Group A

Assignment 10

Data Visualization III

Import all the required Python Libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
```

Load the Dataset into pandas dataframe.

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

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

[150 rows x 5 columns]
```

List down the features and their types (e.g., numeric, nominal) available in the dataset.

```
iris_tf.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
```

```
iris_tf.head()
```

```
   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
```

```
iris_tf.isnull().sum()
```

```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

```
iris_tf.describe()
```

```
       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
```

```
iris_tf.shape
```

```
(150, 5)
```

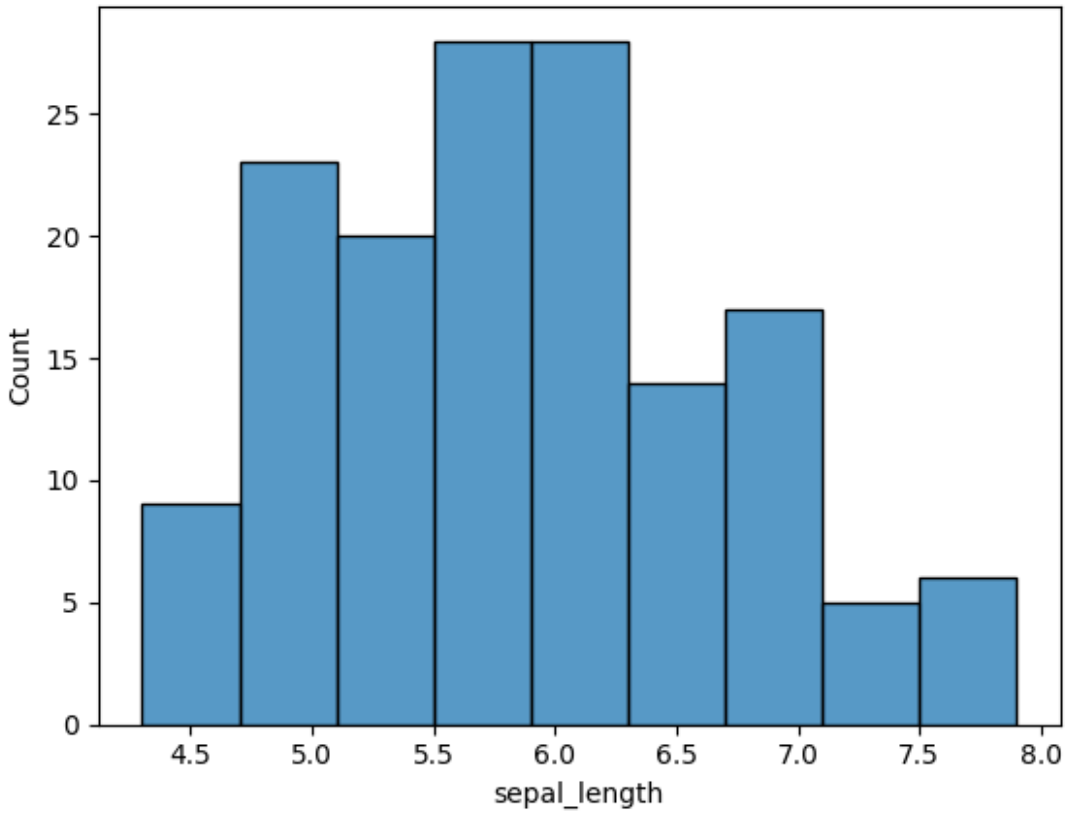
```
iris_tf.dtypes
```

