```
**Data Analytics - III**
```

In [ ]:

In [49]:

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

df1=pd.read_csv('/home/ubuntu/Yash/iris.csv')
df1
```

Out[49]:

	sepal.length	sepal.width	petal.length	petal.width	variety
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 × 5 columns

In [37]:

```
df=pd.DataFrame(df1)
df.head()
```

Out[37]:

	sepal.length	sepal.width	petal.length	petal.width	variety
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

In [38]:

```
df.describe()
```

Out[38]:

	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

In [39]:

```
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
```

In [40]:

```
df.columns
```

Out[40]:

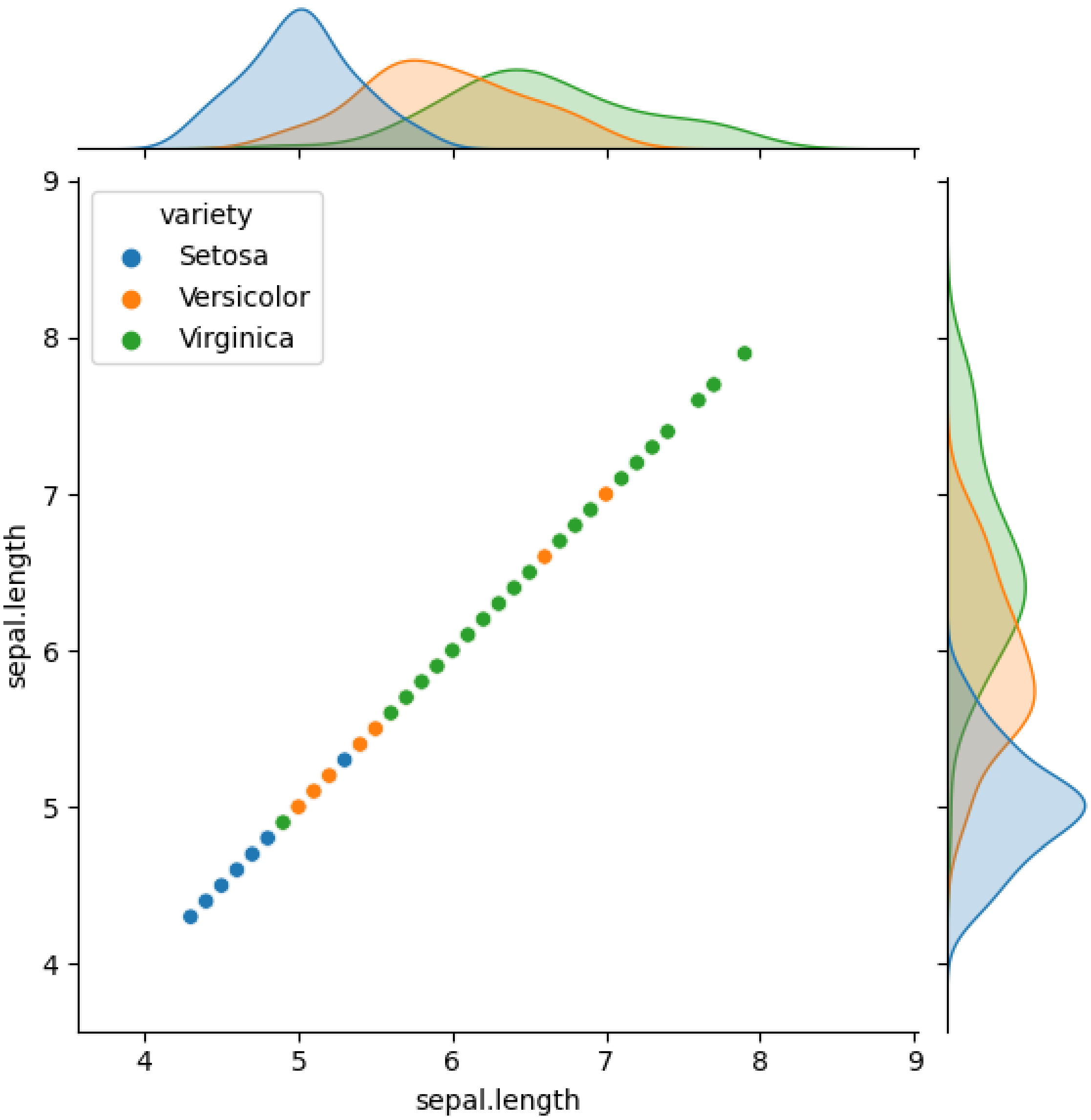
```
Index(['sepal.length', 'sepal.width', 'petal.length', 'petal.width',
       'variety'],
      dtype='object')
```

In [41]:

```
sns.jointplot(x='sepal.length',y='sepal.length',data=df,hue='variety')
```

Out[41]:

```
<seaborn.axisgrid.JointGrid at 0x7f7c186a2af0>
```

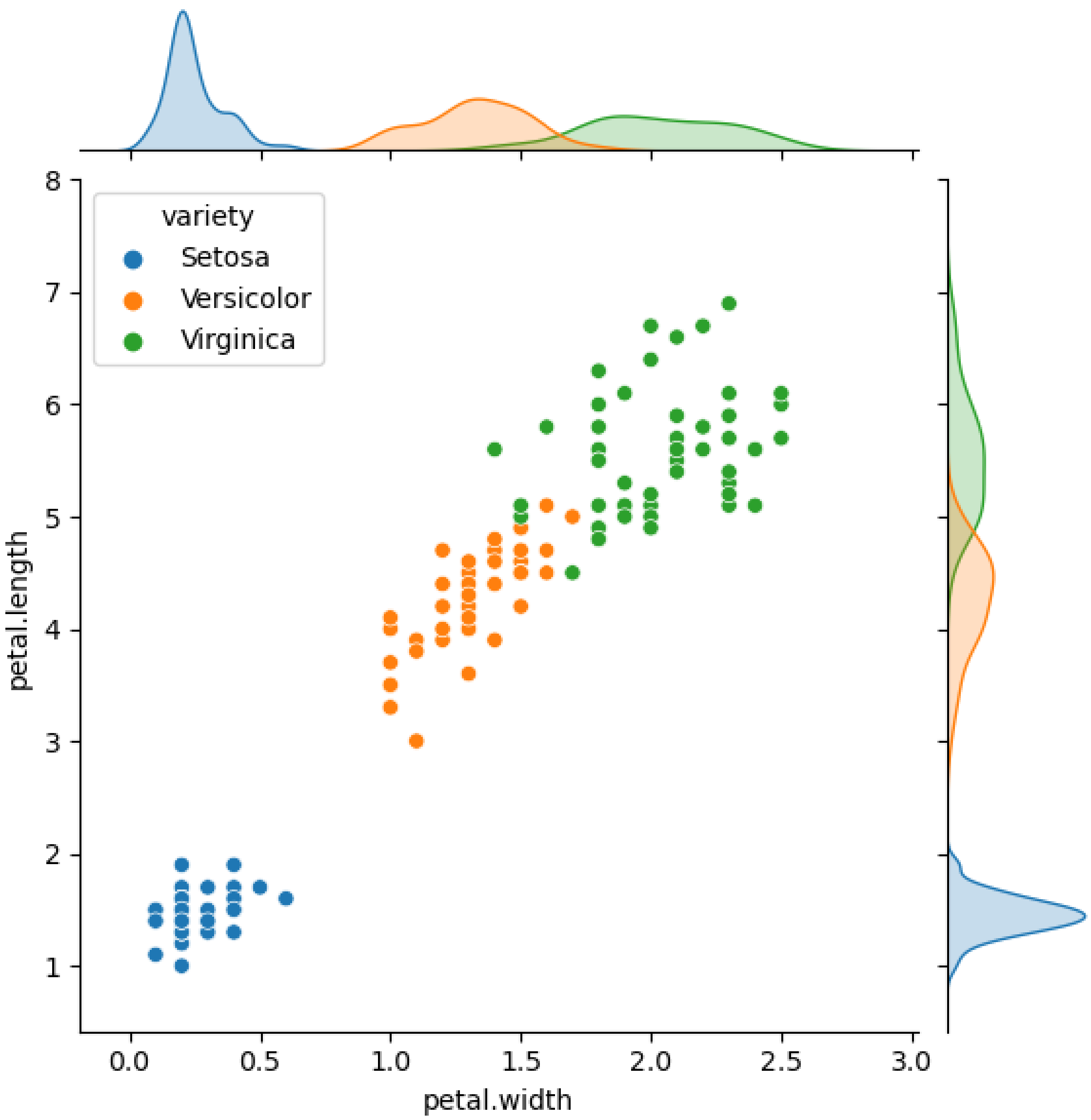


In [42]:

```
sns.jointplot(x='petal.width',y='petal.length',data=df,hue='variety')
```

Out[42]:

```
<seaborn.axisgrid.JointGrid at 0x7f7c17f04ac0>
```

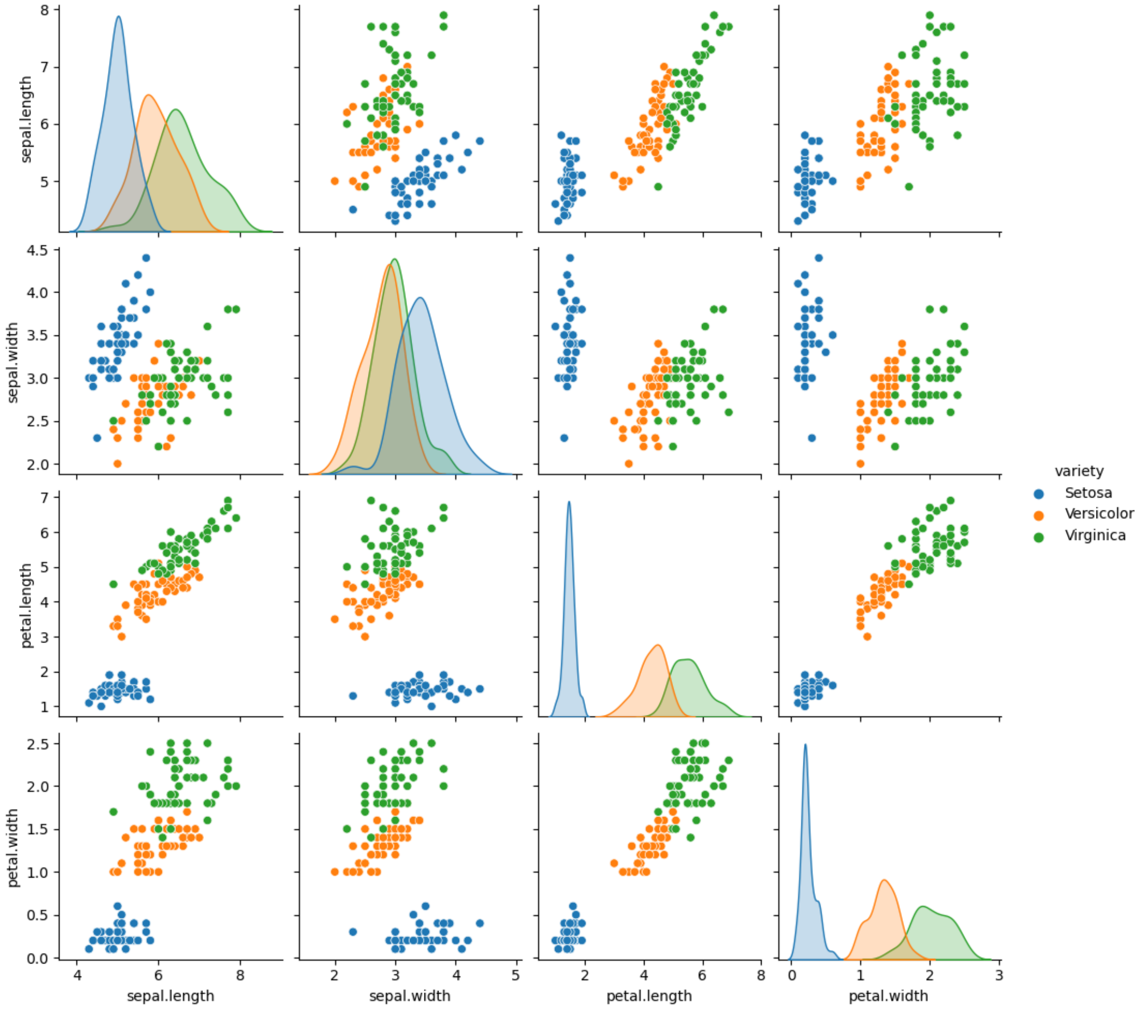


In [43]:

```
sns.pairplot(data=df,hue='variety')
```

Out[43]:

```
<seaborn.axisgrid.PairGrid at 0x7f7c161087f0>
```



In [44]:

```
from sklearn.model_selection import train_test_split
X=df[['sepal.length', 'sepal.width', 'petal.length', 'petal.width']]
Y=df['variety']
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.33, random_
from sklearn.naive_bayes import MultinomialNB
nb=MultinomialNB()
nb.fit(X_train,y_train)
predictions=nb.predict(X_test)
from sklearn.metrics import accuracy_score,classification_report,confusion_matrix
print('Classification Report')
print('\n')
print(classification_report(y_test,predictions))
```

```
Classification Report


              precision    recall  f1-score   support

      Setosa       1.00      1.00      1.00        19
  Versicolor       0.88      1.00      0.94        15
   Virginica       1.00      0.88      0.93        16

    accuracy                           0.96        50
   macro avg       0.96      0.96      0.96        50
weighted avg       0.96      0.96      0.96        50
```

In [45]:

```
print('Accuracy')
print(accuracy_score(y_test,predictions))
```

```
Accuracy
0.96
```

In [48]:

```
print('Confusion Matrix')
print('\n')
print(confusion_matrix(y_test,predictions))
```

```
Confusion Matrix


[[19  0  0]
 [ 0 15  0]
 [ 0  2 14]]
```

In [ ]: