

practical-exam-13-14

May 23, 2023

```
[1]: from google.colab import drive
drive.mount('/content/drive')
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

1 Problem Statement 13 and 14

Use the Iris flower dataset and perform the following operations. 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 boxplot for each feature in the dataset. 4. Compare distributions and identify outliers.

```
[3]: import pandas as pd
```

```
[4]: df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/exam_datasets/
↳5-8-13-14.iris.csv')
```

```
[5]: df.info()
```

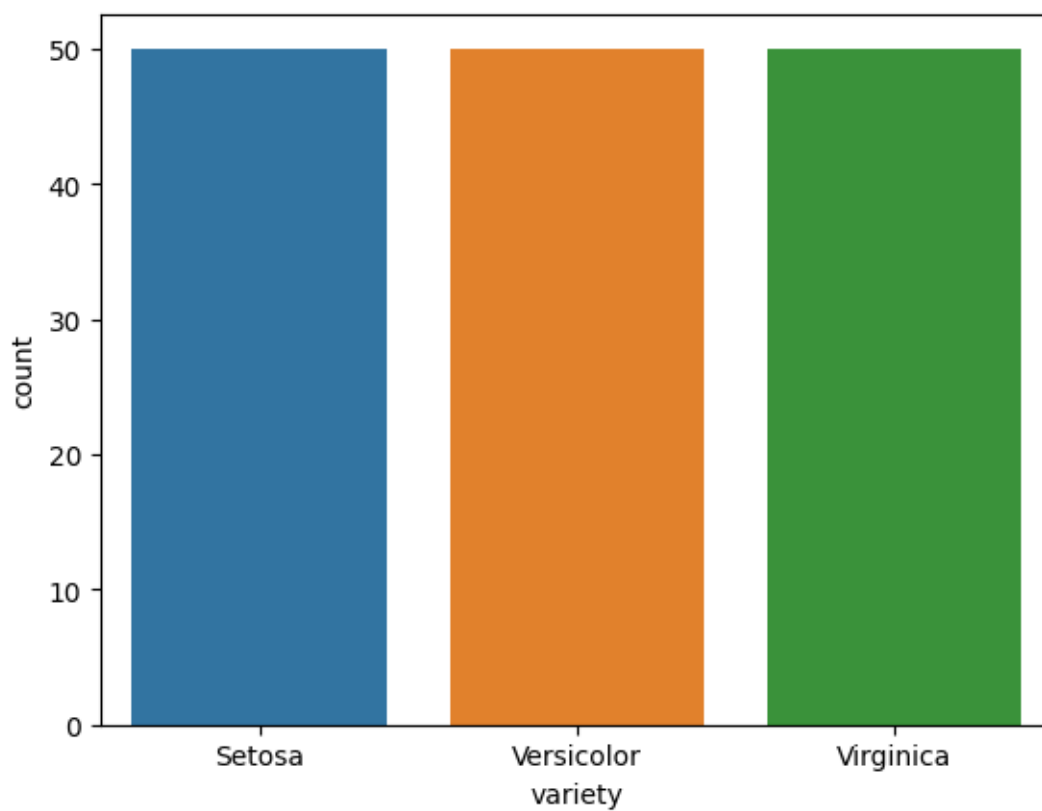
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   sepal.length    150 non-null   float64
1   sepal.width     150 non-null   float64
2   petal.length    150 non-null   float64
3   petal.width     150 non-null   float64
4   variety         150 non-null   object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
[7]: df.value_counts("variety")
```

```
[7]: variety
Setosa      50
Versicolor  50
Virginica   50
dtype: int64
```

