Machine learning

Tigran Melkonyan

Homework 3

AUA ID:
ME 1172366

Problem 1 coding problem

Problem 2 The time complexity for training a decision tree is generally $O(n \cdot m \log n)$ where:

n = the number of training instances

m = the number of features

We are given:

Training time for $n = 1000000$ instances is 1 hour.

Assume the number of features m remains the same, and now we estimate for $n = 10,000,000$ instances:

The time complexity scales with $n \log n$ (in this case since m is the same):

$$\frac{Time_{new}}{Time_{old}} = \frac{n \log n}{n_1 \log n_1} = \frac{10^7 \log 10^7}{10^6 \log 10^6} = \frac{10 \log 10^7}{\log 10^6} = \frac{10 \cdot 7 \log 10}{6 \log 10} = \frac{70}{6} \approx 11,67 \Rightarrow$$

$n_2 = 10,000,000$

$n_1 = 1,000,000$

$\Rightarrow Time_{new} = 11,67 \cdot Time_{old} = 11,67 \cdot 1 = 11,67\,h = 11\frac{2}{3}\,h$

Answer: 11h 40 minutes

Problem 3 Gini index, entropy, and classification errors are impurity measures used in decision tree algorithms to decide how to split data at each node. They help quantify how "mixed" or "pure" a node is.

Gini Index: it measures the probability of incorrectly classifying a randomly chosen element if it was randomly labeled according to the distribution of labels in the node. The greater the value of Gini index, the greater the chances of having misclassifications.

$$Gini = 1 - \sum_{i=1}^{c} (P_i)^2$$ where P_i = the frequency of class i at node t, and c is the total number of classes

(the example is in the next page)

Market Sentiment	Liquidity	Volatility	Return
Bullish	High	Low	Up
Bearish	Low	High	Down
Bullish	High	Low	Up
Bullish	Low	High	Down
Bearish	High	Low	Up
Bullish	Low	Low	Down
Bearish	High	High	Down
Bullish	Low	Low	Down
Bullish	High	Low	Up

The Gini index for a split is

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

where p_i = proportion of class i in the subset

- Calculating the Gini index for Market Sentiment (MS)

Bullish: 6 number of times out of 9
Bearish: 3 number of times out of 9
$\Rightarrow$ $P(MS = Bullish) = 6/9 = 2/3$
$P(MS = Bearish) = 3/9 = 1/3$

- $P(Return = Up \mid MS = Bullish) = \frac{P(Return = Up \cap MS = Bullish)}{P(MS = Bullish)} = \frac{\#(Return = Up \cap MS = Bullish)}{\#(MS = Bullish)} = \frac{3}{6} = \frac{1}{2}$
- $P(Return = Down \mid MS = Bullish) = \frac{\#(Return = Down \cap MS = Bullish)}{\#(MS = Bullish)} = \frac{3}{6} = \frac{1}{2}$

$\Rightarrow$ Gini Index $= 1 - \left[\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2\right] = 1 - \left[\frac{1}{4} + \frac{1}{4}\right] = 1 - \frac{1}{2} = \frac{1}{2}$

- $P(Return = Up \mid MS = Bearish) = \frac{\#(Return = Up \cap MS = Bearish)}{\#(MS = Bearish)} = \frac{1}{3}$
- $P(Return = Down \mid MS = Bearish) = \frac{\#(Return = Down \cap MS = Bearish)}{\#(MS = Bearish)} = \frac{2}{3}$

$\Rightarrow$ Gini Index $= 1 - \left[\left(\frac{1}{3}\right)^2 + \left(\frac{2}{3}\right)^2\right] = 1 - \left(\frac{1}{9} + \frac{4}{9}\right) = 1 - \frac{5}{9} = \frac{4}{9}$

Gini Index (for MS) $= P(MS = Bullish) \cdot \frac{1}{2} + P(MS = Bearish) \cdot \frac{4}{9} = \frac{2}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{4}{9} = \frac{1}{3} + \frac{4}{27} = \frac{13}{27} \simeq 0.48148$ $\Rightarrow$

$\Rightarrow$ Gini Index for MS $= 0.48148$

- Calculating the Gini Index for Liquidity (L)

High: 5 number out of 9
Low: 4 number out of 9
$\Rightarrow$ $P(L = High) = 5/9$
$P(L = Low) = 4/9$

- $P(Return = Up \mid L = High) = \frac{\#(Return = Up \text{ and } L = High)}{\#(L = High)} = \frac{4}{5}$
- $P(Return = Down \mid L = High) = \frac{\#(Return = Down \cap L = High)}{\#(L = High)} = \frac{1}{5}$

$\Rightarrow$ Gini Index $= 1 - \left[\left(\frac{4}{5}\right)^2 + \left(\frac{1}{5}\right)^2\right] = 1 - \left(\frac{16}{25} + \frac{1}{25}\right) = \frac{8}{25}$

- $P(Return = Up \mid L = Low) = \frac{\#(Return = Up \cap L = Low)}{\#(L = Low)} = \frac{0}{4} = 0$
- $P(Return = Down \mid L = Low) = \frac{\#(Return = Down \cap L = Low)}{\#(L = Low)} = \frac{4}{4} = 1$

⇒ Gini Index $= 1 - (1 + 0) = 0$

Gini Index (for liquidity) $= \frac{5}{9} \cdot \frac{8}{25} + \frac{4}{9} \cdot 0 = \frac{8}{45} \simeq 0,1777$ ⇒ Gini Index for liquidity $= 0.1777$

- Calculating the Gini Index for Volatility (V)

High: 3 number out of 9
Low: 6 number out of 9
⇒ $P(V = High) = \frac{1}{3}$, $P(V = Low) = \frac{2}{3}$

- $P(Return = Up \mid V = High) = \frac{\#(Return = Up \cap V = High)}{\#(V = High)} = \frac{0}{3} = 0$
- $P(Return = Down \mid V = High) = \frac{\#(Return = Down \cap V = High)}{\#(V = High)} = \frac{3}{3} = 1$

⇒ Gini Index $= 1 - (0^2 + 1^2) = 0$

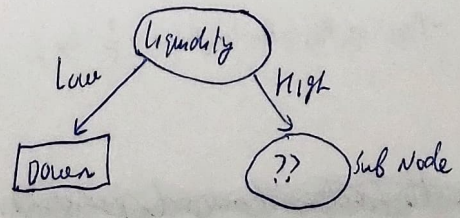
- $P(Return = Up \mid V = Low) = \frac{\#(Return = Up \cap V = Low)}{\#(V = Low)} = \frac{4}{6} = \frac{2}{3}$
- $P(Return = Down \mid V = Low) = \frac{\#(Return = Down \cap V = Low)}{\#(V = Low)} = \frac{2}{6} = \frac{1}{3}$

⇒ Gini Index $= 1 - \left(\left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2\right) = 1 - \left(\frac{4}{9} + \frac{1}{9}\right) =$
$= 1 - \frac{5}{9} = \frac{4}{9}$

Gini Index (for Volatility) $= \frac{1}{3} \cdot 0 + \frac{2}{3} \cdot \frac{4}{9} = \frac{8}{27} \simeq 0,2962$ ⇒ Gini Index for Volatility $\simeq 0,2962$

Features	Gini Index
Market sentiment	0,48148
Liquidity	0,1777
Volatility	0,2962

Since Liquidity has the lowest Gini-Index value, we will use it to split the dataset as follows:



We will repeat the same procedure to determine the sub-nodes or branches of the decision tree.
We will calculate the Gini Index for the "High" branch of Liquidity as follows.

Market Sentiment	Liquidity	Volatility	Return
Bullish	High	Low	UP
Bullish	High	Low	UP
Bearish	High	Low	UP
Bearish	High	High	Down
Bullish	High	Low	UP

And we will do the same process on this new table, and continue building the decision tree.

- **Entropy:** Entropy is a measure of impurity or randomness in a dataset. It can be defined as a method to measure the impurity or uncertainty. A system or model with lowest entropy is considered is considered better than the other with high entropy.

The mathematical formula for entropy is defined as:

$$E(S) = -\sum_{i=1}^{k} P_i \log_2 P_i$$ where S is a dataset (or node)
k is the number of classes
P_i is the probability of a class label i in our data.

For example, if our dataset has 2 classes Yes and No, then the entropy can be calculated as:

$$E(S) = -P_{yes} \log_2 P_{yes} - P_{no} \log_2 P_{no}$$

Example

Income ($)	Experience (years)	loan approval
30,000	2	YES
15,000	10	NO
70,000	5	YES
20,000	8	NO
25,000	1	NO

Calculate Entropy $= -P_{YES} \log_2 P_{YES} - P_{NO} \log_2 P_{NO} = -\frac{2}{5} \log \frac{2}{5} - \frac{3}{5} \log \frac{3}{5} = 0,97$

- **Classification Error:**

Classification Error is a measure of the proportion of misclassified instances in a dataset. In decision trees, it is used to evaluate the impurity of a node by considering how often the majority class fails to represent all the data points.

Formula: $\text{classification Error} = 1 - \max_{i=1,2,\dots,k} P_i$ where k = # of classes
P_i = the proportion of classes belonging to class i.

Consider a dataset:

Class	Count
A	7
B	3

$P_A = \frac{7}{10} = 0,7$

$P_B = \frac{3}{10} = 0,3$

$\max\{P_A, P_B\} = \max\{0,7; 0,3\} = 0,7$

$E = 1 - 0,7 = 0,3$ ← classification error

Problem 4

To extract decision rules, we can use the following methods:

- Direct Method: Direct method extract rules directly from the training data without first building an intermediate model like a decision tree or NN.

These use seperate-and-conquer strategies (also called covering algorithms)
Each rule is generated to cover as many examples of one class as possible without excluding others.
The covered examples are removed, and the process is repeated.

Direct Method: Sequential Covering

1) Start from an empty rule
2) Grow a rule using the learn-one-rule function
3) Remove training records covered by the rule.
4) Repeat step 2) and 3) until stopping criteria is met.

- Indirect Method:

Indirect methods extract rules from an already trained model, such as decision trees, random forest, neural network, or SVM.

It first builds a predictive model then analyzes the model's internal structure or outputs to generate human-readable rules.

Decision Rule Induction is a method for extracting if-then rules from data, typically used for classification tasks. These rules are designed to be interpretable, actionable, and easy to apply. One example of decision rule induction algorithm is RIPPER. It generates a set of if-then classification rules directly from labeled data and is particularly good for datasets with imbalanced classes.

We should use the decision rule induction method, when we need clear, understandable logic (i.e. medical diagnosis, credit scoring, legal decisions, etc). And when the rules can be quickly applied to new data with little computational cost.

Problem 5

- Calculating the Gini-index for Age:

youth: 5 number out of 14
middle-aged: 4 number out of 14
senior: 5 number out of 14

$\Rightarrow \begin{cases} P(age=youth) = 5/14 \\ P(age=middle\text{-}aged) = 4/14 \\ P(age=senior) = 5/14 \end{cases}$

- $P(Buys\_computer = yes \mid age = youth) = \frac{\#(Buys\_computer = yes \wedge age = youth)}{\#(age = youth)} = \frac{2}{5}$
- $P(Buys\_computer = no \mid age = youth) = \frac{\#(Buys = no \wedge age = youth)}{\#(age = youth)} = \frac{3}{5}$

$\Rightarrow Gini\text{-}Index = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2 = 1 - \left(\frac{4}{25} + \frac{9}{25}\right) = 1 - \frac{13}{25} = \boxed{\frac{12}{25}}$

- $P(Buys = yes \mid age = middle\text{-}aged) = \frac{\#(Buys = yes \wedge age = middle\text{-}aged)}{\#(age = middle\text{-}aged)} = \frac{4}{4} = 1$
- $P(Buys = no \mid age = middle\text{-}aged) = \frac{\#(Buys = no \wedge age = middle\text{-}aged)}{\#(age = middle\text{-}aged)} = \frac{0}{4} = 0$

$\Rightarrow Gini\text{-}Index = 1 - 0 - 1 = 0$

- $P(Buys = yes \mid age = senior) = \frac{\#(Buys = yes \wedge age = senior)}{\#(age = senior)} = \frac{3}{5}$
- $P(Buys = no \mid age = senior) = \frac{\#(Buys = no \wedge age = senior)}{\#(age = senior)} = \frac{2}{5}$

$\Rightarrow Gini\text{-}Index = 1 - \left(\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2\right) = 1 - \frac{13}{25} = \frac{12}{25}$

$Weighted\ Gini\text{-}Index\ for\ age = \frac{5}{14} \cdot \frac{12}{25} + \frac{4}{14} \cdot 0 + \frac{5}{14} \cdot \frac{12}{25} = \frac{5}{7} \cdot \frac{12}{25} = \frac{12}{35} \approx 0,34285$

- Calculating the Gini-Index for Income

high: 4 number out of 14
medium: 6 out of 14
low: 4 out of 14

$\Rightarrow \begin{cases} P(income=high) = 4/14 \\ P(income=medium) = 6/14 \\ P(income=low) = 4/14 \end{cases}$

- $P(Buys = yes \mid income=high) = \frac{\#(Buys=yes \wedge income=high)}{\#(income=high)} = \frac{2}{4}$
- $P(Buys=no \mid income=high) = \frac{\#(Buys = no \wedge income=high)}{\#(income=high)} = \frac{2}{4}$

$\Rightarrow Gini\text{-}Index = 1 - \left(\left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2\right) = 1 - \frac{1}{2} = \frac{1}{2}$

- $P(\text{buys} = \text{yes} \mid \text{income} = \text{medium}) = \frac{\#(\text{buys}=\text{yes} \wedge \text{income}=\text{medium})}{\#(\text{income}=\text{medium})} = \frac{4}{6}$

- $P(\text{buys}=\text{no} \mid \text{income}=\text{medium}) = \frac{\#(\text{buys}=\text{no} \wedge \text{income}=\text{medium})}{\#(\text{income}=\text{medium})} = \frac{2}{6}$

$\Rightarrow$ Gini-Index $= 1-\left(\left(\frac{4}{6}\right)^2+\left(\frac{2}{6}\right)^2\right) = 1-\left(\frac{4}{9}+\frac{1}{9}\right) = \frac{4}{9}$

- $P(\text{buys}=\text{yes} \mid \text{income}=\text{low}) = \frac{\#(\text{buys}=\text{yes} \wedge \text{income}=\text{low})}{\#(\text{income}=\text{low})} = \frac{3}{4}$

- $P(\text{buys}=\text{no} \mid \text{income}=\text{low}) = \frac{\#(\text{buys}=\text{no} \wedge \text{income}=\text{low})}{\#(\text{income}=\text{low})} = \frac{1}{4}$

$\Rightarrow$ Gini-Index $= 1-\left(\left(\frac{3}{4}\right)^2+\left(\frac{1}{4}\right)^2\right) = 1-\frac{10}{16} = \frac{3}{8}$

Weighted Gini-Index for Income $= \frac{4}{14}\cdot\frac{1}{2}+\frac{6}{14}\cdot\frac{4}{9}+\frac{4}{14}\cdot\frac{3}{8} = \frac{1}{7}+\frac{4}{21}+\frac{3}{28} = \frac{12+16+9}{84} = \frac{37}{84} \approx 0,4407$

- Calculating the Gini-Index for student

yes : 7 out of 14:
no : 7 out of 14:

$\Rightarrow$ $P(\text{student}=\text{yes}) = \frac{1}{2}$, $P(\text{student}=\text{no}) = \frac{1}{2}$

- $P(\text{buys}=\text{yes} \mid \text{student}=\text{yes}) = \frac{\#(\text{buys}=\text{yes} \wedge \text{student}=\text{yes})}{\#(\text{student}=\text{yes})} = \frac{6}{7}$

- $P(\text{buys}=\text{no} \mid \text{student}=\text{yes}) = \frac{\#(\text{buys}=\text{no} \wedge \text{student}=\text{yes})}{\#(\text{student}=\text{yes})} = \frac{1}{7}$

$\Rightarrow$ Gini-Index $= 1-\left(\left(\frac{6}{7}\right)^2+\left(\frac{1}{7}\right)^2\right) = 1-\frac{37}{49} = \frac{12}{49}$

- $P(\text{buys}=\text{yes} \mid \text{student}=\text{no}) = \frac{\#(\text{buys}=\text{yes} \wedge \text{student}=\text{no})}{\#(\text{student}=\text{no})} = \frac{3}{7}$

- $P(\text{buys}=\text{no} \mid \text{student}=\text{no}) = \frac{\#(\text{buys}=\text{no} \wedge \text{student}=\text{no})}{\#(\text{student}=\text{no})} = \frac{4}{7}$

$\Rightarrow$ Gini-Index $= 1-\left(\left(\frac{3}{7}\right)^2+\left(\frac{4}{7}\right)^2\right) = 1-\frac{25}{49} = \frac{24}{49}$

Weighted Gini-Index for student $= \frac{1}{2}\cdot\frac{12}{49}+\frac{1}{2}\cdot\frac{24}{49} = \frac{1}{2}\cdot\frac{36}{49} = \frac{18}{49} \approx 0,3673$

• Calculating the Gini-Index for Credit-Rating

fair: 8 out of 14
excellent: 6 out of 14
$\Rightarrow$ P(credit = fair) = 8/14
P(credit = excellent) = 6/14

• $P(buys=yes \mid credit=fair) = \frac{\#(buys=yes \wedge credit=fair)}{\#(credit=fair)} = \frac{6}{8}$

• $P(buys=no \mid credit=fair) = \frac{\#(buys=no \wedge credit=fair)}{\#(credit=fair)} = \frac{2}{8}$

$\Rightarrow$ Gini-Index $= 1 - \left|\left(\frac{6}{8}\right)^2 + \left(\frac{2}{8}\right)^2\right| = 1 - \frac{5}{8} = \frac{3}{8}$

• $P(buys=yes \mid credit=excellent) = \frac{\#(buys=yes \wedge credit=excellent)}{\#(credit=excellent)} = \frac{3}{6}$

• $P(buys=no \mid credit=excellent) = \frac{\#(buys=no \wedge credit=excellent)}{\#(credit=excellent)} = \frac{3}{6}$

$\Rightarrow$ Gini-Index $= 1 - \left|\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2\right| = \frac{1}{2}$

Weighted Gini-Index for credit rating $= \frac{8}{14} \cdot \frac{3}{8} + \frac{6}{14} \cdot \frac{1}{2} = \frac{3}{14} + \frac{3}{14} = \frac{6}{14} \simeq 0,42857$

Now, we need to calculate the information gain

Step 1: Calculate Entropy for Parent

$Entropy(parent) = -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{9}{14}\log\frac{9}{14} - \frac{5}{14}\log\frac{5}{14} \simeq 0,94028$

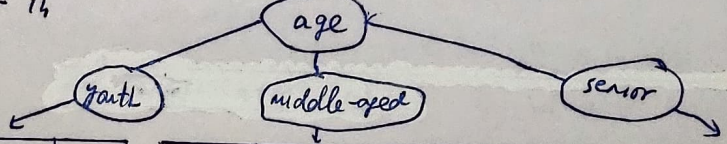
$P_{yes} = \frac{9}{14}$ / $P_{no} = \frac{5}{14}$

Step 2: Calculate Entropy for children

Age: $P(age=youth) = \frac{5}{14}$

$P(age=middle\text{-}aged) = \frac{4}{14}$

$P(age=senior) = \frac{5}{14}$



age	income	student	credit-rating	buys comp
youth	high	no	fair	no
youth	high	no	excellent	no
youth	medium	no	fair	no
youth	low	yes	fair	yes
youth	medium	yes	excellent	yes

age	income	student	credit rating	buys computer
middle-aged	high	no	fair	yes
middle-aged	low	yes	excellent	yes
middle-aged	medium	no	excellent	yes
middle-aged	high	yes	fair	yes

age	income	student	credit rating	buys computer
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
senior	medium	yes	fair	yes
senior	medium	no	excellent	no

Calculate Entropy for youth

- $Entropy(youth) = -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{2}{5}\log\frac{2}{5} - \frac{3}{5}\log\frac{3}{5} \approx 0.97095$

