Data Visualization III Download the Iris flower dataset or any other dataset into a DataFrame. (e.g., https://archive.ics.uci.edu/ml/datasets/Iris). Scan the dataset and give the inference as:

1. List down the features and their types (e.g., numeric, nominal) available in the dataset.
2. Create a histogram for each feature in the dataset to illustrate the feature distributions.
3. Create a box plot for each feature in the dataset.
4. Compare distributions and identify outliers.

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

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[1], line 2
      1 import pandas as pd
----> 2 import seaborn as sns
      3 import matplotlib.pyplot as plt

ModuleNotFoundError: No module named 'seaborn'
```

```
pip install seaborn
```

```
Collecting seaborn
  Downloading seaborn-0.13.2-py3-none-any.whl.metadata (5.4 kB)
Requirement already satisfied: numpy!=1.24.0,>=1.20 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packag
Requirement already satisfied: pandas>=1.2 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (from
Requirement already satisfied: matplotlib!=3.6.1,>=3.4 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-pac
Requirement already satisfied: contourpy>=1.0.1 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages
Requirement already satisfied: cycler>=0.10 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (from
Requirement already satisfied: fonttools>=4.22.0 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages
Requirement already satisfied: packaging>=20.0 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (f
Requirement already satisfied: pillow>=8 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (from ma
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages
Requirement already satisfied: python-dateutil>=2.7 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packag
Requirement already satisfied: pytz>=2020.1 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (from
Requirement already satisfied: tzdata>=2022.7 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (fr
Requirement already satisfied: six>=1.5 in c:\users\rutuja habib\appdata\local\programs\python\python313\lib\site-packages (from pyt
Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)
Installing collected packages: seaborn
Successfully installed seaborn-0.13.2
Note: you may need to restart the kernel to use updated packages.
```

```
import seaborn as sns
```

```
iris = sns.load_dataset('iris')
```

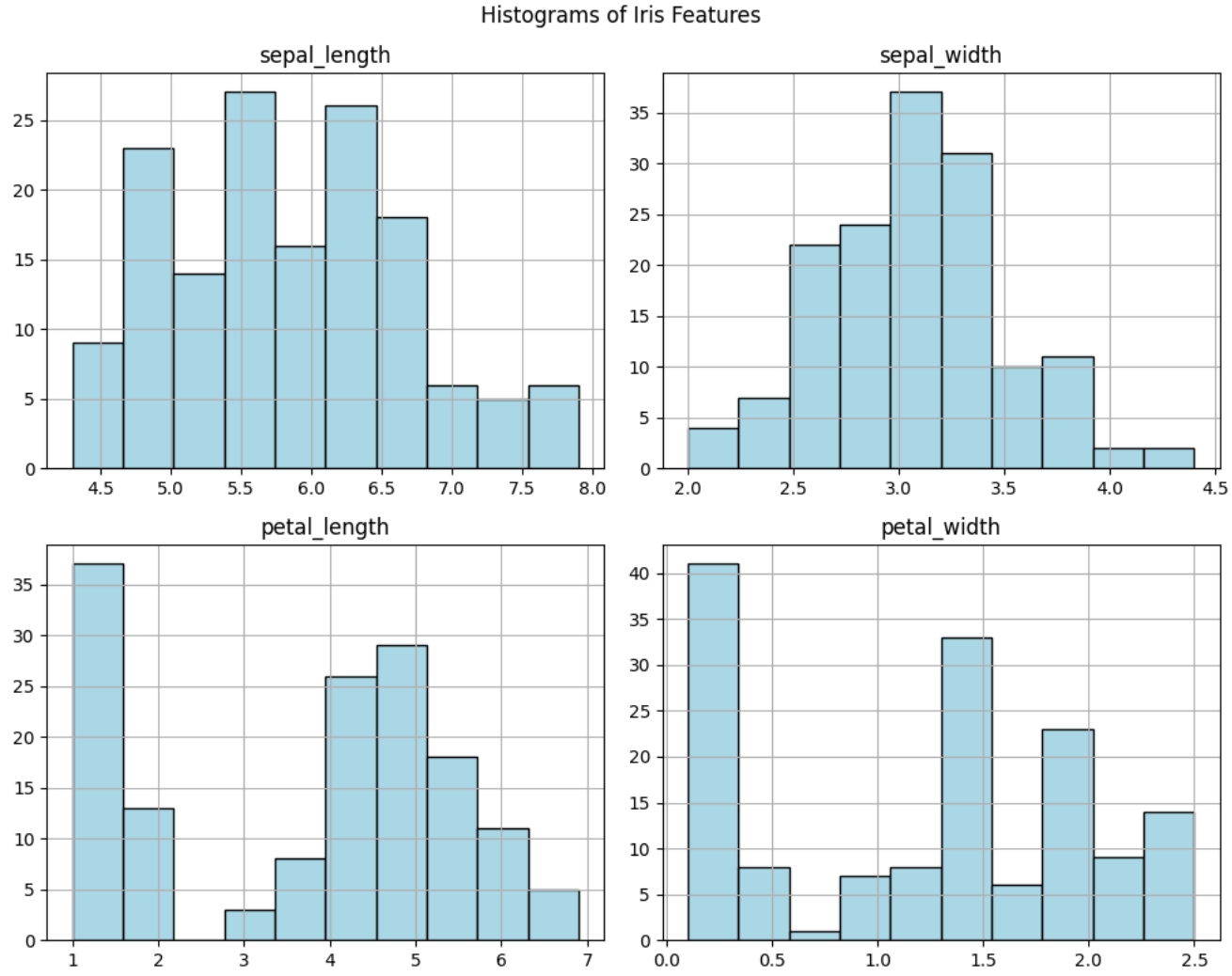
```
# 1. List down features and their types
print("1. Features and their types")
print(iris.dtypes)
print("\nFeature Types:")
for column in iris.columns:
    dtype = iris[column].dtype
    if dtype == 'object':
        print(f"{column}: Nominal")
    else:
        print(f"{column}: Numeric")
```