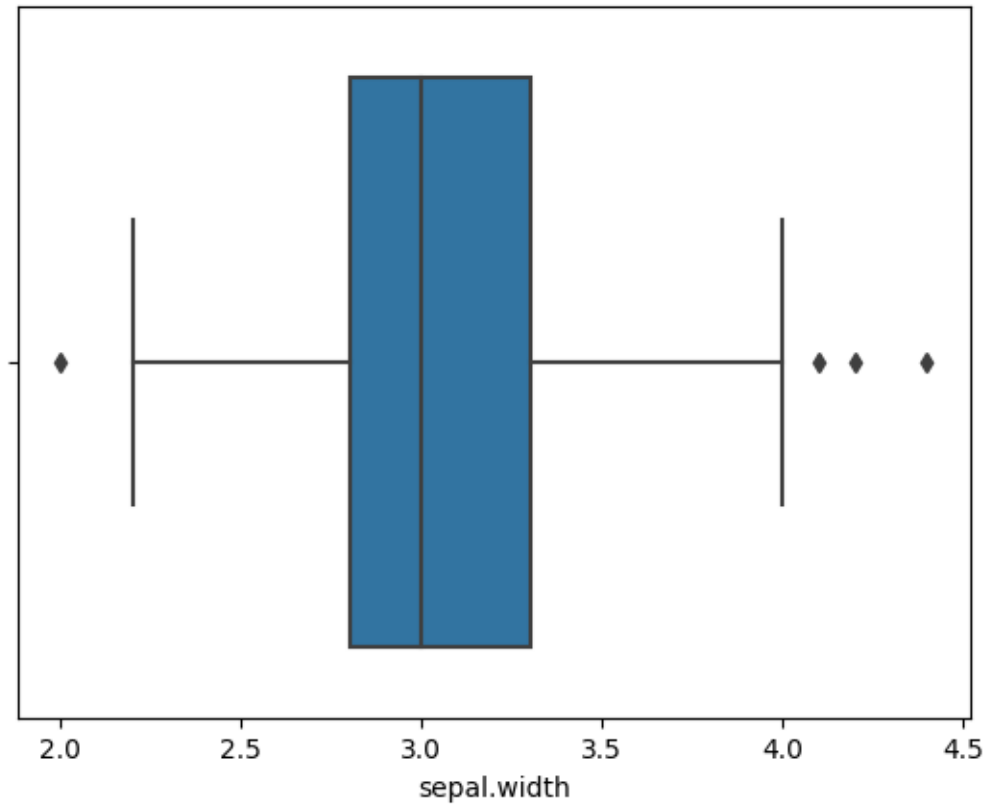
```
[8]: import seaborn as sns
import matplotlib.pyplot as plt
```

```
[9]: sns.countplot(x="variety", data=df)
plt.show()
```



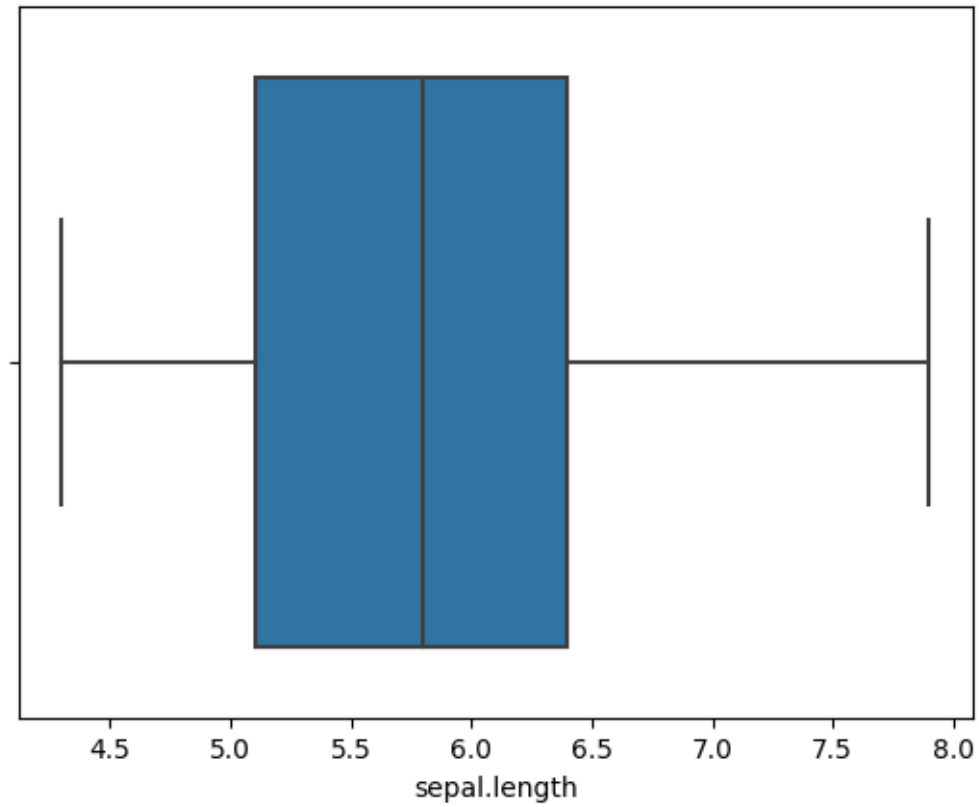
```
[12]: sns.boxplot(x='sepal.width', data=df)
```

```
[12]: <Axes: xlabel='sepal.width'>
```



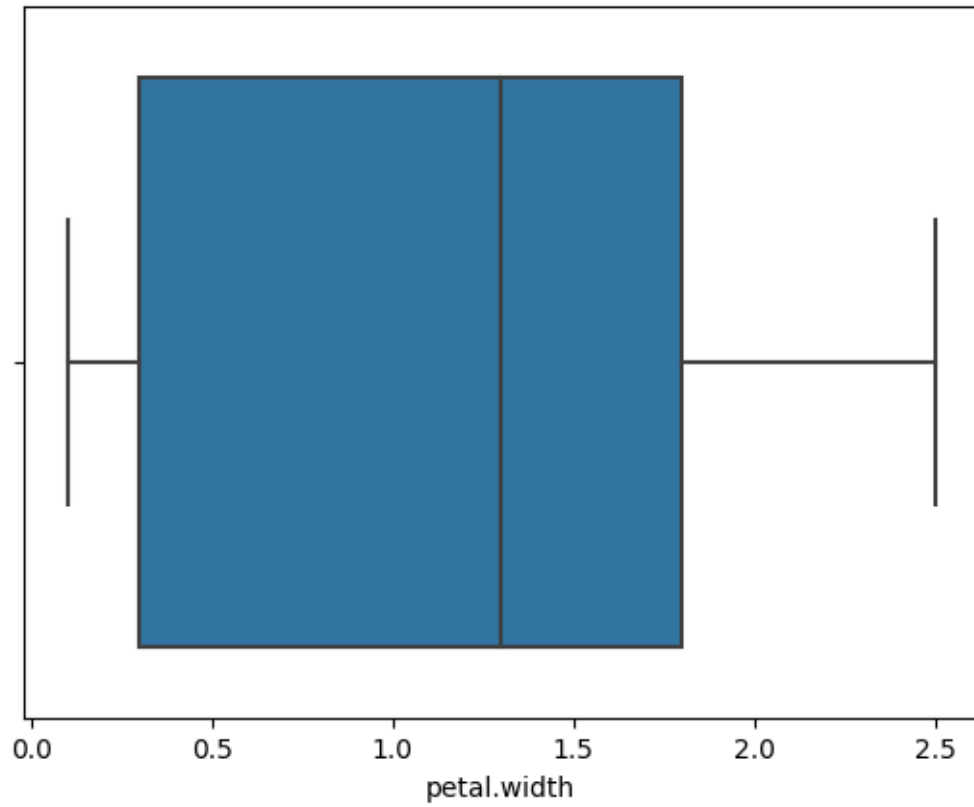
```
[14]: sns.boxplot(x='sepal.length', data=df)
```

```
[14]: <Axes: xlabel='sepal.length'>
```



