practical-10

March 11, 2024

```
[ ]: import pandas as pd
     import numpy as np
     import seaborn as sns
     import matplotlib.pyplot as plt
```

```
[ ]: df=sns.load_dataset('iris')
```

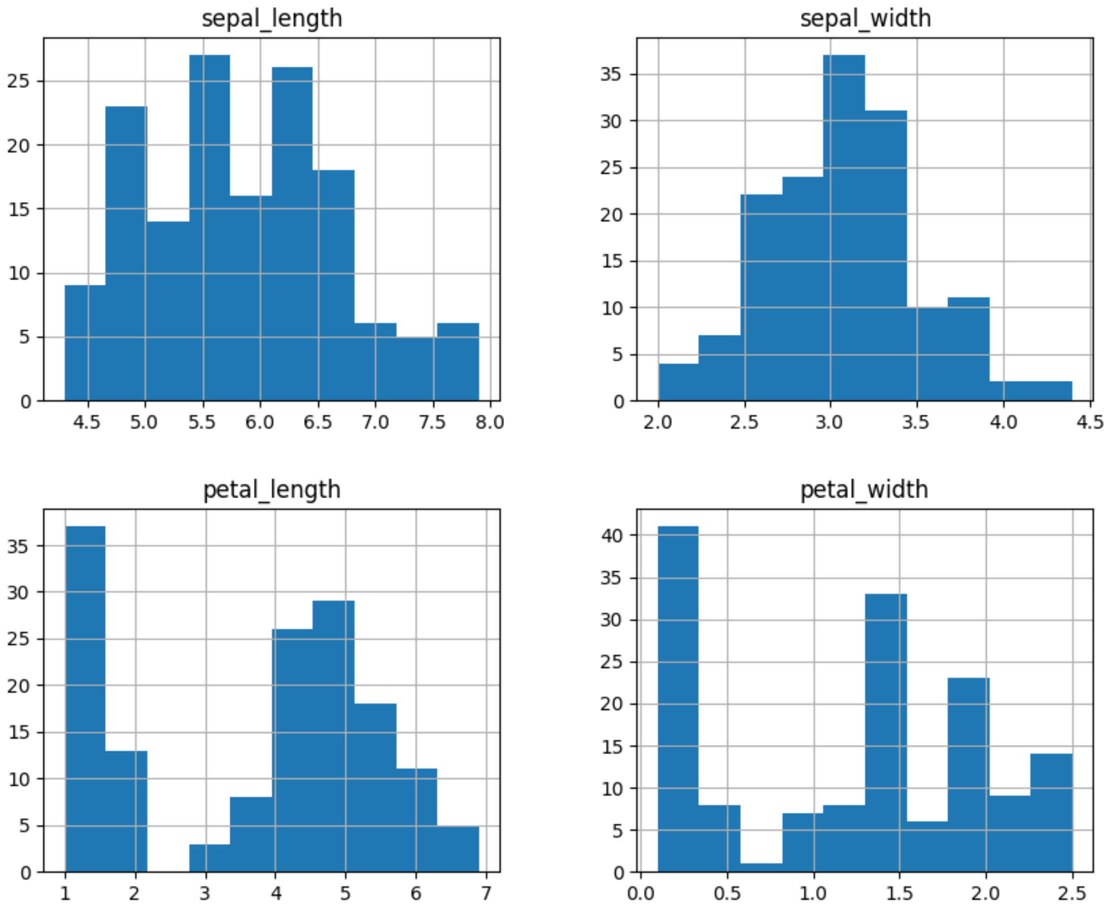
```
[ ]: df.head()
```

```
[ ]:    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
```

```
[ ]: df.dtypes
```

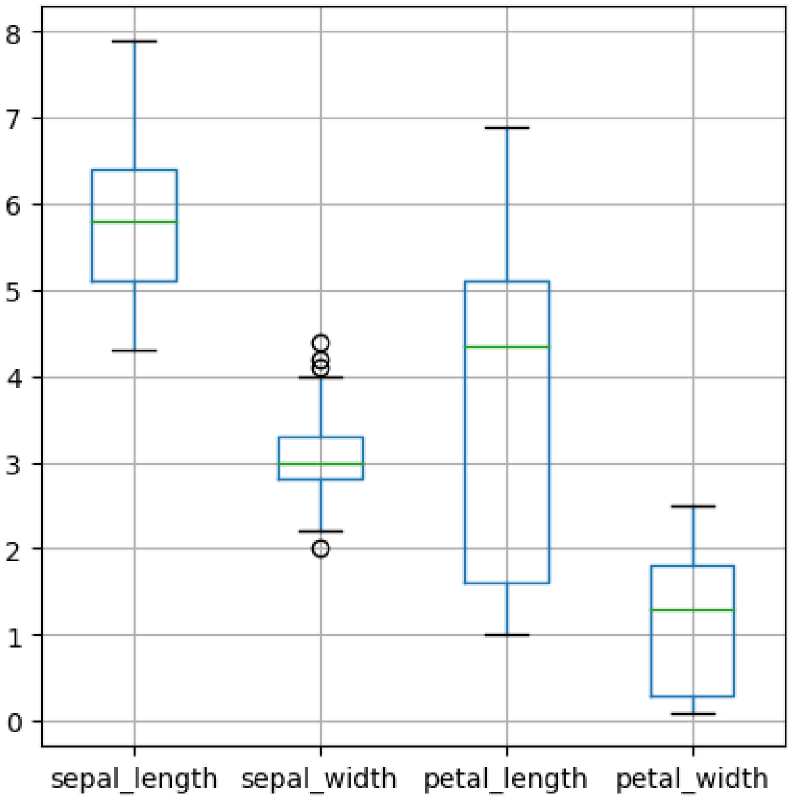
```
[ ]: sepal_length    float64
     sepal_width     float64
     petal_length    float64
     petal_width     float64
     species          object
     dtype: object
```

```
[ ]: df.hist(figsize=(10,8))
     plt.show()
```



[ ]:

```
df.boxplot(figsize=(5,5))
plt.show()
```



[ ]:

```
sort_data = np.sort(df['sepal_width'])
mean = np.mean(sort_data)
std = np.std(sort_data)
print('mean of the dataset is', mean)
print('std. deviation is', std)
threshold = 1
outlier = []
for i in sort_data:
        z = (i-mean)/std
        if z > threshold:
                outlier.append(i)
print('outlier in dataset is', outlier)
```

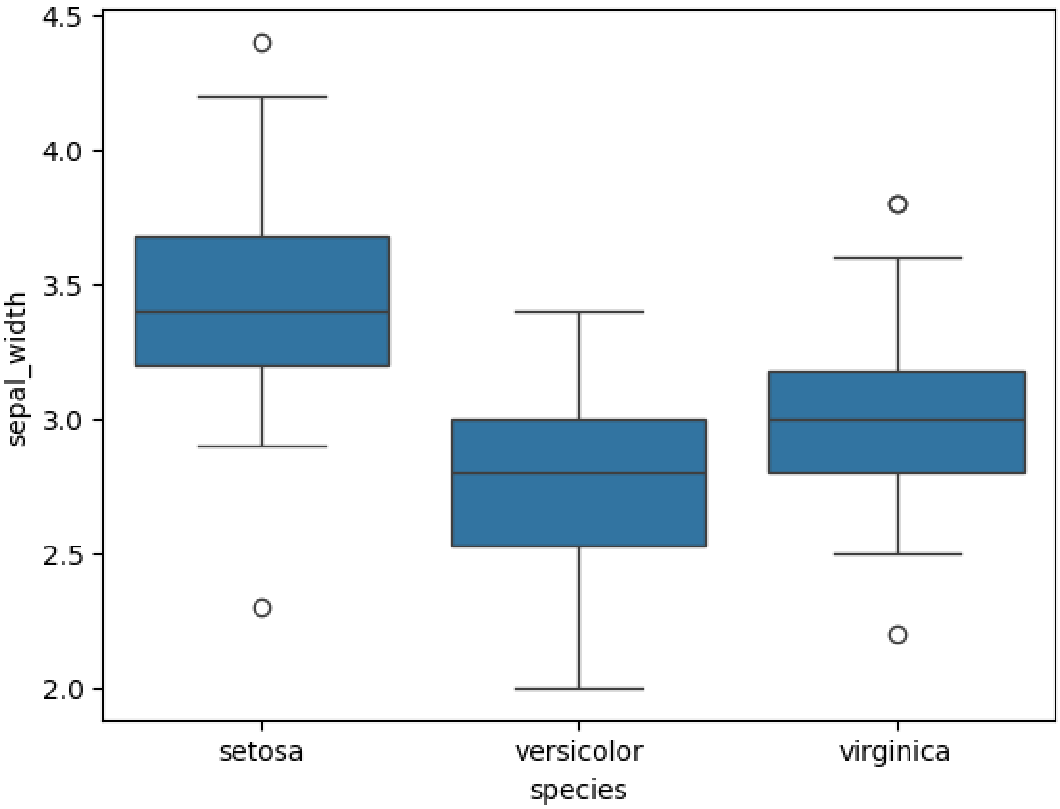
```
mean of the dataset is 3.0573333333333332
std. deviation is 0.4344109677354946
outlier in dataset is [3.5, 3.5, 3.5, 3.5, 3.5, 3.5, 3.6, 3.6, 3.6, 3.6, 3.7,
3.7, 3.7, 3.8, 3.8, 3.8, 3.8, 3.8, 3.8, 3.9, 3.9, 4.0, 4.1, 4.2, 4.4]
```

[ ]:

```
sns.boxplot(data=df,x="species",y="sepal_width")
```

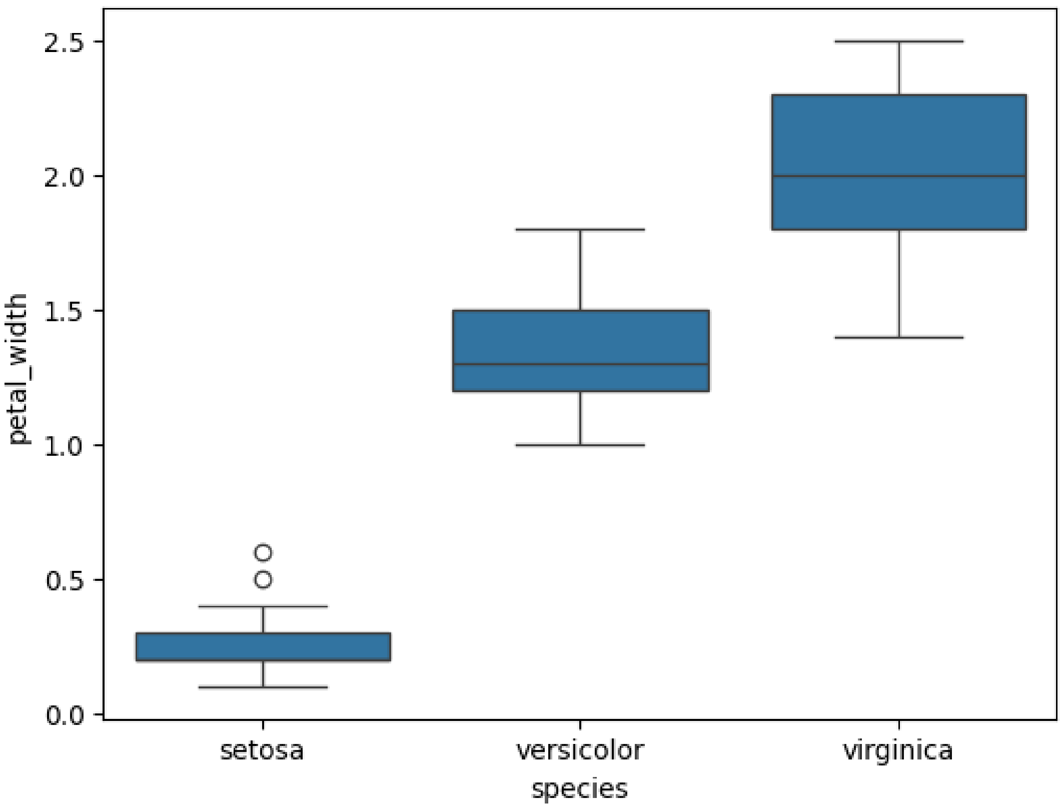
[ ]:

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



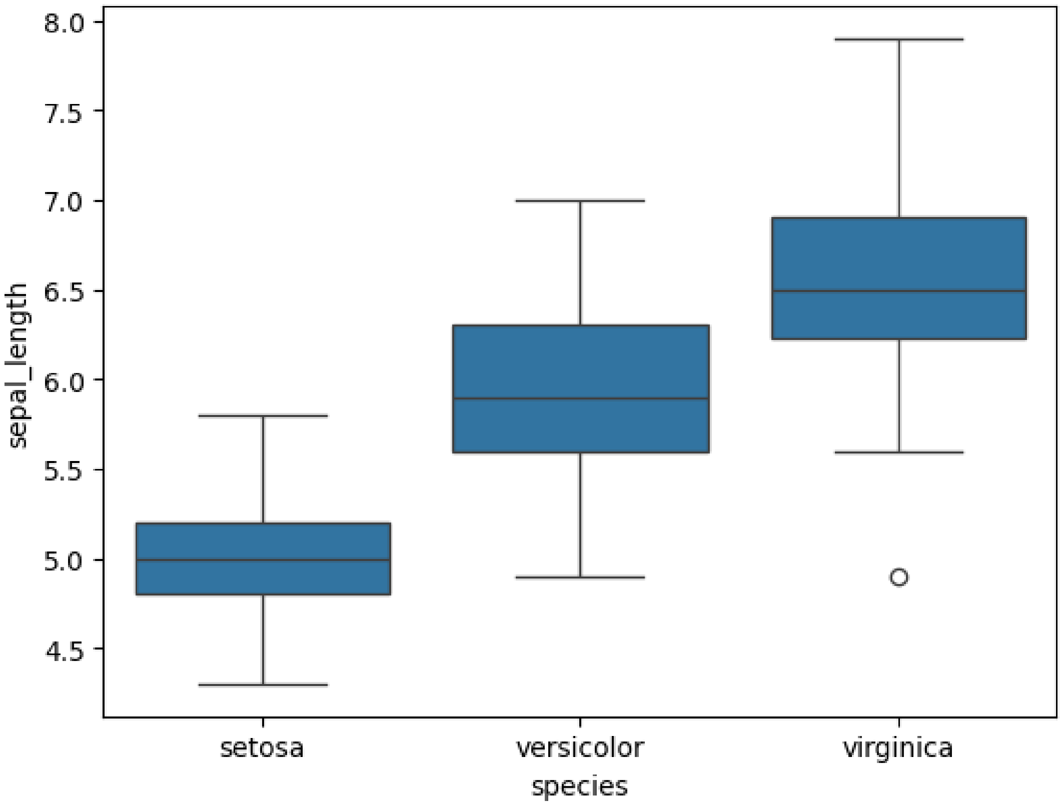
```
[ ]: sns.boxplot(data=df,x="species",y="petal_width")
```

```
[ ]: <Axes: xlabel='species', ylabel='petal_width'>
```



