Unit- 2 (Data analysis)

* Regression Modelling -

(1) Regression models are widely used in analytics, in general being among the most easy to understand and interpret type of analytics techniques.
(2) Regression modelling is a statistical technique used for analyzing the relationship b/w a dependent variable and one an more independent variable.
(3) A regression models can be used to understand which of its marketting activities actually drive sales, and to what extent.
(4) Regression models are built to understand historical data and relationships to assess effectiveness, as in the marketting effectivness models.
(5) Regression techniques are used in financial services, retail, telecom, and medicine

Types -

(1) Linear Regression - It assumes that there is a linear relationship b/w the predictors and the target variable.

(2) Non linear Regression - Models where the relationship between variables in not linear and requires a more complex function forms.

(3) Logistic Regression - logistic regression is useful when our target variable is binomial (Accept or reject).

(4) Time series regression - It is used to forecast future behavior of variables based on historical time ordered data.

* Linear Regression Models -

linear regression is a statistical technique used to model the relationship b/w a dependent variable (target) and one or more independent variables. the fundamental assumption is that the relationship b/w the variables is linear. The goal of linear regression is to find the best fit line that minimizes the diff b/w the predicted and observed values.

Simple linear regression -

(1) Involves one independent variable.
(2) The relationship is represented by a straight line equation : $Y = b_0 + b_1 X + \epsilon$, where Y is the dependent variable, X is the independent variable, b_0 is the intercept, b_1 is the slope and ϵ is the error term.

Multiple linear regression -

(1) Involves two or more independent variable.
(2) The relationship is represented by the equation : $Y = b_0 + b_1 X_1 + b_2 X_2 + \dots + b_n X_n + \epsilon$, where Y is the dependent variable, $X_1, X_2 \dots X_n$ is the independent variable, b_0 is the intercept, $b_1, b_2, \dots b_n$ is the slope and ϵ is the error term.

Multivariate Analysis -

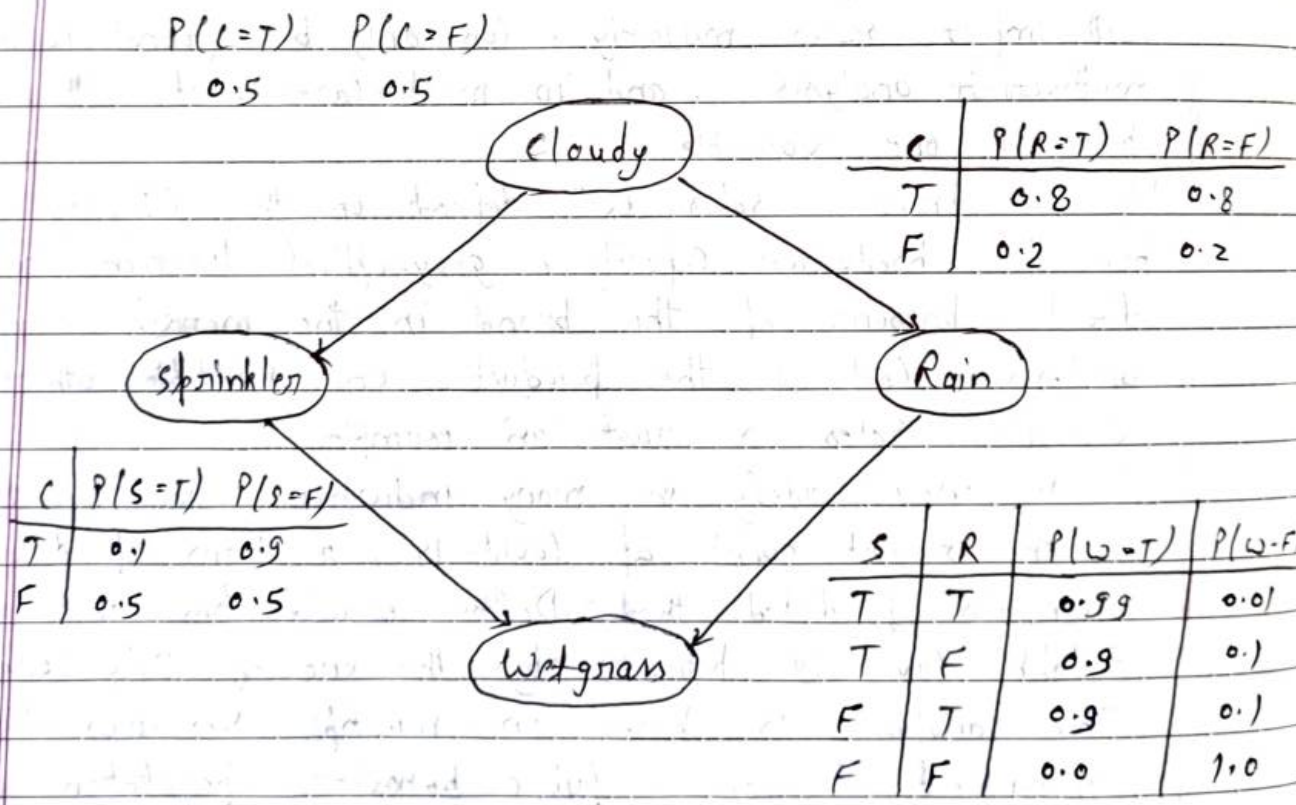
Multivariate means involving multiple dependent variables resulting in one outcome. This explains that the majority of the problems in real world are multivariate. For example, we cannot predict the weather of any year based on the season. There are multiple factors like pollution, humidity, precipitation, etc.

