Project 1: Iris Dataset Basic Analysis

Exploratory Data Analysis (EDA) with Python:

Tools Used:

Pandas: The code features the use of pandas, a powerful data analysis and manipulation tool, used for loading and manipulating the Iris dataset.

Numpy: Numpy appears to be utilized for numerical computations and handling multidimensional arrays.

Seaborn: Seaborn is used for data visualization

Jupyter notebook for python code

EDA Process:

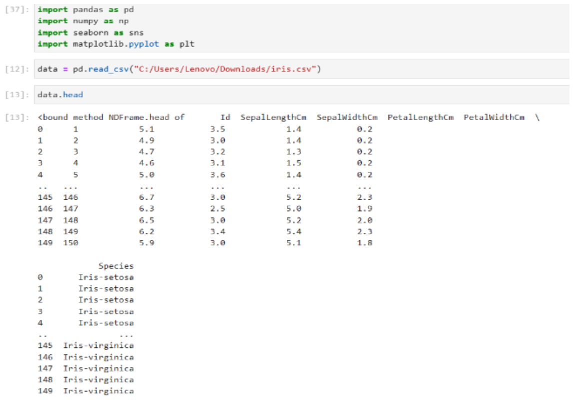
Data Loading: The code loads the Iris dataset from a specified file path using Pandas' read_csv function, creating a DataFrame.

Data Description: The .describe() method displays key statistical measures for each numeric column in the dataset, offering insight into the central tendencies and distribution of the data.

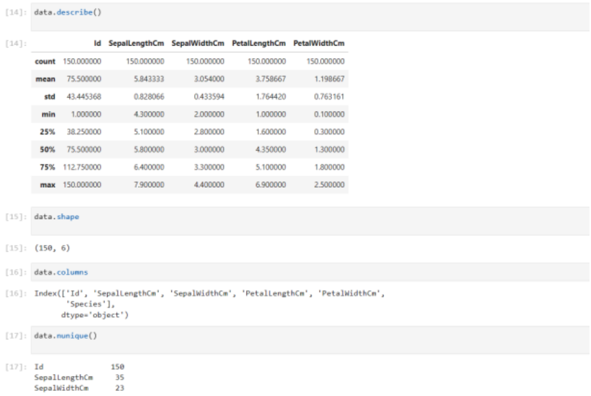
Data Shape and Columns: Output provides details on the shape of the dataset (number of rows and columns) and lists the column names present in the dataset.

Data Unique Values: The nunique() function calculates the number of unique values in each column of the dataset, offering insight into the diversity of values within each column.

Data Visualization: The inclusion of Seaborn's displot function shows a histogram for the'sepal length' column in the Iris dataset, which provides a visual depiction of the feature's distribution.



```
[37]: import pandas as pd
      import numpy as np
      import seaborn as sns
      import matplotlib.pyplot as plt

[12]: data = pd.read_csv("C:/Users/Lenovo/Downloads/iris.csv")

[13]: data.head

[13]: <bound method NDFrame.head of       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
```



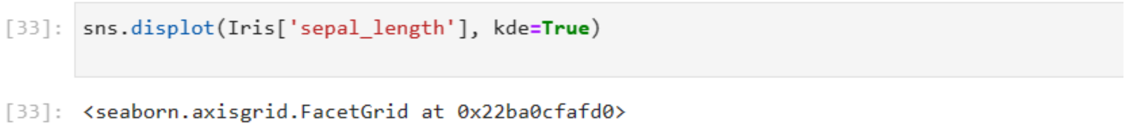
```
[14]: data.describe()
```

[14]:

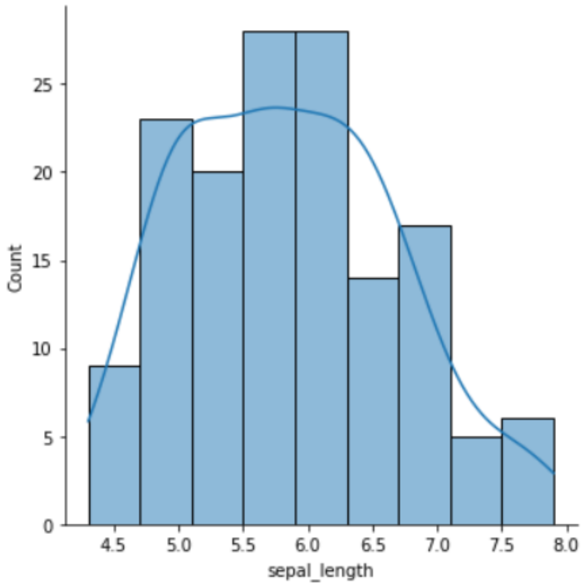
	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	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

```
[15]: data.shape

[15]: (150, 6)

[16]: data.columns

[16]: Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
             'Species'],
            dtype='object')

[17]: data.nunique()

[17]: Id               150
      SepalLengthCm     35
      SepalWidthCm      23
```

