an7skjbyq

February 17, 2026

Task 01: Exploring and Visualizing a Simple Dataset.

Objective: Understand how to read, summarize and visualize a dataset.

Dataset: Iris Dataset(CSV format, available through seaborn or other open source). About Dataset: The Iris Dataset consists of 150 iris samples, each having four numerical features: sepal length, sepal width, petal length and petal width. Each sample is categorized into one of three iris species: Setosa, Versicolor and Virginica. This dataset is widely used as a sample dataset in machine learning and statistics due to its simple and easily understandable structure.

Feature Information: Sepal Length (cm), Sepal Width (cm), Petal Length (cm), Petal Width (cm).

Target Information: Iris Species: Setosa, Versicolor, Virginica.

Source: The Iris Dataset is obtained from the scikit-learn (sklearn) library under the BSD (Berkeley Software Distribution) license.

File Formats: CSV (Comma-Separated Values): CSV format is the most commonly used and easily readable format. Each row represents one sample with its features separated by commas.

Instructions:

Load the dataset using the pandas library.

[1]:
```
import pandas as pd
```

[2]:
```
dataset = pd.read_csv("csv_data_iris.csv")
```

Display dataset structure using .shape, .columns and .head().

[3]:
```
dataset.head()
```

[3]:
```
   sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  \
0                5.1               3.5                1.4               0.2
1                4.9               3.0                1.4               0.2
2                4.7               3.2                1.3               0.2
3                4.6               3.1                1.5               0.2
4                5.0               3.6                1.4               0.2

   target   class
0       0  setosa
1       0  setosa
```

```
2       0  setosa
3       0  setosa
4       0  setosa
```

[4]:
```
print("Print first 05 rows of the dataset:", dataset.head(5))
```

```
Print first 05 rows of the dataset:    sepal length (cm)  sepal width (cm)
petal length (cm)  petal width (cm)  \
0                5.1               3.5                1.4               0.2
1                4.9               3.0                1.4               0.2
2                4.7               3.2                1.3               0.2
3                4.6               3.1                1.5               0.2
4                5.0               3.6                1.4               0.2

   target   class
0       0  setosa
1       0  setosa
2       0  setosa
3       0  setosa
4       0  setosa
```

[5]:
```
dataset.tail()
```

[5]:
```
     sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  \
145                6.7               3.0                5.2               2.3
146                6.3               2.5                5.0               1.9
147                6.5               3.0                5.2               2.0
148                6.2               3.4                5.4               2.3
149                5.9               3.0                5.1               1.8

     target      class
145       2  virginica
146       2  virginica
147       2  virginica
148       2  virginica
149       2  virginica
```

[6]:
```
print("Print last 10 rows of the dataset:",dataset.tail(10))
```

```
Print last 10 rows of the dataset:      sepal length (cm)  sepal width (cm)
petal length (cm)  petal width (cm)  \
140                6.7               3.1                5.6               2.4
141                6.9               3.1                5.1               2.3
142                5.8               2.7                5.1               1.9
143                6.8               3.2                5.9               2.3
144                6.7               3.3                5.7               2.5
145                6.7               3.0                5.2               2.3
146                6.3               2.5                5.0               1.9
```

```
147                6.5               3.0                5.2               2.0
148                6.2               3.4                5.4               2.3
149                5.9               3.0                5.1               1.8

     target      class
140       2  virginica
141       2  virginica
142       2  virginica
143       2  virginica
144       2  virginica
145       2  virginica
146       2  virginica
147       2  virginica
148       2  virginica
149       2  virginica
```

[7]:
```
dataset.shape
```

[7]:
```
(150, 6)
```

[8]:
```
print("The shape of the dataset is:",dataset.shape)
```

```
The shape of the dataset is: (150, 6)
```

[9]:
```
dataset.columns
```

[9]:
```
Index(['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)',
       'petal width (cm)', 'target', 'class'],
      dtype='object')
```

[10]:
```
print("The name of columns of the dataset along with its units of measurment is:
  ↪",dataset.columns)
```

```
The name of columns of the dataset along with its units of measurment is:
Index(['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)',
       'petal width (cm)', 'target', 'class'],
     dtype='object')
```

[11]:
```
dataset.describe
```

[11]:
```
<bound method NDFrame.describe of      sepal length (cm)  sepal width (cm)
petal length (cm)  petal width (cm)  \
0                  5.1               3.5                1.4               0.2
1                  4.9               3.0                1.4               0.2
2                  4.7               3.2                1.3               0.2
3                  4.6               3.1                1.5               0.2
4                  5.0               3.6                1.4               0.2
..                 …                 …                  …                 …
```

```
145                6.7               3.0                5.2               2.3
146                6.3               2.5                5.0               1.9
147                6.5               3.0                5.2               2.0
148                6.2               3.4                5.4               2.3
149                5.9               3.0                5.1               1.8

     target      class
0         0     setosa
1         0     setosa
2         0     setosa
3         0     setosa
4         0     setosa
..      …         …
145       2  virginica
146       2  virginica
147       2  virginica
148       2  virginica
149       2  virginica

[150 rows x 6 columns]>
```

[12]:
```
print("The general description of the dataset are as follows:",dataset.describe)
```

```
The general description of the dataset are as follows: <bound method
NDFrame.describe of      sepal length (cm)  sepal width (cm)  petal length (cm)
petal width (cm)  \
0                  5.1               3.5                1.4               0.2
1                  4.9               3.0                1.4               0.2
2                  4.7               3.2                1.3               0.2
3                  4.6               3.1                1.5               0.2
4                  5.0               3.6                1.4               0.2
..                 …                 …                  …                 …
145                6.7               3.0                5.2               2.3
146                6.3               2.5                5.0               1.9
147                6.5               3.0                5.2               2.0
148                6.2               3.4                5.4               2.3
149                5.9               3.0                5.1               1.8

     target      class
0         0     setosa
1         0     setosa
2         0     setosa
3         0     setosa
4         0     setosa
..      …         …
145       2  virginica
146       2  virginica
```

```
147       2  virginica
148       2  virginica
149       2  virginica

[150 rows x 6 columns]>
```

