SUPERVISED LEARNING

Decision Tree (DT)

Decision Tree (DT)

- Classification algorithm – Supervised Learning
- What is classification ?

It is a process of dividing dataset into different group or classes or categories by adding labels.

- We do classification to perform predictive analysis- e-g when a machine receives an email- it has to classify whether it is spam or intended ?
- Applications : Fraud detection, Abnormality detection many more…

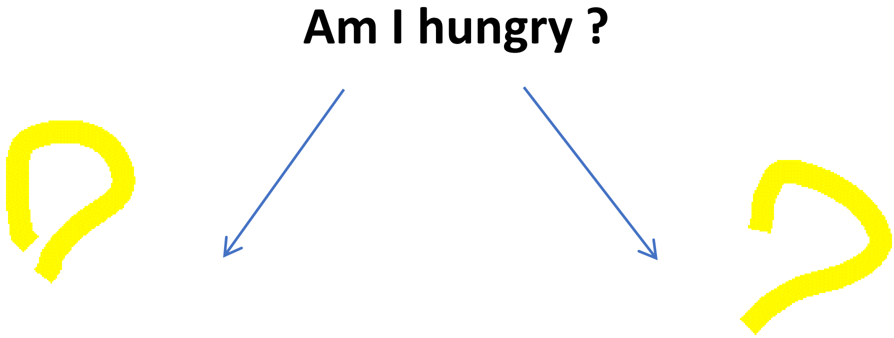
Types of classification:

- Decision Tree (DT)
- Random Forest
- Naïve Bayes
- KNN

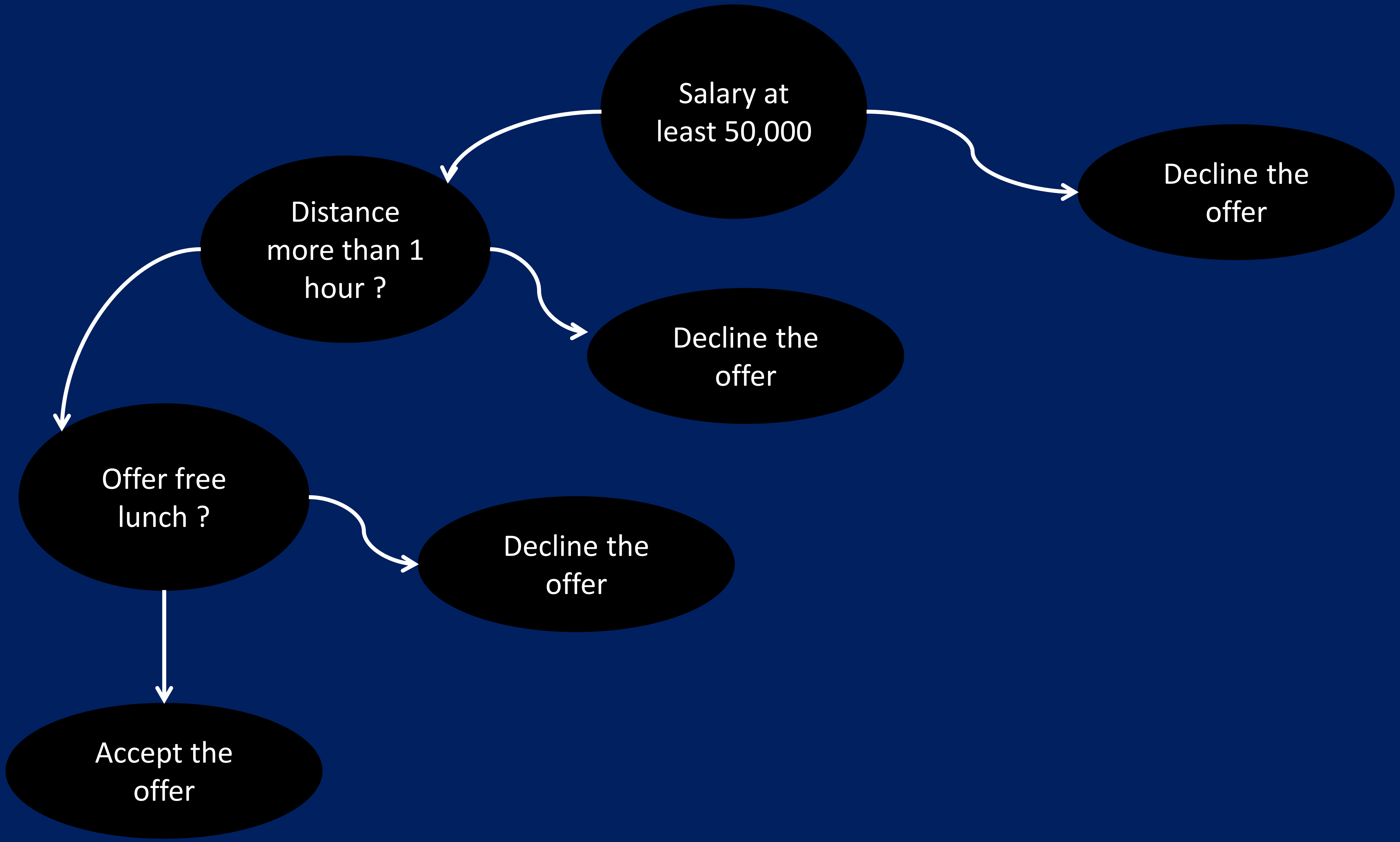
Decision Tree

It is a graphical representation of all possible solution to a decision.

- Decisions are made based on some conditions.
- Decision made can easily be explained.



Am I hungry ?



Salary at least 50,000

Decline the offer

Distance more than 1 hour ?

Decline the offer

Offer free lunch ?

Decline the offer

Accept the offer

Decision Tree Terminologies

- Root Node- It shows the entire population or sample and this further gets divided into two or more homogenous sets.

- Leaf Node- Final node.

- Branch- Formed by splitting the tree / nodes.

- Pruning- Opposite to splitting- Removing unwanted branches from the tree.

How does A tree decide where to split ?

- Gini Index- The measure of impurity (or purity) used in building decision tree in classification and regression tree (CART).
- Information Gain (IG)- The IG is the decrease in entropy after the dataset is split on the basis of an attribute. Constructing a DT is all about finding attribute that return the highest IG. [It decides which attribute should be selected as a decision node].
- Entropy- It is a metric to measure impurity.

$$Entropy\ (S) = -P(Yes)Log_2P(Yes) - P(No)Log_2P(No)\ \ldots\ldots 1$$

-S is the total sample space.

