19MIS1018_ML_LAB-9_K-Means Clustering,Elbow Method and Silhouette Analysis

September 21, 2022

NAME: B DEVI PRASAD

REG.NO : 19MIS1018 SLOT: L13+L14 FACULTY: DR.G. Bharadwaja Kumar

1 TOPIC: K-MEANS

K-Means is one of the most popular clustering algorithms. By having central points to a cluster, it groups other points based on their distance to that central point. A downside of K-Means is having to choose the number of clusters, K, prior to running the algorithm that groups points. Elbow Method and Silhouette Analysis The most commonly used techniques for choosing the number of Ks are the Elbow Method and the Silhouette Analysis.

2 First Kmeans was implemented and then i did Elbow method and Silhouette

[3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.preprocessing import MinMaxScaler
```

[4]:
```
iris = pd.read_csv("iris.csv")
x = iris.iloc[:, [0, 1, 2, 3]].values
```

[5]:
```
iris.info()
iris[0:10]
```

```
<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
```

[5]:

	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
5	5.4	3.9	1.7	0.4	Setosa
6	4.6	3.4	1.4	0.3	Setosa
7	5.0	3.4	1.5	0.2	Setosa
8	4.4	2.9	1.4	0.2	Setosa
9	4.9	3.1	1.5	0.1	Setosa

[10]:
```
#Frequency distribution of species"
iris_outcome = pd.crosstab(index=iris["variety"], columns="count")
iris_outcome
```

[10]:
```
col_0       count
variety
Setosa         50
Versicolor     50
Virginica      50
```

[14]:
```
iris_setosa=iris.loc[iris["variety"]=="Setosa"]
iris_virginica=iris.loc[iris["variety"]=="Virginica"]
iris_versicolor=iris.loc[iris["variety"]=="Versicolor"]
```

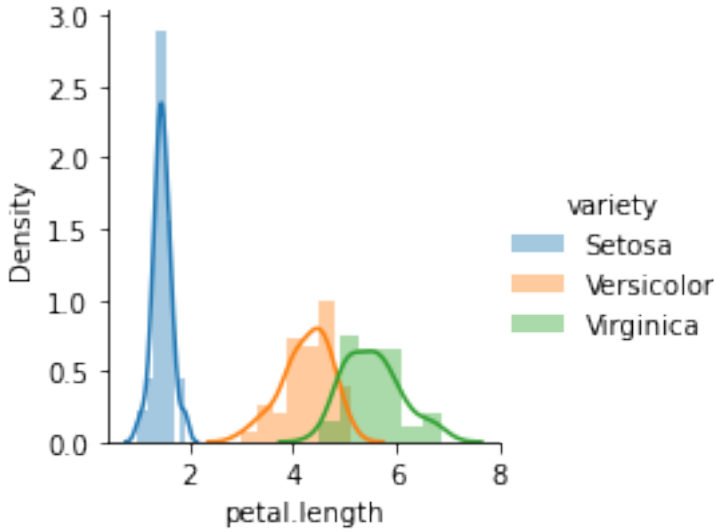
[15]:
```
sns.FacetGrid(iris,hue="variety").map(sns.distplot,"petal.length").add_legend()
sns.FacetGrid(iris,hue="variety").map(sns.distplot,"petal.width").add_legend()
sns.FacetGrid(iris,hue="variety").map(sns.distplot,"sepal.length").add_legend()
plt.show()
```

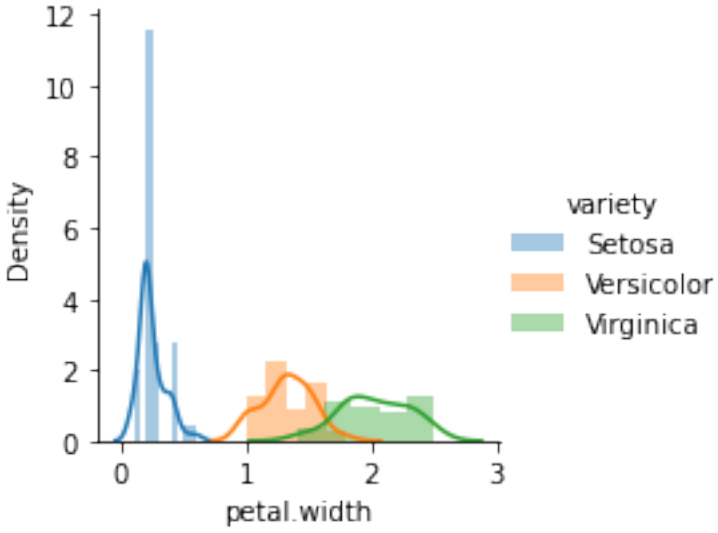
```
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

