untitled5-1

January 24, 2024

Assignment on Python for Data Science

Do the following operation on Iris dataset: 1. Read the dataset to python environment.

[28]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
iris=pd.read_excel('/content/iris.xls')
```

2. Display the columns in the dataset.

[29]:
```
#Displaying first 5 columns of the iris data set
iris.head()
```

[29]:
```
    SL   SW   PL   PW Classification
0  5.1  3.5  1.4  0.2    Iris-setosa
1  4.9  3.0  1.4  0.2    Iris-setosa
2  4.7  3.2  1.3  0.2    Iris-setosa
3  4.6  3.1  1.5  0.2    Iris-setosa
4  5.0  3.6  1.4  0.2    Iris-setosa
```

[30]:
```
#Displaying the column names
iris.columns
```

[30]:
```
Index(['SL', 'SW', 'PL', 'PW', 'Classification'], dtype='object')
```

3. Calculate the mean of each column of the dataset.

[31]:
```
iris.mean()
```

```
<ipython-input-31-7eed97565d6e>:1: FutureWarning: The default value of
numeric_only in DataFrame.mean is deprecated. In a future version, it will
default to False. In addition, specifying 'numeric_only=None' is deprecated.
Select only valid columns or specify the value of numeric_only to silence this
warning.
  iris.mean()
```

[31]:
```
SL    5.843333
SW    3.054000
```

```
PL    3.758667
PW    1.198667
dtype: float64
```

4. Check for the null values present in the dataset.

[32]:

```
iris.isna().sum()
```

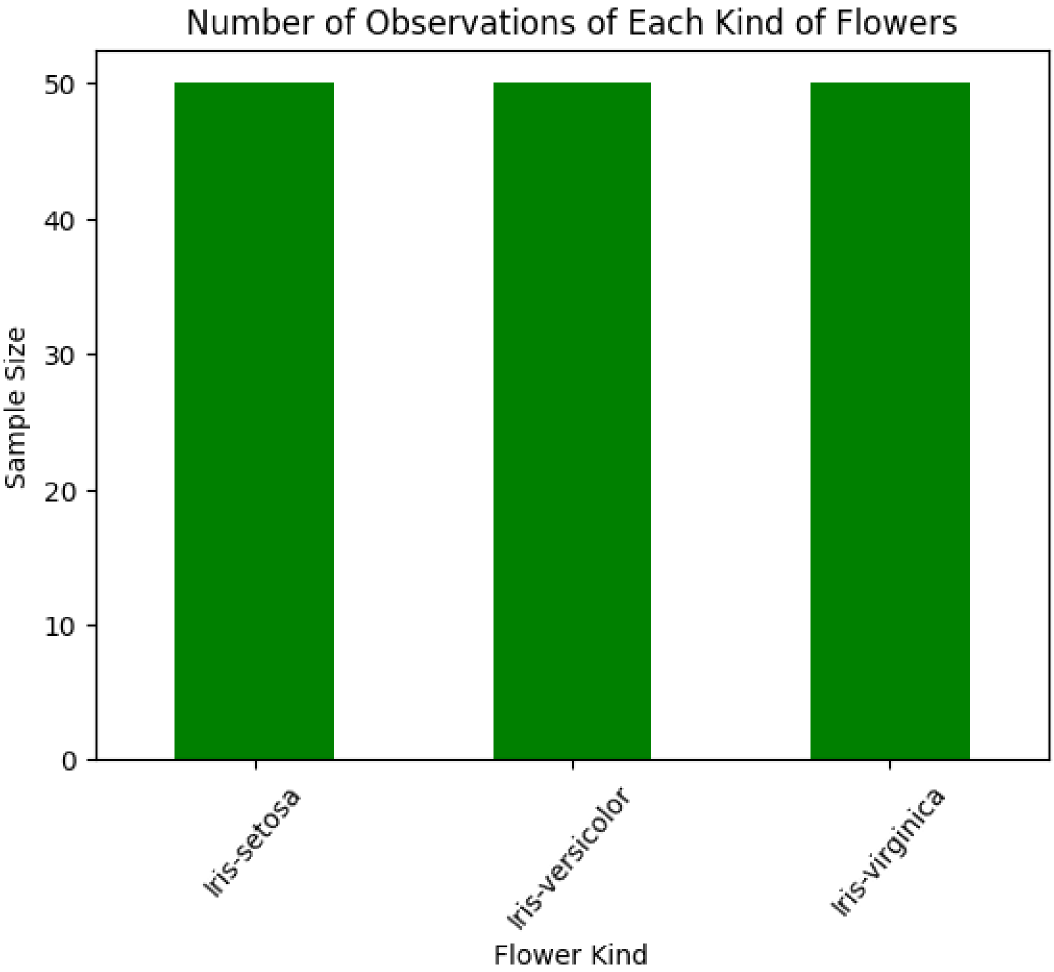
[32]:

```
SL                0
SW                0
PL                0
PW                0
Classification    0
dtype: int64
```

5. Perform meaningful visualizations using the dataset. Bring at least 3 visualizations.

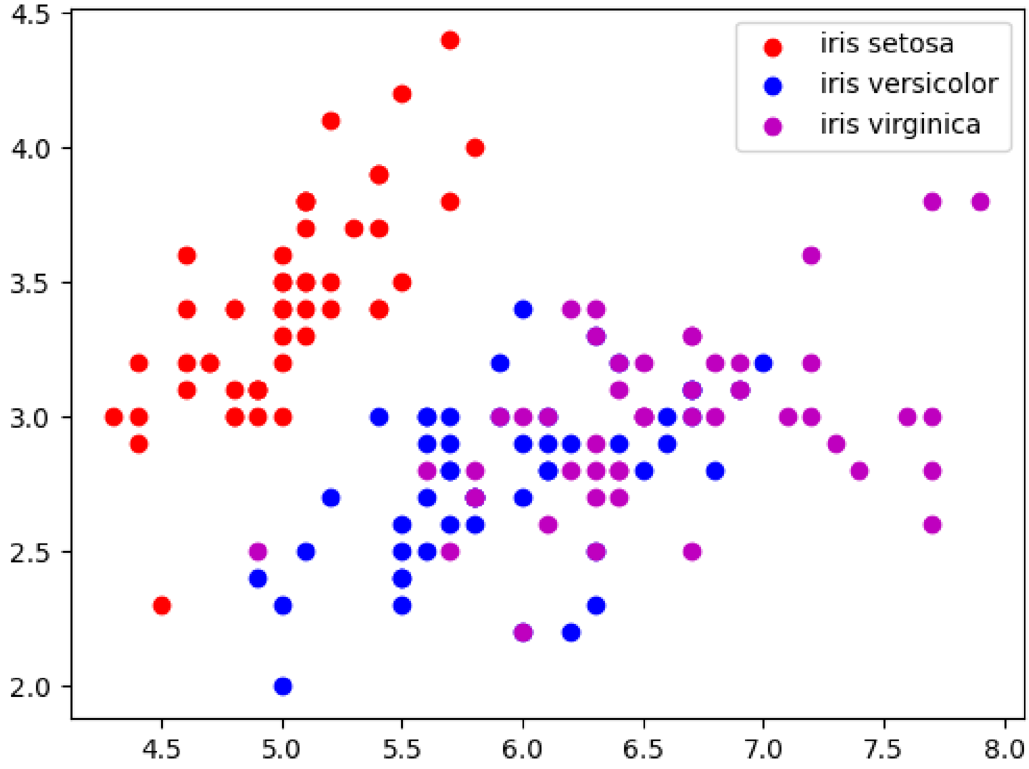
[34]:

```
#The following Bar graph display the number of observations of each flower
iris.Classification.value_counts().plot(kind="bar",color="g")
plt.xticks(rotation=50)
plt.title("Number of Observations of Each Kind of Flowers")
plt.xlabel("Flower Kind")
plt.ylabel("Sample Size")
plt.show()
```



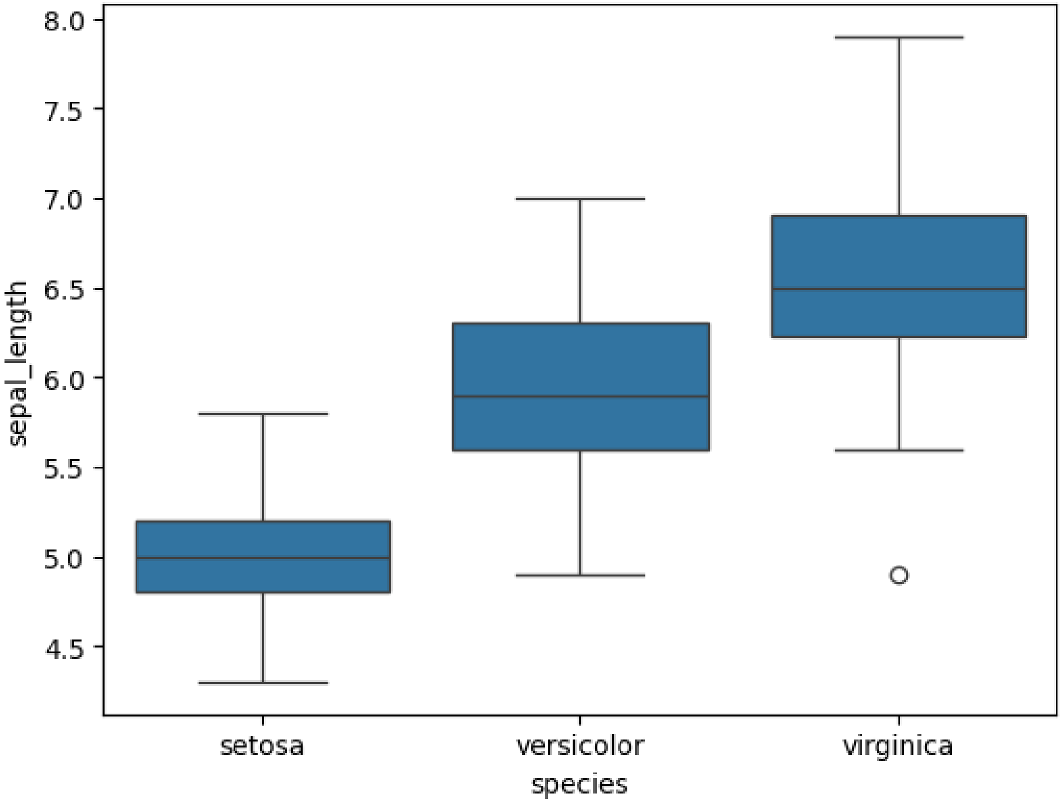
[15]:

```
#The following codes give a scatter plot of
#sepal length vs Sepal width for each type of flower
iset=iris[iris.Classification=="Iris-setosa"]


iver=iris[iris.Classification=="Iris-versicolor"]


ivir=iris[iris.Classification=="Iris-virginica"]


plt.scatter(iset.SL,iset.SW,color="r",label="iris setosa")
plt.scatter(iver.SL,iver.SW,color="b",label="iris versicolor")
plt.scatter(ivir.SL,ivir.SW,color="m",label="iris virginica")
plt.legend()
plt.show()
```

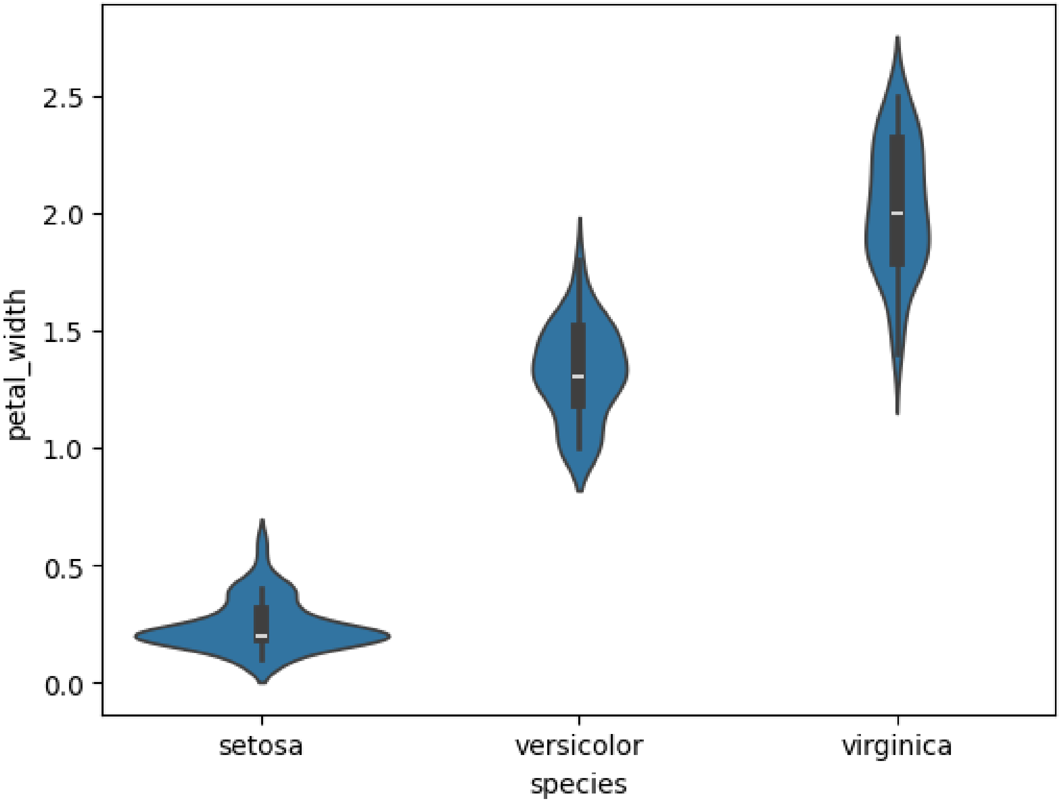


[21]:

```
# Create a boxplot of the 'sepal_length' feature
import seaborn as sns
import matplotlib.pyplot as plt

# Load the Iris dataset
iris = sns.load_dataset('iris')

sns.boxplot(x='species', y='sepal_length', data=iris)

# Show the plot
plt.show()
```



[22]:
```
# Create a violinplot of the 'petal_width' feature
sns.violinplot(x='species', y='petal_width', data=iris)

# Show the plot
plt.show()
```


2.5
2.0
1.5
1.0
0.5
0.0
petal_width
setosa
versicolor
virginica
species