ML written material

Practical 1

Training Data:

Training data is a subset of a dataset that is used to train a machine learning model. It consists of input-output pairs, where the input represents the features or attributes, and the output represents the corresponding labels or target values. During the training phase, the machine learning model learns the patterns, relationships, and associations within the training data, adjusting its parameters or weights to minimize the difference between its predictions and the actual output. The quality and representativeness of the training data significantly impact the performance and generalization ability of the model. A well-constructed training dataset should encompass a diverse range of examples to help the model learn robust and accurate patterns.

Testing Data:

Testing data, also known as a test set, is a separate subset of the dataset that is not used during the training phase but is reserved to assess the model's performance and generalization to new, unseen data. The testing data consists of input-output pairs similar to the training data, but the model has not been exposed to these examples during training. By evaluating the model on the testing data, one can estimate how well the model is expected to perform on real-world, unseen data. The testing phase helps identify potential issues such as overfitting (the model fitting the training data too closely) or underfitting (the model not capturing the underlying patterns). A good practice is to ensure that the testing data is representative of the broader distribution of data that the model may encounter in real-world applications.

Practical 2

DESCRIPTION:

1. Training dataset table (input data):

	A	B	C	D	E	F	G
1	sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
2	Sunny	Warm	Normal	Strong	Warm	Same	Yes
3	Sunny	Warm	High	Strong	Warm	Same	Yes
4	Rainy	Cold	High	Strong	Warm	Change	No
5	Sunny	Warm	High	Strong	Cool	Change	Yes
6							

2.: Write the right hypothesis/function from historical data

One of the often-used statistical concepts in machine learning is the hypothesis.It is notably employed in supervised machine learning, where an ML model uses a dataset to train a function that most effectively translates input to related outputs.

In this code person enjoys sport if weather is sunny, airtemp is warm, wind is strong

3. How Does It Work?

It eliminates attribute that do not affect target column

Practical 3

True Positive (TP):

Represents the number of instances that were correctly predicted as positive by the model. For example, in a medical diagnosis scenario, TP would be the number of actual positive cases correctly identified by the model.

True Negative (TN):

Represents the number of instances that were correctly predicted as negative by the model. Using the medical diagnosis example, TN would be the number of actual negative cases correctly identified by the model.

False Positive (FP):

Represents the number of instances that were incorrectly predicted as positive by the model when they are actually negative. In medical diagnosis, FP would be the number of actual negative cases incorrectly identified as positive.

False Negative (FN):

Represents the number of instances that were incorrectly predicted as negative by the model when they are actually positive. In medical diagnosis, FN would be the number of actual positive cases incorrectly identified as negative.

Support Vector Machine (SVM) for Multiclass Classification:

Support Vector Machines are powerful algorithms primarily designed for binary classification. However, they can be extended to handle multiclass classification using various strategies. One common approach is the one-vs-all (OvA) or one-vs-rest (OvR) method, where the multiclass problem is decomposed into multiple binary classification problems.

In the OvA approach:

For each class, a binary classifier is trained to distinguish that class from the rest.

During prediction, the class with the highest confidence or probability among all binary classifiers is selected as the final predicted class.

Practical 4

Description:

The candidate elimination algorithm incrementally builds the version space given a hypothesis space H and a set E of examples. The examples are added one by one; each example possibly shrinks the version space by removing the hypotheses that are inconsistent with the example. The candidate elimination algorithm does this by updating the general and specific boundary for each new example.

- You can consider this as an extended form of Find-S algorithm.
- Consider both positive and negative examples.
- Actually, positive examples are used here as Find-S algorithm (Basically they are generalizing from the specification).
- While the negative example is specified from generalize form.

Terms :-

General Hypothesis: Not Specifying features to learn the machine.

G = {‘?’, ‘?’,’?’,’?’…}: Number of attributes.

Specific Hypothesis: Specifying features to learn machine (Specific feature).

S= {‘pi’,’pi’,’pi’…}: Number of pi depends on number of attributes.

Version Space: It is intermediate of general hypothesis and Specific hypothesis. It not only just written one hypothesis but a set of all possible hypothesis based on training data-set.

Candidate-elimination algorithm :-

Step1: Load Data set

Step2: Initialize General Hypothesis and Specific Hypothesis.

Step3: For each training example

Step4: If example is positive example

```
if attribute_value == hypothesis_value:
  Do nothing
else:
  replace attribute value with '?' (Basically generalizing it)
```

Step5: If example is Negative example

Make generalize hypothesis more specific.

Practical 5

The Naïve Bayes classifier is a probabilistic machine learning algorithm based on Bayes' theorem, often employed for classification tasks. Its "naïve" nature stems from the assumption of feature independence, meaning that the presence or absence of one feature is considered unrelated to the presence or absence of another feature. This simplifying assumption enables the algorithm to make predictions efficiently, even with limited training data. During the training phase, the

Naïve Bayes classifier learns the probabilities of different features given each class, calculating prior probabilities and feature likelihoods. When making predictions on new data, it calculates the posterior probability of each class given the observed features, ultimately predicting the class with the highest probability as the output. There are different variations of Naïve Bayes, including Gaussian, Multinomial, and Bernoulli, each suitable for different types of data. Applications of Naïve Bayes span across spam filtering, sentiment analysis, document categorization, and medical diagnosis, owing to its simplicity, ease of implementation, and efficiency. While its performance may be affected when the independence assumption is violated or when features are highly correlated, the Naïve Bayes classifier remains a valuable and widely used tool, particularly in scenarios with limited training data or where quick and interpretable predictions are essential.