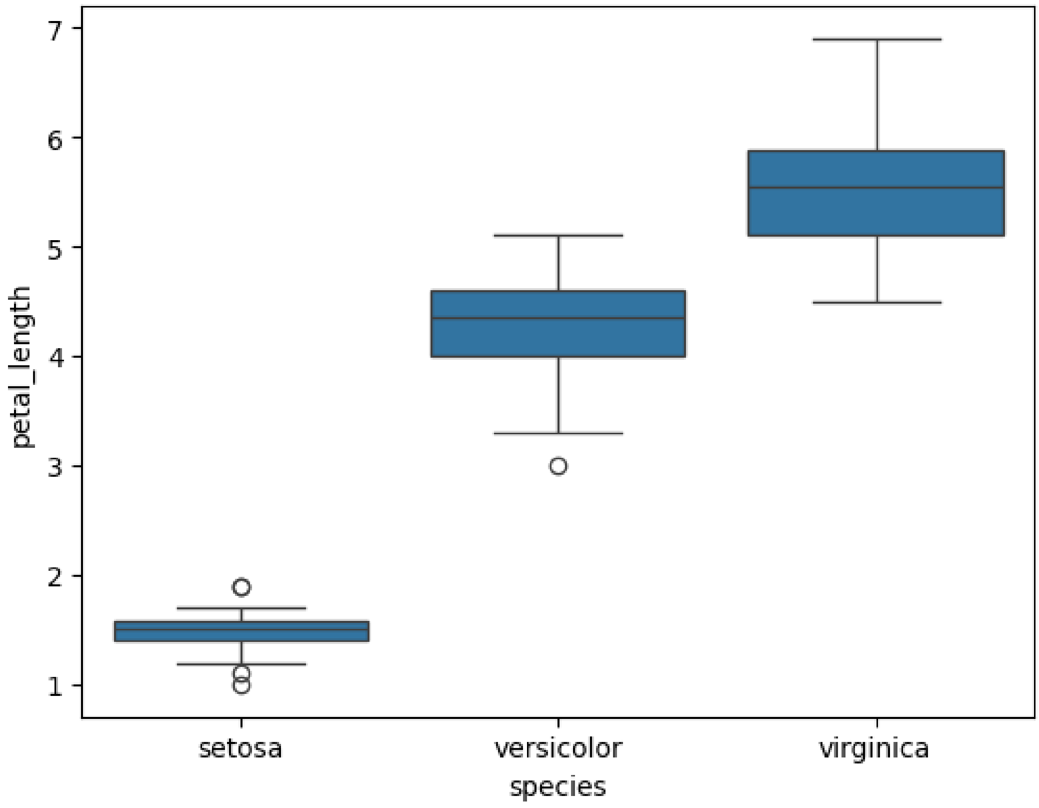
```
[ ]: sns.boxplot(data=df,x="species",y="sepal_length")
```

```
[ ]: <Axes: xlabel='species', ylabel='sepal_length'>
```



```
[ ]: sns.boxplot(data=df,x="species",y="petal_length")
```

```
[ ]: <Axes: xlabel='species', ylabel='petal_length'>
```



[ ]:

```
Q1 = np.percentile(sort_data, 25, interpolation = 'midpoint')
Q3 = np.percentile(sort_data, 75, interpolation = 'midpoint')

print('Q1 25 percentile of the given data is, ', Q1)
print('Q1 75 percentile of the given data is, ', Q3)
IQR = Q3 - Q1
print('Interquartile range is', IQR)
low_lim = Q1 - 1.5 * IQR
up_lim = Q3 + 1.5 * IQR
print('low_limit is', low_lim)
print('up_limit is', up_lim)
outlier =[]
for x in sort_data:
        if ((x> up_lim) or (x<low_lim)):
                outlier.append(x)
print(' outlier in the dataset is', outlier)
```