[13]:
```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   sepal length (cm)  150 non-null    float64
 1   sepal width (cm)   150 non-null    float64
 2   petal length (cm)  150 non-null    float64
 3   petal width (cm)   150 non-null    float64
 4   target             150 non-null    int64
 5   class              150 non-null    object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

[14]:
```
print("The information of the dataset are as follows:",dataset.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   sepal length (cm)  150 non-null    float64
 1   sepal width (cm)   150 non-null    float64
 2   petal length (cm)  150 non-null    float64
 3   petal width (cm)   150 non-null    float64
 4   target             150 non-null    int64
 5   class              150 non-null    object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
The information of the dataset are as follows: None
```

[15]:
```
dataset.dtypes
```

[15]:
```
sepal length (cm)    float64
sepal width (cm)     float64
petal length (cm)    float64
petal width (cm)     float64
target                 int64
class                 object
dtype: object
```

[16]:
```
print("The datatypes of the columns of the dataset are as follows:",dataset.
  ↪dtypes)
```

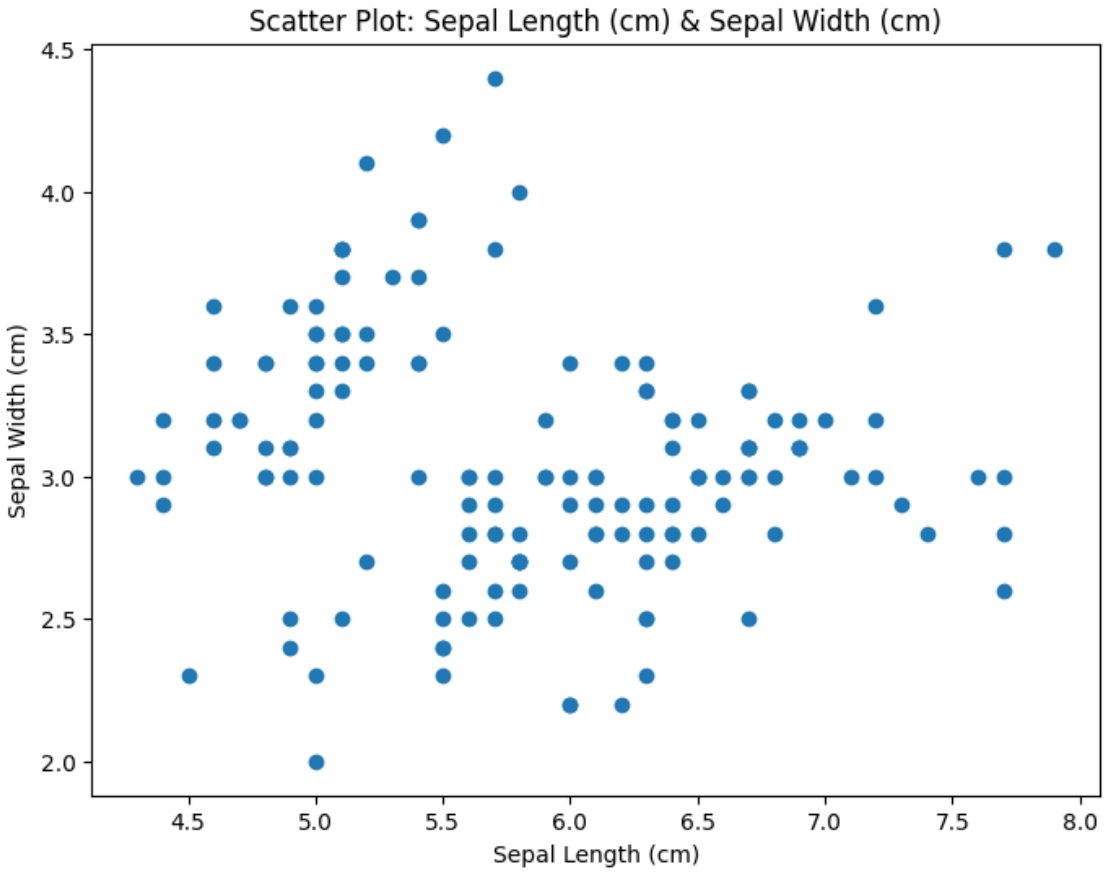
```
The datatypes of the columns of the dataset are as follows: sepal length (cm)
float64
sepal width (cm)     float64
petal length (cm)    float64
petal width (cm)     float64
target                 int64
class                 object
dtype: object
```

Use Matplotlib and Seaborn for visualizations.

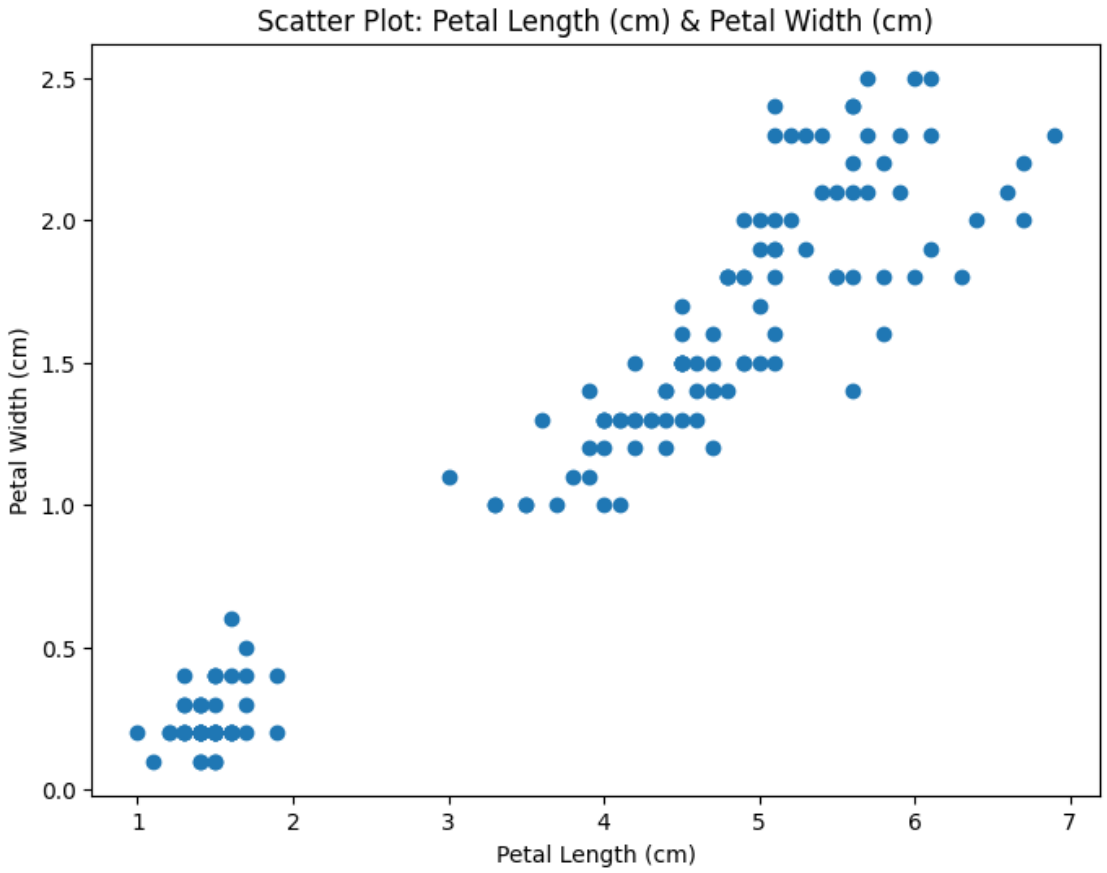
Create basic visualizations: Scatter Plot to analyze the relationships between variables.

[17]:
```
import matplotlib as pyplot
from matplotlib import pyplot as plt
```

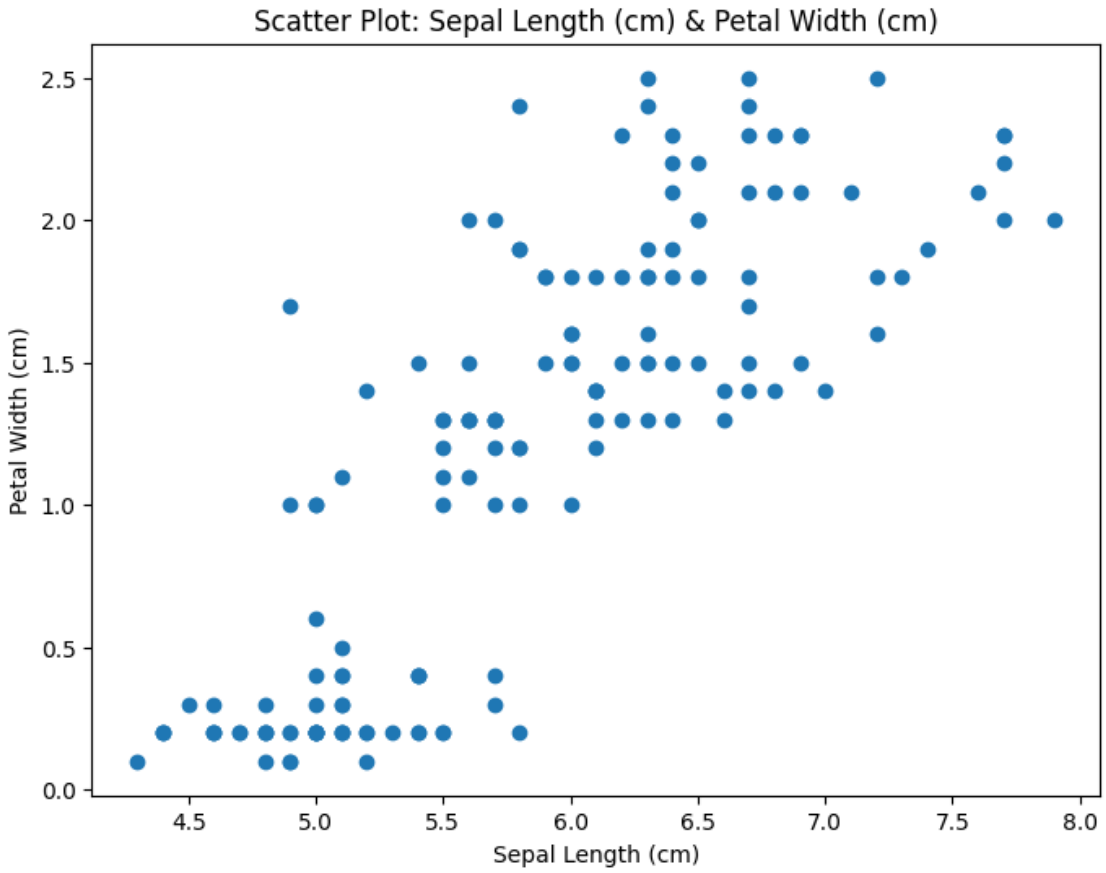
[18]:
```
plt.figure(figsize=(8,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.scatter(dataset["sepal length (cm)"], dataset["sepal width (cm)"])

plt.title("Scatter Plot: Sepal Length (cm) & Sepal Width (cm)")
plt.xlabel("Sepal Length (cm)")
plt.ylabel("Sepal Width (cm)")

plt.show()
```



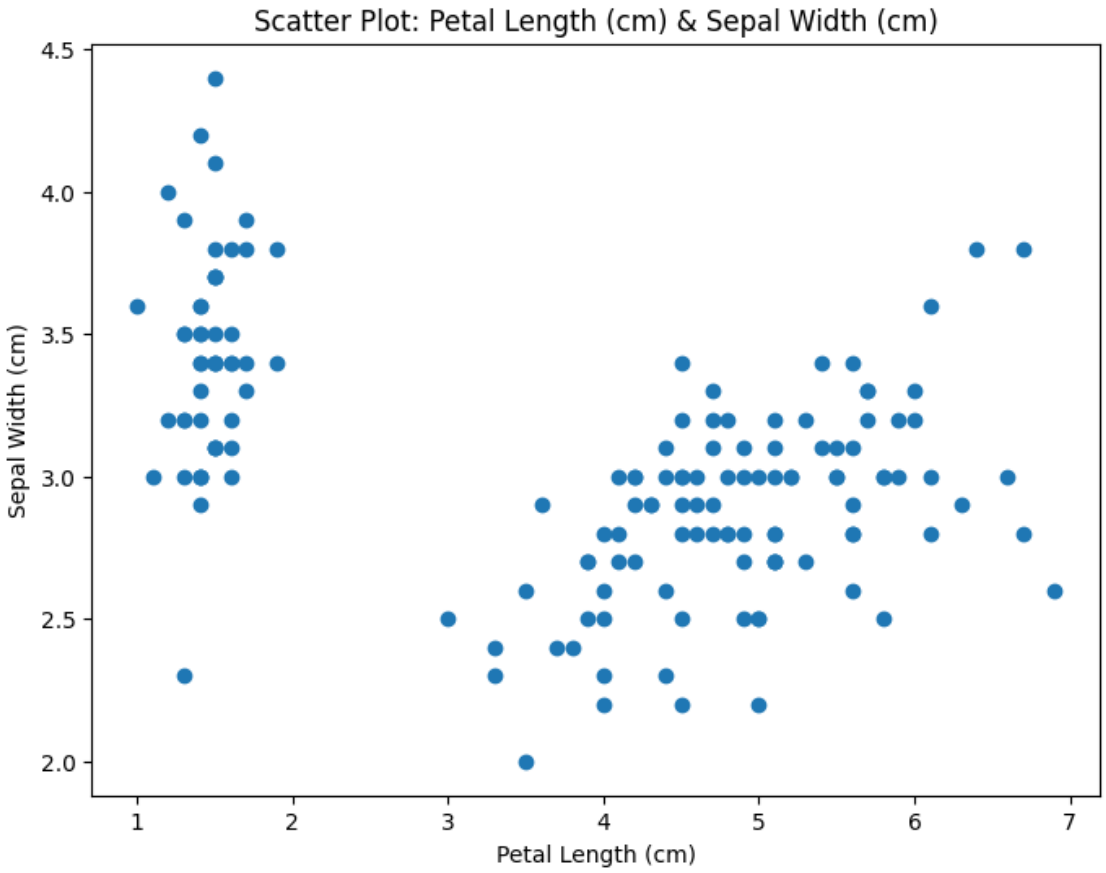
[19]:
```
plt.figure(figsize=(8,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.scatter(dataset["petal length (cm)"], dataset["petal width (cm)"])

plt.title("Scatter Plot: Petal Length (cm) & Petal Width (cm)")
plt.xlabel("Petal Length (cm)")
plt.ylabel("Petal Width (cm)")

plt.show()
```



[20]:
```
plt.figure(figsize=(8,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.scatter(dataset["sepal length (cm)"], dataset["petal width (cm)"])

plt.title("Scatter Plot: Sepal Length (cm) & Petal Width (cm)")
plt.xlabel("Sepal Length (cm)")
plt.ylabel("Petal Width (cm)")

