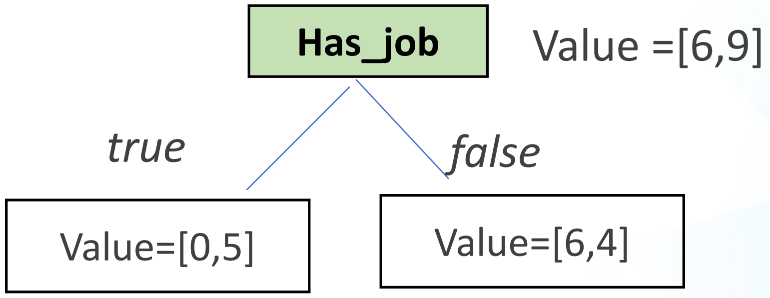
Compute Gini Split

$$\boldsymbol{Gini_{split}(House)} = \left(\frac{6}{15}\right)0 + \left(\frac{9}{15}\right)0.45 = \mathbf{0.27}$$

Gini Index and Split Value (Has_Job)

Compute Gini Index

- for the node and the branches



$$Gini(Has\_Job) = 1 - \left(\frac{6}{15}\right)^2 - \left(\frac{9}{15}\right)^2 = 0.48$$

$$Gini(T) = 1 - \left(\frac{0}{5}\right)^2 - \left(\frac{5}{5}\right)^2 = 0$$

$$Gini(F) = 1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2 = 0.48$$

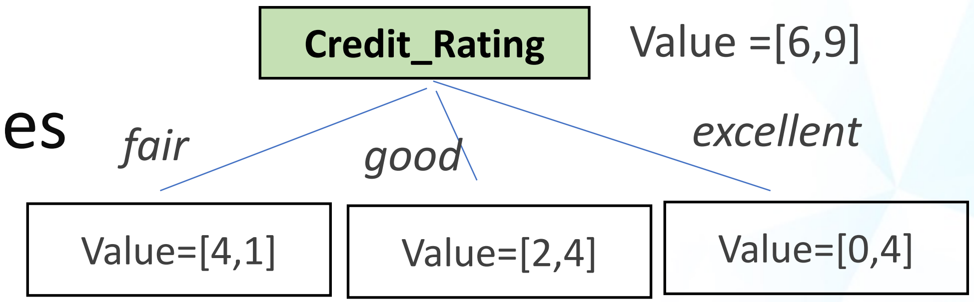
Compute Gini Split

$$\boldsymbol{Gini_{split}(Has\_Job)} = \left(\frac{5}{15}\right)0 + \left(\frac{10}{15}\right)0.48 = \mathbf{0.32}$$

Gini Index and Split Value (Credit_Rating)

Compute Gini Index

- for the node and the branches



$$Gini(Credit\_Rating) = 1 - \left(\frac{6}{15}\right)^2 - \left(\frac{9}{15}\right)^2 = 0.48$$

$$Gini(F) = 1 - \left(\frac{4}{5}\right)^2 - \left(\frac{1}{5}\right)^2 = 0.32$$

$$Gini(G) = 1 - \left(\frac{2}{6}\right)^2 - \left(\frac{4}{6}\right)^2 = 0.45$$

$$Gini(E) = 1 - \left(\frac{0}{4}\right)^2 - \left(\frac{4}{4}\right)^2 = 0$$

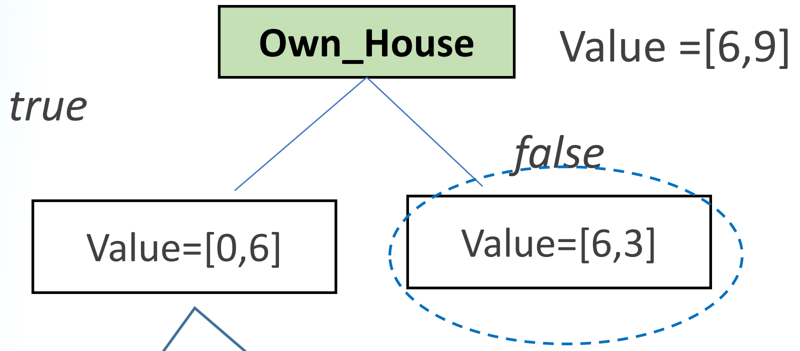
Compute Gini Split

$$\boldsymbol{Gini_{split}(Credit\_Rating)} = \left(\frac{5}{15}\right)0.32 + \left(\frac{6}{15}\right)0.45 + \left(\frac{4}{15}\right)0 = \mathbf{0.285}$$

The First Level

Possible Root Node	Gini Split
Age	0.43
Own_House	0.27
Has_Job	0.32
Credit_Rating	0.285

What Attribute to choose Next?