```
1. Features and their types
sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species          object
dtype: object

Feature Types:
sepal_length: Numeric
sepal_width: Numeric
petal_length: Numeric
petal_width: Numeric
species: Nominal
```

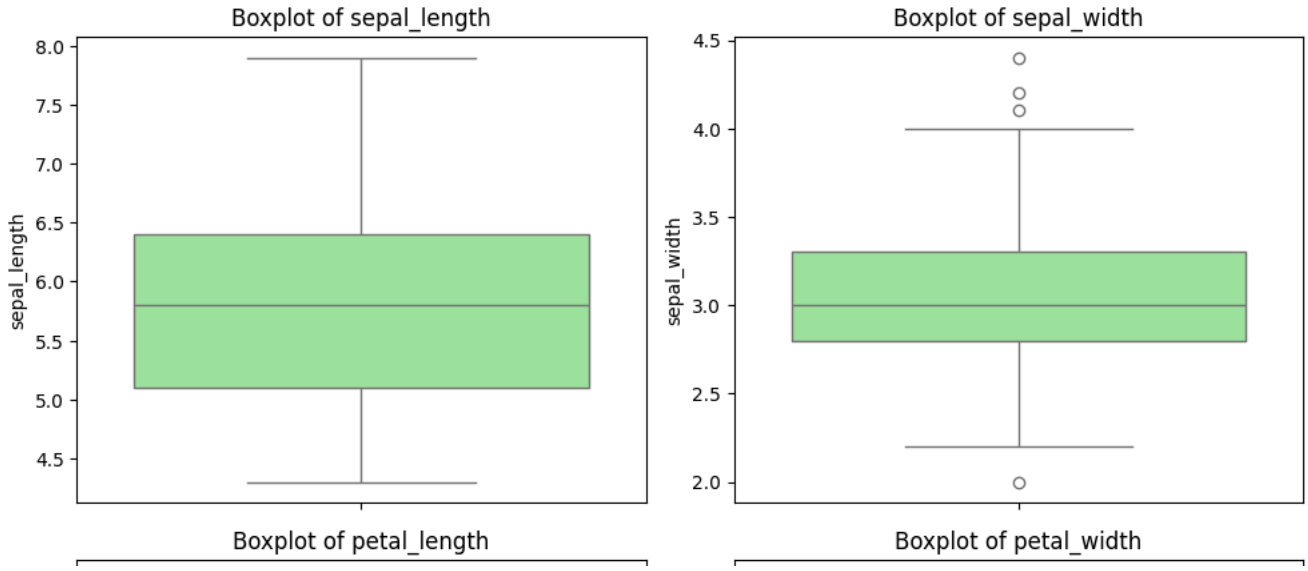
```
# 2. Histogram for each numeric feature
numeric_features = iris.select_dtypes(include='number').columns
print("\n2. Plotting histograms...")
iris[numeric_features].hist(figsize=(10, 8), color='lightblue', edgecolor='black')
plt.suptitle('Histograms of Iris Features')
plt.tight_layout()
plt.show()
```

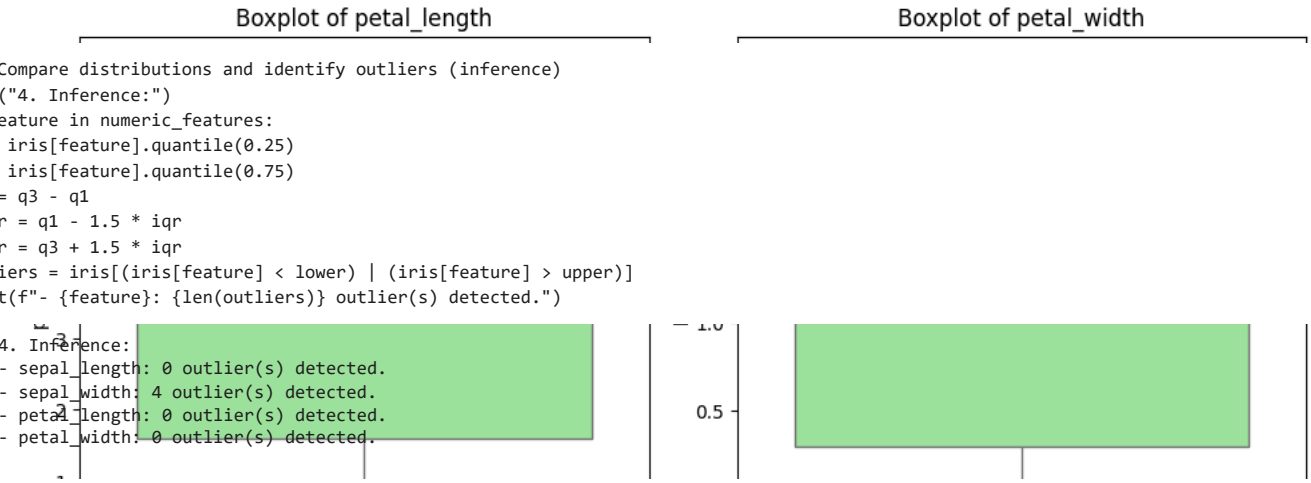
```
2. Plotting histograms...
```




```
print("3. Plotting boxplots...")
plt.figure(figsize=(10, 8))
for i, feature in enumerate(numeric_features, 1):
 plt.subplot(2, 2, i)
 sns.boxplot(y=iris[feature], color='lightgreen')
 plt.title(f'Boxplot of {feature}')
plt.tight_layout()
plt.show()
```

```
3. Plotting boxplots...
```






```
# 4. Compare distributions and identify outliers (inference)
print("4. Inference:")
for feature in numeric_features:
 q1 = iris[feature].quantile(0.25)
 q3 = iris[feature].quantile(0.75)
 iqr = q3 - q1
 lower = q1 - 1.5 * iqr
 upper = q3 + 1.5 * iqr
 outliers = iris[(iris[feature] < lower) | (iris[feature] > upper)]
 print(f"- {feature}: {len(outliers)} outlier(s) detected.")
```

```
4. Inference:
- sepal_length: 0 outlier(s) detected.
- sepal_width: 4 outlier(s) detected.
- petal_length: 0 outlier(s) detected.
- petal_width: 0 outlier(s) detected.
```

Start coding or generate with AI.