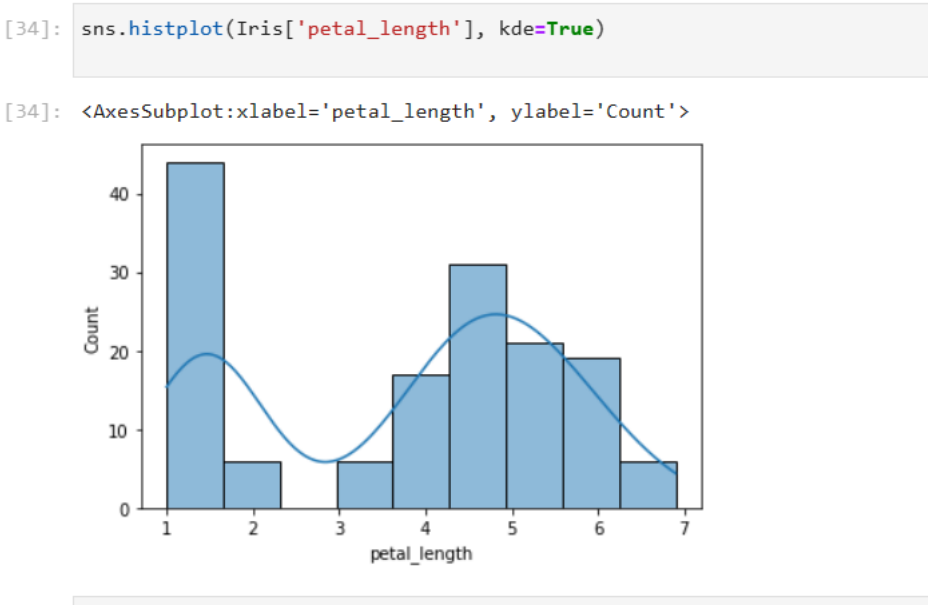
Visualizing statistics and Distribution :



```
[33]: sns.displot(Iris['sepal_length'], kde=True)

[33]: <seaborn.axisgrid.FacetGrid at 0x22ba0cfafd0>
```



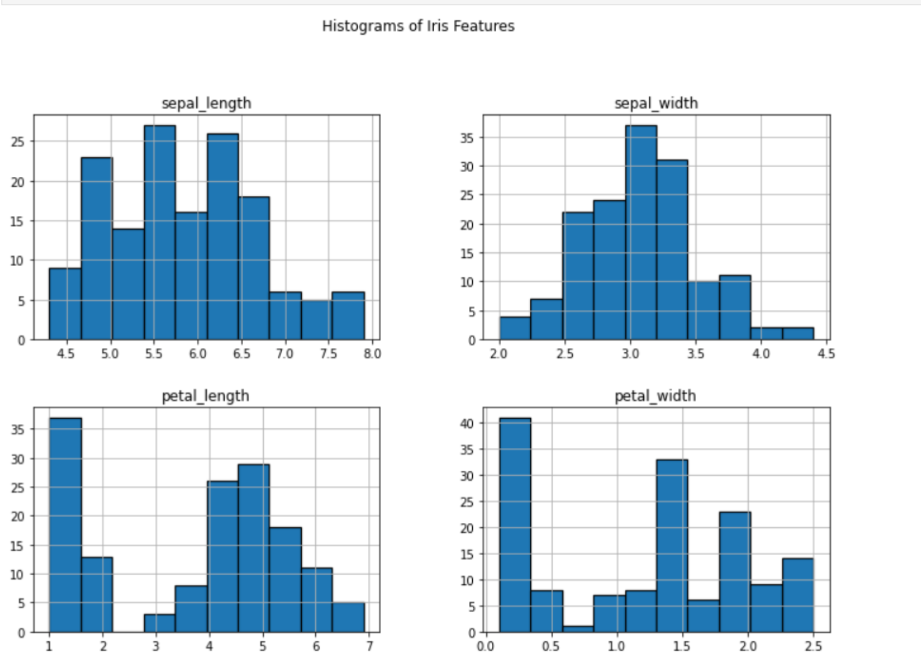
The "sepal length" variable from the Iris dataset appears to be distributed according to the seaborn displot graph. We can see the distribution of sepal length over a range of values from the graph. The sepal length is most commonly represented by the x-axis, and the frequency or density of each sepal length value is most typically displayed on the y-axis. The plot's peaks and valleys can provide information on the sepal length distribution pattern throughout the dataset.