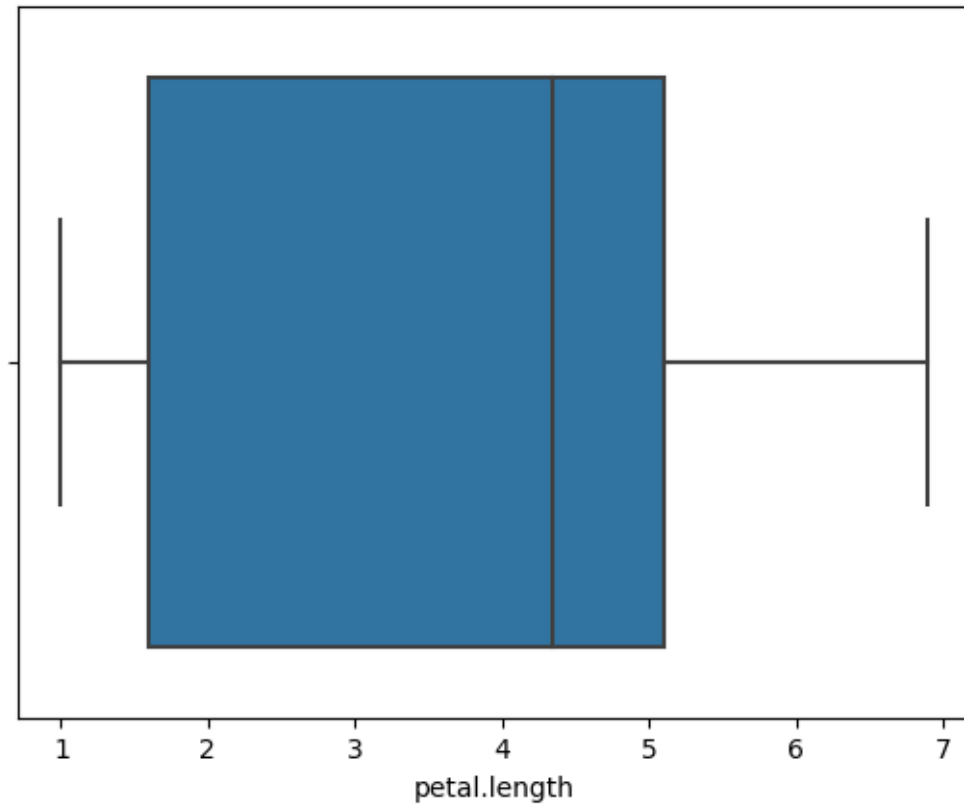
```
[15]: sns.boxplot(x='petal.width', data=df)
```

```
[15]: <Axes: xlabel='petal.width'>
```



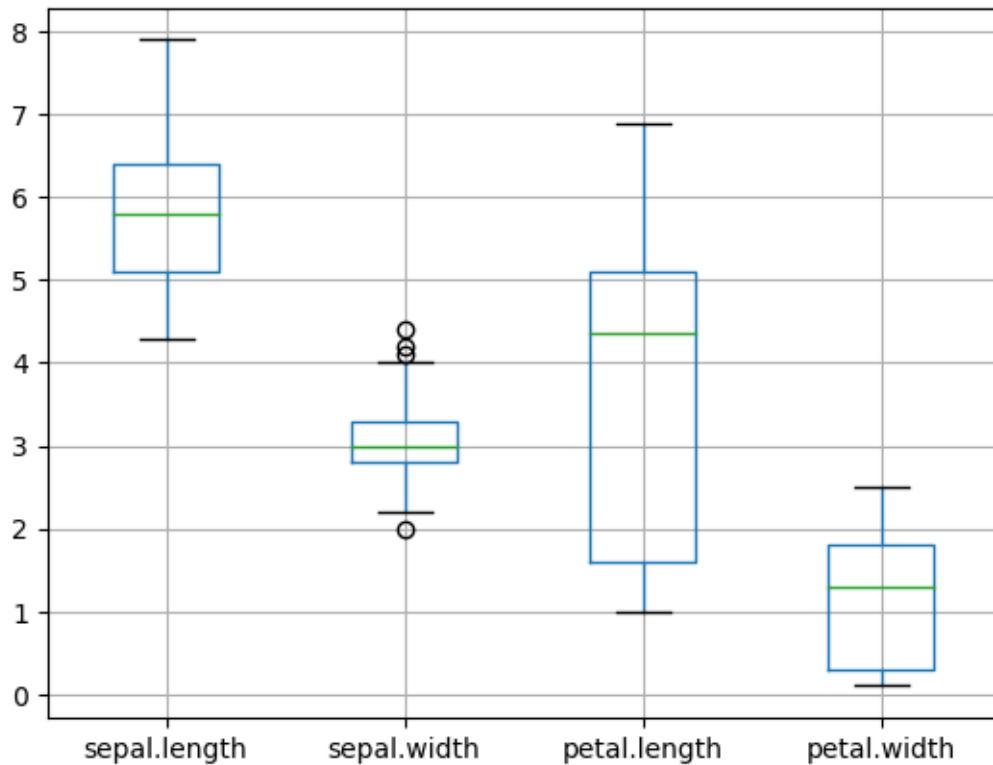
```
[16]: sns.boxplot(x='petal.length', data=df)
```

```
[16]: <Axes: xlabel='petal.length'>
```



```
[17]: df.boxplot()
```

```
[17]: <Axes: >
```



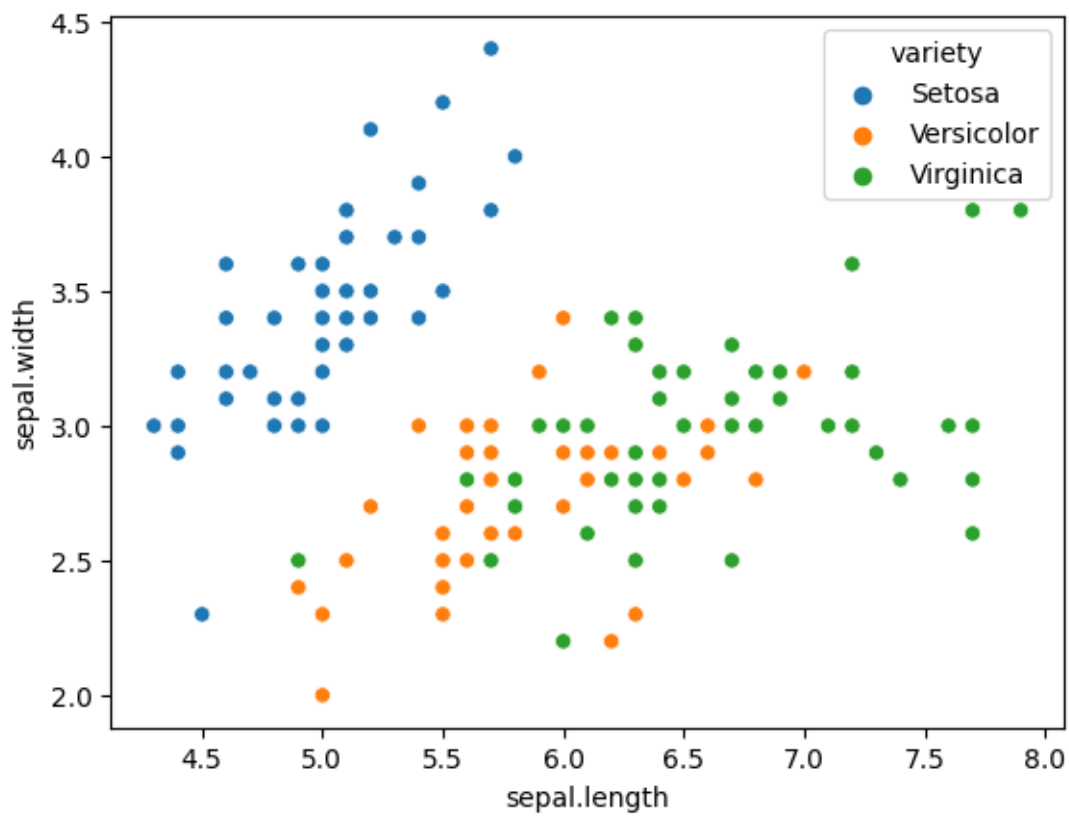
```
[20]: df.describe()
```

```
[20]:
```

