In [42]:

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

In [43]:

```
from sklearn import datasets
x,y = datasets.load_iris(return_X_y = True , as_frame = True)
```

In [44]:

```
x
```

Out[44]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
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
...	...	...	...	...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

150 rows × 4 columns

In [45]:

```
x.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 4 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   sepal length (cm)  150 non-null    float64
 1   sepal width (cm)   150 non-null    float64
 2   petal length (cm)  150 non-null    float64
 3   petal width (cm)   150 non-null    float64
dtypes: float64(4)
memory usage: 4.8 KB
```

In [46]:

```
x.describe()
```

Out[46]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
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 [47]:

```
y.info()
```

```
<class 'pandas.core.series.Series'>
RangeIndex: 150 entries, 0 to 149
Series name: target
Non-Null Count  Dtype
--------------  -----
150 non-null    int64
dtypes: int64(1)
memory usage: 1.3 KB
```

In [48]:

```
y.describe()
```

Out[48]:

```
count    150.000000
mean       1.000000
std        0.819232
min        0.000000
25%        0.000000
50%        1.000000
75%        2.000000
max        2.000000
Name: target, dtype: float64
```

In [49]:
```
X=x.iloc[:,1]
X
```

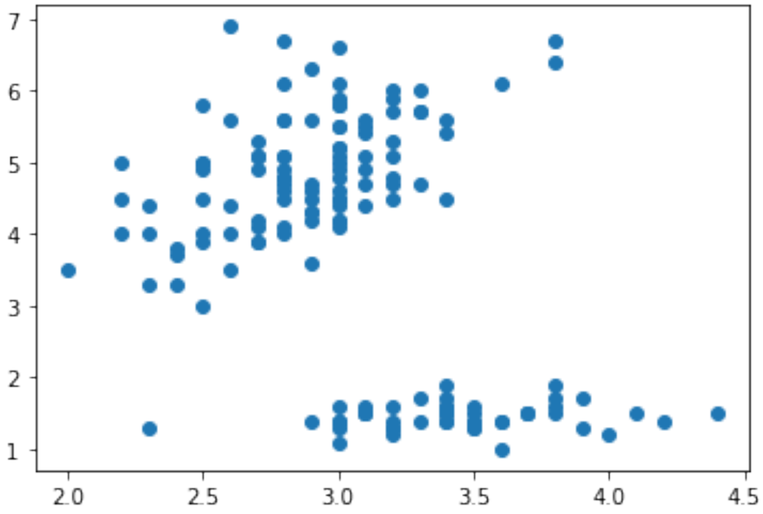
Out[49]:
```
0      3.5
1      3.0
2      3.2
3      3.1
4      3.6
      ...
145    3.0
146    2.5
147    3.0
148    3.4
149    3.0
Name: sepal width (cm), Length: 150, dtype: float64
```

In [50]:
```
Y=x.iloc[:,2]
Y
```

Out[50]:
```
0      1.4
1      1.4
2      1.3
3      1.5
4      1.4
      ...
145    5.2
146    5.0
147    5.2
148    5.4
149    5.1
Name: petal length (cm), Length: 150, dtype: float64
```

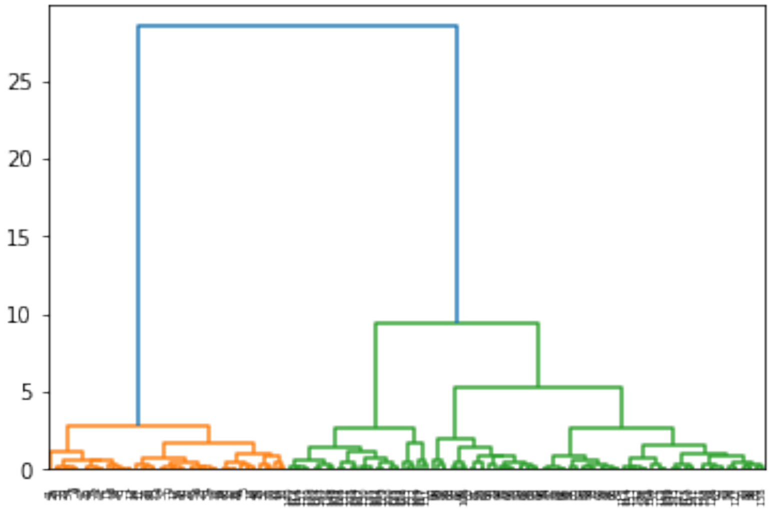
In [51]:
```
plt.scatter(X,Y)
```

Out[51]:
```
<matplotlib.collections.PathCollection at 0x129cca3d0>
```



In [52]:
```
from scipy.cluster.hierarchy import dendrogram, linkage
data=list(zip(X, Y))
```

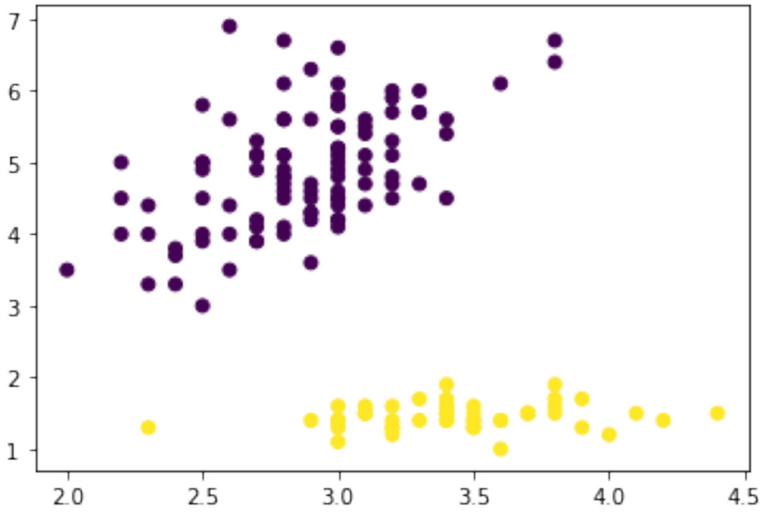
In [54]:
```
link=linkage(data,method='ward',metric='euclidean')
dendrogram(link)
plt.show()
```



In [55]:
```
from sklearn.cluster import AgglomerativeClustering
```

In [57]:
```
hcluster=AgglomerativeClustering(n_clusters=2,affinity='euclidean',
labels=hcluster.fit_predict(data)
```

In [58]:
```
plt.scatter(X,Y,c=labels)
plt.show()
```



In [ ]: