Module 4

Syllabus

Decision Trees, Regression, Cluster Analysis, Association Rule Mining, Artificial Neural Networks.

4.1 Decision Trees

- Decision trees are a simple way to guide one's path to a decision. The decision may be a simple binary one, whether to approve a loan or not.
- Or it may be a complex multi-valued decision, as to what may be the diagnosis for a particular sickness.
- Decision trees are one of the most widely used techniques for classification.
- A good decision tree should be short and ask only a few meaningful questions.

Decision Tree problem

Imagine a conversation between a doctor and a patient. The doctor asks questions to determine the cause of the ailment. The doctor would continue to ask questions, till she is able to arrive at a reasonable decision. If nothing seems plausible, she might recommend some tests to generate more data and options. This is how experts in any field solve problems. They use decision trees or decision rules. For every question they ask, the potential answers create separate branches for further questioning.

A decision tree would have a predictive accuracy based on how often it makes correct decisions.

1. The more training data is provided, the more accurate its knowledge extraction will be, and thus, it will make more accurate decisions.

2. The more variables the tree can choose from, the greater is the likely of the accuracy of the decision tree.

3. In addition, a good decision tree should also be frugal so that it takes the least number of questions, and thus, the least amount of effort, to get to the right decision.

1) Construct a decision tree that helps make decisions about approving the play of an outdoor game.

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	Normal	True	?

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

Decision Tree Construction

A decision tree is a hierarchically branched structure. First thing is to determine the root node of the tree.

Determining root node of the tree:

In this example, there are four choices based on the four variables. One could begin by asking one of the following questions: what is the outlook, what is the temperature, what is the humidity, and what is the wind speed?

- Start with the first variable, in this case outlook. It can take three values, sunny, overcast, and rainy.

- There are five instances where the outlook is sunny. In 2 of the 5 instances the *play* decision was *yes*, and in the other three, the decision was No.
- Thus, if the decision rule was that Outlook:sunny → No, then 3 out of 5 decisions would be correct, while 2 out of 5 such decisions would be incorrect.
- There are 2 errors out of 5. This can be recorded in Row 1.
- Similar analysis would be done for other values of the outlook variable.
- There are four instances where the outlook is overcast. In all 4 out 4 instances the Play decision was yes.
- Thus, if the decision rule was that Outlook:overcast → Yes, then 4 out of 4 decisions would be correct, while none of decisions would be incorrect.
- There are 0 errors out of 4.
- There are five instances where the outlook is rainy.
- In 3 of the 5 instances the *play* decision was *yes*, and in the other three, the decision was *no*.
- Thus, if the decision rule was that Outlook:rainy→ Yes, then 3 out of 5 decisions would be correct, while 2 out of 5 decisions would be incorrect.
- Adding up errors for all values of outlook, there are 4 errors out of 14. In other words, Outlook gives 10 correct decisions out of 14, and 4 incorrect ones.

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	4/14
	Overcast → Yes	0/4	
	Rainy → Yes	2/5	

- A similar analysis can be done for the other three variables. For temperature, the following error table gives total number of errors.

Attribute	Rules	Error	Select
Temp	Hot → No	2/4	5/14
	Mild → Yes	2/6	
	Cool → Yes	1/4	

- Following error table provides number of errors in humidity

Attribute	Rules	Error	Total Error
Humidity	High → No	3/7	4/14
	Normal → Yes	1/7	

- And similar analysis done for windy, hence the error table becomes,

Attribute	Rules	Error	Total Error
Windy	False → Yes	2/8	5/14
	True → No	3/6	

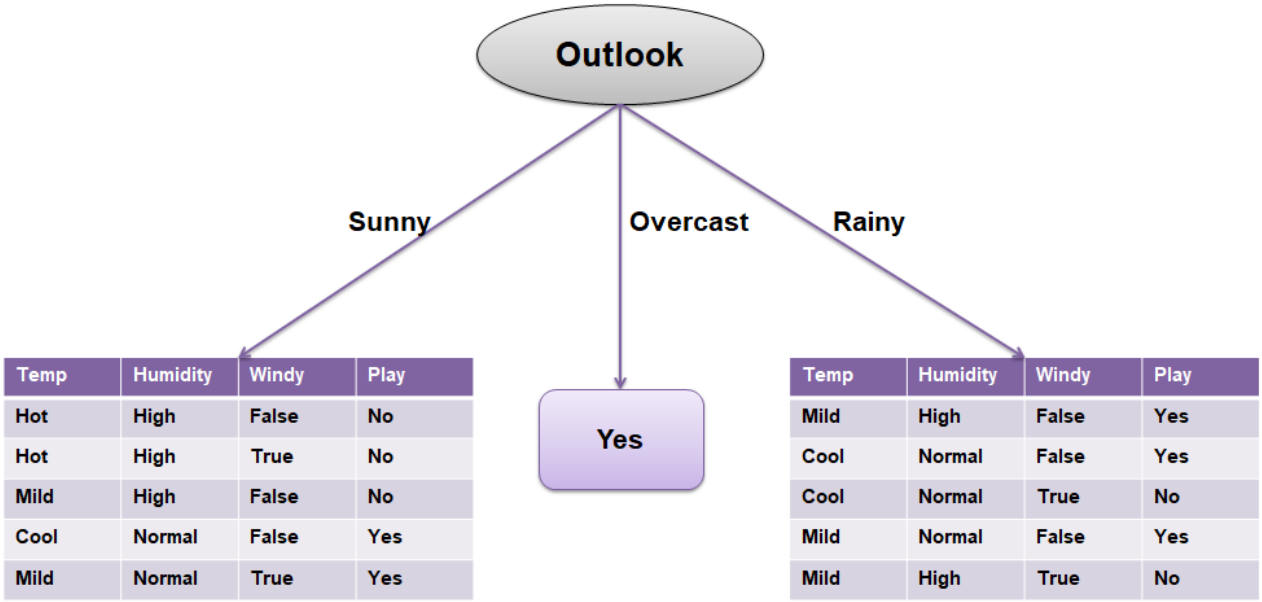
- The variable that leads to the least number of errors (and thus the most number of correct decisions) should be chosen as the first node.
- In this case, two variables have the least number of errors. There is a tie between outlook and humidity, as both have 4 errors out of 14 instances.

Attribute	Rules	Error	Total Error
Outlook	Sunny → No	2/5	4/14
	Overcast → Yes	0/4	
	Rainy → Yes	2/4	
Temperature	Hot → No	2/4	5/14
	Mild → Yes	2/6	
	Cool → Yes	1/4	
Humidity	High → No	3/7	4/14
	Normal → Yes	1/7	
Windy	False → Yes	6/8	5/14
	True → No	3/6	

- The tie can be broken using another criterion, the purity of resulting sub-trees.