$P_{yes} = \frac{2}{5}$

$P_{no} = \frac{3}{5}$

Calculate Entropy for middle-aged

- $Entropy(middle\text{-}aged) = -P_{yes}\log_2 P_{yes} - 0 = 1\log 1 = 0$

$P_{yes} = \frac{4}{4} = 1$

$P_{no} = \frac{0}{4} = 0$

Calculate entropy for senior

- $Entropy(senior) = -\frac{3}{5}\log\frac{3}{5} - \frac{2}{5}\log\frac{3}{5} \approx 0,97095$

$P_{yes} = \frac{3}{5}$

$P_{no} = \frac{2}{5}$

Step3: Calculate average weighted entropy of each children:

Weighted $E(youth) = \frac{5}{14} \cdot 0,97095 \approx 0,34676$

Weighted $E(middle\text{-}aged) = \frac{4}{14} \cdot 0 = 0$

Weighted $E(senior) = \frac{5}{14} \cdot 0,97095 \approx 0,34676$

⇒ Average Weighted $E(age) = 0,34676 + 0 + 0,34676 = 0,69352$

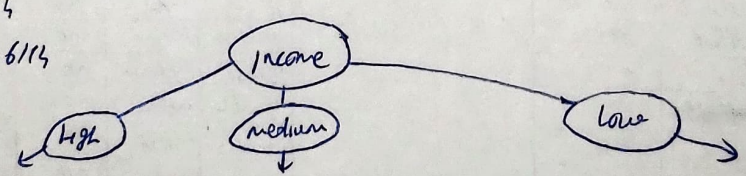
Step4: Calculate information gain IG)

$Gain(R, Age) = Gain(R) - \{Average\ Weighted\ E(age)\} = 0,94028 - 0,69352 = 0,24676$

Income: $P(income = high) = \frac{4}{14}$

Step2: $P(income = medium) = 6/14$

$P(income = low) = 4/14$



age	income	student	credit rating	Buys computer
youth	high	no	fair	no
youth	high	no	excellent	no
middle-aged	high	no	fair	yes
middle-aged	high	yes	fair	yes

age	income	student	credit rating	Buys computer
senior	medium	no	fair	yes
youth	medium	no	fair	no
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle-aged	medium	no	excellent	yes
senior	medium	no	excellent	no

age	income	student	credit rating	Buys computer
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle-age	low	yes	excellent	yes
youth	low	yes	fair	yes

Calculate Entropy for high

- Entropy(high) = $-P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{1}{2}\lg\frac{1}{2} - \frac{1}{2}\lg\frac{1}{2} = 1$

$P_{yes} = \frac{1}{2}$

$P_{no} = \frac{1}{2}$

Calculate Entropy for medium

- Entropy(medium) = $-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0,91829$

$P_{yes} = \frac{4}{6}$

$P_{no} = \frac{2}{6}$

Calculate Entropy for low

- Entropy(low) = $-\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} \approx 0,8127$

$P_{yes} = \frac{3}{4}$

$P_{no} = \frac{1}{4}$

Step3: Calculate average weighted entropy of each children.

Weighted E(high) = $\frac{4}{14} \cdot 1 = \frac{4}{14}$

Weighted E(medium) = $\frac{6}{14} \cdot 0,91829 = 0,39355$

Weighted E(low) = $\frac{4}{14} \cdot 0,81127 = 0,23179$

3) Average Weighted E(income) = $\frac{4}{14} + 0,39355 + 0,23179 = 0,91105$

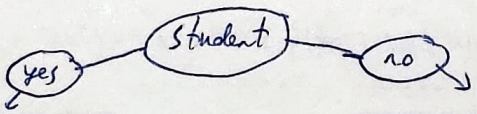
Step4: Calculate Information Gain. (IG)

Gain(R, income) = Gain(R) − (Average Weighted E(income)) = $0,94028 - 0,91105 = 0,02923$

---

student: P(student = yes) = $\frac{7}{14} = \frac{1}{2}$ P(student = no) = $\frac{7}{14} = \frac{1}{2}$

Step2



age	income	student	credit rating	Buys computer
senior	low	yes	fair	yes
senior	low	yes	excellent	no
middle	low	yes	excellent	yes
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
youth	medium	yes	excellent	yes
middle	high	yes	fair	yes

age	income	student	credit rating	Buys computer
youth	high	no	fair	no
youth	high	no	excellent	no
middle	high	no	fair	yes
senior	medium	no	fair	yes
youth	medium	no	fair	no
middle	medium	no	excellent	yes
senior	medium	no	excellent	no

Calculate Entropy for yes (student)

- Entropy(yes) = $-P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{6}{7}\log\frac{6}{7} - \frac{1}{7}\lg\frac{1}{7} = 0,59167$

$P_{yes} = \frac{6}{7}$

$P_{no} = \frac{1}{7}$

Calculate Entropy for no (student)

- Entropy(no) = $-\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} \approx 0,98522$

$P_{yes} = \frac{3}{7}$

$P_{no} = \frac{4}{7}$

Step3: Calculate average weighted entropy of each children

Weighted E(yes) = $\frac{1}{2} \cdot 0,59167 = 0,29583$

Weighted E(no) = $\frac{1}{2} \cdot 0,98522 = 0,49261$

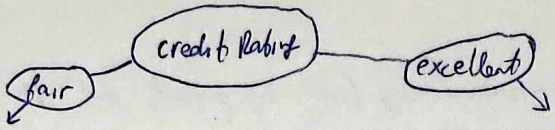
3) Average Weighted E(student) = $0,29583 + 0,49261 = 0,78845$

Step 4: Calculate Information Gain (IG)

Gain(R, student) = Gain(R) − {Average Weighted E(student)} = 0,94028 − 0,78845 = 0,151835

Credit-Rating: P(credit-rating = fair) = 8/14 P(credit-rating = excellent) = 6/14

Step 2:



age	income	student	credit rating	Buys computer
youth	high	no	fair	no
middle	high	no	fair	yes
senior	medium	no	fair	yes
senior	low	yes	fair	yes
youth	medium	no	fair	no
youth	low	yes	fair	yes
senior	medium	yes	fair	yes
middle	high	yes	fair	yes

age	income	student	credit rating	Buys computer
youth	high	no	excellent	no
senior	low	yes	excellent	no
middle	low	yes	excellent	yes
youth	medium	yes	excellent	yes
middle	medium	no	excellent	yes
senior	medium	no	excellent	no

Calculate Entropy for fair

• Entropy(fair) = $-\frac{3}{5}\log\frac{3}{5} - \frac{1}{5}\log\frac{1}{5} = 0,81127$

$P_{yes} = \frac{6}{8}$

$P_{no} = \frac{2}{8}$

• Entropy(excellent) = $-\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = 1$

$P_{yes} = \frac{1}{2}$

$P_{no} = \frac{1}{2}$

Step 3: Calculate average weighted entropy of each division

Weighted E(fair) = $\frac{8}{14} \cdot 0,81127 = 0,46358$

Weighted E(excellent) = $\frac{6}{14} \cdot 1 = \frac{6}{14}$

⇒ Average Weighted E(credit-Rating) = $0,46358 + \frac{6}{14} = 0,89215$

Step 4: Calculate Information Gain (IG)

Gain(R, credit-rating) = Gain(R) − {Average Weighted E(credit-Rating)} = 0,94028 − 0,89215 = 0,04813

We got the following Gini-Indices and Information-Gain for each features

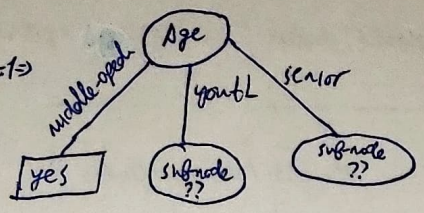
Gini-Index

Feature	Gini-Index	Info-Gain
Age	0,34285	0,24676
Income	0,44047	0,02923
Student	0,3673	0,151835
Credit R.	0,42857	0,04813

Age has the smallest Gini-Index (also the largest Information-Gain), therefore we use age as the splitting attribute for the first split.

When we split by Age feature, we get the following decision tree (so far)

we have $P(\text{Buys\_computer}=\text{yes} \mid \text{age}=\text{middle}) = 1 \Rightarrow$
$\Rightarrow$ we can build the sub-node middle-aged immediately



Now, we need to construct the subnodes coming from age=youth and age=senior. For that we need to split the dataset accordingly: (into smaller datasets)

1)

age	income	student	credit rating	Buys computer
youth	high	no	fair	no
youth	high	no	excellent	no
youth	medium	no	fair	no
youth	low	yes	fair	yes
youth	medium	yes	excellent	yes

2)

age	income	student	credit rating	Buys computer
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
senior	medium	yes	fair	yes
senior	medium	no	excellent	no

- Calculating Gini-Indices for (1) (age=youth) sub-dataset.

- Calculating the Gini-Index for Income

high: 2 out of 5
medium: 2 out of 5
low: 1 out of 5
$\Rightarrow$ $P(\text{income}=\text{high}) = 2/5$
$P(\text{income}=\text{medium}) = 2/5$
$P(\text{income}=\text{low}) = 1/5$

- $P(\text{Buys\_comp}=\text{yes} \mid \text{income}=\text{high}) = \frac{0}{2} = 0$
- $P(\text{Buy\_comp}=\text{no} \mid \text{income}=\text{high}) = \frac{2}{2} = 1$
$\Rightarrow$ $\text{Gini-Index} = 1-(0^2+1^2) = 0$

- $P(\text{Buys\_comp}=\text{yes} \mid \text{income}=\text{medium}) = \frac{1}{2}$
- $P(\text{Buys\_comp}=\text{no} \mid \text{income}=\text{medium}) = \frac{1}{2}$
$\Rightarrow$ $\text{Gini-Index} = 1-((\frac{1}{2})^2+(\frac{1}{2})^2) = \frac{1}{2}$

- $P(\text{Buys\_comp}=\text{yes} \mid \text{income}=\text{low}) = 1$
- $P(\text{Buys\_comp}=\text{no} \mid \text{income}=\text{low}) = 0$
$\Rightarrow$ $\text{Gini-Index} = 1-(0^2+1^2) = 0$

$\Rightarrow$ Weighted Gini-Index for Income
$= \frac{2}{5}\cdot 0 + \frac{1}{2}\cdot\frac{2}{5} + \frac{1}{5}\cdot 0 = \frac{1}{5} = 0,2$

