practice-assignment-10

May 4, 2025

[3]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

[4]:

```
vm=pd.read_csv("Iris.csv")
vm
```

[4]:

```
      Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  \
0      1            5.1           3.5            1.4           0.2
1      2            4.9           3.0            1.4           0.2
2      3            4.7           3.2            1.3           0.2
3      4            4.6           3.1            1.5           0.2
4      5            5.0           3.6            1.4           0.2
..   …            …           …            …           …
145  146            6.7           3.0            5.2           2.3
146  147            6.3           2.5            5.0           1.9
147  148            6.5           3.0            5.2           2.0
148  149            6.2           3.4            5.4           2.3
149  150            5.9           3.0            5.1           1.8

            Species
0       Iris-setosa
1       Iris-setosa
2       Iris-setosa
3       Iris-setosa
4       Iris-setosa
..              …
145  Iris-virginica
146  Iris-virginica
147  Iris-virginica
148  Iris-virginica
149  Iris-virginica

[150 rows x 6 columns]
```

[5]:

```
vm.info()
```

```
<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
```

[6]:

```
vm.describe(include="all")
```

[6]:

```
                Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  \
count   150.000000     150.000000    150.000000     150.000000    150.000000   
unique         NaN            NaN           NaN            NaN           NaN   
top            NaN            NaN           NaN            NaN           NaN   
freq           NaN            NaN           NaN            NaN           NaN   
mean     75.500000       5.843333      3.054000       3.758667      1.198667   
std      43.445368       0.828066      0.433594       1.764420      0.763161   
min       1.000000       4.300000      2.000000       1.000000      0.100000   
25%      38.250000       5.100000      2.800000       1.600000      0.300000   
50%      75.500000       5.800000      3.000000       4.350000      1.300000   
75%     112.750000       6.400000      3.300000       5.100000      1.800000   
max     150.000000       7.900000      4.400000       6.900000      2.500000   

            Species  
count           150  
unique            3  
top     Iris-setosa  
freq             50  
mean            NaN  
std             NaN  
min             NaN  
25%             NaN  
50%             NaN  
75%             NaN  
max             NaN  
```

[7]:

```
vm.head()
```

[7]:

```
   Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
0   1            5.1           3.5            1.4           0.2  Iris-setosa
1   2            4.9           3.0            1.4           0.2  Iris-setosa
```

```
2   3            4.7           3.2            1.3           0.2  Iris-setosa
3   4            4.6           3.1            1.5           0.2  Iris-setosa
4   5            5.0           3.6            1.4           0.2  Iris-setosa
```

```
[8]: vm.tail()
```

```
[8]:       Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  \
     145  146            6.7           3.0            5.2           2.3
     146  147            6.3           2.5            5.0           1.9
     147  148            6.5           3.0            5.2           2.0
     148  149            6.2           3.4            5.4           2.3
     149  150            5.9           3.0            5.1           1.8

                 Species
     145  Iris-virginica
     146  Iris-virginica
     147  Iris-virginica
     148  Iris-virginica
     149  Iris-virginica
```

```
[9]: vm.isnull()
```

```
[9]:         Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm  Species
     0    False          False         False          False         False    False
     1    False          False         False          False         False    False
     2    False          False         False          False         False    False
     3    False          False         False          False         False    False
     4    False          False         False          False         False    False
     ..     …            …           …            …           …      …
     145  False          False         False          False         False    False
     146  False          False         False          False         False    False
     147  False          False         False          False         False    False
     148  False          False         False          False         False    False
     149  False          False         False          False         False    False

     [150 rows x 6 columns]
```

```
[10]: vm.isnull().sum()
```

```
[10]: Id                0
      SepalLengthCm     0
      SepalWidthCm      0
      PetalLengthCm     0
      PetalWidthCm      0
      Species           0
      dtype: int64
```

```
[11]: vm.shape
```

```
[11]: (150, 6)
```

```
[12]: vm.size
```

```
[12]: 900
```

```
[13]: vm.dtypes
```

```
[13]: Id                 int64
      SepalLengthCm    float64
      SepalWidthCm     float64
      PetalLengthCm    float64
      PetalWidthCm     float64
      Species           object
      dtype: object
```

```
[14]: vm.Species.describe()
```

```
[14]: count             150
      unique              3
      top       Iris-setosa
      freq               50
      Name: Species, dtype: object
```

```
[15]: vm.Species.mode()
```

```
[15]: 0        Iris-setosa
      1    Iris-versicolor
      2     Iris-virginica
      Name: Species, dtype: object
```

```
[16]: #mean
      print("Sepal Length",vm.SepalLengthCm.mean())
      print("Sepal Width",vm.SepalWidthCm.mean())
      print("Petal Length",vm.PetalLengthCm.mean())
      print("Petal Width",vm.PetalWidthCm.mean())
```

```
Sepal Length 5.843333333333334
Sepal Width 3.0540000000000003
Petal Length 3.758666666666666
Petal Width 1.1986666666666668
```

```
[17]: #median
      print("Sepal Length",vm.SepalLengthCm.median())
      print("Sepal Width",vm.SepalWidthCm.median())
      print("Petal Length",vm.PetalLengthCm.median())
      print("Petal Width",vm.PetalWidthCm.median())
```

```
Sepal Length 5.8
Sepal Width 3.0
Petal Length 4.35
Petal Width 1.3
```

[18]:

```
#standard deviation
print("Sepal Length",vm.SepalLengthCm.std())
print("Sepal Width",vm.SepalWidthCm.std())
print("Petal Length",vm.PetalLengthCm.std())
print("Petal Width",vm.PetalWidthCm.std())


 #mode
print("Sepal Length",vm.SepalLengthCm.mode())
print("Sepal Width",vm.SepalWidthCm.mode())
print("Petal Length",vm.PetalLengthCm.mode())
print("Petal Width",vm.PetalWidthCm.mode())
```

```
Sepal Length 0.8280661279778629
Sepal Width 0.4335943113621737
Petal Length 1.7644204199522617
Petal Width 0.7631607417008414
Sepal Length 0    5.0
Name: SepalLengthCm, dtype: float64
Sepal Width 0    3.0
Name: SepalWidthCm, dtype: float64
Petal Length 0    1.5
Name: PetalLengthCm, dtype: float64
Petal Width 0    0.2
Name: PetalWidthCm, dtype: float64
```

[19]:

```
#max
print("Sepal Length",vm.SepalLengthCm.max())
print("Sepal Width",vm.SepalWidthCm.max())
print("Petal Length",vm.PetalLengthCm.max())
print("Petal Width",vm.PetalWidthCm.max())
```

```
Sepal Length 7.9
Sepal Width 4.4
Petal Length 6.9
Petal Width 2.5
```

[20]:

```
#min
print("Sepal Length",vm.SepalLengthCm.min())
print("Sepal Width",vm.SepalWidthCm.min())
print("Petal Length",vm.PetalLengthCm.min())
print("Petal Width",vm.PetalWidthCm.min())
```

```
Sepal Length 4.3
Sepal Width 2.0
Petal Length 1.0
Petal Width 0.1
```

[21]:

```
# percentile
print("Sepal Length",vm.SepalLengthCm.quantile(0.25))
print("Sepal Width",vm.SepalWidthCm.quantile(0.25))
print("Petal Length",vm.PetalLengthCm.quantile(0.25))
print("Petal Width",vm.PetalWidthCm.quantile(0.25))
print("")
print("Sepal Length",vm.SepalLengthCm.quantile(0.50))
print("Sepal Width",vm.SepalWidthCm.quantile(0.50))
print("Petal Length",vm.PetalLengthCm.quantile(0.50))
print("Petal Width",vm.PetalWidthCm.quantile(0.50))
print("")
print("Sepal Length",vm.SepalLengthCm.quantile(0.75))
print("Sepal Width",vm.SepalWidthCm.quantile(0.75))
print("Petal Length",vm.PetalLengthCm.quantile(0.75))
print("Petal Width",vm.PetalWidthCm.quantile(0.75))
```

```
Sepal Length 5.1
Sepal Width 2.8
Petal Length 1.6
Petal Width 0.3

Sepal Length 5.8
Sepal Width 3.0
Petal Length 4.35
Petal Width 1.3

Sepal Length 6.4
Sepal Width 3.3
Petal Length 5.1
Petal Width 1.8
```

[22]:

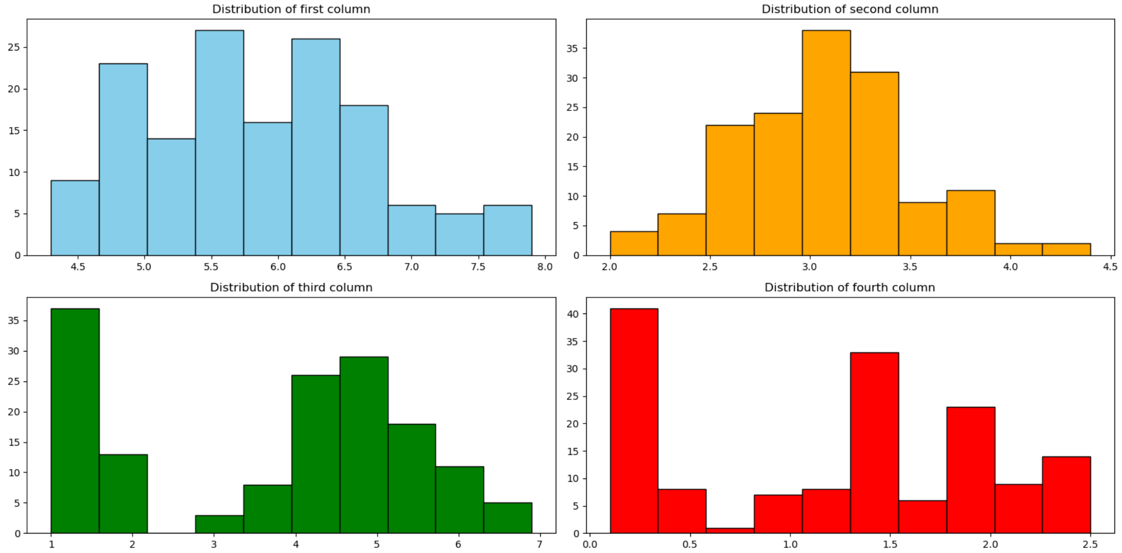
```
column=len(list(vm))
column
np.unique(vm['Species'])
```

