iris-flower-classificaiton-task-1

February 16, 2024

IRIS FLOWER CLASSIFICATION

IMPORTING REQUIRED LIBRARY

```
[ ]: import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
     from sklearn.datasets import load_iris
     from sklearn.model_selection import train_test_split
     from sklearn.preprocessing import StandardScaler
     from sklearn.metrics import classification_report, confusion_matrix
```

READ THE DATA SET

```
[ ]: df = pd.read_csv("/content/sample_data/iris data set.csv")
     df
```

```
[ ]:      sepal_length  sepal_width  petal_length  petal_width         species
     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
     ..            …            …             …            …               …
     145           6.7          3.0           5.2          2.3  Iris-virginica
     146           6.3          2.5           5.0          1.9  Iris-virginica
     147           6.5          3.0           5.2          2.0  Iris-virginica
     148           6.2          3.4           5.4          2.3  Iris-virginica
     149           5.9          3.0           5.1          1.8  Iris-virginica

     [150 rows x 5 columns]
```

```
[ ]: df.head(10)
```

```
[ ]:    sepal_length  sepal_width  petal_length  petal_width      species
     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
5           5.4          3.9           1.7          0.4  Iris-setosa
6           4.6          3.4           1.4          0.3  Iris-setosa
7           5.0          3.4           1.5          0.2  Iris-setosa
8           4.4          2.9           1.4          0.2  Iris-setosa
9           4.9          3.1           1.5          0.1  Iris-setosa
```

[ ]:

```
df.tail(15)
```

[ ]:

```
     sepal_length  sepal_width  petal_length  petal_width         species
135           7.7          3.0           6.1          2.3  Iris-virginica
136           6.3          3.4           5.6          2.4  Iris-virginica
137           6.4          3.1           5.5          1.8  Iris-virginica
138           6.0          3.0           4.8          1.8  Iris-virginica
139           6.9          3.1           5.4          2.1  Iris-virginica
140           6.7          3.1           5.6          2.4  Iris-virginica
141           6.9          3.1           5.1          2.3  Iris-virginica
142           5.8          2.7           5.1          1.9  Iris-virginica
143           6.8          3.2           5.9          2.3  Iris-virginica
144           6.7          3.3           5.7          2.5  Iris-virginica
145           6.7          3.0           5.2          2.3  Iris-virginica
146           6.3          2.5           5.0          1.9  Iris-virginica
147           6.5          3.0           5.2          2.0  Iris-virginica
148           6.2          3.4           5.4          2.3  Iris-virginica
149           5.9          3.0           5.1          1.8  Iris-virginica
```

GETTING INFORMATION ABOUT THE DATA

[ ]:

```
df.shape
```

[ ]:

```
(150, 5)
```

[ ]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   sepal_length  150 non-null    float64
 1   sepal_width   150 non-null    float64
 2   petal_length  150 non-null    float64
 3   petal_width   150 non-null    float64
 4   species       150 non-null    object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
[ ]: df.describe()
```

```
[ ]:        sepal_length  sepal_width  petal_length  petal_width
     count    150.000000   150.000000    150.000000   150.000000
     mean       5.843333     3.054000      3.758667     1.198667
     std        0.828066     0.433594      1.764420     0.763161
     min        4.300000     2.000000      1.000000     0.100000
     25%        5.100000     2.800000      1.600000     0.300000
     50%        5.800000     3.000000      4.350000     1.300000
     75%        6.400000     3.300000      5.100000     1.800000
     max        7.900000     4.400000      6.900000     2.500000
```

CHECK THE MISSING VALUES IN DATA SET

```
[ ]: df.isnull().sum()
```

```
[ ]: sepal_length    0
     sepal_width     0
     petal_length    0
     petal_width     0
     species         0
     dtype: int64
```

CHECKING DUPLICATES