Suppose a project has been assigned to you to predict the sales of the Company. You cannot simply say that 'X' is the factor which will affect the sales.
We know that there are multiple aspects on variables which will impact sales. To analyze the variables that will impact sales majorly, can only be found with multivariate analysis, and in most cases, it will not be just one variable.
like we know, sales will depend on the category of product, production capacity, geographical location, marketing effort, presence of the brand in the market, competitor analysis, cost of the product, and multiple other variables. Sales is just an example.
It is used widely in many industries, like healthcare. In the recent event of covid-19, a team of data scientists predicted that Delhi would have more than 5 lakh covid-19 patients by the end of July 2020. This analysis is based on multiple variables like government decision, public behavior, population, occupation, public transport, healthcare services and overall immunity of the community.

* Bayesian network -

(1) By definition, Bayesian networks are a type of probabilistic graphical model that uses the bayesian inferences for probability computations. It represents a set of variables and its conditional probabilities with a directed acyclic graph. They are primarily suited for considering an event that has occurred and predicting the likelihood that any one of the several possible known cause is the contributing factor.

(2) Bayesian n/w aim to model conditional dependence by representing edges in a directed graph.



(3) In the figure, since rain has an edge going into wetgrass, it means that P(wetgrass / Rain) will be

a factor, whose probability values are specified next to the wetgrass node in a Conditional probability table.

(4) Bayesian n/w satisfy the markov property, which states that a node is conditionally independent of its non descendants given its parents. In the given example, this means that

P(Sprinkler / cloudy, Rain) = P(Sprinkler / cloudy)

Since sprinkler is Conditionally independent of its non descendant, Rain, given cloudy.

* Time series analysis -

Time series analysis is a method of analyzing data points collected over a period of time. The components of a time series are the patterns that can be observed in the data. These patterns can be used to understand the data's underlying behavior and to make predictions about the future.

Time series analysis is used for non stationary data things that are constantly fluctuating overtime or are affected by time. Industries like finance, retail and economics frequently use time series analysis because currency and sales are always changing. Stock market analysis is an excellent example of time series analysis in action. Examples of time series analysis in action include:-

(1) weather data
(2) Rainfall measurements
(3) Temperature readings
(4) Heart rate monitoring
(5) Brain monitoring
(6) Stock prices
(7) Interest rates

Components of time series-

(1) Trends-

(a) The trend refers to the long term movement in a time series.
(b) It indicates whether the observation values are increasing or decreasing over time.

(2) Seasonality-

(a) The seasonality component describes the fixed, periodic fluctuation in the observations over time.
(b) It is often related to the Calendar.

(3) Cyclic -

(a) A cyclic component also refers to a periodic fluctuation, which is not as fixed.

* Supervised learning - Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data machine predicts the o/p. The labelled data means some i/p data is already tagged with the correct o/p.
Training data provided to the machines work as the supervisor that teaches the machines to predict the o/p correctly. Same concept as a student learn in the supervision of the teacher.
It classified into two categories -

(a) Regression - Regression is a type of supervised learning that is used to predict continuous values, such as house prices, stock prices, or customer churn. A regression problem is when the o/p variable is a real value, such as "dollars" or "weight".

(b) Classification - Classification is a type of supervised learning that is used to predict categorical values, such as whether a customer will churn or not, whether an email is spam or not, or whether a medical image shows a tumor or not.

* Unsupervised learning - Unsupervised learning is a machine learning techniques in which models are not supervised using training dataset. Instead, models itself find the hidden patterns and insights from the given data. It can be compared to learning which takes place in the human brain while learning new things.

"Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision"

It can be classified into two categories -

(a) Clustering - Clustering is a type of unsupervised learning that is used to group similar data points together. Clustering algorithms work by iteratively moving data points closer to their cluster centers and further away from data points in other clusters.

(b) Association - Association rule learning is a type of unsupervised learning that is used to identify patterns in a data. This algorithm works by finding relationships b/w different items in a dataset.

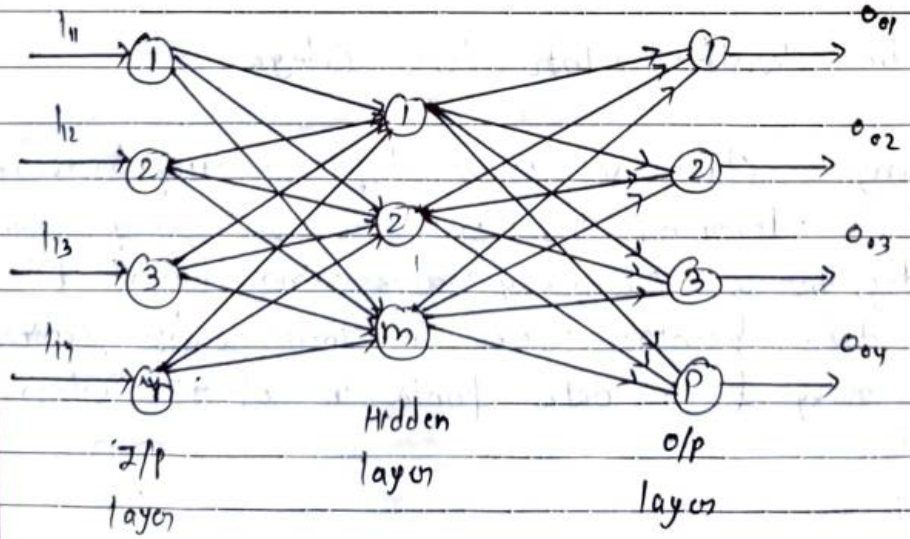
* Multilayer perceptron model -

(1) Multilayer perceptron is a class of feed forward artificial neural n/w.

(2) Multilayer perceptron model has three layers; an input layer, and output layer and layer in b/w not connected directly to the i/p or the o/p and hence called the hidden layer.

(3) For the perceptrons in the i/p layer, we use linear transfer function, and for the perceptrons in the hidden layer and the o/p layer, we use sigmoid or squashed-S function.

(4) The i/p layer serves to distribute the values they receive to the next layer and so, does not perform a weighted sum or threshold.



(5) Multilayer perceptron does not increase computational power over a single layer neural n/w unless there is a non linear activation function b/w layers.

Learning Algorithm -

(1) If the nth number of input set $x(n)$, is correctly classified into linearly separable classes, by the weight vector $w(n)$ then no adjustment of weights are done.

$$w(n+1) = w(n)$$

if $w^T x(n) > 0$ and $x(n)$ belongs to C_1.

$$w(n+1) = w(n)$$

if $w^T x(n) \leq 0$ and $x(n)$ belongs to C_2.

(2) otherwise, the weight vector of the perceptron is updated in accordance with the rule.

* Back Propagation -

Back propagation is a widely used algorithm for training feedforward neural networks. It computes the gradient of the loss function with respect to the n/w weights. It is very efficient, rather than naively directly computing the gradient concerning each weight. This efficiency makes it possible to use gradient methods to train multi layer n/w's and update weights to minimize loss; variants such as gradient descent or stochastic gradient descent

are often used.

* Learning rate -

(1) learning rate is a constant used in learning algorithm that define the speed and extend in weight matrix corrections.
(2) Setting a high learning rate tends to bring instability and the system is difficult to converge even to a near optimum solution.
(3) A low value will improve stability, but will slow down convergence.

* Competitive learning -

Competitive learning is a type of unsupervised learning algorithm where a group of neurons competes to become the most activated in response to a particular input pattern. The basic idea is that only one neuron "wins" the competition by becoming the most responsive to a given i/p, while the other neurons are inhibited or suppressed.

* Principle Component analysis (PCA) -

(1) PCA is a method used to reduce number of variables in dataset by extracting important one from a large data set.
(2) It reduces the dimension of our data with the aim of retaining as much information as possible.

(3) In other words, this method combines highly correlated variables together to form a smaller number of an artificial set of variables which is called principal components (PC) that account for most variance in the data.