| | 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 |

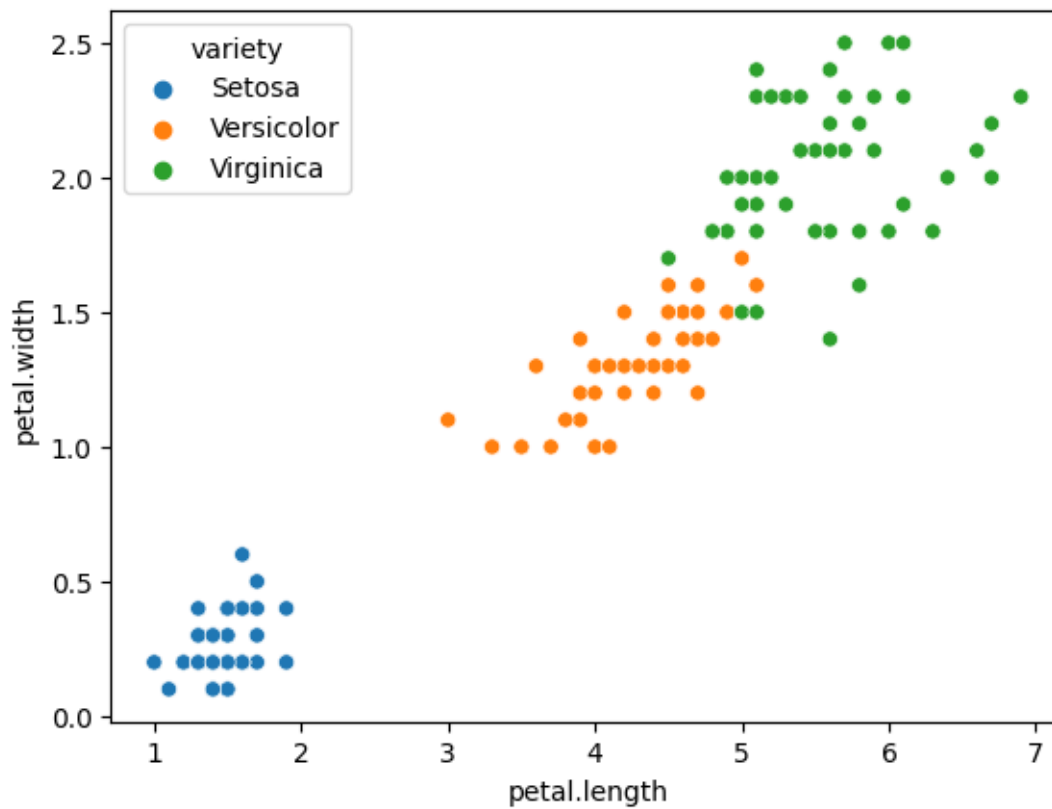
```
[21]: sns.scatterplot(x='sepal.length', y='sepal.width', hue='variety', data=df)
```

```
[21]: <Axes: xlabel='sepal.length', ylabel='sepal.width'>
```



