In [1]:

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

In [3]:

```
#Reading dataset
iris = pd.read_csv(r"C:\Users\CTTC\Downloads\iris.csv",header=None)
iris
```

Out[3]:

	0	1	2	3	4
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 × 5 columns

In [4]:

```
#Renaming the columns
iris.columns = ['SL','SW','PL','PW','Flower']
iris.head()
```

Out[4]:

	SL	SW	PL	PW	Flower
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

In [5]:
```
iris.isnull().sum()
```

Out[5]:
```
SL        0
SW        0
PL        0
PW        0
Flower    0
dtype: int64
```

In [6]:
```
# we have to take only input
ip = iris.drop('Flower',axis=1)
ip.head()
```

Out[6]:

	SL	SW	PL	PW
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

In [7]:
```
#to form clusters/groups based on features we have used K-Means Clustering Algo
from sklearn.cluster import KMeans
km = KMeans(n_clusters=3)
km.fit(ip)
```

```
C:\Users\CTTC\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:870: Fut
ureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.
4. Set the value of `n_init` explicitly to suppress the warning
  warnings.warn(
C:\Users\CTTC\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:1382: Us
erWarning: KMeans is known to have a memory leak on Windows with MKL, when th
ere are less chunks than available threads. You can avoid it by setting the e
nvironment variable OMP_NUM_THREADS=1.
  warnings.warn(
```

Out[7]:
```
▾ KMeans
KMeans(n_clusters=3)
```

In [8]:
```
#predict output based on features of input
kpred = km.predict(ip)
kpred
```

Out[8]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 2, 2, 2, 2, 0, 2, 2, 2,
       2, 2, 2, 0, 0, 2, 2, 2, 2, 0, 2, 0, 2, 0, 2, 2, 0, 0, 2, 2, 2, 2,
       2, 0, 2, 2, 2, 2, 0, 2, 2, 2, 0, 2, 2, 2, 0, 2, 2, 0])
```

In [9]:
```
#all updated values of centroid for each column(here 4 columns r there)
centroid = km.cluster_centers_
print(centroid)
```

```
[[5.9016129  2.7483871  4.39354839 1.43387097]
 [5.006      3.418      1.464      0.244     ]
 [6.85       3.07368421 5.74210526 2.07105263]]
```

In [10]:
```
iris['Predict'] = kpred
```

In [11]:
```
iris
```

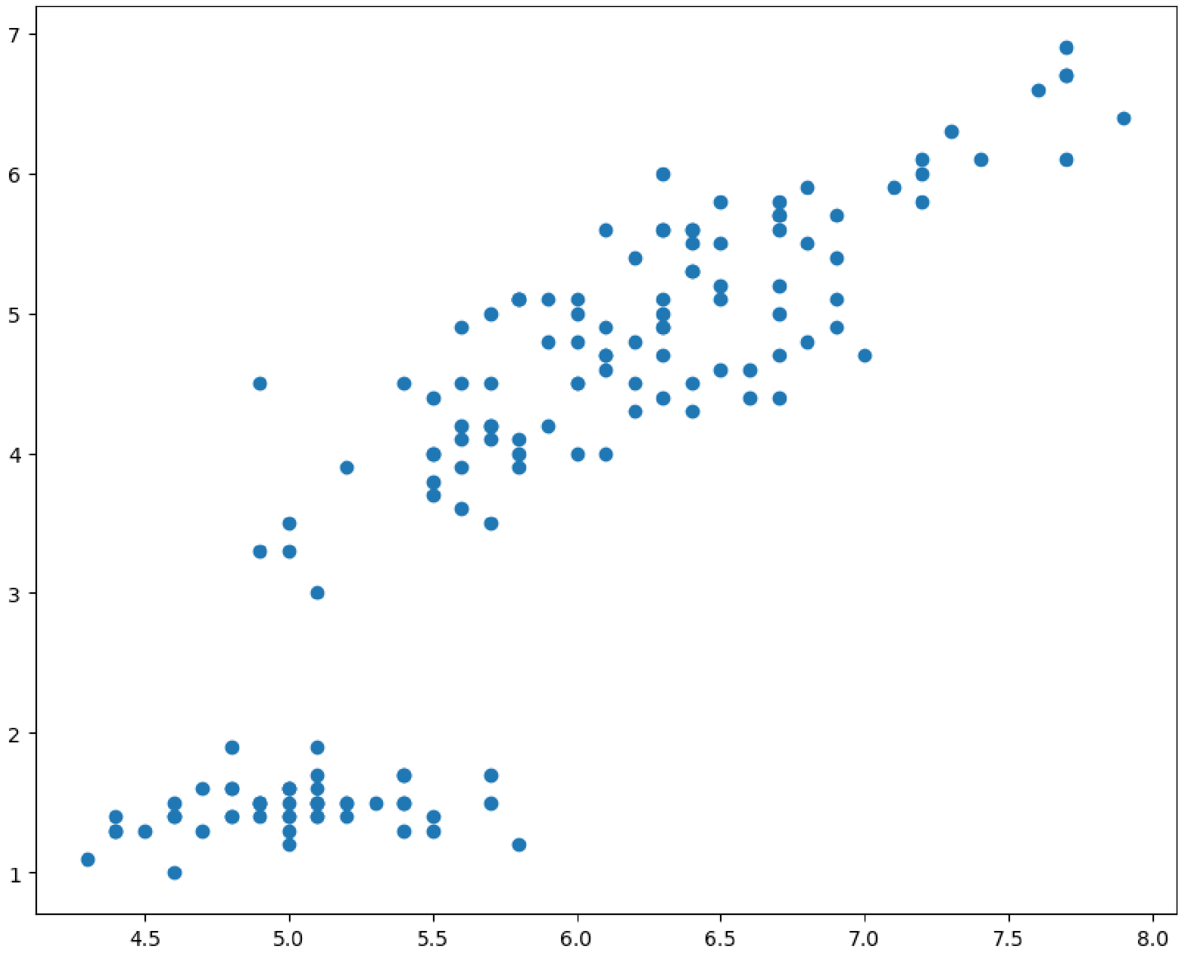
Out[11]:

	SL	SW	PL	PW	Flower	Predict
0	5.1	3.5	1.4	0.2	Iris-setosa	1
1	4.9	3.0	1.4	0.2	Iris-setosa	1
2	4.7	3.2	1.3	0.2	Iris-setosa	1
3	4.6	3.1	1.5	0.2	Iris-setosa	1
4	5.0	3.6	1.4	0.2	Iris-setosa	1
...	...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica	2
146	6.3	2.5	5.0	1.9	Iris-virginica	0
147	6.5	3.0	5.2	2.0	Iris-virginica	2
148	6.2	3.4	5.4	2.3	Iris-virginica	2
149	5.9	3.0	5.1	1.8	Iris-virginica	0

150 rows × 6 columns

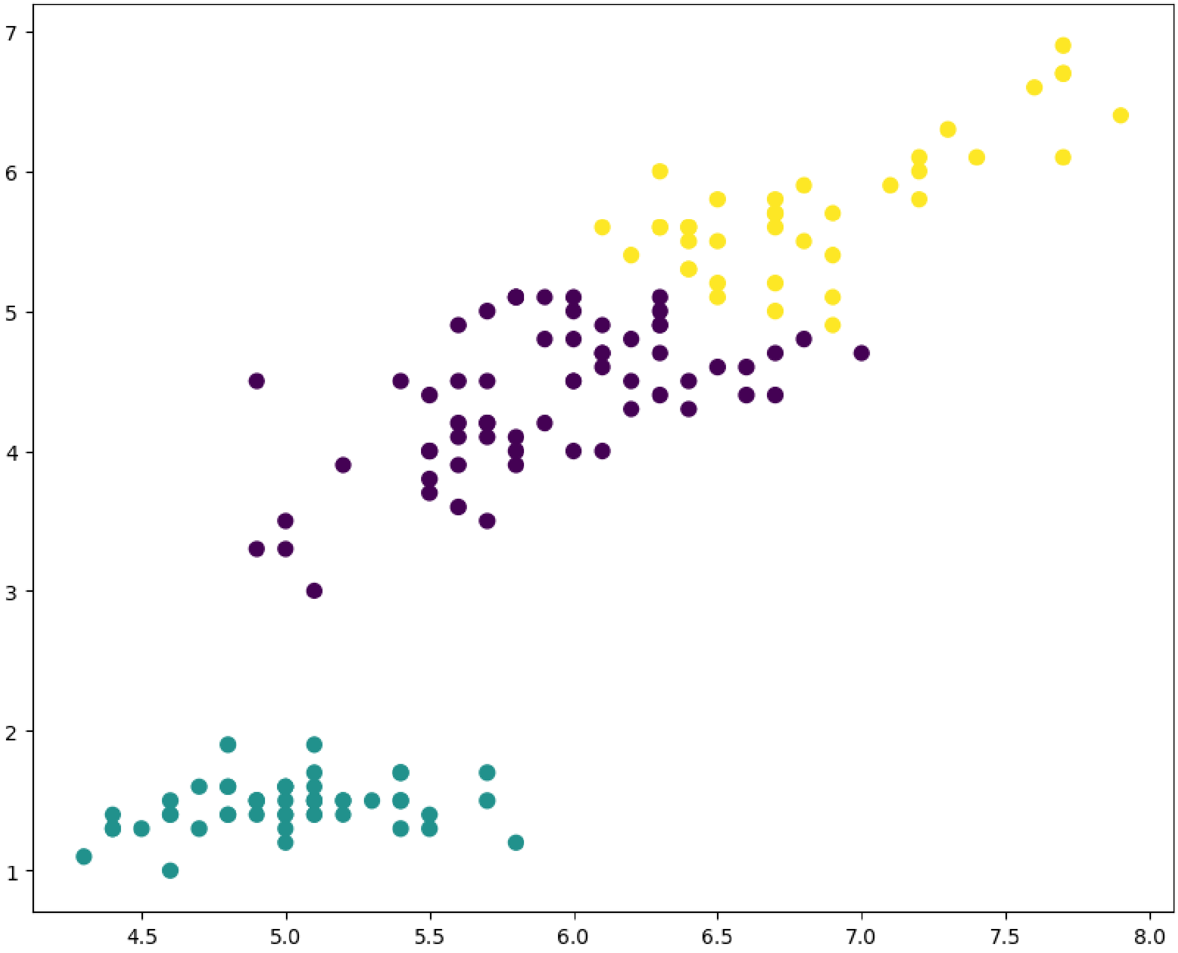
In [12]:

```
plt.figure(figsize=(10,8))
plt.scatter(iris.SL,iris.PL)
plt.show()
```



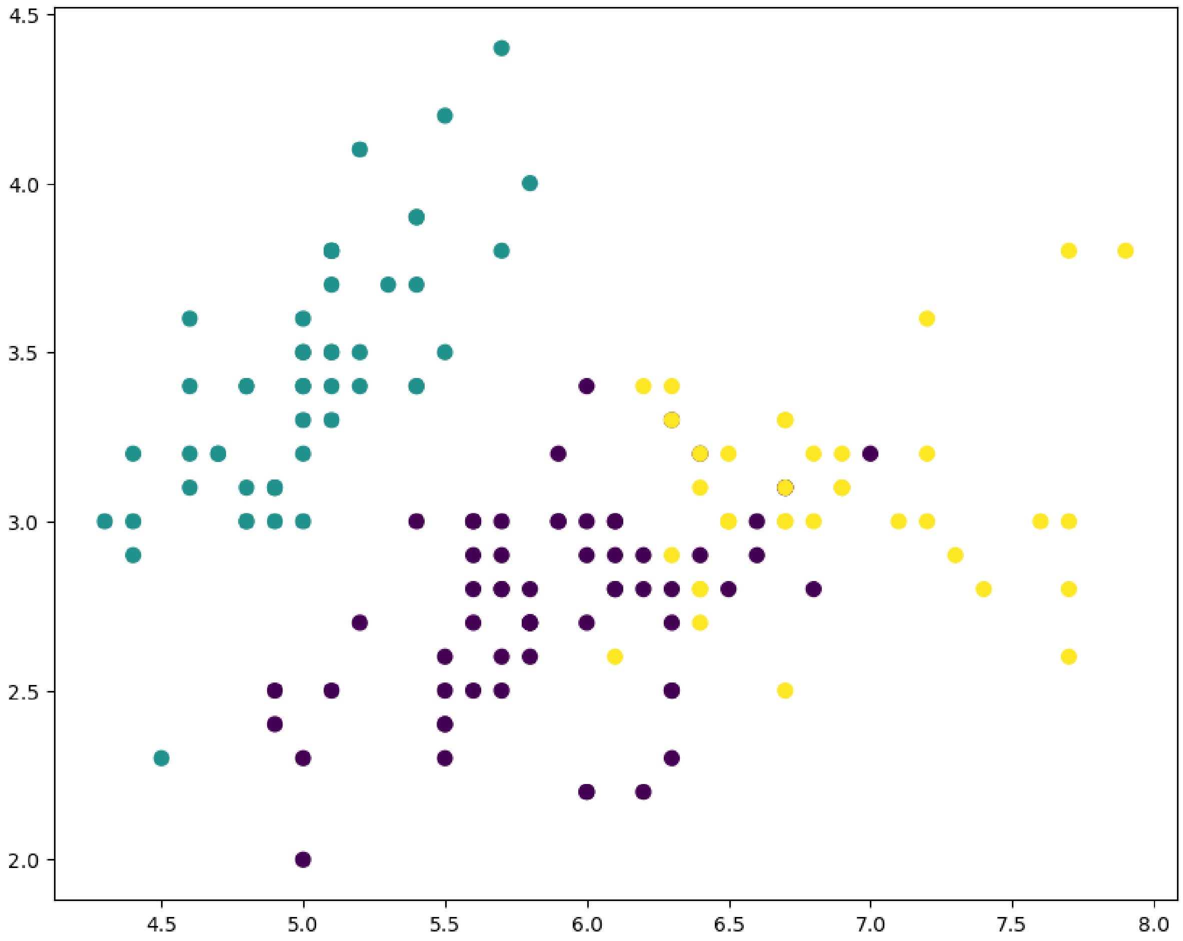
In [15]:

```
plt.figure(figsize=(10,8))
plt.scatter(iris.SL,iris.PL,c=kpred,s=50,cmap='viridis')
plt.show()
```



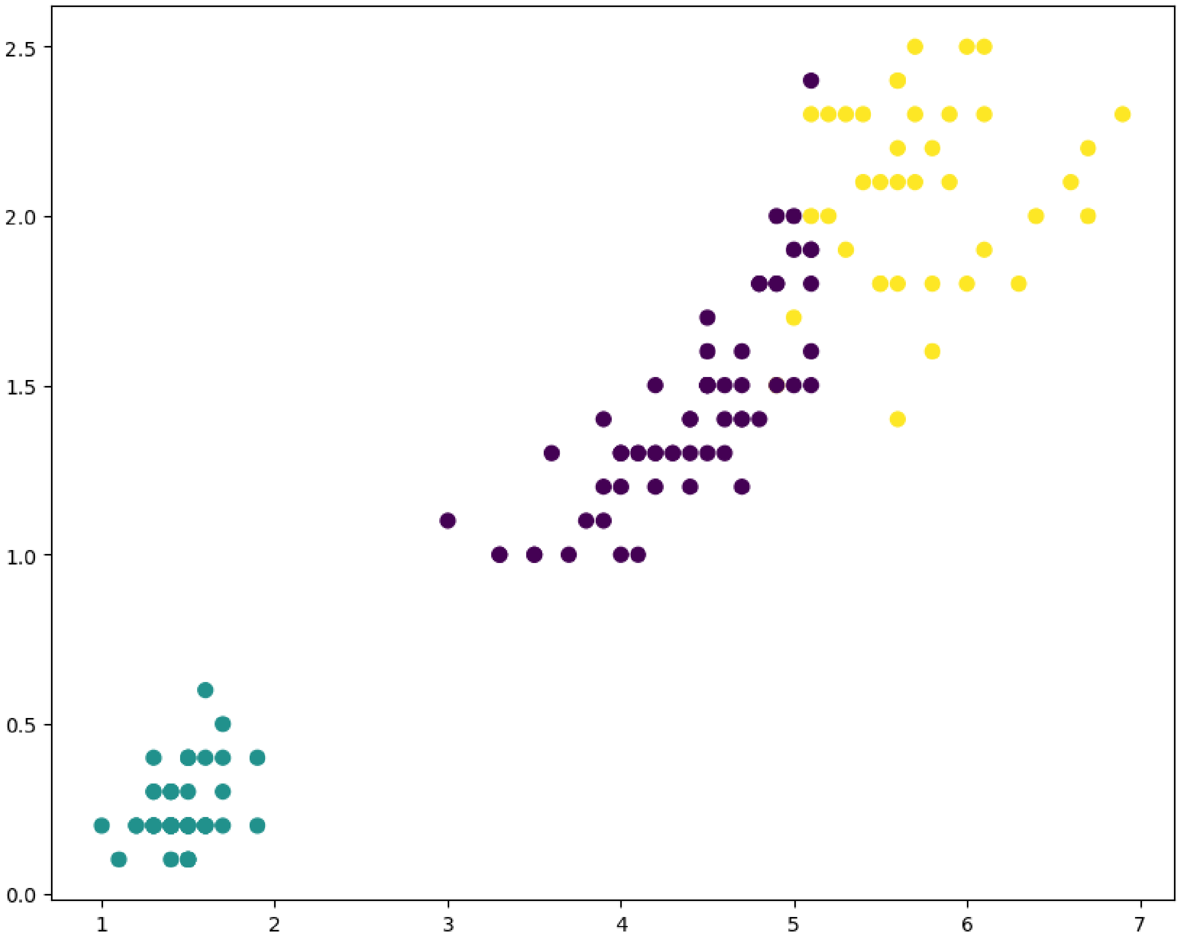
In [16]:

```
plt.figure(figsize=(10,8))
plt.scatter(iris.SL,iris.SW,c=kpred,s=50,cmap='viridis')
plt.show()
```

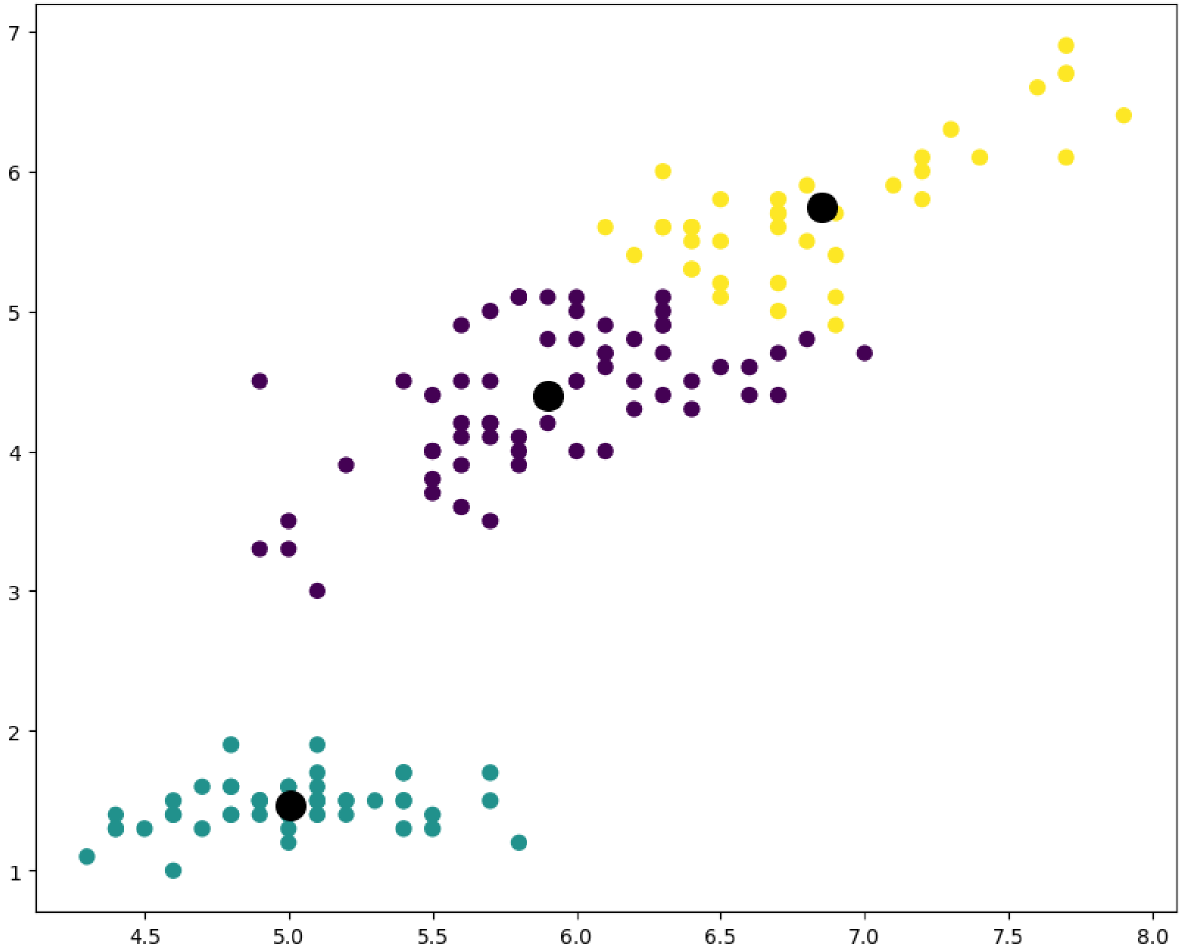


In [17]:

```
plt.figure(figsize=(10,8))
plt.scatter(iris.PL,iris.PW,c=kpred,s=50,cmap='viridis')
plt.show()
```



```
In [21]: plt.figure(figsize=(10,8))
         plt.scatter(iris.SL,iris.PL,c=kpred,s=50,cmap='viridis')
         plt.scatter(centroid[:,0],centroid[:,1],c='black',s=200)
         plt.show()
```



```
In [19]: iris.head(2)
```

Out[19]:

	SL	SW	PL	PW	Flower	Predict
0	5.1	3.5	1.4	0.2	Iris-setosa	1
1	4.9	3.0	1.4	0.2	Iris-setosa	1

```
In [20]: centroid
```

```
Out[20]: array([[5.9016129 , 2.7483871 , 4.39354839, 1.43387097],
                [5.006     , 3.418     , 1.464     , 0.244     ],
                [6.85      , 3.07368421, 5.74210526, 2.07105263]])
```

```
In [25]: centroid[:,0]
```

```
Out[25]: array([5.9016129, 5.006    , 6.85     ])
```

In [27]:

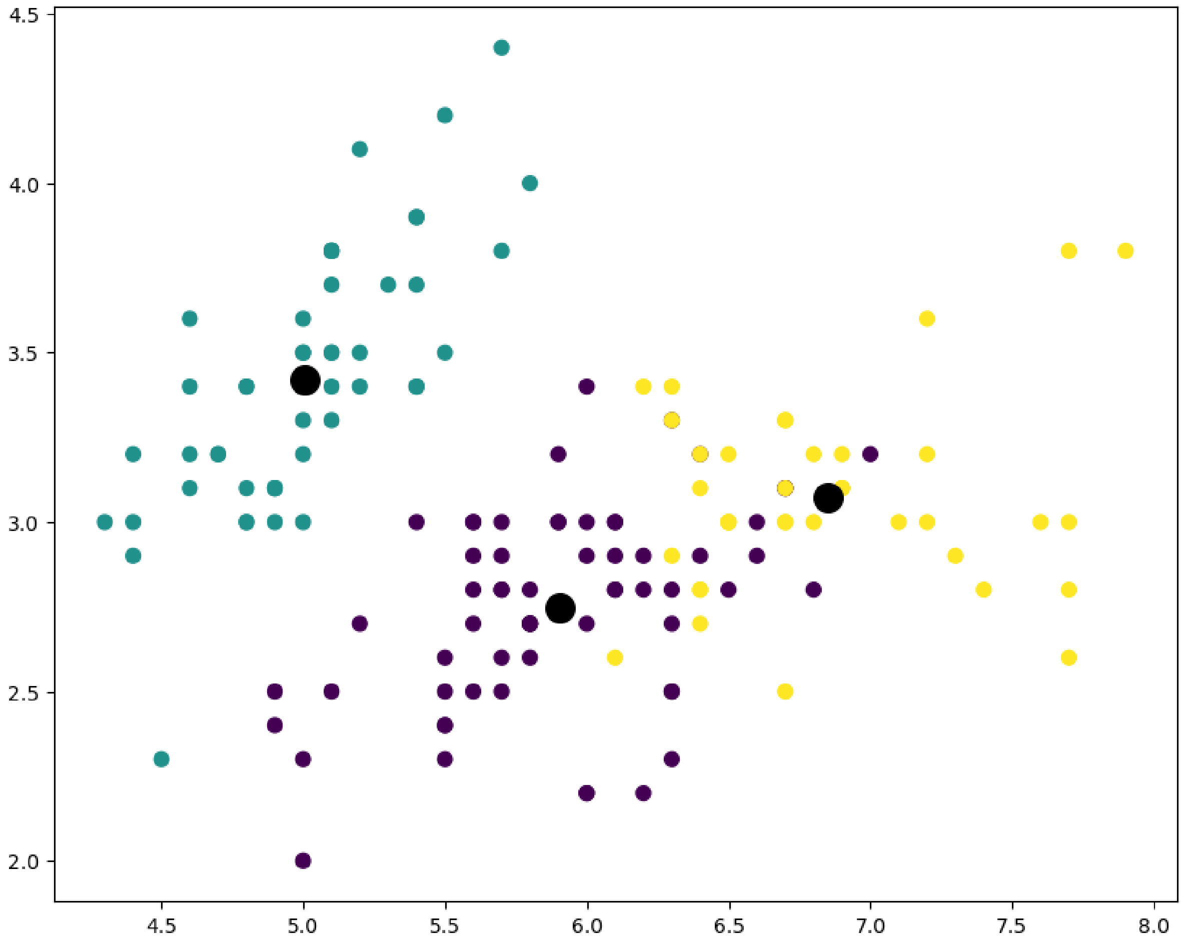
```
centroid[:,2]
```

Out[27]:

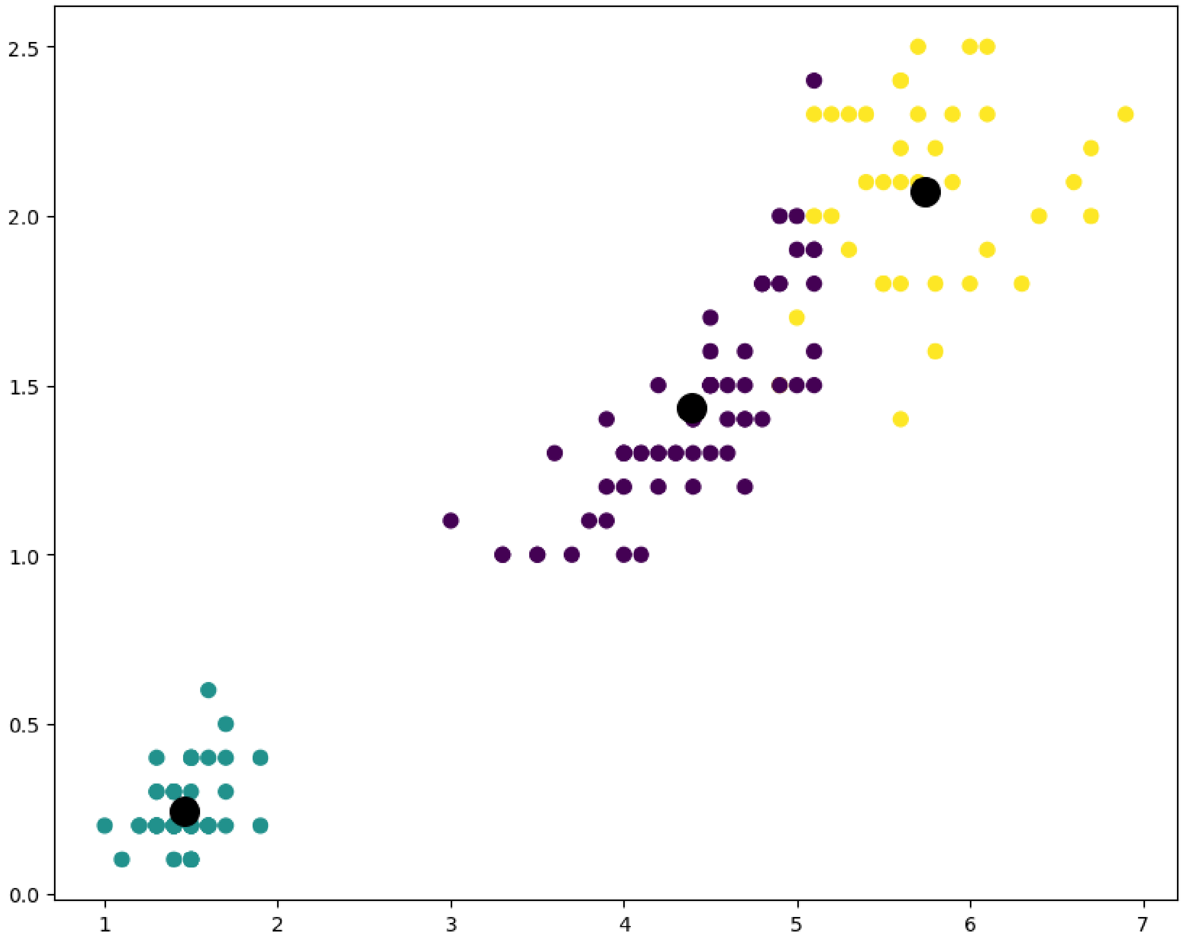
```
array([4.39354839, 1.464     , 5.74210526])
```

In [28]:

```
plt.figure(figsize=(10,8))
plt.scatter(iris.SL,iris.SW,c=kpred,s=50,cmap='viridis')
plt.scatter(centroid[:,0],centroid[:,1],c='black',s=200)
plt.show()
```



```
In [29]: plt.figure(figsize=(10,8))
         plt.scatter(iris.PL,iris.PW,c=kpred,s=50,cmap='viridis')
         plt.scatter(centroid[:,2],centroid[:,3],c='black',s=200)
         plt.show()
```



```
In [ ]:
```