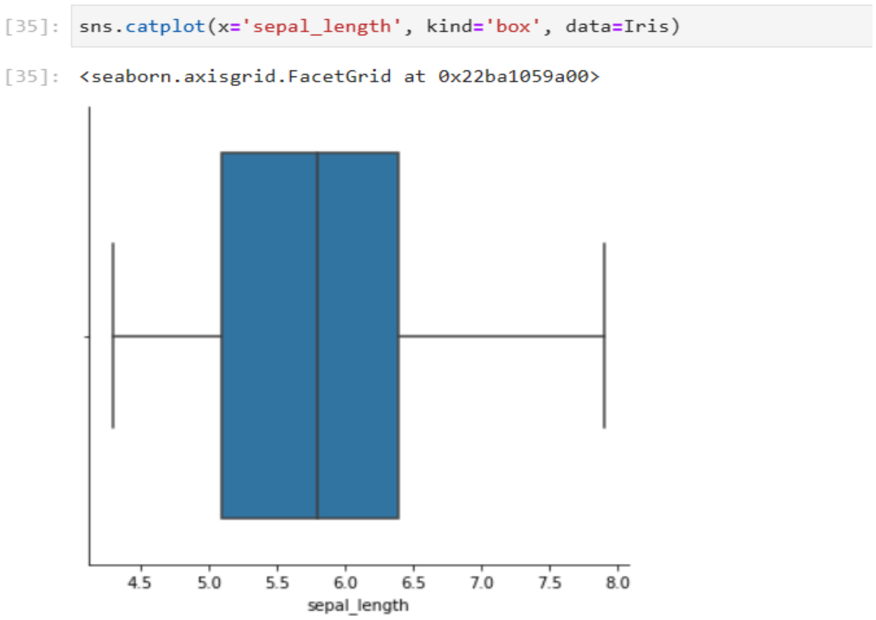
```
[34]: sns.histplot(Iris['petal_length'], kde=True)
```

```
[34]: <AxesSubplot:xlabel='petal_length', ylabel='Count'>
```

A seaborn histplot showing the "petal length" variable's distribution from the Iris dataset. We can see the distribution of petal length over a range of values from the graph. The petal length is represented by the x-axis, and the count or density of each petal length value is probably displayed on the y-axis. The smoothed distribution of the petal length data is suggested by the existence of the KDE (Kernel Density Estimation) curve. The distribution pattern of petal length within the sample is revealed by this graphic. With the Iris dataset, it enables us to comprehend the variety of petal lengths and their frequencies. It also draws attention to any possible patterns or clusters within the petal length measurements.



The four features—petal length, petal width, sepal length, and sepal width—from the Iris dataset seem to be histograms in the picture.



```
[35]: sns.catplot(x='sepal_length', kind='box', data=Iris)
```

```
[35]: <seaborn.axisgrid.FacetGrid at 0x22ba1059a00>
```

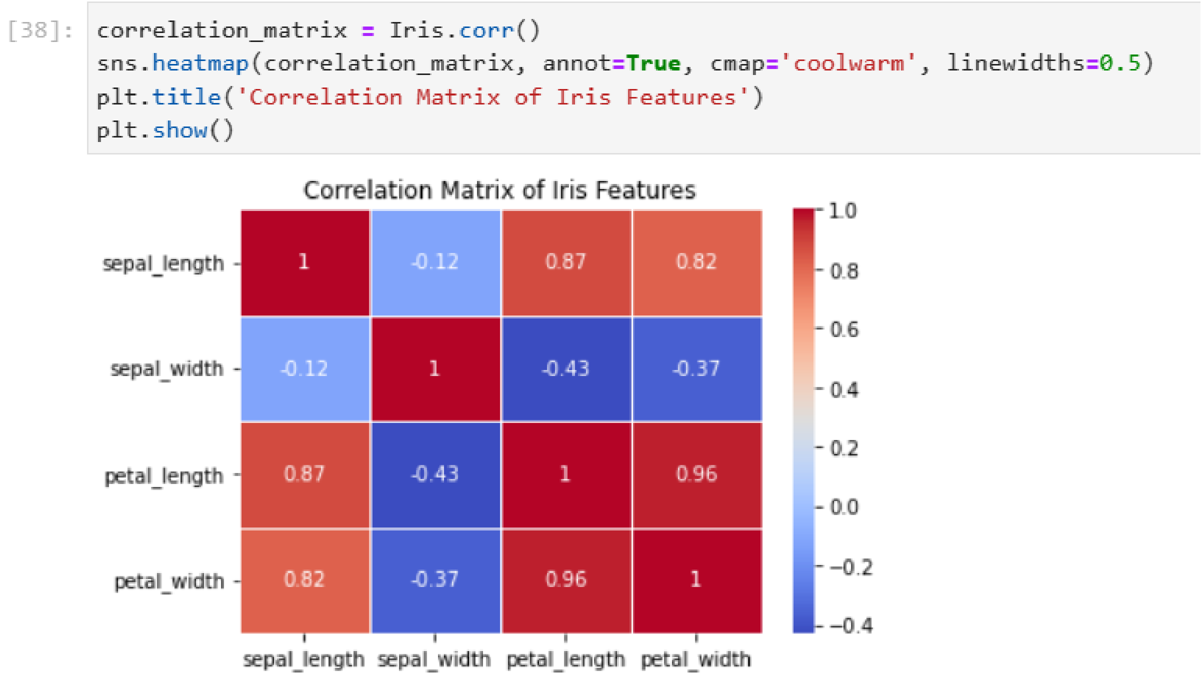
A seaborn catplot with a box plot showing the distribution of the "sepal length" variable from the Iris dataset.

Central Tendency: The box plot shows the median or middle value of sepal length, as depicted by the line inside the box. The length of the box represents the interquartile range, which is the spread of the middle 50% of the data.

Variability: The length of the whiskers on each side of the box represents the range of sepal length measurements. Any potential outliers outside of this range are represented as separate points on the plot.

Distribution: The box plot allows us to examine the distribution of sepal length and detect any asymmetry or grouping in the data.

Comparison: If the catplot has hue or column characteristics, it may represent numerous box plots, allowing for the comparison of sepal length distributions across categories.



```
[38]: correlation_matrix = Iris.corr()
      sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
      plt.title('Correlation Matrix of Iris Features')
      plt.show()
```

The figure displays a correlation matrix of the characteristics from the Iris dataset, namely the length, width, length, and width of the petals. A heatmap is used to illustrate the matrix, with each cell showing the association between two characteristics.

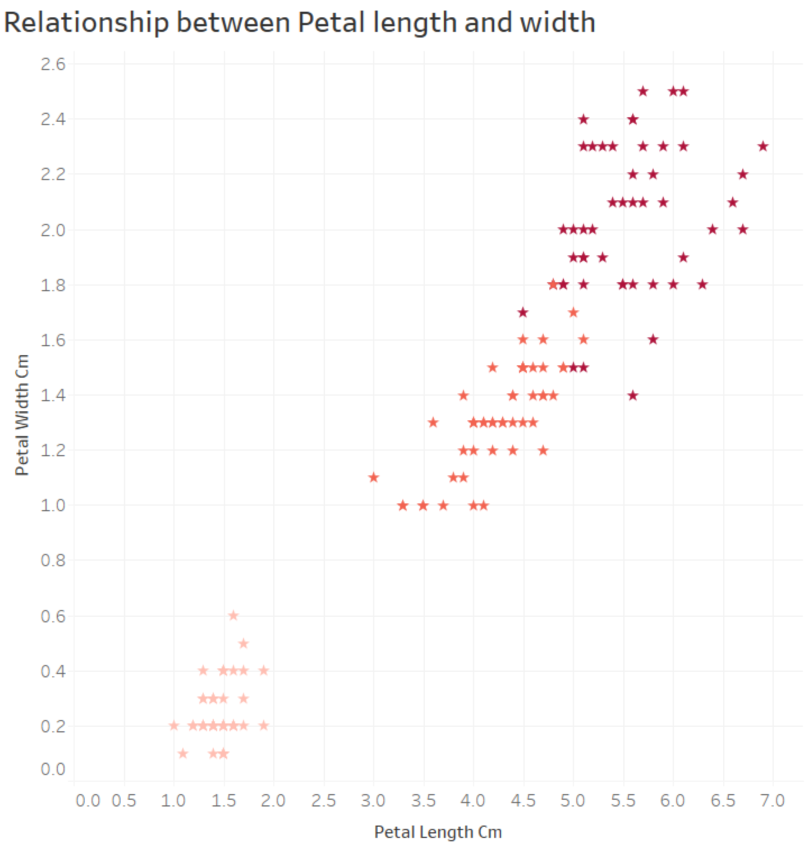
Findings from the heatmap of the correlation matrix:

Correlation Strength and Direction: The heatmap's color gradient shows the correlation between the characteristics' strengths and directions. Lighter hues indicate positive correlations, whereas darker shades suggest negative correlations. The correlation's strength is shown by the shade's intensity.

