**My Assignment topic**

**Assignment 3**

**Data Analysis and Visualization Using Pandas**

**By**

**Fahad Mughal**

**ROLL #:1220100509**

**ADP(IT)**

**4TH SEMESTER**

**Fall 2024**

**COURSE TITLE:** **Artifical Intellegence**

**SUBMITTED TO : Sir Zubair**

****

**Department of Information technology International Institute of Science, Arts and Technology (IISAT), Gujranwala**

Data Analysis and Visualization Using Pandas

**Answer :**

**1. Data Set Selection and Loading**

For this example, I'll use the Iris dataset from scikit-learn. You can replace this with any dataset of your choice by downloading it and loading it into a Pandas DataFrame.

**python**

# Import necessary libraries

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.datasets import load\_iris

# Load the Iris dataset from scikit-learn

iris = load\_iris()

# Create a DataFrame

df = pd.DataFrame(data=iris.data, columns=iris.feature\_names)

# Add the target variable 'species' to the DataFrame

df['species'] = iris.target

# Map target numbers to actual species names

df['species'] = df['species'].map({0: 'setosa', 1: 'versicolor', 2: 'virginica'})

# Display the first few rows of the DataFrame to understand its structure

print(df.head())

**2. Data Exploration**

Now, let's perform a detailed exploration of the dataset to understand its structure, features, and statistical summary.

**python**

# Check the dimensions of the DataFrame (rows, columns)

print(f"Dataset Dimensions: {df.shape}")

# Summary statistics

print("\nSummary Statistics:")

print(df.describe())

# Information about the dataset (column names, data types, non-null values)

print("\nInformation about the dataset:")

print(df.info())

**3. Data Cleaning**

In this example with the Iris dataset, there's no need for extensive data cleaning as it's a clean dataset. However, you might perform tasks like handling missing values, dealing with duplicates, or performing data transformations based on the dataset you choose.

**4. Data Visualization**

Let's create various graphs and charts using Pandas, Matplotlib, and Seaborn to visualize the dataset.

**Example Visualizations:**

python

# Pairplot to visualize pairwise relationships between variables

sns.pairplot(df, hue='species', height=2.5)

plt.title('Pairplot of Iris Dataset')

plt.show()

# Boxplot to visualize distribution of each feature by species

plt.figure(figsize=(10, 6))

sns.boxplot(x='species', y='petal length (cm)', data=df)

plt.title('Distribution of Petal Length by Species')

plt.show()

# Histogram of each feature colored by species

plt.figure(figsize=(10, 6))

for feature in iris.feature\_names:

sns.histplot(df, x=feature, hue='species', element='step', kde=True)

plt.title('Distribution of Features by Species')

plt.tight\_layout()

plt.show()

**5. Analysis and Insights**

After each visualization, provide an analysis and insights derived from it. Here's an example of what you might discuss:

* **Pairplot**: This visualization shows the pairwise relationships between all features in the Iris dataset. It's evident from the plots that the setosa species is well separated from versicolor and virginica, especially in terms of petal and sepal dimensions.
* **Boxplot**: The boxplot provides a clear view of the distribution of petal length across different species. It highlights that setosa typically has shorter petals compared to versicolor and virginica, which have overlapping distributions but differ slightly in median petal length.
* **Histograms**: These histograms provide insights into the distribution of individual features (sepal length, sepal width, petal length, petal width) across species. For instance, petal length seems to be a distinguishing feature where setosa has a shorter range compared to versicolor and virginica, which have longer petals on average.