**SVKM’s NMIMS**

**Data WareHousing and Data Mining**

**BTech CS- Cyber**

**Sem IV ,2nd Year**

**Academic Year :2024-25**

**Lab Experiment**

|  |  |
| --- | --- |
| **Name: Jal Bafana** | **Roll no: K005** |
| **Class: Btech. Cyber Security** | **Batch: K1** |
| **Date of Experiment: 28.01.2025** | **Date of Submission: 28.01.2025** |

**Aim: To analyze and implement Exploratory Data Analysis**

**Analyze the following Python libraries**

1. Numpy
2. Pandas
3. NLTK
4. SCIKIT-LEARN

Analyze the below for EACH Python LIBRARY GIVEN above :

1. Significance
2. Functions /Features
3. EXAMPLE code (in Python)

**Answers and Code Screenshot**

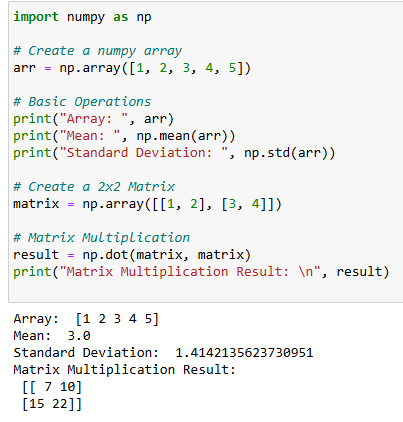
**Numpy**

**Significance:**

* Numpy (Numerical Python) is the foundational library for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

**Functions / Features:**

* Array Creation: np.array(), np.arange(), np.linspace(), etc.
* Array Operations: Arithmetic operations, reshaping, slicing, and indexing.
* Mathematical Functions: np.add(), np.subtract(), np.mean(), np.std(), etc.
* Linear Algebra: np.dot(), np.linalg.inv(), np.linalg.eig().
* Random Number Generation: np.random.rand(), np.random.randn().



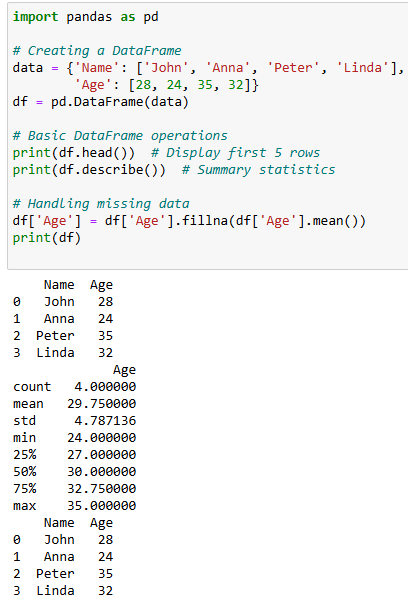
**Pandas**

**Significance:**

* Pandas is a powerful data manipulation and analysis library. It provides data structures like DataFrame and Series that allow for easy handling and analysis of structured data (like tables in databases or CSV files).

**Functions / Features:**

* DataFrame Creation: pd.DataFrame()
* Data Import/Export: pd.read\_csv(), pd.to\_csv()
* Data Operations: df.head(), df.describe(), df.dropna(), df.fillna()
* Group and Aggregate: df.groupby(), df.agg()
* Handling Missing Data: df.isna(), df.dropna()



**NLTK (Natural Language Toolkit)**

**Significance:**

* NLTK is a powerful library for working with human language data. It provides tools for text processing, classification, tokenization, stemming, and other common natural language processing tasks.

**Functions / Features:**

* Text Tokenization: nltk.word\_tokenize(), nltk.sent\_tokenize()
* Stemming: nltk.stem.PorterStemmer()
* Part-of-Speech Tagging: nltk.pos\_tag()
* Named Entity Recognition: nltk.chunk.ne\_chunk()
* Text Classification: nltk.classify()



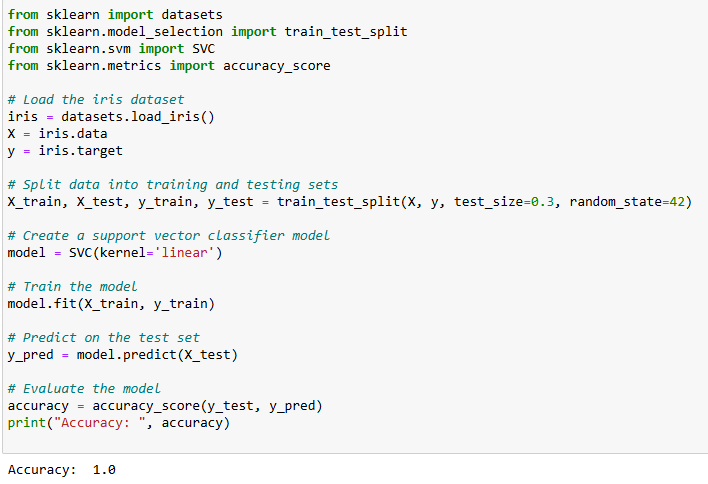
**Scikit-learn**

**Significance:**

* Scikit-learn is a machine learning library built on top of Numpy and Scipy. It provides simple and efficient tools for data mining, data analysis, and machine learning tasks, including classification, regression, clustering, and dimensionality reduction.

**Functions / Features:**

* Classification: sklearn.svm.SVC(), sklearn.tree.DecisionTreeClassifier()
* Regression: sklearn.linear\_model.LinearRegression()
* Clustering: sklearn.cluster.KMeans()
* Preprocessing: sklearn.preprocessing.StandardScaler()
* Model Evaluation: sklearn.metrics.accuracy\_score(), sklearn.metrics.confusion\_matrix()

****

**Observation**

These libraries collectively enable efficient data manipulation, analysis, and machine learning tasks, each addressing different aspects of data processing. Together, they provide a comprehensive toolkit for handling numerical, structured, and textual data.

**Conclusion**

Mastering these libraries is crucial for performing effective data analysis and building machine learning models. They offer powerful, efficient solutions for a wide range of data science tasks.