**Assignment 2**

**Name**: Shubhankar D. Jakate  
**Roll No**: 282019  
**Batch**: B1

**Statement**

Perform the following operations using Python on the provided dataset:  
a) Compute and display summary statistics for each feature available in the dataset. (e.g. minimum, maximum, mean, range, standard deviation, variance, percentiles)  
b) Illustrate the feature distributions using histograms.  
c) Perform data cleaning, data integration, data transformation, and build a classification model.

**Objective**

1. To perform statistical analysis on features to understand the data distribution.
2. To visualize the distribution of features using histograms.
3. To preprocess the data (cleaning, integration, transformation) and prepare it for machine learning model building.
4. To implement and evaluate a classification model for prediction based on the dataset.

**Resources Used**

* **Software**: Jupyter Notebook
* **Programming Language**: Python
* **Libraries Used**: Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn

**Introduction to Dataset and Libraries**

1. **Pandas**: Used for data loading, preprocessing, and basic statistical functions.
2. **NumPy**: Provides mathematical functions for numerical operations.
3. **Matplotlib/Seaborn**: Visualization libraries to plot distributions and trends.
4. **Scikit-learn**: Used for model building (classification), data splitting, and evaluation metrics.

**Methodology**

**1. Data Loading and Exploration**

* Load dataset using pd.read\_csv()
* Use .head(), .info(), .describe() for initial data insights
* Check for missing values and data types

**2. Summary Statistics**

* Compute metrics using:
  + .min(), .max(), .mean(), .std(), .var()
  + .quantile([0.25, 0.5, 0.75]) for percentiles
* Use .apply() to calculate range (max - min)

**3. Feature Distribution Visualization**

* Plot histograms using df.hist() or sns.histplot() to show how each feature is distributed

**4. Data Cleaning**

* Handle missing values with strategies like mean/median imputation or removal
* Encode categorical features using one-hot encoding or label encoding

**5. Data Integration & Transformation**

* If data came from multiple sources, merge them using pd.merge() or concat()
* Normalize or scale features using StandardScaler or MinMaxScaler

**6. Model Building – Classification**

* Select a classification model (e.g., Logistic Regression, Decision Tree)
* Split data into training and testing sets using train\_test\_split()
* Train the model using .fit()
* Predict and evaluate using accuracy\_score, confusion\_matrix, classification\_report

**Advantages**

* Improved understanding of dataset features through visualization and summary statistics
* Cleaning and transformation helps in building accurate machine learning models
* Classification models can be easily built using Scikit-learn with minimum code

**Disadvantages**

* Real-world datasets may have large volumes of missing or inconsistent data
* Model accuracy heavily depends on quality of preprocessing and feature selection

**Conclusion**

This assignment helped in understanding the complete pipeline of data analysis and machine learning—from basic statistical analysis and visualization to data cleaning and model building. Such practice is fundamental in real-world data science and ML workflows, where each step significantly impacts the final model performance.