(4) A principal component can be defined as a linear combination of optimally weighted observed variables.

(5) The first principal component retains maximum variations that was present in the original components.

(6) The PC are the eigenvectors of a covariance matrix and hence they are orthogonal.

(7) The o/p of PCA are these principal components, the number of which is less than or equal to the no. of original variables.

6.

X	2	3	4	5	6	7
Y	1	5	3	6	7	8

Two attributes data set find PC_1 and PC_2.

Sol- Covariance Matrix $= \begin{bmatrix} cov(x,x) & cov(x,y) \\ cov(y,x) & cov(y,y) \end{bmatrix} / n-1$

$$cov(x,x) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})^2}{n-1}$$

$(x-\bar{x})$ A	$(y-\bar{y})$ B	AB	A^2	B^2
-2.5	-4	10	6.25	16
-1.5	0	0	2.25	0
-0.5	-2	1	0.25	4
0.5	1	0.5	0.25	1
1.5	2	3	2.25	4
2.5	3	7.5	6.25	9

$\bar{x} = \frac{27}{6} = 4.5$

$\bar{y} = \frac{30}{6} = 5$

Cov matrix (C) = $\begin{bmatrix} 17.5 & 22 \\ 22 & 34 \end{bmatrix} / 5$

$$C = \begin{bmatrix} 3.5 & 4.4 \\ 4.4 & 6.8 \end{bmatrix}$$

$$C - \lambda I = 0$$

$$\begin{bmatrix} 3.5 - \lambda & 4.4 \\ 4.4 & 6.8 - \lambda \end{bmatrix} = 0$$

$$(3.5 - \lambda)(6.8 - \lambda) - (4.4)^2 = 0$$

$$\lambda^2 - 10.3\lambda + 4.44 = 0$$

$$\lambda_1 = 0.4507 \quad , \quad \lambda_2 = 9.849$$

$$\begin{bmatrix} 3.5 - 0.4507 & 4.4 \\ 4.4 & 6.8 - 0.4507 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$3.0493\, x_1 + 4.4 y_1 = 0$$
$$4.4 x_1 + 6.3493\, y_1 = 0$$

$$7.4493\, x_1 = -10.7493\, y_1$$

$$\boxed{x_1 = -1.442\, y_1}$$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} -1.442 \\ 1 \end{bmatrix} = \sqrt{(-1.442)^2 + (1)^2}$$
$$= \sqrt{3.079}$$
$$= 1.754$$

$$= \begin{bmatrix} -1.442/1.754 \\ 1/1.754 \end{bmatrix}$$

$$\text{Aigen Vector} = \begin{bmatrix} -0.822 \\ 0.570 \end{bmatrix}$$

$$\begin{bmatrix} 3.5 - 9.849 & 4.4 \\ 4.4 & 6.8 - 9.849 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$-6.349\, x_2 + 4.4\, y_2 = 0$$
$$4.4\, x_2 - 3.049\, y_2 = 0$$

$$6.349\, x_2 = 4.4\, y_2$$
$$4.4\, x_2 = 3.049\, y_2$$

$$10.749\, x_2 = 7.449\, y_2$$

$$\boxed{x_2 = 0.692\, y_2}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0.692 \\ 1 \end{bmatrix} = \sqrt{(0.692)^2 + (1)^2}$$
$$= 1.216$$

$$= \begin{bmatrix} 0.692/1.216 \\ 1/1.216 \end{bmatrix}$$

$$\text{aigen vector} = \begin{bmatrix} 0.569 \\ 0.822 \end{bmatrix}$$

Total Variance = $\frac{\lambda_1}{\lambda_1+\lambda_2}$, $\frac{\lambda_2}{\lambda_1+\lambda_2}$

= $\frac{0.4507}{10.2997}$, $\frac{9.849}{10.2997}$

= 0.043 , 0.956

= 4.3 % , 95.6 %

* Overfitting - Overfitting occurs when a machine learning model learns the training data too well, capturing noise or random fluctuations in the data rather than the underlying patterns. As a result, the model performs well on the training data but poorly on new, unseen data, as it has essentially memorized the training examples instead of generalizing from them.

* Underfitting - Underfitting happens when a machine learning model is too simple to capture the underlying patterns in the training data. The model performs poorly on both the training data and new, unseen data because it fails to capture the complexity of the relationships in the data.

* Fuzzy logic -

Fuzzy logic is a reasoning process that extends classical binary logic to handle fuzzy or uncertain

information. Fuzzy logic allows for the representation of degrees of truth, which means that statement can be partially false rather than just true or false.
Key Components of fuzzy logic -

(1) Fuzzy set - Fuzzy sets are used to represent linguistic variable and their membership functions. Membership function assigns degree of membership to elements in the universe of discourse.

(2) Fuzzy Rules - Fuzzy rules defines relationship b/w fuzzy set and are typically expressed in the form of If-then rules, Exm - If temperature is cold and humidity is high then increase heating.

(3) Fuzzy inference engine - The inference engine processes fuzzy rules and membership functions to make fuzzy logic based decisions. It calculates a fuzzy o/p that represents the degree of truth of the conclusion.

(4) Defuzzification - The final stp involves converting the fuzzy o/p back into a crisp value, This process is called defuzzification and results in a specific numerical value of decision.

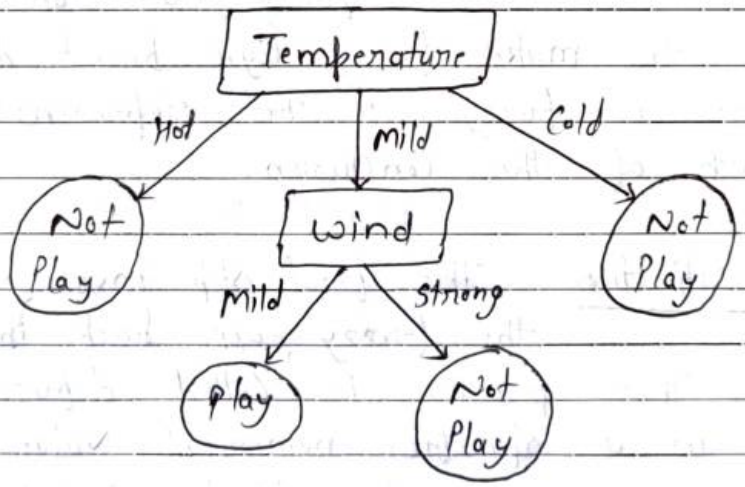
Role of crisp sets in fuzzy logic -

(1) It contains the precise location of the set boundaries.
(2) It provides the membership value of the set.

* Compare Crisp (classical) logic and Fuzzy logic -

	Crisp logic	Fuzzy logic
(1)	In classical logic an element either belongs to or does not belong to a set.	Fuzzy logic supports a flexible sense of membership of elements to a set.
(2)	Crisp logic is built on a 2 state truth values (True/False).	Fuzzy logic is built on a multistate truth values.
(3)	The statement which is either True or False but not both is called a proposition in Crisp logic.	A fuzzy proposition is a statement which acquires a fuzzy truth value.

* Fuzzy decision tree -



(1) A fuzzy decision tree is a natural extension of traditional decision tree.

(2) The major diff b/w fuzzy decision tree and traditional, at each node of fuzzy decision tree the considered attribute is compare to a group of linguistic term predicted by fuzzy set.

(3) Each attribute node will have membership according to the membership function of the linguistic term.
(4) Similarly considering all the branches for leaf node we can obtain its firing strength by the minimum of the firing strength of the all the branches from the root node to leaf node.

IDT Algorithm - Crate smasher

Weather	Temp	Humidity	Wind	Play Football
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Cloudy	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Cloudy	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Cloudy	Mild	High	Strong	Yes
Cloudy	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Calculate IG for weather -

Step-1 Entropy of entire dataset

$$S[+9,-5] = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.94$$

Step-2 Entropy of all attributes -

Entropy of sunny $\{+2, -3\} = -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log\left(\frac{3}{5}\right)$

$= 0.97$

Entropy of cloudy $\{+4, -0\} = -\frac{4}{4}\log_2\left(\frac{4}{4}\right) - \frac{0}{4}\log_2\left(\frac{0}{4}\right)$

$= 0$

Entropy of Rain $\{+3, -2\} = -\frac{3}{5}\log\left(\frac{3}{5}\right) - \frac{2}{5}\log\left(\frac{2}{5}\right)$

$= 0.97$

Information Gain = Entropy (whole data) $- \frac{5}{14}$ Ent(S) $- \frac{4}{14}$ Ent(C

$- \frac{5}{14}$ Ent(R)

$= 0.246$

Calculate IG of temperature -

Step-1 Entropy of Entire dataset

$S\{+9, -5\} = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.94$

Step-2 Entropy of all attributes -

Entropy of Hot $\{+2, -2\} = -\frac{2}{4}\log_2\left(\frac{2}{4}\right) - \frac{2}{4}\log_2\left(\frac{2}{4}\right) = 1$