[22]:

```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

[23]:

```
# Jupyter Notebook magic command
%matplotlib inline


# Create subplots
fig, axes = plt.subplots(2, 2, figsize=(16, 8))

# Plot histograms
axes[0, 0].set_title("Distribution of first column")
axes[0, 0].hist(vm['SepalLengthCm'], color='skyblue', edgecolor='black')

axes[0, 1].set_title("Distribution of second column")
axes[0, 1].hist(vm['SepalWidthCm'], color='orange', edgecolor='black')

axes[1, 0].set_title("Distribution of third column")
axes[1, 0].hist(vm['PetalLengthCm'], color='green', edgecolor='black')

axes[1, 1].set_title("Distribution of fourth column")
axes[1, 1].hist(vm['PetalWidthCm'], color='red', edgecolor='black')

# Adjust layout
plt.tight_layout()
plt.show()
```



Distribution of first column

Distribution of second column

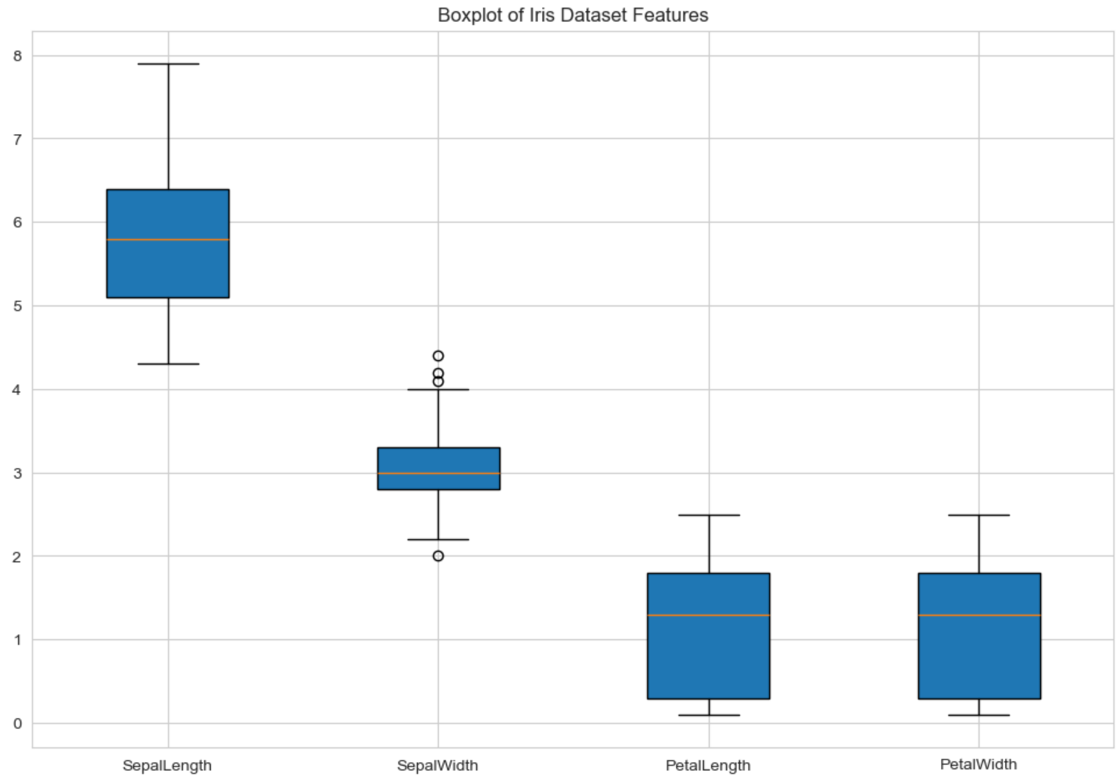
Distribution of third column

Distribution of fourth column

[24]:

```
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Prepare data for boxplot
data_to_plot = [vm['SepalLengthCm'], vm['SepalWidthCm'], vm['PetalWidthCm'],␣
 ↪vm['PetalWidthCm']]

# Set style
sns.set_style("whitegrid")

# Create figure and axis
fig = plt.figure(1, figsize=(12, 8))
ax = fig.add_subplot(111)

# Create boxplot
bp = ax.boxplot(data_to_plot, patch_artist=True)

# Set x-axis labels
ax.set_xticklabels(['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth'])

# Set title
ax.set_title('Boxplot of Iris Dataset Features')

# Show plot
plt.show()
```



9

[ ]:

```
```