```
sepal_length    float64
sepal_width     float64
petal_length    float64
petal_width     float64
species          object
dtype: object
```

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

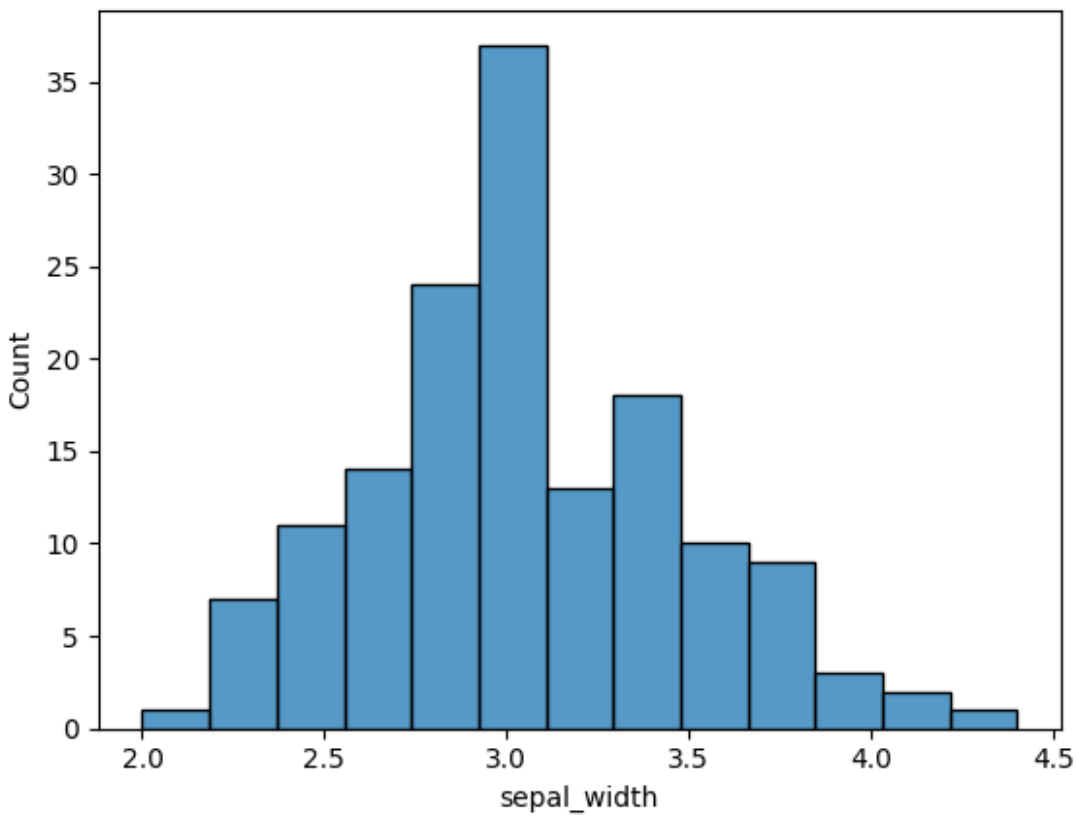
```
sns.histplot(iris_tf['sepal_length']) #Sepal length showing maximum
count between 5.5 and 6.3
```

```
<Axes: xlabel='sepal_length', ylabel='Count'>
```



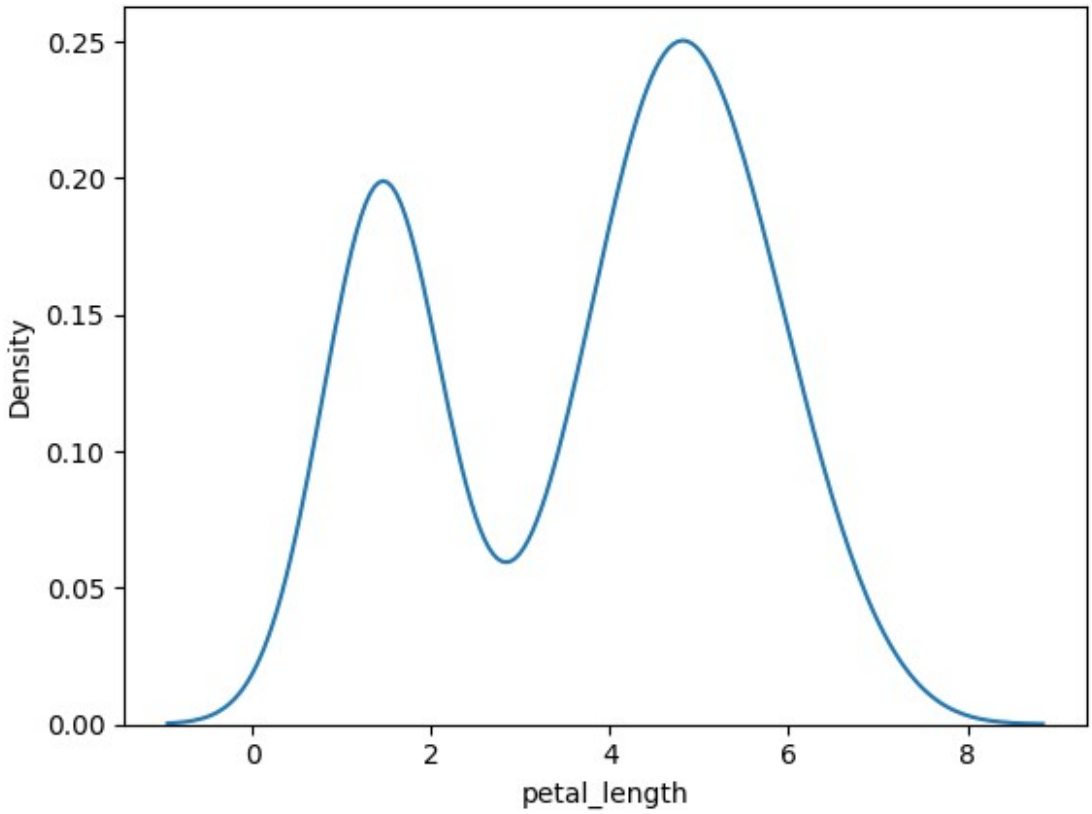

```
sns.histplot(iris_tf['sepal_width']) #Sepal width showing maximum
count on 3

<Axes: xlabel='sepal_width', ylabel='Count'>
```



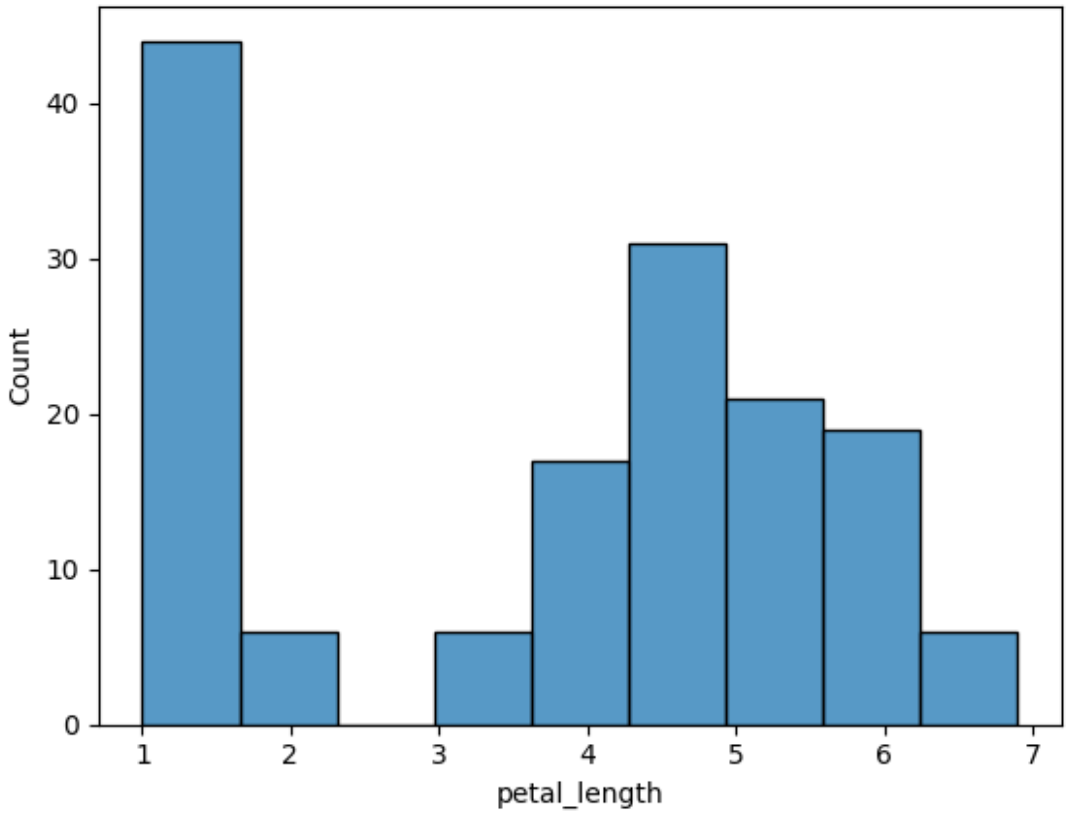

```
sns.kdeplot(iris_tf['petal_length']) #Petal length showing max density
on 5
```

```
<Axes: xlabel='petal_length', ylabel='Density'>
```




```
sns.histplot(iris_tf['petal_length']) #Petal length is max on 1
```

```
<Axes: xlabel='petal_length', ylabel='Count'>
```




```
iris_tf['sepal_length'].skew() #Normally Distributed
```

```
0.3149109566369728
```

```
iris_tf['sepal_width'].skew() #Normally Distributed
```

```
0.31896566471359966
```

```
iris_tf['petal_length'].skew() #Normally Distributed
```

```
-0.27488417975101276
```

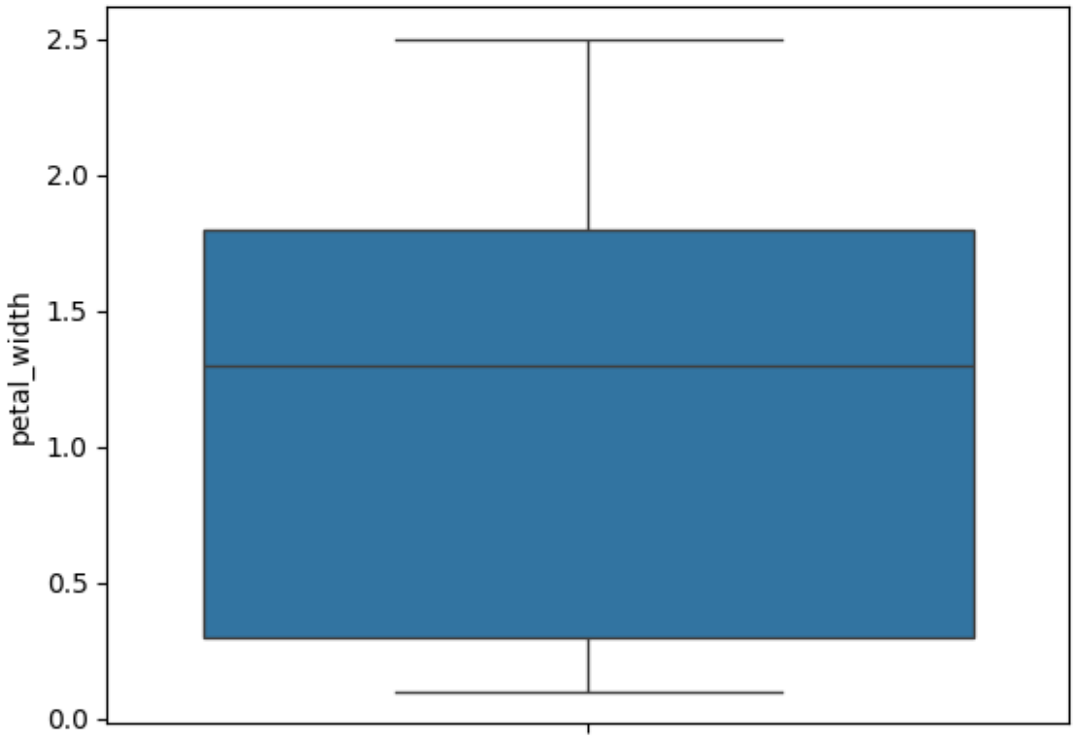
```
iris_tf['petal_width'].skew() #Normally Distributed
```

```
-0.10296674764898116
```

Create a boxplot for each feature in the dataset

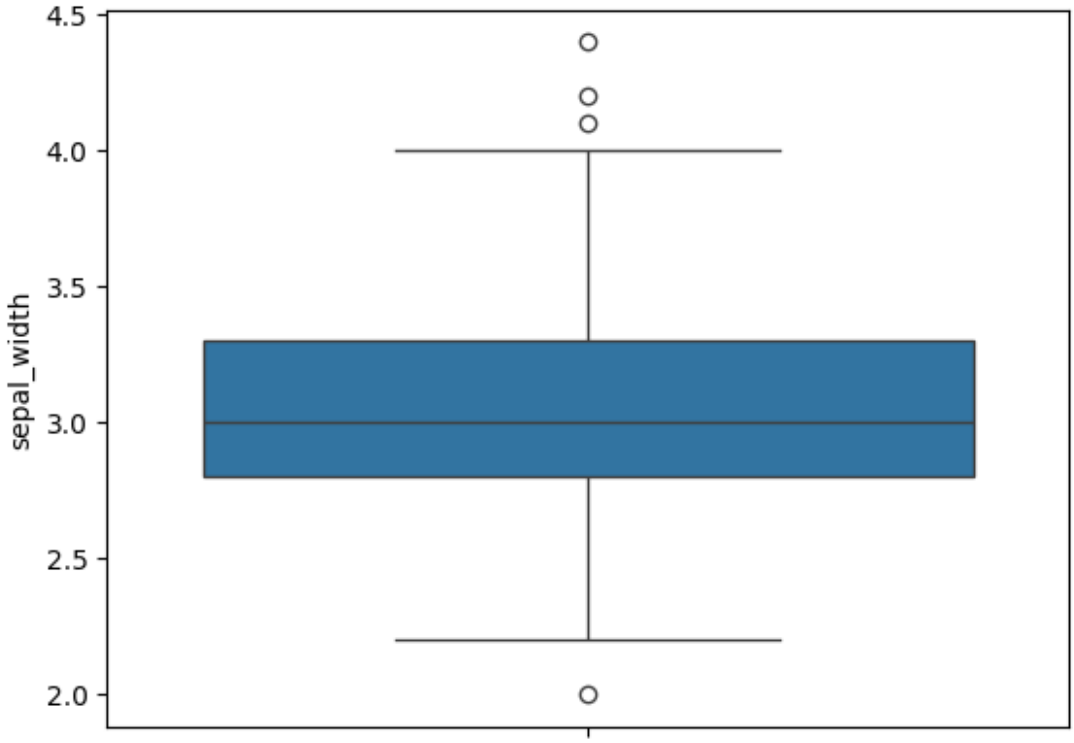
```
sns.boxplot(iris_tf['petal_width']) #Petal width has no outliers
```

```
<Axes: ylabel='petal_width'>
```



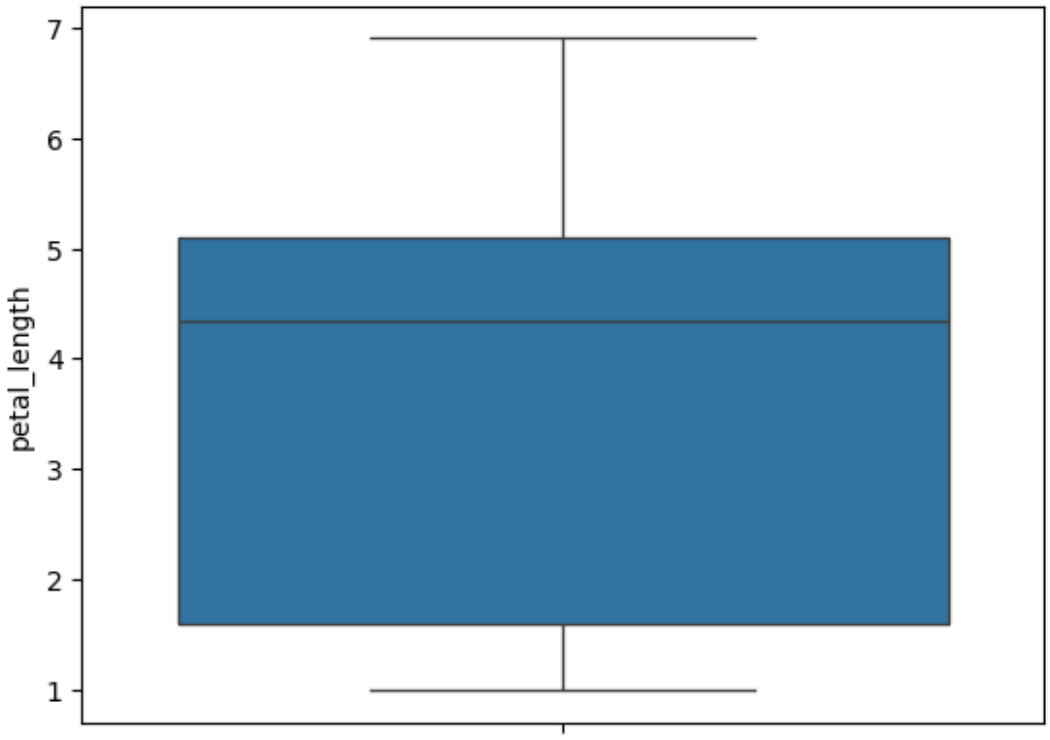

```
sns.boxplot(iris_tf['sepal_width']) #Sepal width has outliers on the
upper fence and lower fence
```