Entropy of Mild $\{+4, -2\} = -\frac{4}{6}\log_2\left(\frac{4}{6}\right) - \frac{2}{6}\log_2\left(\frac{2}{6}\right) = 0.91$

Entropy of Cold $\{+3, -1\} = -\frac{3}{4}\log_2\left(\frac{3}{4}\right) - \frac{1}{4}\log_2\left(\frac{1}{4}\right) = 0.81$

Information Gain = Entropy (whole data) $- \frac{4}{14}$ Ent (H) $- \frac{6}{14}$ Ent (N)

$- \frac{4}{14}$ Ent (C)

$= 0.029$

Calculate IG of Humidity -

Step-1 -

$S\{+9, -5\} = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.94$

Step-2

Entropy of High $\{+3, -4\} = -\frac{3}{7}\log_2\left(\frac{3}{7}\right) - \frac{4}{7}\log\left(\frac{4}{7}\right) = 0.98$

Entropy of Normal $\{+6, -1\} = -\frac{6}{7}\log_2\left(\frac{6}{7}\right) - \frac{1}{7}\log_2\left(\frac{1}{7}\right) = 0.59$

Info Gain = Entropy (whole data) $- \frac{7}{14}$ Ent (H) $- \frac{7}{14}$ Ent (N)

$= 0.15$

Calculate IG of Wind -

Step-1

$S\{+9, -5\} = -\frac{9}{14}\log\left(\frac{9}{14}\right) - \frac{5}{14}\log\left(\frac{5}{14}\right) = 0.94$

Step - 2

Entropy of strong $\{+3, -3\} = -\frac{3}{6}\log\left(\frac{3}{6}\right) - \frac{3}{6}\log\left(\frac{3}{6}\right)$

$= 1$

Entropy of Normal $\{+6, -2\} = -\frac{6}{8}\log\left(\frac{6}{8}\right) - \frac{2}{8}\log\left(\frac{2}{8}\right)$

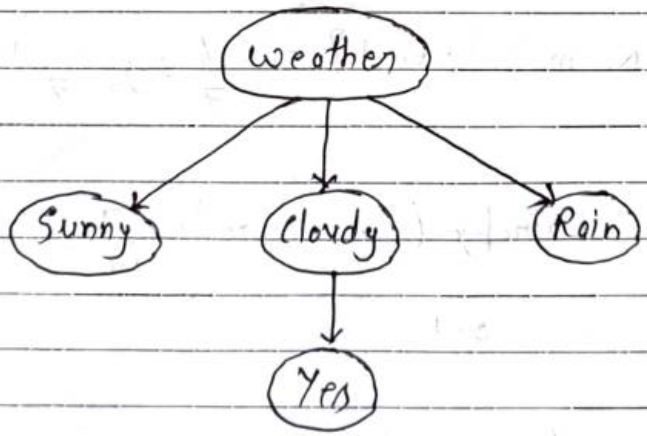
$= 0.81$

Information Gain = Entropy (whole data) $- \frac{6}{14}$ Ent (S)

$- \frac{8}{14}$ Ent (N)

$= 0.0478$

Gain (S, weather) = 0.246 ✓ (Find max value)
Gain (S, Temp) = 0.029
Gain (S, Humidity) = 0.15
Gain (S, wind) = 0.0478



Calculate Gain of temp -

Step - 1 Entropy of Sunny

$S\{+2,-3\} = -\frac{2}{5}\log\left(\frac{2}{5}\right) - \frac{3}{5}\log\left(\frac{3}{5}\right) = 0.97$

Step-2 - Entropy of all attributes

Entropy of Hot $\{0,-2\} = -\frac{0}{2}\log\left(\frac{0}{2}\right) - \frac{2}{2}\log\left(\frac{2}{2}\right) = 0$

Entropy of Mild $\{+1,-1\} = -\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 1$

Entropy of Cold $\{+1,-0\} = -\frac{1}{1}\log\left(\frac{1}{1}\right) - \frac{0}{1}\log\left(\frac{0}{1}\right) = 0$

Information Gain = Entropy (sunny) $- \frac{2}{5}$ Ent (H) $- \frac{2}{5}$ Ent (M) $- \frac{1}{5}$ Ent (C)

$= 0.57$

Calculate IG of Humidity -

Step-1 Entropy of Sunny $\{+2,-3\} = 0.97$

Step-2 -

Entropy of High $\{+0,-3\} = -\frac{0}{3}\log\left(\frac{0}{3}\right) - \frac{3}{3}\log\left(\frac{3}{3}\right) = 0$