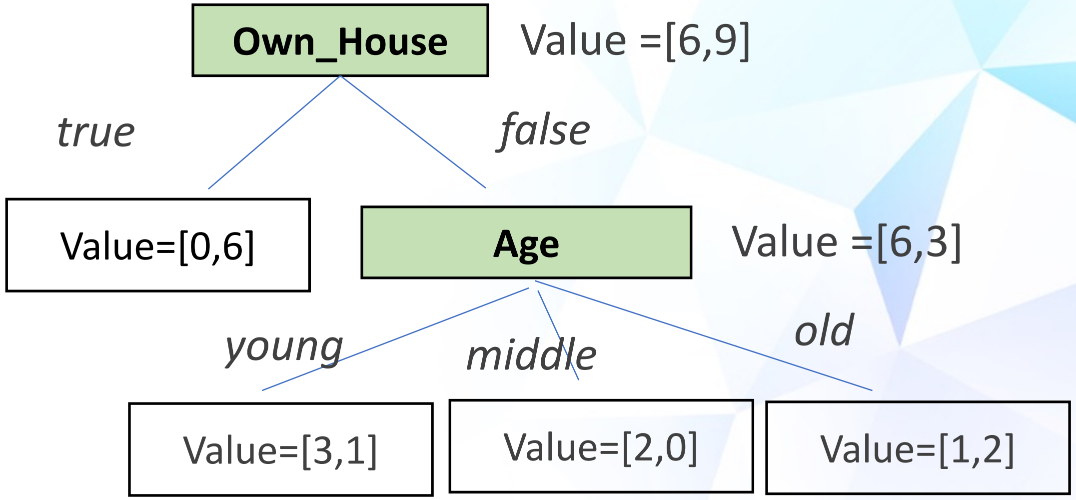
When an applicant owns a house, the outcome is always 1. Hence, there is no need to further split this node.

When an applicant does not owns a house, the outcome could be 0 or 1. Hence, we need to further split this node.

Age	Has_job	Own_ house	Credit_ rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0

The Next Split - By Age

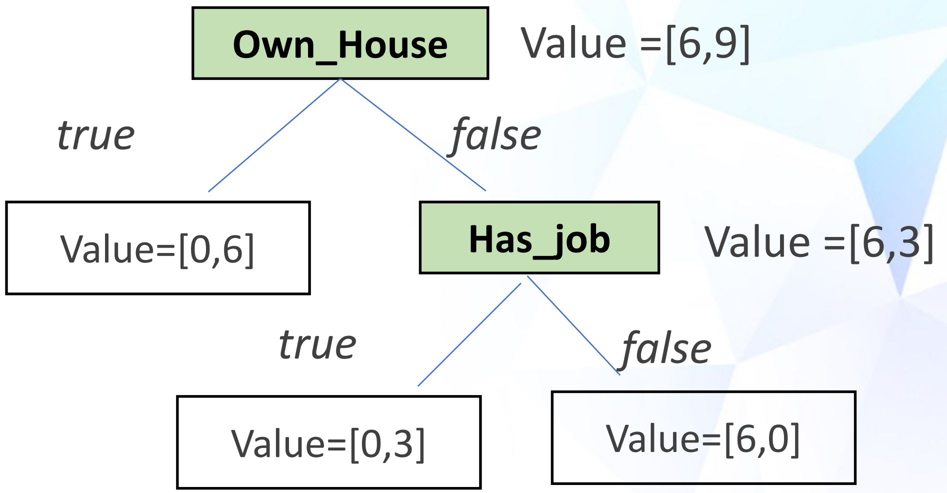
Age	Has_job	Own_house	Credit_rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0



When age group is young, there are 3 “0“ ‘s and 1 “1”’s

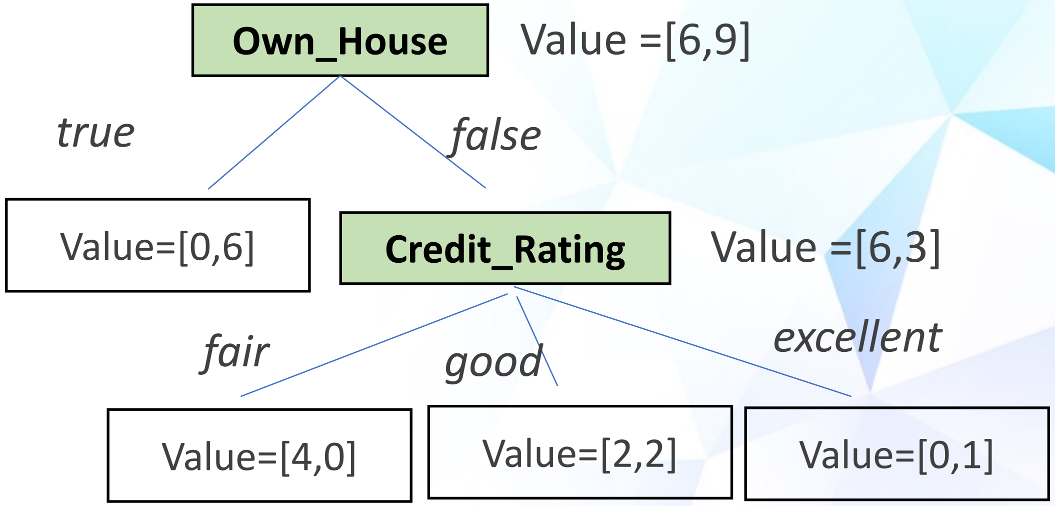
The Next Split - By Has_Job

Age	Has_job	Own_ house	Credit_ rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0



The Next Split - By Credit_Rating

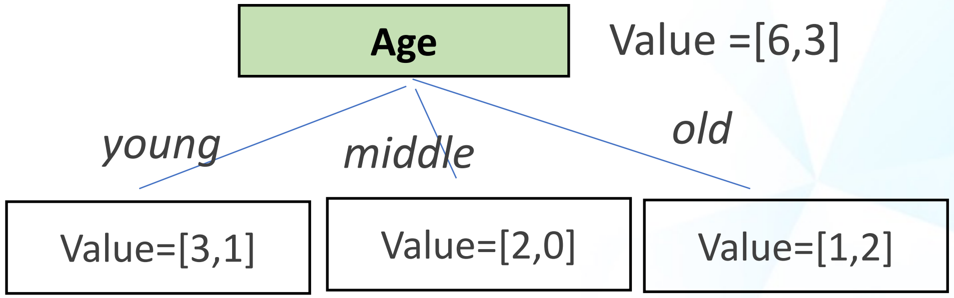
Age	Has_job	Own_house	Credit_rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0



The Next Level-
Gini Split (Age)

Compute Gini Index

- for the node and the branches



$$Gini(Age) = 1 - \left(\frac{6}{9}\right)^2 - \left(\frac{3}{9}\right)^2 = 0.45$$

$$Gini(Y) = 1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^2 = 0.38$$

$$Gini(M) = 1 - \left(\frac{2}{2}\right)^2 - \left(\frac{0}{2}\right)^2 = 0$$

$$Gini(O) = 1 - \left(\frac{1}{3}\right)^2 - \left(\frac{2}{3}\right)^2 = 0.45$$

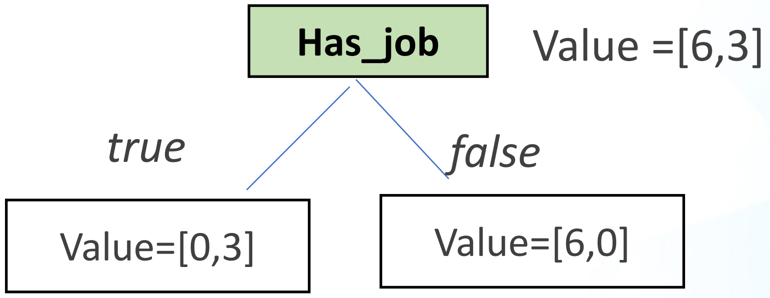
Compute Gini Split

$$\boldsymbol{Gini_{split}(Age)} = \left(\frac{4}{9}\right)0.38 + \left(\frac{2}{9}\right)0 + \left(\frac{3}{9}\right)0.45 = \mathbf{0.31}$$

The Next Level-
Gini Split (Has_Job)

Compute Gini Index

- for the node and the branches



$$Gini(Has\_Job) = 1 - \left(\frac{6}{9}\right)^2 - \left(\frac{3}{9}\right)^2 = 0.45$$

$$Gini(T) = 1 - \left(\frac{0}{3}\right)^2 - \left(\frac{3}{3}\right)^2 = 0$$

$$Gini(F) = 1 - \left(\frac{6}{6}\right)^2 - \left(\frac{0}{6}\right)^2 = 0$$

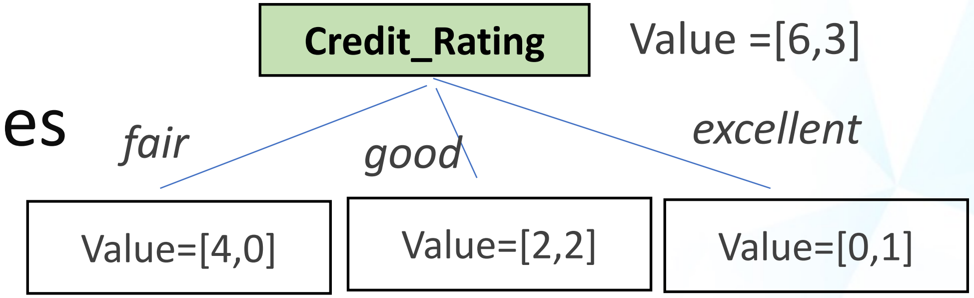
Compute Gini Split

$$\boldsymbol{Gini_{split}(Has\_Job)} = \left(\frac{3}{9}\right)0 + \left(\frac{6}{9}\right)0 = \mathbf{0}$$

