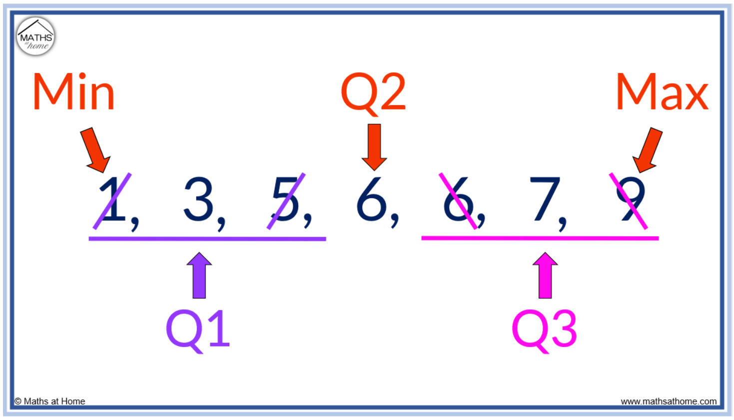
Data Mining Numerical

Basic Formula

$$\bar{A} = \frac{\Sigma A_i}{n}$$

$$\sigma = \sqrt{\frac{\Sigma(A_i - \bar{A})^2}{Nor(n-1)}}$$

Five Number Summary



Min-Max Normalization

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

▼ Q. Min-Max Normalization 8, 10, 15, and 20:

Before	After
8	0
10	0.167
15	0.583
20	1

Normalization by Decimal Scale

$$v_i' = \frac{v_i}{10^j}$$

▼ Q. Apply Normalization using Decimal Scale. 5000, 10000, 20000, 500000, 2500000:

vi	vi’
5000	0.0005
10000	0.001
20000	0.002
500000	0.05
2500000	0.25

Z-Score Normalization

$$v_i' = \frac{v_i - \bar{A}}{\sigma_A}$$

▼ Q. Implement Z-Score Normalization for the data 80, 10, 15, 20, and 30:

Mean = 31 and SD = 25.377

Before	After
80	1.931
10	-0.829
15	-0.630
20	-0.433
30	-0.039

Apriori Algorithm

▼ Q. Solve the following using Apriori Algorithm with minimum support count as 2:

TID	List of item_IDs
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

C_1

Items	Support Count
I1	6
I2	7
I3	6
I4	2
I5	2

L_1

Items	Support Count
I1	6
I2	7
I3	6
I4	2
I5	2

C_2: (Candidate 2-itemset)

Items	Support Count
I1, I2	4
I1, I3	4
I1, I4	1
I1, I5	2
I2, I3	4
I2, I4	2
I2, I5	2
I3, I4	0
I3, I5	1
I4, I5	0

L_2: (Frequent 2-itemset)

Items	Support Count
I1, I2	4
I1, I3	4
I1, I5	2
I2, I3	4
I2, I4	2
I2, I5	2

C_3:

L_3:

Items	Support Count
I1, I2, I3	2
I1, I2, I5	2
I1, I3, I5	1
I2, I3, I4	0
I2, I3, I5	1
I2, I4, I5	0

Items	Support Count
I1, I2, I3	2
I1, I2, I5	2

C_4:

Items	Support Count
I1, I2, I3, I5	1

L_4:

As candidate 4-itemset support count = 1, which doesn't satisfy the minimum support count hence, L4 = Φ.

$$Confidence(A \Rightarrow B) = \frac{SupportCount(A \cup B)}{SupportCount(A)}$$

▼ Q. Continuing with the previous example, we have got the frequent itemset L_3:

L_3: {{I1, I2, I5}, {I1, I2, I3}}

Consider X = {I1, I2, I5}

The non-empty subsets of X are {{I1, I2}, {I1, I5}, {I2, I5}, {I1}, {I2}, {I5}}

Let the minimum confidence threshold be 70%

Rule 1) Confidence({I1, I2} ⇒ {I5}) = $\frac{SupportCount(\{I_1,I_2\}\cup\{I_5\})}{SupportCount(\{I_1,I_2\})}$ = $\frac{2}{4}$ = 50%

Rule 2) Confidence({I1, I5} ⇒ {I2}) = 2/2 = 100%

Rule 3) Confidence({I2, I5} ⇒ {I1}) = 2/2 = 100%

Rule 4) Confidence({I1} ⇒ {I2, I5}) = 2/6 = 33.33%

Rule 5) Confidence({I2} ⇒ {I1, I5}) = 2/7 = 28.57%

Rule 6) Confidence({I5} ⇒ {I1, I2}) = 2/2 = 100%

Only three rules are strongly associated.

Consider X = {I1, I2, I3}

non-empty subsets of X are {{I1, I2}, {I1, I3}, {I2, I3}, {I1}, {I2}, {I3}}

Rule 1) Confidence({I1, I2} ⇒ {I3}) = 2/4 = 50%

Rule 2) Confidence({I1, I3} ⇒ {I2}) = 2/4 = 50%

Rule 3) Confidence({I2, I3} ⇒ {I1}) = 2/4 = 50%

Rule 4) Confidence({I1} ⇒ {I2, I3}) = 2/6 = 33.33%

Rule 5) Confidence({I2} ⇒ {I1, I3}) = 2/7 = 28.57%

Rule 6) Confidence({I3} ⇒ {I1, I2}) = 2/6 = 33.33%

None of the rules are strongly associated.

Vertical Data Format

▼ Q. Solve the following using Vertical Data Format Algorithm with minimum support count as 2:

TID	List of items
T100	I1, I2, I5
200	I2, I4
300	I2, I3
400	I1, I2, I4
500	I1, I3

TID	List of items
600	I2, I3
700	I1, I3
800	I1, I2, I3, I5
900	I1, I2, I3

C_1:

Items	TID_sets
I1	{T100, T400, T500, T700, T800, T900}
I2	{100, 200, 300, 400, 600, 800, 900}
I3	{300, 500, 600, 700, 800, 900}
I4	{200, 400}
I5	{100, 800}

L_1:

Items	TID_sets
I1	{T100, T400, T500, T700, T800, T900}
I2	{100, 200, 300, 400, 600, 800, 900}
I3	{300, 500, 600, 700, 800, 900}
I4	{200, 400}
I5	{100, 800}

C_2:

2-itemsets	Transactions set items
{I1, I2}	{100, 400, 800, 900}
{I1, I3}	{500, 700, 800, 900}
{I1, I4}	{400}
{I1, I5}	{100, 800}
{I2, I3}	{300, 600, 800, 900}
{I2, I4}	{200, 400}
{I2, I5}	{100, 800}
{I3, I4}	{Φ}
{I3, I5}	{800}
{I4, I5}	{Φ}

L_2:

2-itemsets	Transactions set items
{I1, I2}	{100, 400, 800, 900}
{I1, I3}	{500, 700, 800, 900}
{I1, I5}	{100, 800}
{I2, I3}	{300, 600, 800, 900}
{I2, I4}	{200, 400}
{I2, I5}	{100, 800}

C_3:

3-itemsets	Transactions set items
{I1, I2, I3}	{800, 900}
{I1, I2, I5}	{100, 800}
{I1, I3, I5}	{800}
{I2, I3, I4}	{Φ}
{I2, I3, I5}	{800}
{I2, I4, I5}	{Φ}

L_3

3-itemsets	Transactions set items
{I1, I2, I3}	{800, 900}
{I1, I2, I5}	{100, 800}

C_4

4-itemsets	Transactions set items
{I1, I2, I3, I5}	{800}

We will consider L_3 as our solution

FP-Growth Algorithm

▼ Q. Implement FP-Growth algorithm to find out frequent item, given support count = 2. Also construct FP-Tree.

T_ID	List of Items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3

T_ID	List of Items
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

<u>Solution:</u>

Step 1: Find support count of each item

Items	Support Count
I1	6
I2	7
I3	6
I4	2
I5	2

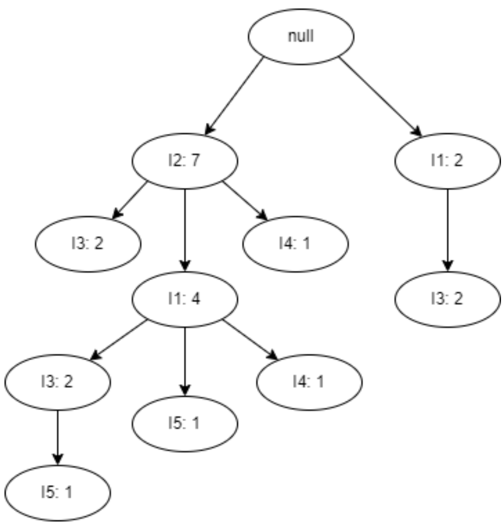
Step 2: Rearrange in descending order of support count

Items	Support Count
I2	7
I1	6
I3	6
I4	2
I5	2

T_ID	List of Items
T100	I2, I1, I5
T200	I2, I4
T300	I2, I3
T400	I2, I1, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I2, I1, I3, I5
T900	I2, I1, I3

Step 3: Construct FP-Tree as follows:

1. Create null root node.
2. Create branch for each transaction.
3. Increase the node count if it the transaction is repeated.



Step 4: Mining FP-Tree by creating conditional(sub) pattern basis:

Items	Conditional Pattern Base	Conditional FP-Tree	Frequent Pattern Generated
I5	{I2, I1, I3, I5: 1}, {I2, I1: 1}	{I2: 2, I1: 2}	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{I2: 1}, {I2, I1: 1}	{I2: 2}	{I2, I4: 2}
I3	{I2: 2}, {I2, I1: 2}, {I1: 2}	{I2: 4, I1: 2}, {I1: 2}	{I2, I3: 4}, {I1, I3: 2}, {I2, I1, I3: 2}
I1	{I2: 4}	{I2: 4}	{I2, I1: 4}
I2			

Gini Index

$$G(S) = 1 - \sum P_i^2 \qquad\qquad G(S) = (\frac{n_1}{S})G(s_1) + (\frac{n_2}{S})G(s_2)$$

▼ Q. Consider the following table to implement decision tree using Gini index:

Sr. No.	Own's Home	Married	Gender	Employee	Credit Rating	Risk Class
1	Yes	Yes	Male	Yes	A	B
2	No	No	Female	Yes	A	A
3	Yes	Yes	Female	Yes	B	C
4	Yes	No	Male	No	B	B
5	No	Yes	Female	Yes	B	C
6	No	No	Female	Yes	B	A
7	No	No	Male	No	B	B
8	Yes	No	Female	Yes	A	A
9	No	Yes	Female	Yes	A	C
10	Yes	Yes	Female	Yes	A	C

We consider an artificial example of building a decision tree to classify bank loan applications by assigning applications to one of the three risk classes.

Solution:

There are 10 samples i.e. $S = 10$ and three classes, the frequency of these classes are $A = 3, B = 3,$ and $C = 4.$

The Gini index for the distribution of applicants in the three classes is given as $G(S) = 1 - [(\frac{3}{10})^2 + (\frac{3}{10})^2 + (\frac{4}{10})^2] = 0.66.$

Now, consider using each of the attributes to split the sample:

1. Attribute: Own's home
    a. Consider value Yes = 5. Find out the frequency for the value Yes which belongs to respective classes A = 1, B = 2, and C = 2. Compute the Gini index for value Yes. $G(Yes) = 1 - [(\frac{1}{5})^2 + (\frac{2}{5})^2 + (\frac{2}{5})^2] = 0.64$
    b. Consider the value No = 5. It comes under A 2 times, B = 1, and C = 2. $G(No) = 1 - [(\frac{2}{5})^2 + (\frac{1}{5})^2 + (\frac{2}{5})^2] = 0.64$
    c. The total value of Gini index for attribute Own's home is $G(Ownshome) = \frac{n_1}{s}(G(Yes)) + \frac{n_2}{s}(G(No)) = \frac{5}{10}(0.64) + \frac{5}{10}(0.64) = 0.64$
2. Attribute: Married
    a. Yes = 5, A = 0, B = 1, C = 4. $G(Yes) = 1 - [(\frac{0}{5})^2 + (\frac{1}{5})^2 + (\frac{4}{5})^2] = 0.32$
    b. No = 5, A = 3, B = 2, C = 0. $G(No) = 1 - [(\frac{3}{5})^2 + (\frac{2}{5})^2 + (\frac{0}{5})^2] = 0.48$
    c. $G(Married) = \frac{5}{10}(0.32) + \frac{5}{10}(0.48) = 0.4$
3. Attribute: Gender
    a. Male = 3, A = 0, B = 3, C = 0. $G(Male) = 1 - [(\frac{0}{3})^2 + (\frac{3}{3})^2 + (\frac{0}{3})^2] = 0$

b. Female = 7, A = 3, B = 0, C = 4. $G(Female) = 1 - [(\frac{3}{7})^2 + (\frac{0}{7})^2 + (\frac{4}{7})^2] = 0.49$

c. $G(Gender) = \frac{3}{10}(0) + \frac{7}{10}(0.49) = 0.34$

4. Attribute: Employee

   a. Yes = 8, A = 3, B = 1, C = 4. $G(Yes) = 1 - [(\frac{3}{8})^2 + (\frac{1}{8})^2 + (\frac{4}{8})^2] = 0.59$

   b. No = 2, A = 0, B = 2, C = 0. $G(No) = 1 - [(\frac{0}{2})^2 + (\frac{2}{2})^2 + (\frac{0}{2})^2] = 0$

   c. $G(Employee) = \frac{8}{10}(0.59) + \frac{2}{10}(0) = 0.47$

5. Attribute: Credit Rating

   a. A = 5, A = 2, B = 1, C = 2. $G(A) = 1 - [(\frac{2}{5})^2 + (\frac{1}{5})^2 + (\frac{2}{5})^2] = 0.64$

   b. B = 5, A = 1, B = 2, C = 2. $G(B) = 1 - [(\frac{1}{5})^2 + (\frac{2}{5})^2 + (\frac{2}{5})^2] = 0.64$

   c. $G(CreditRating) = \frac{5}{10}(0.64) + \frac{5}{10}(0.64) = 0.64$

Prepare the table consisting of Gini index values of all attributes:

Attributes	Gini Index (before split)	Gini Index (after split)
Own home	0.66	0.64
Married	0.66	0.4
Gender	0.66	0.34
Employee	0.66	0.47
Credit rating	0.66	0.64

The attribute with the largest reduction (minimum) in the Gini index is selected as the split attribute. The split attribute is Gender. Now, we can reduce the data by removing the attribute gender and removing class B since all class B have gender male.

We redraw the table again by removing gender column and class B.

Sr. No.	Own's Home	Married	Employee	Credit Rating	Risk Class
1	No	No	Yes	A	A
2	Yes	Yes	Yes	B	C
3	No	Yes	Yes	B	C
4	No	No	Yes	B	A
5	Yes	No	Yes	A	A
6	No	Yes	Yes	A	C
7	Yes	Yes	Yes	A	C

There are 7 sample values in the reduced table. The Gini index value is equal to 0.489.

Now, consider using each of the attributes to split the sample:

1. Attribute: Own's home

   a. Yes = 3, A = 1, C = 2. $G(Yes) = 1 - [(\frac{1}{3})^2 + (\frac{2}{3})^2] = 0.44$

   b. No = 4, A = 2, C = 2. $G(No) = 1 - [(\frac{2}{4})^2 + (\frac{2}{4})^2] = 0.5$

   c. $G(OwnsHome) = \frac{3}{7}(0.44) + \frac{4}{7}(0.5) = 0.189 + 0.286 = 0.47$

2. Attribute: Married

   a. Yes = 4, A = 0, C = 4. $G(Yes) = 1 - [(\frac{0}{4})^2 + (\frac{4}{4})^2] = 0$

   b. No = 3, A = 3, C = 0. $G(No) = 1 - [(\frac{3}{3})^2 + (\frac{0}{3})^2] = 0$

   c. $G(Married) = \frac{4}{7}(0) + \frac{3}{7}(0) = 0$

3. Attribute: Employed

   a. Yes = 7, A = 3, C = 4. $G(Yes) = 1 - [(\frac{3}{7})^2 + (\frac{4}{7})^2] = 0.489$

   b. $G(No) = 0$

c. $G(Employee) = \frac{7}{7}(0.489) + 0 = 0.489$

4. Attribute: Credit Rating

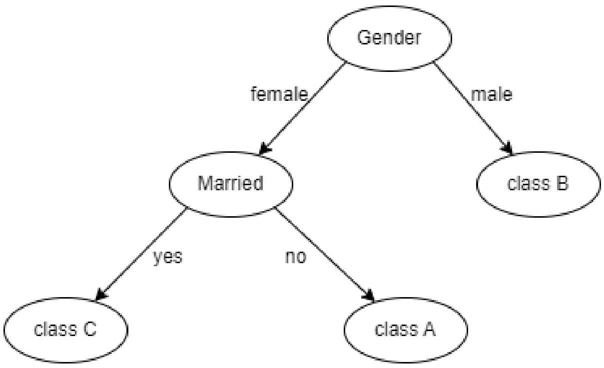
a. A = 4, A = 2, C = 2. $G(A) = 1 - [(\frac{2}{4})^2 + (\frac{2}{4})^2] = 0.5$

b. B = 3, A = 1, C = 2. $G(B) = 1 - [(\frac{1}{3})^2 + (\frac{2}{3})^2] = 0.44$

c. $G(CreditRating) = \frac{4}{7}(0.5) + \frac{3}{7}(0.44) = 0.285 + 0.188 = 0.473$

Attributes	Gini Index (before split)	Gini Index (after split)
Own home	0.489	0.476
Married	0.489	0
Employee	0.489	0.489
Credit rating	0.489	0.473

Tip: If an attribute produces Gini index 0 than stop the process of finding other Gini index's.



Bayesian Theorem

$$P(C_i/X) = \frac{P(X/C_i).P(C_i)}{P(X)}$$

▼ Q. Consider this data to predict a class label using Bayesian Classification:

RID	age	income	student	credit_rating	Class: buys_computer
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

Given: Class label attribute for $buys\_computer(Yes, No)$

Classify tuple: $X = (age = youth, income = medium, student = yes, credit_rating = fair)$

<u>Solution:</u>

Let, $C_1 = class : buys\_computer = yes$ and $C_2 = class : buys\_computer = no$

We need to maximize $P(X/C_i).P(C_i)$ for $i = 1, 2$.

The prior probability of each class can be computed based on the training tuples as mentioned below.

$P(buys\_computer = yes) = \frac{9}{14} = 0.643$

$P(buys\_computer = no) = \frac{5}{14} = 0.357$

To compute $P(X/C_i)$ for $i = 1, 2$. We compute the conditional probabilities:

1. Check X : age = youth
   a. $P(age = youth/buys\_computer = yes) = \frac{2}{9} = 0.222$
   b. $P(age = youth/buys\_computer = no) = \frac{3}{5} = 0.6$
2. Check X : income = medium
   a. $P(income = medium/buys\_computer = yes) = \frac{4}{9} = 0.444$
   b. $P(income = medium/buys\_computer = no) = \frac{2}{5} = 0.4$
3. Check X : student = yes
   a. $P(student = yes/buys\_computer = yes) = \frac{6}{9} = 0.667$
   b. $P(student = yes/buys\_computer = no) = \frac{1}{5} = 0.2$
4. Check X : credit_rating = fair
   a. $P(credit\_rating = fair/buys\_computer = yes) = \frac{6}{9} = 0.667$
   b. $P(credit\_rating = fair/buys\_computer = no) = \frac{2}{5} = 0.4$

Using the above probabilities we obtain:

$P(X/buys\_computer = yes) = P(age = youth/buys\_computer = yes) * P(income = medium/buys\_computer = yes) * P(student = yes/buys\_computer = yes) * P(credit\_rating = fair/buys\_computer = yes) = 0.222 * 0.444 * 0.667 * 0.667 = 0.044$

$P(X/buys\_computer = no) = P(age = youth/buys\_computer = no) * P(income = medium/buys\_computer = no) * P(student = yes/buys\_computer = no) * P(credit\_rating = fair/buys\_computer = no) = 0.6 * 0.4 * 0.2 * 0.4 = 0.019$

To find the class C_i that maximizes $P(X/C_i).P(C_i)$ we compute:

$C_1 = yes \rightarrow P(X/buys\_computer = yes) * P(buys\_computer = yes) = 0.044 * 0.643 = 0.028$

$C_2 = no \rightarrow P(X/buys\_computer = no) * P(buys\_computer = no) = 0.019 * 0.357 = 0.007$

Hence, tuple X belongs to yes class because $P(X/buys\_computer = yes)$ is highest.

Information Gain

$$Entropy(S) = -\frac{+ve}{N} log_2 \frac{+ve}{N} - \frac{-ve}{N} log_2 \frac{-ve}{N}$$

$$Gain(S, attribute) = S - \sum \frac{|S_v|}{|S|}(S_v)$$

▼ Q. Consider the following table to implement decision tree using Information Gain:

Day	Outlook	Temp	Humidity	Wind	Play tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No

Day	Outlook	Temp	Humidity	Wind	Play tennis
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Solution:

$S = -\frac{9}{14} log_2 \frac{9}{14} - \frac{5}{14} log_2 \frac{5}{14} = 0.94$

Attribute: Outlook

$$S_{Sunny} = -\frac{2}{5} log_2 \frac{2}{5} - \frac{3}{5} log_2 \frac{3}{5} = 0.971$$
$$S_{Overcast} = -\frac{4}{4} log_2 \frac{4}{4} - \frac{0}{4} log_2 \frac{0}{4} = 0$$
$$S_{Rain} = -\frac{3}{5} log_2 \frac{3}{5} - \frac{2}{5} log_2 \frac{2}{5} = 0.971$$
$$Gain(S, Outlook) = 0.94 - \frac{5}{14}(0.971) - \frac{4}{14}(0) - \frac{5}{14}(0.971) = 0.2464$$

Attribute: Temp

$$S_{Hot} = -\frac{2}{4} log_2 \frac{2}{4} - \frac{2}{4} log_2 \frac{2}{4} = 1$$
$$S_{Mild} = -\frac{4}{6} log_2 \frac{4}{6} - \frac{2}{6} log_2 \frac{2}{6} = 0.9183$$
$$S_{Cool} = -\frac{3}{4} log_2 \frac{3}{4} - \frac{1}{4} log_2 \frac{1}{4} = 0.8113$$
$$Gain(S, Temp) = 0.94 - \frac{4}{14}(1) - \frac{6}{14}(0.9183) - \frac{4}{14}(0.8113) = 0.0289$$

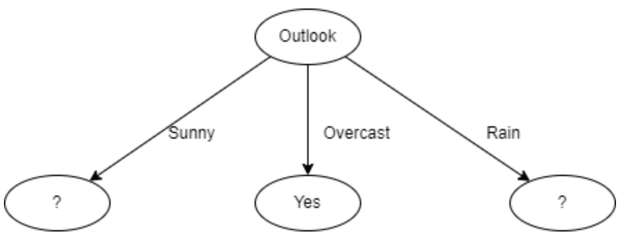
Attribute: Humidity

$$S_{High} = -\frac{3}{7} log_2 \frac{3}{7} - \frac{4}{7} log_2 \frac{4}{7} = 0.9852$$
$$S_{Normal} = -\frac{6}{7} log_2 \frac{6}{7} - \frac{1}{7} log_2 \frac{1}{7} = 0.5916$$
$$Gain(S, Humidity) = 0.94 - \frac{7}{14}(0.9852) - \frac{7}{14}(0.5916) = 0.1516$$

Attribute: Wind

$$S_{Strong} = 1$$
$$S_{Weak} = -\frac{6}{8} log_2 \frac{6}{8} - \frac{2}{8} log_2 \frac{2}{8} = 0.8113$$
$$Gain(S, Wind) = 0.94 - \frac{6}{14}(1) - \frac{8}{14}(0.8113) = 0.0478$$

Outlook has the maximum gain hence we consider it as the root node.



Now we solve for Sunny, we get the table as:

Day	Temp	Humidity	Wind	Play tennis
1	Hot	High	Weak	No
2	Hot	High	Strong	No
3	Mild	High	Weak	No
4	Cool	Normal	Weak	Yes

Day	Temp	Humidity	Wind	Play tennis
5	Mild	Normal	Strong	Yes

$S_{Sunny} = -\frac{2}{5}log_2\frac{2}{5} - \frac{3}{5}log_2\frac{3}{5} = 0.971$

Attribute: Temp

$S_{Hot} = 0$

$S_{Mild} = 1$

$S_{Cool} = 0$

$Gain(S_{Sunny}, Temp) = 0.97 - \frac{2}{5}(0) - \frac{2}{5}(1) - \frac{1}{5}(0) = 0.570$

Attribute: Humidity

$S_{High} = 0$

$S_{Normal} = 0$

$Gain(S_{Sunny}, Humidity) = 0.97 - \frac{3}{5}(0) - \frac{2}{5}(0) = 0.97$

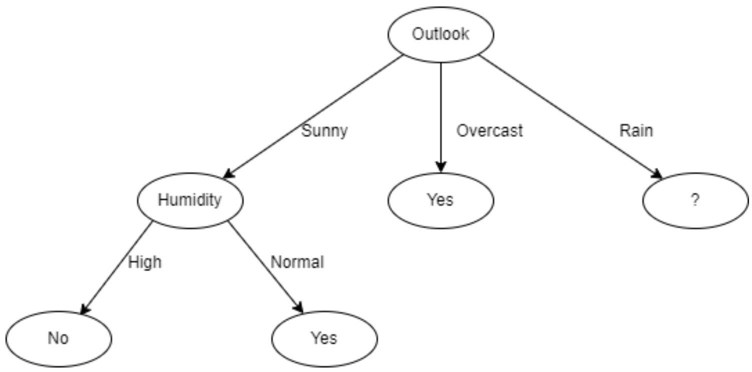
Attribute: Wind

$S_{Strong} = 1$

$S_{Weak} = -\frac{1}{3}log_2\frac{1}{3} - \frac{2}{3}log_2\frac{2}{3} = 0.9183$

$Gain(S_{Sunny}, Wind) = 0.97 - \frac{3}{5}(0.9183) - \frac{2}{5}(1) = 0.0192$

As Humidity have the highest information gain we will consider it as next node.



Now solving for Rain, we get the table as:

Day	Temp	Humidity	Wind	Play tennis
1	Mild	High	Weak	Yes
2	Cool	Normal	Weak	Yes
3	Cool	Normal	Strong	No
4	Mild	Normal	Weak	Yes
5	Mild	High	Strong	No

$S_{Rain} = -\frac{3}{5}log_2\frac{3}{5} - \frac{2}{5}log_2\frac{2}{5} = 0.971$

Attribute: Temp

$S_{Mild} = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.9183$

$S_{Cool} = 1$

$Gain(S_{Sunny}, Temp) = 0.97 - \frac{3}{5}(0.9183) - \frac{2}{5}(1) = 0.0192$

Attribute: Humidity

$S_{High} = 1$

$S_{Normal} = -\frac{2}{3}log_2\frac{2}{3} - \frac{1}{3}log_2\frac{1}{3} = 0.9183$

$$Gain(S_{Sunny}, Humidity) = 0.97 - \frac{2}{5}(1) - \frac{3}{5}(0.9183) = 0.9623$$

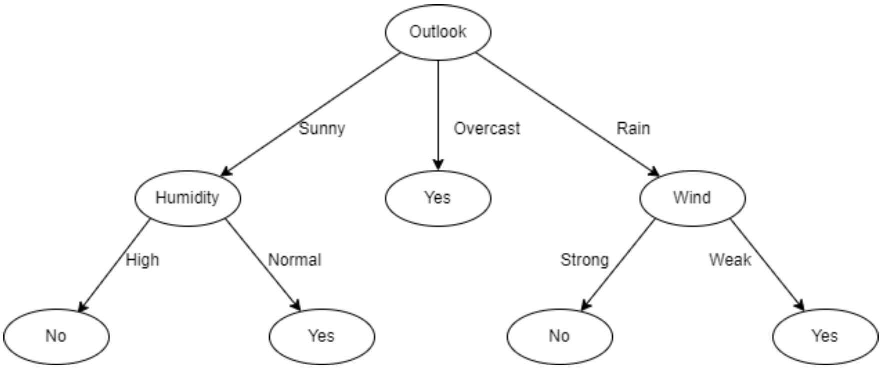
Attribute: Wind

$$S_{Strong} = 0$$

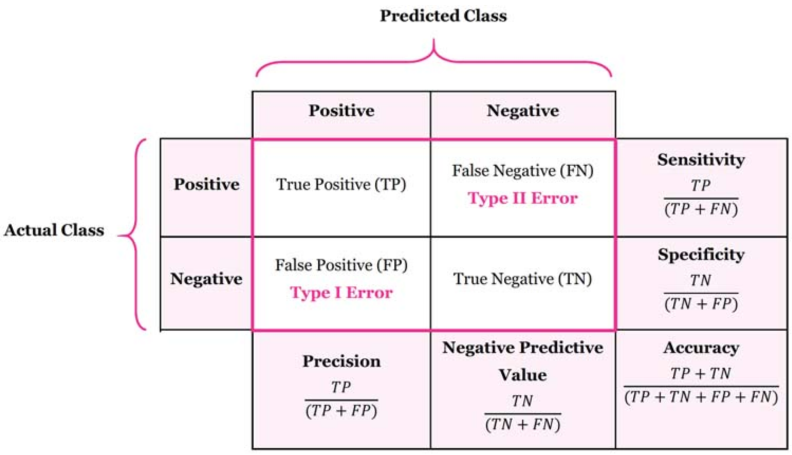
$$S_{Weak} = 0$$

$$Gain(S_{Sunny}, Wind) = 0.97 - \frac{3}{5}(0) - \frac{2}{5}(0) = 0.97$$

As Wind have the highest information gain we will consider it as next node:



Confusion Matrix



KNN(K-Nearest Neighbor) Algorithm

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

▼ Q. Find the class of the sepal length = 5.2 and sepal width = 3.1

Sr. No.	Sepal Length	Sepal Width	Species
1	5.3	3.7	Setosa
2	5.1	3.8	Setosa
3	7.2	3.0	Virginica
4	5.4	3.4	Setosa
5	5.1	3.3	Setosa
6	5.4	3.9	Setosa
7	7.4	2.8	Virginica
8	6.1	2.8	Versicolor
9	7.3	2.9	Virginica
10	6.0	2.7	Versicolor

Sr. No.	Sepal Length	Sepal Width	Species
11	5.8	2.8	Virginica
12	6.3	2.3	Versicolor
13	5.1	2.5	Versicolor
14	6.3	2.5	Versicolor
15	5.5	2.4	Versicolor

<u>Solution:</u>

Step 1: We use Euclidean distance formula to calculate distance from the given values

1. For Sr. No. 1 = $\sqrt{(5.2-5.3)^2+(3.1-3.7)^2}$ = $\sqrt{0.37}$ = 0.608
2. For Sr. No. 2 = $\sqrt{(5.2-5.1)^2+(3.1-3.8)^2}$ = 0.707
3. For Sr. No. 3 = 2.002
4. For Sr. No. 4 = 0.361
5. For Sr. No. 5 = 0.224
6. For Sr. No. 6 = 0.825
7. For Sr. No. 7 = 2.22
8. For Sr. No. 8 = 0.949
9. For Sr. No. 9 = 2.11
10. For Sr. No. 10 = 0.894
11. For Sr. No. 11 = 0.671
12. For Sr. No. 12 = 1.36
13. For Sr. No. 13 = 0.608
14. For Sr. No. 14 = 1.253
15. For Sr. No. 15 = 0.762

Step 2: Append the distance column to the table

Sr. No.	Sepal Length	Sepal Width	Species	Distance
1	5.3	3.7	Setosa	0.608
2	5.1	3.8	Setosa	0.707
3	7.2	3.0	Virginica	2.002
4	5.4	3.4	Setosa	0.361
5	5.1	3.3	Setosa	0.224
6	5.4	3.9	Setosa	0.825
7	7.4	2.8	Virginica	2.22
8	6.1	2.8	Versicolor	0.949
9	7.3	2.9	Virginica	2.11
10	6.0	2.7	Versicolor	0.894
11	5.8	2.8	Virginica	0.671
12	6.3	2.3	Versicolor	1.36
13	5.1	2.5	Versicolor	0.608
14	6.3	2.5	Versicolor	1.253
15	5.5	2.4	Versicolor	0.762

Step 3: Consider k = 5, select 5 values which are nearest to each other

We get five species which are Setosa, Setosa, Setosa, Virginica, and Versicolor

Here, three nearest neighbor belongs to Setosa species

Hence, which choose class as Setosa

Agglomerative Clustering using Single Linkage

▼ Q. Solve the following using Agglomerative Clustering using Single Linkage:

	P1	P2	P3	P4	P5
P1	0				
P2	9	0			
P3	3	7	0		
P4	6	5	9	0	
P5	11	10	2	8	0

<u>Solution:</u>

Step 1: Check points for minimum distance

	P1	P2	[P3, P5]	P4
P1	0			
P2	9	0		
[P3, P5]	_	_	0	
P4	6	5	_	0

Step 2: Find minimum distance

min(distance(P1, P3), distance(P1, P5)) = min(3, 11) = 3

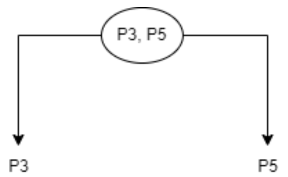
min(d(P2, P3), d(P2, P5)) = min(7, 10) = 7

min(d(P4, P3), d(P4, P5)) = min(9, 8) = 8

Step 3: Inserting the values

	P1	P2	[P3, P5]	P4
P1	0			
P2	9	0		
[P3, P5]	3	7	0	
P4	6	5	8	0

Step 4: Draw 1st cluster link between [P3, P5]



Step 5: Check points for the minimum distance from the last matrix

	[P1, P3, P5]	P2	P4
[P1, P3, P5]	0		
P2	_	0	
P4	_	5	0

Step 6: Find minimum distance

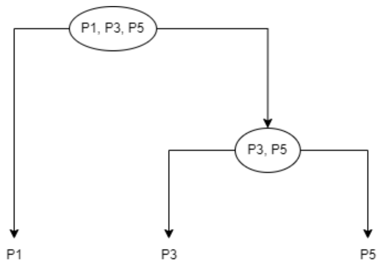
min(d(P2, P1), d(P2, P3), d(P2, P5)) = min(9, 7, 10) = 7

min(d(P4, P1), d(P4, P3), d(P4, P5)) = min(6, 9, 8) = 6

Step 7: Inserting the values

	[P1, P3, P5]	P2	P4
[P1, P3, P5]	0		
P2	7	0	
P4	6	5	0

Step 8: Draw 1st cluster link between [P1, P3, P5]



Step 9: Check points for the minimum distance from the last matrix

	[P1, P3, P5]	[P2, P4]
[P1, P3, P5]	0	
[P2, P4]	_	0

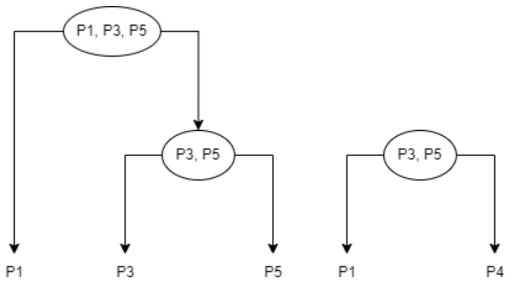
Step 10: Find minimum distance

min(d(P2, P1), d(P2, P3), d(P2, P5), d(P4, P1), d(P4, P3), d(P4, P5)) = min(9, 7, 10, 6, 9, 8) = 6

Step 11: Inserting the values

	[P1, P3, P5]	[P2, P4]
[P1, P3, P5]	0	
[P2, P4]	6	0

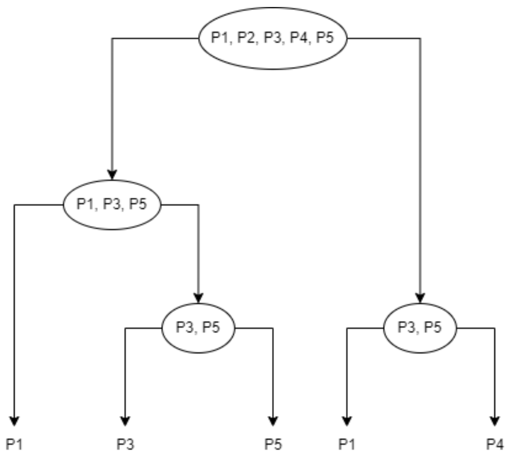
Step 12: Draw 1st cluster link between [P2, P4]



Step 13: Check points for the minimum distance from the last matrix

	[P1, P2, P3, P4, P5]
[P1, P2, P3, P4, P5]	0

Step 14: Draw final cluster link


P1, P2, P3, P4, P5
P1, P3, P5
P3, P5
P3, P5
P1
P3
P5
P1
P4