Entropy of Normal $\{+2,-0\} = -\frac{2}{2}\log\left(\frac{2}{2}\right) - \frac{0}{2}\log\left(\frac{0}{2}\right) = 0$

Information Gain = Entropy (sunny) $- \frac{3}{5}$ Ent (H) $- \frac{2}{5}$ Ent (N)

$= 0.97$

Calculate IG of Wind -

Step-1 Entropy of Sunny $\{+2, -3\} = 0.97$

Step-2 Entropy of all attributes

Entropy of strong $\{+1, -1\} = -\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) =$

Entropy of weak $\{+1, -2\} = -\frac{1}{3}\log\left(\frac{1}{3}\right) - \frac{2}{3}\log\left(\frac{2}{3}\right)$

$= 0.918$

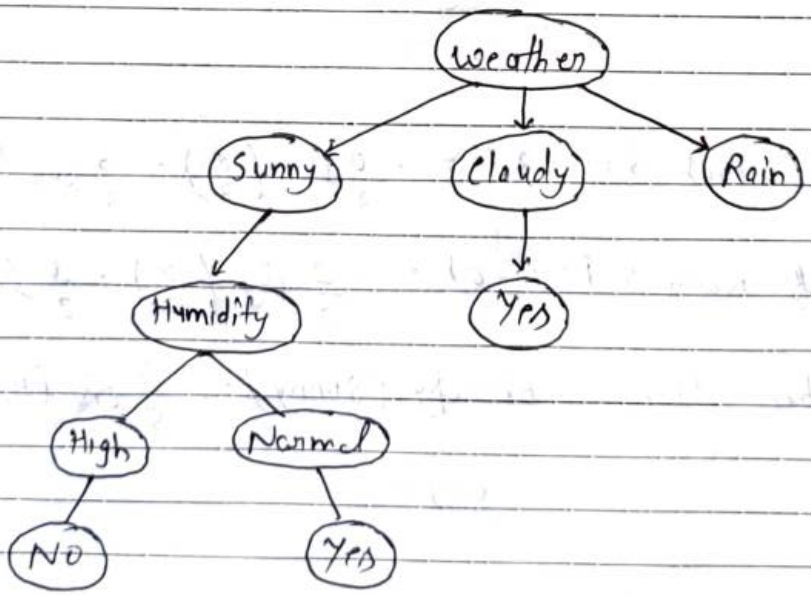
Info. Gain $=$ Entropy(sunny) $- \frac{2}{5}$ Ent(s) $- \frac{3}{5}$ Ent(w)

$= 0.019$

Gain $(S_{sunny}, temp) = 0.57$

Gain $(S_{sunny}, Humidity) = 0.97$ ✓

Gain $(S_{sunny}, wind) = 0.019$



TARANG P. No. Date Plus

Calculate IG of temp -

Step-1 Entropy of Rain {+3, -2} = $-\frac{3}{5}\log\left(\frac{3}{5}\right) - \frac{2}{5}\log\left(\frac{2}{5}\right)$

= 0.97

Step-2 Entropy of all attributes

Entropy (Hot) {+0, -0} = $-\frac{0}{2}\log_2\left(\frac{0}{2}\right) - \frac{0}{2}\log_2\left(\frac{0}{2}\right)$

= 0

Entropy (mild) {+2, -1} = $-\frac{2}{3}\log\left(\frac{2}{3}\right) - \frac{1}{3}\log\left(\frac{1}{3}\right)$

= 0.918

Entropy (cool) {+1, -1} = $-\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 1$

Info. Gain = Ent(Rain) - $\frac{0}{5}$ Ent(H) - $\frac{3}{5}$ Ent(M) - $\frac{2}{5}$ Ent(C)

= 0.019

Calculate IG of Wind -

Step-1 Ent(Rain) = 0.97

Step-2

Entropy (Strong) {+0, -2} = $-\frac{0}{2}\log\left(\frac{0}{2}\right) - \frac{2}{2}\log\left(\frac{2}{2}\right) = 0$

Entropy (weak) {+3, -0} = $-\frac{3}{3}\log\left(\frac{3}{3}\right) - \frac{0}{3}\log\left(\frac{0}{3}\right) = 0$

