Objective - Feature Distribution and its visualization

```
#1. Loadingthe required
library import numpy as np
import pandas as pd import
matplotlib.pyplot as plt
import seaborn as sns

## loading the dataset
iris=pd.read_csv('Iris.csv')
iris.head()
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
iris.isnull().sum()

# Display the count of null values for each column
# print(null_values)
```

```
Id              0
SepalLengthCm   0
SepalWidthCm    0
PetalLengthCm   0
PetalWidthCm    0
Species         0
dtype: int64
```

```
iris.dtypes
```

```
Id                int64
SepalLengthCm   float64
SepalWidthCm    float64
PetalLengthCm   float64
PetalWidthCm    float64
Species          object
dtype: object
```

```
iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
0   Id             150 non-null    int64
1   SepalLengthCm  150 non-null    float64
2   SepalWidthCm   150 non-null    float64
3   PetalLengthCm  150 non-null    float64
4   PetalWidthCm   150 non-null    float64 5
    Species        150 non-null    object
    dtypes: float64(4), int64(1), object(1)
    memory usage: 7.2+ KB
```

```
iris.describe()
```

	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

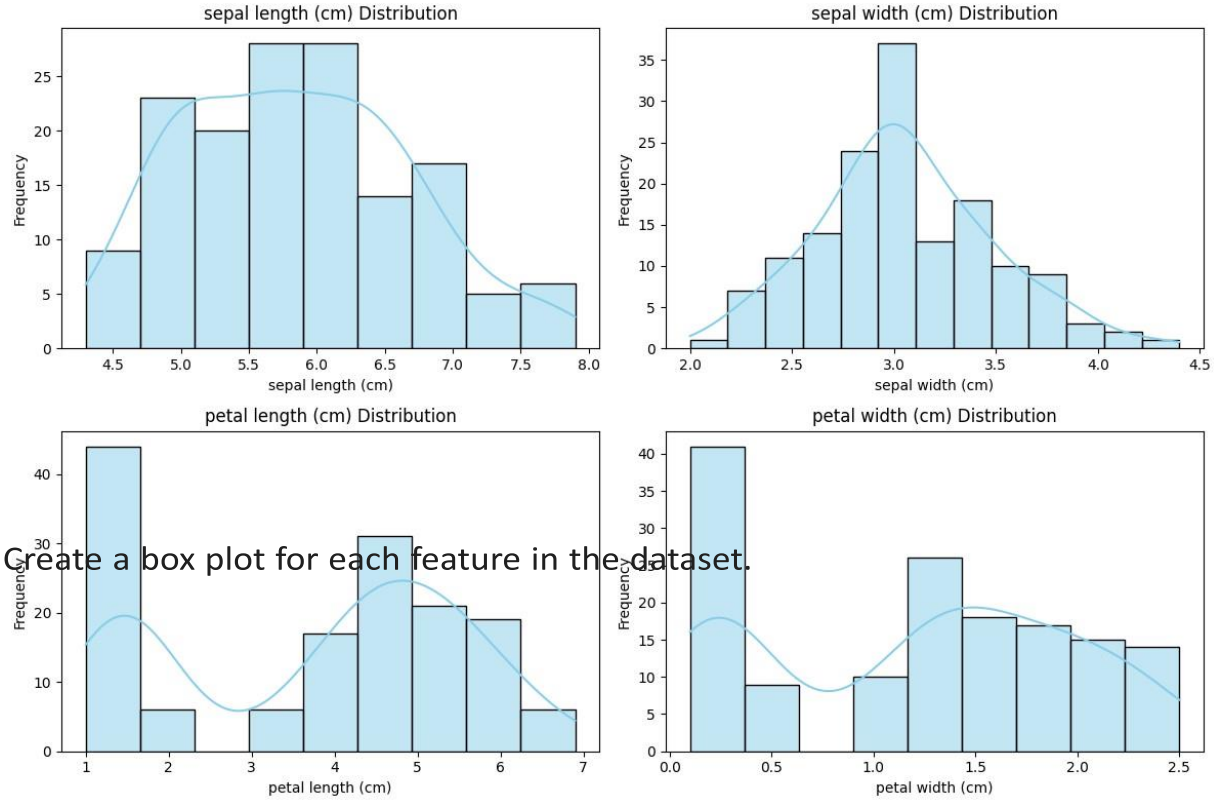
Note -

1. Numrecal Feature - Here after seeing this data we got four columns that are the numerical type sepalLength,sepalwidth,petalLength,petalwidth.
2. Categorical Feature - we have only one categorical feature that is species.

