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	UCS2612 & Machine Learning Algorithms Laboratory		
Academic year	2025-2026 (Even)	Batch:2023-2027	**Due date: 23/12/25**

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

Name: Mahadev Ramesh Ramya
Reg.No: 3122235001075
Class: CSE-B

Aim:
To explore and work with Python packages like Numpy, Scikit-learn, and Matplotlib on datasets from public repositories and identify ML tasks, feature selection techniques, and suitable algorithms.

Libraries used:

- Numpy (imported as `np`)
- Pandas (imported as `pd`)
- Matplotlib.pyplot (imported as `plt`)
- Seaborn (imported as `sns`)
- OpenCV (`cv2`)
- OS (Standard Library)
- Math (Standard Library)

Mathematical/Theoretical description of the algorithm/objective performed:

- **Objective:** Exploratory Data Analysis (EDA) — summarize distributions, detect outliers, assess class balance, and reveal pairwise relationships/correlations.
- **Descriptive statistics**
  - Mean: $\mu = \frac{1}{n}\sum x_i$
  - Sample variance: $s^2 = \frac{1}{n-1}\sum(x_i - \mu)^2$
  - Quantiles (Median, $Q1$, $Q3$) used for spread and boxplot construction.
- **Histograms**
  - Empirical density approximation via bin counts; visualizes frequency/mode structure.
  - Kernel Density Estimate (KDE) used alongside histograms:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

where K is the kernel (e.g., Gaussian) and h is the bandwidth.

- **Boxplots**
  - Show Median, $Q1$, $Q3$; Interquartile Range $IQR = Q3 - Q1$.
  - Whiskers typically extend to min/max within $[Q1 - 1.5 \cdot IQR,\ Q3 + 1.5 \cdot IQR]$; points outside are shown as outliers.
- **Pairplots / Scatterplots**
  - Scatter for each feature pair to inspect linear/nonlinear relationships and class separation.
  - Diagonal: KDE or histogram per feature to show marginal distributions.
- **Correlation Heatmap**
  - Pearson correlation coefficient:

$$r_{XY} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y}$$

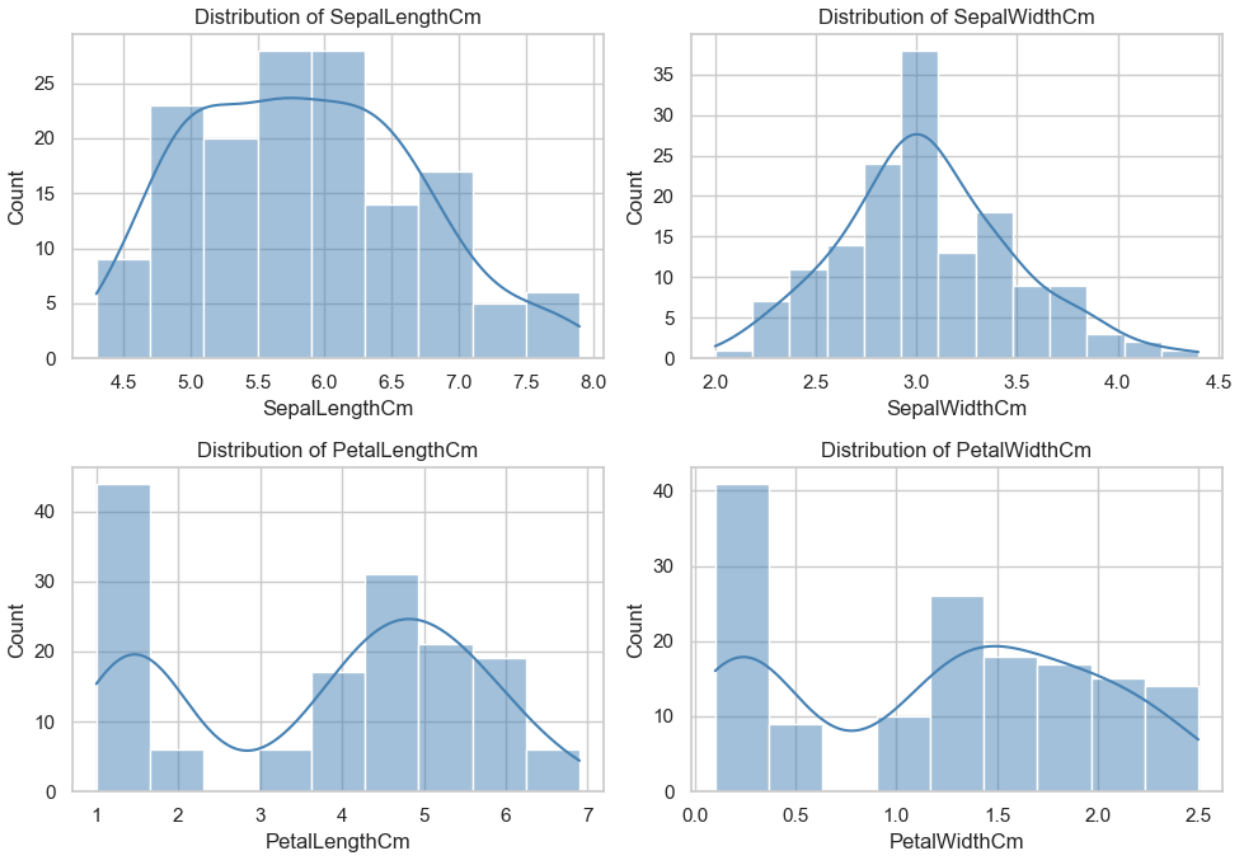
$$\text{cov}(X, Y) = \frac{1}{n-1} \sum (X_i - \bar{X})(Y_i - \bar{Y})$$

  - Values in $[-1, 1]$ indicate linear association strength/direction.
- **Countplots (Categorical)**
  - Bar heights = counts per category; optionally use hue for class-conditional counts to inspect imbalance/conditional distributions.
- **Image Dataset Overview**
  - Distributions of image heights/widths (histograms/KDE) to assess resizing needs.
  - Grid of sample images for qualitative inspection.
- **Missing Values**
  - Count missing per column to decide imputation/removal.
- **Practical Interpretation Aims**
  - Identify skewness, multimodality, outliers, collinearity, and class imbalance to guide preprocessing and modeling choices.

Results and Discussions:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

Figure 1: DATASET COLUMNS


Distribution of SepalLengthCm
Count
SepalLengthCm
Distribution of SepalWidthCm
Count
SepalWidthCm
Distribution of PetalLengthCm
Count
PetalLengthCm
Distribution of PetalWidthCm
Count
PetalWidthCm

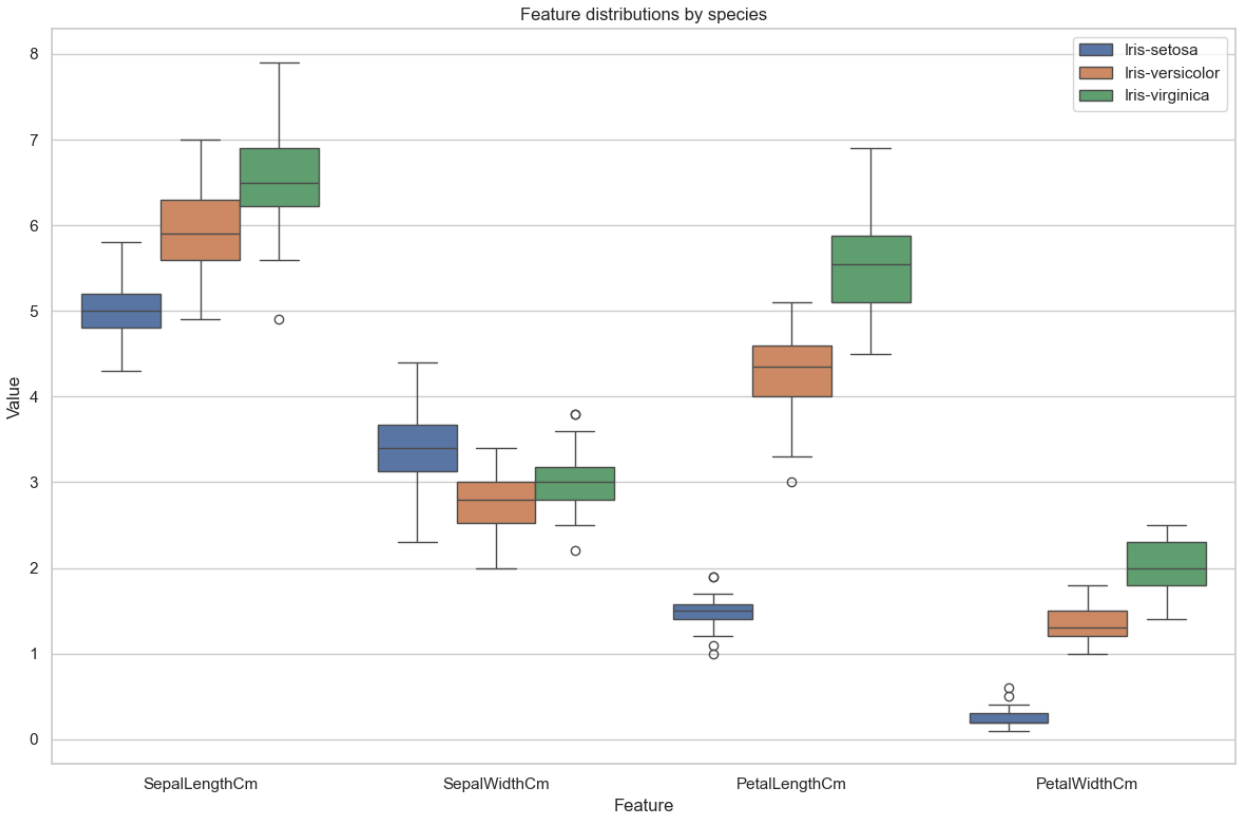


Figure 2: Distribution

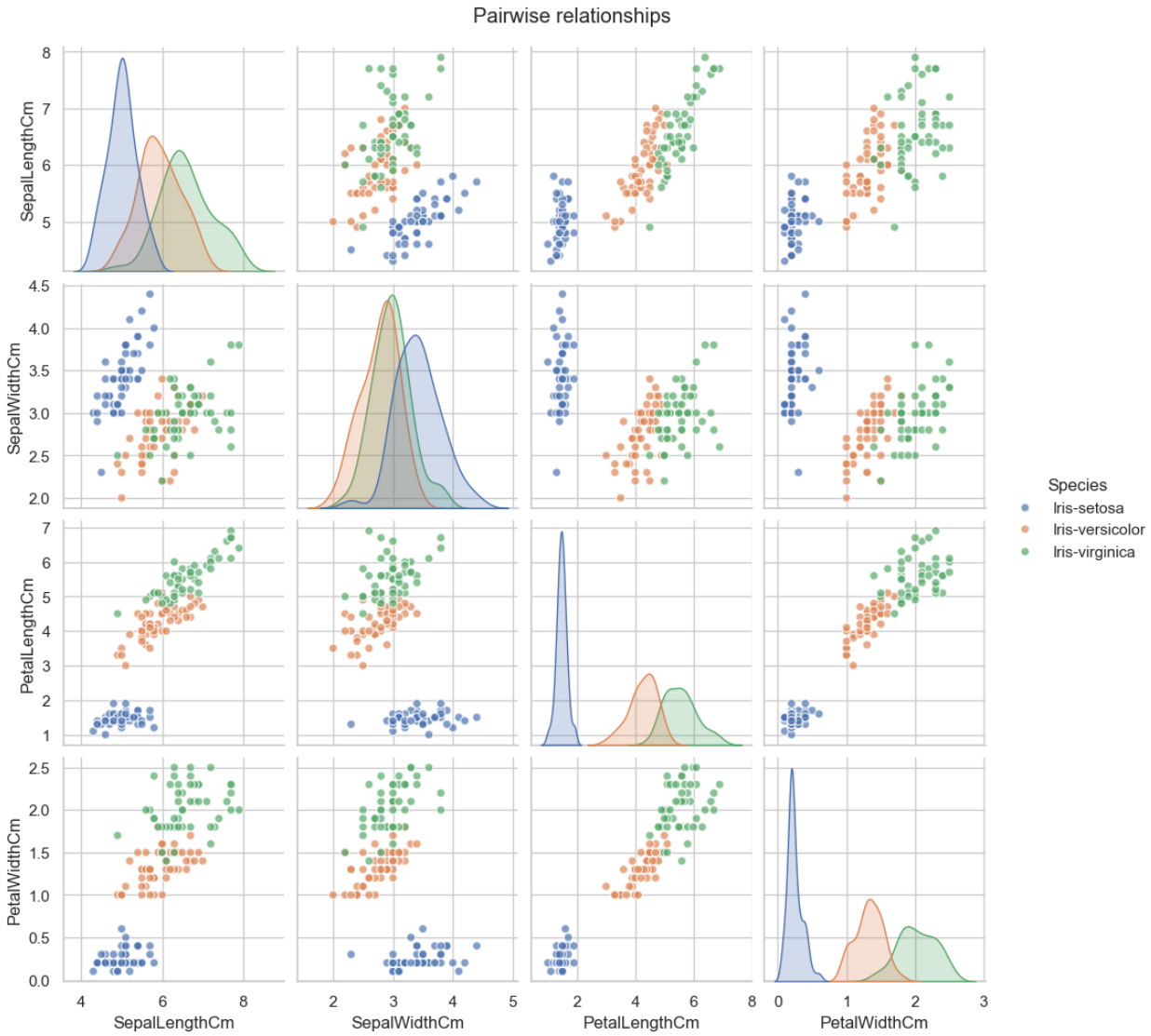


Figure 3: Relationships

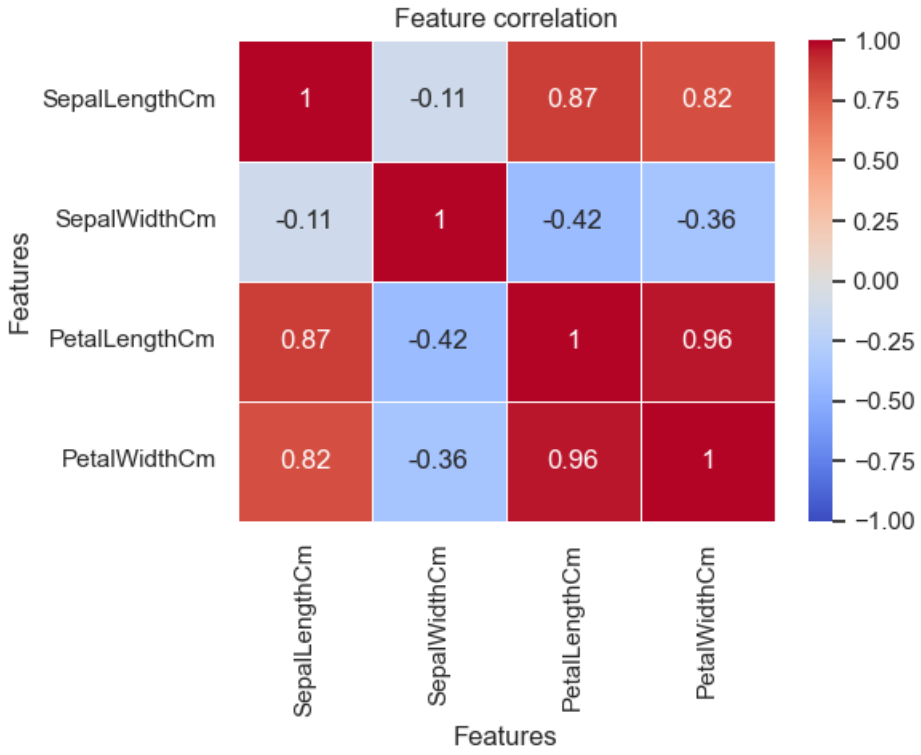


Figure 4: Correlation

Dataset	**Type of ML Task**	**Feature Selection Technique**	**Suitable ML Algorithm**
Iris Dataset	Multi-class Classification	Correlation Matrix / ANOVA F-value	k-Nearest Neighbors (k-NN), Decision Trees
Loan Amount Prediction	Regression	Recursive Feature Elimination (RFE) / Pearson Correlation	Linear Regression, Random Forest Regressor
Predicting Diabetes	Binary Classification	Chi-Square Test / SelectKBest	Logistic Regression, Support Vector Machine (SVM)
Classification of Email Spam	Binary Classification (NLP)	Information Gain / Chi-Square (on word vectors)	Naive Bayes, SVM
Handwritten Character Recognition / MNIST	Multi-class Image Classification	Principal Component Analysis (PCA)	Convolutional Neural Networks (CNN), SVM

Learning Practices:

- Interpret dataset structure: Learn to inspect shape, info, and missing values.
- Visualize distributions: Gain skills in plotting histograms, boxplots, and correlation heatmaps.
- Identify class balance: Understand how label distribution affects model performance.
- Spot feature relationships: Use pairplots and correlation matrices to detect predictive variables.
- Apply statistical tests: Use ANOVA F-test and correlation filtering.
- Leverage model-based importance: Interpret Random Forest feature importances.
- Perform dimensionality reduction: Use PCA for visualization and clustering.