Practical 6

The Decision Tree classifier is a popular machine learning algorithm used for both classification and regression tasks. It works by recursively partitioning the dataset into subsets based on the values of different features, creating a tree-like structure. At each node, the algorithm selects the feature that best separates the data according to certain criteria, such as Gini impurity or information gain. This process continues until a stopping criterion is met, resulting in a tree that can be used to make predictions on new, unseen data. Decision Trees are easy to interpret and visualize, making them valuable for understanding the decision-making process of the model. However, they are susceptible to overfitting, capturing noise in the training data.

Random Forest Classifier, on the other hand, is an ensemble learning method that builds multiple Decision Trees and combines their predictions. It introduces randomness by training each tree on a random subset of the features and a random subset of the training data. The final prediction is then determined by aggregating the predictions of all the individual trees, often through a majority voting mechanism. Random Forests mitigate the overfitting issue associated with individual Decision Trees and generally offer improved generalization performance. They are robust, versatile, and suitable for a variety of applications, including classification and regression tasks. Random Forests also provide insights into feature importance, aiding in the identification of the most influential features in the dataset. Overall, Random Forest Classifier extends the capabilities of Decision Trees by harnessing the strength of multiple trees and enhancing predictive accuracy.

Practical 7

Principal Component Analysis (PCA) is a dimensionality reduction technique widely utilized for feature selection and data preprocessing. Its primary purpose is to identify and preserve the most informative features within a dataset. The process begins with the standardization of features, ensuring a consistent scale across all variables. PCA offers benefits such as dimensionality reduction, noise reduction, handling collinearity, and improved computational efficiency. However, it is crucial to consider the balance between dimensionality reduction and interpretability, as principal components are combinations of original features. Additionally, PCA assumes linear relationships between features, which may impact its performance in non-linear datasets. In summary, PCA is a valuable tool for feature selection, especially in scenarios with a large number of correlated features, where it enhances computational efficiency and mitigates the impact of noise and irrelevant features.

Practical 8

Least Squares Regression Algorithm:

The Least Squares Regression Algorithm is a linear regression method used to model the relationship between a dependent variable and one or more independent variables. The algorithm aims to minimize the sum of the squared differences between the observed and predicted values. It achieves this by adjusting the model parameters, represented by coefficients, through an optimization process. The resulting model can be expressed as a linear equation, making it suitable for predicting continuous outcomes. Least Squares Regression is widely applied in various fields, including economics, finance, and engineering, due to its simplicity and interpretability. While it assumes a linear relationship between variables, it serves as a foundational algorithm for regression analysis.

Logistic Regression Algorithm:

Contrary to its name, Logistic Regression is a classification algorithm used to predict the probability of an instance belonging to a particular class. It models the relationship between the dependent binary variable and one or more independent

variables by employing the logistic function (sigmoid function). The logistic regression algorithm estimates the probabilities and classifies instances based on a chosen threshold. It is particularly useful in binary classification problems, such as spam detection or medical diagnosis. Despite the term "regression" in its name, logistic regression is fundamentally a classification algorithm, and it can be extended to handle multiple classes through techniques like one-vs-all or softmax regression. Logistic Regression provides interpretable results and is sensitive to outliers, making it a valuable tool in various domains where understanding the impact of features on the outcome is crucial.

Practical 9

Backpropagation

Artificial Neural Network (ANN) with Backpropagation is a machine learning model designed to mimic the human brain's learning process. By iteratively adjusting connection weights during training, it learns intricate patterns in data, making it capable of making predictions and generalizing from the provided datasets.

1. Text Pre-processing:

Clean and prepare the restaurant review text data by removing non-alphabetic characters, converting to lowercase, stemming, and eliminating common English stopwords, ensuring the dataset is ready for analysis.

2. Text Clustering:

Utilize a Bag of Words model with CountVectorizer to transform the pre-processed text data into numerical features, enabling the application of clustering algorithms to group similar reviews together and identify patterns within the dataset.

3. Classification with Prediction:

Train a Gaussian Naive Bayes classifier on the pre-processed and transformed data to predict sentiment labels (positive or negative) for restaurant reviews, allowing the model to learn from the training set and make predictions on unseen data.

4. Test Score:

Evaluate the performance of the Naive Bayes classifier by calculating accuracy scores, using metrics such as accuracy_score to measure the model's effectiveness in correctly predicting sentiments on the test set.

5. Confusion Matrix:

Generate a confusion matrix to provide a detailed breakdown of the model's predictions, showcasing true positive, true negative, false positive, and false negative results. This matrix offers insights into the classifier's strengths and weaknesses in sentiment classification.

Practical 10

K-Means Clustering is a popular unsupervised machine learning algorithm used for partitioning a dataset into distinct and homogeneous groups, known as clusters. The algorithm aims to minimize the sum of squared distances between data points and the centroid of their assigned cluster. The "K" in K-Means refers to the predetermined number of clusters that the algorithm seeks to identify in the data. The process begins by randomly assigning K centroids, and then iteratively, data points are assigned to the nearest centroid, and the centroids are recalculated based on the mean of the assigned points. This iterative assignment and recalculation continue until convergence, where centroids stabilize, and the algorithm has effectively grouped similar data points together.

K-Means has a broad range of applications, including customer segmentation, image compression, and anomaly detection. However, it has certain limitations, such as sensitivity to initial centroid placement and the assumption that clusters are spherical and equally sized. Variations like K-Means++ for better initialization and the ability to handle non-spherical clusters, and K-Means clustering with hierarchical structures, have been introduced to address some of these limitations. Despite its drawbacks, K-Means remains widely used for its simplicity, scalability, and efficiency in clustering large datasets, providing insights into underlying structures within the data.