

Task 1: Exploring and Visualizing a Simple Dataset

Importing Necessary Libraries

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

Load the Iris Dataset

```
# Loading the Iris dataset from seaborn
df = sns.load_dataset('iris')
```

shape, column names, and the first few rows using .head()

```
print("Shape of dataset:", df.shape)          # (rows, columns)
print("Columns:", df.columns.tolist())         # Column names
print("\nFirst 5 rows:")
print(df.head())                             # Top 5 rows
```

```
➦ Shape of dataset: (150, 5)
Columns: ['sepal_length', 'sepal_width', 'petal_length', 'petal_width', 'species']
```

```
First 5 rows:
   sepal_length  sepal_width  petal_length  petal_width  species
0           5.1           3.5           1.4           0.2   setosa
1           4.9           3.0           1.4           0.2   setosa
2           4.7           3.2           1.3           0.2   setosa
3           4.6           3.1           1.5           0.2   setosa
4           5.0           3.6           1.4           0.2   setosa
```

.info() and .describe()

```
print("\nDataset Info:")
df.info()

print("\nSummary Statistics:")
print(df.describe())
```

```
➦ Dataset 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

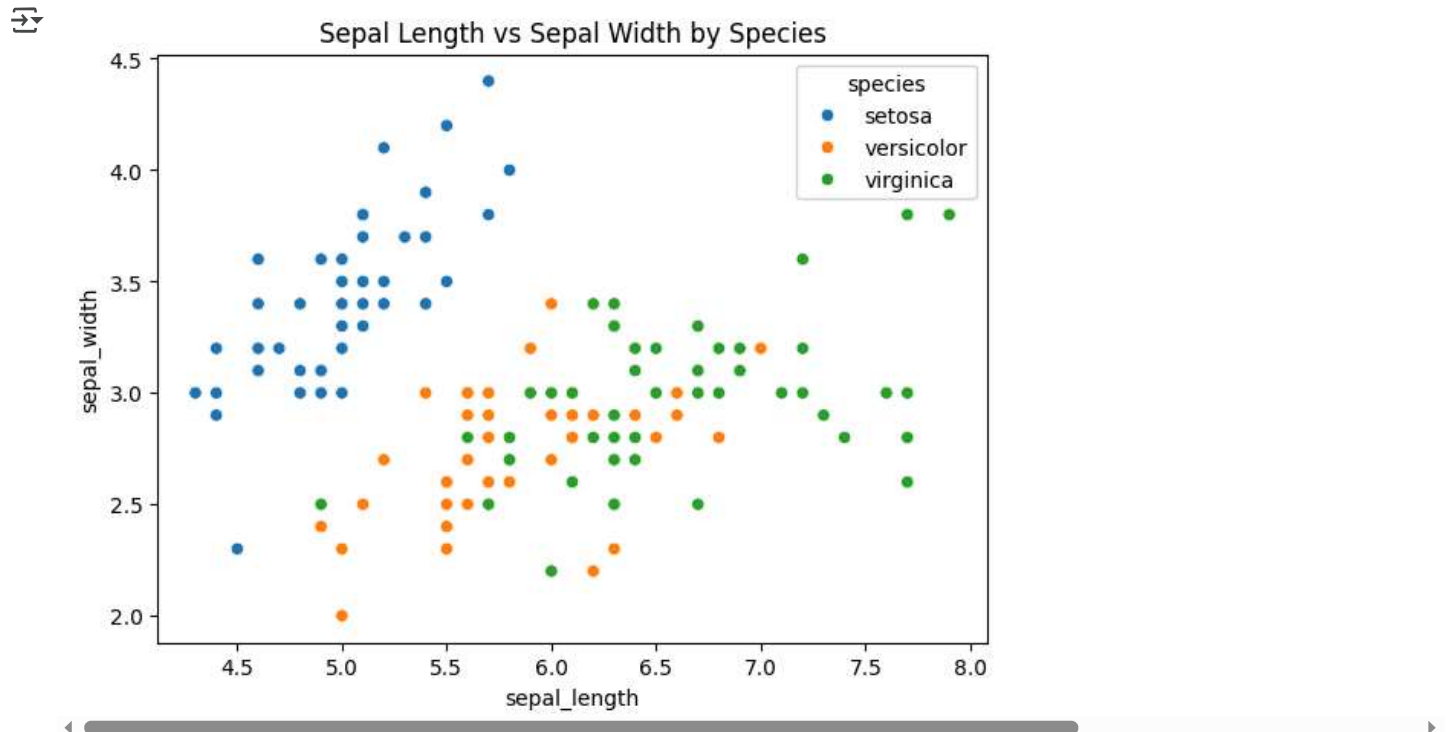
Summary Statistics:
   sepal_length  sepal_width  petal_length  petal_width
count    150.000000    150.000000    150.000000    150.000000
mean         5.843333     3.057333     3.758000     1.199333
std          0.828066     0.435866     1.765298     0.762238
min          4.300000     2.000000     1.000000     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

Data Visualization

Scatter Plot: Sepal Length vs Sepal Width

```
plt.figure(figsize=(7,5))
sns.scatterplot(data=df, x='sepal_length', y='sepal_width', hue='species')
plt.title('Sepal Length vs Sepal Width by Species')
plt.show()
```

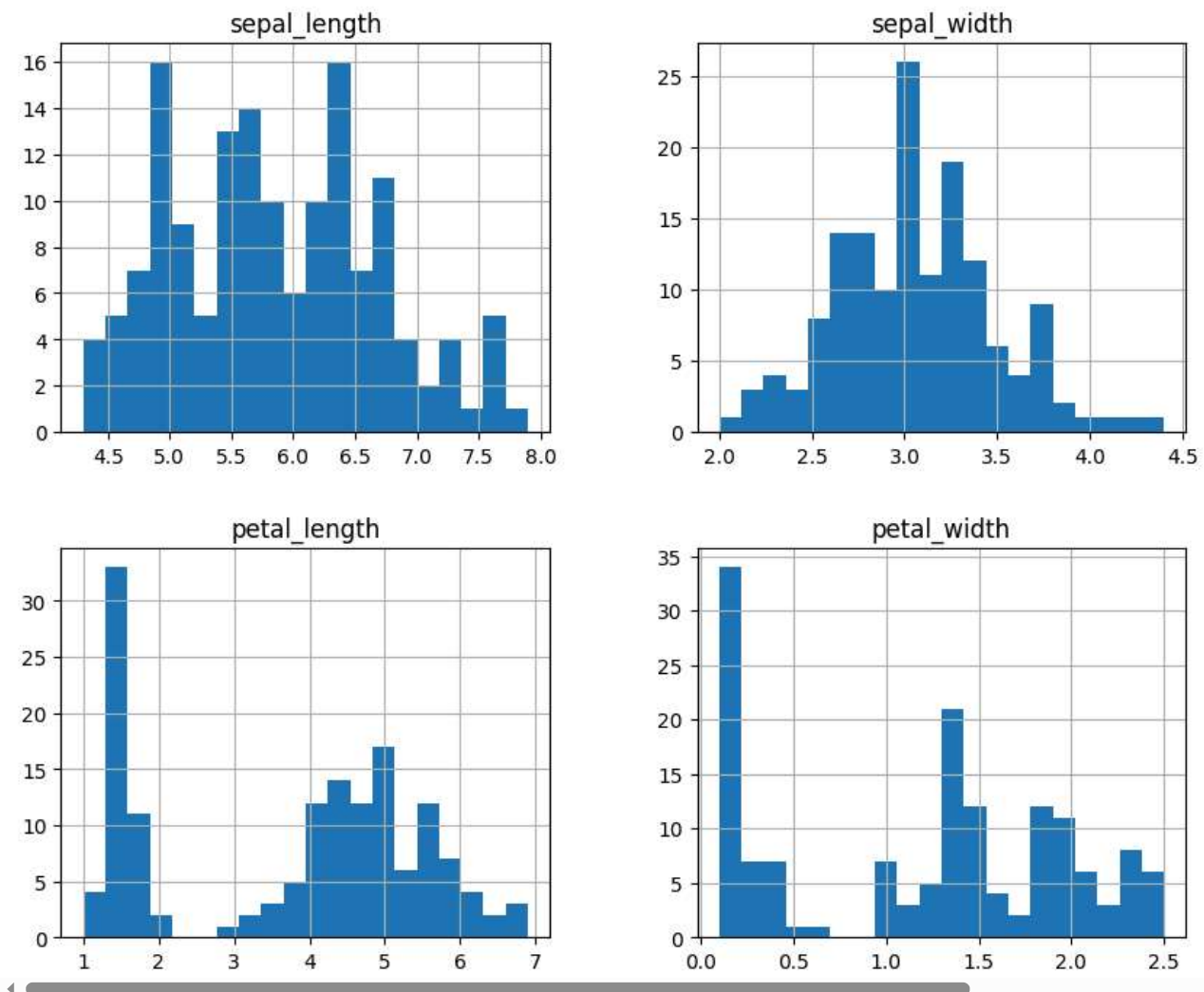


Histograms for All Features

```
df.hist(figsize=(10,8), bins=20)
plt.suptitle('Feature Distributions')
plt.show()
```

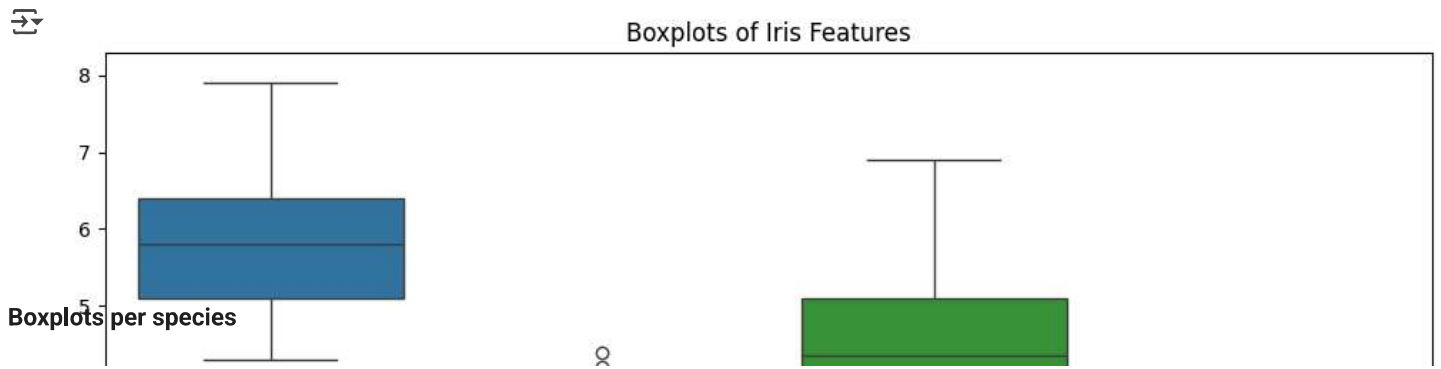


Feature Distributions

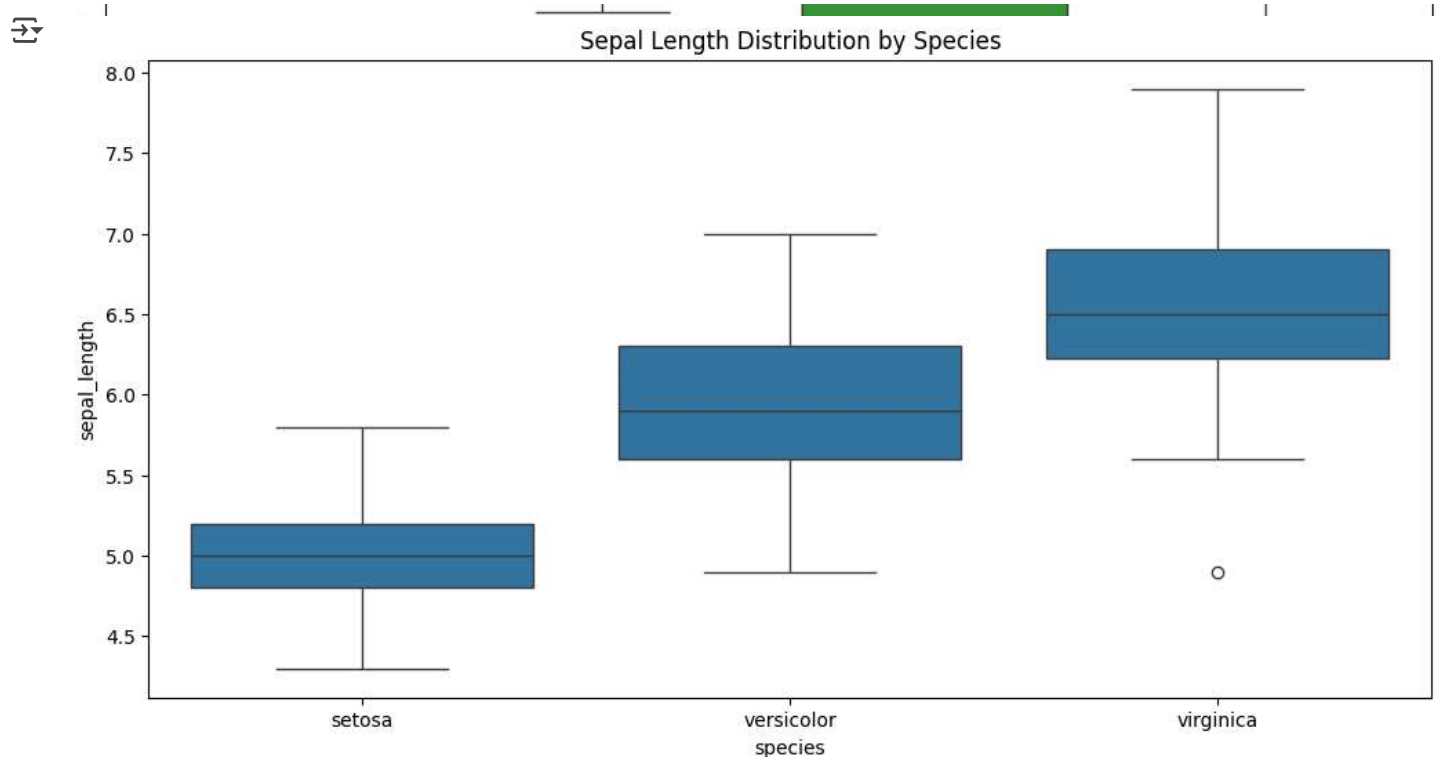


Box Plots for Outlier Detection

```
plt.figure(figsize=(12,6))
sns.boxplot(data=df)
plt.title('Boxplots of Iris Features')
plt.show()
```



```
plt.figure(figsize=(12,6))
sns.boxplot(data=df, x='species', y='sepal_length')
plt.title('Sepal Length Distribution by Species')
plt.show()
```



Explanation of Results and Final Insights:

After loading and inspecting the Iris dataset, we explored its structure, summary statistics, and distributions using Pandas, Matplotlib, and Seaborn. The visualizations—such as scatter plots, histograms, and box plots—revealed clear patterns and distinctions between the different Iris species. The scatter plot between sepal length and sepal width showed that species are somewhat separable based on these features. Histograms indicated the distribution of each numerical feature, while box plots helped identify potential outliers. Overall, this exploratory analysis helped us understand the key trends and separability in the data, laying a solid foundation for any further machine learning tasks.