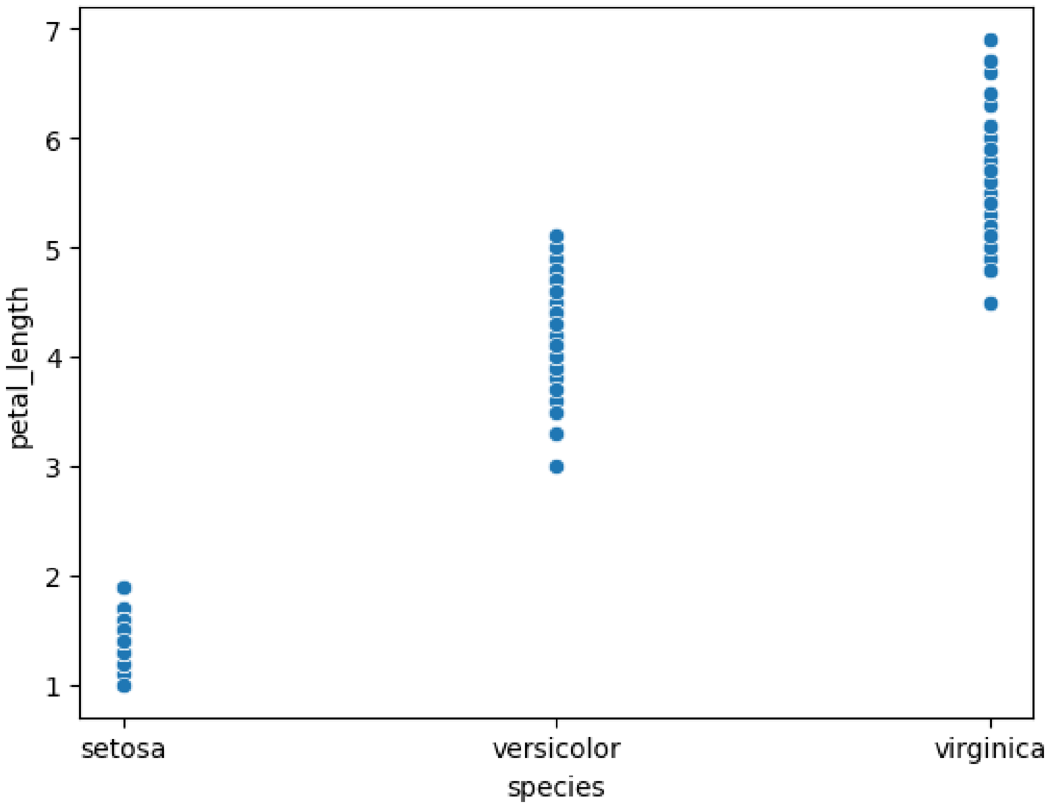
```
Q1 25 percentile of the given data is,  2.8
Q1 75 percentile of the given data is,  3.3
Interquartile range is 0.5
low_limit is 2.05
```

```
up_limit is 4.05
 outlier in the dataset is [2.0, 4.1, 4.2, 4.4]
```

```
<ipython-input-16-81e3a60e931d>:1: DeprecationWarning: the `interpolation=`
argument to percentile was renamed to `method=`, which has additional options.
Users of the modes 'nearest', 'lower', 'higher', or 'midpoint' are encouraged to
review the method they used. (Deprecated NumPy 1.22)
  Q1 = np.percentile(sort_data, 25, interpolation = 'midpoint')
<ipython-input-16-81e3a60e931d>:2: DeprecationWarning: the `interpolation=`
argument to percentile was renamed to `method=`, which has additional options.
Users of the modes 'nearest', 'lower', 'higher', or 'midpoint' are encouraged to
review the method they used. (Deprecated NumPy 1.22)
  Q3 = np.percentile(sort_data, 75, interpolation = 'midpoint')
```