The Next Level-
Gini Split (Credit_Rating)

Compute Gini Index

- for the node and the branches



$$Gini(Credit\_Rating) = 1 - \left(\frac{6}{9}\right)^2 - \left(\frac{3}{9}\right)^2 = 0.45$$

$$Gini(F) = 1 - \left(\frac{4}{4}\right)^2 - \left(\frac{0}{4}\right)^2 = 0$$

$$Gini(G) = 1 - \left(\frac{2}{4}\right)^2 - \left(\frac{2}{4}\right)^2 = 0.5$$

$$Gini(E) = 1 - \left(\frac{0}{1}\right)^2 - \left(\frac{1}{1}\right)^2 = 0$$

Compute Gini Split

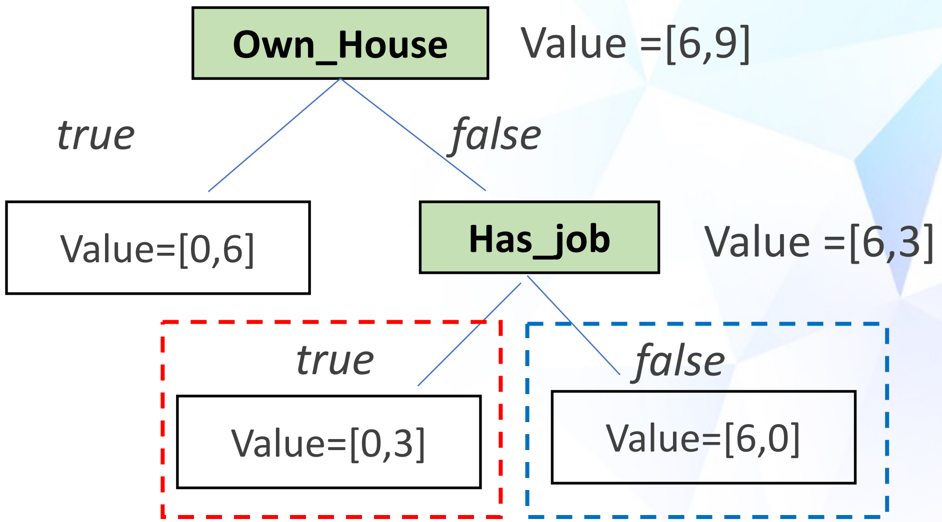
$$\boldsymbol{Gini_{split}(Credit\_Rating)} = \left(\frac{4}{9}\right)0 + \left(\frac{4}{9}\right)0.5 + \left(\frac{1}{9}\right)0 = \mathbf{0.22}$$

The Next Level

Possible Split	Gini Split
Age	0.31
Has_Job	0
Credit_Rating	0.22

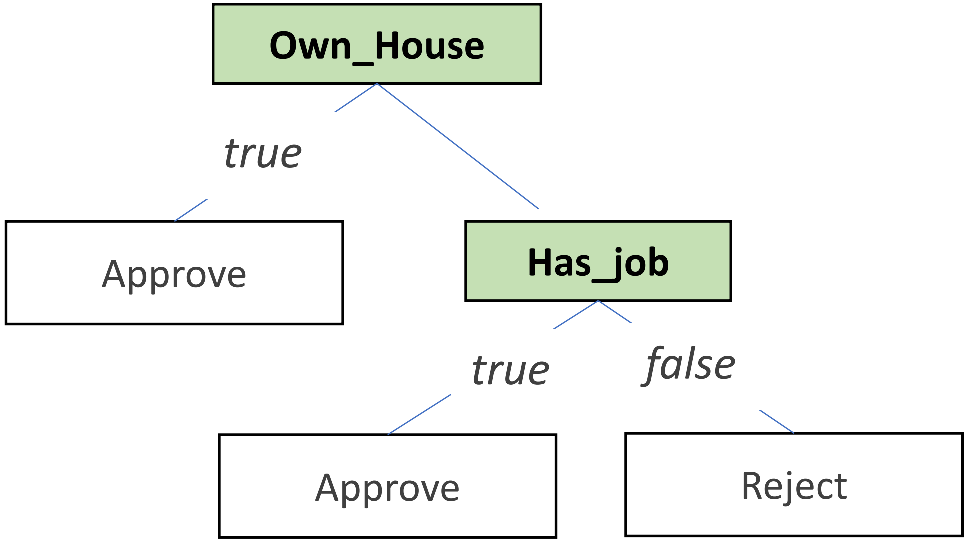
Need Further Split?

Age	Has_job	Own_ house	Credit_ rating	Outcome
young	false	false	fair	0
young	false	false	good	0
young	true	false	good	1
young	true	true	fair	1
young	false	false	fair	0
middle	false	false	fair	0
middle	false	false	good	0
middle	true	true	good	1
middle	false	true	excellent	1
middle	false	true	excellent	1
old	false	true	excellent	1
old	false	true	good	1
old	true	false	good	1
old	true	false	excellent	1
old	false	false	fair	0



No further split is required!

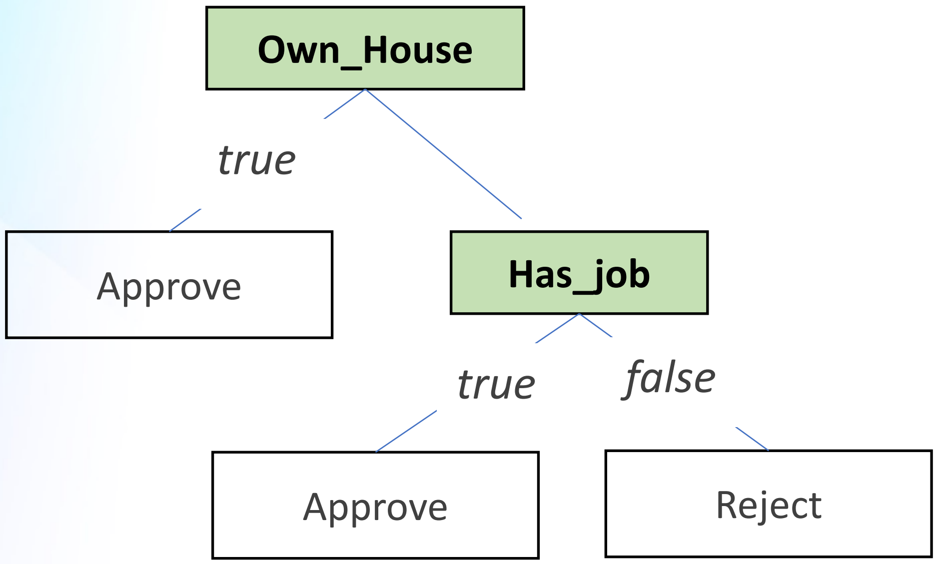
Final Decision Tree



Decision Tree - Features

- Features
    - The machine could handle both **numerical and categorical** data

Decision Tree Advantages - 1



Forming of Business Rules

1) Decision Logic yielded by the tree:

IF *own_house=true*

THEN *Approve*

IF *own_house=false* ***AND*** *has_job=true*

THEN *Approve*

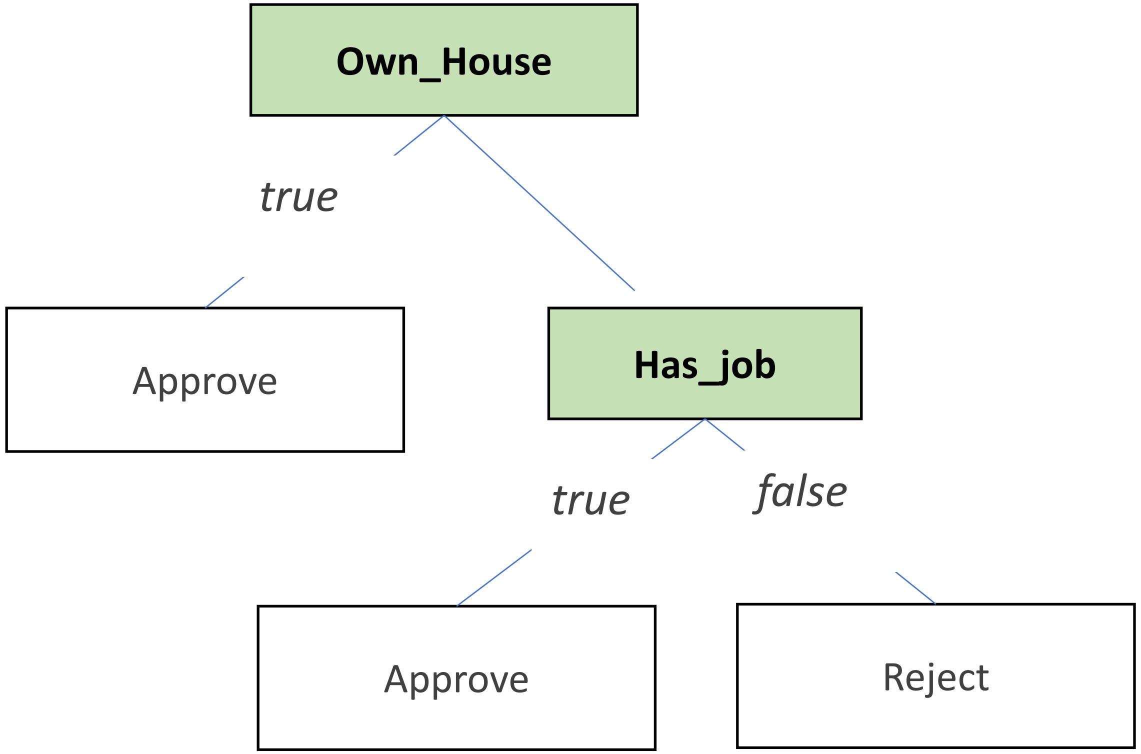
IF *own_house=false* ***AND*** *has_job=false*

THEN *Reject*

2) Based on this dataset, **only two attributes** are needed to classify new applicants

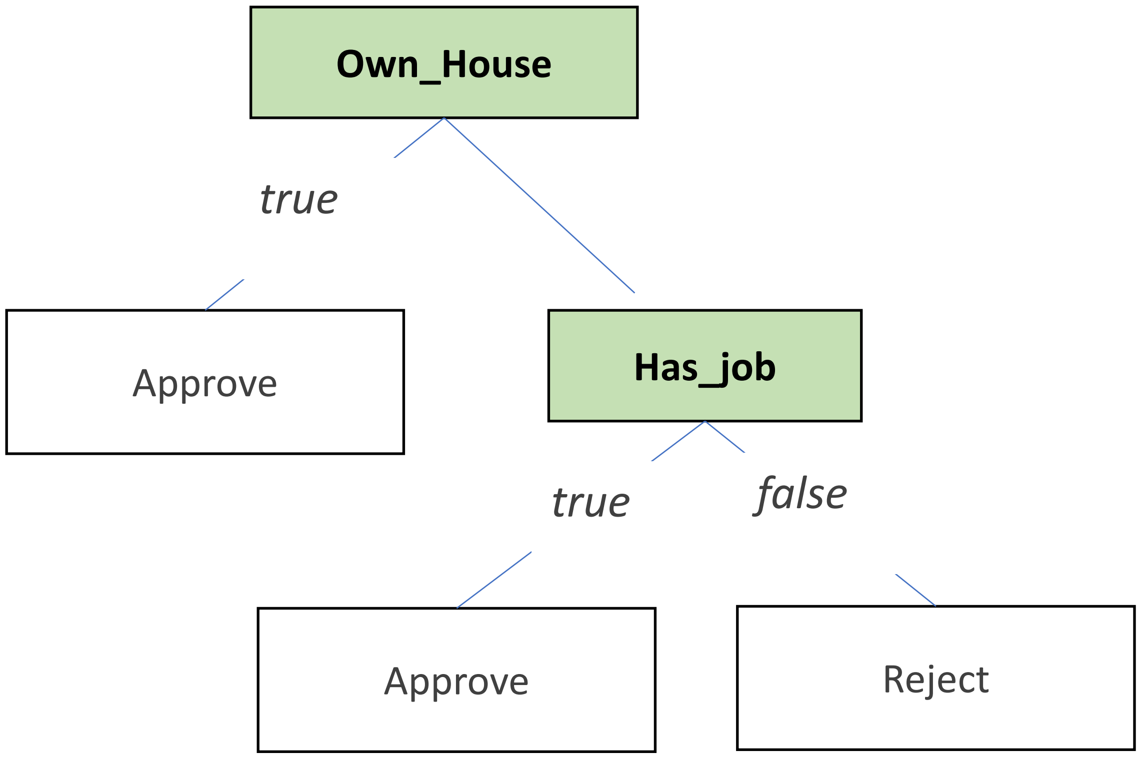
Decision Tree Advantages - 2

- Decision Tree **can be visualized** - simple to understand



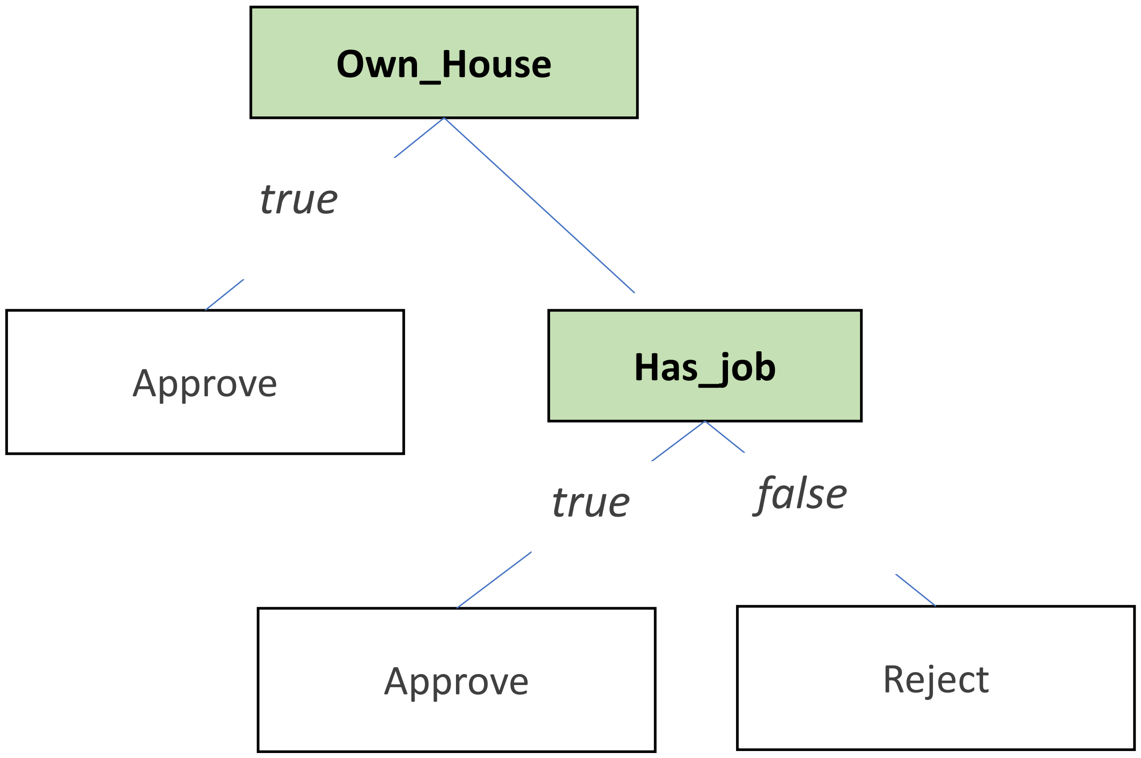
Decision Tree Advantages - 3

- The model can be **easily explained**



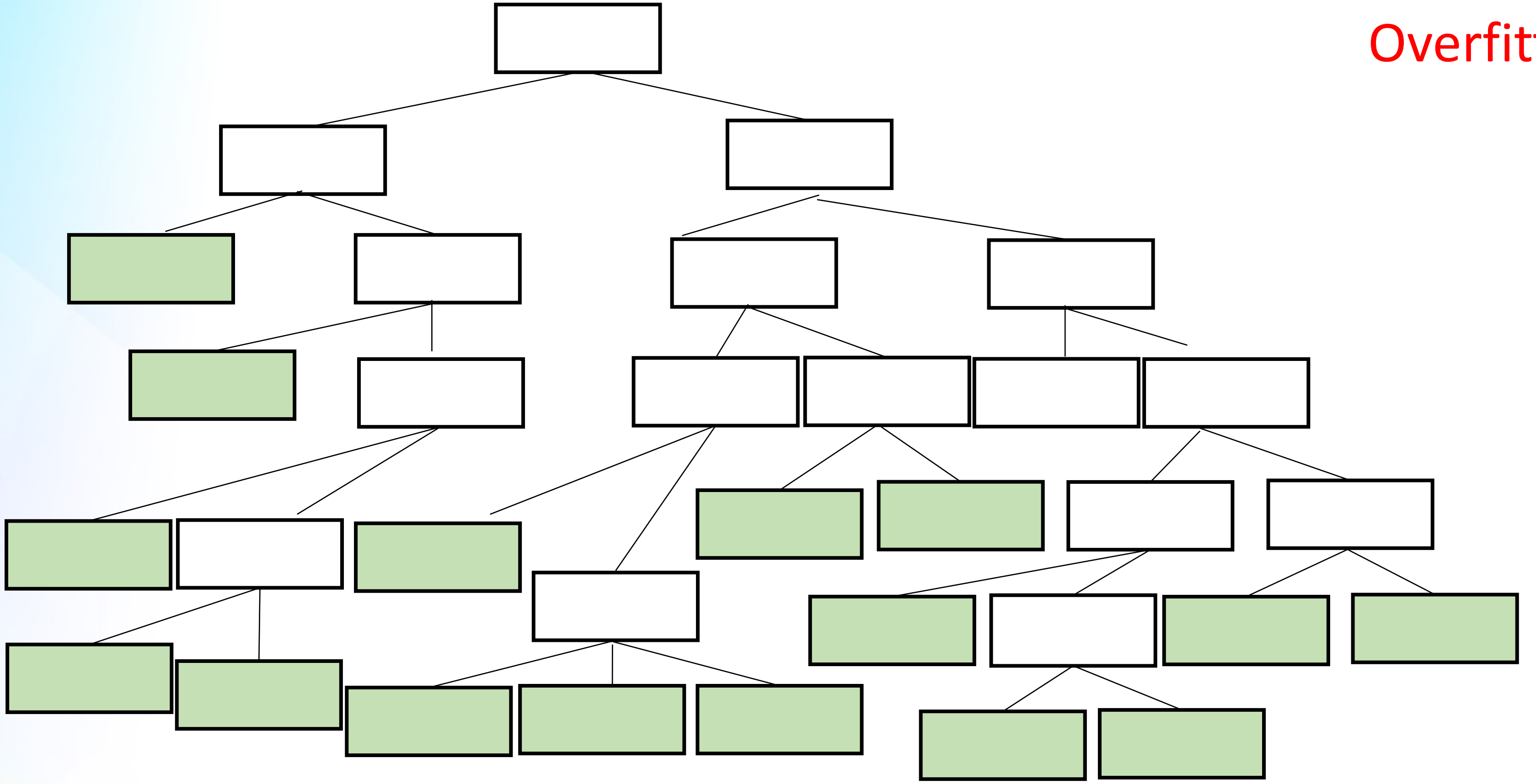
Decision Tree Advantages - 4

- Model **can be validated** by the domain expert



Overfitting

Decision Trees can suffer from
Overfitting

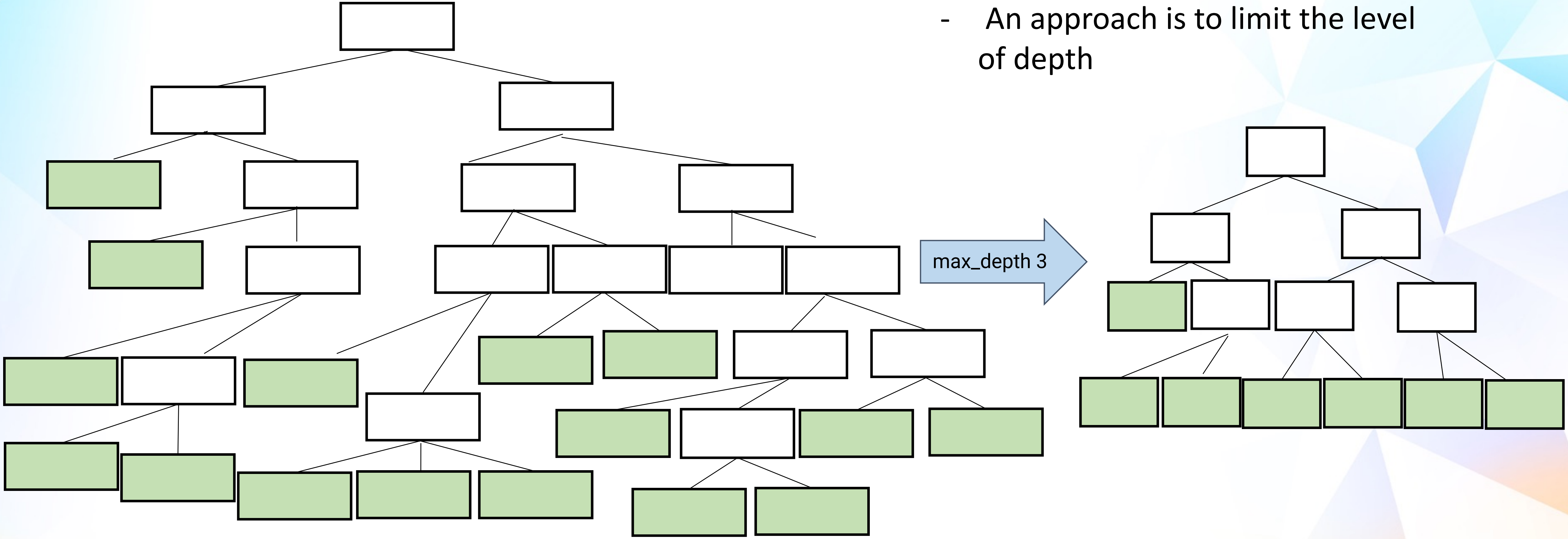


Level of depth = 6
Some outcome need 6 levels of split!

Level of Depth

Handling Overfitting

- An approach is to limit the level of depth



Level of depth = 6

What Have We Learnt?

- What is a decision tree classification model
- How the model constructs an optimal decision tree
- The advantages of a decision tree model
- Handling overfitting of a decision tree model