plt.show()
```

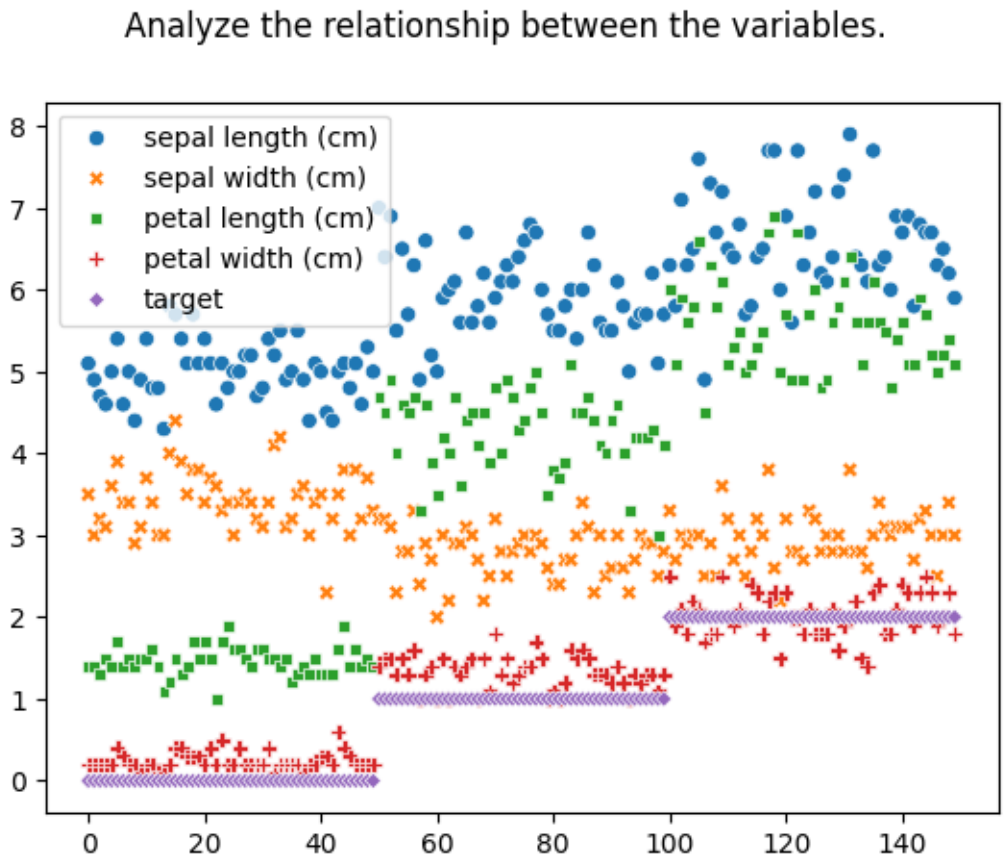


[21]:
```
plt.figure(figsize=(8,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.scatter(dataset["petal length (cm)"], dataset["sepal width (cm)"])

plt.title("Scatter Plot: Petal Length (cm) & Sepal Width (cm)")
plt.xlabel("Petal Length (cm)")
plt.ylabel("Sepal Width (cm)")

plt.show()
```



[22]:
```
import seaborn as sns
```

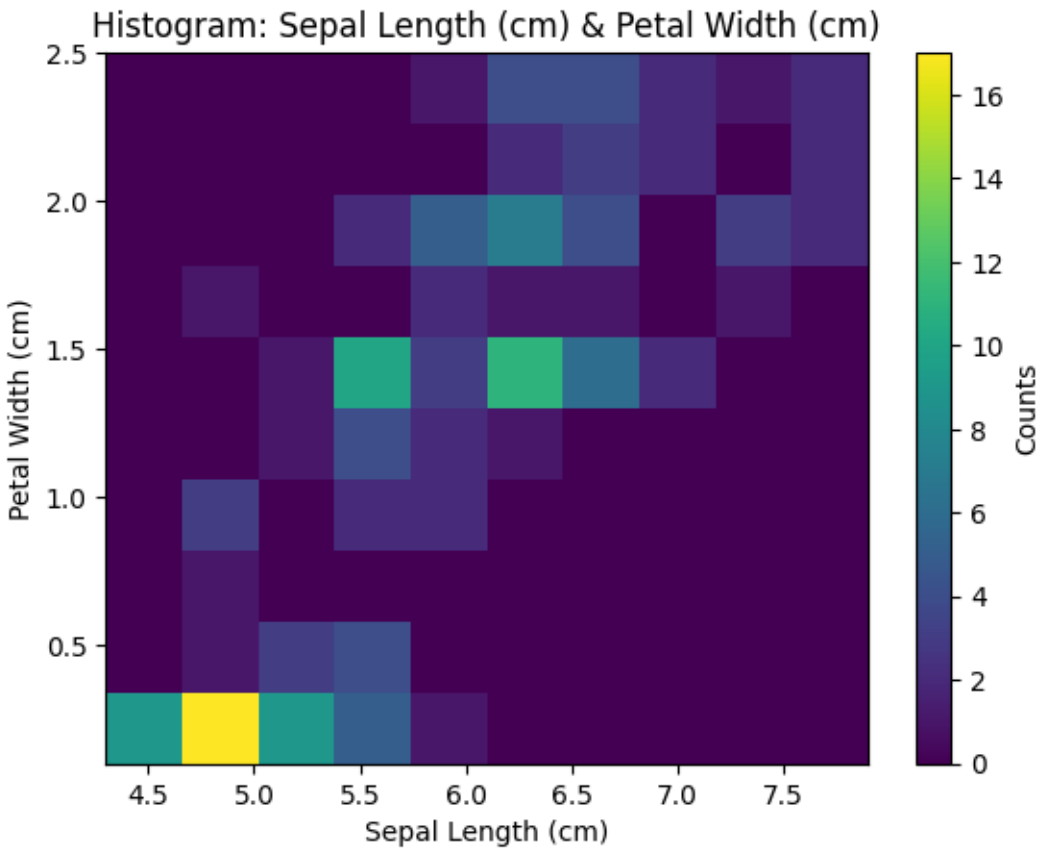
[23]:
```
sns.scatterplot(dataset)

plt.suptitle("Analyze the relationship between the variables.")

plt.show()
```



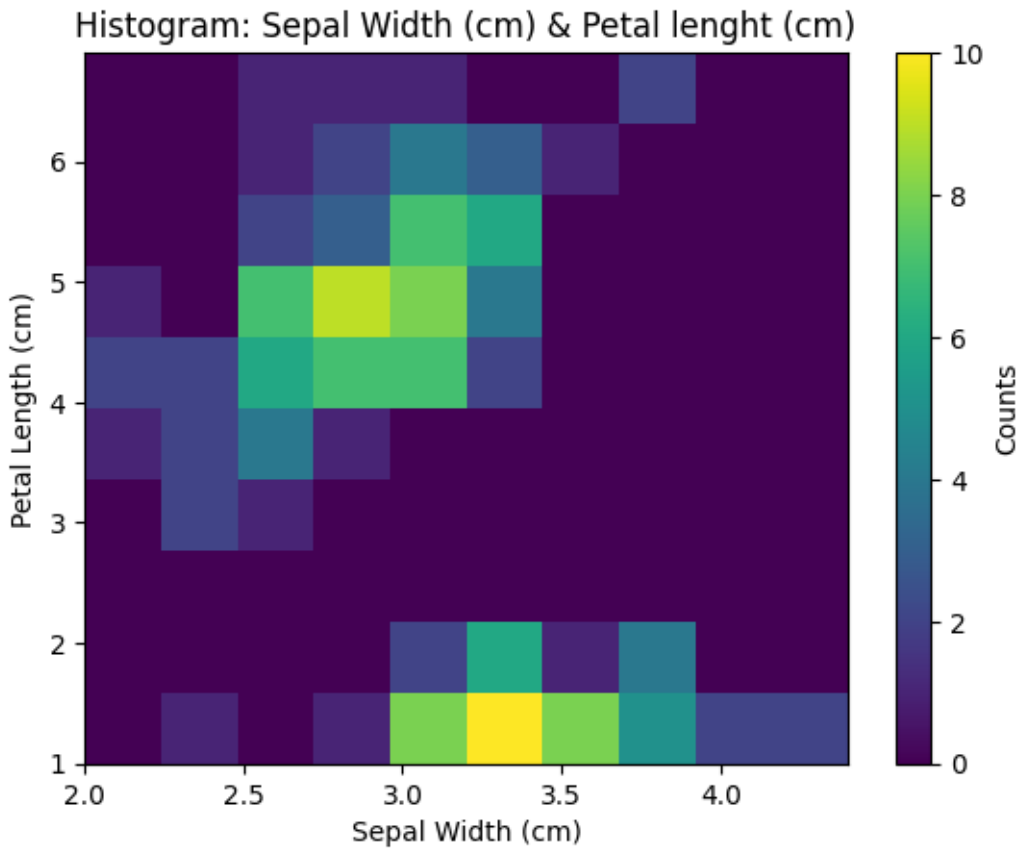
Create basic visualizations: Histogram to examine the data distribution.

[24]:
```
import matplotlib as pyplot
from matplotlib import pyplot as plt
```

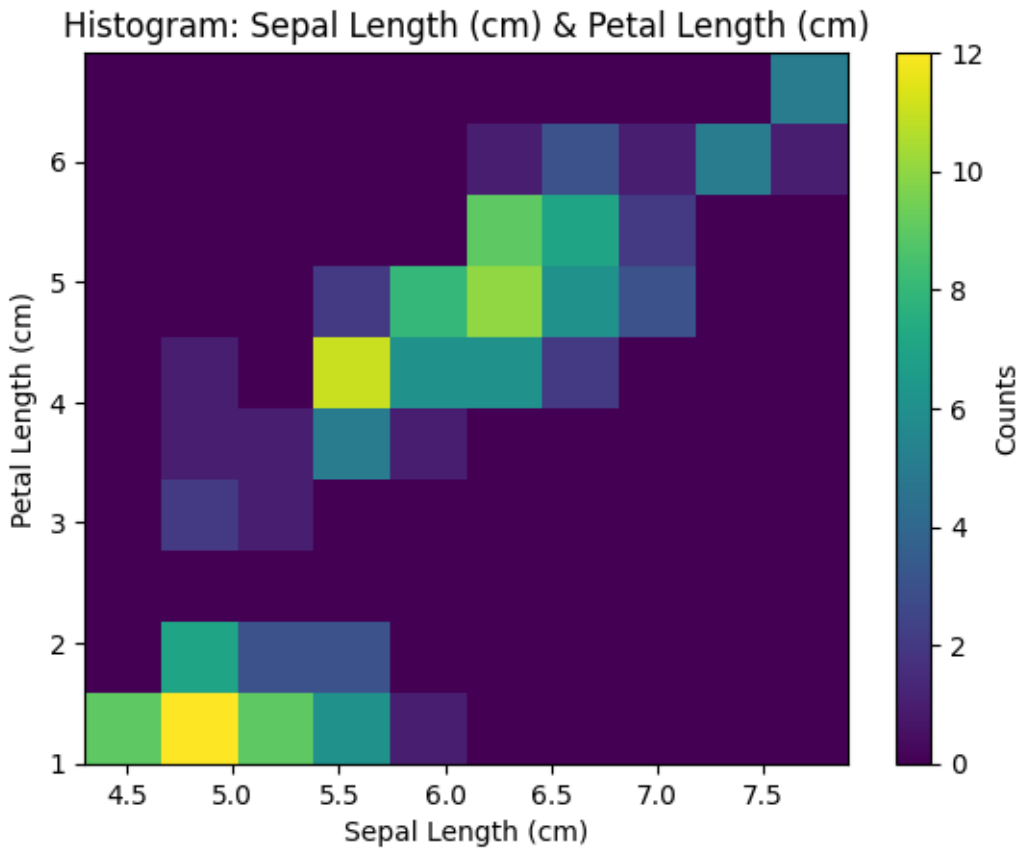
[25]:
```
plt.hist2d(dataset["sepal length (cm)"],dataset["petal width (cm)"])

plt.title("Histogram: Sepal Length (cm) & Petal Width (cm)")
plt.xlabel("Sepal Length (cm)")
plt.ylabel("Petal Width (cm)")
plt.colorbar(label= "Counts")

plt.show()
```



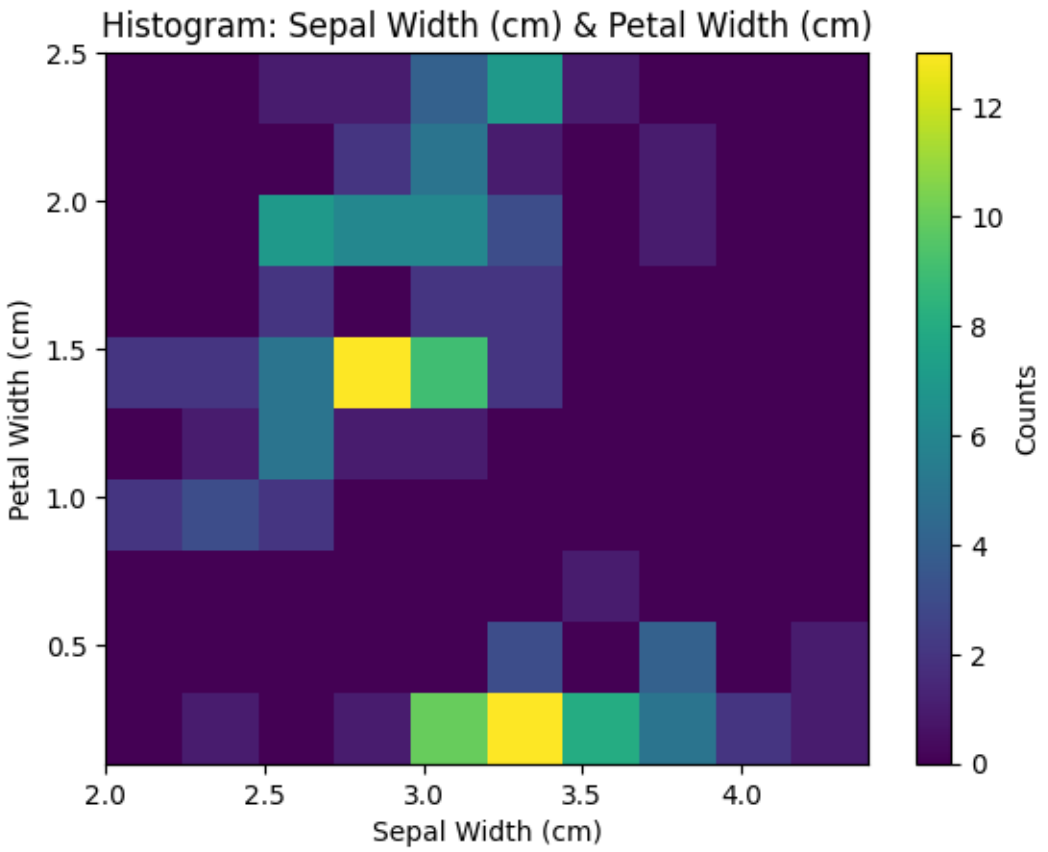
[26]:
```
plt.hist2d(dataset["sepal width (cm)"],dataset["petal length (cm)"])

plt.title("Histogram: Sepal Width (cm) & Petal lenght (cm)")
plt.xlabel("Sepal Width (cm)")
plt.ylabel("Petal Length (cm)")
plt.colorbar(label= "Counts")

plt.show()
```



[27]:
```
plt.hist2d(dataset["sepal length (cm)"],dataset["petal length (cm)"])

plt.title("Histogram: Sepal Length (cm) & Petal Length (cm)")
plt.xlabel("Sepal Length (cm)")
plt.ylabel("Petal Length (cm)")
plt.colorbar(label= "Counts")

plt.show()
```

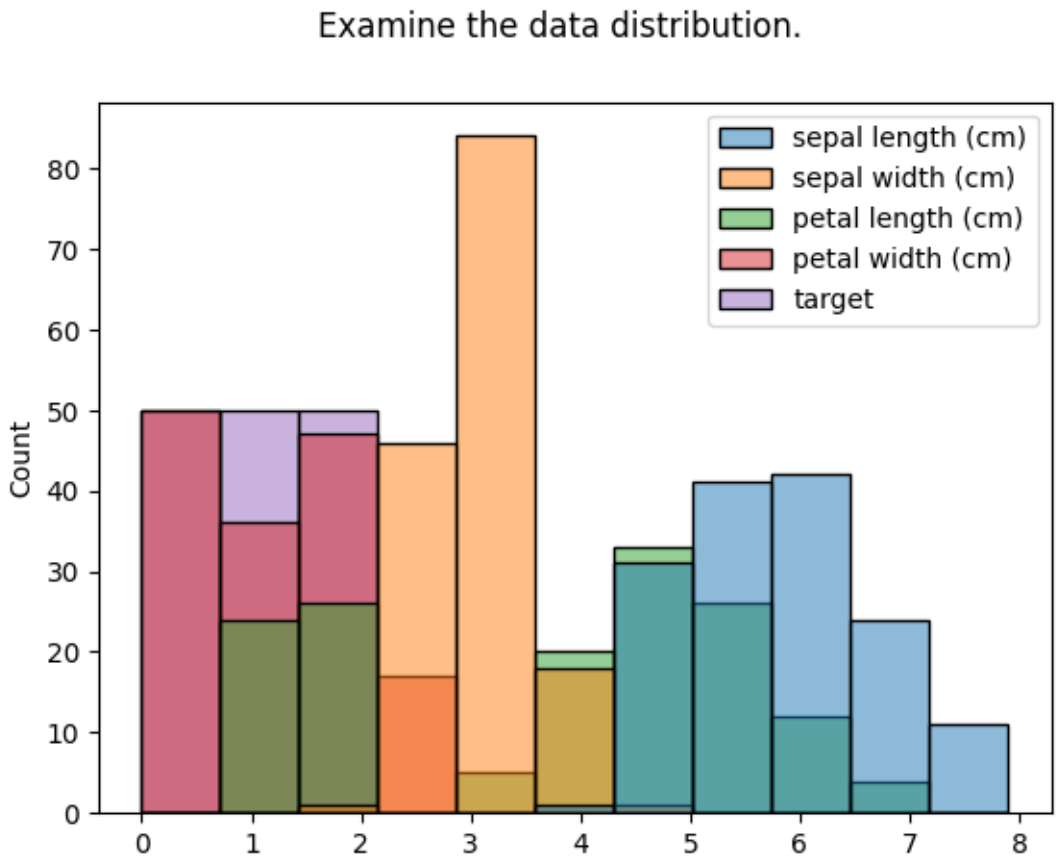


[28]:
```
plt.hist2d(dataset["sepal width (cm)"],dataset["petal width (cm)"])

plt.title("Histogram: Sepal Width (cm) & Petal Width (cm)")
plt.xlabel("Sepal Width (cm)")
plt.ylabel("Petal Width (cm)")
plt.colorbar(label= "Counts")

plt.show()
```



[29]:
```
import seaborn as sns
```

[30]:
```
sns.histplot(dataset)

plt.suptitle("Examine the data distribution.")

plt.show()
```

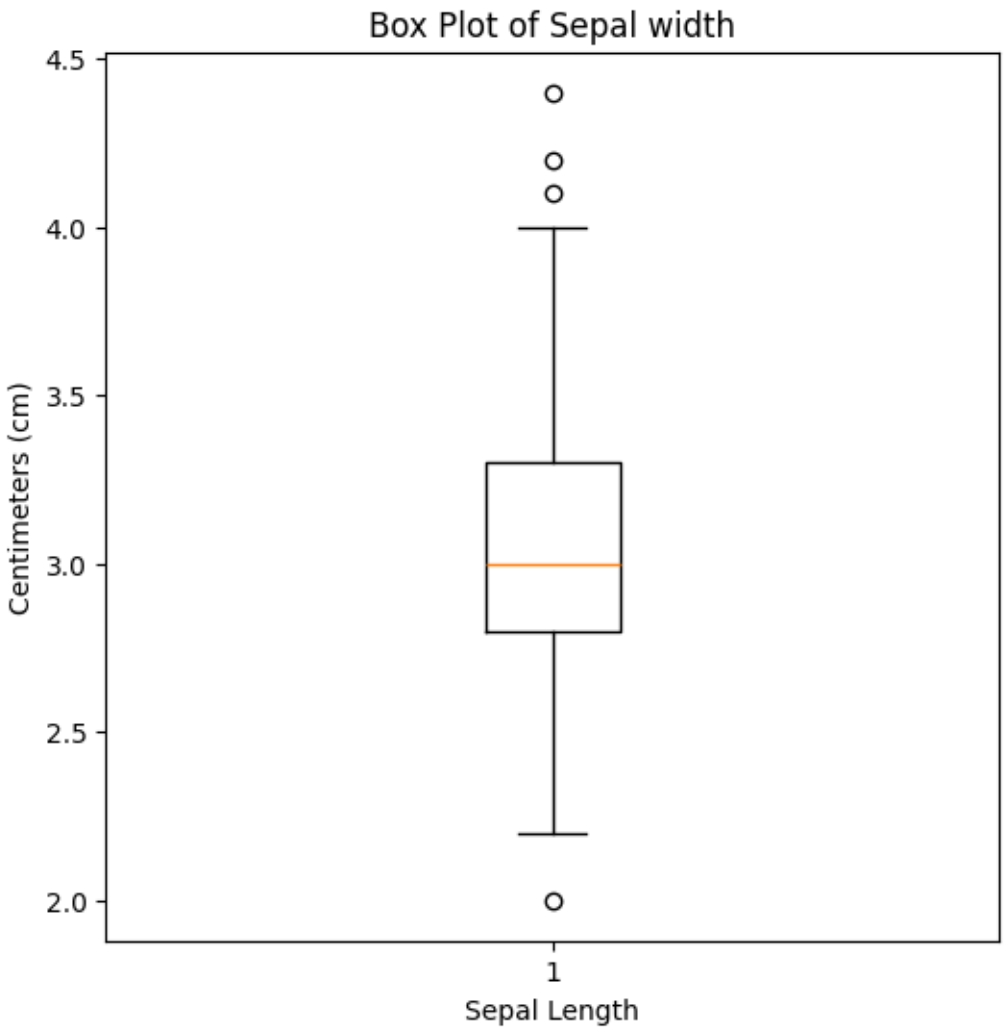


Create basic visualizations: Box plot to detect outliers and spread of values.

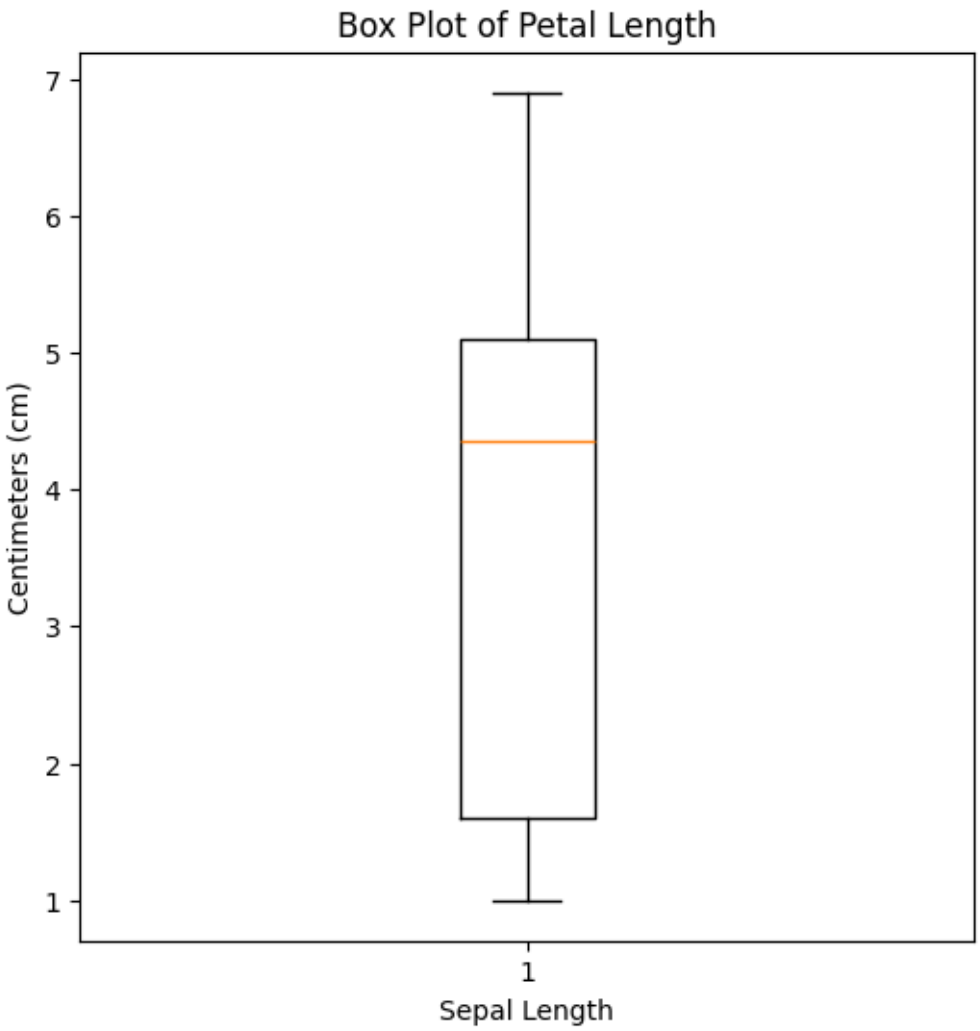
[31]:
```
import matplotlib as pyplot
from matplotlib import pyplot as plt
```

[32]:
```
plt.figure(figsize=(6,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.boxplot(dataset["sepal length (cm)"])

plt.xlabel("Sepal Length")
plt.ylabel("Centimeters (cm)")
plt.title("Box Plot of Sepal Length")

plt.show()
```

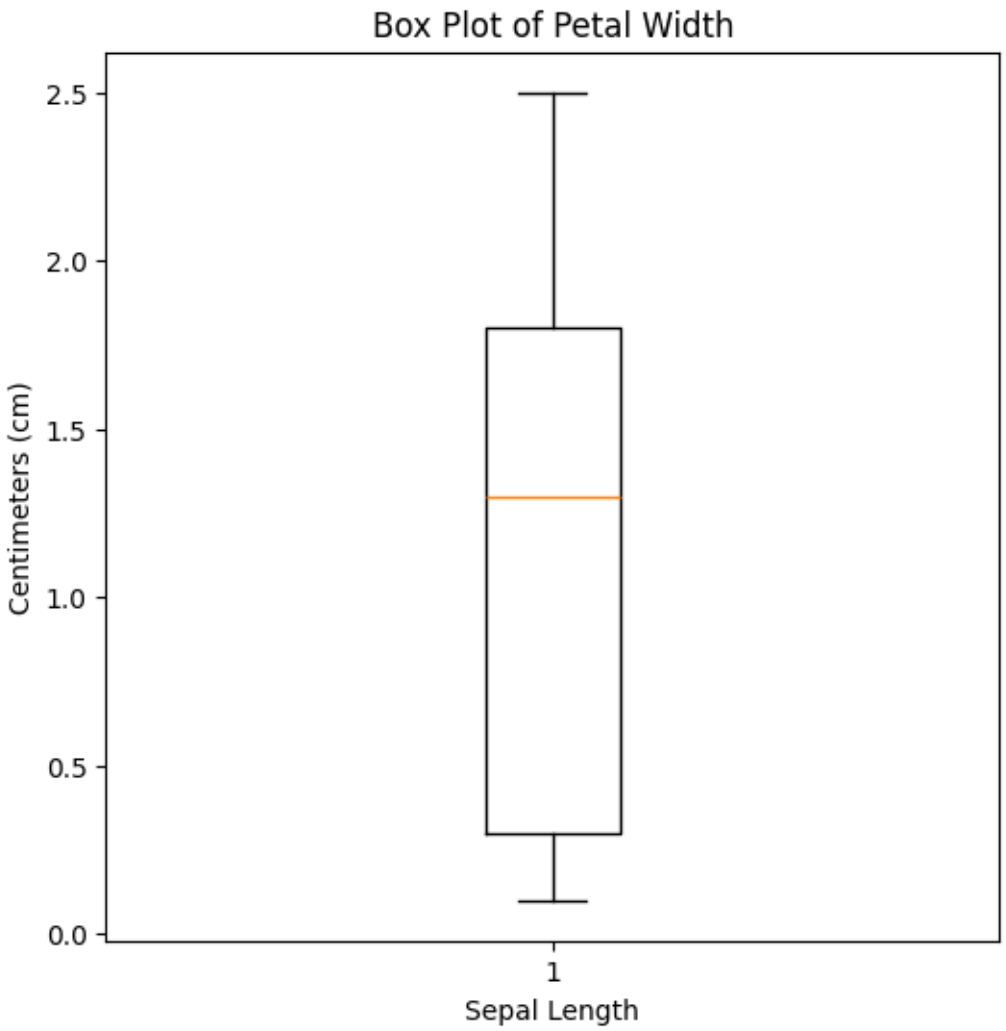
[33]:
```
plt.figure(figsize=(6,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.boxplot(dataset["sepal width (cm)"])

plt.xlabel("Sepal Length")
plt.ylabel("Centimeters (cm)")
plt.title("Box Plot of Sepal width")

plt.show()
```



[34]:
```
plt.figure(figsize=(6,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.boxplot(dataset["petal length (cm)"])

plt.xlabel("Sepal Length")
plt.ylabel("Centimeters (cm)")
plt.title("Box Plot of Petal Length")