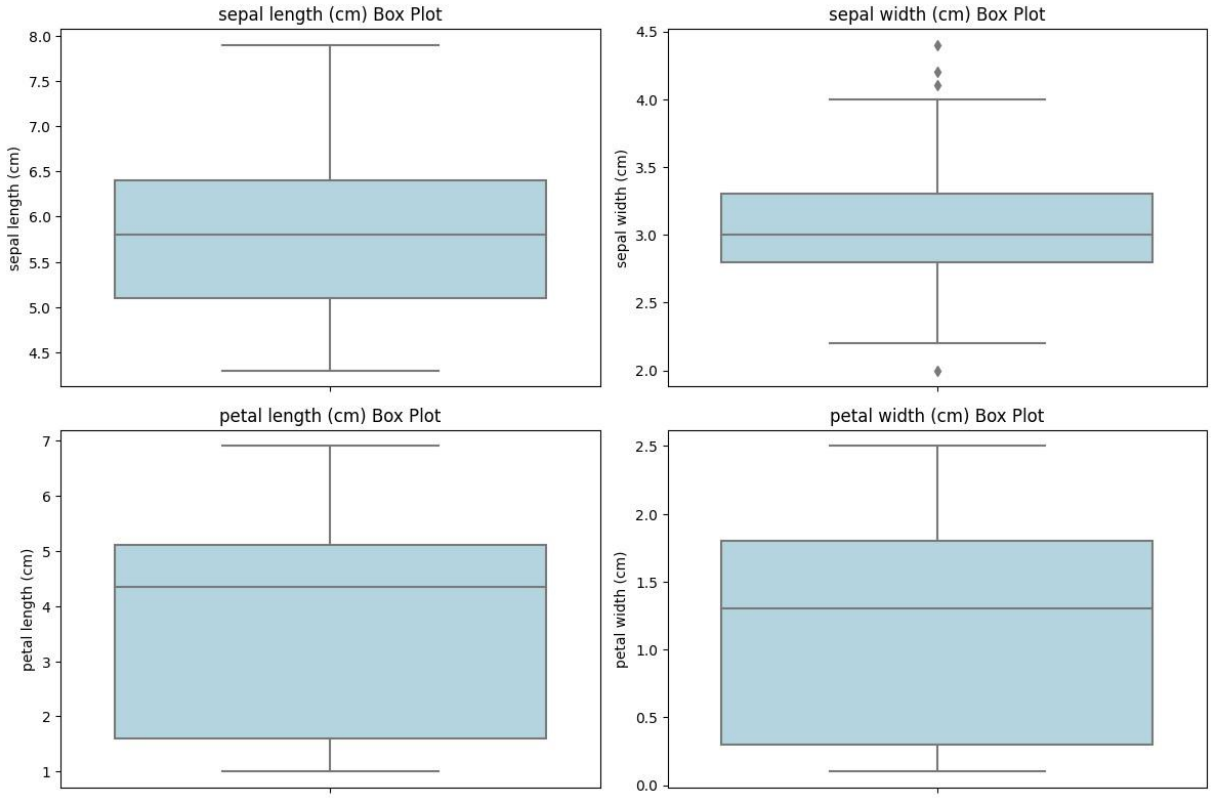
(b) Create a histogram for each feature in the dataset to illustrate the feature distributions.

```
import seaborn as sns import
matplotlib.pyplot as plt
import pandas as pd
from sklearn.datasets import load_iris

# Load the Iris dataset
iris = load_iris()
iris_df = pd.DataFrame(data=iris.data, columns=iris.feature_names)

# Create histograms for each feature
plt.figure(figsize=(12, 8))

for i, feature in enumerate(iris_df.columns):
    plt.subplot(2, 2, i + 1)
sns.histplot(iris_df[feature], kde=True, color='skyblue')
plt.title(f'{feature} Distribution')     plt.xlabel(feature)
plt.ylabel('Frequency')

plt.tight_layout()
plt.show()
```

```
import seaborn as sns import
matplotlib.pyplot as plt
import pandas as pd
from sklearn.datasets import load_iris

# Load the Iris dataset
iris = load_iris()
iris_df = pd.DataFrame(data=iris.data, columns=iris.feature_names)

# Create box plots for each feature
plt.figure(figsize=(12, 8))

for i, feature in enumerate(iris_df.columns):
    plt.subplot(2, 2, i + 1)     sns.boxplot(data=iris_df,
y=feature, color='lightblue')     plt.title(f'{feature}
Box Plot')

plt.tight_layout()
plt.show()
```



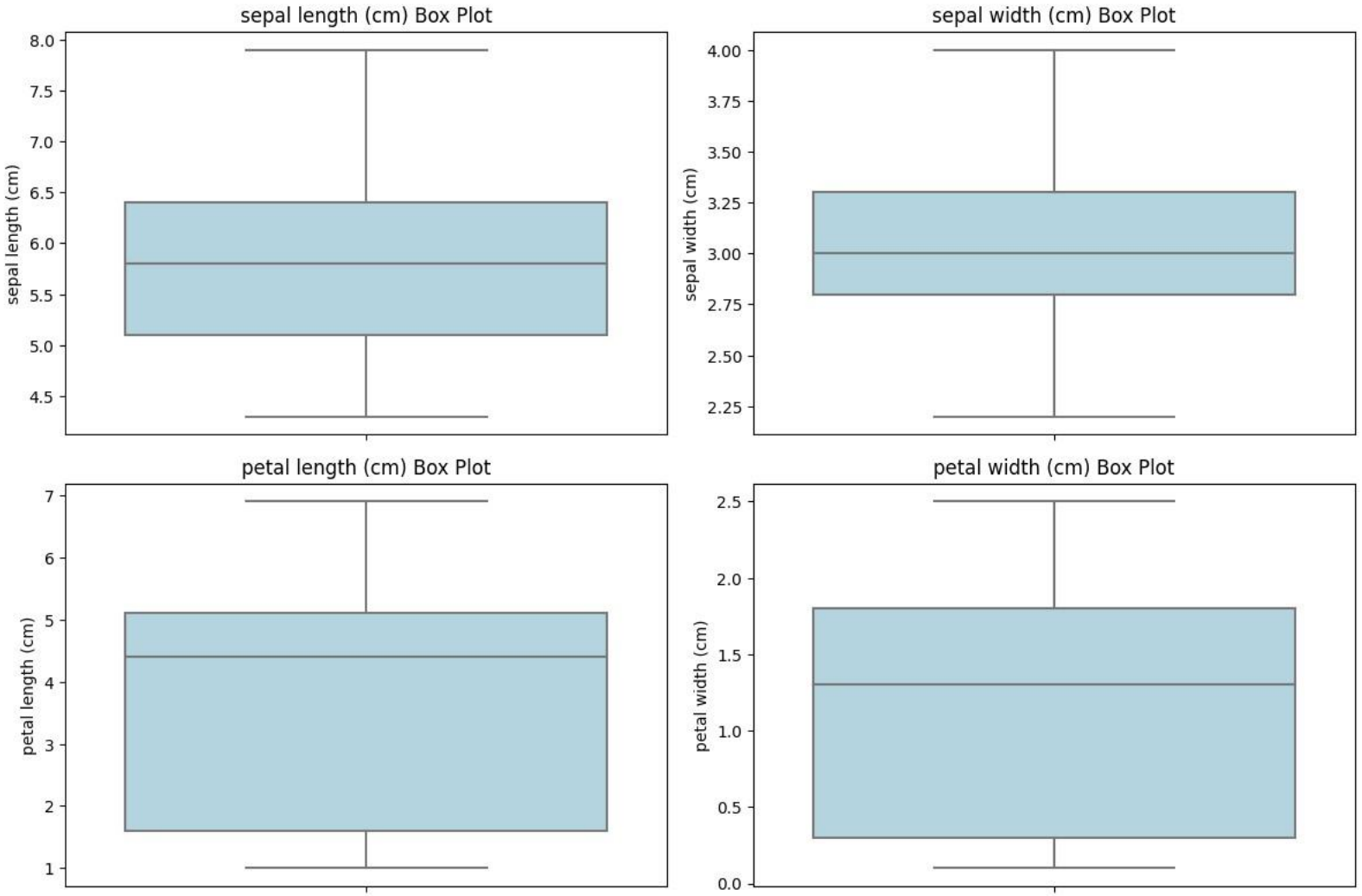
(c) Create a box plot for each feature in the dataset.



(d)Compare distributions and identify outliers.

```
# Calculate the IQR for sepal width
sepal_width = iris_df['sepal width (cm)']
Q1 = sepal_width.quantile(0.25)
Q3 = sepal_width.quantile(0.75)
IQR = Q3 - Q1

# Define the lower and upper bounds for outliers
lower_bound = Q1 - 1.5 * IQR upper_bound = Q3 +
1.5 * IQR

# Filter the dataframe to remove outliers
iris_df = iris_df[(sepal_width >= lower_bound) & (sepal_width <= upper_bound)]

# Now, iris_df contains the data without outliers in the sepal width column
```

```
# Create box plots for each feature
plt.figure(figsize=(12, 8))

for i, feature in enumerate(iris_df.columns):
    plt.subplot(2, 2, i + 1)     sns.boxplot(data=iris_df,
y=feature, color='lightblue')     plt.title(f'{feature}
Box Plot')

plt.tight_layout()
plt.show()
```


sepal length (cm) Box Plot
sepal length (cm)
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
sepal width (cm) Box Plot
sepal width (cm)
4.00
3.75
3.50
3.25
3.00
2.75
2.50
2.25
petal length (cm) Box Plot
petal length (cm)
7
6
5
4
3
2
1
petal width (cm) Box Plot
petal width (cm)
2.5
2.0
1.5
1.0
0.5
0.0