***Splitting the Tree*:**

- From the root node, the decision tree will be split into three branches or sub-trees, one for each of the three values of outlook.
- Data for the root node (the entire data) will be divided into the three segments, one for each of the value of outlook.



Temp	Humidity	Windy	Play
Hot	High	False	No
Hot	High	True	No
Mild	High	False	No
Cool	Normal	False	Yes
Mild	Normal	True	Yes

Temp	Humidity	Windy	Play
Mild	High	False	Yes
Cool	Normal	False	Yes
Cool	Normal	True	No
Mild	Normal	False	Yes
Mild	High	True	No

Determining the next nodes of the tree:

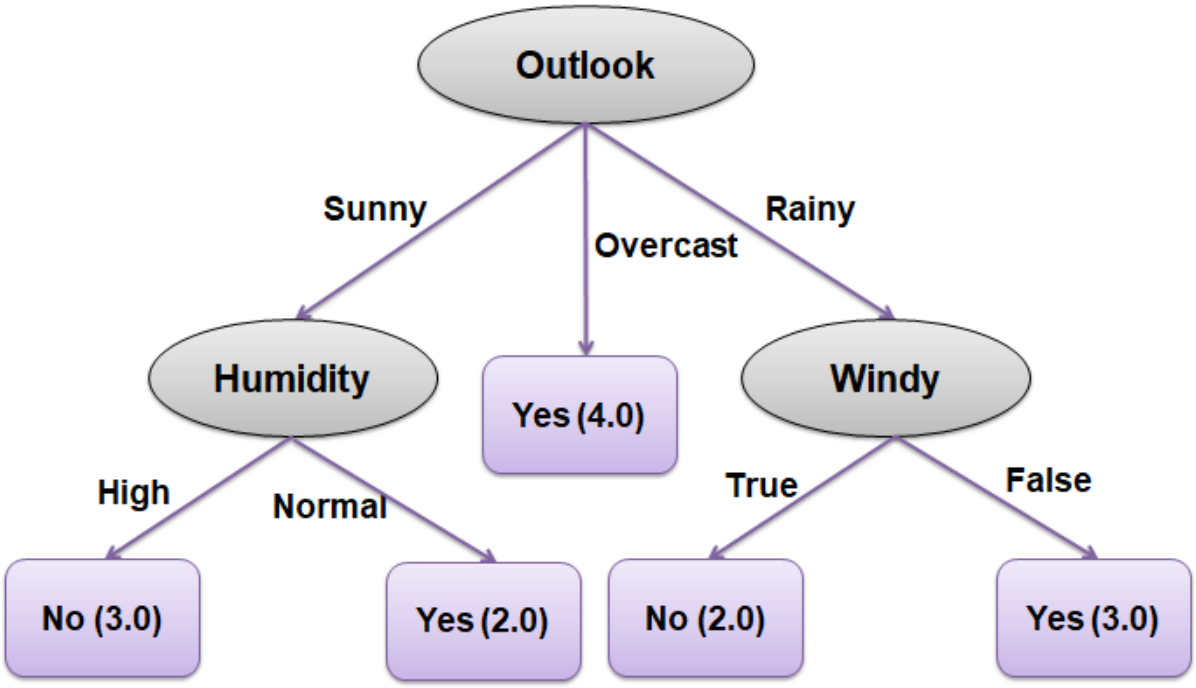
- Similar recursive logic of tree building should be applied to each branch. For the sunny branch on the left, error values will be calculated for the three other variables – temp, humidity and windy.
  - The variable of humidity shows the least amount of error, i.e. zero error. The other two variables have non-zero errors.
  - Thus the Outlook: sunny branch on the left will use humidity as the next splitting variable.

Attribute	Rules	Error	Total Error
Temperature	Hot → No	0/2	1/5
	Mild → No	1/2	
	Cool → Yes	0/1	
Humidity	High → No	0/3	0/5
	Normal → Yes	0/2	
Windy	False → No	1/3	2/5
	True → No	1/2	

- Similar analysis should be done for the 'rainy' value of the tree. The analysis would look like this.
- For the Rainy branch, it can similarly be seen that the variable Windy gives all the correct answers, while none of the other two variables makes all the correct decisions.
- Thus the Outlook: rainy branch on the right will use windy as the next splitting variable.

Attribute	Rules	Error	Total Error
Temperature	Mild → Yes	1/3	2/5
	Cool → Yes	1/2	
Humidity	High → No	1/2	2/5
	Normal → Yes	1/3	
Windy	False → Yes	0/3	0/5
	True → No	0/2	

- Hence we can construct a final decision tree with root node as outlook



- So, the decision problem moves to the Sunny branch of the tree. The node in that sub-tree is humidity. In the problem, Humidity is Normal.
- That branch leads to an answer Yes. Thus, the answer to the play problem is Yes.

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	Normal	True	YES

2) Compare decision tree with the lookup table

	Decision Tree	Table Lookup
Accuracy	Varied level of accuracy	100% accurate
Generality	General. Applies to all situations	Applies only when a similar case had occurred earlier
Frugality	Only three variables needed	All four variables are needed
Simple	Only one, or max two variable values are needed	All four variable values are needed
Easy	Logical, and easy to understand	Can be cumbersome to look up; no understanding of the logic behind the decision

3) Write a pseudo code(Algorithm) for making decision trees.

The following is a pseudo code for making decision trees:

Step1: Create a root node and assign all of the training data to it.

Step 2: Select the best splitting attribute according to certain criteria.

Step 3: Add a branch to the root node for each value of the split.

Step 4: Split the data into mutually exclusive subsets along the lines of the specific split.

Step 5: Repeat steps 2 and 3 for each and every leaf node until a stopping criteria is reached.

I. Splitting criteria

1. Which variable to use for the first split? How should one determine the most important variable for the first branch, and subsequently, for each sub-tree? There are many measures like least errors, information gain, gini's coefficient, etc.

2. What values to use for the split? If the variables have continuous values such as for age or blood pressure, what value-ranges should be used to make bins?

3. How many branches should be allowed for each node? There could be binary trees, with just two branches at each node. Or there could be more branches allowed.

II. Stopping criteria:

- When to stop building the tree? There are two major ways to make that determination.
- The tree building could be stopped when a certain depth of the branches has been reached and the tree becomes unreadable after that.

- The tree could also be stopped when the error level at any node is within predefined tolerable levels.

III. Pruning:

- The tree could be trimmed to make it more balanced and more easily usable.
- The pruning is often done after the tree is constructed, to balance out the tree and improve usability.
- The symptoms of an overfitted tree are a tree too deep, with too many branches, some of which may reflect anomalies due to noise or outliers. Thus, the tree should be pruned.

4.2 Regression Analysis

1) Explain Regression along with its steps.

Regression is a well-known statistical technique to model the predictive relationship between several independent variables (DVs) and one dependent variable. The objective is to find the best-fitting curve for a dependent variable in a multidimensional space, with each independent variable being a dimension. The curve could be a straight line, or it could be a nonlinear curve. The quality of fit of the curve to the data can be measured by a coefficient of correlation (r), which is the square root of the amount of variance explained by the curve.

Linear Regression Analysis

- Analysis of the strength of the ***linear relationship*** between predictor (independent) variables and outcome (dependent/criterion) variables.
- In two dimensions (one predictor, one outcome variable) data can be plotted on a scatter diagram.

$$Y=mX+b$$

Here,

- Y is the dependent variable we are trying to predict.
- X is the independent variable we are using to make predictions.
- m is coefficient and b is intercept

The key steps for regression are simple:

1. List all the variables available for making the model.
2. Establish a Dependent Variable (DV) of interest.
3. Examine visual (if possible) relationships between variables of interest.
4. Find a way to predict DV using the other variables.

2) List out the advantages and disadvantages of regression models.

Regression Models are very popular because they offer many advantages:

- Regression models are easy to understand as they are built upon basic statistical principles such as correlation and least square error.
- Regression models provide simple algebraic equations that are easy to understand and use.
- The strength (or the goodness of fit) of the regression model is measured in terms of the correlation coefficients, and other related statistical parameters that are well understood.
- Regression models can match and beat the predictive power of other modeling techniques.
- Regression models can include all the variables that one wants to include in the model.
- Regression modeling tools are pervasive. They are found in statistical packages as well as data mining packages. MS Excel spreadsheets can also provide simple regression modeling capabilities.

Regression models can however prove inadequate under many circumstances.

- Regression models can not cover for poor data quality issues. If the data is not prepared well to remove missing values, or is not well-behaved in terms of a normal distribution, the validity of the model suffers.
- Regression models suffer from collinearity problems (meaning strong linear correlations among some independent variables). If the independent variables have strong correlations among themselves, then they will eat into each other's predictive power and the regression coefficients will lose their ruggedness. Regression models will not automatically choose between highly collinear variables, although some packages attempt to do that.
- Regression models can be unwieldy and unreliable if a large number of variables are included in the model. All variables entered into the model will be reflected in the regression equation, irrespective of their contribution to the predictive power of the model. There is no concept of automatic pruning of the regression model.
- Regression models do not automatically take care of non-linearity. The user needs to imagine the kind of additional terms that might be needed to be added to the regression model to improve its fit.
- Regression models work only with numeric data and not with categorical variables. There are ways to deal with categorical variables though by creating multiple new variables with a yes/no value.

4.3 Cluster Analysis

Cluster analysis is used for automatic identification of natural groupings of things. It is also known as the segmentation technique.

- In this technique, data instances that are similar to (or near) each other are categorized into one cluster.
- Similarly, data instances that are very different (or far away) from each other are moved into different clusters.
- Clustering is an unsupervised learning technique as there is no output or dependent variable for which a right or wrong answer can be computed.

1) Define a cluster.

An operational definition of a cluster is that, given a representation of n objects, find K groups based on a measure of similarity, such that objects within the same group are similar but the objects in different groups are not similar.

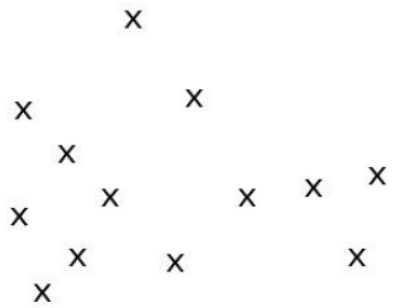


Figure A: Cluster

2) What are the Applications of Cluster Analysis?

1. ***Market Segmentation*:**

Categorizing customers according to their similarities, for instance by their common wants and needs, and propensity to pay, can help with targeted marketing.

2. ***Product portfolio*:**

People of similar sizes can be grouped together to make small, medium and large sizes for clothing items.

3. ***Text Mining*:**

Clustering can help organize a given collection of text documents according to their content similarities into clusters of related topics.

Note: *A cluster can be defined as the **centroid** of the collection of points belonging to it. A **centroid** is a measure of central tendency. It is the point from where the sum total of squared distance from all the points is the minimum.*

3) Mention the generic pseudocode for clustering.

1. Pick an arbitrary number of groups/segments to be created
2. Start with some initial randomly-chosen center values for groups
3. Classify instances to closest groups
4. Compute new values for the group centers.
5. Repeat step 3 & 4 till groups converge
6. If clusters are not satisfactory, go to step 1 and pick a different number of groups/segments.

4) Identify and form clusters for the following data.

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

A scatter plot of 10 items in 2 dimensions shows them distributed fairly randomly.

- The points are distributed randomly en that it could be considered one cluster as shown in Figure 4.1. The solid circle would represent the central point (centroid) of these points.

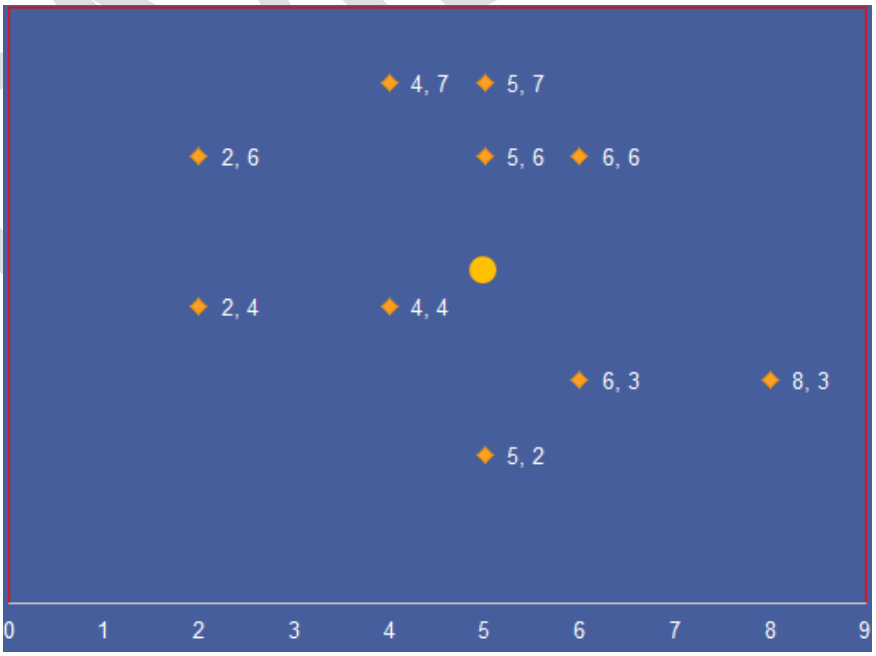


Figure 4.1

- There is a big distance between the points (2,6) and (8,3). This data could be broken into 2 clusters. The three points at the bottom right could form one cluster and the other seven could form the other cluster as shown in Figure 4.2.