- Calculating the Gini-Index for student

yes: 2 out of 5
no: 3 out of 5
$\Rightarrow$ $P(\text{student}=\text{yes}) = \frac{2}{5}$
$P(\text{student}=\text{no}) = \frac{3}{5}$

- $P(\text{Buys\_comp}=\text{yes} \mid \text{student}=\text{yes}) = \frac{2}{2} = 1$
- $P(\text{Buys\_comp}=\text{no} \mid \text{student}=\text{yes}) = \frac{0}{2} = 0$
$\Rightarrow$ $\text{Gini-Index} = 1-1^2+1^2 = 0$

- $P(\text{Buys\_comp}=\text{yes} \mid \text{student}=\text{no}) = \frac{0}{3} = 0$
- $P(\text{Buys\_comp}=\text{no} \mid \text{student}=\text{no}) = \frac{3}{3} = 1$
$\Rightarrow$ $\text{Gini-Index} = 1-(0^2+1^2) = 0$

$\Rightarrow$ Weighted Gini-Index for student $= \frac{2}{5}\cdot 0 + \frac{3}{5}\cdot 0 = 0$

• Calculating the Gini-Index for Credit-Rating

fair: 3 out of 5 | excellent: 2 out of 5 | ⇒ P(credit=fair)=3/5, P(credit=excellent)=2/5

• P(buys=yes | credit=fair)=1/3
• P(buys=no | credit=fair)=2/3
⇒ Gini-Index = 1−((1/3)²+(2/3)²) = 1−5/9 = 4/9

• P(buys=yes | credit=excellent)=1/2
• P(buys=no | credit=excellent)=1/2
⇒ Gini-Index = 1−((1/2)²+(1/2)²) = 1/2 ⇒ Weighted Gini Index for credit-Rating =

= 3/5 · 4/9 + 2/5 · 1/2 = 12/45 + 1/5 = 11/45 = 7/15 ≈ 0,4666

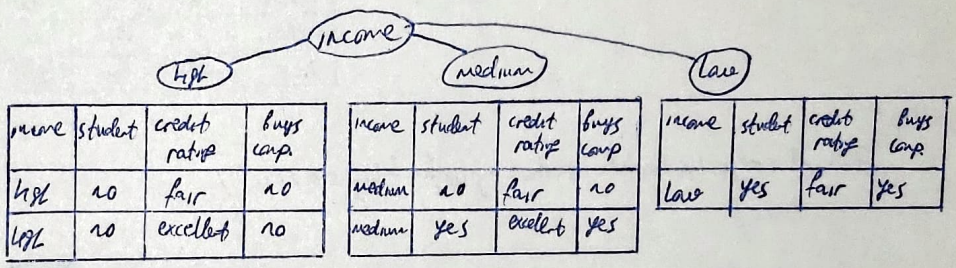
Now, we need to calculate the information-Gain

Step1 Calculate Entropy for parent

E(parent) = −P_yes log₂ P_yes − P_no log₂ P_no = −2/5 log 2/5 − 3/5 log 3/5 = 0,97095

Step2: Calculate Entropy for children

Income: P(income=high)=2/5
P(income=medium)=2/5
P(income=low)=1/5



income	student	credit rating	buys comp.
high	no	fair	no
high	no	excellent	no

income	student	credit rating	buys comp.
medium	no	fair	no
medium	yes	excellent	yes

income	student	credit rating	buys comp.
low	yes	fair	yes

• Entropy(high) = −P_yes log₂ P_yes − P_no log₂ P_no = 1 log 1 = 0
P_yes = 0
P_no = 1

• Entropy(medium) = −1/2 log 1/2 − 1/2 log 1/2 = 1.
P_yes = 1/2
P_no = 1/2

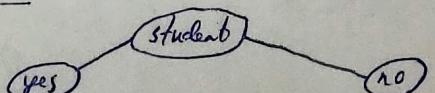
• E(low) = 1 log 1 = 0
P_yes = 1
P_no = 0

Step3: Weighted E(high) = 2/5 · 0 = 0
Weighted E(medium) = 2/5 · 1 = 2/5
Weighted E(low) = 1/5 · 0 = 0
⇒ Average Weighted Entropy(Income) = 0 + 2/5 + 0 = 2/5 = 0,4

Step4: Gain(Youth, Income) = Gain(Y) − {Average Weighted E(Income)} = 0.97095 − 0,4 = 0,57095

Student: Step2: Calculate Entropy for children

student P(student=yes) = 2/5
P(student=no) = 3/5



Income	Student	Credit rating	Buys comp.
low	yes	fair	yes
medium	yes	excellent	yes

Income	Student	Credit Rating	Buys comp.
high	no	fair	no
high	no	excellent	no
medium	no	fair	no

Step3:

- $E(yes) = -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = 1\lg 1 = 0$

  $P_{yes} = \frac{2}{2} = 1$

  $P_{no} = 0$

- $E(no) = 1\lg 1 + 0 = 0$

  $P_{yes} = 0$

  $P_{no} = 1$

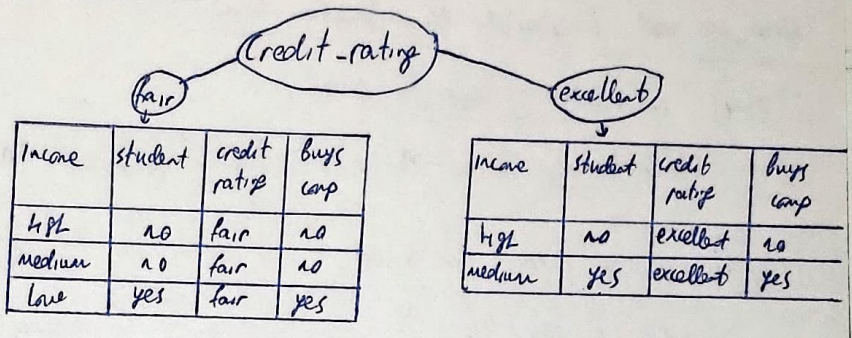
$\Rightarrow$ Average Weighted E(student) $= \frac{2}{5}\cdot 0 + \frac{3}{5}\cdot 0 = 0$

Step4: I.Gain(Youth, student) = Gain(Y) − Gain(student) = 0,97095 − 0 = 0,97095

Step2: Calculate Entropy for children.

Credit_Rating $P(\text{credit}=\text{fair}) = \frac{3}{5}$

$P(\text{credit}=\text{excellent}) = \frac{2}{5}$



Income	student	credit rating	buys comp
high	no	fair	no
medium	no	fair	no
low	yes	fair	yes

Income	student	credit rating	buys comp
high	no	excellent	no
medium	yes	excellent	yes

Step3:

$E(fair) = -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{1}{3}\lg\frac{1}{3} - \frac{2}{3}\lg\frac{2}{3} = 0,91829$

$P_{yes} = \frac{1}{3}$

$P_{no} = \frac{2}{3}$

$E(excellent) = -\frac{1}{2}\lg\frac{1}{2} - \frac{1}{2}\lg\frac{1}{2} = 1$

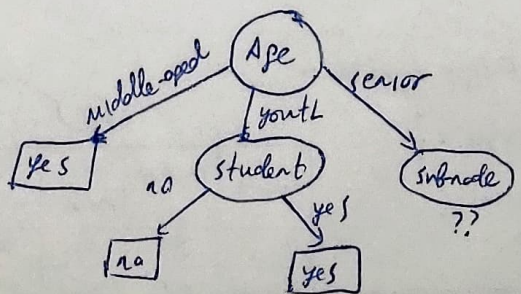
$\Rightarrow$ Average Weighted E(credit) $= 0,91829\cdot\frac{3}{5} + \frac{2}{5}\cdot 1 = 0,950974$

Step4: I.Gain(Y, credit-rating) = Gain(Y) − Gain(Credit) = 0,97095 − 0,950974 = 0,019976

We got the following Gini-index and InfoGain:

feature	Gini-Index	InfoGain
Income	0,2	0,57095
student	0	0,97095
credit-rat.	0,46667	0,019976

$\Rightarrow$ student feature has the smallest Gini-Index (Also the highest IG=0,97095), hence we take student as the splitting attribute for the subnode Youth:



Also, we can say $P(buys = yes \mid (age = youth \text{ and } student = yes)) = 1$

$P(buys = no \mid (age = youth \text{ and } student = no)) = 0$

$\Rightarrow$ we can construct the whole student sub-branch

Calculating Gini-Indices for (2) (age = senior) sub-dataset

(2)

age	income	student	credit rating	Buys computer
senior	medium	no	fair	yes
senior	low	yes	fair	yes
senior	low	yes	excellent	no
senior	medium	yes	fair	yes
senior	medium	no	excellent	no

- Calculating the Gini-Index for income

high: 0 out of 5
medium: 3 out of 5
low: 2 out of 5

⇒ P(income = high) = 0/5 = 0
P(income = medium) = 3/5 = 0,6
P(income = low) = 2/5 = 0,4

- P(Buys = yes | income = high) = 0
- P(Buys = no | income = high) = 0 (not included in the dataset)

- P(Buys = yes | income = medium) = 2/3
- P(Buys = no | income = medium) = 1/3

⇒ Gini-Index = $1 - (\left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2) = 1 - \frac{5}{9} = \frac{4}{9}$

- P(Buys = yes | income = low) = 1/2
- P(Buys = no | income = low) = 1/2

⇒ Gini-Index $= 1 - (\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2) = \frac{1}{2}$

⇒ Weighted-Gini-Index = $0,6 \cdot \frac{4}{9} + 0,4 \cdot \frac{1}{2}$ (=)

(=) $\frac{3}{5} \cdot \frac{4}{9} + \frac{2}{5} \cdot \frac{1}{2} = \frac{4}{15} + \frac{1}{5} = \frac{7}{15} \approx 0,46667$

- Calculating the Gini-Index for student

yes: 3 out of 5
no: 2 out of 5

⇒ P(student = yes) = 3/5
P(student = no) = 2/5

- P(Buys = yes | student = yes) = 2/3
- P(Buys = no | student = yes) = 1/3

⇒ Gini-Index $= 1 - (\left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2) = \frac{4}{9}$

- P(Buys = yes | student = no) = 1/2
- P(Buys = no | student = no) = 1/2

⇒ Gini-Index $= 1 - (\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2) = \frac{1}{2}$

(=)

(=) Weighted Gini-Index (for student) $= \frac{3}{5} \cdot \frac{4}{9} + \frac{2}{5} \cdot \frac{1}{2} = \frac{4}{15} + \frac{1}{5} = \frac{7}{15} \approx 0,46667$

- Calculating the Gini-Index for Credit-Rating

fair: 3 out of 5
excellent: 2 out of 5

⇒ P(credit = fair) = 3/5
P(credit = excellent) = 2/5

- P(Buys = yes | credit = fair) = 3/3 = 1
- P(Buys = no | credit = fair) = 0/3 = 0

⇒ Gini-Index = $1 - 0^2 - 1^2 = 0$

- P(Buys = yes | credit = excellent) = 0
- P(Buys = no | credit = excellent) = 1

⇒ Gini-Index = $1 - 0^2 - 1^2 = 0$

(=)

(=) Weighted Gini-Index (for Credit-rating) $= \frac{3}{5} \cdot 0 + \frac{2}{5} \cdot 0 = 0$

Now, we calculate Information Gains for Senior-Dataset's features.

Step1: Calculate entropy for parent:

$$\text{Entropy(parent)} = -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} \approx 0,97095$$

$P_{yes} = \frac{3}{5}$

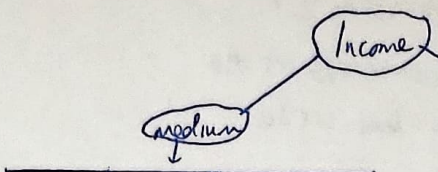
$P_{no} = \frac{2}{5}$

Step2: Calculate for children.

Income: $P(\text{income}=high) = 0$

$P(\text{income}=medium) = \frac{3}{5}$

$P(\text{income}=low) = \frac{2}{5}$



Income	student	credit rating	Buys computer
medium	no	fair	yes
medium	yes	fair	yes
medium	no	excellent	no

Income	student	credit rating	Buys computer
low	yes	fair	yes
low	yes	excellent	no

- Entropy(high) = 0
- Entropy(medium) $= -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0,91829$
  $P_{yes} = \frac{2}{3}$
  $P_{no} = \frac{1}{3}$
- Entropy(low) $= -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$
  $P_{yes} = \frac{1}{2}$
  $P_{no} = \frac{1}{2}$

Step3:

Weighted E(medium) $= \frac{3}{5} \cdot 0,91829 \approx 0,550974$

Weighted E(low) $= \frac{2}{5} \cdot 1 = \frac{2}{5}$

$\Rightarrow$ Average Weighted Entropy(Income) $= 0,550974 + 0,4 = 0,950974$

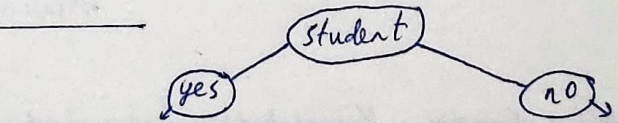
Step4: Calculate Information Gain

$$\text{I.Gain(Senior, income)} = \text{Gain}(S) - \{\text{Average Weighted E(income)}\} = 0,97095 - 0,950974 = 0,019976$$

Step2: Calculate entropy for children

Student: $P(\text{student}=yes) = \frac{3}{5}$

$P(\text{student}=no) = \frac{2}{5}$



Income	student	credit rating	Buys computer
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes

Income	student	credit rating	Buys computer
medium	no	fair	yes
medium	no	excellent	no

- Entropy(yes) $= -P_{yes}\log_2 P_{yes} - P_{no}\log_2 P_{no} = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} \approx 0,91829$
  $P_{yes} = \frac{2}{3}$
  $P_{no} = \frac{1}{3}$
- Entropy(no) $= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2} = 1$
  $P_{yes} = \frac{1}{2}$
  $P_{no} = \frac{1}{2}$

Step3:

Weighted E(yes) $= \frac{3}{5} \cdot 0,91829 \approx 0,550974$

Weighted E(no) $= \frac{2}{5} \cdot 1 = \frac{2}{5}$

$\Rightarrow$ Average Weighted Entropy(student) $= 0,550974 + 0,4 = 0,950974$

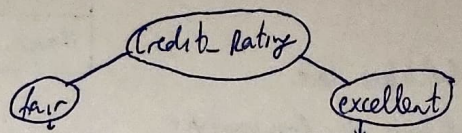
Step 4: Calculate Information Gain.

I. Gain(senior, student) = Gain(S) − {Average Weighted E(student)} = 0,97095 − 0,950974 = 0,019976

Step 2: Calculate Entropy for children

Credit_rating P(credit=fair) = 3/5

P(credit=excellent) = 2/5



Income	student	credit rating	buys computer
medium	no	fair	yes
low	yes	fair	yes
medium	yes	fair	yes

Income	student	credit rating	buys computer
low	yes	excellent	no
medium	no	excellent	no

- Entropy(fair) = 1 log 1 = 0
  - $p_{yes} = 1$
  - $p_{no} = 0$
- Entropy(excellent) = 0
  - $p_{no} = 1$
  - $p_{yes} = 0$

Step 3:

⇒ Average Weighted Entropy(credit) = 3/5 · 0 + 2/5 · 0 = 0

Weighted E(fair) = 0

Weighted E(excellent) = 0

Step 4: Calculate Information Gain

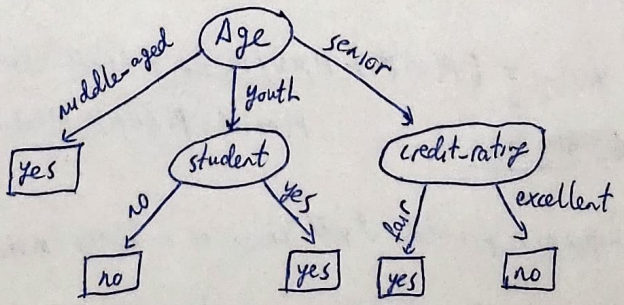
Information_Gain(senior, credit_rating) = Gain(S) − {Average Weighted E(credit)} (=)

(=) 0,97095 − 0 = 0,97095

We got the following Gini-Indices and Information Gains

feature	Gini-Index	Info-Gain
income	0,46667	0,019976
student	0,46667	0,019976
credit_rating	0	0,97095

Credit_Rating feature has the smallest Gini Index = 0 (also the largest Information Gain), therefore we use credit_rating as the splitting attribute for the subnode senior:



← Final Decision tree

From the table, we can say: P(buys=yes | (age=senior and credit=yes)) = 1

P(buys=no | (age=senior and credit=no)) = 1

⇒ we can build (fully) the subBranch credit_rating.

Answer: We got the decision tree!

b)

$P(buys\_computer = yes) = \frac{9}{14}$

$P(buys\_computer = no) = \frac{5}{14}$

Age	YES	NO
youth	2/9	3/5
middle-aged	4/9	0/5
senior	3/9	2/5

Income	YES	NO
High	2/9	2/5
Medium	4/9	2/5
Low	3/9	1/5

Student	YES	NO
yes	6/9	1/5
No	3/9	4/5

Credit rating	YES	NO
fair	6/9	2/5
Excellent	3/9	3/5

The prior probabilities here are the probabilities when buys_comp=yes and when buys_computer=no:

Prior probabilities:

$P(buys\_computer = yes) = \frac{9}{14}$

$P(buys\_computer = no) = \frac{5}{14}$

The above tables are tables of conditional probabilities.

For example, for age: $P(age = youth | buys = yes) = \frac{2}{9}$

$P(age = youth | buys = no) = \frac{3}{5}$

$P(age = middle | buys = yes) = 4/9$

$P(age = middle | buys = no) = 0$

$P(age = senior | buys = yes) = 3/9$

$P(age = senior | buys = no) = 2/5$

Now, we need to classify a test instance.

Denote $X = \{ age = A_1, income = A_2, student = A_3, credit\_rating = A_4 \}$

Since A_1 represents age $\Rightarrow A_1 \in \{youth, middle\text{-}aged, senior\}$

With the same principle: $A_2 \in \{high, medium\}$

$A_3 \in \{yes, no\}$

$A_4 \in \{fair, excellent\}$

- Now, we need to calculate posterior probabilities i.e. calculate $P(yes|X)$ and $P(no|X)$.

The posterior probability $P(yes|X)$ is calculated using Bayes' Theorem, which in Naïve Bayes simplifies to $P(yes|X) = \frac{P(X|yes) \cdot P(yes)}{P(X)}$

$P(yes|X) = P(buys\_computer = yes | X) = \overset{\text{Bayes}}{\uparrow} P(X|yes) \cdot P(yes) / P(X) = P(A_1|yes) \cdot P(A_2|yes) \cdot P(A_3|yes) \cdot P(A_4|yes) \cdot P(yes) / P(X)$

$P(no|X) = P(A_1|no) \cdot P(A_2|no) \cdot P(A_3|no) \cdot P(A_4|no) \cdot P(no) / P(X)$

Since no specific test instance is provided, I will choose an arbitrary instance:

let $X = \{ age = youth, income = medium, student = yes, credit\_rating = fair \}$

Calculate $P(yes|X)$: $P(yes|X) = P(youth|yes) \cdot P(medium|yes) \cdot P(yes|yes) \cdot P(fair|yes) \cdot P(yes) = \frac{2}{9} \cdot \frac{4}{9} \cdot \frac{6}{9} \cdot \frac{6}{9} \cdot \frac{9}{14} \approx 0,0282 / P(X)$

Calculate $P(no|X)$: $P(no|X) = P(youth|no) \cdot P(medium|no) \cdot P(yes|no) \cdot P(fair|no) \cdot P(no) = \frac{3}{5} \cdot \frac{2}{5} \cdot \frac{1}{5} \cdot \frac{2}{5} \cdot \frac{5}{14} / P(X) = 0,0069 / P(X)$

$P(yes|X) = 0,0282 / P(X) > 0,0069 / P(X) = P(no|X) \Rightarrow$ Classify X as yes (in this case).

c) RIPPER is a rule-based classification algorithm that generates a set of IF-THEN rules to make prediction. It is particularly effective for datasets with categorical attributes and binary class labels. RIPPER creates rules for one class by adding conditions that maximize accuracy, then it simplifies each rule by removing conditions that don't contribute to predictive performance. The algorithm repeatedly refines the rules to reduce overfitting and improve generalization.

