In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from urllib.request import urlretrieve
```

1.Data collection

In [2]:
```
url='https://raw.githubusercontent.com/amankharwal/Website-data/master/IRIS.csv'
```

In [3]:
```
urlretrieve(url,'file.csv')
```

Out[3]:
```
('file.csv', <http.client.HTTPMessage at 0x19287f99350>)
```

In [4]:
```
df=pd.read_csv(R"C:\Users\Shobhit Khaiwal\file.csv")
```

2. Data Preprocessing

In [5]:
```
df.info
```

Out[5]:
```
<bound method DataFrame.info of      sepal_length  sepal_width  petal_length  p
etal_width         species
0             5.1          3.5           1.4          0.2     Iris-setosa
1             4.9          3.0           1.4          0.2     Iris-setosa
2             4.7          3.2           1.3          0.2     Iris-setosa
3             4.6          3.1           1.5          0.2     Iris-setosa
4             5.0          3.6           1.4          0.2     Iris-setosa
..            ...          ...           ...          ...             ...
145           6.7          3.0           5.2          2.3  Iris-virginica
146           6.3          2.5           5.0          1.9  Iris-virginica
147           6.5          3.0           5.2          2.0  Iris-virginica
148           6.2          3.4           5.4          2.3  Iris-virginica
149           5.9          3.0           5.1          1.8  Iris-virginica

[150 rows x 5 columns]>
```

In [6]:
```
df.shape
```

Out[6]:
```
(150, 5)
```

In [7]:
```
print(df.columns.values)
```

```
['sepal_length' 'sepal_width' 'petal_length' 'petal_width' 'species']
```

In [8]:
```
df['species'].unique()
```

Out[8]:
```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

In [9]:
```
df.describe(include=['O'])
```

Out[9]:

	species
count	150
unique	3
top	Iris-setosa
freq	50

In [10]:
```
df.isnull().sum()
```

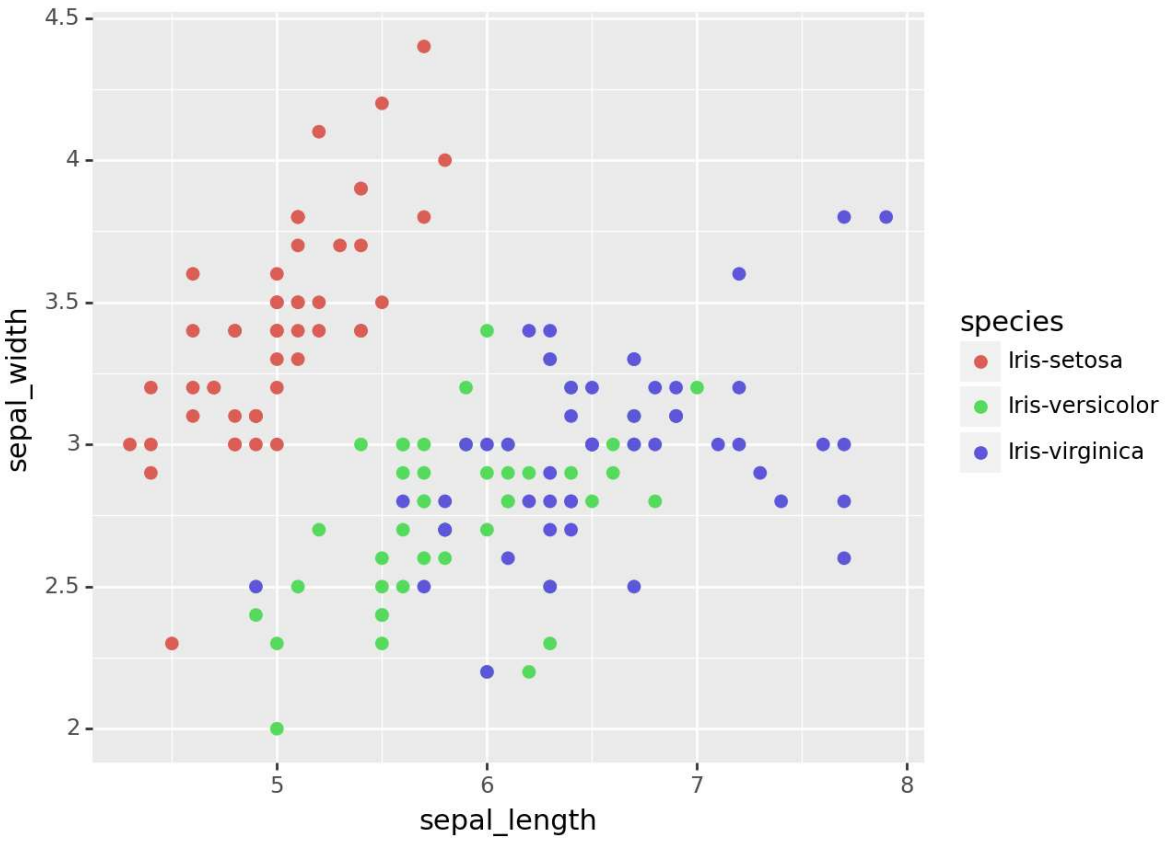
Out[10]:
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

3. Data Visualization

In [11]:
```
x=df['sepal_length']
y=df['sepal_width']
```

In [12]:
```
from plotnine import *
```

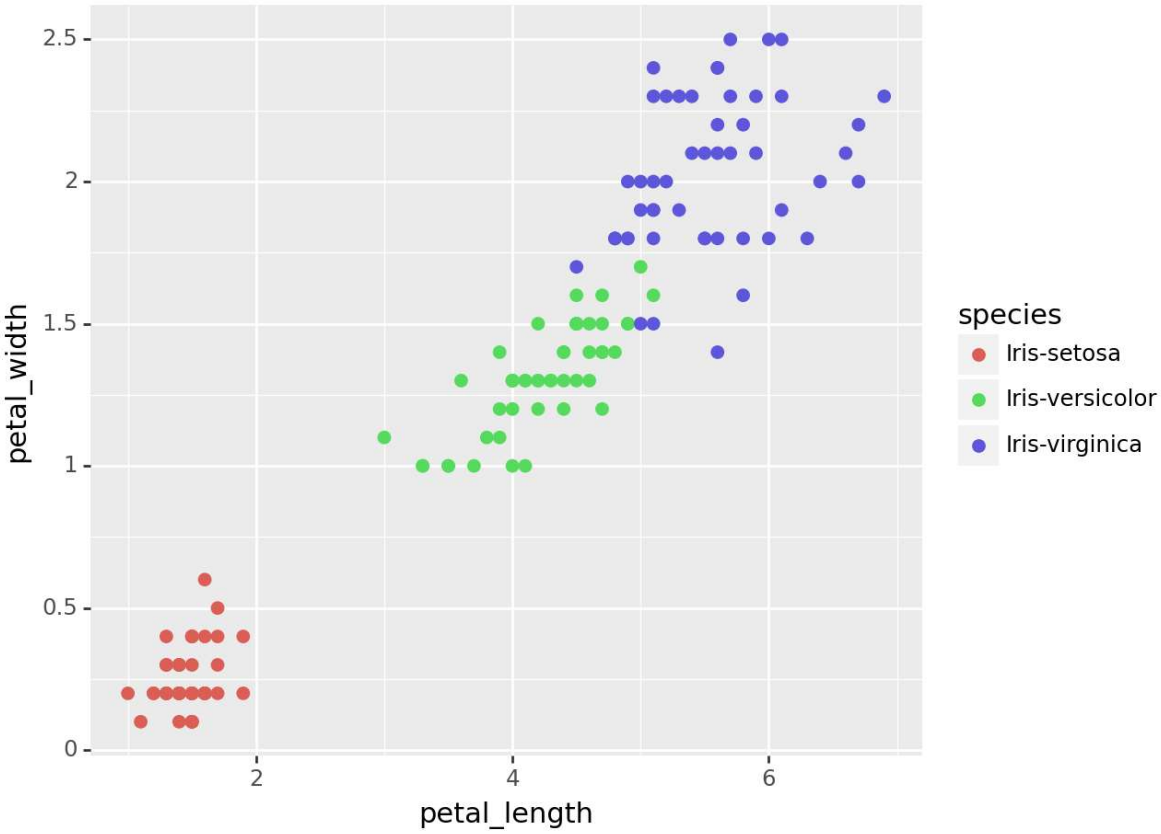
In [13]:
```
(ggplot(df)+aes(x,y,color='species')+geom_point(size=2))
```



Out[13]:
```
<Figure Size: (640 x 480)>
```

In [14]:
```
x=df['petal_length']
y=df['petal_width']
```

In [15]:
```
(ggplot(df)+aes(x,y,color='species')+geom_point(size=2))
```



Out[15]: <Figure Size: (640 x 480)>

4. Data Prepration for model

In [16]:
```
species_map={'Iris-setosa':0, 'Iris-versicolor':1, 'Iris-virginica':2}
```

In [17]:
```
df['species']=df['species'].replace(species_map)
```

In [18]:
```
df
```

Out[18]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	2
146	6.3	2.5	5.0	1.9	2
147	6.5	3.0	5.2	2.0	2
148	6.2	3.4	5.4	2.3	2
149	5.9	3.0	5.1	1.8	2

150 rows × 5 columns

In [19]:
```
x=df.drop('species',axis=1)
```

In [20]:
```
y=df['species']
```

In [21]:
```
print(x.shape,y.shape)
```

```
(150, 4) (150,)
```

5.Splitting data into training and test data set

In [22]:
```
from sklearn.model_selection import train_test_split as tts
```

In [23]:
```
x_train,x_test,y_train,y_test=tts(x,y,test_size=0.3,random_state=50)
```

6.Support Vector Machine with kernel

In [24]:
```
from sklearn import svm
```

In [25]:
```
clr=svm.SVC(kernel='poly')
```

In [26]:
```
clr.fit(x_train,y_train)
```

Out[26]:

▾ SVC

SVC(kernel='poly')

In [27]:
```
y_pred=clr.predict(x_test)
```

7.Accuracy for SVM

In [29]:
```
from sklearn.metrics import accuracy_score as acs
```

In [30]:
```
y_pred_accuracy=acs(y_test,y_pred)
```

In [31]:
```
print('accuracy:',y_pred_accuracy)
```

```
accuracy: 0.9555555555555556
```