Machine Learning
Supervised learning – Decision Trees

DR. BHARGAVI R

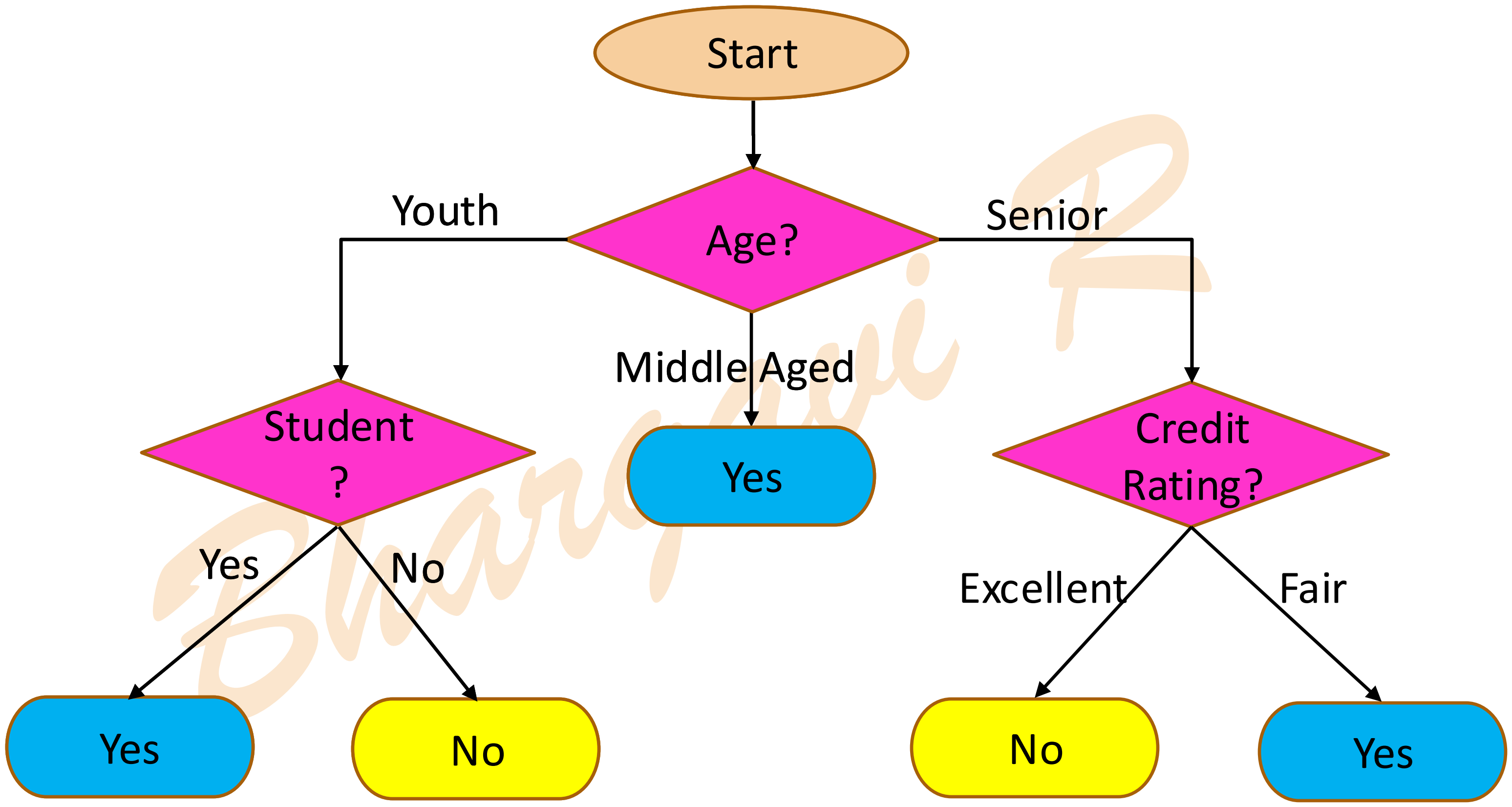
SCOPE

VIT CHENNAI

Predicting Potential Computer Buyer

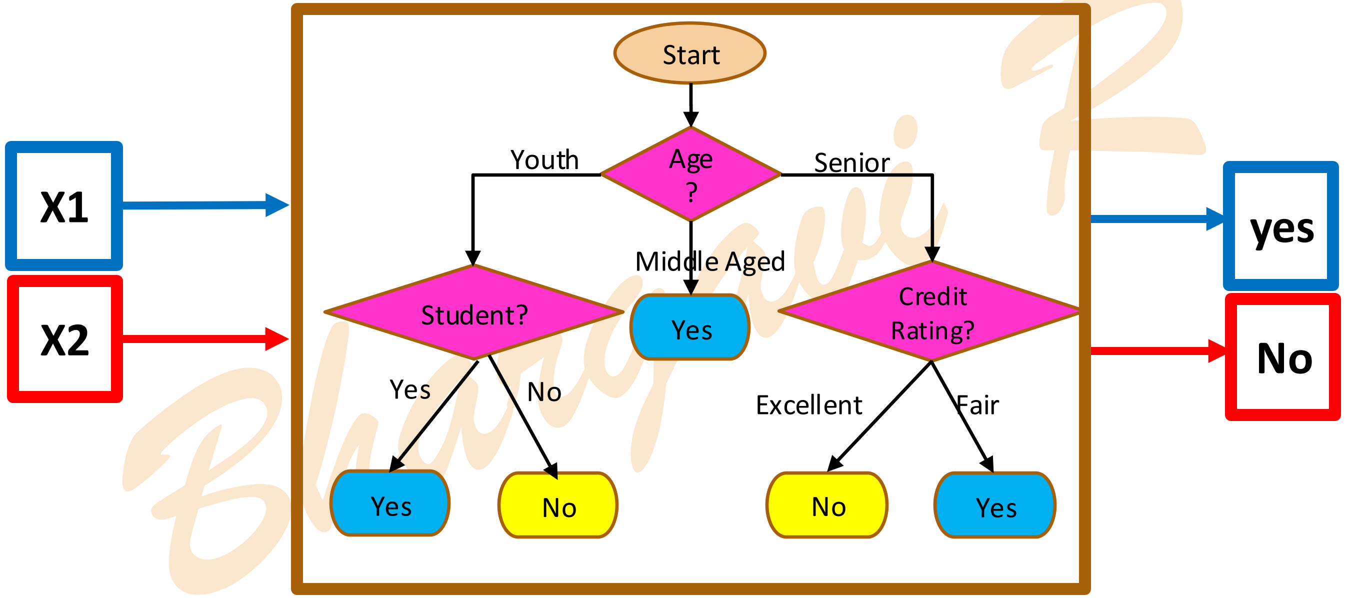
ID	Age	Income	Student	Credit Rating	Buy_Status
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
3	Middle Aged	High	No	Fair	Yes
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
7	Middle Aged	Low	Yes	Excellent	Yes
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
10	Senior	Medium	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes
12	Middle Aged	Medium	No	Excellent	Yes
13	Middle Aged	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

Decision Tree



Classifying a new test data

X = (age = youth, Income = medium, Student = yes, Credit_rating = Fair)



Attribute Selection – Classification Error

- Classification Error = $\frac{\#\ incorrect\ predictions}{Total\ \#\ Predictions}$ (= 0 if all correct predictions and =1 all incorrect)
- Other popular Attribute selection criteria:
  - Gini Index
  - Information Gain

Learning Task

- The objective of Decision Tree learning is to model a tree in such a way that classification error (or any other chosen quality metric) in the training data is minimized.
- This is a hard (NP- hard) problem as exponential large number of possible trees exists.
- We will use Greedy approach to model the decision tree.
- Incrementally built the Decision tree that can result in best classification error rate.

Decision Tree Algorithm -Steps

Step 1: Start with an empty tree

Step2: Select a feature to split the data

For each split of the data

Step 3: If no further splitting, then, Make predictions (based on majority class)

Step 4: Else, Goto step 2 and recurse on this split

Conditions for Stopping Partitioning

Conditions for stopping partitioning

- All samples for a given node belong to the same class
- There are no remaining attributes for further partitioning – majority voting is employed for classifying the leaf
- There are no samples left
- Limiting the depth of the tree

Feature Split Selection

- Choose the Best Feature to split on.
- What is Best?
  - The Feature that results in lowest Classification Error rate.
- For Prediction in a partition - Use Majority class

Predicting Potential Computer Buyer

ID	Age	Income	Student	Credit Rating	Buy_Status
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
3	Middle Aged	High	No	Fair	Yes
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
7	Middle Aged	Low	Yes	Excellent	Yes
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
10	Senior	Medium	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes
12	Middle Aged	Medium	No	Excellent	Yes
13	Middle Aged	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

Feature Selection (cont…)

- Split on Age:
- Classification error rate = 4/14 = 0.28

Youth (5)	Buy(yes) = 2 Buy(No) = 3	Mistakes = 2 (Prediction= No)
Middleaged(4)	Buy(yes) = 4	
Senior(5)	Buy(yes) = 3 Buy(No) = 2	Mistakes = 2 (Prediction= Yes)

- Split on Income:
- Classification error rate = 5/14 = 0.35

High (4)	Buy(yes) = 2 Buy(No) = 2	Mistakes = 2
Medium(6)	Buy(yes) = 4 Buy(No) = 2	Mistakes = 2 (Prediction= Yes)
Low(4)	Buy(yes) = 3 Buy(No) = 1	Mistakes = 1 (Prediction= Yes)

Feature Selection (cont…)

- Split on Credit_Rating:
- Classification error rate = 5/14 = 0.35

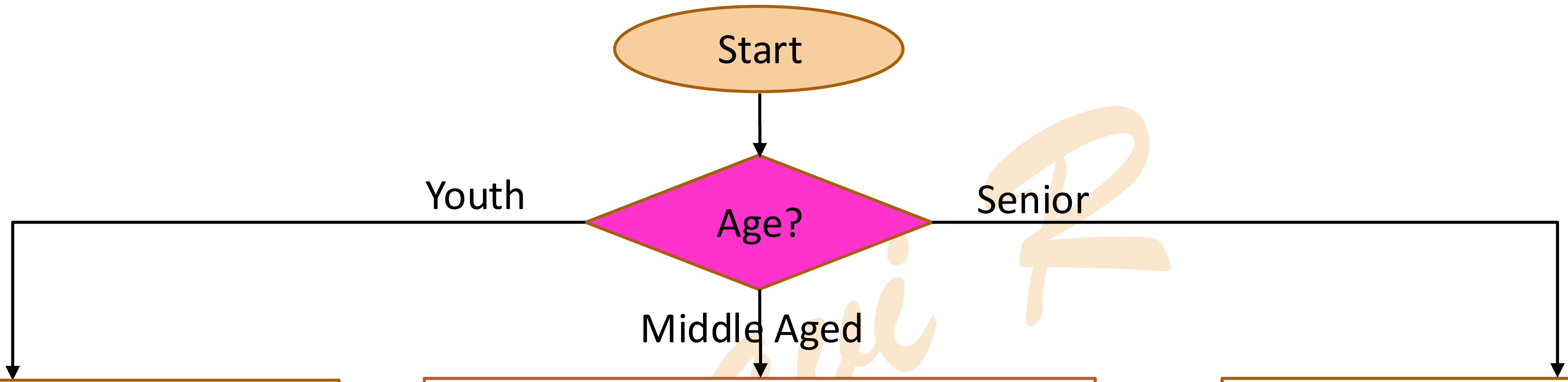
Excellent(6)	Buy(yes) = 3 Buy(No) = 3	Mistakes = 3 (Prediction= Yes)
Fair(8)	Buy(yes) = 6 Buy(No) = 2	Mistakes = 2 (Prediction= Yes)

- Split on Student
- Classification error rate = 4 / 14 = 0.28

Yes(7)	Buy(yes) = 6 Buy(No) = 1	Mistakes = 1 (Prediction= Yes)
No(7)	Buy(yes) = 3 Buy(No) = 4	Mistakes = 3 (Prediction= No)

- Since Error rate with split on “age” is less Choose “Age” as split attribute

Decision Stump



Income	Student	credit	buy
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

Income	Student	credit	buy
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

Income	Student	credit	buy
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Feature Split Selection-Second level

Subset with Age = Youth

- Split on Credit_Rating:
- Classification error rate = 2/5 = 0.4

Excellent(2)	Buy(yes) = 1 Buy(No) = 1	Mistakes = 1 (Prediction= Yes)
Fair(3)	Buy(yes) = 1 Buy(No) = 2	Mistakes = 1 (Prediction= No)

- Split on Student
- Classification error rate = 0 / 5 = 0

Yes(2)	Buy(yes) = 3 Buy(No) = 0	Mistakes = 0 (Prediction= Yes)
No(3)	Buy(yes) = 0 Buy(No) = 2	Mistakes = 0 (Prediction= No)

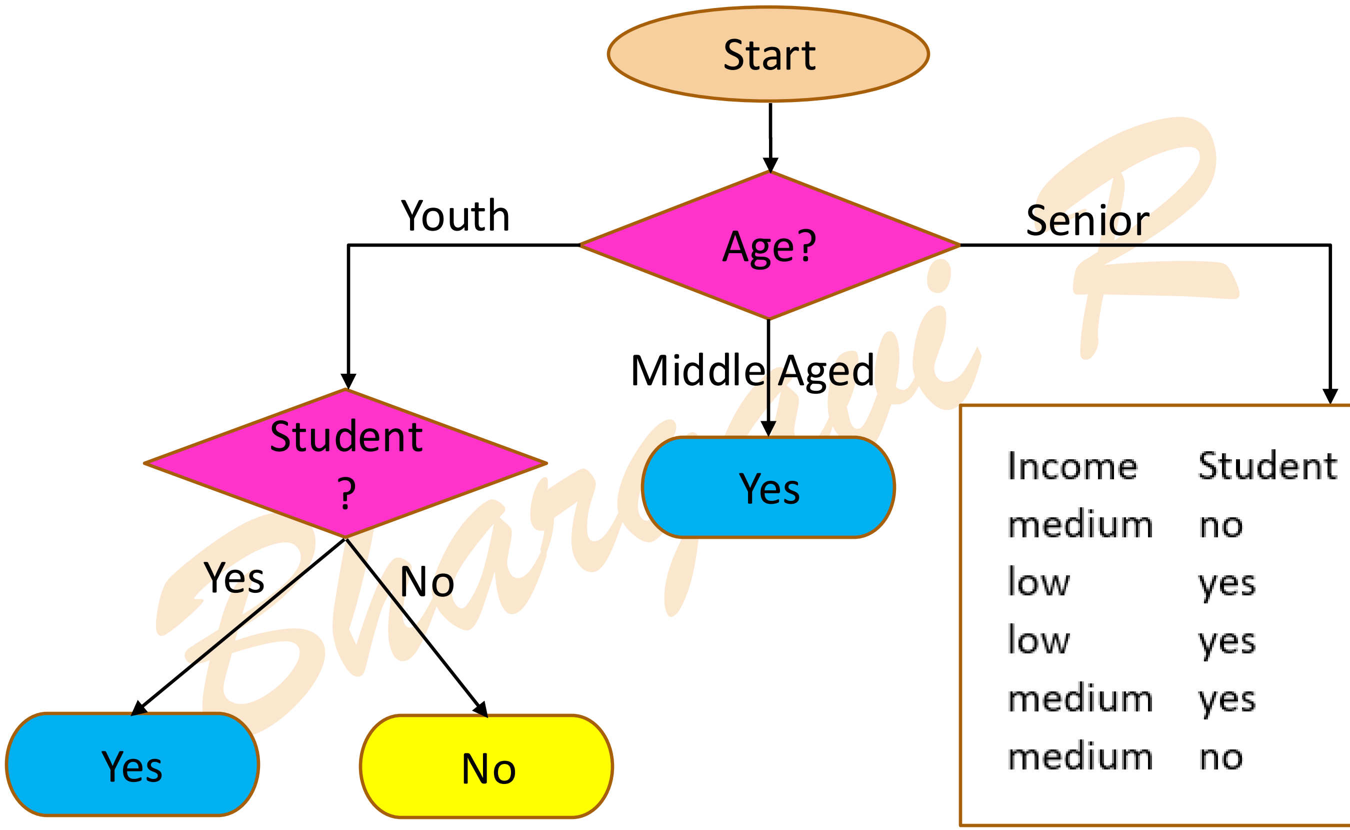
Feature Split Selection – Second Level (cont…)

- Split on Income:
- Classification error rate = 1/5 = 0.2

High (2)	Buy(No) = 2	Mistakes = 0
Medium(2)	Buy(yes) = 1 Buy(No) = 1	Mistakes = 1 Prediction(Yes)
Low(1)	Buy(yes) = 1	Mistakes = 0

- Since Error rate with split on “Student” is less Choose “Student” as split attribute

Feature Split Selection – Second Level (cont…)



Income	Student	credit	buy
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Feature Split Selection-Second level (cont…)

Subset with Age = Senior

- Split on Credit_Rating:
- Classification error rate = 0/5 = 0

Excellent(2)	Buy(No) = 2	Mistakes = 0 (Prediction= No)
Fair(3)	Buy(Yes) = 3	Mistakes = 0 (Prediction= Yes)

- Split on Student
- Classification error rate = 2 / 5 = 0.4

Yes(3)	Buy(yes) = 2 Buy(No) = 1	Mistakes = 1 (Prediction= Yes)
No(2)	Buy(yes) = 1 Buy(No) = 1	Mistakes = 1 (Prediction= Yes)

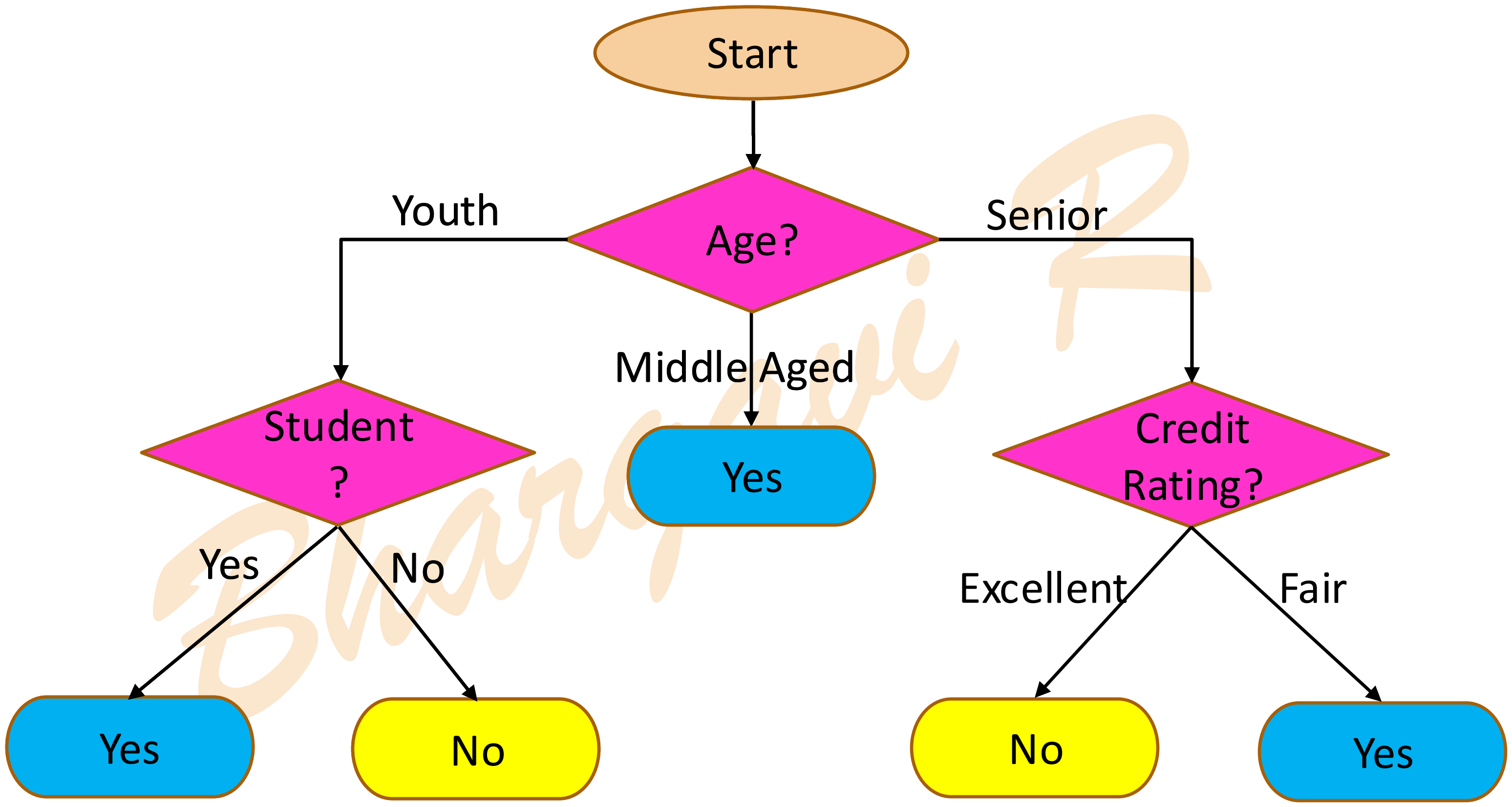
Feature Split Selection – Second Level (cont…)

- Split on Income:
- Classification error rate = 2/5 = 0.4

Medium(3)	Buy(yes) = 2 Buy(No) = 1	Mistakes = 1 Prediction(Yes)
Low(2)	Buy(yes) = 1 Buy(No) = 1	Mistakes = 1 Prediction(Yes)

- Since Error rate with split on “Credit_Rating” is less Choose “Credit_Rating” as split attribute

Decision Tree

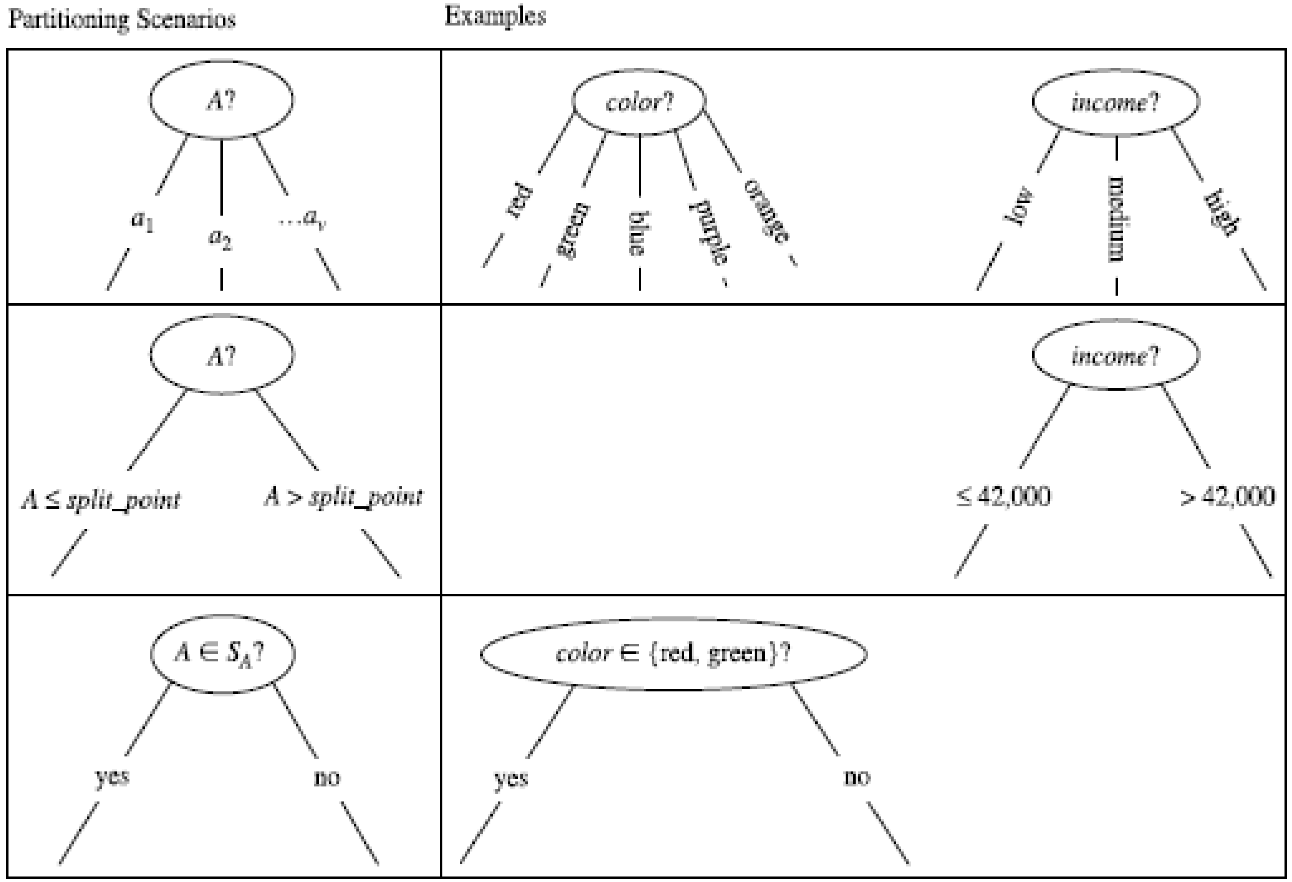


Partitioning Possibilities

Discrete

Continuous

Discrete and Binary tree to be produced



Overfitting

Overfitting: A decision tree may overfit the training data

- Too many branches, some may reflect anomalies due to noise or outliers
- Poor accuracy for unseen samples

Tree Pruning

Prepruning

- Halt tree construction early—do not split a node if this would result in the goodness measure falling below a threshold
- Difficult to choose an appropriate threshold

Postpruning

- Remove branches from a “fully grown” tree - get a sequence of progressively pruned trees
- Use a set of data different from the training data to decide which is the “best pruned tree”

Entropy

Entropy is a measure of uncertainty or randomness in a system.

It quantifies the amount of disorder or unpredictability in a set of possible outcomes.

Entropy = $-\sum_{i=1}^{m} p_i \log_2(p_i)$

Here m is the number of classes.

Example: Consider that we have a dataset having three colors of fruits, 2 red, 2 green, and 4 yellow.

Entropy of the dataset = $-(2/8\ (\log_2(2/8)) + 2/8\ (\log_2(2/8)) + 4/8\ (\log_2(4/8))\ ) = 1.5$

Suppose if we had all fruits of the same color then Entropy = 0

ID3 – Iterative Dichotomiser 3

- Feature split Selection - **Information Gain**
- Select the attribute with the highest information gain for splitting the data D.
- Expected information (entropy) needed to classify an observation in D is

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

- Where p_i is the probability that an arbitrary observation in D belongs to class C_i, (i = 1,,2,3 … ,m) and it is computed as

$$p_i = \frac{|c_i, D|}{|D|}$$

$|c_i, D|$ - Number of observations of class C_i in D

$|D|$ - Number of observations in D

ID3 – Feature Split Selection (cont…)

- Information still needed (after using A to split D into v partitions) to classify D

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

- Information gained by splitting on attribute A i.e Gain(A) = Info(D) – $Info_A$(D)

ID3 – Feature Split Selection (cont…)

- Class P: buy_Status = “yes”
- Class N: buy_Status= “no”
- Information gained by splitting on attribute A i.e Gain(A) = Info(D) – Info_A(D)

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

CART with Gini Split Selection (Binary Tree)

- If a data set D contains examples from k classes, gini index is defined as

$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

Where p_i is the relative frequency of class i in D

- If a data set D is split on attribute A into two subsets D1 and D2, the Gini index $gini_A(D)$ is defined as

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

- Reduction in impurity is given by $\Delta Gini(A) = Gini(D) - Gini_A(D)$
- The attribute that provides the smallest $Gini_{split}(D)$ (or the largest reduction in impurity) is chosen to split the node

Gini - Example

- For the example Data set we have

$$Gini(D) = 1 - \left(\left(\frac{9}{14}\right)^2 + \left(\frac{5}{14}\right)^2\right) = 0 \cdot 459$$

- If we consider split on attribute Age then we have the following possible splitting subset

{{youth, middle aged},{youth, senior}, {middle aged, Senior}, {youth}, {middle aged}, {senior}}

- We need to consider all possible combinations for binary split

Gini – Example (cont…)

- If we consider {youth, middle aged } as one subset and rest in the other subset

$$Gini_{Age\in\{youth,middleaged\}}(D) = Gini_{Age\in\{Senior\}}(D) =$$

$$\frac{9}{14}\left(1-\left(\left(\frac{6}{9}\right)^2+\left(\frac{3}{9}\right)^2\right)\right)+\frac{5}{14}\left(1-\left(\left(\frac{3}{5}\right)^2+\left(\frac{2}{5}\right)^2\right)\right)=0.461$$

Age	Buys -Yes	Buys -No
{youth,middle aged} (9)	6	3
Senior (5)	3	2

- If we consider {youth, senior } as one subset and rest in the other subset

$$Gini_{Age\in\{youth,senior\}}(D) = Gini_{Age\in\{middle\ aged\}}(D) =$$

$$\frac{10}{14}\left(1-\left(\left(\frac{5}{10}\right)^2+\left(\frac{5}{10}\right)^2\right)\right)+\frac{4}{14}\left(1-\left(\left(\frac{4}{4}\right)^2+\left(\frac{0}{4}\right)^2\right)\right)=0\cdot 357$$

Age	Buys -Yes	Buys -No
{youth,senior} (10)	5	5
Middle aged (4)	4	0

Gini – Example (cont…)

- If we consider {middle aged, senior } as one subset and rest in the other subset

$$Gini_{Age\in\{middle\ aged,senior\}}(D) = Gini_{Age\in\{youth\}}(D) = \frac{9}{14}\left(1-\left(\left(\frac{7}{9}\right)^2+\left(\frac{2}{q}\right)^2\right)\right)+\frac{5}{14}\left(1-\left(\left(\frac{2}{5}\right)^2+\left(\frac{3}{5}\right)^2\right)\right) = 0\cdot 394$$

Age	Buys -Yes	Buys -No
{middle aged,senior} (9)	7	2
youth (5)	2	3

- Now, Since $Gini_{Age\in\{youth,senior\}}(D)$ has the lowest value we will assign $\text{Gini}_{\text{Age}} = 0.357$

Gini – Example (cont…)

- Next we consider Student attribute for splitting

$Gini_{Student\}}(D)$

$$\frac{7}{14}\left(1-\left(\left(\frac{6}{7}\right)^2+\left(\frac{1}{7}\right)^2\right)\right)+\frac{7}{14}\left(1-\left(\left(\frac{3}{7}\right)^2+\left(\frac{4}{7}\right)^2\right)\right)=0.367$$

Student	Buys - Yes	Buys - No
Yes(7)	6	1
No(7)	3	4

- Next, we consider splitting on Credit_rating

$$\frac{6}{14}\left(1-\left(\left(\frac{3}{6}\right)^2+\left(\frac{3}{6}\right)^2\right)\right)+\frac{8}{14}\left(1-\left(\left(\frac{6}{8}\right)^2+\left(\frac{2}{8}\right)^2\right)\right)=0.428$$

Credit_rating	Buys - Yes	Buys - No
Excellent(6)	3	3
Fair(8)	6	2

Gini – Example (cont…)

Attribute	Gini Index	Reduction in Impurity
Age	0.357	0.459 – 0.357 =0.102
Income	0.443	0.459 – 0.443 = 0.016
Student	0.367	0.459 – 0.367 = 0.092
Credit rating	0.428	0.459 – 0.428 = 0.031

- Since Split on Age (with{youth, Senior}in one branch and {middle aged}on the other) has lowest Gini index or highest reduction in impurity, Age is selected for split.
- Now split the data set on Age into two subsets and continue the process with each subset.

Gain Ratio – C4.5

- Gain Ratio is an extension to information gain.
- To overcome the biased nature of Information gain, it applies normalization to information gain using "split information", defined as:

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

- Gain Ratio is defined as: $GainRatio(A) = \dfrac{Gain(A)}{SplitInfo(A)}$

Predicting Potential Computer Buyer

ID	Age	Income	Student	Credit Rating	Buy_Status
1	Youth	High	No	Fair	No
2	Youth	High	No	Excellent	No
3	Middle Aged	High	No	Fair	Yes
4	Senior	Medium	No	Fair	Yes
5	Senior	Low	Yes	Fair	Yes
6	Senior	Low	Yes	Excellent	No
7	Middle Aged	Low	Yes	Excellent	Yes
8	Youth	Medium	No	Fair	No
9	Youth	Low	Yes	Fair	Yes
10	Senior	Medium	Yes	Fair	Yes
11	Youth	Medium	Yes	Excellent	Yes
12	Middle Aged	Medium	No	Excellent	Yes
13	Middle Aged	High	Yes	Fair	Yes
14	Senior	Medium	No	Excellent	No

C4.5 numerical

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

SplitInfo(credit_rating) = $-(8/14).\log_2(8/14) - (6/14).\log_2(6/14) = 0.461 + 0.524 = 0.985$

GainRatio(credit_rating) = Gain(credit_rating) / SplitInfo(credit_rating)

= 0.049 / 0.985 = 0.049

C4.5 calculates the gain ratio for each potential attribute and chooses the one with the highest value.

This ensures that the selected attribute provides the most information while also being less biased towards attributes with many values.

Comparison – Split Criteria

Information gain: (ID3)

- Biased towards multivalued attributes

Gain ratio: (C4.5)

- Tends to prefer unbalanced splits in which one partition is much smaller than the others

Gini index: (CART)

- Biased to multivalued attributes
- Has difficulty when the number of classes is large
- Tends to favor tests that result in equal-sized partitions and purity in both partitions

Regression

Day	Outlook	Temp	Humidity	Wind	Golf Players
1	Sunny	Hot	High	Weak	25
2	Sunny	Hot	High	Strong	30
3	Overcast	Hot	High	Weak	46
4	Rain	Mild	High	Weak	45
5	Rain	Cool	Normal	Weak	52
6	Rain	Cool	Normal	Strong	23
7	Overcast	Cool	Normal	Strong	43
8	Sunny	Mild	High	Weak	35
9	Sunny	Cool	Normal	Weak	38
10	Rain	Mild	Normal	Weak	46
11	Sunny	Mild	Normal	Strong	48
12	Overcast	Mild	High	Strong	52
13	Overcast	Hot	Normal	Weak	44
14	Rain	Mild	High	Strong	30

Regression (Cont…)

Standard Deviation metric

Golf players :

- Average of Golf Players = 39.78
- SD = 9.32

Outlook:

Sunny –

Average of Golf players =35.2

SD of Golf Players = 7.78

Outlocok	Temp	Humidity	Wind	Golf Players
Sunny	Hot	High	Weak	25
Sunny	Hot	High	Strong	30
Sunny	Mild	High	Weak	35
Sunny	Cool	Normal	Weak	38
Sunny	Mild	Normal	Strong	48

Regression (Cont…)

Overcast :

Average of players = 46.25

SD of Players= 3.49

Day	Outlook	Temp	Humidity	Wind	Golf Players
3	Overcast	Hot	High	Weak	46
7	Overcast	Cool	Normal	Strong	43
12	Overcast	Mild	High	Strong	52
13	Overcast	Hot	Normal	Weak	44

Rainy:

Average of Players= 39.2

SD of players= 10.87

Day	Outlook	Temp	Humidity	Wind	Golf Players
4	Rain	Mild	High	Weak	45
5	Rain	Cool	Normal	Weak	52
6	Rain	Cool	Normal	Strong	23
10	Rain	Mild	Normal	Weak	46
14	Rain	Mild	High	Strong	30

Regression (Cont…)

Weighted SD for Outlook :

(4/14)x3.49 + (5/14)x10.87 + (5/14)x7.78 = 7.66

Therefore SD reduction for Outlook = 9.32 – 7.66 = 1.66

Temperature:

Hot :SD of Golf Players = 8.95

Cool : SD of Golf players = 10.51

Mild : SD of Golf Players = 7.65

Temp	SD of Golf Players	Instances
Hot	8.95	4
Cool	10.51	4
Mild	7.65	6

Weighted SD for Temp =

(4/14)x8.95 + (4/14)x10.51 + (6/14)x7.65 = 8.84

Therefore SD reduction for Outlook = 9.32 – 8.84 = 0.47

Regression (Cont…)

Similarly we can compute and find the Weighted SD and reduction for Humidity and Wind attributes as –

Weighted SD Humidity = 9.04

SD reduction for Humidity = 0.27

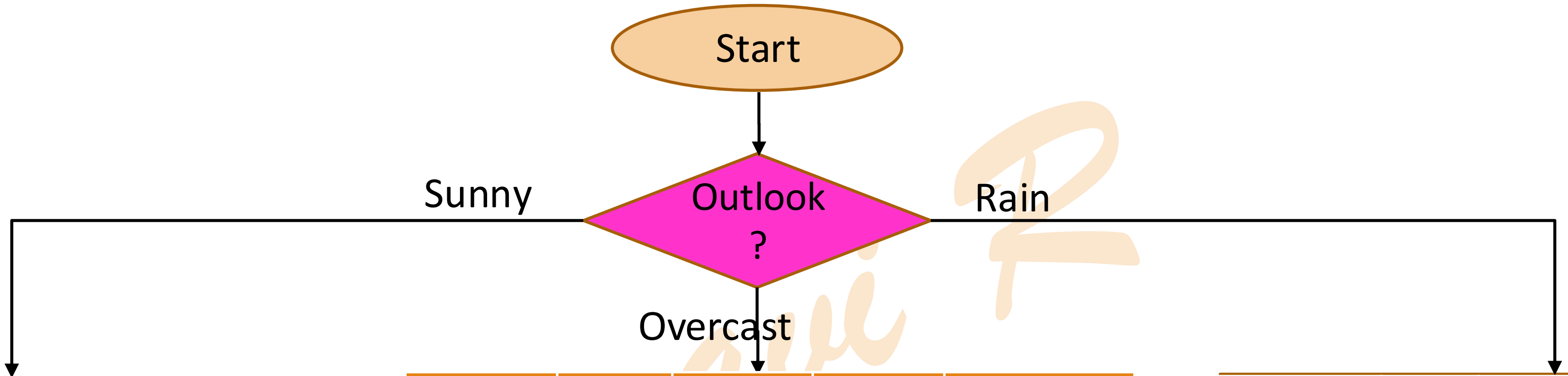
Weighted SD for Wind = 9.03

SD reduction for Wind = 0.29

The attribute with Highest SD reduction is selected for splitting.

Here Outlook with 1.66 is the winner

Regression (Cont…)



Outlocok	Temp	Humidit y	Wind	Golf Players
Sunny	Hot	High	Weak	25
Sunny	Hot	High	Strong	30
Sunny	Mild	High	Weak	35
Sunny	Cool	Normal	Weak	38
Sunny	Mild	Normal	Strong	48

Outlook	Temp	Humidit y	Wind	Golf Players
Overcast	Hot	High	Weak	46
Overcast	Cool	Normal	Strong	43
Overcast	Mild	High	Strong	52
Overcast	Hot	Normal	Weak	44

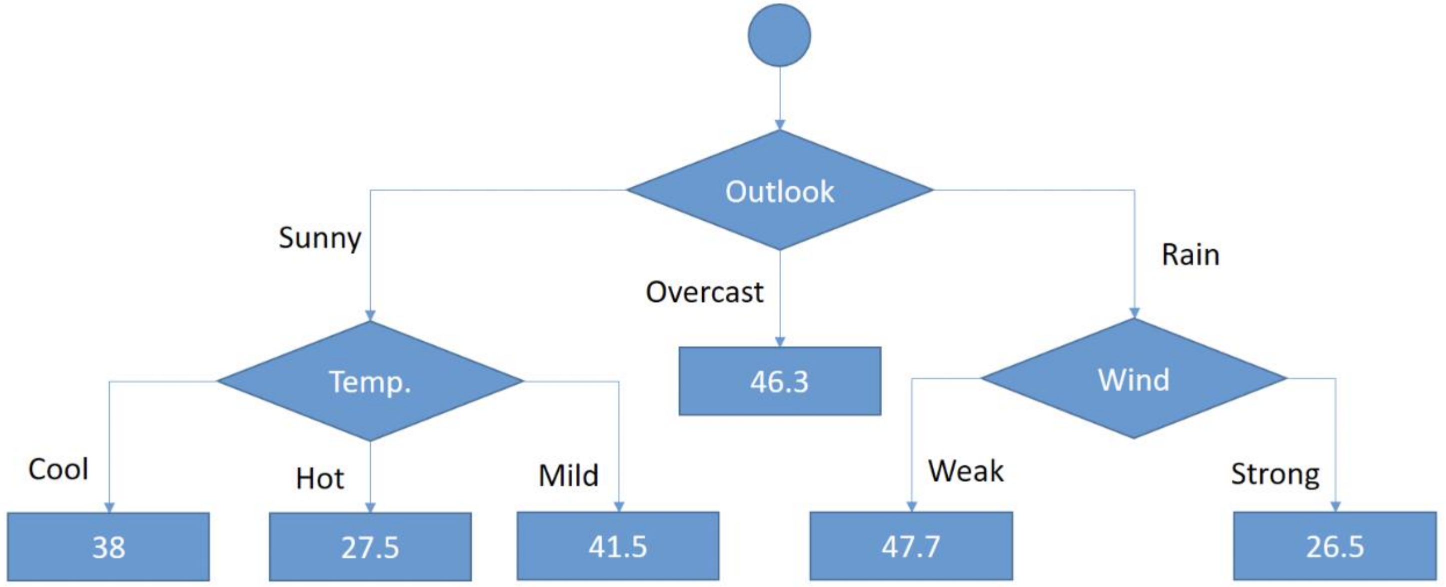
Outlook	Temp	Humidit y	Wind	Golf Players
Overcast	Hot	High	Weak	46
Overcast	Cool	Normal	Strong	43
Overcast	Mild	High	Strong	52
Overcast	Hot	Normal	Weak	44

Regression (Cont…)

Continue the process with each of the subsets of the data.

We can terminate further splitting when the SD of the sub data set is less than 5% of the entire dataset (or number of instances is less than 5) .

If we apply the terminate condition of < 5 then it will result in the following tree



Handling Continuous Valued Attribute

Humidity	Decision
85	No
90	No
78	Yes
96	Yes
80	Yes
70	No
65	Yes
95	No
70	Yes
80	Yes
70	Yes
90	Yes
75	Yes
80	No

Continuous Valued Attributes (cont…)

Sorted on Humidity

Humidity	Decision
65	Yes
70	No
70	Yes
70	Yes
75	Yes
78	Yes
80	Yes
80	Yes
80	No
85	No
90	No
90	Yes
95	No
96	Yes

Unique on Humidity

65	70	75	78	80	85	90	95	96

Continuous Valued Attributes (cont…)

<= 70:

Humidity	Decision
65	Yes
70	No
70	Yes
70	Yes

>70

Humidity	Decision
75	Yes
78	Yes
80	Yes
80	Yes
80	No
85	No
90	No
90	Yes
95	No
96	Yes

Compute Gini/Information gain

Continuous Valued Attributes (cont…)

<=75

Humidity	Decision
65	Yes
70	No
70	Yes
70	Yes
75	Yes

>75

Humidity	Decision
78	Yes
80	Yes
80	Yes
80	No
85	No
90	No
90	Yes
95	No
96	Yes

Compute Gini/Information gain

Continuous Valued Attributes (cont…)

Gini - 70

~~≤70~~	yes	No
≤70 (4)	3	1
>70 (10)	6	4

$$\frac{4}{14}\left(1-\left(\left(\frac{3}{4}\right)^2+\left(\frac{1}{4}\right)^2\right)\right)+\frac{10}{14}\left(1-\left(\left(\frac{6}{10}\right)^2+\left(\frac{4}{10}\right)^2\right)\right)$$

Gini - 75

	yes	No
≤75 (5)	4	1
>75 (9)	5	4

$$\frac{5}{14}\left(1-\left(\left(\frac{4}{5}\right)^2+\left(\frac{1}{5}\right)^2\right)\right)+\frac{9}{14}\left(1-\left(\left(\frac{5}{9}\right)^2+\left(\frac{4}{9}\right)^2\right)\right)$$

Gini - 78

	yes	No
≤78 (6)	5	1
>78 (8)	4	4

$$\frac{6}{14}\left(1-\left(\left(\frac{5}{6}\right)^2+\left(\frac{1}{6}\right)^2\right)\right)+\frac{8}{14}\left(1-\left(\left(\frac{4}{8}\right)^2+\left(\frac{4}{8}\right)^2\right)\right)$$

Continuous Valued Attributes (cont…)

Summarize the Information gain for all the splits

Humidity	Information Gain/Reduction in impurity(Gini)
65	0.047
70	0.014
75	0.045
78	0.09
80	0.1
85	0.025
90	0.01
95	0.047
96	0

Choose the split with highest Information gain. Here <=80 is the Threshold for split