Practical-1

Machine learning basics:

In this lab, we will go through the basics of machine learning. The student needs to make a soft copy note on the following topics:

Topics:

1. **What is Machine learning?**

- Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

2. **Steps in collection of data**

- Identify issues and opportunities for collecting data: Every tool for collecting data has its own pros and cons. Thus, for deciding the best method, it is important to identify issues and opportunities for collecting data according to the method. It might be helpful to engage in a pilot study to review our tools and sample size.
- Setting goals and objectives: The researcher uses data to address his/her research questions and must design his/her methodology accordingly. Thus, every tool used by the researcher must have certain objectives which could be used for addressing these questions after analysis.
- Planning approach and methods: Researcher would make decisions pertaining to who will be surveyed, how data will be collected, sources and tools for data collection, and duration of the project.
- Collect data: While planning the data collection, it is important to understand logistical challenges and prepare accordingly.

3. **Steps in importing the data in python (Through: csv, json, and other data formats)**

- Importing data into Python typically involves using libraries that support various data formats. Here are the general steps to import data into Python:

   1. Install necessary libraries: If you haven't already installed the required libraries for data import, you'll need to do so. Some common libraries for data import are pandas, NumPy, csv, xlrd, openpyxl, etc.

2. Import the required libraries: In your Python script or Jupyter Notebook, import the necessary libraries using the import statement. For example:

4. Preprocessing

a) Remove Outliers

- An Outlier is a data-item/object that deviates significantly from the rest of the (so-called normal)objects. They can be caused by measurement or execution errors. The analysis for outlier detection is referred to as outlier mining. There are many ways to detect the outliers, and the removal process is the data frame same as removing a data item from the panda's data frame.
  Here pandas data frame is used for a more realistic approach as in real-world projects need to detect the outliers arouse during the data analysis step, the same approach can be used on lists and series-type objects.

b) Normalize Datasets, Data encoding:-

- Normalizing datasets is a common preprocessing step in machine learning and data analysis. Normalization is done to scale numerical features in a dataset to a standard range, usually between 0 and 1 or -1 and 1, without distorting the original distribution of the data. This process is important when dealing with features that have different units or scales, as it helps algorithms converge faster and improves the overall performance of the model.

There are different methods to normalize datasets, and the choice of method depends on the nature of the data and the requirements of the model. Here are some commonly used normalization techniques:

1. Min-Max Normalization: This method scales the data to a specific range, typically between 0 and 1. The formula to perform min-max normalization is:
   X_normalized = (X - X_min) / (X_max - X_min)

2. Decimal Scaling: In this method, you scale the data by dividing it by a factor of 10 raised to an appropriate power, so that the decimal point is shifted to a desired position.

3. Log Transformation: This method applies the logarithm function to the data, which can help to compress a wide range of values, especially for skewed distributions.

4. Softmax Transformation: Primarily used in multi-class classification problems, softmax normalization converts the output of a model into probability distributions. It ensures that the probabilities of all classes sum up to 1.

c) Handling Missing Data

- Here are some common techniques for handling missing data:

1. Deletion:

a. Listwise Deletion: Also known as complete case analysis, this method involves removing entire rows with missing values. It's a straightforward approach but can lead to a loss of valuable information, especially if missing values are not randomly distributed.
b. Pairwise Deletion: In this method, missing values are ignored for specific analyses, and calculations are performed using available data. It can be useful when dealing with different missing patterns across variables.

2. Imputation: Imputation involves filling in missing values with estimated or predicted values. The goal is to retain as much information as possible while avoiding bias in the data.
   a) Mean/Median/Mode Imputation: In this approach, missing values are replaced with the mean, median, or mode of the available data for that feature. It's a simple method but can lead to an underestimation of variance.
   b) Regression Imputation: Missing values are predicted using a regression model based on other available features. This method captures relationships between variables and can be more accurate than simple mean imputation.
   c) K-Nearest Neighbors (KNN) Imputation: Missing values are imputed by taking the average of the K-nearest data points in the feature space. It considers the similarity between samples, making it effective for complex data distributions.

5. Machine Models

a) Types of machine learning models – Supervised learning, Unsupervised learning, reinforcement learning.

1. Supervised Learning: Supervised learning is a type of machine learning where the model is trained on a labeled dataset, meaning that the input data is associated with corresponding target labels or outputs. The goal of supervised learning is to learn a mapping from input features to the desired output labels, enabling the model to make predictions on new, unseen data.

   The training process involves presenting the model with input-output pairs and adjusting its parameters to minimize the prediction error. Common algorithms used in supervised learning include linear regression, logistic regression, support vector machines (SVM), decision trees, random forests, and neural networks.

2. Unsupervised Learning: Unsupervised learning involves training the model on an unlabeled dataset, where the algorithm tries to find patterns or structure within the data without explicit guidance. In other words, the model discovers the underlying relationships and groupings in the data without knowing the correct output labels.

   Common algorithms used in unsupervised learning include K-means clustering, hierarchical clustering, Gaussian mixture models, principal component analysis (PCA), and autoencoders.

3. Reinforcement Learning: Reinforcement learning is a type of machine learning where an agent learns to make decisions by interacting with an environment and receiving feedback in the form of rewards or penalties. The agent's goal is to learn the optimal strategy to maximize cumulative rewards over time.

   The agent takes actions in the environment, observes the state transitions, receives rewards, and updates its policy to improve future decision-making. Reinforcement learning is often used in scenarios where there is no labeled dataset, and the agent must learn by trial and error.

b) Parameters of machine learning model (Learning rate, regularization, etc.)

1. Learning Rate: The learning rate is a hyperparameter used in optimization algorithms, particularly in gradient-based optimization methods like gradient descent and its variants (e.g., stochastic gradient descent - SGD). It determines the step size at which the model's parameters are updated during training. The learning rate controls how quickly or slowly the model learns from the data.If the learning rate is too large, the model might overshoot the optimal values of the parameters and diverge, leading to unstable learning. On the other hand, if the learning rate is too small, the training process might be too slow and take a long time to converge.

2. Regularization: Regularization is a technique used to prevent overfitting in machine learning models. Overfitting occurs when a model becomes too complex and captures noise or random fluctuations in the training data, leading to poor generalization on new, unseen data.

   Regularization methods add penalty terms to the loss function during training, discouraging the model from learning overly complex patterns. This encourages the model to prioritize simpler and more generalizable solutions. Two common types of regularization techniques are L1 regularization (Lasso) and L2 regularization (Ridge)

6. Test-train data split: using constant ration, k-fold cross validation

1. Test-Train Data Split: In the test-train data split method, the original dataset is randomly divided into two disjoint subsets: a training set and a test set. The training set is used to train the machine learning model, while the test set is used to evaluate its performance.The typical ratio for the test-train data split is 80-20 or 70-30, where the training set contains the majority of the data (e.g., 70% or 80%), and the test set contains the remaining portion (e.g., 30% or 20%).

2. k-Fold Cross-Validation:

   k-Fold Cross-Validation is a more robust method for evaluating the model's performance by partitioning the dataset into k subsets (folds) of approximately equal size. The model is trained and tested k times, each time using a different fold as the test set and the remaining k-1 folds as the training set.

7. Output Inference

- Output inference, also known as inference or prediction, is the process of making predictions or drawing conclusions based on the trained machine learning model. After the model has been trained on a labeled dataset and has learned the underlying patterns and relationships in the data, it can be used to make predictions on new, unseen data.The output inference process involves passing the new input data through the trained model, and the model then produces predictions or classifications as output.

8. Validation: different metrics – Confusion Matrix, Precision, Recall, F1-score

Confusion Matricx: The confusion matrix is a table that helps visualize the performance of a classification model by summarizing the results of predictions against actual class labels. It contains four values:

The confusion matrix is usually represented as follows:

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Precision: Precision is a metric that measures the accuracy of positive predictions made by the model. It is defined as the ratio of true positive predictions to the total number of positive predictions (true positive and false positive). It shows how many of the positive predictions were actually correct.

Precision = TP / (TP + FP)

Recall: Recall is a metric that measures the ability of the model to correctly identify positive instances. It is defined as the ratio of true positive predictions to the total number of actual positive samples (true positive and false negative). It shows how many of the actual positive samples were correctly identified.

Recall = TP / (TP + FN)

F1-score: The F1-score is a single metric that combines precision and recall into a harmonic mean. It provides a balance between precision and recall, as sometimes, increasing one of these metrics can lead to a decrease in the other. The F1-score is useful when you want to find an optimal trade-off between precision and recall.

F1-score = 2 * (Precision * Recall) / (Precision + Recall)