I.G. = Entropy (Rain) - $\frac{2}{5}$ Ent(S) - $\frac{3}{5}$ Ent(W) = 0.97

Calculate Info of Humidity -

Step-1 → Ent(Rain) = 0.97

Step-2

$$Ent(High)\{+1,-1\} = -\frac{1}{2}\log\left(\frac{1}{2}\right) - \frac{1}{2}\log\left(\frac{1}{2}\right) = 1$$

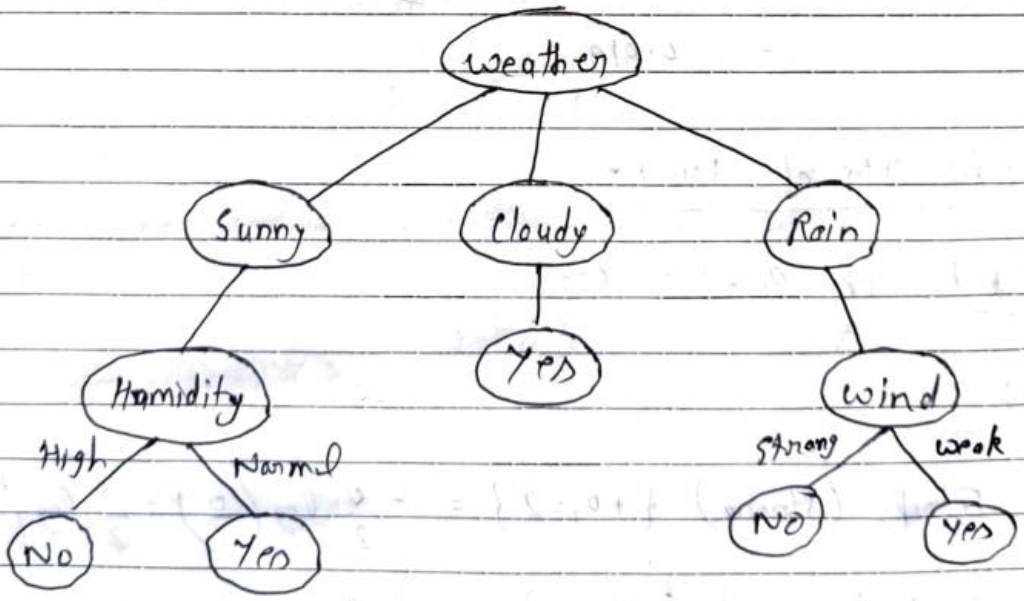
$$Ent(Normal)\{+2,-1\} = -\frac{2}{3}\log\left(\frac{2}{3}\right) - \frac{1}{3}\log\left(\frac{1}{3}\right) = 0.912$$

$$I.G. = Ent(Rain) - \frac{2}{5}Ent(H) - \frac{3}{5}Ent(N) = 0.019$$

$Gain(S_{Rain}, temp) = 0.019$

$Gain(S_{Rain}, Humidity) = 0.019$

$Gain(S_{Rain}, wind) = 0.97$ ✓



* Support Vector Machine -

SVM are a class of supervised machine learning algo used for classification and regression tasks. They are known for their ability to handle high-dimensional data and perform well in a variety of application.

(1) **linear SVM** - linear SVM aim to find a hyperplane that best separate two classes in a linearly separable data set. The goal is to maximize the margin b/w the classes while minimizing the classification error. It is appropriate when the data can be separated by a straight line, a plane or a hyperplane.

(2) **Non linear SVM** - Non linear SVM extend the capabilities of linear SVM to handle data sets that are not linearly separable. They do this by using kernel function to map the original data into a higher dimensional space where it becomes linearly separable.

* Kernel Methods -

Kernel is a function that transforms the I/P data from its original feature space into a higher dimensional space. The purpose of using a kernel in SVM's is to enable the algorithm to find a non linear decision boundary in transformed space, even when the original feature space may not be linearly separable.

(1) Linear kernel - The linear kernel is the simplest kernel function, representing a dot product b/w data points in the original feature space. It is used when the data is linearly separable.

(2) Polynomial kernel - The polynomial kernel is used to transform data into a higher dimensional space, where it can become linearly separable. It introduces non linearity by considering all possible polynomial combinations of the features.

(3) Radial basis function kernel - The RBF kernel also known as the gaussian kernel. It is a popular choice for SVM's in non linear problems. It creates a high dimensional space where data points are transformed based on their similarity to a reference point.

(4) Sigmoid kernel - The sigmoid kernel is another kernel function that can be used in SVM's. It is based on the sigmoid function and can handle data that does not necessarily exhibit polynomial or radial basis function behaviour. It is also used in neural n/w like SVM model.