Name: Rugved Sawant
Roll No: 281065
Batch: A3

Assignment 1

Statement

Q. Perform the following operations using R/Python on suitable datasets:

a) Read data from different formats (CSV, XLS)
b) Find the shape of the data
c) Identify missing values
d) Determine the data type of each column
e) Count zeros in the dataset
f) Index, select, and sort data
g) Describe attributes of data and check data types
h) Count unique values, check format of each column, and convert data types (e.g., long to short, and vice versa)

Objective

1. To introduce the **Pandas** library and its core functionalities for reading files in formats such as CSV and Excel.
2. To gain familiarity with **data cleaning and preprocessing** techniques.
3. To enhance data handling and manipulation skills using Python, fostering proficiency in **basic data analysis**.

Resources Used

- **Software**: Google Colab
- **Library**: Pandas

Introduction to Pandas

Pandas is a powerful and widely adopted **open-source Python library** designed for **data manipulation and analysis**. It offers intuitive data structures and tools that simplify working with structured data.

Key data structures in Pandas:

- **Series**: A one-dimensional labeled array.
- **DataFrame**: A two-dimensional labeled data structure with columns of potentially different types.

Pandas supports a wide range of operations, such as:

- Reading data from various file formats (CSV, Excel, SQL, etc.)
- Sorting, filtering, and grouping data
- Performing statistical and analytical tasks

Basic Functions Used

1. pd.read_csv() – Reads data from a CSV file.
2. shape – Returns the number of rows and columns.
3. isnull().sum() – Detects missing values.
4. dtypes – Displays the data type of each column.
5. (df == 0).sum() – Counts the number of zeros in each column.
6. sort_values() – Sorts the DataFrame by values in specified columns.
7. describe() – Generates descriptive statistics for numerical columns.
8. unique() – Returns unique values in a column, useful for analyzing categorical data.

Methodology

1. **Data Collection and Exploration**
   - Load a relevant dataset into a Pandas DataFrame.
   - Analyze the dataset's structure, sample size, features, data types, and missing values.
2. **Data Preprocessing**
   - **Missing Values**: Handle them via imputation or removal.
   - **Data Cleaning**: Remove duplicates, correct erroneous entries, and standardize formats.
3. **Feature Engineering**
   - **Feature Selection**: Choose important features based on domain knowledge.
   - **Feature Encoding**: Convert categorical data into numerical format using encoding techniques like one-hot or label encoding.

Advantages of Pandas

1. User-friendly and intuitive for beginners.
2. Offers powerful structures like **Series** and **DataFrame**.
3. Provides extensive capabilities for **data manipulation and analysis**.

Disadvantages of Pandas

1. Can be memory-intensive when handling large datasets.
2. Primarily Python-based, limiting its use with other programming ecosystems.

Conclusion

This assignment provided a comprehensive introduction to the **Pandas** library—an essential tool for data manipulation in Python. We practiced reading data from various formats, analysing and cleaning it, and understanding its structure. Through hands-on implementation, we built foundational skills that will be invaluable for more advanced data science and analysis projects in the future.