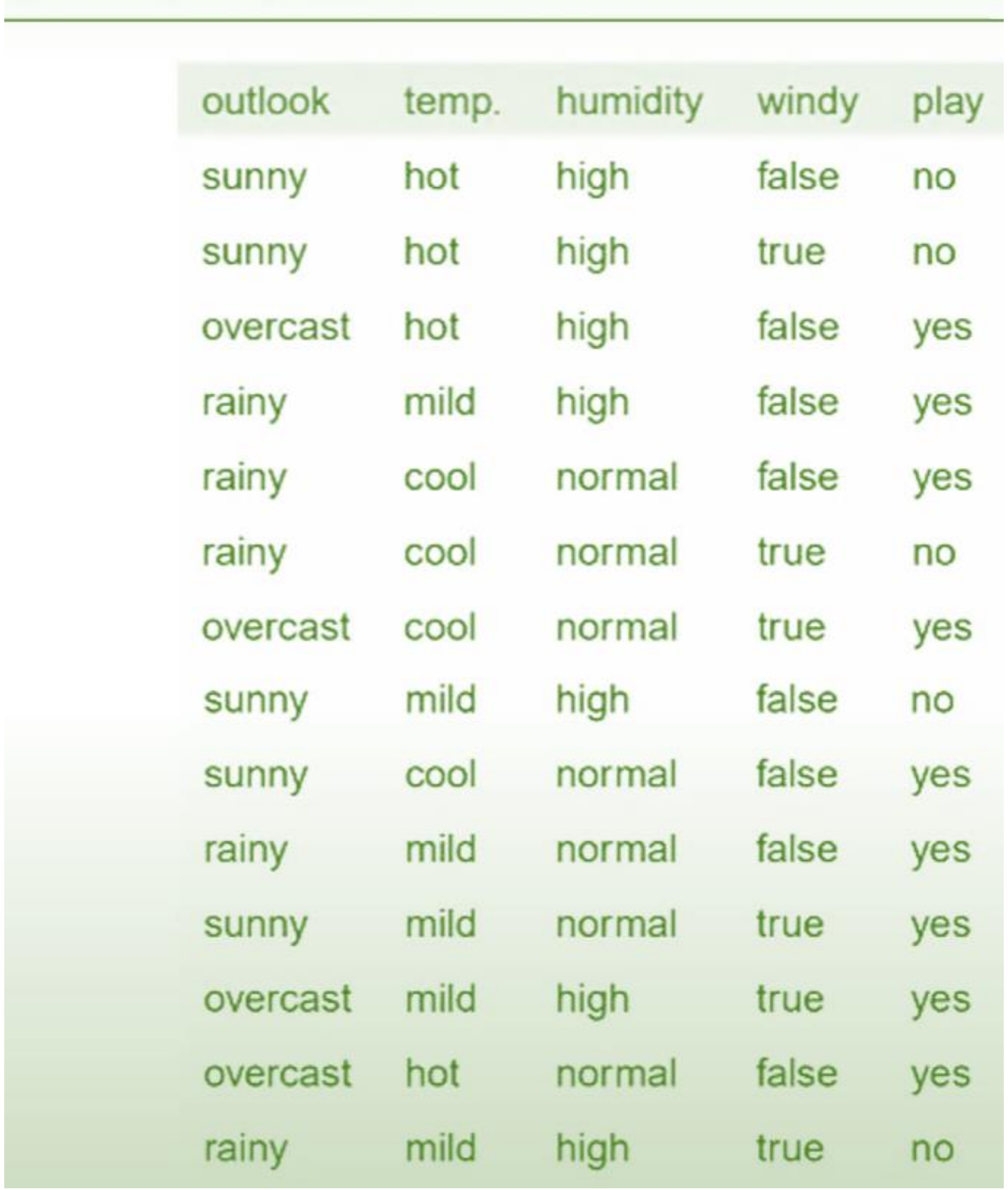
$$IG = Entropy(S) - [Weighted\ Avg]\ x\ Entropy\ (each\ feautre)\ \ldots\ldots\ 2$$

Step to build a DT

- STEP 1: Compute entropy of the entire Dataset.

- STEP 2: Which node to select as a Root Node ? [Based on the highest IG]

Problem:



outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

STEP NO 1: Compute the entropy for the Dataset.

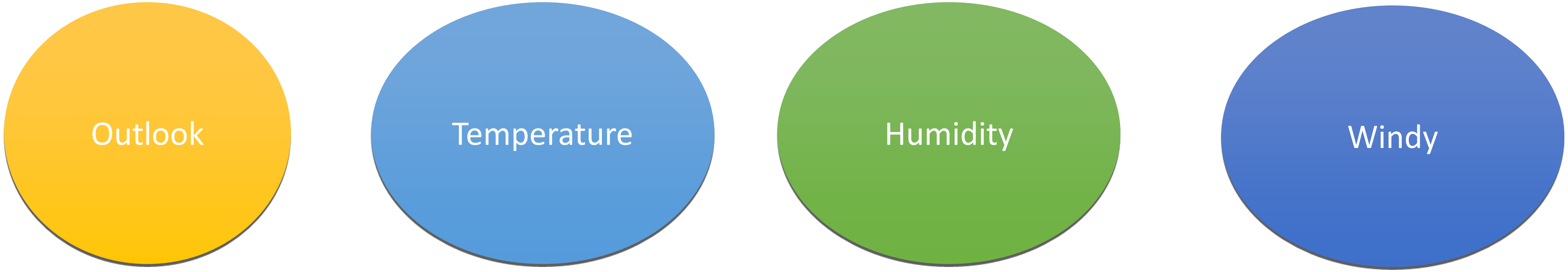
$$Entropy\ (S) = -P(Yes)Log_2P(Yes) - P(No)Log_2P(No)$$

