Amrita School Of Engineering
Bengaluru, India

ML Assignment 03

Name: Raman Kumar
Reg. NO. BL.EN.P2DSC22009
Course: Machine Learning

Q.01. Define MAP function for Bayes Classification. Explain Naïve-Bayes algorithm with justification for feature independence as a necessary condition.
Ans. Maximum A Posteriori (MAP) classification is a method in Bayesian classification where the class with the highest probability is assigned to a new data point.

Naive Bayes is a popular algorithm for Bayesian classification. It makes an assumption of feature independence, meaning that the presence of a certain feature in a class is not dependent on the presence of other features. This assumption simplifies the calculation of the probability of a class given the features.
Justification for feature independence:
The Naive Bayes algorithm works well for large datasets where feature independence is a reasonable assumption. It also performs well on high-dimensional datasets where the number of features is much larger than the number of samples. Additionally, this algorithm is computationally efficient and easy to implement.

In conclusion, MAP classification is a method used in Bayesian classification to assign classes to new data points, while Naive Bayes is a simple and popular algorithm for Bayesian classification that assumes feature independence.

Q.02. What is a first order Markov-Process? Explain with an example.
Ans. A first-order Markov process is a type of Markov chain where the probability of moving from one state to another is only dependent on the current state, not any previous states.
Example: Weather prediction, where the weather tomorrow only depends on today's weather and not the weather from the past few days.

Real-time example: Predicting the next word in a sentence based on the current word, without considering previous words in the sentence.
In this example, the probability of the next word is only dependent on the current word and not on any previous words in the sentence. For instance, given the current word "I", the next word could be "like" or "am", but the probability of each of these words would only depend on the occurance of the word "I" being followed by either "like" or "am" in a training corpus of text. This kind of prediction models the conditional probability of the next word given the current word, which can be used to generate sequences of words, such as complete sentences, that are systematic and meaningful.

Q.03 Explain HMM with illustrations and example. How is this classifier different from other classifiers?

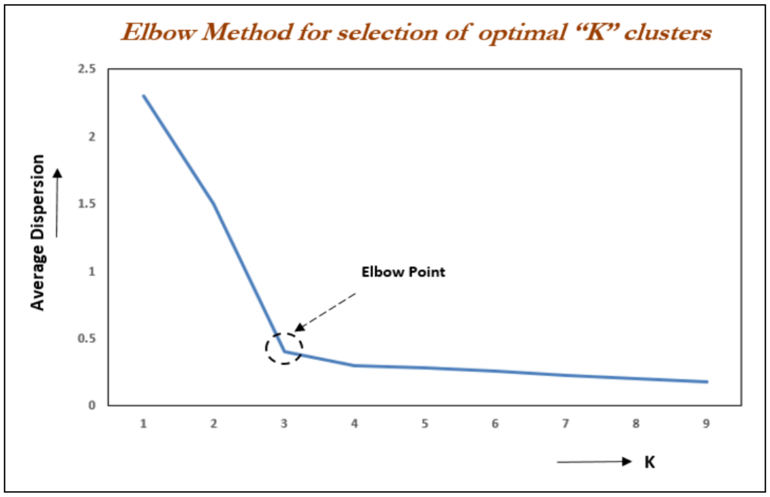
Ans. Hidden Markov Model (HMM) is a probabilistic sequence model used for modeling and prediction of sequences. It is composed of hidden states that influence observed events.
Example: In speech recognition, hidden states can be the phonemes that influence the observed speech signal.

HMM vs Other classifiers:

1. HMM considers sequence information, while other classifiers (e.g. Naive Bayes, SVM, etc.) focus on individual data points.
2. HMM models the probabilistic dependencies between events, while other classifiers model the probabilistic relationship between inputs and outputs.
3. HMM is often used for sequential data, while other classifiers are suitable for non-sequential data.
4. In conclusion, HMM is a unique classifier that is suitable for modeling sequences and considering their probabilistic dependencies.

Q.04 What is Elbow technique? How can elbow technique be used to determine the number of clusters in k-means clustering?

Ans. The elbow method is a way to determine the ideal number of clusters in k-means clustering. This method plots the cost function values for various k values. As k increases, the average distortion decreases, clusters have fewer instances, and instances are closer to their centroids. However, as k increases, the improvement in distortion decreases. The point at which this improvement starts to decline the most is called the "elbow" and is the point at which we should stop creating more clusters.



Ref: Statistics for Machine Learning by Pratap Dangeti

Q.05 Explain Single linkage and complete linkage in agglomerative clustering. Explain with justification which is better for a given data set.

Ans. Agglomerative clustering is a bottom-up clustering approach where each data point starts as a single cluster and then pairs of clusters are merged together based on similarity metric until all points are in one cluster.
This produces a cluster tree; the top is a list of all the observations, and these are then joined to form subclusters as one moves down the tree until all cases are merged in a single large cluster.
Single linkage clustering merges the two closest clusters based on the minimum distance between any two points of the two clusters. It is sensitive to noise and outliers.

Complete linkage clustering merges the two closest clusters based on the maximum distance between any two points of the two clusters. It is robust to noise and outliers

- If the data set is dense and well separated, then Single Linkage
- If the data set is sparse or noisy, then complete linkage.

Q.06 How can categorical data be clustered in agglomerative clustering?
Ans. Categorical data can be clustered in agglomerative clustering by first converting the categorical data into numerical form, such as through one-hot encoding or ordinal encoding. This transformed data can then be used as input for the agglomerative clustering algorithm, which performs hierarchical clustering based on the distance between data points. It starts with each point being a separate cluster and then merges the closest clusters based on some linkage criteria until all points are in one cluster.

Q.07 A training data set contains only 5% positive classes.What approaches can you suggest to bring class balance?

Ans: One approach to bring class balance is called oversampling. This method involves creating copies of the minority class in the training dataset to increase its representation.
Another approach is called undersampling. This method involves removing some of the majority class from the training dataset to reduce its representation.
A combination of the above two approaches called Synthetic Minority Over-sampling Technique (SMOTE) can be used to create synthetic samples of the minority class by interpolating between existing samples.
Another technique is called class weighting, where different weight is assigned to different class and these weights are used during model training to balance the class representation.
One more approach is to use different training-testing set split. It is often observed that using different split can lead to a better balance of class in the training set.

Q.08 A training dataset has attributes which are not linearly independent. How can Naïve-Bayes algorithm be employed on this dataset?

Ans: Naive-Bayes algorithm assumes that the features in the dataset are independent of each other, which is not the case when the attributes are not linearly independent.

- To use the Naive-Bayes algorithm on such a dataset, we need to find a way to make the attributes independent or at least close to independent.
- One way to do this is by using a technique called principal component analysis (PCA) which projects the data onto a new set of linearly independent variables.
- Another way is by using a technique called linear discriminant analysis (LDA) which finds a new set of variables that maximizes the separation between classes.
- These techniques can be used to transform the attributes in the dataset and make them independent, and then the Naive-Bayes algorithm can be applied on the transformed dataset.

Q.09 Explain the floating feature selection algorithm. Compare this algorithm with the forward feature selection algorithm with respect to its efficacy and computational complexity.

Ans: Floating feature selection is a technique that uses a sliding window to iteratively add and remove features from a model to find the best subset of features that give the optimal performance. It's more efficient than forward feature selection but also more computationally complex.

Comparison:

Feature	Floating Feature Selection	Forward Feature Selection
Efficacy	More efficient as it can quickly discard irrelevant features and retain the relevant ones	Less efficient as it only adds features one by one to the model Computational Complexity
Computational Complexity	More complex as it requires more iterations to find the optimal subset of features	Less complex as it only adds features one by one to the model
Flexibility	Can discard and add features at any point of time during the iterations	Only adds features one by one to the model and cannot discard any features once added
Performance	Can achieve better performance as it considers both adding and removing features	May not achieve the optimal performance as it only considers adding features

Q.10 What are the principal components? What would happen if we apply PCA to a dataset with uniform distribution inside a hypersphere in N-dimensional feature space? Explain with diagrams & justification.

Ans: Principal components are linear combinations of the original features in a dataset that capture the largest possible amount of variation in the data. They are used to reduce the dimensionality of the data in a process called principal component analysis (PCA) which is a popular technique in machine learning.

Steps:

1. Mean Centering/ Normalize data
2. Compute covariance matrix
3. Eigen-decomposition
4. Compute Variance & dimensionality reduction
5. Data transformation

If you apply PCA to a dataset with a uniform distribution inside a hypersphere in an N-dimensional feature space, the main objective of PCA, which is to identify the directions with the highest variance, will not be achieved. This is because all the points in the dataset are evenly spaced and there is no clear direction of maximum variance. Therefore, the resulting principal components will have equal magnitude and no clear direction of maximum variance.

In conclusion, PCA is not appropriate for datasets with a uniform distribution inside a hypersphere in N-dimensional feature space, as it does not effectively capture the structure of the data.