Feature Relationships: We can understand the numerical correlation coefficient between the characteristics by looking at the annotated values in each cell. This tells us how closely variations in one trait are related to variations in another one.

Data Visualization with Power BI or Tableau:

I used Tableau for iris dataset



Several conclusions may be drawn from the above picture of the link between petal length and width:

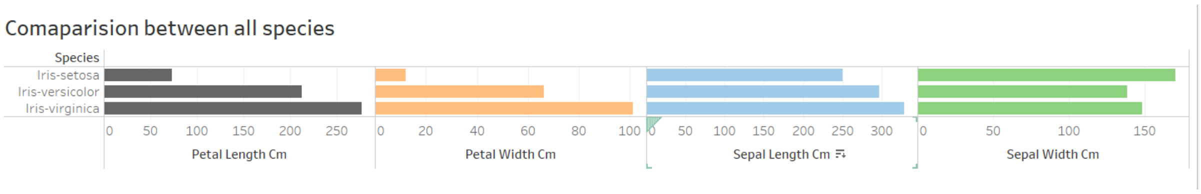
Positive Correlation: The scatter plot shows that the length and width of the petals are positively correlated. Petal breadth often rises in tandem with petal length. This implies that the two variables have a linear connection.

Cluster Analysis: Different clusters are visible on the figure, suggesting that the dataset may contain subclasses or other possible categories. This implies that some petal length and breadth ranges can be more common than others.

Outliers: A small number of data points seem to differ noticeably from the overall pattern, indicating the possibility of outliers. These anomalies can have an impact on the analysis and call for more research.

Range and Distribution: The figure helps to visualize the dataset's petal length and breadth values' range and distribution. The distribution of data points on the plot can provide information about the density and variability of the values.

Model Considerations: Building models to predict or classify based on these variables requires knowledge about the observed connection between petal length and breadth, which can be essential for predictive modeling and classification.



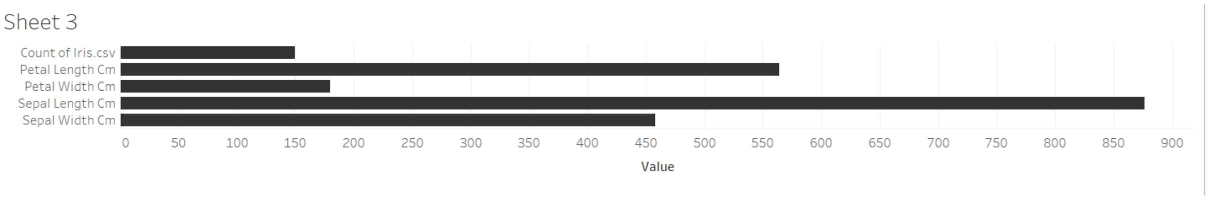
It seems to show a comparison of three species—Iris virginica, Iris versicolor, and Iris setosa—found in the Iris dataset. Four characteristics—petal length, petal breadth, sepal length, and sepal width—are probably the basis for the comparison.

Distribution: We may deduce the distribution and spread of the four traits inside each species by examining the displayed dots for each species. This can provide insight into the differences and unique traits of every species with regard to the size of its petals and sepals.

Species Differentiation: Based on the measured traits, the graphic most likely illustrates how the various Iris species are identified from one another. This can offer important information regarding the characteristics' ability to discriminate when it comes to categorizing the Iris species.

Feature Relationships: An understanding of the links between these features and how they could support species distinction can be gained by looking at how each species' data points are distributed throughout the four features.

Outliers: Any outliers that may be present in the plotted data may also be found and recognized. This information can be useful in determining the variability and possible uniqueness of individual specimens within each species.



A bar chart or count plot that depicts the count of a categorical variable in the "Iris.csv" dataset across multiple categories. Petal length, petal width, sepal length, and sepal width appear to be connected to the categories.

Frequency Distribution: The figure shows how frequently various categories appear in the dataset, either by count or by frequency. This might highlight any common or uncommon values and offer insights into the categorical variable's distribution.

Comparison of Categories: We can compare the frequency of each category in the dataset by looking at the lengths of the bars for each category. This makes it possible to identify any groups that are overrepresented or prominent.

Data Value Distribution: The number of occurrences for each category is shown on the y-axis, whilst the x-axis most usually shows the value or range of the categories. This makes it possible to comprehend the distribution and density of the values for the category variable.

Data Completeness and Quality: Regarding the petal and sepal dimensions, the chart can also provide information on the accuracy and completeness of the data