RIPPER is especially useful when interpretability and concise rule sets are needed, Decision Trees provide a visual and hierarchical understanding of decision-making. Naive Bayes is efficient and performs surprisingly well on many datasets, though it relies on the assumption that attributes are independent.

In our case, we have 14 records in our dataset that is very small, all attributes are categorical, the target is binary, therefore RIPPER will perform the best because it's designed for rule-based classification on categorical data. It might extract rules like:

IF Age = middle-aged THEN buys_computer = yes
or IF student = yes AND credit_rating = fair THEN buys_computer = yes

The second place is Decision Tree. It is still performing well on categorical and small datasets, but without careful pruning, the tree might overfit the small training set.

The third place is Naive Bayes: It assumes independence among features, which doesn't hold for this dataset. Also, the dataset is small, which can make the model unstable.

Problem 6 In the left figure, the model is too simple. It only uses three trees and has a very high learning rate = 1. As a result, it seems to jump too quickly to conclusions and doesn't capture the pattern in the data very well. The predictions are rigid which misses the smooth curvature that the data has. This is underfitting i.e. the model hasn't had enough complexity or time to learn the structure in the data. Possible solution is to increase the number of trees, for example use 50 or 100 trees to learn more complex patterns and lowering the learning rate (to 0,1 or 0,05) to allow for more refined learning.

In the right figure the model does a good job at first place. It uses a low learning rate = 0,1 and a large number of estimators (trees) = 200, which allows it to gradually fit the data. But this model might be overfitting. We see it because it picks up the noise in the data, we can lower the number of trees a bit (making 100-no trees).

Problem 7 Ensemble classification combines the predictions of multiple base models to improve overall performance. In Ensemble classification we use Hard Voting and Soft Voting.

In Hard Voting, each base classifier predicts a class label, and the final prediction is the one that receives the most votes. Each model makes its own independent decision and vote for a class.

a) Most voted class is the winner
b) If there is a tie, then soft voting can be used, or weights can be assigned to the classifiers or odd number of classifiers can be used in ensemble.

Soft Voting is an ensemble technique for classification where multiple models don't vote on class labels, instead they provide the class probabilities for each class.

The final prediction is made by averaging the predicted probabilities for each class and selecting the class with the highest average.

Bagging stands for Bootstrap Aggregating, and it's an ensemble method that builds multiple methods on random subsets of the training data with replacement.

First, it randomly samples with replacement from the training set to create several new subsets of the data. Second, train a separate model on each sample of the dataset. Further, it aggregates their predictions. Since sampling is with replacement, some samples may appear multiple times, and some may be left out in each subset.

Its advantages are: 1) reduces overfitting 2) Easy to parallelize 3) Stable and robust

~~Loop~~ one of popular examples is Random Forest.

Boosting refers to any Ensemble method that can combine several weak learners into a strong learner. The general idea of most boosting methods is to train predictors sequentially, each trying to correct its predecessor.

The most popular boosting methods are AdaBoost, Gradient Boosting and Extreme gradient boosting.

The process

1) Train the first model on the training data.
2) Evaluate Errors and give higher weights to misclassified samples.
3) Train the next model to focus more on difficult examples
4) Combine models using weighted voting or gradient optimization.

key feature is that the models are trained sequentially, each one correcting its predecessor.

Its advantages include high predictive power, reducing both bias and variance often winning ML competitors.

Problem 8: Naive Bayes is a probabilistic classification algorithm based on Bayes' Theorem with the assumption that features are conditionally independent given the class. The theorem is:

Bayes Theorem: $P(y|X) = \frac{P(X|y) \cdot P(y)}{P(X)}$ where $P(y|X)$ is the posterior probability of y given X.

$P(y)$ = prior probability of class y

$P(X)$ = marginal probability of features X.

Prior Probability is the initial probability of an event before seeing any evidence. In our case, it's the probability of "play-football=YES" or "play-football=no" without considering any weather conditions.

Posterior probability is the probability of an event after any evidence. In our case, it's the probability of "Play-Football=YES" or "Play-Football=no" with given specific conditions.

To find the posterior probability of X,

$X = \{$Outlook=sunny, Temp=Hot, Humidity=High, Wind=Weak$\}$

We need to calculate $P(yes|X) = P(X|yes) \cdot P(yes) / P(X) = P(sunny|yes) \cdot P(Hot|yes) \cdot P(High|yes) \cdot P(weak|yes) \cdot P(yes) / P(X)$

By Bayes Theorem

we don't need to find $P(X)$ because it is in the product of both $P(yes|X)$ and $P(no|X)$ and we can compare them.

$P(sunny|yes) = \frac{\#(\text{Outlook}=sunny \cap play=yes)}{\#(play=yes)} = \frac{1}{3}$

$P(Hot|yes) = \frac{\#(Temp=Hot \cap play=yes)}{\#(play=yes)} = \frac{0}{3} = 0$

$P(High|yes) = \frac{\#(Humidity=high \cap play=yes)}{\#(play=yes)} = \frac{1}{3}$

$P(weak|yes) = \frac{\#(Wind=weak \cap play=yes)}{\#(play=yes)} = \frac{2}{3}$

$\Rightarrow P(yes|X) = \frac{1}{3} \cdot 0 \cdot \frac{1}{3} \cdot \frac{2}{3} \cdot P(yes)$ $\forall P(X) = 0 \Rightarrow$ **$P(yes|X) = 0$**

$P(no|X) = P(sunny|no) \cdot P(Hot|no) \cdot P(High|no) \cdot P(weak|no) \cdot P(no) = \frac{2}{4} \cdot \frac{2}{4} \cdot \frac{3}{4} \cdot \frac{1}{4} \cdot P(no)/P(X) = \frac{1}{32} \cdot \frac{4}{7} / P(X) = \frac{1}{56} / P(X)$ ①

$P(sunny|no) = \frac{2}{4}$

$P(Hot|no) = \frac{2}{4}$

$P(High|no) = \frac{3}{4}$

$P(weak|no) = \frac{1}{4}$

$(P(no) = \frac{4}{7}, P(yes) = \frac{3}{7})$

① **$P(no|X) = \frac{1}{56} / P(X)$**

In our case $P(no|X) = \frac{1}{56} / P(X) > 0 = P(yes|X) \Rightarrow$ the naive Bayes classifier will predict "No" for X.

page

Problem 9 Here we will use the Hamming Distance $d_H(x,y) = \sum_{i=1}^{n} \delta(x_i, y_i)$ where $\delta(x_i,y_i) = \begin{cases} 0 & \text{if } x_i = y_i \\ 1 & \text{if } x_i \neq y_i \end{cases}$

It's the count of mismatches.

k=2: We will chose random initial centroids: **Cluster 1:** Row 1 - (Sunny, Hot, High, Weak) $= c_1$

Cluster 2: Row 3 - (Overcast, Cold, Normal, Strong) $= c_2$

- Now, we need to calculate the distances between the datapoints and assign each one its cluster.

Row 1: $d_H(c_1, x_1) = 0$ (c_1 := cluster 1)
$d_H(c_2, x_1) = 1+1+1+1 = 4$ | $0 < 4 \Rightarrow$ Assign x_1 to c_1

Row 2: $d_H(c_1, x_2) = 0+1+1+0 = 2$
$d_H(c_2, x_2) = 1+1+0+1 = 3$ | $\Rightarrow 2 < 3 \Rightarrow$ Assign x_2 to c_1
(c_2 := cluster 2)
(row 3)

Row 3: $d_H(c_1, x_3) = 1+1+1+1 = 4$
$d_H(c_2, x_3) = 0+0+0+0 = 0$ | $\Rightarrow 0 < 4 \Rightarrow$ Assign x_3 to c_2

Row 4: $d_H(c_1, x_4) = 0+1+1+1 = 3$
$d_H(c_2, x_4) = 1+1+0+0 = 2$ | $\Rightarrow 2 < 3 \Rightarrow$ Assign x_4 to c_2

Row 5: $d_H(c_1, x_5) = 1+1+0+0 = 2$
$d_H(c_2, x_5) = 1+1+1+1 = 4$ | $\Rightarrow 2 < 4 \Rightarrow$ Assign x_5 to c_1

Row 6: $d_H(c_1, x_6) = 1+0+1+1 = 3$
$d_H(c_2, x_6) = 0+1+0+0 = 1$ | $\Rightarrow 1 < 3 \Rightarrow$ Assign x_6 to c_2

Row 7: $d_H(c_1, x_7) = 1+1+0+1 = 3$
$d_H(c_2, x_7) = 1+1+1+0 = 3$ | $\Rightarrow 3 = 3 \Rightarrow$ let's assign to c_1

As a result, we got **Cluster 1:** Rows: 1, 2, 5, 7

Cluster 2: Rows: 3, 4, 6

- Now, we need to update Centroids: For cluster 1 and 2 we got the following tables

rows	Outlook	Temperature	Humidity	Wind
1	Sunny	Hot	High	Weak
2	Sunny	Warm	Normal	Weak
5	Rainy	Warm	High	Weak
7	Rainy	Warm	High	Strong

	Outlook	Temperature	Humidity	Wind
3	Overcast	Cold	Normal	Strong
4	Sunny	Warm	Normal	Strong
6	Overcast	Hot	Normal	Strong

From the tables in the previous page, we need to update centroids.

