Machine Learning Assignment 2

1. **What are types of SVM?**

Ans.

There are 2 types of SVM,

Linear SVM: Linear SVM is used for data that are linearly separable i.e. for a dataset that can be categorized into two categories by utilizing a single straight line. Such data points are termed as linearly separable data, and the classifier is used described as a Linear SVM classifier.

Non-linear SVM: Non-Linear SVM is used for data that are non-linearly separable data i.e. a straight line cannot be used to classify the dataset. For this, we use something known as a kernel trick that sets data points in a higher dimension where they can be separated using planes or other mathematical functions. Such data points are termed as non-linear data, and the classifier used is termed as a Non-linear SVM classifier.

2. **What is popular Algorithm for Multiclass Classifications?**

Ans. Popular algorithms that can be used for multi-class classification include:

- k-Nearest Neighbors.
  - K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
  - K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- Decision Trees.
  - A Decision tree is a flowchart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.
- Naive Bayes.
  - Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.
  - Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.
- Random Forest.
  - Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

3. **What is Spectral Clustering?**

Ans.

Spectral Clustering is a growing clustering algorithm which has performed better than many traditional clustering algorithms in many cases. It treats each data point as a graph-node and thus transforms the clustering problem into a graph-partitioning problem

Steps for spectral clustering is: -

a. Building the Similarity Graph
b. Projecting the data onto a lower Dimensional Space
c. Clustering the Data

4. Write short note on Epsilon neighborhood graph

Ans.

Epsilon-neighborhood Graph is used in spectral Clustering to build a similarity graph.
A parameter epsilon is fixed beforehand. Then, each point is connected to all the points which lie in its epsilon-radius. If all the distances between any two points are similar in scale then typically the weights of the edges i.e. the distance between the two points are not stored since they do not provide any additional information. Thus, in this case, the graph built is an undirected and unweighted graph.

5. Explain K-means and Spectral Clustering.

Ans. K means clustering

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science.
The k-means clustering
algorithm mainly performs two tasks:

Determines the best value for K center points or centroids by an iterative process.
Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

How does the K-Means Algorithm Work?
Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.
Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.

Spectral Clustering

Spectral Clustering is a growing clustering algorithm which has performed better than many traditional clustering algorithms in many cases. It treats each data point as a graph-node and thus transforms the clustering problem into a graph-partitioning problem. A typical implementation consists of three fundamental steps:-

1. Building the Similarity Graph: This step builds the Similarity Graph in the form of an adjacency matrix which is represented by A. The adjacency matrix can be built in the following manners:-

- Epsilon-neighbourhood Graph: A parameter epsilon is fixed beforehand. Then, each point is connected to all the points which lie in it's epsilon-radius. If all the distances between any two points are similar in scale then typically the weights of the edges ie the distance between the two points are not stored since they do not provide any additional information. Thus, in this case, the graph built is an undirected and unweighted graph.

- K-Nearest Neighbours A parameter k is fixed beforehand. Then, for two vertices u and v, an edge is directed from u to v only if v is among the k-nearest neighbours of u. Note that this leads to the formation of a weighted and directed graph because it is not always the case that for each u having v as one of the k-nearest neighbours, it will be the same case for v having u among its k-nearest neighbours. To make this graph undirected, one of the following approaches are followed:-
    a. Direct an edge from u to v and from v to u if either v is among the k-nearest neighbours of u OR u is among the k-nearest neighbours of v.
    b. Direct an edge from u to v and from v to u if v is among the k-nearest neighbours of u AND u is among the k-nearest neighbours of v.
- Fully-Connected Graph: To build this graph, each point is connected with an undirected edge-weighted by the distance between the two points to every other point. Since this approach is used to model the local neighbourhood relationships thus typically the Gaussian similarity metric is used to calculate the distance.

2. Projecting the data onto a lower Dimensional Space: This step is done to account for the possibility that members of the same cluster may be far away in the given dimensional space. Thus the dimensional space is reduced so that those points are closer in the reduced dimensional space and thus can be clustered together by a traditional clustering algorithm. It is done by computing the Graph Laplacian Matrix.

3. Clustering the Data: This process mainly involves clustering the reduced data by using any traditional clustering technique – typically K-Means Clustering. First, each node is assigned a row of the normalized of the Graph Laplacian Matrix. Then this data is clustered using any traditional technique. To transform the clustering result, the node identifier is retained.

Properties:

a. Assumption-Less: This clustering technique, unlike other traditional techniques do not assume the data to follow some property. Thus, this makes this technique to answer a more-generic class of clustering problems.
b. Ease of implementation and Speed: This algorithm is easier to implement than other clustering algorithms and is also very fast as it mainly consists of mathematical computations.
c. Not-Scalable: Since it involves the building of matrices and computation of eigenvalues and eigenvectors it is time-consuming for dense datasets.

6. Why dimension Reduction is very important step in machine Learning?

Ans.

Dimensionality reduction finds a lower number of variables or removes the least important variables from the model. That will reduce the model's complexity and also remove some noise in the data. In this way, dimensionality reduction helps to mitigate overfitting.

So, as you understand by now, dimensionality reduction is required for the following.

1. To reduce complexity when working with data
2. To make the learning models computationally less intensive.
3. Reduce overall training time for the algorithm.
4. Reduce noise and minimize errors.
5. Reduce storage requirements.
6. Be able to visualize and describe datasets and results more prominently.