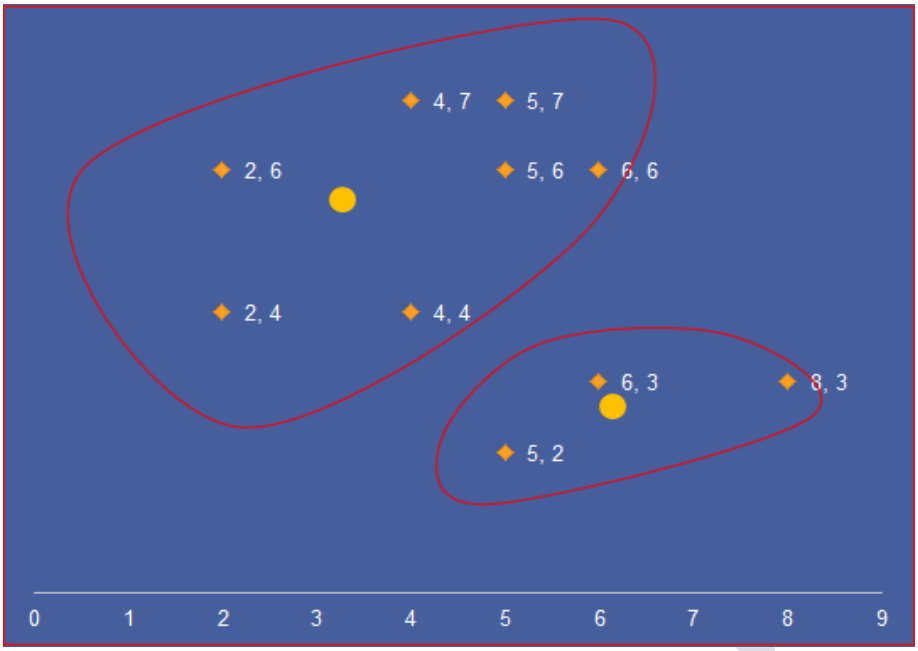


Figure 4.2

- The bigger cluster seems too far apart. It seems like the 4 points on the top will form a separate cluster. The three clusters could look like in Figure 4.3.

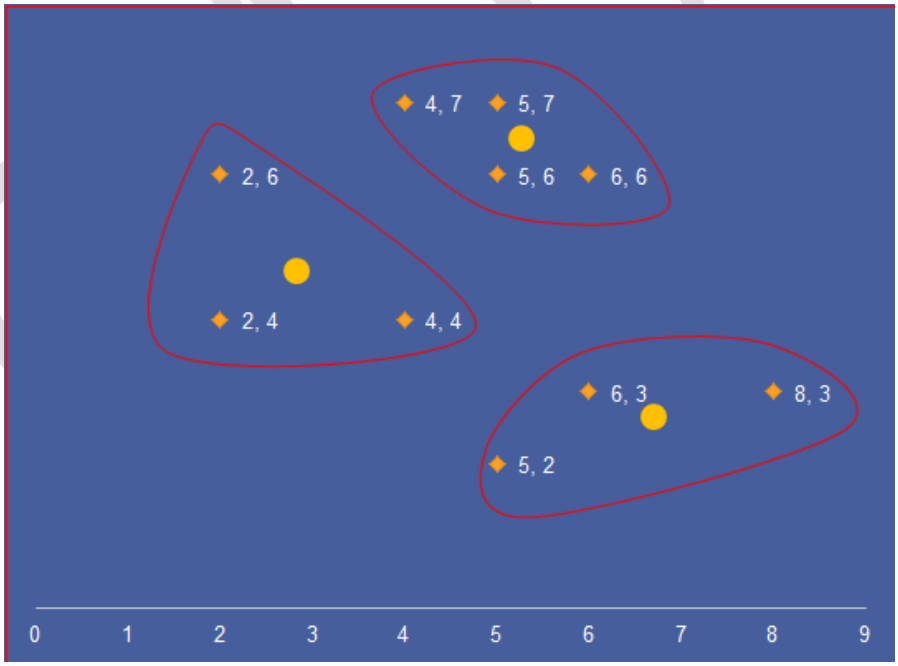


Figure 4.3

5) Demonstrate the steps involved in K- Means algorithm for clustering with an example.

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

Points to be remembered:

- K-means is the most popular clustering algorithm.
- It iteratively computes the clusters and their centroids.
- It is a top down approach to clustering.

Start with a given number of K clusters, say 3 clusters. Thus three random centroids will be created as starting points of the centers of three clusters. The circles are initial cluster centroids (Figure 4.4).

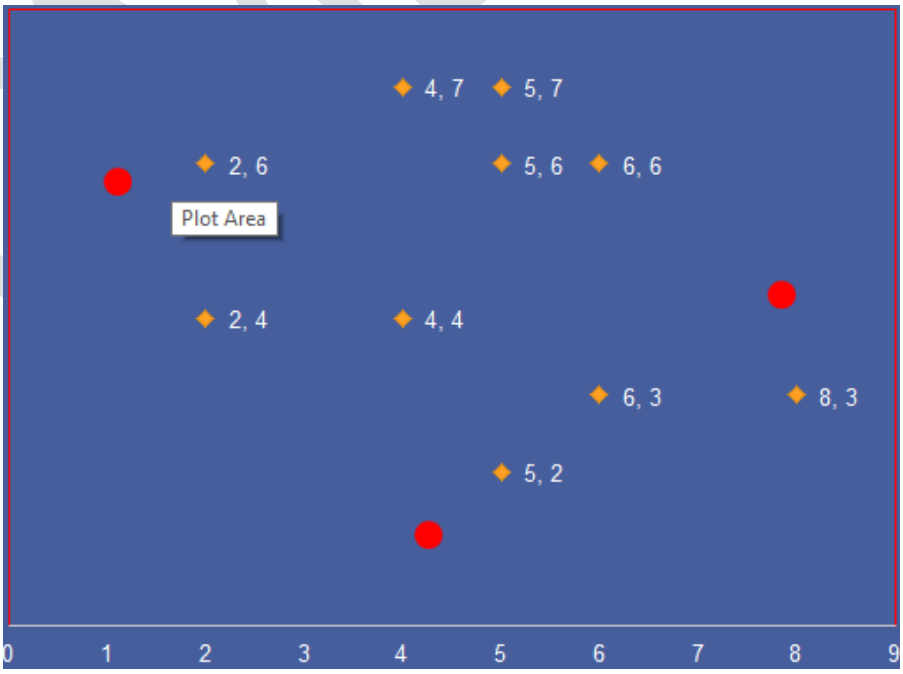


Figure 4.4

Step 1: For a data point, distance values will be from each of the three centroids. The data point will be assigned to the cluster with the shortest distance to the centroid. All data points will thus, be assigned to one data point or the other (Figure 4.5). The arrows from each data element show the centroid that the point is assigned to.