```
[ ]: data = df.drop_duplicates(subset = "species" )
     data
```

```
[ ]:      sepal_length  sepal_width  petal_length  petal_width          species
     0             5.1          3.5           1.4          0.2      Iris-setosa
     50            7.0          3.2           4.7          1.4  Iris-versicolor
     100           6.3          3.3           6.0          2.5   Iris-virginica
```

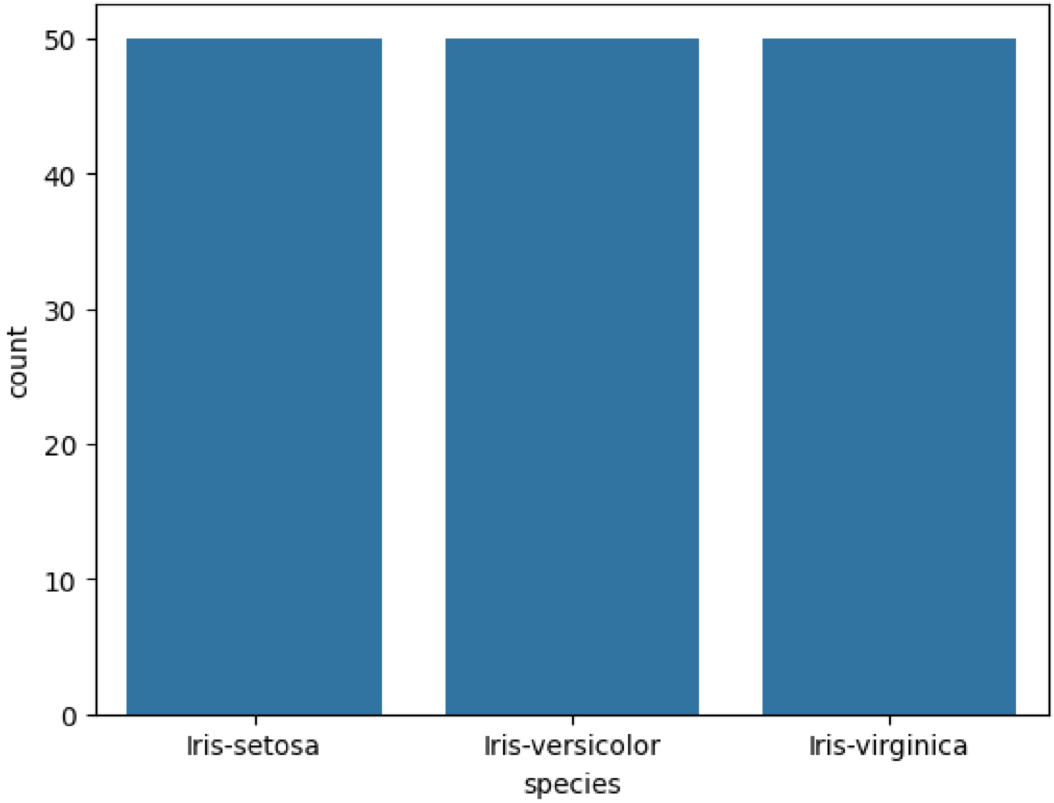
```
[ ]: df.value_counts("species")
```

```
[ ]: species
     Iris-setosa        50
     Iris-versicolor    50
     Iris-virginica     50
     dtype: int64
```

VISUALIZATION OF DATA

COUNT PLOT

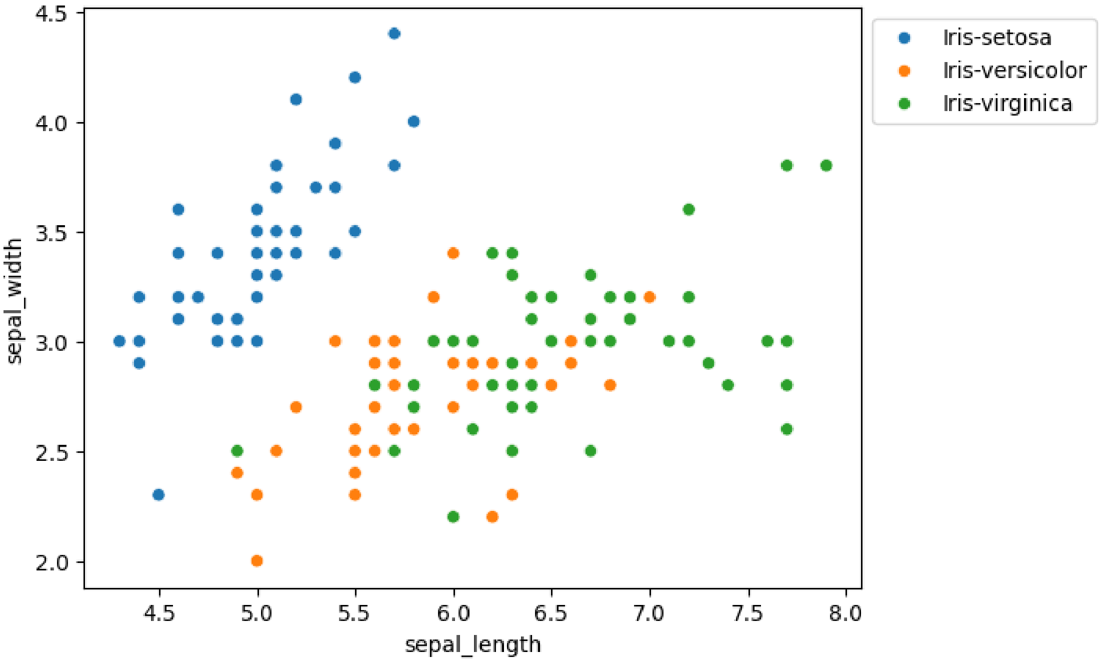
```
[ ]: sns.countplot(x='species', data=df, )
     plt.show()
```



COMPARING SEPAL LENGTH AND WIDTH

SCATTER PLOT

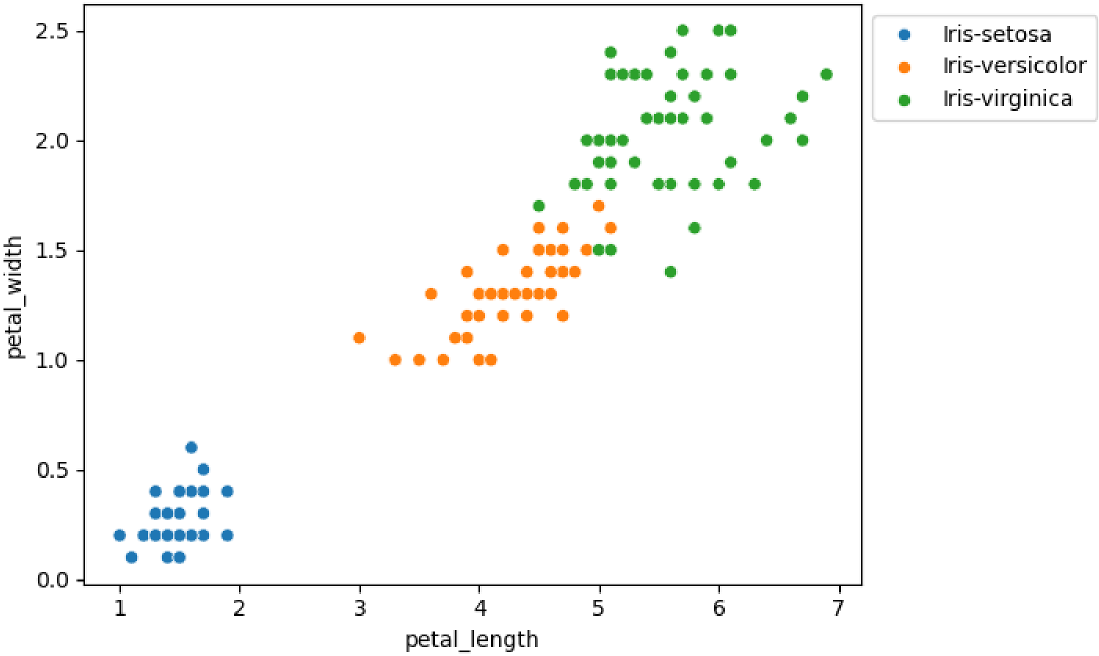
```
[ ]: sns.scatterplot(x="sepal_length", y="sepal_width",hue="species", data=df, )
     plt.legend(bbox_to_anchor=(1, 1), loc=2)
     plt.show()
```



COMPARING PETAL LENGTH AND WIDTH

SCATTER PLOT

[ ]:
```
sns.scatterplot(x="petal_length", y="petal_width",hue="species", data=df, )
plt.legend(bbox_to_anchor=(1, 1), loc=2)
plt.show()
```



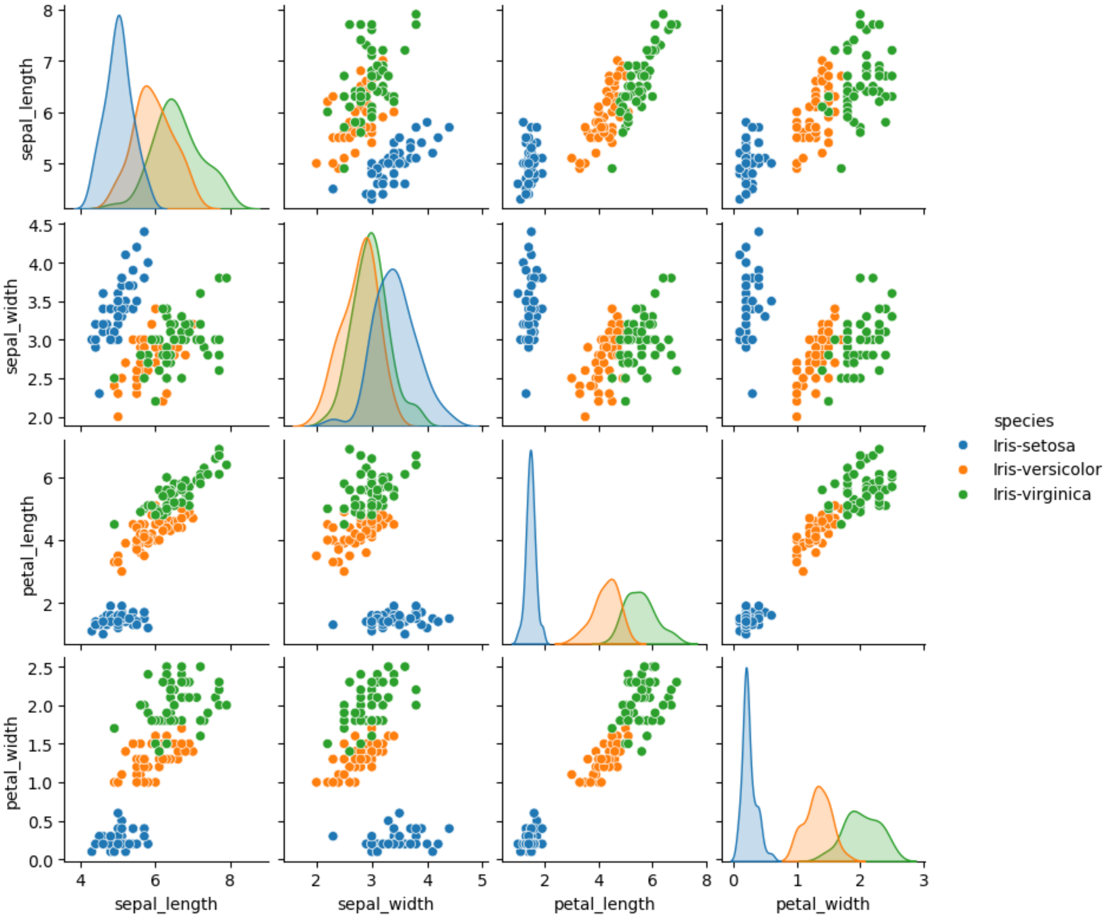
PAIR PLOT

[ ]:

```
sns.pairplot(df.drop([], axis = 1),  hue="species", height=2)
```

[ ]:

```
<seaborn.axisgrid.PairGrid at 0x7dc6348fe5f0>
```

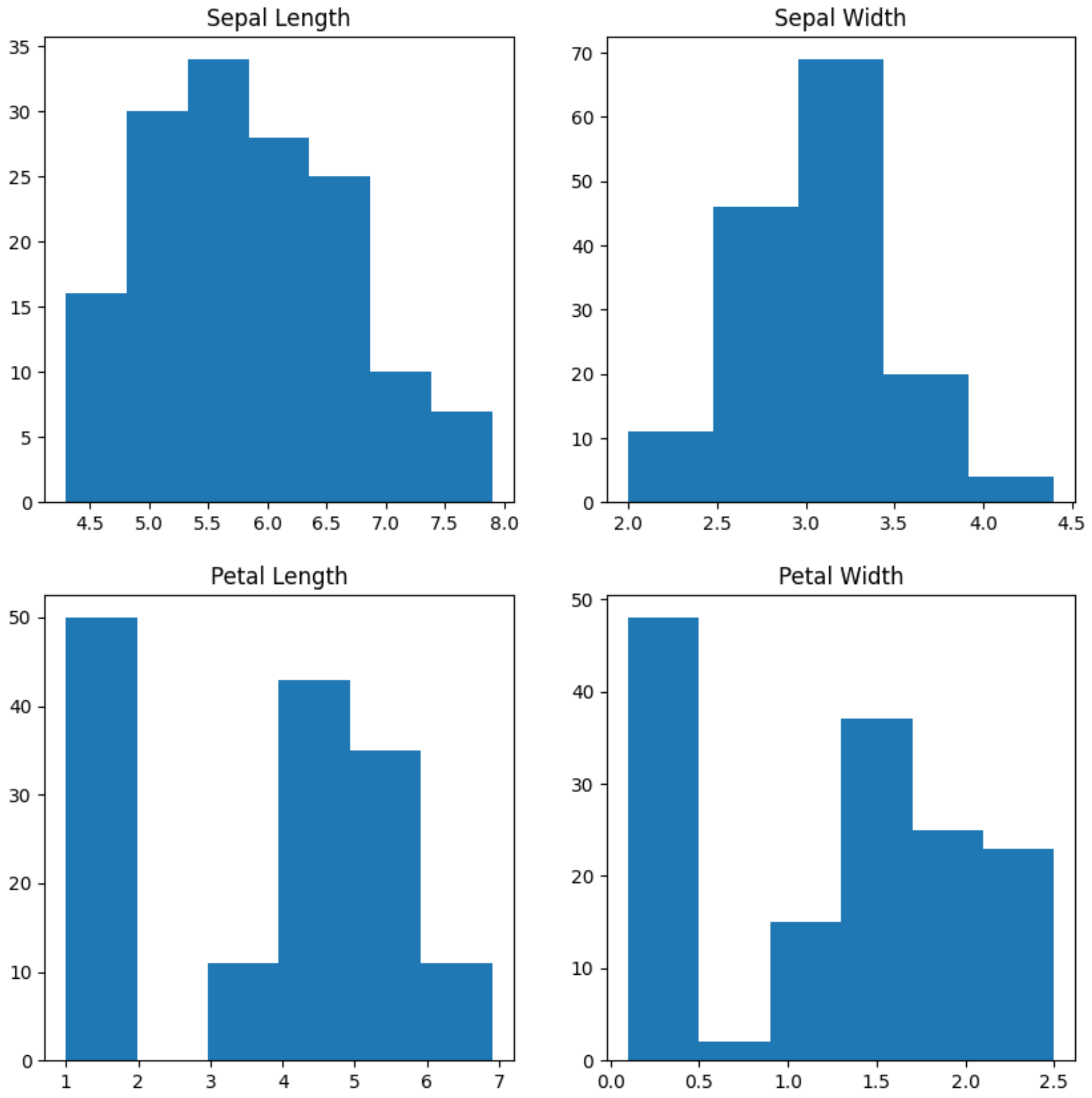


HISTOGRAM

[ ]:

```
fig, axes = plt.subplots(2, 2, figsize=(10,10))
axes[0,0].set_title("Sepal Length")
axes[0,0].hist(df["sepal_length"], bins=7)
axes[0,1].set_title("Sepal Width")
axes[0,1].hist(df["sepal_width"], bins=5)
axes[1,0].set_title("Petal Length")
axes[1,0].hist(df["petal_length"], bins=6)
axes[1,1].set_title("Petal Width")
axes[1,1].hist(df["petal_width"], bins=6)
```

```
[ ]: (array([48.,  2., 15., 37., 25., 23.]),
      array([0.1, 0.5, 0.9, 1.3, 1.7, 2.1, 2.5]),
      <BarContainer object of 6 artists>)
```



HISTOGRAM WITH DISTPLOT

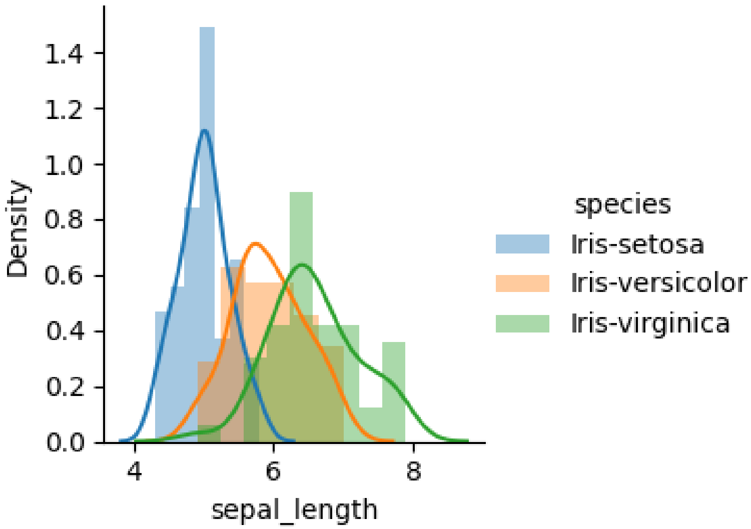
```
[ ]: plot = sns.FacetGrid(df, hue="species")
     plot.map(sns.distplot, "sepal_length").add_legend()
     plot = sns.FacetGrid(df, hue="species")
     plot.map(sns.distplot, "sepal_width").add_legend()
     plot = sns.FacetGrid(df, hue="species")
     plot.map(sns.distplot, "petal_length").add_legend()
     plot = sns.FacetGrid(df, hue="species")
```

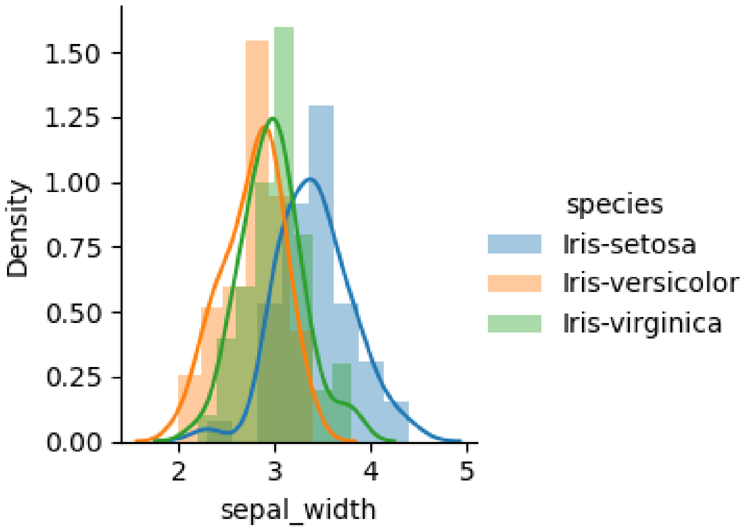
```python
plot.map(sns.distplot, "petal_width").add_legend()
plt.show()
```

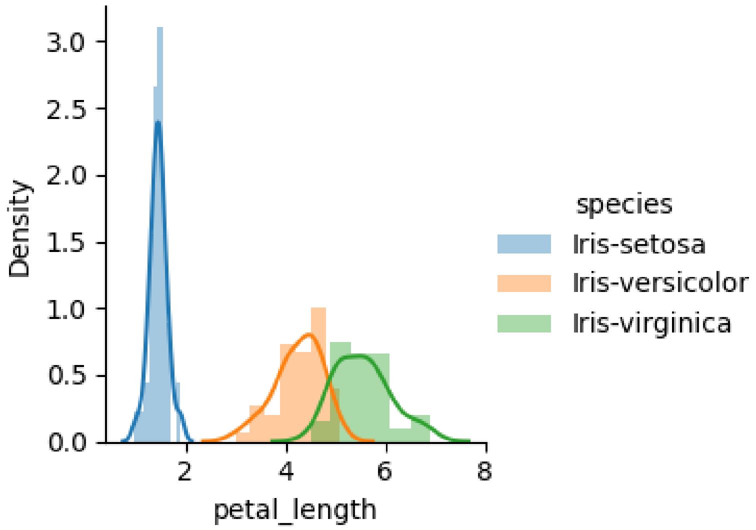
```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```

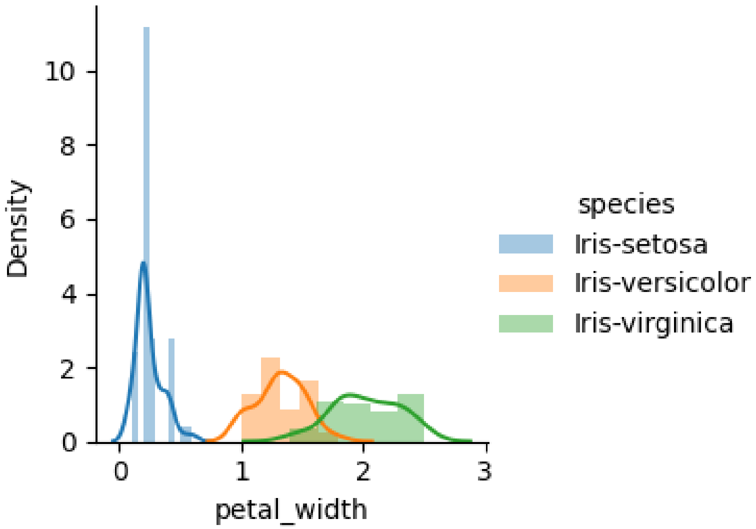
```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
```

```
Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:854: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
```


1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Density
4
6
8
sepal_length
species
Iris-setosa
Iris-versicolor
Iris-virginica


1.50
1.25
1.00
0.75
0.50
0.25
0.00
Density
2
3
4
5
sepal_width
species
Iris-setosa
Iris-versicolor
Iris-virginica





HANDLING CORRELATION

[ ]:

```
data.corr(method='pearson')
```

```
<ipython-input-32-c50c7eb58c83>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  data.corr(method='pearson')
```

[ ]:

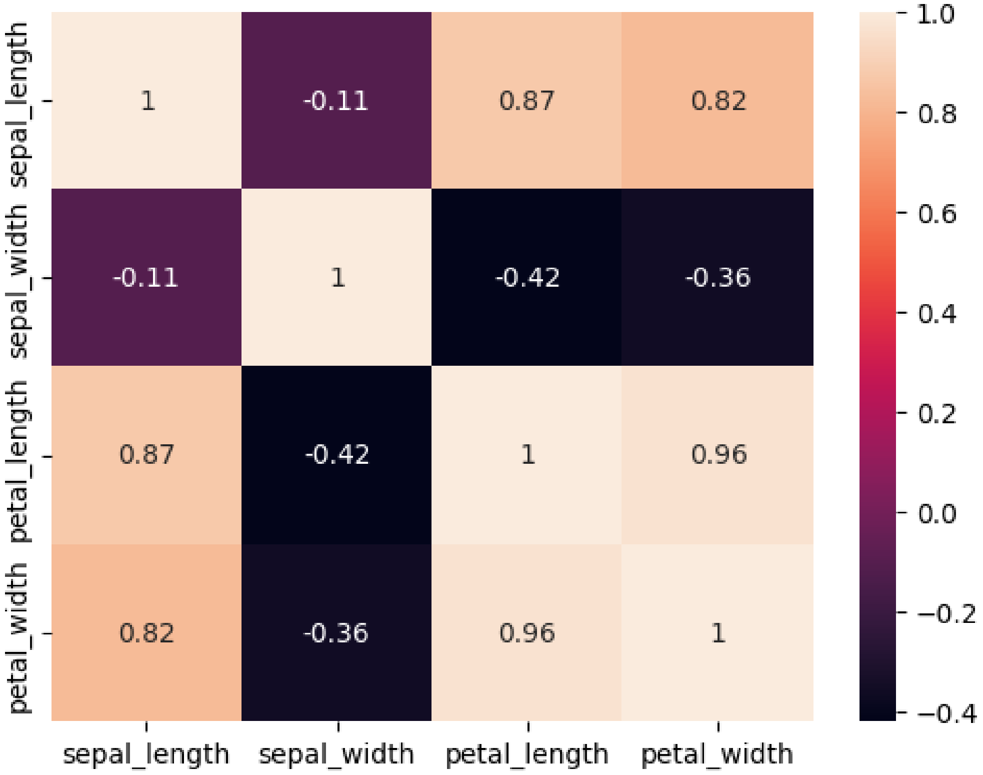
```
              sepal_length  sepal_width  petal_length  petal_width
sepal_length      1.000000    -0.999226      0.795795     0.643817
sepal_width      -0.999226     1.000000     -0.818999    -0.673417
petal_length      0.795795    -0.818999      1.000000     0.975713
petal_width       0.643817    -0.673417      0.975713     1.000000
```

HEAT MAP

[ ]:

```
sns.heatmap(df.corr(method='pearson').drop([], axis=1).drop([], axis=0),
            annot = True);
plt.show()
```

```
<ipython-input-38-e16c3bd8be60>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(df.corr(method='pearson').drop(
```

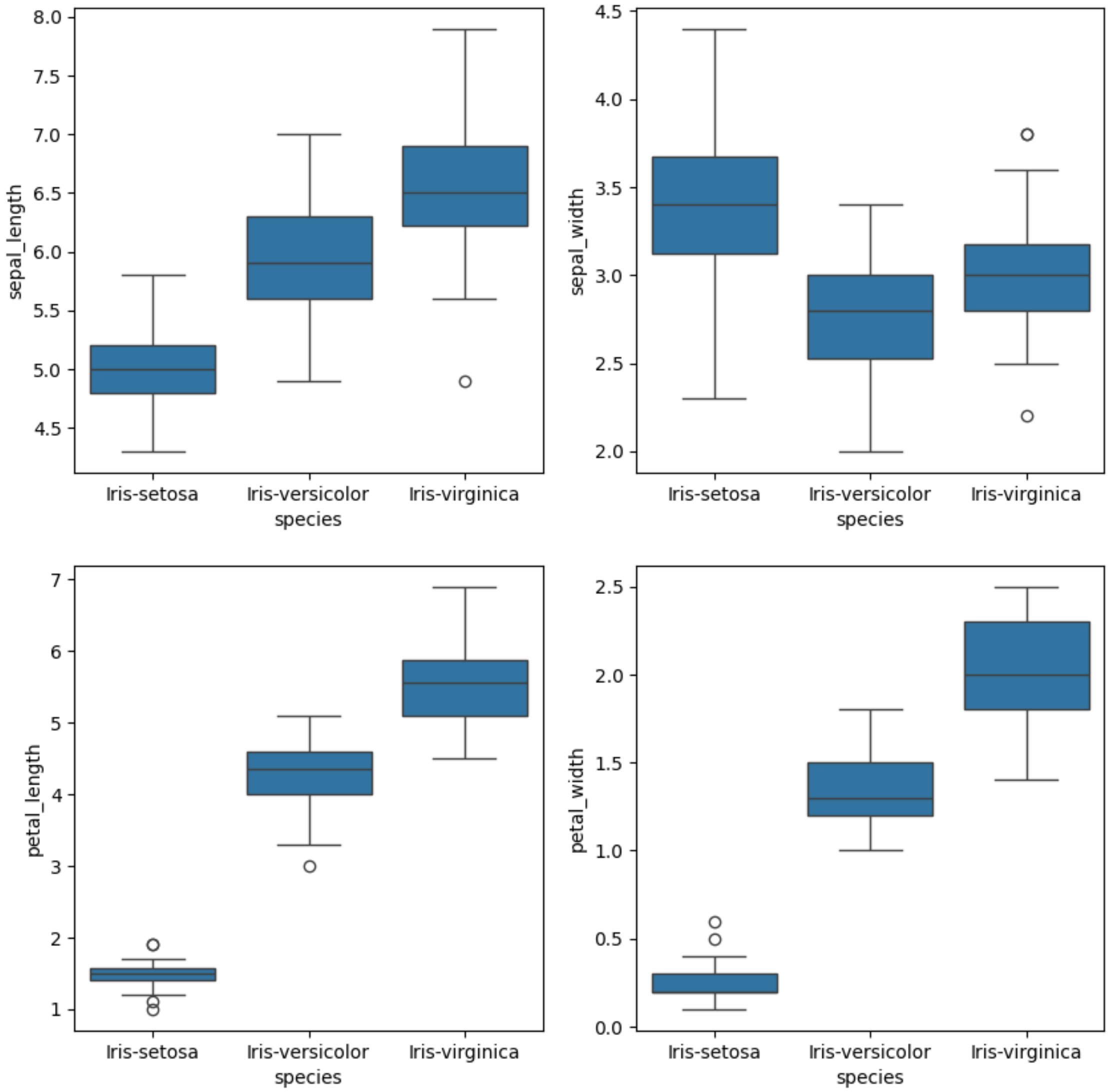


BOX PLOT

[ ]:

```
def graph(y):
    sns.boxplot(x="species", y=y, data=df)
plt.figure(figsize=(10,10))
```

```python
plt.subplot(221)
graph("sepal_length")
plt.subplot(222)
graph("sepal_width")
plt.subplot(223)
graph("petal_length")
plt.subplot(224)
graph("petal_width")
plt.show()
```



REMOVING OUTLIERS

[ ]:

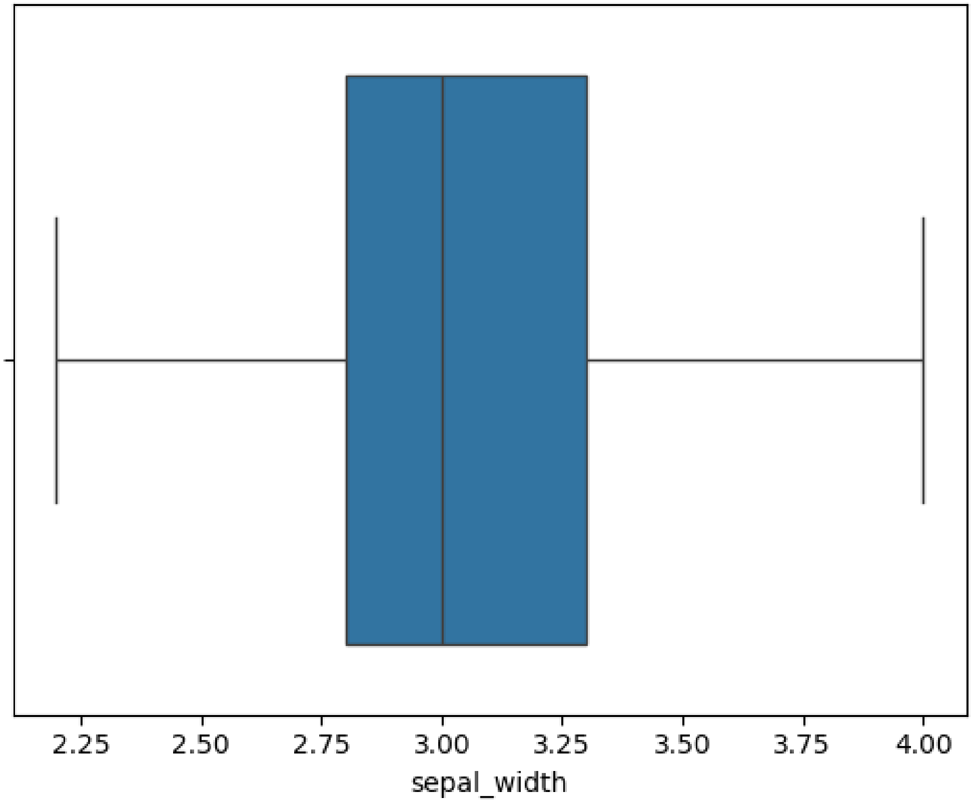
```
Q1 = np.percentile(df["sepal_width"], 25,
                interpolation = "midpoint")
Q3 = np.percentile(df["sepal_width"], 75,
                interpolation = "midpoint")
IQR = Q3 - Q1
print("Old Shape: ", df.shape)
# Upper bound
upper = np.where(df["sepal_width"] >= (Q3+1.5*IQR))
# Lower bound
lower = np.where(df["sepal_width"] <= (Q1-1.5*IQR))
# Removing the Outliers
df.drop(upper[0], inplace = True)
df.drop(lower[0], inplace = True)
print("New Shape: ", df.shape)
sns.boxplot(x="sepal_width", data=df)
```

```
Old Shape:  (150, 5)
New Shape:  (146, 5)
```

```
<ipython-input-41-836e3c8e96a4>:9: DeprecationWarning: the `interpolation=`
argument to percentile was renamed to `method=`, which has additional options.
Users of the modes 'nearest', 'lower', 'higher', or 'midpoint' are encouraged to
review the method they used. (Deprecated NumPy 1.22)
  Q1 = np.percentile(df["sepal_width"], 25,
<ipython-input-41-836e3c8e96a4>:11: DeprecationWarning: the `interpolation=`
argument to percentile was renamed to `method=`, which has additional options.
Users of the modes 'nearest', 'lower', 'higher', or 'midpoint' are encouraged to
review the method they used. (Deprecated NumPy 1.22)
  Q3 = np.percentile(df["sepal_width"], 75,
```

[ ]:

```
<Axes: xlabel='sepal_width'>
```


2.25
2.50
2.75
3.00
3.25
3.50
3.75
4.00
sepal_width