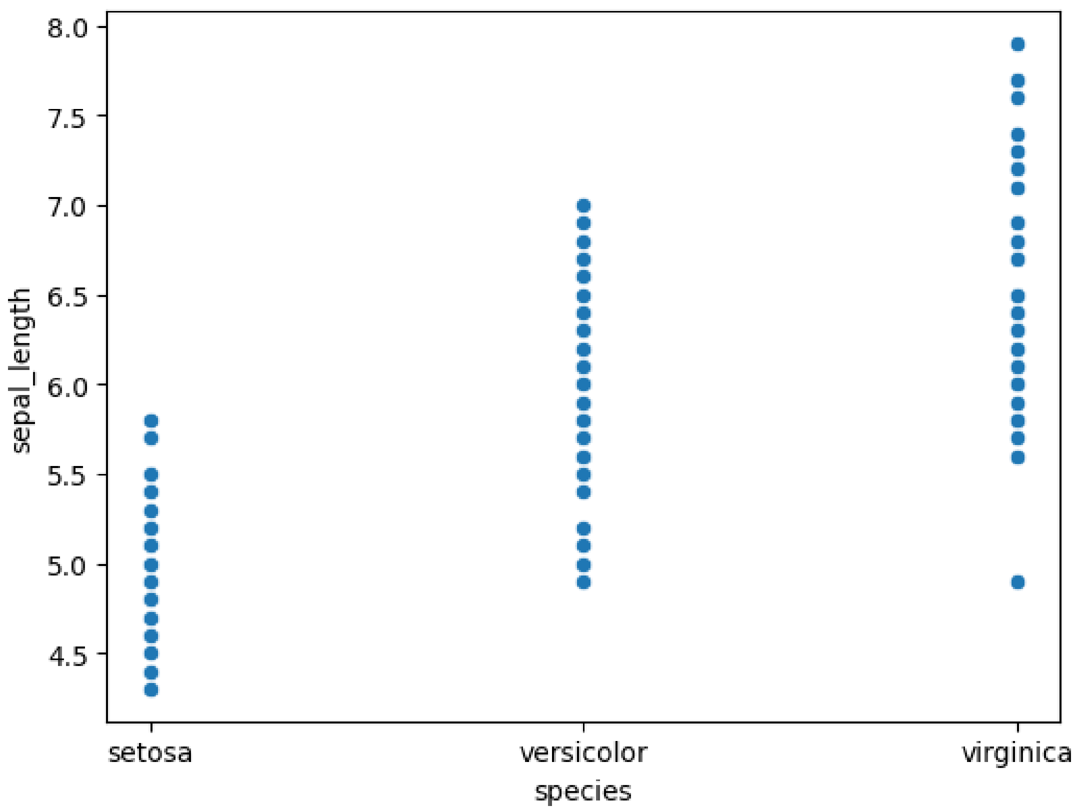
```
[ ]: sns.scatterplot(data=df,x="species",y="petal_length")
```

```
[ ]: <Axes: xlabel='species', ylabel='petal_length'>
```



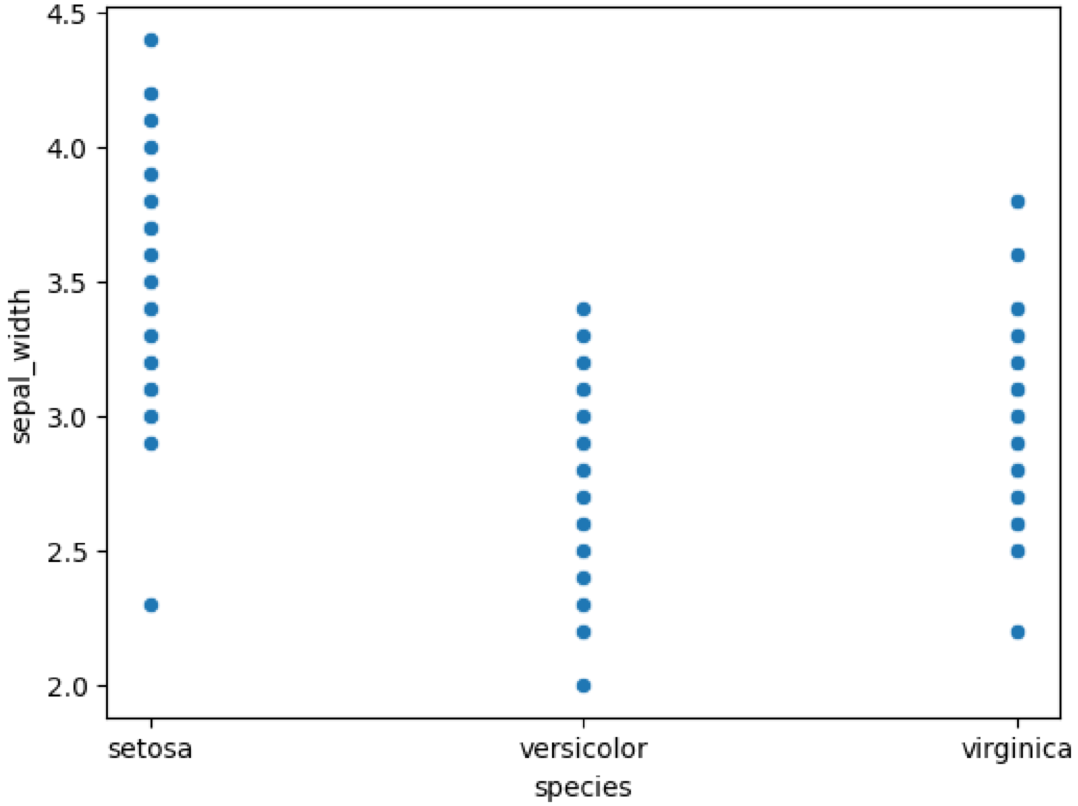
```
[ ]: sns.scatterplot(data=df,x="species",y="sepal_length")
```

```
[ ]: <Axes: xlabel='species', ylabel='sepal_length'>
```