Cluster 1 — Outlook: {sunny, sunny, rainy, rainy} ⇒ Mode = Sunny
Temperature: {Hot, warm, warm, warm} ⇒ Mode = warm
Humidity: {High, Normal, High, High} ⇒ Mode = High
Wind: {weak, weak, weak, strong} ⇒ Mode = weak
⇒ New C_1 = (Sunny, Warm, High, Weak)

Cluster 2: Outlook: {Overcast, Sunny, Overcast} ⇒ Mode = Overcast
Temperature: {Cold, Warm, Hot} ⇒ let's pick Mode = Cold
Humidity: {Normal, normal, normal} ⇒ Mode = Normal
Wind: {Strong, Strong, Strong} ⇒ Mode = Strong
⇒ New C_2 = {Overcast, Cold, Normal, Strong}

• Reassign points

Row 1: $d_H(C_1, x_1) = 0+1+0+0 = 1$
$d_H(C_2, x_1) = 1+1+1+1 = 4$
⇒ $1 < 4$ ⇒ Assign x_1 to C_1

Row 2: $d_H(C_1, x_2) = 0+0+1+0 = 1$
$d_H(C_2, x_2) = 1+1+0+1 = 3$
⇒ $1 < 3$ ⇒ Assign x_2 to C_1

Row 3: $d_H(C_1, x_3) = 1+1+1+1 = 4$
$d_H(C_2, x_3) = 0+0+0+0 = 0$
⇒ $0 < 4$ ⇒ Assign x_3 to C_2

Row 4: $d_H(C_1, x_4) = 0+0+1+1 = 2$
$d_H(C_2, x_4) = 1+1+0+0 = 2$
⇒ $2 = 2$ ⇒ let's assign x_4 to C_2 (original cluster)

Row 5: $d_H(C_1, x_5) = 1+0+0+0 = 1$
$d_H(C_2, x_5) = 1+1+1+1 = 4$
⇒ $1 < 4$ ⇒ Assign x_5 to C_1

Row 6: $d_H(C_1, x_6) = 1+1+1+1 = 4$
$d_H(C_2, x_6) = 0+1+1+0 = 1$
⇒ $1 < 4$ ⇒ Assign x_6 to C_2

Row 7: $d_H(C_1, x_7) = 1+0+0+1 = 2$
$d_H(C_2, x_7) = 1+1+1+0 = 3$
⇒ $2 < 3$ ⇒ Assign x_7 to C_1

As a result: we got Cluster 1: 1, 2, 5, 7
Cluster 2: 3, 4, 6
⇒ Same as previous iteration ⇒ Algorithm has converged

The final clusters for k=2

Cluster 1

Rows	Outlook	Temperature	Humidity	Wind
1	Sunny	Hot	High	Weak
2	Sunny	Warm	Normal	Weak
5	Rainy	Warm	High	Weak
7	Rainy	Warm	High	Strong

last centroid C_1 = {Sunny, Warm, High, Weak}

Cluster 2

Rows	Outlook	Temperature	Humidity	Wind
3	overcast	Cold	Normal	Strong
4	Sunny	Warm	Normal	Strong
6	overcast	Hot	Normal	Strong

last centroid C_2 = {Overcast, Cold, Normal, Strong}

---

k=3: We choose random initial centroids:

Centroid for Cluster 1: C_1 = Row 1 = {Sunny, Hot, High, Weak}

Centroid for Cluster 2: C_2 = Row 3 = {overcast, Cold, Normal, Strong}

Centroid for Cluster 3: C_3 = Row 5 = {Rainy, Warm, High, Weak}

- Calculate the distances and assign clusters.

Row 1: $d_H(C_1, X_1) = 0+0+0+0 = 0$
$d_H(C_2, X_1) = 1+1+1+1 = 4$
$d_H(C_3, X_1) = 1+1+0+0 = 2$
⇒ $0 < 2 < 4$ ⇒ Assign X_1 to C_1

Row 2: $d_H(C_1, X_2) = 0+1+1+0 = 2$
$d_H(C_2, X_2) = 1+1+0+1 = 3$
$d_H(C_3, X_2) = 1+0+1+0 = 2$
⇒ $2 = 2 < 3$ ⇒ Assign X_2 to C_1

Row 3: $d_H(C_1, X_3) = 1+1+1+1 = 4$
$d_H(C_2, X_3) = 0+0+0+0 = 0$
$d_H(C_3, X_3) = 1+1+1+1 = 4$
⇒ $0 < 4 = 4$ ⇒ Assign X_3 to C_2

Row 4: $d_H(C_1, X_4) = 0+1+1+1 = 3$
$d_H(C_2, X_4) = 1+1+0+0 = 2$
$d_H(C_3, X_4) = 1+0+1+1 = 3$
⇒ $2 < 3 = 3$ ⇒ Assign X_4 to C_2

Row 5: $d_H(C_1, X_5) = 1+1+0+0 = 2$
$d_H(C_2, X_5) = 1+1+1+1 = 4$
$d_H(C_3, X_5) = 0+0+0+0 = 0$
⇒ $0 < 2 < 4$ ⇒ Assign X_5 to C_3

Row 6: $d_H(C_1, X_6) = 1+0+1+1 = 3$
$d_H(C_2, X_6) = 0+1+0+0 = 1$
$d_H(C_3, X_6) = 1+1+1+1 = 4$
⇒ $1 < 3 < 4$ ⇒ Assign X_6 to C_2

Row 7: $d_H(C_1, X_7) = 1+1+0+1 = 3$
$d_H(C_2, X_7) = 1+1+1+0 = 3$
$d_H(C_3, X_7) = 0+0+0+1 = 1$
⇒ $1 < 3$ ⇒ ⑤
⑤ Assign X_7 to C_3

As a result for 1 iteration, we get:

Cluster 1 Points 1, 2

Cluster 2: points: 3, 4, 6

Cluster 3 points 5, 7

We got the following clusters so far:

	Outlook	Temperature	Humidity	Wind
1	Sunny	Hot	High	Weak
2	Sunny	Warm	Normal	Weak

cluster 1

	Outlook	Temperature	Humidity	Wind
3	Overcast	Cold	Normal	Strong
4	Sunny	Warm	Normal	Strong
6	Overcast	Hot	Normal	Strong

cluster 2

	Outlook	Temperature	Humidity	Wind
5	Rainy	Warm	High	Weak
7	Rainy	Warm	High	Strong

Cluster 3

- Now, we update the centroids.

Cluster 1: Outlook: {Sunny, Sunny} => Mode = Sunny

Temperature: {Hot, Warm} => let's choose Mode = Hot

Humidity: {High, Normal} => let's choose Mode = High

Wind: {Weak, Weak} => Mode = Weak

=> Updated C_1 = {Sunny, Hot, High, Weak} =>

=> updated C_1 is the same as C_1 (unchanged)

Cluster 2: Outlook: {Overcast, Sunny, Overcast} => Mode = Overcast

Temperature: {Cold, Warm, Hot} => let's choose Mode = Cold (as in previous)

Humidity: {Normal, Normal, Normal} => Mode = Normal

Wind: {Strong, Strong, Strong} => Mode = Strong

=>

=> Updated C_2 = {Overcast, Cold, Normal, Strong} => updated C_2 is equal to the previous C_2 (unchanged)

Cluster 3: Outlook: {Rainy, Rainy} => Mode = Rainy

Temperature: {Warm, Warm} => Mode = Warm

Humidity: {High, High} => Mode = High

Wind: {Weak, Strong} => Mode = Weak

=> Updated C_3 = {Rainy, Warm, High, Weak} =>

=> updated C_3 is the same as C_3 (unchanged)

- We got, that after the first iteration, the cluster centroids remain unchanged from our initial selection, therefore we can say that the algorithm converges.

The final clusters

cluster 1

	Outlook	Temperature	Humidity	Wind
1	Sunny	Hot	High	Weak
2	Sunny	Warm	Normal	Weak

cluster 2

	Outlook	Temperature	Humidity	Wind
3	Overcast	Cold	Normal	Strong
4	Sunny	Warm	Normal	Strong
6	Overcast	Hot	Normal	Strong

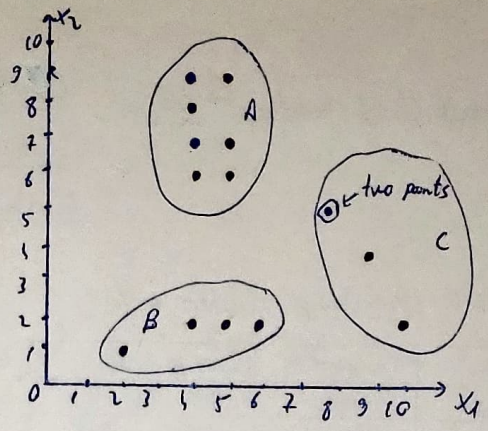
cluster 3

	Outlook	Temperature	Humidity	Wind
5	Rainy	Warm	High	Weak
7	Rainy	Warm	High	Strong

Ties

Problem 10

To choose the appropriate number of clusters k, we can first plot the datapoints and calculate approximate number of groupings. (since the data is small and is from $\mathbb{R}^2$)



From the graph, we can assume that $k=3$ is a good choice (at first glance)

Step 1: Random Initialization of Centroids.

Centroid A = (4, 8)

Centroid B = (4, 2)

Centroid C = (9, 4)

- Now, we need to calculate the distance (Euclidean) of each datapoints from centroids and do the grouping.

1) $a_1 = (10, 2)$
$d(A, a_1) = \sqrt{(10-4)^2 + (8-2)^2} = \sqrt{72} = 6\sqrt{2}$
$d(B, a_1) = \sqrt{(10-4)^2 + (2-2)^2} = \sqrt{36} = 6$
$d(C, a_1) = \sqrt{(10-9)^2 + (2-4)^2} = \sqrt{5}$
$\Rightarrow \sqrt{5} < 6 < 6\sqrt{2} \Rightarrow$ Assign (10, 2) to C

2) $a_2 = (9, 4)$
$d(A, a_2) = \sqrt{(9-4)^2 + (4-8)^2} = \sqrt{41}$
$d(B, a_2) = \sqrt{(9-4)^2 + (4-2)^2} = \sqrt{29}$
$d(C, a_2) = \sqrt{(9-9)^2 + (4-4)^2} = 0$
$\Rightarrow 0 < \sqrt{29} < \sqrt{41} \Rightarrow$ Assign (9, 4) to C

3) $a_3 = a_4 = (8, 5)$
$d(A, a_3) = \sqrt{(8-4)^2 + (5-8)^2} = 5$
$d(B, a_3) = \sqrt{(8-4)^2 + (5-2)^2} = 5$
$d(C, a_3) = \sqrt{(8-9)^2 + (5-4)^2} = \sqrt{2}$
$\Rightarrow \sqrt{2} < 5 = 5 \Rightarrow$ Assign (8, 5) to C

5) $(6, 2) = a_5$
$d(A, a_5) = \sqrt{(6-4)^2 + (2-8)^2} = \sqrt{40}$
$d(B, a_5) = \sqrt{(4-6)^2 + (2-2)^2} = 2$
$d(C, a_5) = \sqrt{(9-6)^2 + (4-2)^2} = \sqrt{13}$
$\Rightarrow 2 < \sqrt{13} < \sqrt{40} \Rightarrow$ Assign a_5 to B

6) $a_6 = (5, 2)$
$d(A, a_6) = \sqrt{(4-5)^2 + (8-2)^2} = \sqrt{37}$
$d(B, a_6) = \sqrt{(5-4)^2 + (2-2)^2} = 1$
$d(C, a_6) = \sqrt{(9-5)^2 + (4-2)^2} = \sqrt{20}$
$\Rightarrow 1 < \sqrt{20} < \sqrt{37} \Rightarrow$ Assign a_6 to B

7) $a_7 = (4, 2)$
$d(A, a_7) = \sqrt{(4-4)^2 + (8-2)^2} = 6$
$d(B, a_7) = \sqrt{(4-4)^2 + (2-2)^2} = 0$
$d(C, a_7) = \sqrt{(9-4)^2 + (4-2)^2} = \sqrt{29}$
$\Rightarrow 0 < \sqrt{29} < 6 \Rightarrow$ Assign (4, 2) to B

8) $a_8 = (7,1)$: $d(A,a_8) = \sqrt{(4-7)^2+(8-1)^2} = \sqrt{53}$
$d(B,a_8) = \sqrt{(4-7)^2+(2-1)^2} = \sqrt{5}$
$d(C,a_8) = \sqrt{(12-7)^2+(4-1)^2} = \sqrt{58}$
$\Rightarrow \sqrt{5} < \sqrt{53} < \sqrt{58} \Rightarrow$ Assign $(7,1)$ to B

9) $a_9 = (5,6)$ $d(A,a_9) = \sqrt{(4-5)^2+(8-6)^2} = \sqrt{5}$
$d(B,a_9) = \sqrt{(4-5)^2+(2-6)^2} = \sqrt{17}$
$d(C,a_9) = \sqrt{(15-9)^2+(6-4)^2} = \sqrt{40}$
$\Rightarrow \sqrt{5} < \sqrt{17} < \sqrt{40} \Rightarrow$ Assign $(5,6)$ to A

10) $a_{10} = (4,6)$ $d(A,a_{10}) = \sqrt{(4-4)^2+(8-6)^2} = 2$
$d(B,a_{10}) = \sqrt{(4-4)^2+(2-6)^2} = 4$
$d(C,a_{10}) = \sqrt{(4-9)^2+(4-6)^2} = \sqrt{29}$
$\Rightarrow 2 < 4 < \sqrt{29} \Rightarrow$ Assign $(4,6)$ to A

11) $a_{11} = (5,7)$ $d(A,a_{11}) = \sqrt{(4-5)^2+(8-7)^2} = \sqrt{2}$
$d(B,a_{11}) = \sqrt{(5-4)^2+(2-7)^2} = \sqrt{26}$
$d(C,a_{11}) = \sqrt{(12-5)^2+(4-7)^2} = 5$
$\Rightarrow \sqrt{2} < 5 < \sqrt{26} \Rightarrow$ Assign $(5,7)$ to A

12) $a_{12} = (4,7)$ $d(A,a_{12}) = \sqrt{(4-4)^2+(8-7)^2} = 1$
$d(B,a_{12}) = \sqrt{(4-4)^2+(2-7)^2} = 5$
$d(C,a_{12}) = \sqrt{(12-4)^2+(4-7)^2} = \sqrt{73}$
$\Rightarrow 1 < 5 < \sqrt{73} \Rightarrow$ Assign $(4,7)$ to A

13) $a_{13} = (5,9)$ $d(A,a_{13}) = \sqrt{(5-4)^2+(9-8)^2} = \sqrt{2}$
$d(B,a_{13}) = \sqrt{(5-4)^2+(9-2)^2} = \sqrt{50}$
$d(C,a_{13}) = \sqrt{(9-5)^2+(4-9)^2} = \sqrt{41}$
$\Rightarrow \sqrt{2} < \sqrt{41} < \sqrt{50} \Rightarrow$ Assign $(5,9)$ to A

14) $a_{14} = (4,8)$ $d(A,a_{14}) = 0$
$d(B,a_{14}) = \sqrt{(4-4)^2+(8-2)^2} = 6$
$d(C,a_{14}) = \sqrt{(9-4)^2+(4-8)^2} = \sqrt{41}$
$\Rightarrow 0 < 6 < \sqrt{41} \Rightarrow$ Assign $(4,8)$ to A

15) $a_{15} = (4,9)$ $d(a_{15},A) = \sqrt{(4-4)^2+(8-9)^2} = 1$
$d(B,a_{15}) = \sqrt{(4-4)^2+(2-9)^2} = 7$
$d(C,a_{15}) = \sqrt{(9-4)^2+(4-9)^2} = 5$
$\Rightarrow 1 < 5 < 7 \Rightarrow$ Assign a_{15} to A

After the first iteration, we got the clusters: $A = \{(5,6), (4,6), (5,7), (4,7), (5,9), (4,8), (4,9)\}$
$B = \{(6,2), (5,2), (4,2), (7,1)\}$
$C = \{(11,2), (9,4), (8,5), (8,5)\}$

Now we calculate the means for each cluster to get new centroids.

A: $x\text{-mean} = \frac{5+4+5+4+5+4+4}{7} = 4,4$

$y\text{-mean} = \frac{6+6+7+7+8+9+9}{7} = 7,4$

⇒ New Centroid - A = (4,4 ; 7,4)

B: $x\text{-mean} = \frac{6+5+4+2}{4} = 4,25$

$y\text{-mean} = \frac{7+7+2+4}{4} = 1,75$

⇒ New Centroid - B = (4,25 ; 1,75)

C $x\text{-mean} = \frac{10+8+8+8}{4} = 8,75$

$y\text{-mean} = \frac{2+4+5+5}{4} = 4$

⇒ New Centroid - C = (8,75 ; 4).

- Now we need to again calculate each distance and regroup.

1) $a_1 = (10,2)$ $d(A,a_1) \simeq \sqrt{(4,4-10)^2+(7,4-2)^2} \simeq 7,78$
$d(B,a_1) \simeq 5,76$
$d(C,a_1) = 2,36$
⇒ $2,36 < 5,76 < 7,78$ ⇒ Assign to $^{(10,2)}$C

2) $a_2 = (9,4)$ $d(A,a_2) = 5,74$
$d(B,a_2) = 5,27$
$d(C,a_2) = 0,75$
⇒ $0,75 < 5,27 < 5,74$ ⇒ Assign to $^{(9,4)}$C

3) $a_3 = a_4 = (8,5)$ 4)
$d(A,a_3) = 4,32$
$d(B,a_3) = 4,95$
$d(C,a_3) = 1,25$
⇒ $1,25 < 4,32 < 4,95$ ⇒ Assign (8,5) to C

5) $a_5 = (6,2)$ $d(A,a_5) = 5,63$
$d(B,a_5) = 1,76$
$d(C,a_5) = 3,39$
⇒ $1,76 < 3,39 < 5,63$ ⇒ Assign (6,2) to B

6) $a_6 = (5,2)$ $d(A,a_6) = 5,43$
$d(B,a_6) = 0,79$
$d(C,a_6) = 4,28$
⇒ $0,79 < 4,28 < 5,43$ ⇒ Assign (5,2) to B

7) $a_7 = (4,2)$ $d(A,a_7) = 5,41$
$d(B,a_7) = 0,354$
$d(C,a_7) = 5,16$
⇒ $0,354 < 5,16 < 5,41$ ⇒ Assign (4,2) to B

8) $a_8 = (7,1)$ $d(A,a_8) \approx 6,83$
$d(B,a_8) \approx 2,37$ / 5) $2,37 < 6,83 \Rightarrow$ Assign (7,1) to B
$d(C,a_8) \approx 7,39$

9) $a_9 = (5,6)$ $d(A,a_9) \approx 1,52$
$d(B,a_9) \approx 4,31$ / 5) Assign (5,6) to A
$d(C,a_9) \approx 4,26$

10) $a_{10} = (4,6)$ $d(A,a_{10}) \approx 1,458$
$d(B,a_{10}) \approx 4,26$ / 5) Assign (4,6) to A
$d(C,a_{10}) \approx 5,16$

11) $a_{11} = (5,7)$ $d(A,a_{11}) \approx 0,724$
$d(B,a_{11}) \approx 5,30$ / 5) Assign (5,7) to A
$d(C,a_{11}) \approx 4,77$

12) $a_{12} = (4,7)$ $d(A,a_{12}) \approx 0,566$
$d(B,a_{12}) \approx 5,26$ / 5) Assign (4,7) to A
$d(C,a_{12}) \approx 5,63$

13) $a_{13} = (5,9)$ $d(A,a_{13}) \approx 1,71$
$d(B,a_{13}) \approx 7,29$ / 5) Assign (5,9) to A
$d(C,a_{13}) \approx 6,28$

14) $a_{14} = (4,8)$ $d(A,a_{14}) \approx 0,721$
$d(B,a_{14}) \approx 6,25$ / 5) Assign (4,8) to A
$d(C,a_{14}) \approx 6,72$

15) $a_{15} = (4,9)$ $d(A,a_{15}) \approx 1,648$
$d(B,a_{15}) \approx 7,25$ / 5) Assign (4,9) to A
$d(C,a_{15}) \approx 6,89$

- After second iteration, we got the clusters $A = \{(5,6), (4,6), (5,7), (4,7), (5,9), (4,8), (4,9)\}$

$B = \{(5,2), (4,2), (6,2), (2,1)\}$

$C = \{(10,2), (2,4), (8,5), (8,8)\}$

Which are the same clusters we got in the previous iteration $\Rightarrow$ Algorithm converges

Final Clusters

A: {(5,8), (4,8), (5,7), (4,7), (5,9), (6,8), (4,9)}

B: {(6,2), (5,2), (4,2), (7,1)}

C: {(10,2), (3,4), (8,5), (8,5)}.

To close the appropriate number of clusters k, we can use the Elbow Method which involves running k-means for a range of k values (for example k=1-10) and plotting the WCSS (within-Cluster Sum of Squares) for each. The elbow point is where the WCSS curve starts to flatten.