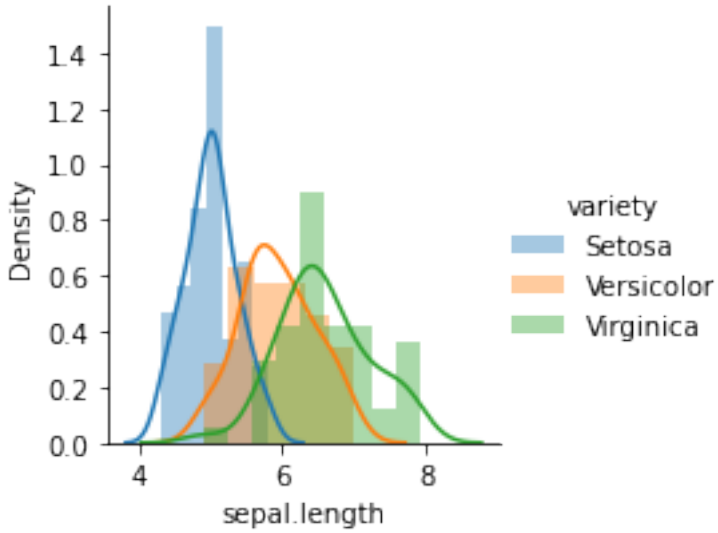
```
  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

```
  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  func(*plot_args, **plot_kwargs)
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:848: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

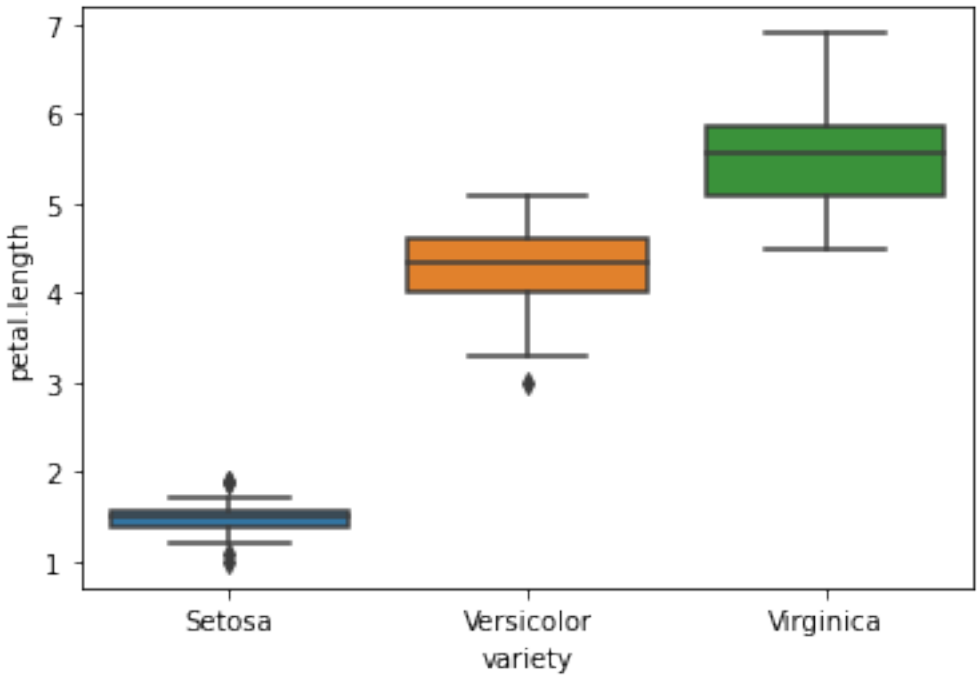
```
func(*plot_args, **plot_kwargs)
```



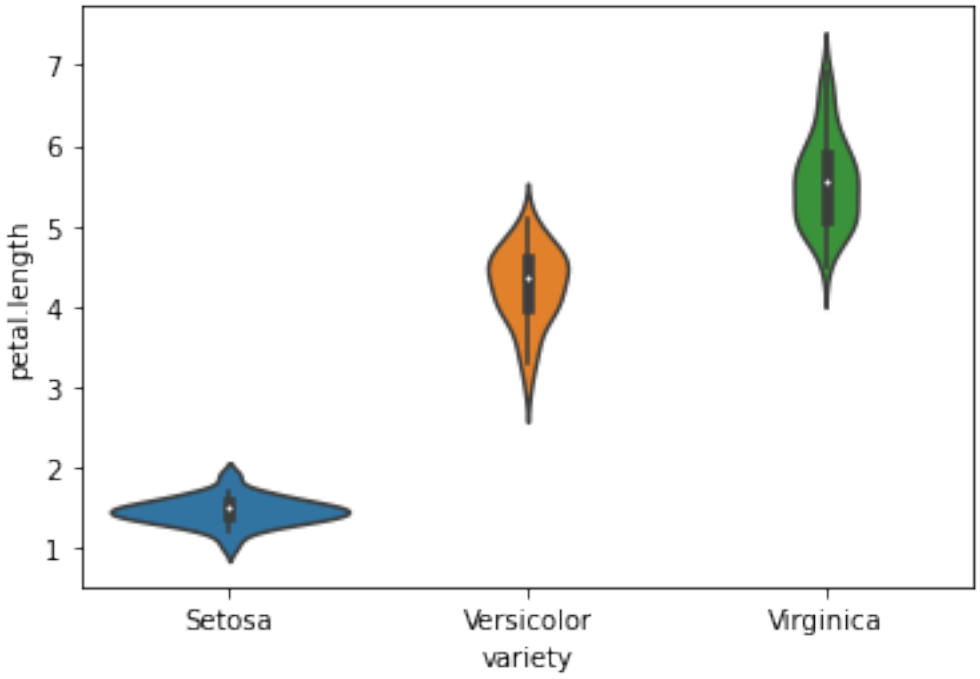




[16]:
```
sns.boxplot(x="variety",y="petal.length",data=iris)
plt.show()
```



[18]:
```
sns.violinplot(x="variety",y="petal.length",data=iris)
plt.show()
```



[19]:
```
sns.set_style("whitegrid")
sns.pairplot(iris,hue="variety",size=3);
plt.show()
```

```
C:\Users\admin\AppData\Local\Programs\Python\Python39\lib\site-
packages\seaborn\axisgrid.py:2095: UserWarning: The `size` parameter has been
renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```

3 Elbow Method and Silhouette Analysis

[1]:
```
! pip install yellowbrick
```

```
Collecting yellowbrick
  Downloading yellowbrick-1.5-py3-none-any.whl (282 kB)
     ------------------------------------ 282.6/282.6 kB 136.2 kB/s eta 0:00:00
Requirement already satisfied: numpy>=1.16.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
yellowbrick) (1.21.4)
Requirement already satisfied: cycler>=0.10.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
yellowbrick) (0.11.0)
Requirement already satisfied: scikit-learn>=1.0.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
```

```
yellowbrick) (1.0.2)
Requirement already satisfied: matplotlib!=3.0.0,>=2.0.2 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
yellowbrick) (3.4.3)
Requirement already satisfied: scipy>=1.0.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
yellowbrick) (1.8.0)
Requirement already satisfied: pyparsing>=2.2.1 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (3.0.6)
Requirement already satisfied: pillow>=6.2.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (8.4.0)
Requirement already satisfied: python-dateutil>=2.7 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (2.8.2)
Requirement already satisfied: kiwisolver>=1.0.1 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
matplotlib!=3.0.0,>=2.0.2->yellowbrick) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
scikit-learn>=1.0.0->yellowbrick) (3.1.0)
Requirement already satisfied: joblib>=0.11 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
scikit-learn>=1.0.0->yellowbrick) (1.1.0)
Requirement already satisfied: six>=1.5 in
c:\users\admin\appdata\local\programs\python\python39\lib\site-packages (from
python-dateutil>=2.7->matplotlib!=3.0.0,>=2.0.2->yellowbrick) (1.16.0)
Installing collected packages: yellowbrick
Successfully installed yellowbrick-1.5
```

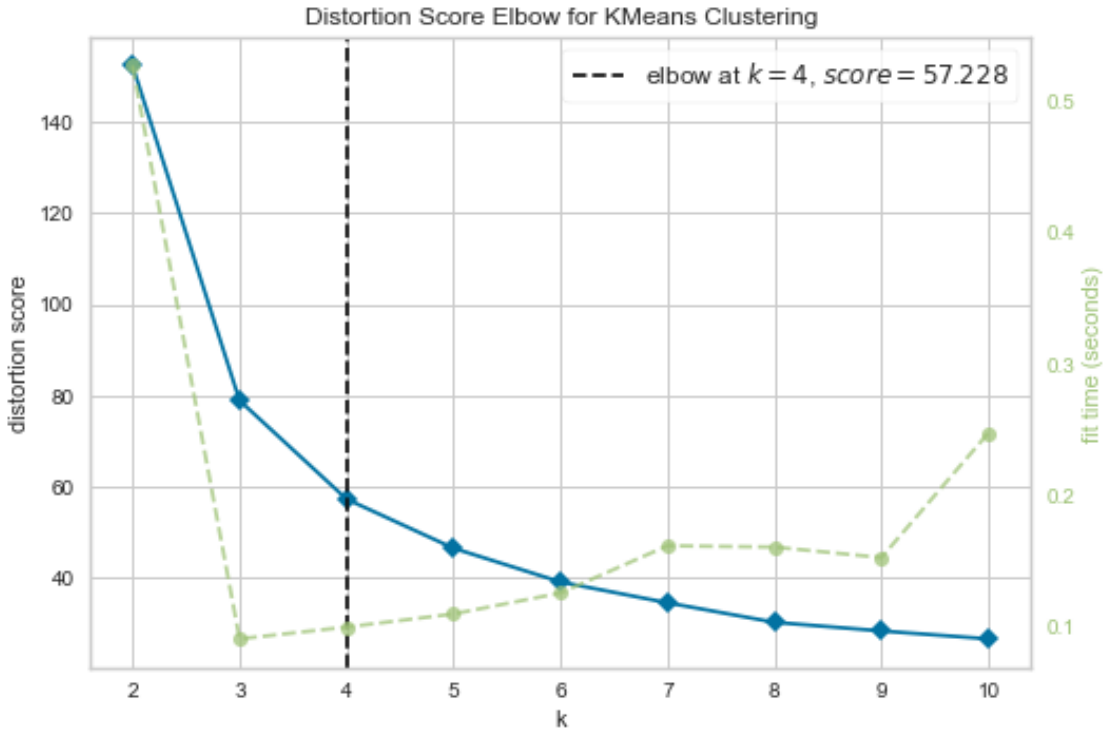
[2]:
```
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans
from yellowbrick.cluster import KElbowVisualizer, SilhouetteVisualizer

iris = load_iris()
```

[3]:
```
print(iris['feature_names'])
print(iris['target_names'])
X = iris['data']
```

```
['sepal length (cm)', 'sepal width (cm)', 'petal length (cm)', 'petal width
(cm)']
['setosa' 'versicolor' 'virginica']
```

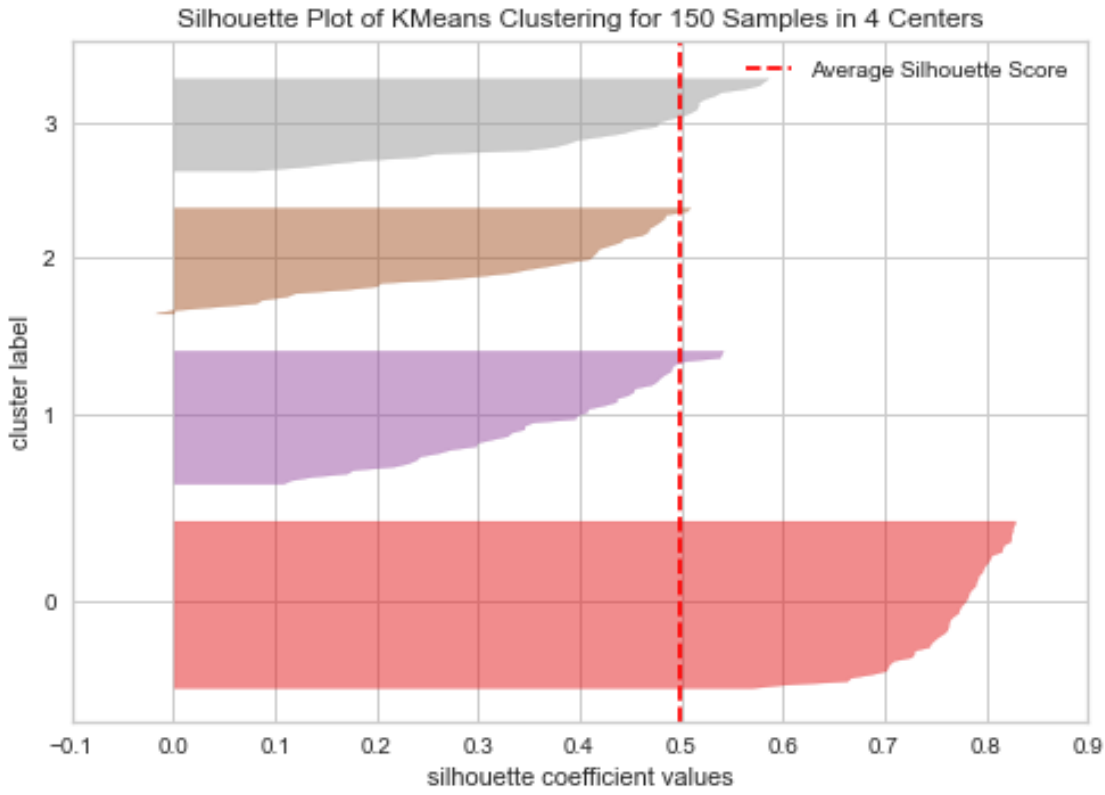
[4]:
```
model = KMeans(random_state=42)

elb_visualizer = KElbowVisualizer(model, k=(2,11))
```

```
elb_visualizer.fit(X)
elb_visualizer.show()
```



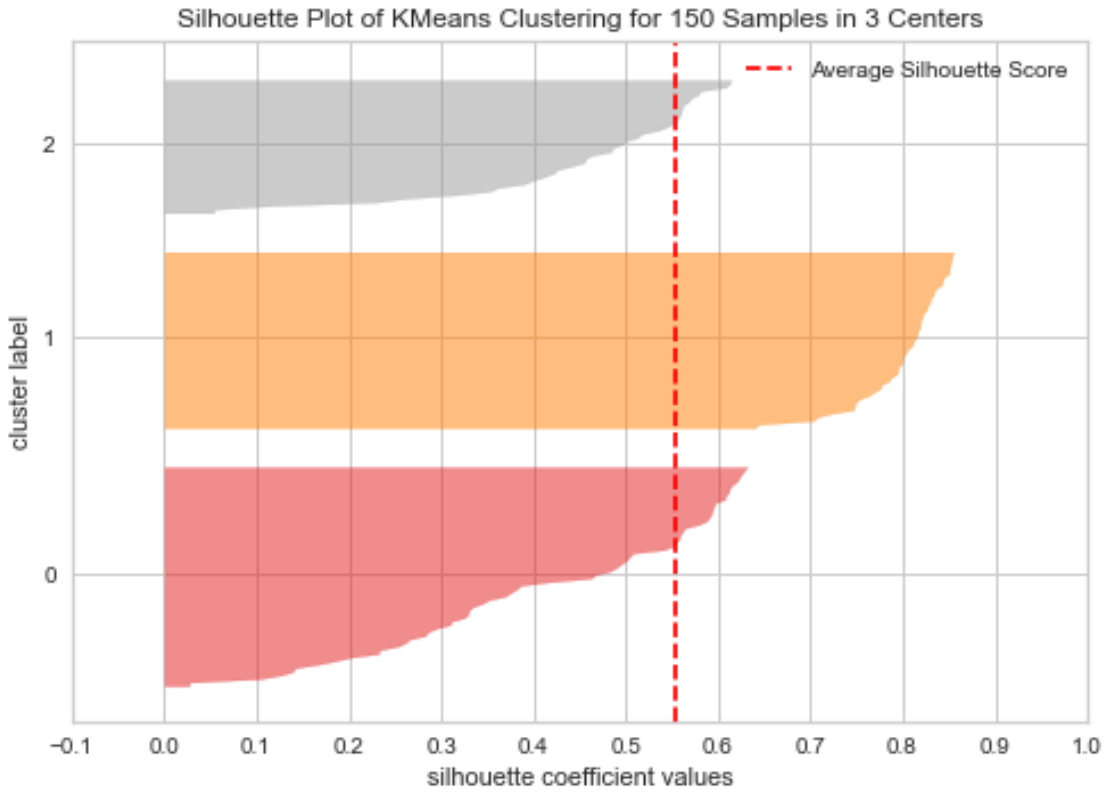
[4]:
```
<AxesSubplot:title={'center':'Distortion Score Elbow for KMeans Clustering'},
xlabel='k', ylabel='distortion score'>
```

[5]:
```
model_4clust = KMeans(n_clusters = 4, random_state=42)

sil_visualizer = SilhouetteVisualizer(model_4clust)
sil_visualizer.fit(X)
sil_visualizer.show()
```



[5]:
```
<AxesSubplot:title={'center':'Silhouette Plot of KMeans Clustering for 150
Samples in 4 Centers'}, xlabel='silhouette coefficient values', ylabel='cluster
label'>
```

[6]:
```
model_3clust = KMeans(n_clusters = 3, random_state=42)

sil_visualizer = SilhouetteVisualizer(model_3clust)
sil_visualizer.fit(X)
sil_visualizer.show()
```



[6]: `<AxesSubplot:title={'center':'Silhouette Plot of KMeans Clustering for 150 Samples in 3 Centers'}, xlabel='silhouette coefficient values', ylabel='cluster label'>`

4 OBSERVATION

By changing the number of clusters, the silhouette score got 0.05 higher and the clusters are more balanced. If we didnt know the actual number of clusters, by experimenting and combining both techniques, we would have chosen 3 instead of 2 as the number of Ks.

This is an example how combining and comparing different metrics, vizualizing data, and experimenting with different values of clusters is important to lead the result in the right direction. And also, how having a library that facilitates that analysis can help in that process!