Experiment - 10. (Group A).

- Title of the assignment - Data Visualization III.
- Problem statement - Download the iris flower dataset or any other dataset into a dataframe. Use python/R to perform following operations.
  - How many features are there and what are their types (eg numeric, nominal)?
  - Compute and display summary statistics for each feature available in the dataset (eg. minimum value, maximum value, mean, range, standard deviation, variance and percentile
  - Data visualization - Create a histogram for each feature in the dataset to illustrate the feature distributions plot each diagram histogram.
  - Create a boxplot for each feature in the dataset to illustrate the feature distribution.
    plot each histogram.
  - Create a boxplot for each feature in the dataset. All of the boxplots should be combined into a single plot.
  - Compare distributions and identify outliers.
- Pre-requisites - Fundamentals of python or R languages.

- **Objectives:** To learn the concept of how to display summary statistics for each feature available in the dataset.

Implement a dataset into a dataframe. Implement the following operations. –

1) Display data set details.
2) Calculate min, max, mean, range, SD, variance.
3) Create histograph using hist function.
4) Create boxplot using boxplot function.

- **THEORY:**

Simplify comparisons of sets of numbers, especially large sets of number, by calculating the center values using mean, mode and median. Use the ranges and S.D of the sets to examine the variability of data.

1] **Calculating Measures of Central tendency –**

These describe the center or average of a dataset.

a) **Mean (Arithmetic average) –**

Formula – $$\text{Mean} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

- Where x_i is each value and n is the number of observations.

- Represents "typical" value.
- Sensitive to outliers.

b) Median (Middle value).

- The middle value when the data is stored.
- If n is odd, median is the middle value.
- If n is even, median is the average of the two middle values.
- Robust to outliers.

c) Mode -

- The value that appears most frequently in the dataset.
- A dataset can have more than one mode (bimodal, multimodal).

---

2) Measures of Variability -

These describe the spread or dispersion of the data.

a) Range.

Range = Maximum Value - Minimum value.

- Simple measure of spredd.
- Very sensitive to others.

b) Variance (Sample) -

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2$$

- $\bar{x}$ is the mean, and x_i are individual values.
- Tell us how spread out the values are from the mean.
- Units are squared of the original data.

(c). Standard deviation -

$$S = \sqrt{s^2} = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

- Square root of variance.
- Same units as the original data.
- Commonly used to understand data variability.

(d) Interquartile range (IQR) -

$$IQR = Q_3 - Q_1$$

- Difference between 75th percentile (Q3) and 25th percentile (Q1).
- Helps detect outliers.
- Focuses on the middle 50% of the data.

* Algorithm (Steps for program) -

1) Load the dataset into data frame -

```
import pandas as pd.
import seaborn as sns.
df = sns.load_dataset("iris")
```

(2) Display dataset details

- head()
- tail()
- info()
- describe (include = "all")
- shape
- size
- ndim
- columns

(3) Feature details and types -

df.columns.tolist()
df.dtypes

(4) Summary statistics for each feature -

- min()
- max()
- mean
- median
- mode
- std
- quantile

(5) Data visualization -

A. df.hist(figsize=(10,8), color='skyblue', edgecolor='black')
plt.show()

B. Boxplot for all features -

plt.figure(figsize=(10,6))
sns.boxplot(data=df.drop('species', axis=1), palette='pastel')
plt.show()

- Boxplot help detect outliers in features.

C. Pair plot.

```
sns.pairplot(df, hue="species", palette="set2")
```

- Visualizes correlation between pairs of features.
- Highlights separation between species.

D. Heatmap.

```
sns.heatmap(df.corr(numeric_only=True),
annot=True, comp="Coolwarm", fmt=".2f")
```

- Identifies strong linear relationships.
- Helps select relevant features.

E. Violin plot.

```
sns.violinplot(data=df, x='species', y='petal_length',
palette="Pastel1")
```

- Shows distribution + density + median per category.

D. Swarm plot.

```
sns.swarmplot(data=df, x='species', y='sepal_width'
, palette="set3")
```

- Great for small datasets.
- Displays distribution and clustering per category.

- **Conclusion:** Hence, we have studied and applied expected visualization on dataset.

* * * * * * *