plt.show()
```



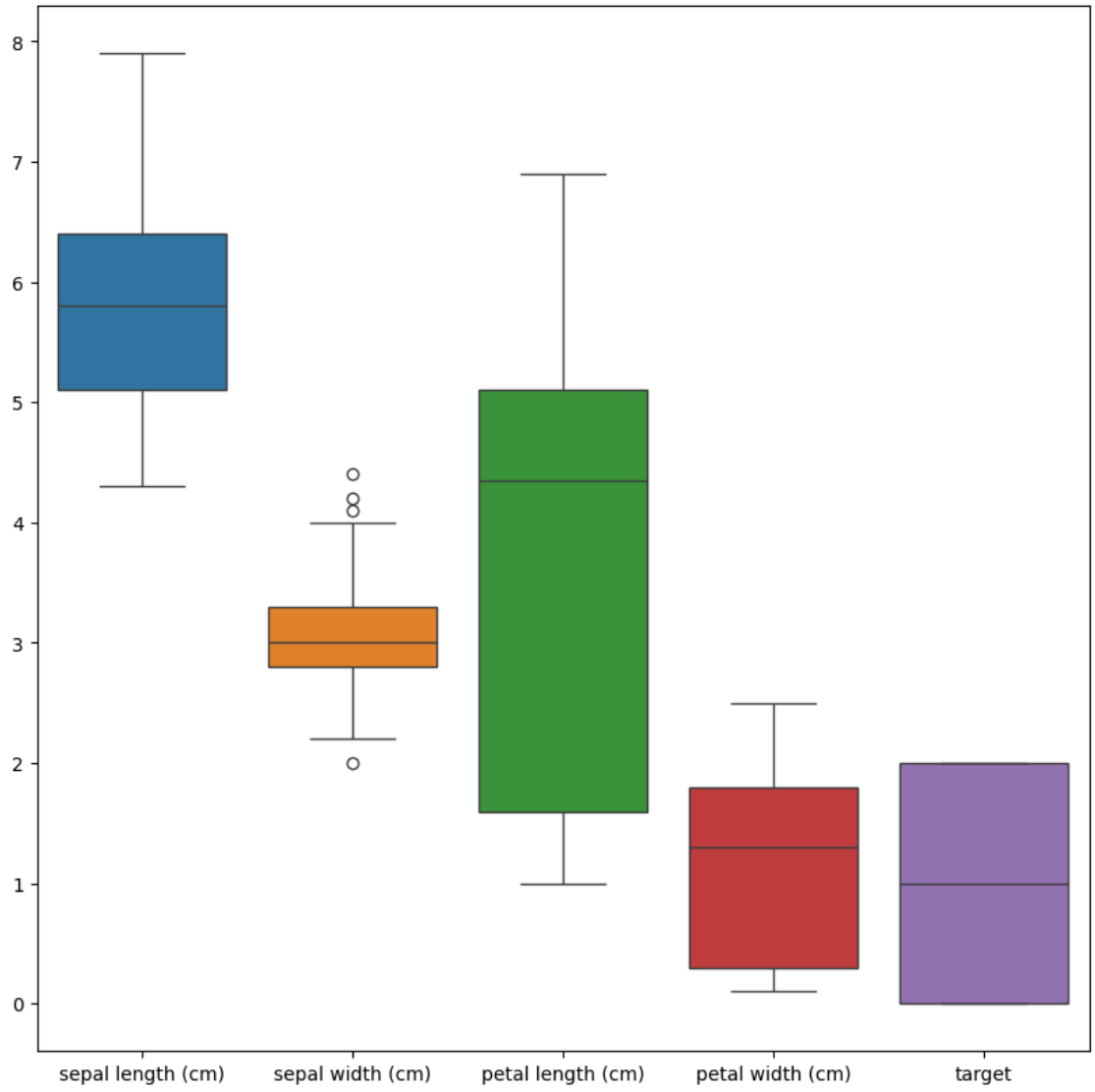
[35]:
```
plt.figure(figsize=(6,6))

# Why we use the plt.figure(figsize(width,height)).
# This sets the size of the figure.
# The values are given in inches.
# 8 = Width of the figure.
# 6 = Height of the figure.

plt.boxplot(dataset["petal width (cm)"])

plt.xlabel("Sepal Length")
plt.ylabel("Centimeters (cm)")
plt.title("Box Plot of Petal Width")

plt.show()
```



[36]:
```
import seaborn as sns
```

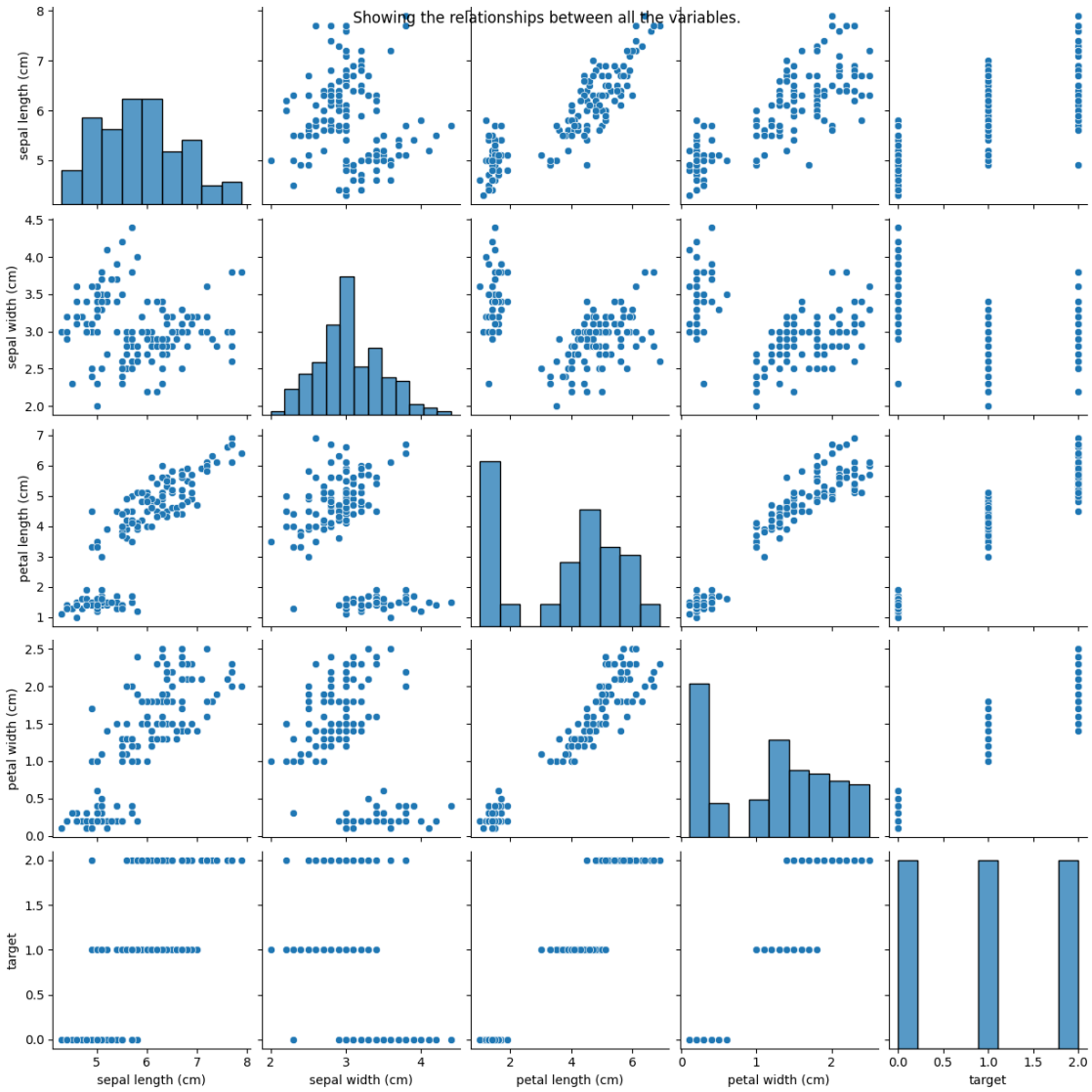
[37]:
```
plt.figure(figsize=(10,10))

sns.boxplot(dataset)

plt.suptitle("Detect outliers and spread of values.")

plt.show()
```

Detect outliers and spread of values.



[38]:
```
import seaborn as sns
```

[39]:
```
sns.pairplot(dataset)

plt.suptitle("Showing the relationships between all the variables.")

plt.show()
```



Skills: Data loading and inspection using pandas. Basic data summarization. Visualization using matplotlib and seaborn.

Task Completed. Best Wishes. Zaigham Abbas.