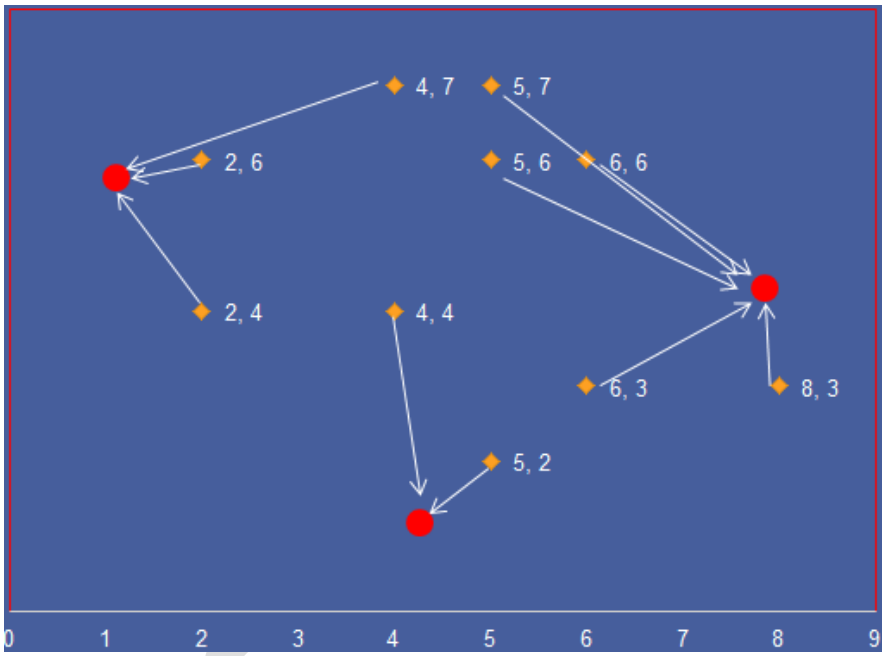


Figure 4.5

- ***Step 2:*** The centroid for each cluster will now be recalculated such that it is closest to all the data points allocated to that cluster. The dashed arrows show the centroids being moved from their old (shaded) values to the revised new values (Figure 4.6)

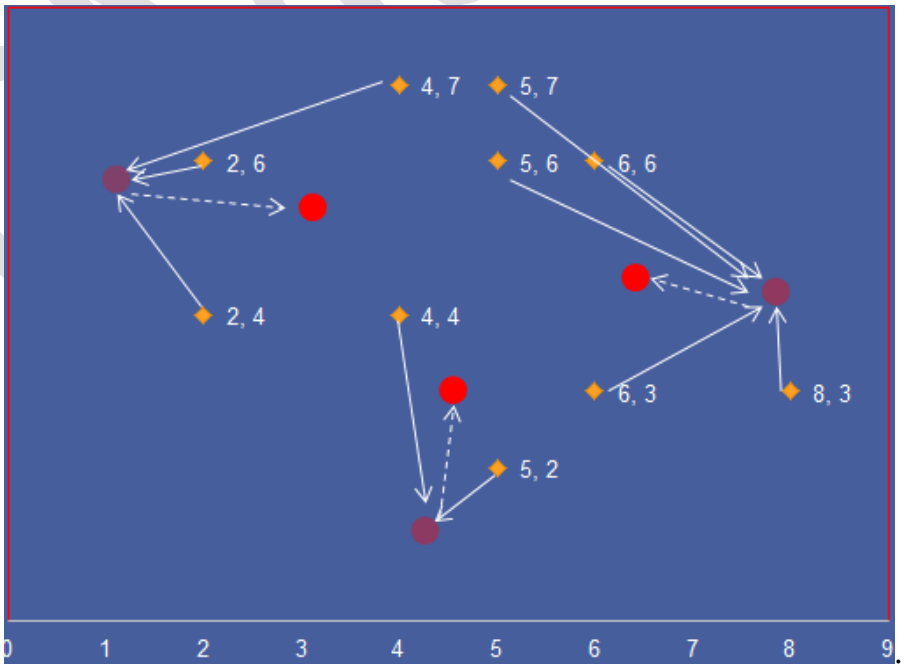


Figure 4.6

- ***Step 3:*** Once again, data points are assigned to the three centroids closest to it (Figure 4.7).

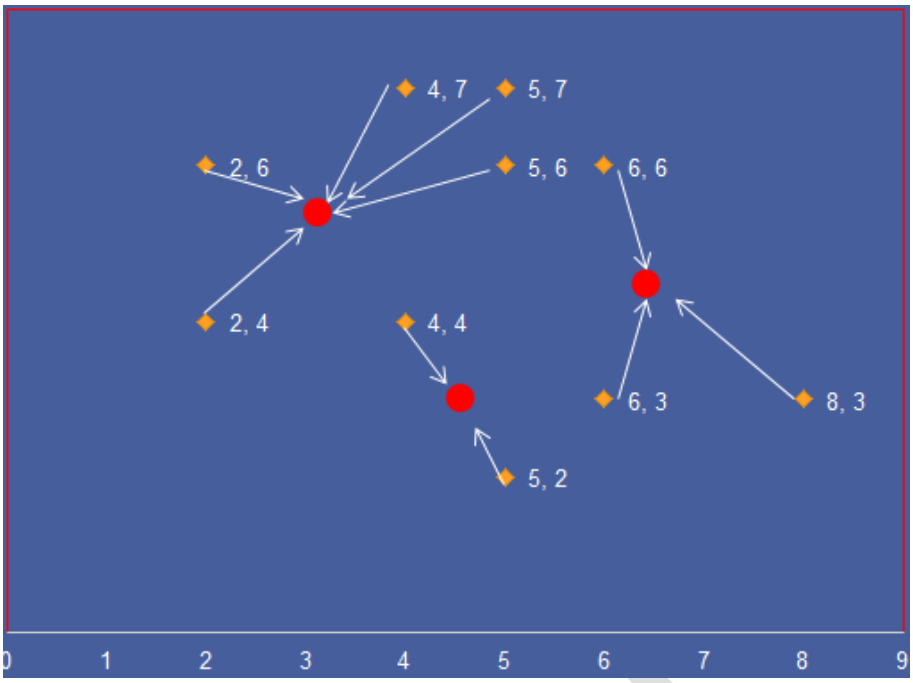


Figure 4.7

- The new centroids will be computed from the data points in the cluster until finally, the centroids stabilize in their locations. These are the three clusters computed by this algorithm.
- The three clusters shown are: a 3-datapoints cluster with centroid (6.5,4.5), a 2- datapoint cluster with centroid (4.5,3) and a 5-datapoint cluster with centroid (3.5,3) (Figure 4.8).

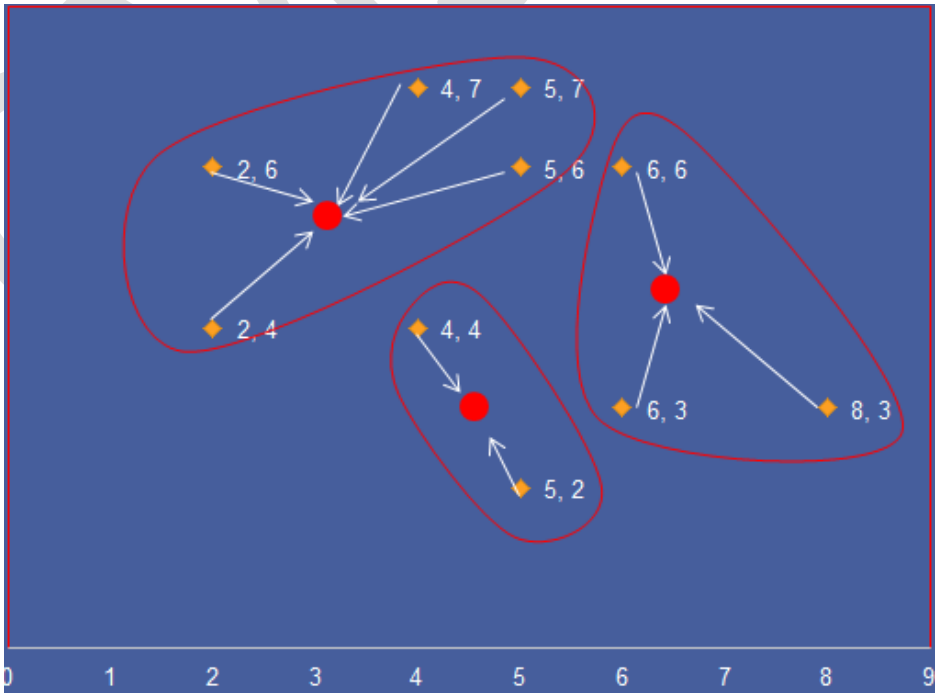


Figure 4.8

6) Write K- means Algorithm for clustering.

Algorithm **K-means**(K number of clusters, D list of data points)

1. Choose K number of random data points as initial centroids (cluster centers)
2. Repeat the process till cluster-centers stabilize.
    a. Allocate each point in D to the nearest of K centroids;
    b. Compute centroid for the cluster using all points in the cluster.
    c.

OR

Algorithm *k*-means(*k*, *D*)

```
1  Choose k data points as the initial centroids (cluster centers)
2  repeat
3     for each data point x ∈ D do
4        compute the distance from x to each centroid;
5        assign x to the closest centroid          // a centroid represents a cluster
6     endfor
7     re-compute the centroids using the current cluster memberships
8  until the stopping criterion is met
```

7) Mention the Advantages and Disadvantages of K-Means algorithm.

Advantages of K-Means Algorithm

1. K-Means algorithm is simple, easy to understand and easy to implement.

2. It is also efficient, in that the time taken to cluster k-means, rises linearly with the number of data points.

3. No other clustering algorithm performs better than K-Means, in general.

Disadvantages of K-Means Algorithm

1. The user needs to specify an initial value of K.

2. The process of finding the clusters may not converge.

3. It is not suitable for discovering clusters shapes that are not hyper ellipsoids.

4.4 Association rule mining.

1) Explain Association rule mining.

- A very popular DM method in business
- Also known as market basket analysis
- Finds interesting relationships (affinities) between variables (items or events)
- Assume all data are categorical.
- Employs unsupervised learning
- There is no output variable
- Part of machine learning family
- Often used as an example to describe DM to ordinary people, such as the famous "relationship between diapers and beers!"
- Input: the simple point-of-sale transaction data
- Output: Most frequent affinities among items
- Example: according to the transaction data…

  "Customer who bought a laptop computer and a virus protection software, also bought extended service plan 70 percent of the time."

2) Explain the application of Association rule mining.

- Applications of association rule mining
- In business
  - cross-marketing, cross-selling
  - store design, catalog design, e-commerce site design
  - optimization of online advertising
  - product pricing, and sales/promotion configuration
- In medicine
  - relationships between symptoms and illnesses
  - diagnosis and patient characteristics and treatments
  - genes and their functions (genomics projects)…

Sample example

Raw Transaction Data

Transaction No	SKUs (Item No)
1	1, 2, 3, 4
1	2, 3, 4
1	2, 3
1	1, 2, 4
1	1, 2, 3, 4
1	2, 4

One-item Itemsets

Itemset (SKUs)	Support
1	3
2	6
3	4
4	5

Two-item Itemsets

Itemset (SKUs)	Support
1, 2	3
1, 3	2
1, 4	3
2, 3	4
2, 4	5
3, 4	3

Three-item Itemsets

Itemset (SKUs)	Support
1, 2, 4	3
2, 3, 4	3

Problem 1: Apply Apriori for the following example. Assume support count is 2

Transaction	List of items
T1	I1,I2,I5
T2	I2,I4
T3	I2,I3
T4	I1,I2,I4
T5	I1,I3
T6	I2,I3
T7	I1,I3
T8	I1,I2,I3,I5
T9	I1,I2,I3

find out the occurrences of 1-itemset

Table 1: Occurrence of 1-itemset

Item	Count
I1	6
I2	7
I3	6
I4	2
I5	2

All items satisfy min_sup count=2. So not removing any items from Table 1.

find out the occurrences of 2-itemset

Table 2: Occurrence of 2-itemset

Item	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

Item sets (I1,I4), (I3,I4), (I3,I5) and (I4,I5) does not satisfies min_sup = 2. So remove these items from Table 2.

Table 3: List of items satisfy min_sup=2

Item	Count
I1,I2	4
I1,I3	4
I1,I5	2
I2,I3	4
I2,I4	2
I2,I5	2

find out the occurrences of 3-itemset

Table 4: Occurrence of 3-itemset

Item	Count
I1,I2,I3	2
I1,I2,I4	0
I1,I2,I5	2
I1,I3,I4	0
I1,I3,I5	1
I1,I4,I5	0
I2,I3,I4	0
I2,I3,I5	1
I3,I4,I5	0

Only (I1, I2, I3) and (I1,I2,I5) satisfies min_sup=3. So remove remaining item sets from Table 4.

Table 5: List of items satisfy min_sup=3

Item	Count
I1,I2,I3	2
I1,I2,I5	2

Generate Association Rules: From the frequent itemset discovered above the association could be:

$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence (A} \longrightarrow \text{B)} = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

{I1, I2} => {I3}

Confidence = support {I1, I2, I3} / support {I1, I2} = (2/ 4)* 100 = 50%

{I1, I3} => {I2}

Confidence = support {I1, I2, I3} / support {I1, I3} = (2/ 4)* 100 = 50%

{I2, I3} => {I1}

Confidence = support {I1, I2, I3} / support {I2, I3} = (2/ 4) * 100 = 50%

{I1} => {I2, I3}

Confidence = support {I1, I2, I3} / support {I1} = (2/ 6) * 100 = 33.33%

{I2} => {I1, I3}

Confidence = support {I1, I2, I3} / support {I2} = (2/ 7) * 100 = 28.57 %

{I3} => {I1, I2}

Confidence = support {I1, I2, I3} / support {I3} = (2/ 6) * 100 = 33.33 %

{I1, I2} => {I5}

Confidence = support {I1, I2, I5} / support {I1, I2}

= (2/ 4) * 100 = 50%

{I1, I5} => {I2}

Confidence = support {I1, I2, I5} / support {I1, I5}

= (2/ 2) * 100 = 100%

{I2, I5} => {I1}

Confidence = support {I1, I2, I5} / support {I2, I5} = (2/ 2) * 100 = 100%

{I1} => {I2, I5}

Confidence = support {I1, I2, I5} / support {I1} = (2/ 6) * 100 = 33.33%

{I2} => {I1, I5}

Confidence = support {I1, I2, I5} / support {I2} = (2/ 7) * 100 = 28.57 %

{I3} => {I1, I2}

Confidence = support {I1, I2, I5} / support {I3} = (2/ 6) * 100 = 33.33 %

Problem 2: Apply Apriori for the following example. Assume Support threshold=50%, Confidence= 60%

Transaction	List of items
T1	I1,I2,I3
T2	I2,I3,I4
T3	I4,I5
T4	I1,I2,I4
T5	I1,I2,I3,I5
T6	I1,I2,I3,I4

Support threshold=50% => 0.5*6= 3 => min_sup = 3

find out the occurrences of 1-itemset

Table 1: Occurrence of 1-itemset

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

In Table 1, I5 item does not meet min_sup=3, thus it is deleted, only I1, I2, I3, I4 meet min_sup count.

Table 2: List of items satisfy min_sup=3

Item	Count
I1	4
I2	5
I3	4
I4	4

find out the occurrences of 2-itemset

Table 3: Occurrence of 2-itemset

Item	Count
I1,I2	4
I1,I3	3
I1,I4	2
I2,I3	4
I2,I4	3
I3,I4	2

In Table 3, item set {I1, I4} and {I3, I4} does not satisfies min_sup=3, thus it is deleted.

Table 4: List of items satisfy min_sup=3

Item	Count
I1,I2	4
I1,I3	3
I2,I3	4
I2,I4	3

find out the occurrences of 3-itemset

Table 3: Occurrence of 3-itemset

Item	Count
I1,I2,I3	3
I1,I2,I4	2
I1,I3,I4	1
I2,I3,I4	2

Only {I1, I2, I3} meet min_sup=3

Hence {I1, I2, I3} is frequent item set.

Generate Association Rules: From the frequent itemset discovered above the association could be:

$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence } (A \longrightarrow B) = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

{I1, I2} => {I3}

Confidence = support {I1, I2, I3} / support {I1, I2} = (3/ 4)* 100 = 75%

{I1, I3} => {I2}

Confidence = support {I1, I2, I3} / support {I1, I3} = (3/3)* 100 = 100%

{I2, I3} => {I1}

Confidence = support {I1, I2, I3} / support {I2, I3} = (3/ 4) * 100 = 75%

{I1} => {I2, I3}

Confidence = support {I1, I2, I3} / support {I1} = (3/4) * 100 = 75%

{I2} => {I1, I3}

Confidence = support {I1, I2, I3} / support {I2} = (3/5) * 100 = 75 %

{I3} => {I1, I2}

Confidence = support {I1, I2, I3} / support {I3} = (3/4) * 100 = 75%

Problem 3: Create an association rule for the following data with 50% support and 70% confidence

ID	Items
1	Bread, cheese, egg, juice
2	Bread, cheese, juice
3	Bread, milk, yogurt
4	Bread, juice, milk
5	Cheese, juice, milk

Date / /20

Given,

Support = 50%

Confidence = 70%

No of transactions = 5.

1 Item set	frequency	Support
Bread	4	4/5 = 80% ✓
Cheese	3	3/5 = 60% ✓
Egg	1	1/5 = 20%
Milk	3	3/5 = 60% ✓
Yogurt	1	1/5 = 20%
Juice	4	4/5 = 80% ✓

Since minimum support is 50%. Comparing support of each item with minimum support we get,

1 Item Set	frequency
Bread	4
Cheese	3
Milk	3
Juice	4

2 Item Set	frequency	Support
(Bread, Cheese)	2	2/5 = 40%
(Bread, Milk)	2	2/5 = 40%
(Bread, Juice)	3	3/5 = 60% ✓
(Cheese, Milk)	1	1/5 = 20%
(Cheese, Juice)	3	3/5 = 60% ✓
(Milk, Juice)	2	2/5 = 40%

Date / /20

2 Item Set	frequency
(Bread, Juice)	3
(Cheese, Juice)	3

3 Item Set	frequency	Support
(Bread, Juice, Cheese)	2	2/5 ≯ 50%

Since Support of 3 itemset is < 50%. we take (Bread, Juice) i.e, Bread → Juice
Juice → Bread
(Cheese, juice) i.e, Juice → cheese
Cheese → Juice

Bread → Juice :

$$\text{Confidence}(\text{Bread} \rightarrow \text{Juice}) = \frac{\text{Support}(B \cup J)}{\text{Support}(B)}$$

$$= \frac{3}{4} = 75\%$$

Juice → Bread

$$\text{Confidence}(\text{Juice} \rightarrow \text{Bread}) = \frac{\text{Support}(J \cup B)}{\text{Support}(J)}$$

$$= \frac{3}{4} = 75\%$$

Juice → cheese

$$\text{Confidence}(\text{Juice} \rightarrow \text{Cheese}) = \frac{\text{Support}(J \cup C)}{\text{Support}(J)}$$

$$= \frac{3}{4} = 75\%$$

Cheese → Juice.

Confidence (Cheese → Juice) = $\frac{\text{Support}(C \cup J)}{\text{Support}(C)}$

$= \frac{3}{3} = 100\%$

4.5 Artificial Neural Networks

1. Explain the design principle of Artificial Neural Networks.

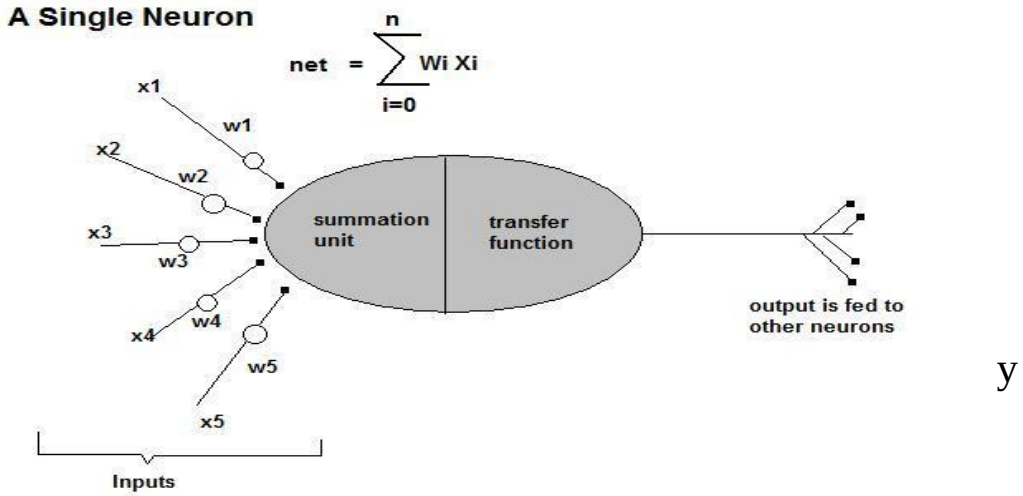


Figure 4.9: Model for a single artificial neuron

1. A neuron is the basic processing unit of the network. The neuron (or processing element) receives inputs from its preceding neurons (or PEs), does some nonlinear weighted computation on the basis of those inputs, transforms the result into its output value, and then passes on the output to the next neuron in the network (Figure 4.9). X's are the inputs, w's are the weights for each input, and y is the output.
2. A Neural network is a multi-layered model. There is at least one input neuron, one output neuron, and at least one processing neuron. An ANN with just this basic structure would be a simple, single-stage computational unit. A simple task may be processed by just that one neuron and the result may be communicated soon. ANNs however, may have multiple layers of processing elements in sequence. There could be many neurons involved in a sequence depending upon the complexity of the predictive action. The layers of PEs could work in sequence, or they could work in parallel (Figure 5.0).

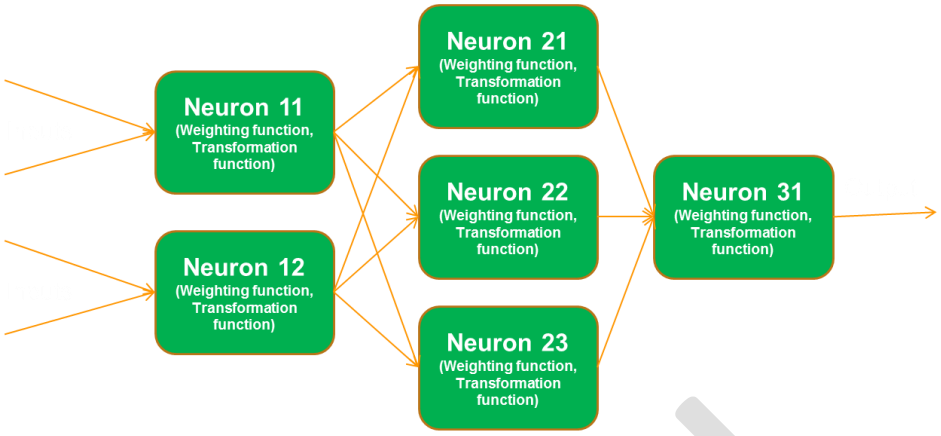


Figure 5.0: Model for a multi-layer ANN

3. The processing logic of each neuron may assign different weights to the various incoming input streams. The processing logic may also use nonlinear transformation, such as a sigmoid function, from the processed values to the output value. This processing logic and the intermediate weight and processing functions are just what works for the system as a whole, in its objective of solving a problem collectively. Thus, neural networks are considered to be an opaque and a black-box system.
4. The neural network can be trained by making similar decisions over and over again with many training cases. It will continue to learn by adjusting its internal computation and communication based on feedback about its previous decisions. Thus, the neural networks become better at making a decision as they handle more and more decisions.

Depending upon the nature of the problem and the availability of good training data, at some point the neural network will learn enough and begin to match the predictive accuracy of a human expert. In many practical situations, the predictions of ANN, trained over a long period of time with a large number of training data, have begun to decisively become more accurate than human experts. At that point ANN can begin to be seriously considered for deployment in real situations in real time.

2) Explain Business Applications of ANN

Neural networks are used most often when the objective function is complex, and where there exists plenty of data, and the model is expected to improve over a period of time. A few sample applications:

1. They are used in stock price prediction where the rules of the game are extremely complicated, and a lot of data needs to be processed very quickly.

2. They are used for character recognition, as in recognizing hand-written text, or damaged or mangled text. They are used in recognizing fingerprints. These are complicated patterns and are unique for each person. Layers of neurons can progressively clarify the pattern leading to a remarkably accurate result.

3. They are also used in traditional classification problems, like approving a financial loan application.

3) Discuss steps to developing an ANN

It takes resources, training data, skill and time to develop a neural network. Most data mining platforms offer at least the Multi-Layer-Perceptron (MLP) algorithm to implement a neural network. Other neural network architectures include Probabilistic networks and Self organizing feature maps.

The steps required to build an ANN are as follows:

1. Gather data. Divide into training data and test data. The training data needs to be further divided into training data and validation data.
2. Select the network architecture, such as Feedforward network.
3. Select the algorithm, such as Multi-Layer Perception.
4. Set network parameters.
5. Train the ANN with training data.
6. Validate the model with validation data.
7. Freeze the weights and other parameters.
8. Test the trained network with test data.
9. Deploy the ANN when it achieves good predictive accuracy.

4) Discus Advantages and Disadvantages of using ANNs

There are many benefits of using ANN.

1. ANNs impose very little restrictions on their use. ANN can deal with (identify/model) highly nonlinear relationships on their own, without much work from the user or analyst. They help find practical data-driven solutions where algorithmic solutions are non-existent or too complicated.
2. There is no need to program neural networks, as they learn from examples. They get better with use, without much programming effort.
3. They can handle a variety of problem types, including classification, clustering, associations, etc.
4. ANN are tolerant of data quality issues and they do not restrict the data to follow strict normality and/or independence assumptions.
5. They can handle both numerical and categorical variables.
6. ANNs can be much faster than other techniques.
7. Most importantly, they usually provide better results (prediction and/or clustering) compared to statistical counterparts, once they have been trained enough.

The key disadvantages arise from the fact that they are not easy to interpret or explain or compute.

1. They are deemed to be black-box solutions, lacking explain ability. Thus they are difficult to communicate about, except through the strength of their results.
2. Optimal design of ANN is still an art: it requires expertise and extensive experimentation.
3. It can be difficult to handle a large number of variables (especially the rich nominal attributes).
4. It takes large data sets to train an ANN.