Sri Sivasubramaniya Nadar College of Engineering, Chennai
(An autonomous Institution affiliated to Anna University)

Degree & Branch	B.E. Computer Science & Engineering	Semester	V
Subject Code & Name	ICS1512 & Machine Learning Algorithms Laboratory		
Academic year	2025-2026 (Odd)	Batch:2023-2028	**Due date: 12.07.2025**

Experiment 1: Working with Python packages-Numpy, Scipy, Scikit-Learn, Matplotlib

1 Aim:

To explore the various functions and methods available in the Python libraries and understand the key operations such as array manipulations, data preprocessing, mathematical computing, machine learning workflows, and data visualization.

2 Libraries used:

- Numpy
- Pandas
- Matplotlib
- Scikit-Learn
- Seaborn

3 Mathematical/Theoretical Description of the Algorithm/Objectives Performed

In this assignment, several fundamental data preprocessing and analytical techniques were applied to prepare the datasets for effective machine learning model training. The following summarizes the theoretical background and purpose of each technique used:

3.1 1. Handling Missing Values

Missing data in a dataset can lead to biased model outcomes or training failures. To address this:

- Columns with missing values were either **removed** if deemed non-essential, or
- **Imputed using the mode** (most frequent value) for categorical columns, ensuring no distortion in label distributions.

This ensures data integrity and completeness before model training.

3.2 2. Feature Importance via Word Frequency Comparison

In the spam email classification task:

- Each email was represented as a **bag-of-words vector**, with each feature representing the frequency of a unique word.
- To identify which words were most indicative of spam, the **relative frequency** of each word in spam vs. non-spam emails was calculated.
- A **frequency threshold** was applied to ignore rarely occurring words, ensuring that only statistically significant features were considered.

This helped in identifying **high-impact words** that contribute most to spam classification.

3.3 3. Correlation Analysis Between Features and Target

For datasets with numeric input features (e.g., diabetes and iris datasets):

- **Pearson correlation coefficients** were calculated between each input feature and the target label.
- For categorical targets (e.g., species in the iris dataset), the labels were first **converted into numerical format using Label Encoding**.

3.4 4. Standardization of Features

Input features in real-world datasets often have **different scales and units**. To address this:

- **Z-score standardization** was applied to numeric features using the formula:

$$z = \frac{x - \mu}{\sigma}$$

  where x is the feature value, μ is the mean, and σ is the standard deviation.

- This transformation ensures that all features have **zero mean and unit variance**, preventing models from being biased toward features with larger magnitudes.

3.5 5. Label Encoding

For datasets containing **categorical target variables** (like `species` in the iris dataset):

- Labels were encoded into numeric format using **LabelEncoder**, which assigns a unique integer to each category.
- This step is essential for machine learning algorithms that **require numerical inputs** for training and evaluation.

These preprocessing techniques lay a strong foundation for building robust and interpretable machine learning models by ensuring clean, consistent, and meaningful input data.

Dataset	Type of ML Task	Suitable ML Algorithm
Iris Dataset	Multi-class Classification	KNN, SVM
Loan Amount Prediction	Regression	Linear Regression
Predicting Diabetes	Binary Classification	SVM, XGBoost
Classification of Email Spam	Binary Classification	Logistic Regression, SVM
Handwritten Character Recognition	Multi-class Classification	CNN, SVM

Table 1: ML Task and Suitable Algorithms for Different Datasets

4 Results and Discussions:

Iris Dataset: The Iris dataset involves classifying flowers into three species using measurements of sepal and petal dimensions. Since this is a multi-class classification problem, algorithms like **K-Nearest Neighbors (KNN)** and **Support Vector Machine (SVM)** are well-suited.

Loan Amount Prediction: This task typically involves predicting either the loan approval status (classification) or the exact loan amount (regression). In this case, it was treated as a regression problem, for which **Linear Regression** is a suitable algorithm.

Predicting Diabetes: This binary classification problem uses features like glucose level, BMI, to predict the presence of diabetes. **Support Vector Machine (SVM)** is effective for such structured datasets.

Classification of Email Spam: This involves analyzing word frequencies in emails to determine if they are spam. Algorithms such as **Logistic Regression** and **SVM** are efficient due to their ability to handle high-dimensional sparse data.

Handwritten Character Recognition: Using the MNIST dataset, this task classifies grayscale images of digits (0–9). **Convolutional Neural Networks (CNNs)** are state-of-the-art for image classification, while **SVM** can also perform well with extracted features.

Click here to go to the code repository.

5 Learning Practices:

Throughout this assignment, various practical and theoretical skills were applied and reinforced through hands-on experimentation. The following learning practices were followed:

- **Data Cleaning:** Missing values were handled appropriately by either removing the affected records or imputing with meaningful statistics (e.g., mode for categorical features).
- **Text Analysis:** In the spam detection task, the bag-of-words model was used to quantify word importance. Words were ranked based on their frequency difference between spam and non-spam classes, while low-frequency noise was filtered using a threshold.
- **Feature Relevance:** Correlation analysis was performed to identify which features had the strongest relationship with the output label.