

Exp 9

I'll load the dataset you provided to write the AIM, theory, algorithm, and code for the experiment.

Python

```
import pandas as pd

pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)

# Read the CSV file into a DataFrame
df = pd.read_csv('Iris (1).csv')

# Display the first 5 rows
print(df.head().to_markdown(index=False, numalign="left", stralign="left"))

# Print the column names and their data types
print(df.info())
```

Code output

```
| Id   | SepalLengthCm   | SepalWidthCm   | PetalLengthCm   | PetalWidthCm
| Species |
|:-----|:-----|:-----|:-----|:-----
-|:-----|
| 1     | 5.1             | 3.5            | 1.4             | 0.2
| Iris-setosa |
| 2     | 4.9             | 3              | 1.4             | 0.2
| Iris-setosa |
| 3     | 4.7             | 3.2            | 1.3             | 0.2
| Iris-setosa |
| 4     | 4.6             | 3.1            | 1.5             | 0.2
| Iris-setosa |
| 5     | 5               | 3.6            | 1.4             | 0.2
| Iris-setosa |
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype
```

```
---  -----  -----  -----
0   Id           150 non-null  int64
1   SepalLengthCm 150 non-null  float64
2   SepalWidthCm  150 non-null  float64
3   PetalLengthCm 150 non-null  float64
4   PetalWidthCm  150 non-null  float64
5   Species       150 non-null  object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
None
```

Here's the experiment setup using the dataset you provided:

Experiment Number: 10

AIM: To analyze the Iris dataset and calculate descriptive statistics for the 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', and 'PetalWidthCm' features, grouped by the 'Species' column.

Objectives:

- To understand the structure and characteristics of the Iris dataset.
- To learn how to group data based on categorical variables (Species).
- To apply descriptive statistics to summarize and compare the data for each species.
- To interpret the statistical results and gain insights into the differences between species.

Course Outcomes: CO3, CO4 (These likely relate to data exploration, analysis, and descriptive statistics)

Resource/Tools: Jupyter Notebook, Python with Pandas library.

Theory:

The Iris dataset is a classic dataset in statistics and machine learning. It contains measurements (sepal length, sepal width, petal length, petal width) for 150 iris flowers from three different species: 'Iris-setosa', 'Iris-versicolor', and 'Iris-virginica'. Analyzing the statistical properties of these measurements for each species can reveal insights into the characteristics of each species and their differences.

Dataset Details:

- The dataset contains 6 columns: 'Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm', and 'Species'.
- The 'Id' column is a unique identifier for each row.

- The 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', and 'PetalWidthCm' columns contain the measurements of the sepals and petals in centimeters.
- The 'Species' column is a categorical variable indicating the species of the iris flower.

Algorithm:

1. Load the dataset: Import the Iris dataset using Pandas.
2. Group data by species: Group the data by the 'Species' column.
3. Calculate descriptive statistics: Calculate descriptive statistics for each numerical feature ('SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm') for each group (species).
4. Display the results: Print the calculated statistics in a clear and organized format.

Code:

Python

```
import pandas as pd

# Load the Iris dataset
df = pd.read_csv('Iris (1).csv')

# Group data by species
grouped_data = df.groupby('Species')

# Calculate descriptive statistics
for species, data in grouped_data:
    print(f"\nStatistics for {species}:")
    print(data[['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm',
'PetalWidthCm']].describe().to_markdown(numalign="left", stralign="left"))
    print('-' * 60)
```