```
<Axes: ylabel='sepal_width'>
```



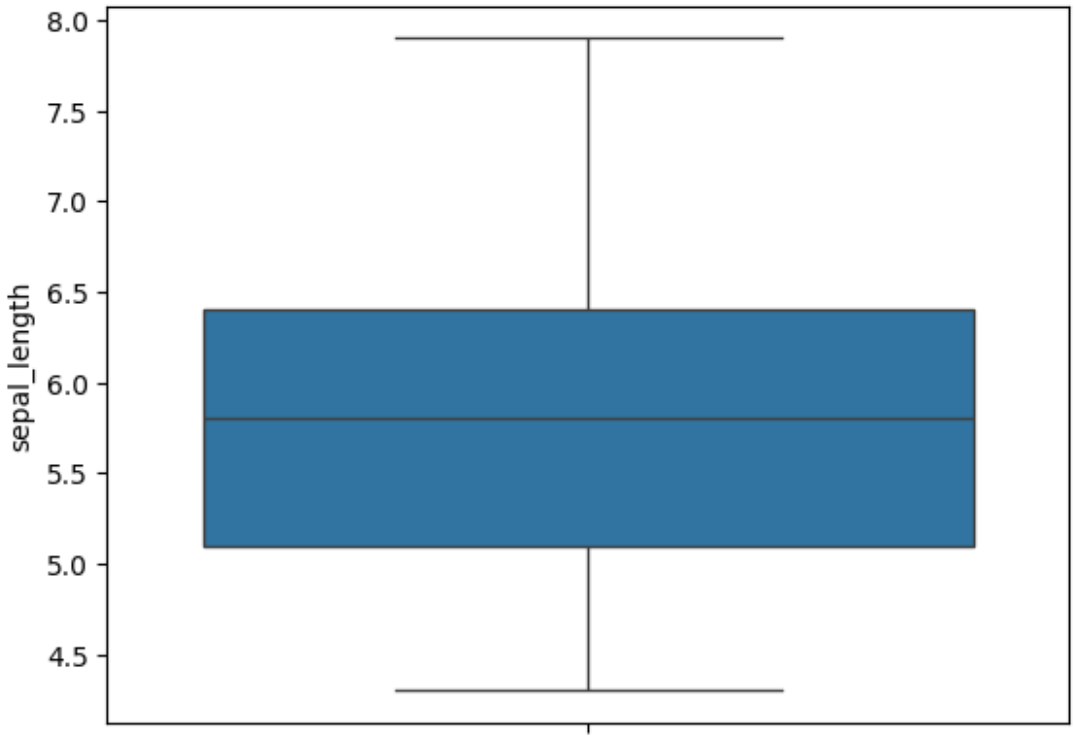

```
sns.boxplot(iris_tf['petal_length']) #Petal length has no outliers
```

```
<Axes: ylabel='petal_length'>
```



```
sns.boxplot(iris_tf['sepal_length']) #Sepal length has no outliers
```

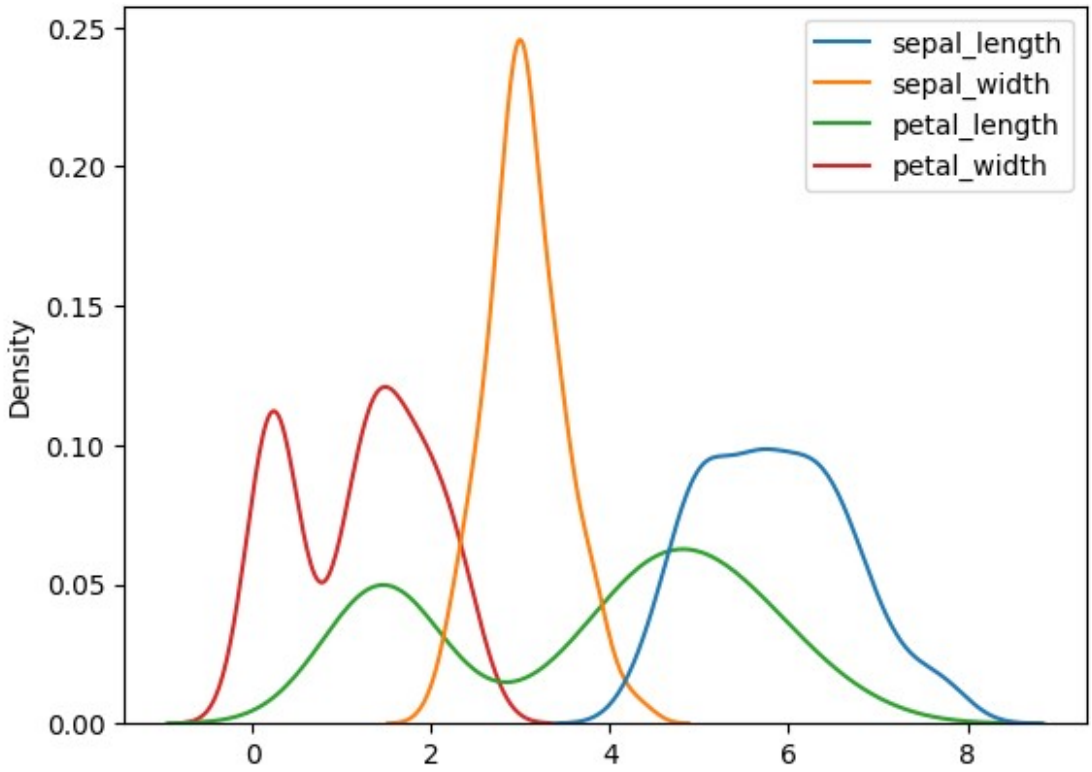
```
<Axes: ylabel='sepal_length'>
```



Compare distributions and identify outliers

```
sns.kdeplot(iris_tf) #Sepal width showing more density
```

```
<Axes: ylabel='Density'>
```




```
iris_tf[(iris_tf['sepal_width']>4.0) | (iris_tf['sepal_width']<2.1)]
```

	sepal_length	sepal_width	petal_length	petal_width	species
15	5.7	4.4	1.5	0.4	setosa
32	5.2	4.1	1.5	0.1	setosa
33	5.5	4.2	1.4	0.2	setosa
60	5.0	2.0	3.5	1.0	versicolor