Out of 14, we have 9 Yes and 5 No,

Hence,

E(s) = -P(9/14)Log2(9/14)- P(5/14)Log2(5/14) = 0.41 +0.53 = 0.94

STEP NO 2: Which node to select as a root node ?



Outlook

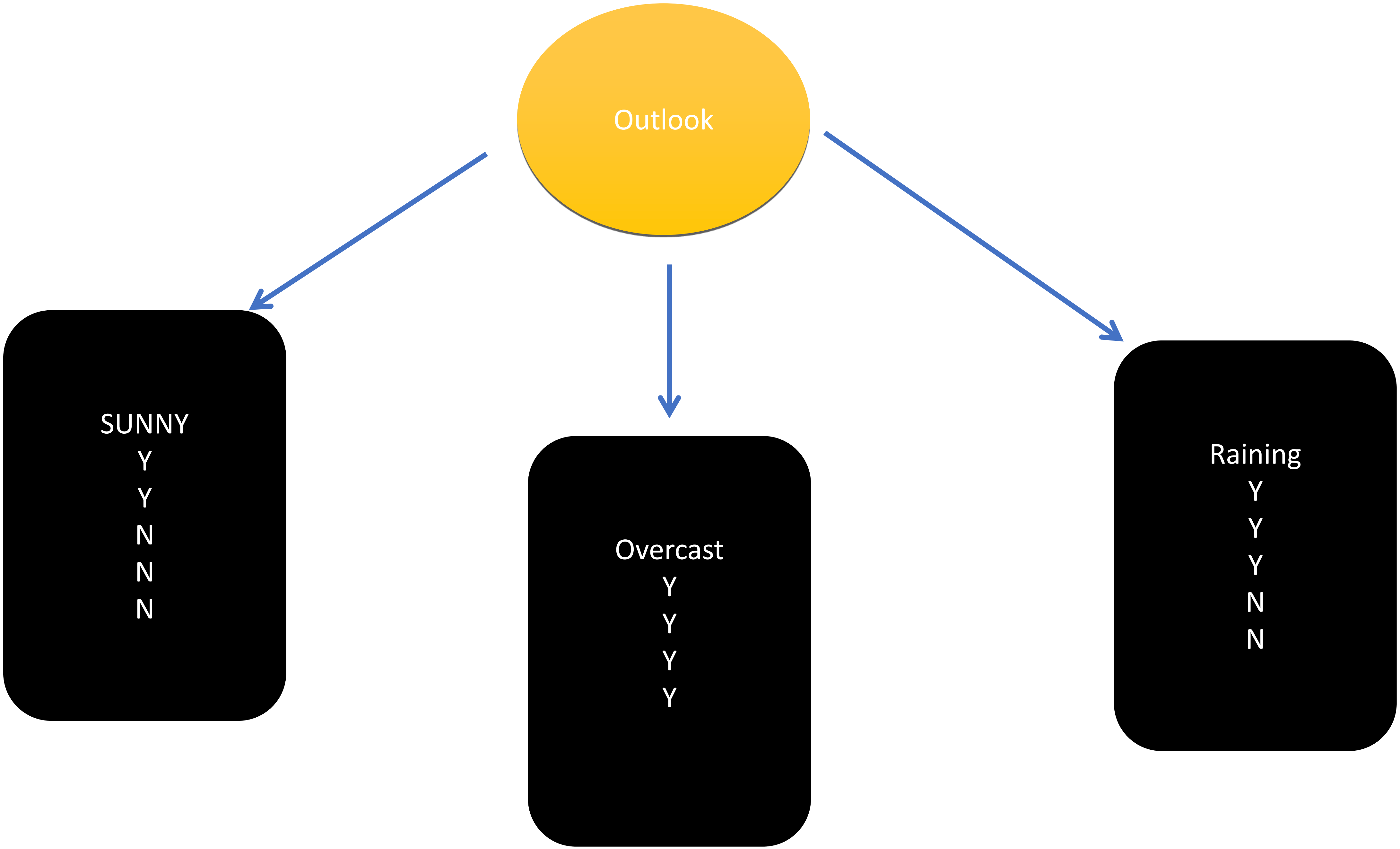
Temperature

Humidity

Windy

Select one by one:

Select Outlook first



Outlook

SUNNY
Y
Y
N
N
N

Overcast
Y
Y
Y
Y

Raining
Y
Y
Y
N
N

$E(\text{OutLook} = \text{Sunny}) = -2/5\log2(2/5)-3/5\log2(3/5) = 0.971$

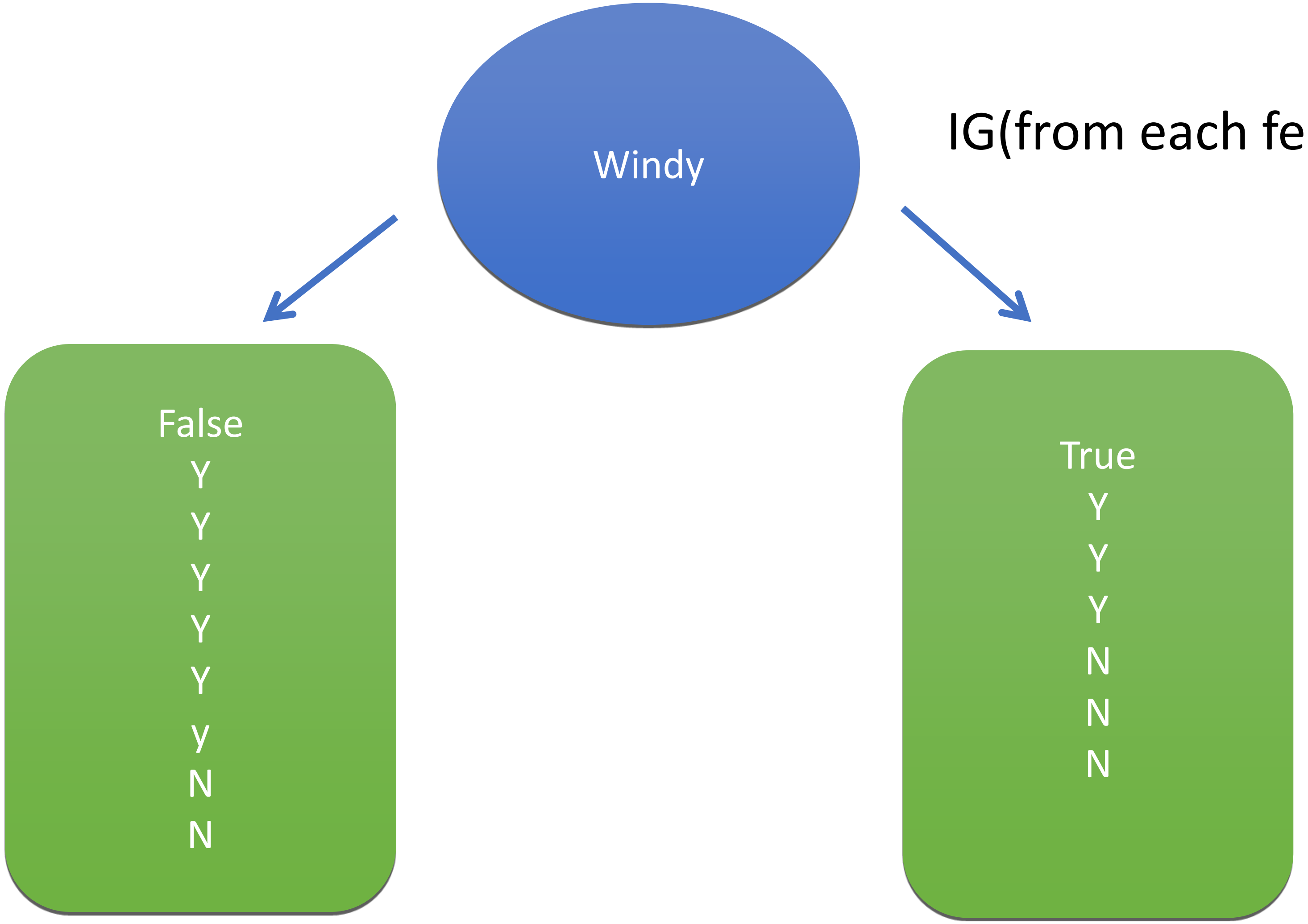
$E(\text{OutLook} = \text{Overcast}) = -4/4\log2(4/4)-0/4\log2(0/4) = 0$

$E(\text{OutLook} = \text{Raining}) = -3/5\log2(3/5)-2/5\log2(2/5) = 0.971$

$IG(\text{from each feature }) = 5/14 \times 0.971 + 4/14 \times 0 + 5/14 \times 0.971 = 0.693$

$IG(\text{OutLook }) = 0.94 - 0.693 = 0.247$

Now consider windy



Windy

False
Y
Y
Y
Y
Y
y
N
N

True
Y
Y
Y
N
N
N

E(windy = True) = 1

E(windy = False) = 0.811

IG(from each feature)= 8/14 x 0.811 + 6/14 x 1 = 0.892

IG(Windy) = 0.94 – 0.892 = 0.048

Outlook
IG = 0.247

Temp
IG = 0.029

Windy
IG = 0.048

Humidity
IG = 0.152

Hence, Outlook is selected

KNN

KNN-

- It is the simplest algorithm.

- Can be used for classification and prediction.

- Works on a distance metric. [Euclidian distance, city block distance etc..].

Algorithm:

1- Determine k

2- Estimate distance between new sample data point and training samples.

3- Sort the distance.

4- Collect the class of the 3 (having less distance) if k = 3.

5- Select the min one.

Classification – Problem

Use KNN algorithm to classify the D5 document. Use Euclidian distance and K =3.

Documents	X1	X2	Class
[redacted]	[redacted]	[redacted]	[redacted]
D2	7	4	C2
[redacted]	[redacted]	[redacted]	[redacted]
D4	1	4	C1
[redacted]	[redacted]	[redacted]	[redacted]

Classification – Problem

Use KNN algorithm to classify the D5 document. Use Euclidian distance.

Documents	X1	X2	Class
[redacted]	[redacted]	[redacted]	[redacted]
D2	7	4	C2
[redacted]	[redacted]	[redacted]	[redacted]
D4	1	4	C1
[redacted]	[redacted]	[redacted]	[redacted]

C_1

$$\sqrt{(3-7)^2+(7-7)^2} = 4$$

$C_2 = 5$

$C_1 = 3$

$C_1 = 3$

Collect 3 as k = 3
D1--C1
D4---C1
D3---C1

Select C1 as a class of D5 since the min distance is 3 of C1 of D3

Prediction – Problem

Use KNN algorithm to predict the weight of the Patient P8 document.
Use Euclidian distance and K =3.

Sr.no	Height	Age	Weight
P1	6	40	60
P2	6.11	26	55
P3	5.9	30	56
P4	5.8	32	58
P5	5.3	33	75
P6	5.6	34	78
P7	5.5	35	80
P8	5.8	37	??

Step no 1:

- Calculate the distance between P8 and rest of the Patients.

P1 = 3.006
P2 = 11.007
P3 = 7.007
P4 = 5
P5 = 4.03
P6 = 3.006
P7 = 2.022

Step no 02

Sort the least three values and take average.

P1, P6, P7

P7 + P6 + P1 / 3 = 72.66 would be the weight of the patient.