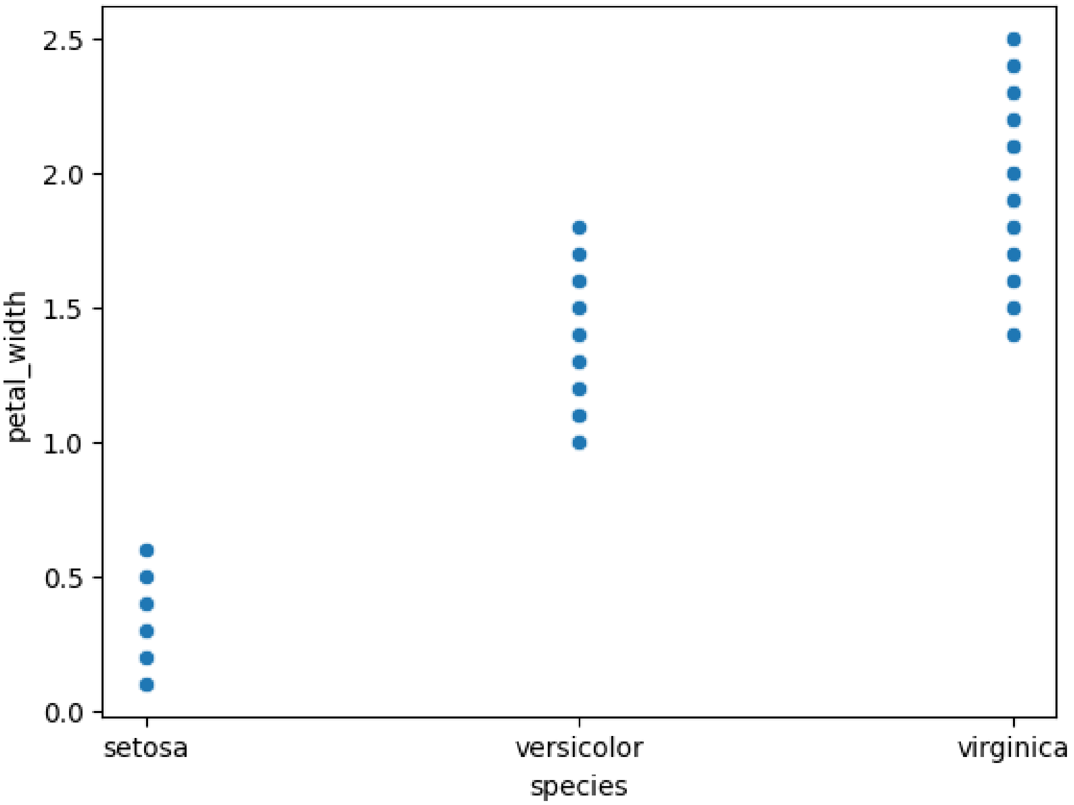
```
[ ]: sns.scatterplot(data=df,x="species",y="sepal_width")
```

```
[ ]: <Axes: xlabel='species', ylabel='sepal_width'>
```



```
[ ]: sns.scatterplot(data=df,x="species",y="petal_width")
```

```
[ ]: <Axes: xlabel='species', ylabel='petal_width'>
```


2.5
2.0
1.5
1.0
0.5
0.0
petal_width
setosa
versicolor
virginica
species