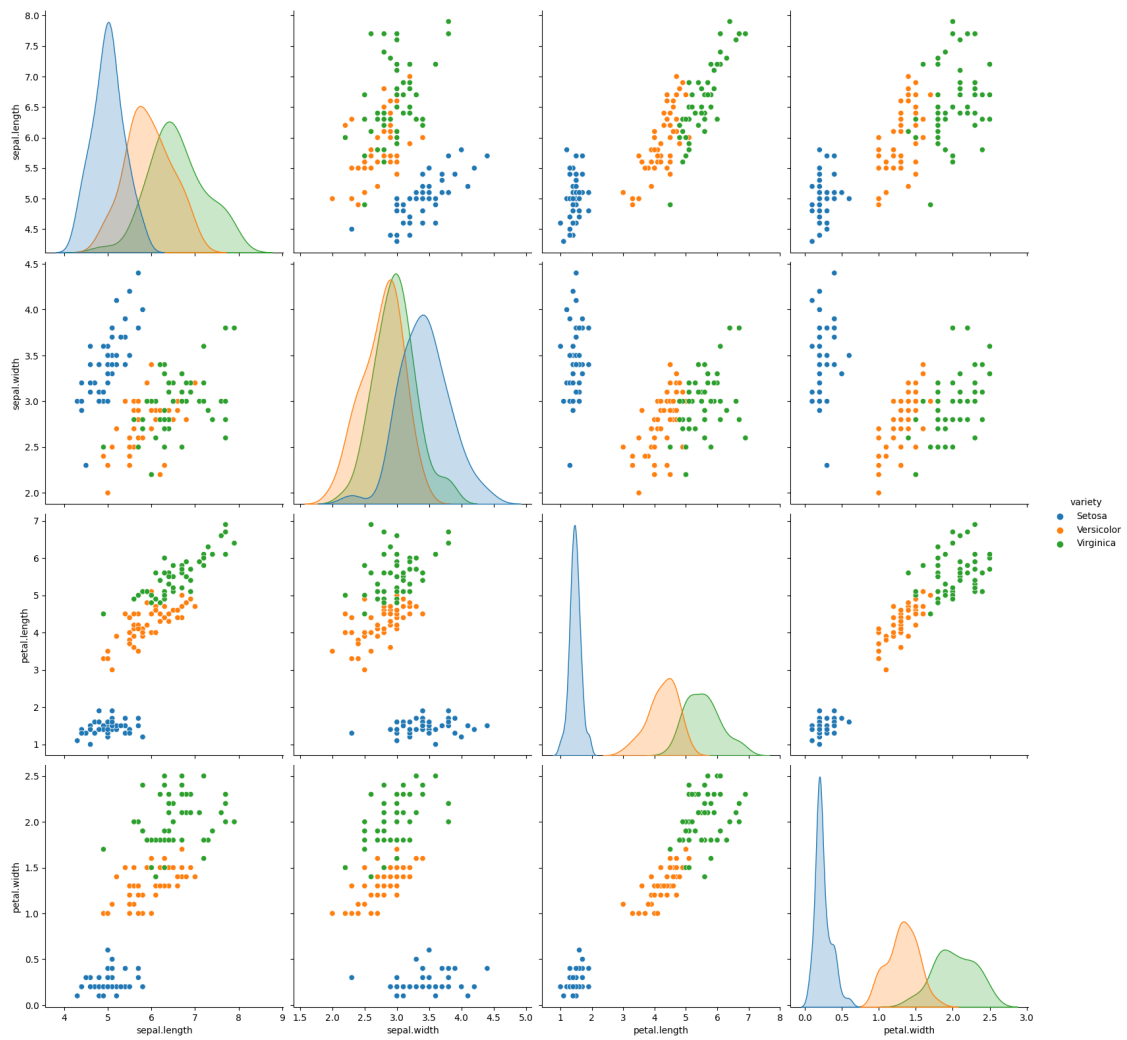
```
[22]: sns.scatterplot(x='petal.length', y='petal.width', hue='variety', data=df)
```

```
[22]: <Axes: xlabel='petal.length', ylabel='petal.width'>
```

```
[23]: sns.pairplot(df, hue='variety', height=4)
```

```
[23]: <seaborn.axisgrid.PairGrid at 0x7fd373ec7e20>
```



[]: