L. J Institutes of Engineering and Technology

Remedial MSE List of Questions

SEM: 7
Subject Name: - Machine Learning
Subject Code: 3170724

1. **What are the different types of Supervised Learning approaches? Explain by giving examples.**

Answer:

Types of Supervised Learning

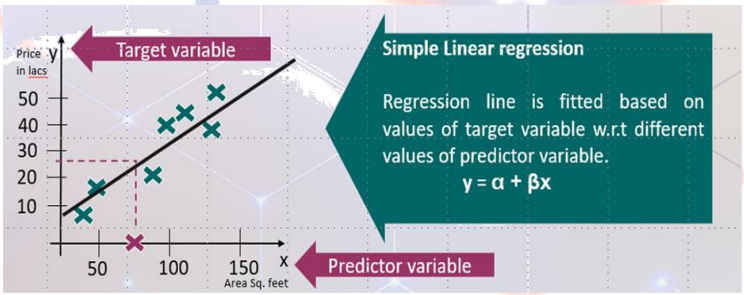
Supervised Learning has been broadly classified into 2 types.

- Regression
- Classification

Regression is the kind of Supervised Learning that learns from the Labelled Datasets and is then able to predict a continuous-valued output for the new data given to the algorithm. It is used whenever the output required is a number such as money or height etc.

Linear Regression – This algorithm assumes that there is a linear relationship between the 2 variables, Input (X) and Output (Y), of the data it has learnt from. The Input variable is called the Independent Variable and the Output variable is called the Dependent Variable. When unseen data is passed to the algorithm, it uses the function, calculates and maps the input to a continuous value for the output.

Example:



Predicting price of house based on area in square feet.

Classification is type of supervised learning where target feature, which is of type categorical, is predicted for test data based on information imparted by training data. The target categorical feature is known as **class**.

Example:

Spam Detection – This application is used where the unreal or computer-based messages and E-Mails are to be blocked. It has an algorithm that learns the different keywords which could be fake such as “You are the winner of something” and so forth and blocks those

messages directly. Many app gives the user the task of making the application learn which keywords need to be blocked and the app will block those messages with the keyword.

Name	Aptitude	Communication	Class
Karuna	2	5	Speaker
Bobby	5	3	Intel
Bhuvana	2	6	Speaker
Ravi	6	2	Intel

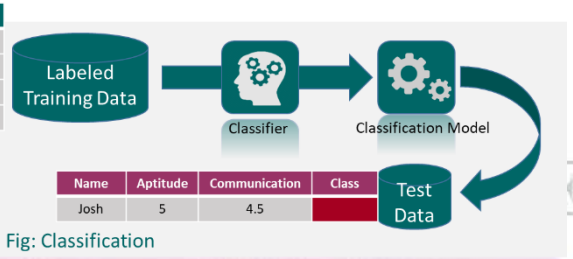


Fig: Classification

Whole problem revolves around assigning label or category to test data based on label or category that is imparted by training data. Algorithms: Naïve Bayes, Decision tree, K-nearest neighbor, support vector machine.

2. **What are the different types of Unsupervised Learning approaches? Explain by giving examples.**

Answer:

Types of Unsupervised Learning

Unsupervised Learning has been split up majorly into 2 types:

- Clustering
- Association

Clustering is the type of Unsupervised Learning where you find patterns in the data that you are working on. It may be the shape, size, colour etc. which can be used to group data items or create clusters. It tends to group or organize similar objects together. Object belonging to same cluster are similar to each other. Objects belonging to different clusters are quite dissimilar.

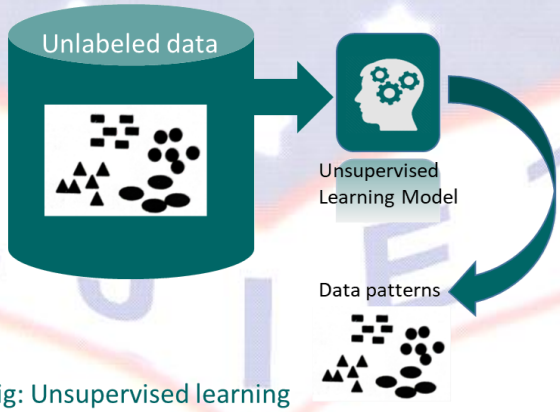


Fig: Unsupervised learning

Example:

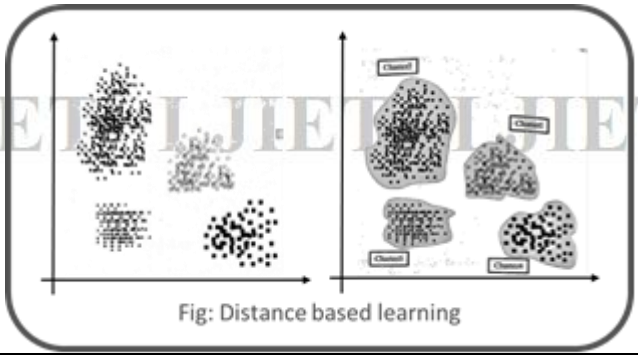


Fig: Distance based learning

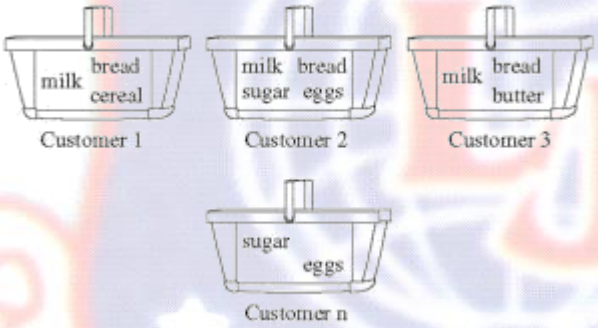
Credit-Card Fraud Detection –algorithms learn about various patterns of the user and their usage of the credit card. If the card is used in parts that do not match the behavior, an alarm is generated which could possibly be marked fraud and calls are given to you to confirm whether it was you using the card or not.

Association is the kind of Unsupervised Learning where you find the dependencies of one data item to another data item and map them such that they help you profit better. Association between data elements is identified.

Example:

Market basket analysis, Recommender System

Apriori algorithm – The Apriori Algorithm is a breadth-first search based which calculates the support between items. This support basically maps the dependency of one data item with another which can help us understand what data item influences the possibility of something happening to the other data item. For example, bread influences the buyer to buy milk and eggs. So that mapping helps increase profits for the store. That sort of mapping can be learnt using this algorithm which yields rules as for its output.



3. **What are main activities involved in machine learning? What are main activities involved when you are preparing to start with modelling in machine?**

Answer:

Main Activities in Machine Learning are :

1. preparing to model
2. Learning
3. Performance evaluation
4. Performance improvement

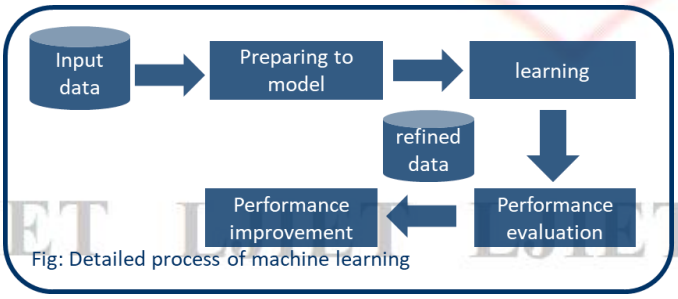


Fig: Detailed process of machine learning

While *preparing to model* following are the main activities:

- Understand the **type of data** in given input data set
- Explore the data to understand **data quality** and **nature**
- Explore **relationship** among data elements(inter-feature relationship)

- Find **potential issues** in data
- **Remediate** data, if needed(impute missing values)
- Apply **pre-processing** steps, a necessary:
  - ✓ Dimensionality reduction
  - ✓ Feature subset selection

While *learning* following are the main activities:

- Data partitioning/ holdout
- Model selection
- Cross validation

4. **Write short note on any two: (a) Histogram (b) Scatter Plot (c) Box Plot**

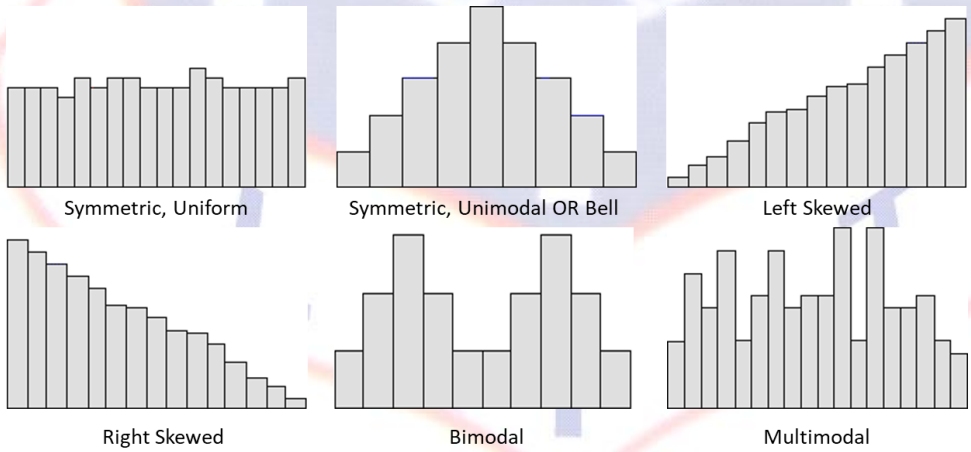
Answer:

(a) Histogram

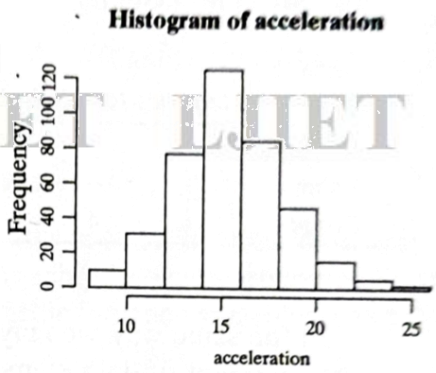
Histogram is effective for visualizing numerical attributes. Helps in understanding the distribution of numerical data into series of intervals, termed as 'bins'.

Histogram – focuses to plot ranges of data values (acting as 'bins'), elements in bin will depend upon data distribution. Hence, size of bar correspond to bin will vary.

It takes different shapes of skewness depending upon nature of data. These pattern gives quick understanding of data and act as great data exploration tool.



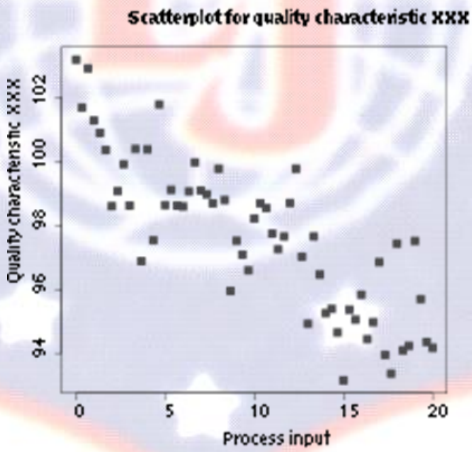
Example:



- The histogram composed of number of bars, one for each bin.
- Height of bar reflects total count of data elements whose value falls within specific bin value or frequency.
- Each bin is interval of 2 units.
- First bin reflects acceleration value of 8 to 10 units.
- Second bin reflects acceleration value of 10 to 12 units.
- Given histogram spans over acceleration value of 8 to 26 units.
- Frequency of data elements corresponding to bin keep on increasing till it reaches bin of range 14-16 units.
- After this range bar size starts decreasing till the end of whole range at acceleration of 26 units.

(b) Scatter Plot

Helps in visualizing bivariate relationship – relationship between two variables.
2D plots in which points or dots are drawn on coordinates provided by values of coordinates. The data are displayed as a collection of points, each having the value of one variable determining the position on the horizontal axis and the value of the other variable determining the position on the vertical axis.



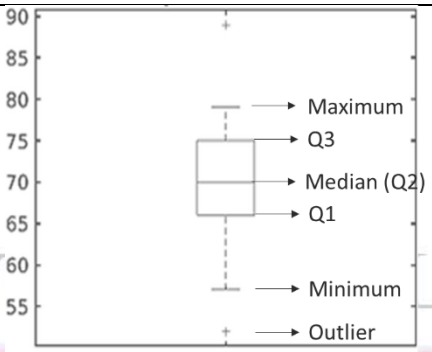
Example: in data set there are two attributes – process input and Quality. To understand relationship between two attributes, we draw each attribute on x and y axis respectively. Each dot on plot is showing value of one variable determining the position on the horizontal axis and the value of the other variable determining the position on the vertical axis.

A scatter plot can suggest various kinds of correlations between variables with a certain confidence interval. Correlations may be positive (rising), negative (falling), or null (uncorrelated). If the dots' pattern from lower left to upper right indicates a positive correlation between the variables being studied. If the pattern of dots slopes from upper left to lower right, it indicates a negative correlation.

(c) Box Plot

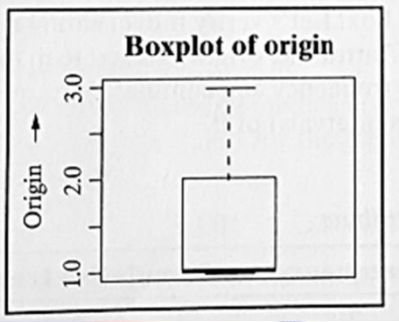
A Box plot is extremely effective mechanism to get one-shot view of data and understand nature of data.

It gives standard visualization of five-number summary statistics of a data namely minimum, first quartile (Q1), second quartile (Q2), third quartile (Q3) and maximum.



- Minimum (Q0 or 0th percentile): the lowest data point excluding any outliers.
- Maximum (Q4 or 100th percentile): the largest data point excluding any outliers.
- Median (Q2 or 50th percentile): the middle value of the dataset.
- First quartile (Q1 or 25th percentile): also known as the lower quartile qn(0.25), is the median of the lower half of the dataset.
- Third quartile (Q3 or 75th percentile): also known as the upper quartile qn(0.75), is the median of the upper half of the dataset.
- Central rectangle / box span from Q1 to Q3 is inter quartile range (IQR).
- Median is given by line or band within box.
- Lower whisker extends up to 1.5 times of IQR from bottom of box.
- Upper whisker extends up to 1.5 times of IQR from top of box.
- Data values coming beyond lower and upper whisker are outliers, deserves special consideration.

Origin	Frequency	Cumulative Frequency
1	249	249
2	70	319
3	79	398



The frequency of data value 1 is extremely high. Cumulative frequency is 398.
Median will be at (398/2) 199th Observation. i.e. Q2=1
First quartile will be at (199/2) 99.5th Observation i.e. Q1=1
Third quartile will be at (398-99.5) 298.5th observation i.e. Q3=2.
No data value beyond 3, maximum=3.
No data value lower than 1, minimum=1.

5. **Explain in details about the methods to train a learning model in supervised learning.**

Answer:

Supervised Learning – model trained using labelled input data.

Training methods for Supervised Learning

- Holdout method
- K-fold Cross Validation method
- Bootstrap Sampling

Holdout method

- Test data – may not available immediately or label value of test data is not known.
- So, part of input data (used to train a model) is held back for evaluating performance.
- Generally 70-80% of input data is used for model training, remaining 20-30% is used as test data for validating performance of model.
- Nature of data in training and test bucket must be similar in nature.

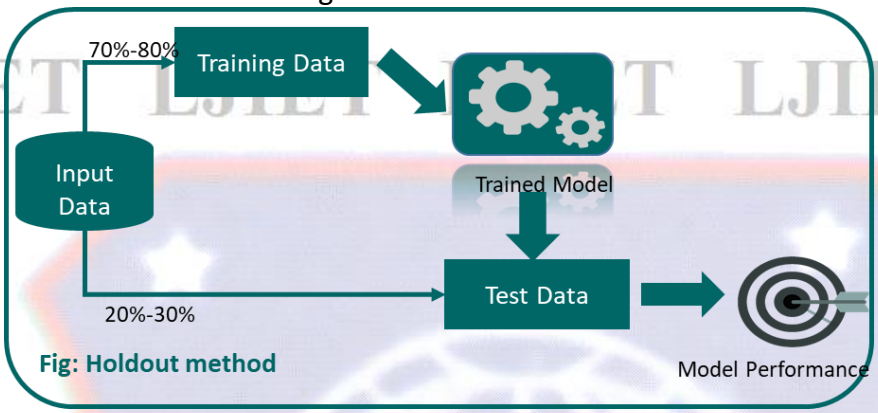


Fig: Holdout method

- After model is trained – labels of test data are predicted using model's target function.
- Performance of Model – measured by accuracy of prediction of label value.

Pros: Fully independent data; only needs to be run once so has lower computational costs.
Cons: Performance evaluation is subject to change highly/ dramatically given the smaller size of the data.
Challenge: Division of data among different classes of training and test.

K-fold Cross Validation

- Special variant of holdout – called as repeated holdout.
- K-fold validation evaluates the data across the entire training set, but it does so by dividing the training set into K folds – or subsections – (where K is a positive integer)
- Then training the model K times, each time leaving a different fold out of the training data and using it instead as a validation set.
- Later the performance metric (e.g. accuracy, ROC, etc. — choose the best one for your needs) is averaged across all K tests.
- Finally, once the best parameter combination has been found, the model is retrained on the full data.

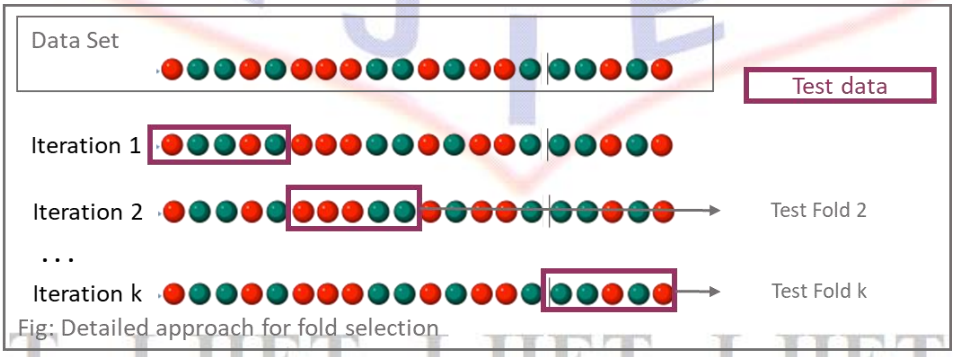


Fig: Detailed approach for fold selection

In this method, data is divided into k-completely distinct or non-overlapping random partitions called folds.

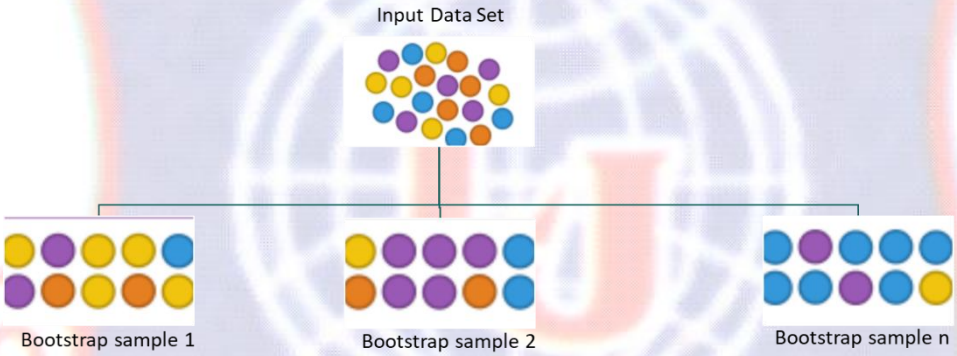
As multiple handouts have been drawn, the training and test data are more likely to represent or resemble original data.

The value of k can be set to any number. Extremely popular are:

1. 10-fold cross-validation(10-fold CV)
2. Leave-one-out-cross-validation(LOOCV)

Bootstrap Sampling

- Popular way of identifying training and test data sets from input data set.
- It uses the technique Simple Random Sampling with Replacement (SRSWR) for drawing random samples.
- Samples are constructed by drawing observations from a large data sample one at a time and returning them to the data sample after they have been chosen. This allows a given observation to be included in a given small sample more than once. This approach to sampling is called sampling with replacement.
- It randomly picks data instances from the input data set, with possibility of the same data instances to be picked multiple times.



- Input data set having n-data instances, boot strapping can create one or more training data sets having n-instances, some of the data instances are being repeated multiple times.
- Useful in case of input data sets of small size i.e. having very less number of data instances.

6. **Explain in the context of Machine Learning Model: (a) bias-variance trade-offs (b) under-fitting and over-fitting.**

Answer:

(a) Bias-variance trade off

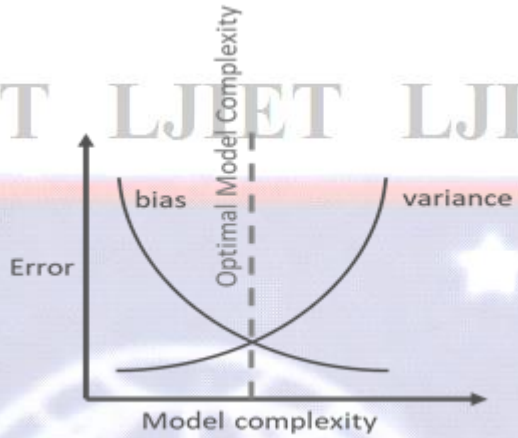
Bias measures how far off in general these models' predictions are from the correct value.

- Arises from simplifying assumption made by the model to make target function less complex and easy to learn.
- Parametric models generally have higher bias making them easier to understand and faster to learn.
- These algorithms have poor performance on data sets, which are complex in nature.
- Under fitting results in high bias.

Variance is how much the predictions for a given point vary between different realizations of the model.

- Arises from difference in training data sets used to train model.

- Different randomly sampled training data sets are used to train the model, ideally this difference should not be significant.
- In case of over fitting, since model closely attached to training data, even a small difference in training data set gets magnified in model.

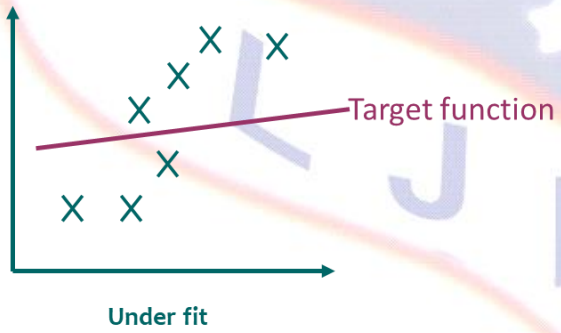


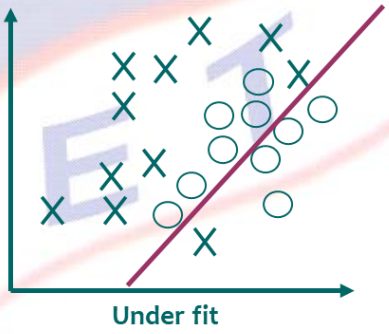
Also it is quite understandable that increasing bias will decrease the variance and increasing variance will decrease the bias. Best solution: model with low bias and low variance.

(b) Under-fitting and over-fitting

Under fitting refers to a model that can neither model the training data nor generalize to new data.

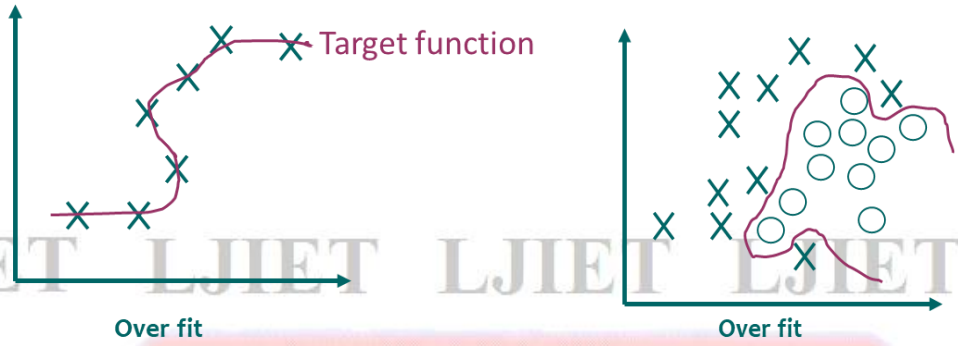
- It is resulted when target function is kept too simple – may not able to capture essential nuances and represent underlying data well.
- It may also resulted from unavailability of sufficient training data.
- Example – Trying to represent non-linear data with linear model.
- Result of under fitting – poor generalization of training data and poor performance with test data.



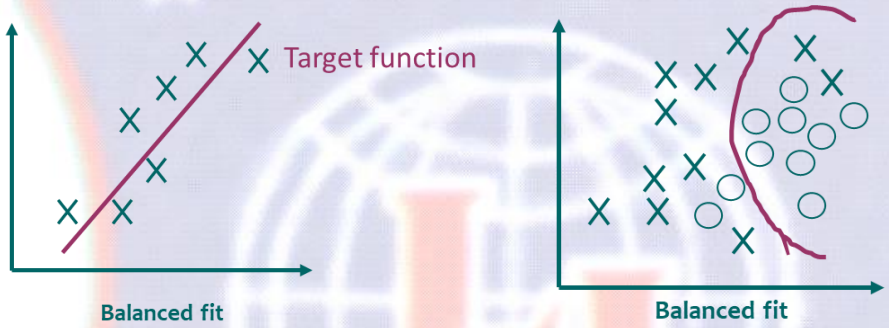


Over fitting refers to a model that fits the training data too well.

- Over fitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.
- It is resulted from trying to fit an excessively complex model to closely match the training data.
- Example – target function trying to make sure all training points are correctly placed on decision boundary.
- Result of over fitting –gives good performance with training data set but poor generalization and poor performance with test data set.



Balanced fit - Ideally, want to select a model at the sweet spot between under fitting and over fitting. This is the goal, but is very difficult to do in practice, is achieved using bias-variance trade-off.

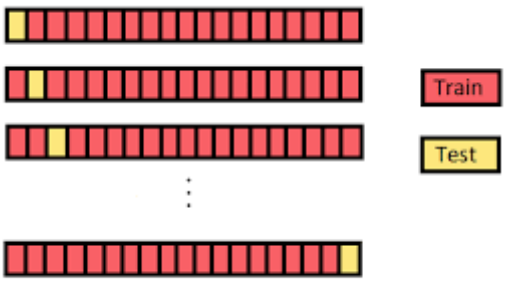


7. Write short note on: (a) LOOCV (b) F-measure (c) Silhouette width (d) ROC curve

Answer:

(a) LOOCV

- Leave-one-out-cross-validation (LOOCV)
- It is an extreme case using one record or instance at a time as a test data.
- This is done to maximize the count of data used to train the model.
- Number of iteration in this case is equal to number of data in input set.
- It is very expensive and not used much in practice.



(b) F-measure

- F-measure or F1 score – is a weighted average of the recall (sensitivity) and precision.
- Also called as Harmonic mean of precision and recall.
- F1 score might be good choice when you seek to balance between Precision and Recall.
- F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$

(c) Silhouettes Width

- Uses distance between data element as similarity measures.

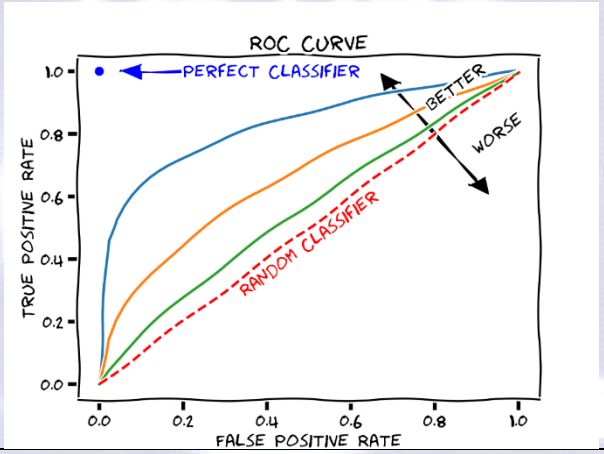
- Silhouettes width ranges from -1 to 1.
- Data set clustered into k clusters, silhouette width is calculated as:

$$Silhouettes\ Width = \frac{b(i) - a(i)}{\max\{\ a(i),\ b(i)\}}$$

- *a(i)* is the average distance between i^{th} data instance with all instance in same cluster
- *b(i)* is lowest average distance between ith data instance and data instance of all other clusters

(d) ROC curve

- Receiver Operating Characteristics (ROC) Curves – helps in visualizing the performance of classification model.
- It shows the efficiency of a model in the detection of true positive while avoiding occurrences of false positives.
- The curves of different models can be compared directly in general or for different thresholds.



8. **What is a feature? What is feature engineering? What are the major elements of Feature Engineering? Explain them.**

Answer:

Feature

2. It is an attribute of data set that is used in ML process.
3. Those attributes that are meaningful to ML problem are called feature.
4. Also called dimension of data set. *N-dimensional* data set is having *'n'* features.

Feature Engineering - It is a process of translating a data set into features such that

5. These features are able to represent data set more effectively and result in a better learning performance.
6. It is important pre-processing step for ML. It has two elements.
    1. Feature Transformation
    2. Feature Subset Selection

Feature Transformation

Transform data (structured/ unstructured) into new set of features which can represent underlying problem w.r.t ML problem

- *Feature construction*

Discover missing information about relationship between features or creating additional features. Originally *n* dimension in data set, after construction *m* dimensions are added. Data set dimensions will be *n+m.*

- *Feature extraction*

Extracting new features by combining existing features using some functional mapping
Derive a subset of features *(F_1, F2, … , F_m)* from full feature set *(F_1, F2, … , F_n)* ,where m<n, which is most meaningful in context of specific ml problem.

9. 4 **State and explain the methods to find out the similarity or redundancy aspect of the attributes in a dataset.**

Answer:

Multiple measures of similarity or redundancy of an attribute in a dataset:

1. Correlation based measures
2. Distance-based measures
3. Other coefficient based measures

Considering F1- feature1 and F2- feature2
Correlation is a measure of linear dependency between two random variable.
Pearson's (product moment) correlation coefficient :

$$\rho = \frac{cov(F_1, F_2)}{\sqrt{var(F_1) \cdot var(F_2)}}$$

$$cov(F_1, F_2) = \sum (F_{1_i} - \overline{F_1}) \cdot (F_{2_i} - \overline{F_2})$$

$$var(F_1) = \sum (F_{1_i} - \overline{F_1})^2 \text{, where } \overline{F_1} = \frac{1}{n}\sum F_{1_i}$$

$$var(F_2) = \sum (F_{2_i} - \overline{F_2})^2 \text{, where } \overline{F_2} = \frac{1}{n}\sum F_{2_i}$$

- Ranges from -1 to +1.
- Perfect correlation is indicated by value 1.
- Value 0 indicates no relationship.
- Threshold value is adopted to decide adequate similarity.

Distance-based measures: Euclidean, Minkowski Distance, Manhattan Distance
Euclidean Distance:

$$d(F_1, F_2) = \sqrt{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^2}$$

Minkowski Distance:

$$d(F_1, F_2) = \sqrt[r]{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^r}$$

Minkowski distance with r=2 is equal to Euclidean distance (L2 norm)

Manhattan Distance:

$$d(F_1, F_2) = \sum_{i=1}^{n} |F_{1_i} - F_{2_i}|$$

It is also called L1 norm.

Other coefficient based measures: Jaccard distance, Simple Matching Coefficient (SMC), Cosine Similarity

Jaccard distance:

It measure of dissimilarity between two features.

$d_j = 1 - J$ where $J = \frac{n_{11}}{n_{01}+n_{10}+n_{11}}$

Simple Matching Coefficient (SMC):

It measure of similarity between two features, includes cases where both features having value 0.

$$SMC = \frac{n_{11} + n_{00}}{n_{00} + n_{01} + n_{10} + n_{11}}$$

Cosine Similarity:

One of the most popular measure in text classification. Cosine similarity measures the angle between x and y vectors. Value of cosine similarity 1 indicates angle between x and y is 0°. Means x and y are same except magnitude. Value of cosine similarity 0 indicated angle between x and y is 90°. Means they do not share similarity. In term of text data – no word or term is common between two sentences.

$$\cos(x, y) = \frac{x.y}{\|x\|.\|y\|}$$

Where $x.y = \sum_{i=1}^{n} x_i.y_i$

$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$ and $\|y\| = \sqrt{\sum_{i=1}^{n} y_i^2}$

10. Explain Naïve Bayes classifier with example and its use in practical life.

Answer:

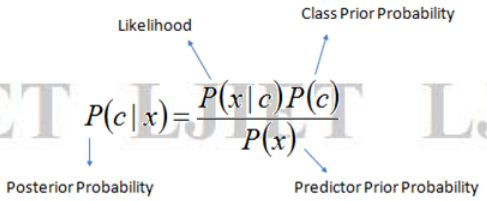
A classifier is a machine learning model that is used to discriminate different objects based on certain features.

Naïve Bayes is a conditional probability model:

given a problem instance to be classified, represented by a vector x = (x_1,x_2,…,x_n) representing some *n* features (independent variables), it assigns to this instance probabilities $P(C_k|x_1,x_2,\ldots,x_n)$ for each of *K* possible outcomes or *classes* C_k.

$$P(c \mid \mathrm{X}) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

It is based on Bayes theorem:



Say data set have 1000 fruits which could be either 'banana', 'orange' or 'other'. These are the 3 possible classes of the Y variable. We have data for X variables: Long, Sweet, Yellow.

Aggregate the training dataset look like this:

Type	Long	Not Long	Sweet	Not Sweet	Yellow	Not Yellow	**Total**
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

If given a fruit that is: Long, Sweet and Yellow, can you predict what fruit it is?

Step 1: Compute the 'Prior' probabilities for each of the class of fruits.

Out of 1000 records in training data, 500 Bananas, 300 Oranges and 200 Others. So the respective priors are 0.5, 0.3 and 0.2.

P(Banana) = 500 / 1000 = 0.5

P(Orange) = 300 / 1000 = 0.3

P(Other) = 200 / 1000 = 0.2

Step 2: Compute the probability of likelihood of evidences.

Probability of Likelihood for Banana

P(Long |Banana) = 400 / 500 = 0.8

P(Sweet |Banana) = 350 / 500 = 0.7

P(Yellow |Banana) = 450 / 500 = 0.9

The overall probability of Likelihood of evidence for Banana:

P(Long | Banana) * P(Sweet | Banana)* P(Yellow | Banana) = 0.8 * 0.7 * 0.9 = 0.504

Likewise, the overall probability of Likelihood of evidence for Orange:

P(Long | Orange) * P(Sweet | Orange)* P(Yellow | Orange) = 0

Likewise, the overall probability of Likelihood of evidence for Other:

P(Long | Other) * P(Sweet | Other)* P(Yellow | Other) = 0.09375

Step 3: Substitute evidence and prior values into the Naive Bayes formula, to get the probability that it is a banana or orange or other.

P(Banana | Long, Sweet, Yellow) = P(Long | Banana) * P(Sweet | Banana)* P(Yellow | Banana)* P(Banana)

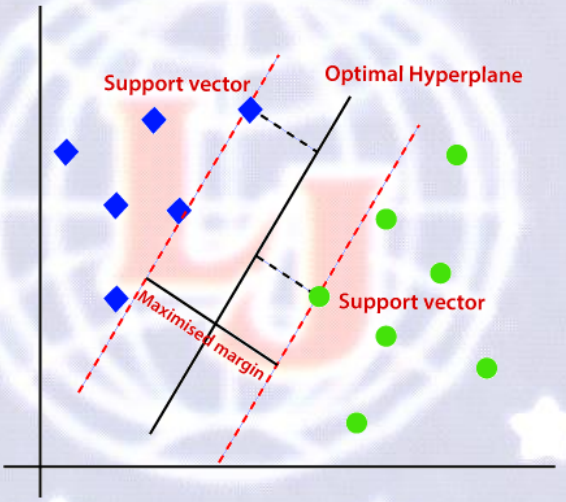
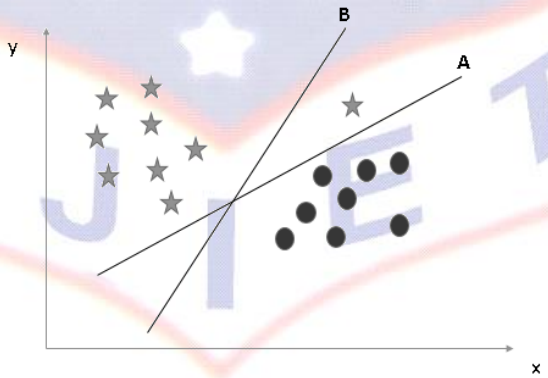
P(Banana | Long, Sweet, Yellow) = 0.504 * 0.5 = 0.252

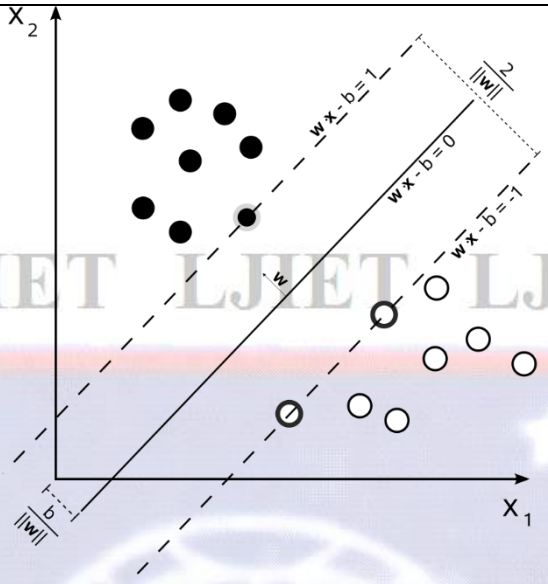
P(Orange | Long, Sweet, Yellow) = P(Long | Orange) * P(Sweet | Orange)* P(Yellow | Orange)* P(Orange)

P(Orange | Long, Sweet, Yellow) = 0 * 0.3 = 0

P(Other | Long, Sweet, Yellow) = P(Long | Other) * P(Sweet | Other)* P(Yellow | Other)* P(Other)

P(Other | Long, Sweet, Yellow) = 0.09375 * 0.2 = 0.01875

	Clearly, Banana gets the highest probability, so that will be our predicted class.
11.	**Explain in brief, the SVM model.** **Answer:** Support Vector Machine – ▪ Used for classification as well as regression. ▪ Based on concept of surface – Hyperplane, draws boundary between data instances plotted in multi-dimensional feature space. ▪ Output prediction is one of two conceivable classes which are already defined in training data. ▪ Goal of SVM is to find a plane – Hyperplane – which separates the instance on the basis of their classes  Some methods find a separating hyperplane, but not the optimal one  ▪ Some methods find a separating hyperplane, but not the optimal one ▪ Support Vector Machine (SVM) finds an optimal* solution. ▪ Maximizes the distance between the hyperplane and the “difficult points” close to decision boundary



- **w**: decision hyperplane normal vector
- $\mathbf{x}_i$: data point *i*
- y_i: class of data point *i* (+1 or -1) NB: Not 1/0
- Classifier is: $f(\mathbf{x}_i) = \text{sign}(\mathbf{w}^T\mathbf{x}_i + b)$
- Examples closest to the hyperplane are support vectors.
- Margin ρ of the separator is the width of separation between support vectors of classes.
- Assume that all data is at least distance 1 from the hyperplane, then the following two constraints follow for a training set $\{(\mathbf{x}_i, y_i)\}$

$$\vec{\boldsymbol{W}}\mathbf{x}_i + b \geq 1 \quad \text{if } y_i = 1$$

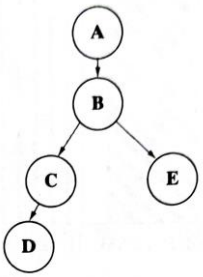
$$\vec{\boldsymbol{W}}\mathbf{x}_i + b \leq -1 \quad \text{if } y_i = -1$$

- For support vectors, the inequality becomes an equality
- The margin must be maximized. That is minimize $\frac{1}{2}\vec{w}$

12. What are Bayesian Belief Networks? Where are they used? Explain with example.

Answer:

Bayesian Belief Networks Considers/assumes within the set of attributes, probability distribution can have conditional probability relationship as well as conditional independence assumptions.



For a given figure,

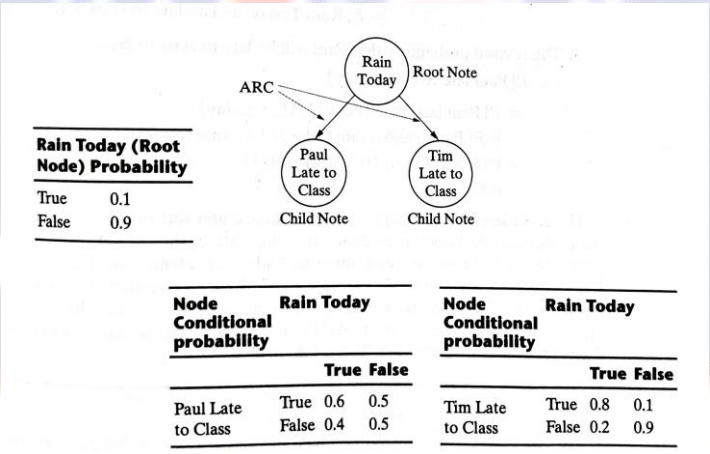
$p(A,B,C,D,E) = p(A|B)\cdot p(B|C,E)\cdot p(C|D)\cdot p(D)\cdot p(E)$ ***Chain Rule***

- A Bayesian network is a probabilistic graphical model which represents a set of variables and their conditional dependencies using a directed acyclic graph.
- Real world applications are probabilistic in nature, and to represent the relationship between multiple events, we need a Bayesian network.

Common use:

- To find out updated knowledge about the state of subset of variables, while state of other subset (evidence variable) is observed.
- Widely used to choose the values for a subset of variables in order to minimize some expected loss function or decision errors.
- Used for modelling beliefs in domains like biology and bioinformatics such as protein structure and gene regulatory network, medicines, forensics, document classification, information retrieval, image processing, decision support system, sport betting and gaming, property market analysis.
- It can also be used in various tasks including prediction, anomaly detection, diagnostics, automated insight, reasoning, time series prediction, and decision making under uncertainty.

Example: For a given network let us find that If Tim is late for a class, What is probability Paul will also get late to class?



Rain Today (Root Node) Probability	
True	0.1
False	0.9

Node Conditional probability	Rain Today		
		True	False
Paul Late to Class	True	0.6	0.5
	False	0.4	0.5

Node Conditional probability	Rain Today		
		True	False
Tim Late to Class	True	0.8	0.1
	False	0.2	0.9

Step 1: Find unconditional probability Tim is late to class.

$$p(Tim\ late\ to\ class) = p(Tim\ late|Rain) \cdot p(Rain) + p(Tim\ late|\overline{Rain}) \cdot p(\overline{Rain})$$
$$= (0.8 * 0.1) + (0.1 * 0.9)$$
$$= 0.17$$

Step 2: Find (reverse) probability of rain given Tim is late to class.

$$p(Rain|Tim\ late) = \frac{p(Tim\ late\ |Rain) \cdot p(Rain)}{p(Tim\ late)}$$
$$= \frac{0.8 * 0.1}{0.17}$$
$$= 0.47$$

Step 3: Find (revised) probability of Paul is late to class using probability calculated in step 2

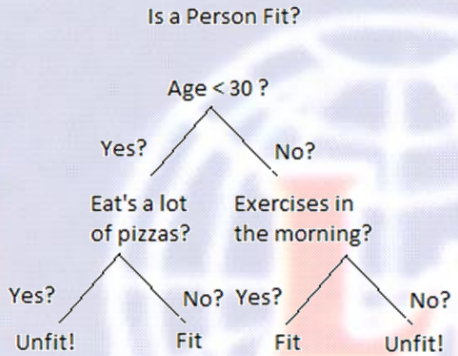
$$p(Paul\ late\ to\ class) = p(Paul\ late|Rain) \cdot p(Rain) + p(Paul\ late|\overline{Rain}) \cdot p(\overline{Rain})$$
$$= (0.6 * 0.47) + (0.5 * (1 - 0.47))$$
$$= 0.55$$

If Tim is late for a class, 55% probability that Paul will also get late to class.

13. Explain in brief: Decision tree. How to avoid over fitting decision tree?

Answer:

- One of widely adopted algorithm for classification, Builds model in a form of tree structure
- Used for multi-dimensional analysis with multiple classes
- Fast execution time and ease in the interpretation of rules
- Builds a model which predicts a value of output variable based on input variable in the feature vector
- Every node in DT corresponds to one feature vector
- Each **internal node** tests an attribute
- Each **branch corresponds** to an attribute value(true or false)
- Each leaf node assigns classification
- First node is called **root node**. Other nodes are **branch nodes**. Leaf nodes are **classification node**



Decision tree corresponding training data – follows approach **– recursive partitioning:**
It splits data into multiple subset on basis of feature values.

- Starts with root node – entire data set.
- Select feature which predicts target class in strongest way.
- Split data into multiple partitions.
- Data in each partition having distinct value for a feature which is selected for partitioning.
- Likewise algorithm continues splitting nodes on basis of feature which helps in partitioning.
- Continue this process till stopping criteria is reached.

Stopping Criteria:

1. All or most of the example at particular node have same class
2. All features have been used up in the partitioning
3. Tree has grown to pre-defined threshold limit

$Entropy(S) = \sum_{i=1}^{c} -p_i \cdot log_2 p_i$ $Information\ Gain(S,A) = Entropy(S\_bs) - Entropy(S\_as)$

$$Entropy(S_{as}) = \sum_{i=1}^{n} w_i \cdot Entropy(p_i)$$

Avoid over fitting in Decision Tree:

- Unless stopping criterion applied – decision tree keep growing indefinitely.
- Splitting over every feature and dividing into smaller portion till data is perfectly classified – results in **over fitting**.
- To avoid over fitting – pruning is essential.
- **Pruning** – reduces the size of the tree such that model is **more generalized** and **can classify** unknown labeled data in **better way**.

Approach of pruning:

Pre-pruning

- Stop growing the tree before it reaches perfection
- Tree stopped from further growing once it reaches a certain number of decision nodes
- This strategy avoids over fitting as well as optimize computational cost
- Chances of ignoring important information contributed by feature which was skipped
- Resulting in missing out certain pattern of data

Post-pruning

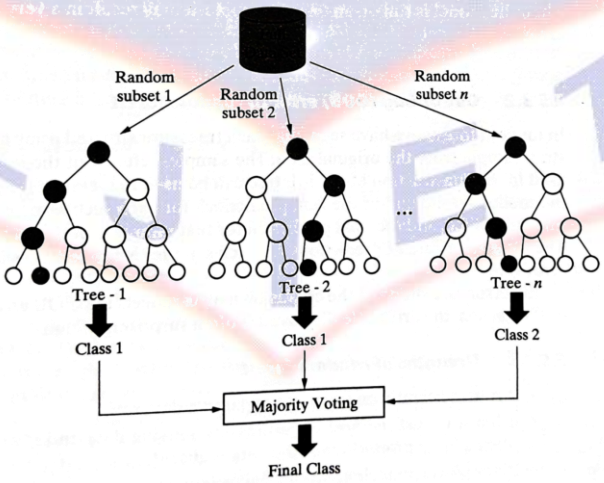
- Allow tree to grow entirely and then post-prune some branches from it
- Tree allowed to grow to full extent.
- Using certain criterion (e.g. error rates at the nodes) the size of the tree is reduced
- More effective approach in terms of classification accuracy
- Considers all minute information available from training data
- Computational cost is more

14. **Explain in brief: Random forest model. Discuss OOB error and variable.**

Answer:

Random forest model

- It is an ensemble classifier
- Combining classifier that uses and combines many decision tree classifier
- Ensemble usually done – using concept of bagging with different feature sets
- Large number of trees in random forest – to train trees enough such that contribution from each feature comes in number of models
- After generating random forest – by combining the trees – majority vote is applied



How it works:

1. If there are N variables or features in input data set, select subset of 'm' (m<N) features at random. Also the observations or data instances should be picked randomly.
2. Use best split principle on these 'm' features to calculate number of nodes 'd'.
3. Keep splitting the nodes to child nodes till the tree grown to maximum possible extent.
4. Select a different subset of training data 'with replacement' to train another decision tree following steps 1 to 3. Repeat this to build and train 'n' decision tree.
5. Final class assignment is done on the basis of the majority votes from the 'n-trees'.

Out-of-bag in random forest:

- Uses bootstrap sampling to construct each tree in forest.
- Sample left out of the bootstrap (not used in the construction) of i^{th} tree can be used to measure the performance of the model.
- Prediction made for each such sample evaluated are tallied.
- Final prediction for sample is obtained by taking vote.
- The total error rate of predictions such samples is termed as out-of-bag (OOB) error rate.

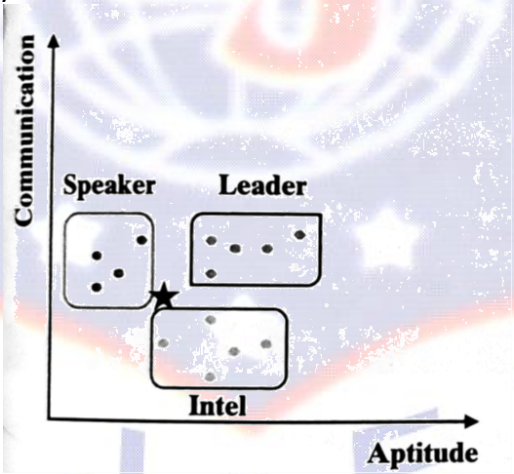
15. **Explain in brief: kNN. How to choose value of k in kNN, Discuss.**

Answer:

k-Nearest Neighbor Label of unknown element is assigned on the basis of class label of the similar training data set elements.

- Simple but extremely powerful classification algorithm.
- Underlying philosophy of kNN – people having similar background or mindset tend to stay close to each other. Neighbor in locality have similar background.
- Unknown and unlabeled data which comes for prediction problem is judged on the basis of training data set elements which are similar to the unknown element.

If class value predicted label for test data element matches with actual class value, classification model possess good accuracy.



Challenges of kNN algorithm:

1. *What is the basis of similarity or when can we say that two data elements are similar?*
   Many measure of similarity, common approach adopted by kNN is Euclidean distance.
   $$Euclidean\ Distance = \sqrt{(f11 - f12)^2 + (f21 - f22)^2}$$
   Test data – Josh represented by asterisk in 2d space. Find closest or nearest neighbor of test point.
2. *How many similar elements should be consider for deciding the class of each test data element?*
   The value of – k, user defined parameter given as input to the algorithm. Number of neighbours that need to be considered.

Deciding value of k:

1. *If the value of k is very large*

(Extreme case – equal to the total number of records in the training data) the class label of the majority class of training data set will be assigned to the test data regardless of the class label of the neighbor nearest to the test data.

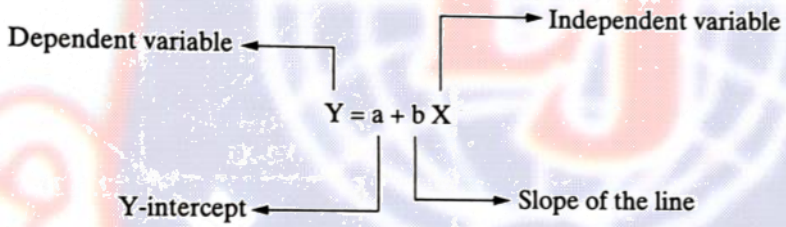
2. *If the value of k is very small*
   (Extreme case – equal to 1) the class label of a noisy data or outlier in the training data set which is the nearest neighbor to the test data will be assigned to the test data.

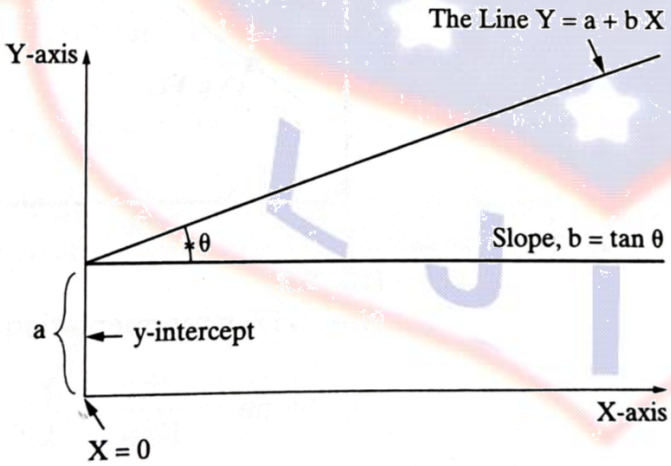
16. What is simple linear regression? Explain OLS algorithm.

Answer:

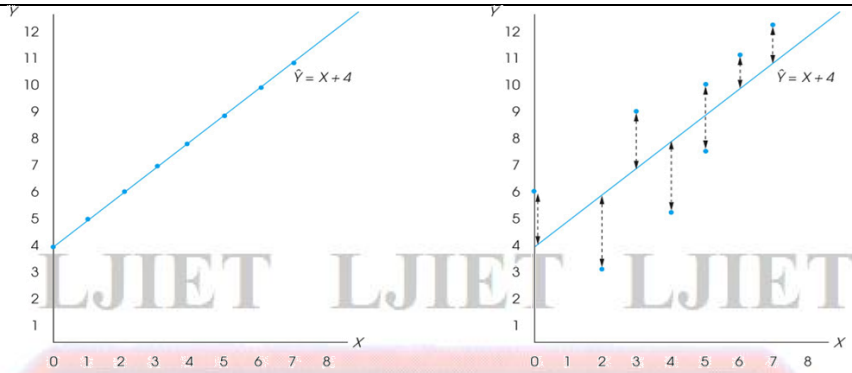
Simple Linear Regression is a statistical procedure that determines the equation for the straight line that best fits a specific set of data.

- Any straight line can be represented by an equation of the form Y = bX + a, where b and a are constants.
- The value of b is called the slope constant and determines the direction and degree to which the line is tilted.
- The value of a is called the Y-intercept and determines the point where the line crosses the Y-axis.





- How well a set of data points fits a straight line can be measured by calculating the distance between the data points and the line - **Residual**.
- The total error between the data points and the line is obtained by squaring each distance and then summing the squared values.
- The regression equation is designed to produce the minimum sum of squared errors.



Ordinary Least Square Algorithm

Step1: calculate mean of x and y i.e. $\bar{X}$ $\bar{Y}$

Step2: calculate errors of x and y i.e. (X-$\bar{X}$) (Y-$\bar{Y}$)

Step 3: get product i.e. (X-$\bar{X}$)*(Y-$\bar{Y}$)

Step 4: get summation of product $\sum$ (X-$\bar{X}$)*(Y-$\bar{Y}$)

Step 5: Square Difference of x i.e. (X-$\bar{X}$)^2

Step 6: Sum Square difference of x i.e. $\sum$ (X-$\bar{X}$)^2

Step 7: Divide output of step 4 by output of step 6 to calculate b

Step 8: calculate a using b $\mathbf{a = M_Y - bM_X}$

17. **Discuss error in linear regression. What is multiple linear regression? Give example.**

Answer:

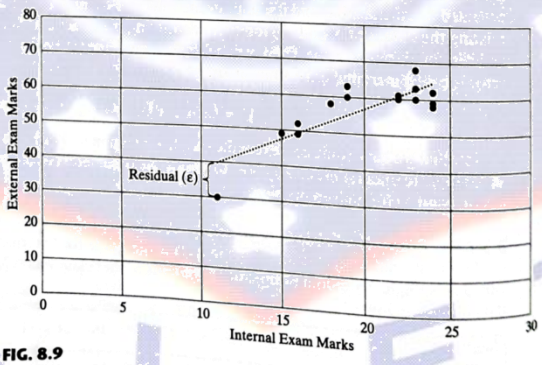


FIG. 8.9
Residual error

ŷ= a + bx + *random error*.

- How well a set of data points fits a straight line can be measured by calculating the distance between the data points and the line - *Residual.*
- The total error between the data points and the line is obtained by squaring each distance and then summing the squared values.

Mean square error

The mean square error (MSE) is squares the difference of actual value and observed value. We can see this difference in the equation below.

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \hat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

Multiple Linear Regression

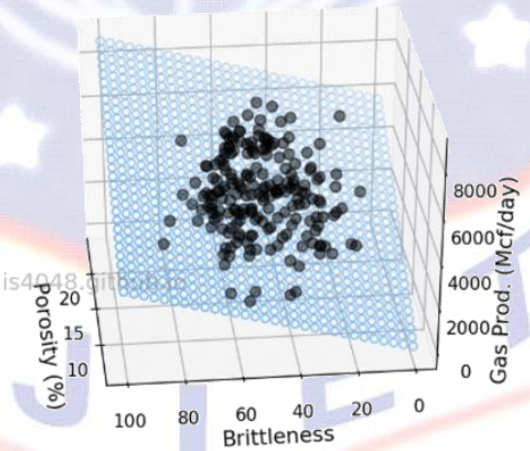
- Multiple regression is an extension of linear (OLS) regression that uses just one explanatory variable.
- Multiple linear regression refers to a statistical technique that is used to predict the outcome of a variable based on the value of two or more variables.
- Consider, suppose you have to estimate the price of a certain house you want to buy. You know the floor area, the age of the house, its distance from your workplace, the crime rate of the place, etc.
- Now, some of these factors will affect the price of the house positively. For example more the area, the more the price. On the other hand, factors like distance from the workplace, and the crime rate can influence your estimate of the house negatively.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_i X_i$$

Y : Dependent variable
β_0 : Intercept
β_i : Slope for X_i
X = Independent variable

- Here, Y is the output variable, and X terms are the corresponding input variables. Notice that this equation is just an extension of Simple Linear Regression, and each predictor has a corresponding slope coefficient (β).

- The first β term (βo) is the intercept constant and is the value of Y in absence of all predictors (i.e when all X terms are 0). It may or may or may not hold any significance in a given regression problem. It's generally there to give a relevant nudge to the line/plane of regression.



18. Differentiate between Lazy and Eager Learners.

Answer:

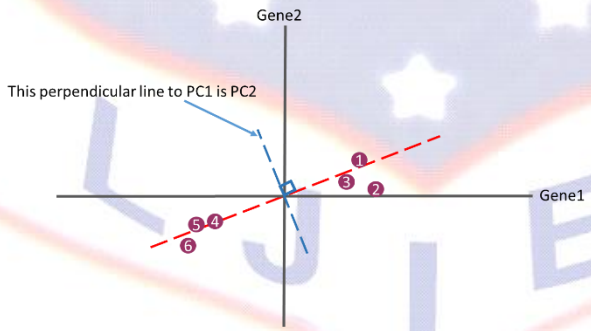
	• **Eager learning** – follows the general principles of machine learning – tries to construct generalized, input-independent target function during model training phase. • It follows typical follows steps of ML i.e. abstraction and generalization and comes up with trained model at the end of learning phase. • When test data comes in, eager learner is ready with model, doesn't need to refer back to training data. • Take more time in learning phase as compared to lazy learners • Algorithms which adopt eager learning: Decision Tree, Support Vector Machine, Neural Network.	• **Lazy learning** – Skips the general principles of machine learning – skips abstraction and generalization phase. • They do not 'learn' anything, uses training data as-it-is, also called as rote learning or memorization. • Due to dependency on input data , also called as instance learning or non-parametric learning. • Take little time in training, because not much of training actually happened • Algorithm which adopt lazy learning: k-nearest neighbor.

19. **Write short note on any two: (a)PCA (b) LDA (c) SVD**

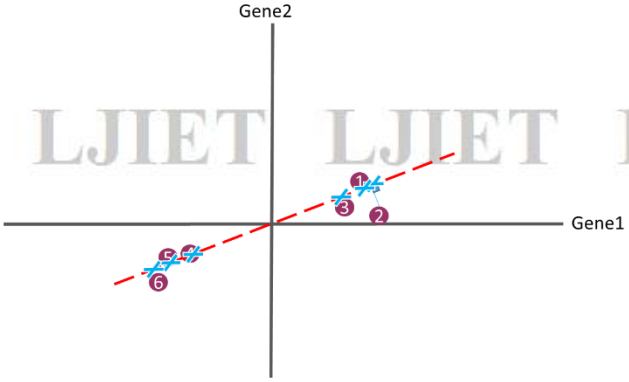
Answer:

PCA:

- Principal component analysis (PCA) is a technique for reducing the dimensionality of such datasets, increasing interpretability but at the same time minimizing information loss. It does so by creating new uncorrelated variables that successively maximize variance.
- N-dimensional feature space gets transformed into k-dimensional feature space, where dimensions are independent of each other or orthogonal to each other.
- A best-fitting line is defined as one that minimizes the average squared distance from the points to the line.



- These directions constitute an orthonormal basis in which different individual dimensions of the data are linearly uncorrelated.



- Project points on best fitted lines (perpendicularly)

- **Eigen Value PC1** = $d1^2 + d2^2 + \ldots + dn^2$
- Singular Value PC1 = $\sqrt{EigenvaluePC1}$
- Distance di = distance from origin to projected data point i.

How PCA works

1. Calculate Covariance Matrix of data set.
2. Calculate Eigen values of covariance matrix.
3. Eigen having highest value represents highest variance. That is PC1.
4. Eigen having next highest value represents PC2
5. Like this identify top 'k' Eigen values to find k-principal components.

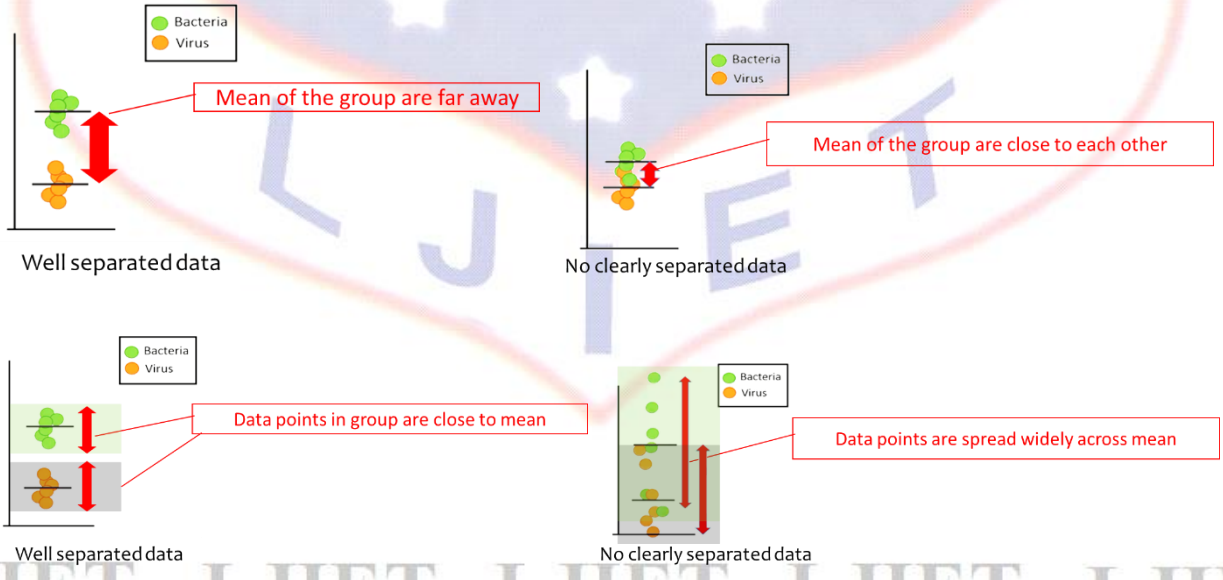
(Method Used Eigen value decomposition and covariance matrix.)

Objective PCA :

1. Features identified by PCA are distinct.
2. PC are generated in order of variability in data that it captures.
3. Sum of variance of PCs should be equal to sum of variance of original feature.

LDA

- Linear Discriminant Analysis is to find a linear combination of features that characterizes or separates two or more classes of objects or events
- Objective – transform higher dimensional (n features) data set into lower dimensional (k features) data set.
- Difference between PCA and LDA
  PCA – focusing on maximizing variance. LDA – focuses on class seperatability.
  LDA works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis.



Variance

$s_i = \sum_{x \in D_i}^{n} (x - m_i)\,(x - m_i)^T$

Intra Class scatter matrix ${s_w}^{-1}$

$$s_w = \sum_{i=1}^{c} s_i$$

Inter Class scatter matrix s_B

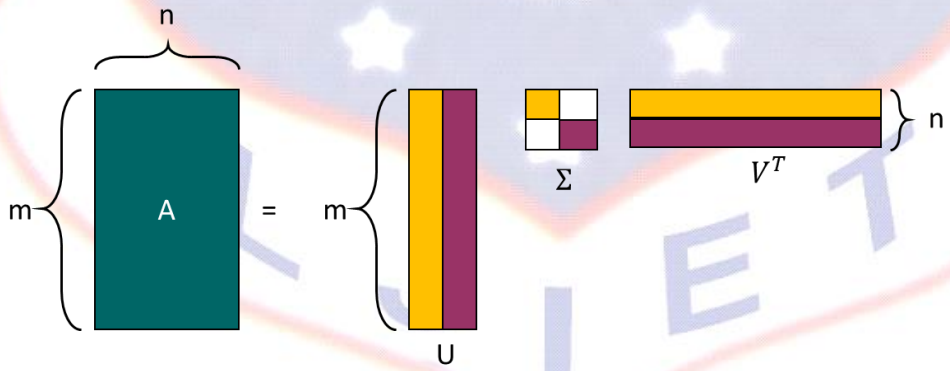
$$s_B = \sum_{i=1}^{c} N_i (m_i - m)(m_i - m)^T$$

LDA calculated eigenvalues and eigenvectors within intra-class and inter-class scatter matrices.

1. Calculate mean vectors for individual class
2. Calculate inter-class and intra-class matrices
3. Calculate eigenvalues and eigenvectors for $\boldsymbol{s_w}^{-1}$ and $\boldsymbol{s_B}$
4. Identify top *k* eigenvectors having top *k* eigenvalues.

SVD

- Singular Value Decomposition is matrix factorization technique.
- $A = U \Sigma V^T$
- A: input data matrix (m X n)
- U: left singular matrix (m X r), orthonormal ($U \cdot U^T = I$).
  User to Concept similarities matrix.
- V^T: Right singular matrix (r X n), orthonormal ($V \cdot V^T = I$).
  Movie to Concept similarities.
- Σ: Singular values, diagonal matrix (r X r).
  New Feature-Concept Matrix and Strength of each concept.
- Entries are always positive and sorted in ascending order



1. Patterns in features is captured by right singular matrix.
2. Patterns among the instances are captured by left singular matrix.
3. Larger a singular value, Larger the part of matrix A that is accountable for associated vectors.
4. New data matrix with k features is obtained using equation

$$D' = D\ X\ [v1\ v2\ \ldots vk]$$

$$D' = D\ .\ V$$

20. How apriori algorithm helps in reducing the calculation overhead for market basket analysis? Give example.

Answer:

Market Basket Analysis

- Market Basket Analysis is one of the key techniques used by large retailers to uncover associations between items.
- It works by looking for combinations of items that occur together frequently in transactions.
- It allows retailers to identify relationships between the items that people buy.

Association Rules

- Association Rules are widely used to analyse retail basket or transaction data, and are intended to identify strong rules discovered in transaction data using measures of interestingness, based on the concept of strong rules.

Build the apriori principle rules

1. If an item set is frequent, then all of its subsets must also be frequent.
   This principle significantly restricts the number of itemsets to be searched for rule generation. For example, if in a market basket analysis, it is found that an item like 'Salt' is not so frequently bought along with the breakfast items, then it is fine to remove all the itemsets containing salt for rule generation as their contribution to the Support and confidence of the rule will be insignificant.
2. If an item set is frequent, then all the supersets must be frequent too.
   These are very powerful principles which help in pruning the exponential search pace based on the support measure and is known as support-based pruning.

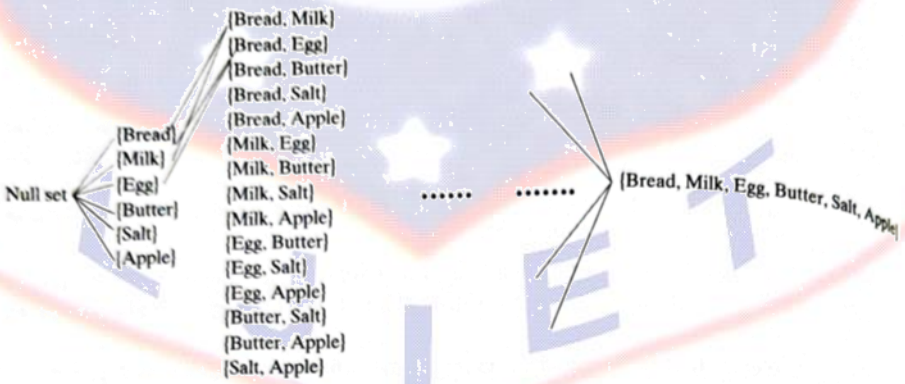


FIG. 9.15
Sixty-four ways to create itemsets from 6 items

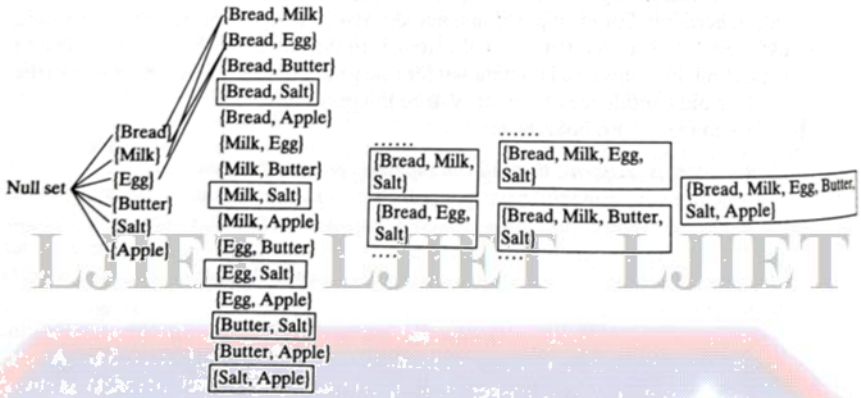


FIG. 9.16
Discarding the itemsets consisting of Salt

- Without applying any filtering logic, the brute-force approach would involve calculating the support count for each item set in Figure 9.15.
- Many of the computations may get wasted at a later point of time because some itemsets will be found to be infrequent in the transactions.
- To get an idea of the total computations to be done, the number of comparisons to be done is T * N * L, where T is the number of transactions, N is the number of candidate Item sets, and L is the maximum transaction width.
- *Apriori principle on this data set reduce the number of candidate item set (N)*
- So we can reduce computational efforts to calculate frequent item sets and make search more efficient.

21. **A database has 4 transactions, shown below.**

TID	Date	items_bought
T100	10/15/04	{K, A, D, B}
T200	10/15/04	{D, A, C, E, B}
T300	10/19/04	{C, A, B, E}
T400	10/22/04	{B, A, D}

Assuming a minimum level of support min_sup = 60% and a minimum level of confidence min_conf = 80%.

a. **Find all frequent item-sets of all lengths using the Apriori algorithm.**
b. **List all of the strong association rules, along with their support and confidence values for buys (item1, item2) ⇒ buys(item3) items can be A, B etc.**
c.

Answer:

min Support = 0.6
min Conf = 0.8

itemset	Support_Count	Support
{A}	4	4/4 = 1
{B}	4	4/4 = 1
X{C}	2	2/4 = 0.5
{D}	3	3/4 = 0.75
X{K}	2	2/4 = 0.5

n = 4

Since item set {C} and {K} have support lesser than min_sup (0.6), they are discarded.

item set	Support_Count	Support
{A,B}	4	4/4 = 1
{B,D}	3	3/4 = 0.75
{A,D}	3	3/4 = 0.75
{A,B,D}	3	3/4 = 0.75

Ans 1 Frequent item sets = {A}, {B}, {D}, {A,B}, {B,D}, {A,D}, {A,B,D}

with min_sup = 60%.

To generate association rules of type
buys (item1, item2) ⇒ item3

Consider {A,B,D}

possible rules are:

{A,B} → {D}	{A,D} → {B}	{B,D} → {A}
conf({AB→D}) = S(AB,D)/S(A,B)	C(AD→B) = S(ABD)/S(A,D)	C(BD→A) = S(ABD)/S(BD)
= 3/4	= 3/3	= 3/3
= 0.75	= 1	= 1

Since min_conf=(80%) = 0.9 Rule AB → D is discarded.

Ans 2 Strong association rules with min_conf=80% are {A,D}→{B} and {B,D}→{A}.

22. **What are three broad categories of clustering technique? Explain characteristics of each briefly.**

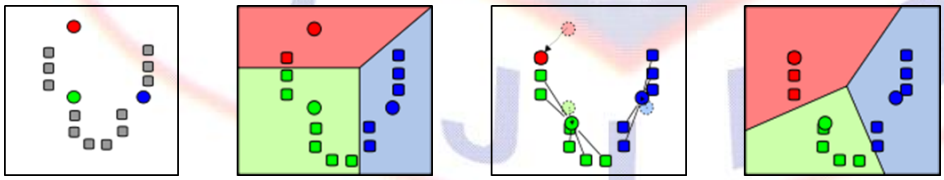
Answer:

Method	Characteristics
Partitioning methods	• Uses mean or medoid (etc.) to represent cluster centre • Adopts distance-based approach to refine clusters • Finds mutually exclusive clusters of spherical or nearly spherical shape • Effective for data sets of small to medium size
Hierarchical methods	• Creates hierarchical or tree-like structure through decomposition or merger • Uses distance between the nearest or furthest points in neighbouring clusters as a guideline for refinement • Erroneous merges or splits cannot be corrected at subsequent levels
Density-based methods	• Useful for identifying arbitrarily shaped clusters • Guiding principle of cluster creation is the identification of dense regions of objects in space which are separated by low-density regions • May filter out outliers

23. Explain k-means method with example/diagram and step-by-step algorithm.

Answer:

- Given a set of n distinct objects, the k-Means clustering algorithm partitions the objects into k number of clusters such that intra-cluster similarity is high but the inter-cluster similarity is low.
- In this algorithm, user has to specify k, the number of clusters and consider the objects are defined with numeric attributes and thus using any one of the distance metric to demarcate the clusters.



```
Algorithm K-means Clustering (Input: data set, k )
        – Initialize: Pick K random points as cluster centers
        – Alternate:
            1. Assign data points to closest cluster center
            2. Change the cluster center to the average of its
               assigned points
        – Stop when no points' assignments change
```

Working:

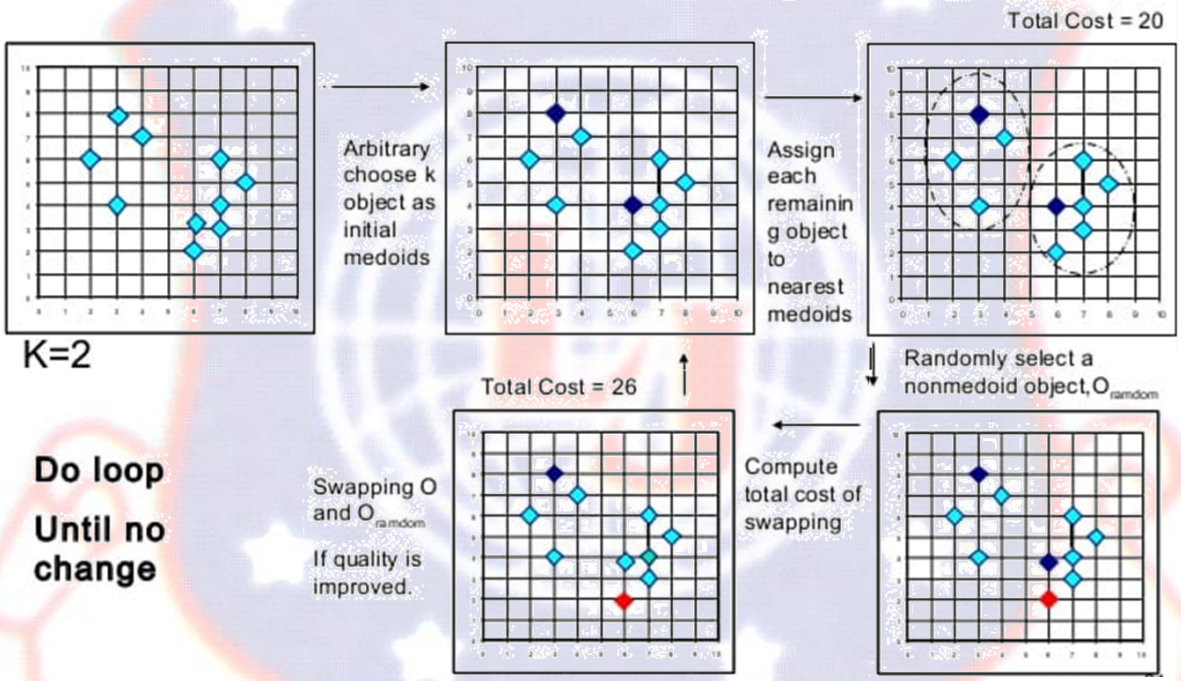
- First it selects k number of objects at random from the set of n objects. These k objects are treated as the centroids or centre of gravities of k clusters.
- For each of the remaining objects, it is assigned to one of the closest centroid. Thus, it forms a collection of objects assigned to each centroid and is called a cluster.

- Next, the centroid of each cluster is then updated (by calculating the mean values of attributes of each object).
- The assignment and update procedure is until it reaches some stopping criteria (such as, number of iteration, centroids remain unchanged or no assignment, etc.)
-

24. **Explain with neat diagram k-medoids clustering. Write PAM algorithm.**

Answer:

A medoid can be defined as the point in the cluster, whose dissimilarities with all the other points in the cluster is minimum.



Partitioning Around Medoids (PAM) : Input data set, k value

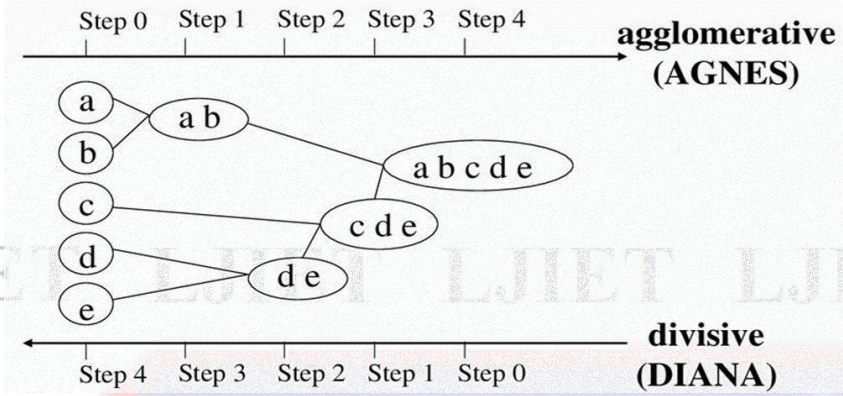
1. Randomly choose k points in the data set as initial representative points (O_j)
2. Assign each of remaining points to the cluster which has nearest representative point O_j and compute cost.
3. Randomly select a non-representative point O_r in a cluster
4. Swap O_j with O_r and compute the new Cost after swapping.
5. If $Cost_{new} < Cost_{old}$ then swap O_j with O_r to form new set of k representative objects.
6. Refine the k clusters on the basis of nearest representative point.

Continue until there is no change

25. **Write short note on: Hierarchical clustering. Discuss dendrogram.**

Answer:

- Hierarchical clustering is characterized by the development of a hierarchy or tree-like structure.
- The goal is to produce a hierarchical series of nested clusters, ranging from clusters of individual points at the bottom to an all-inclusive cluster at the top.



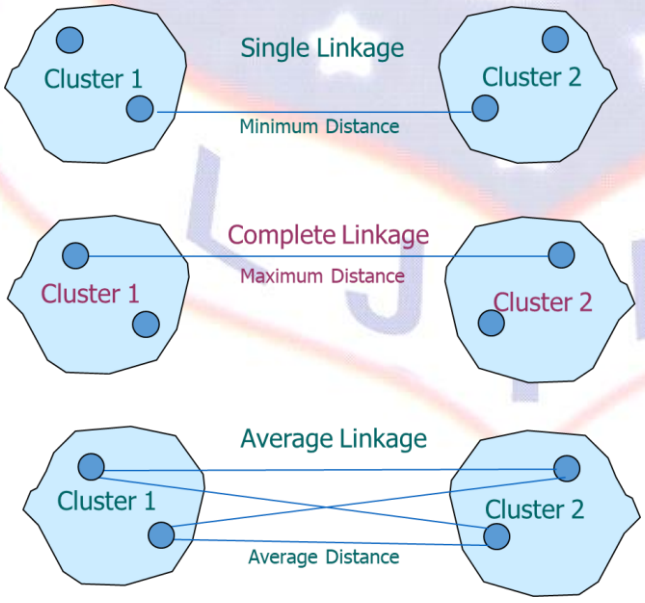
- Strategies for hierarchical clustering generally fall into two types:

  Agglomerative: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

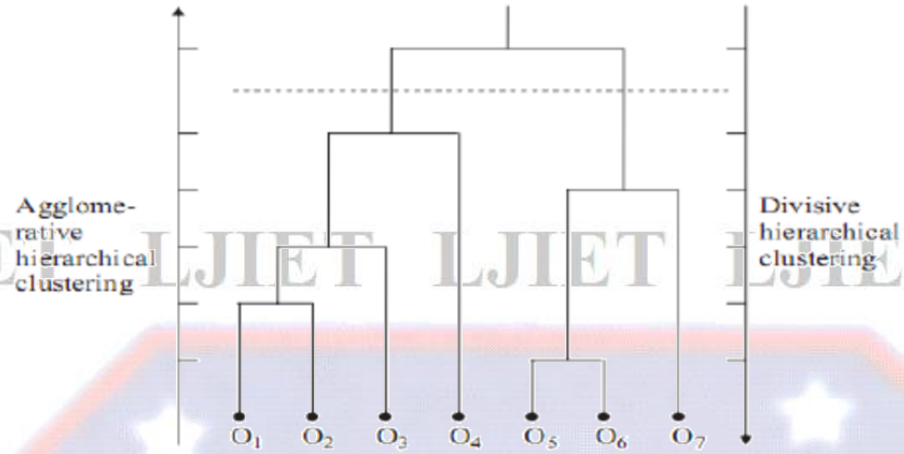
  Divisive: This is a "top-down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

- In order to decide which clusters should be combined (for agglomerative), or where a cluster should be split (for divisive), a measure of dissimilarity between sets of observations is required.

- In most methods of hierarchical clustering, this is achieved by use of an appropriate metric (a measure of distance between pairs of observations), and a linkage criterion which specifies the dissimilarity of sets as a function of the pairwise distances of observations in the sets.



- A **dendrogram** is a diagram representing a tree. In hierarchical clustering, it illustrates the arrangement of the cluster produced by the corresponding analysis.



- The individual compounds are arranged along the bottom of the dendrogram and referred to as leaf nodes.
- Compound clusters are formed by joining individual compounds or existing compound clusters with the join point referred to as a node.