

LAB EXERCISE

K-Means Algorithm

Aim of the Exercise

The main aim of this exercise is to explore k-Means clustering algorithm. The objectives of this exercise are

1. Explore kmeans algorithm for any toy dataset of your choice or

X_1	X_2
2	4
4	6
6	8
10	4
12	4

2. Explore kmeans algorithm for **Iris Dataset**

Kmeans model can be created by scikit-learn as

```
kmeans = KMeans(n_clusters = 2)
```

The algorithm can be applied to the given data as

```
kmeans.fit(df)
```

The predictions of the constructed model can be done as

```
df['pred'] = kmeans.predict(df)
```

The scatter diagram can be constructed using seaborn lmlplot as

```
sns.lmlplot('X','Y',scatter=True,fit_reg=False,data=df,hue='pred')
```

Listing - 1

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn import metrics
import matplotlib.pyplot as plt
import seaborn as sns

data = [[2,4],[4,6],[6,8],[10,4],[12,4]]
df = pd.DataFrame(data, columns = ['X','Y'])
plt.scatter(df['X'],df['Y'],color='green')
plt.title('Scatter Diagram')
```

```

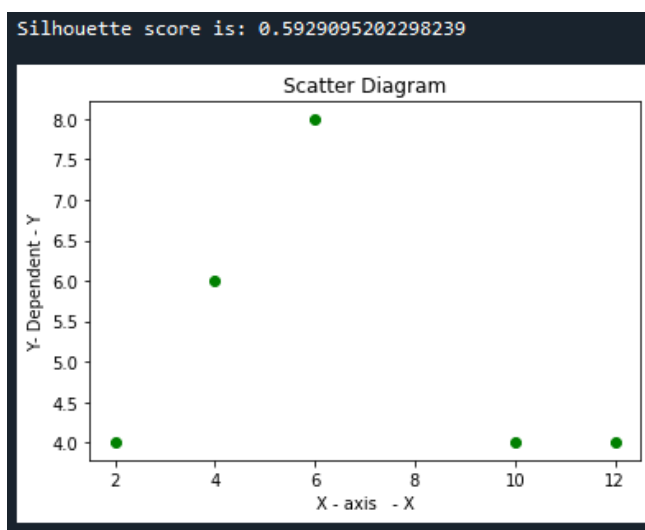
plt.xlabel('X - axis - X')
plt.ylabel('Y- Dependent - Y')
kmeans = KMeans(n_clusters = 2)
kmeans.fit(df)

df['pred'] = kmeans.predict(df)
sns.lmplot('X','Y',scatter=True,fit_reg=False,data=df,hue='pred')
labels = kmeans.labels_
score=metrics.silhouette_score(df, labels, metric='euclidean')
# Print Silhouette score
print('\nSilhouette score is:',score)

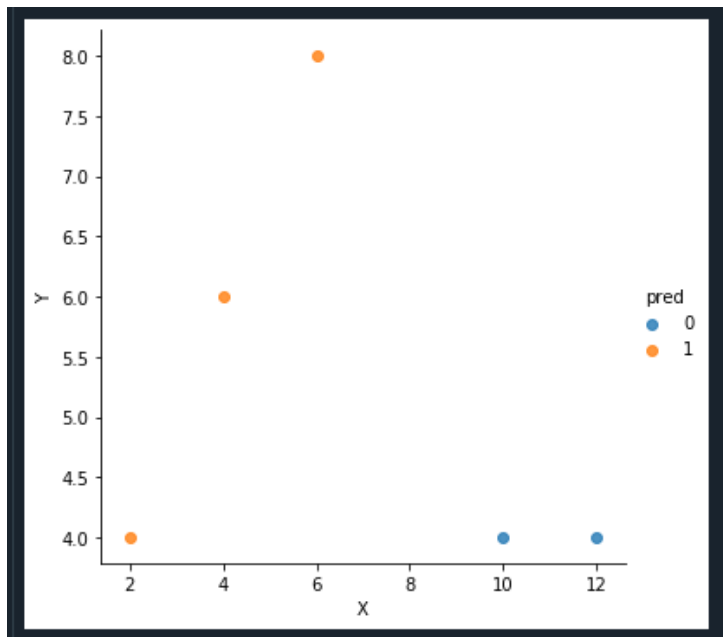
```

Output

The scatter diagram of the given data is given as follows:



The constructed clusters are shown as follows:



Listing 2

```
import pandas as pd
from sklearn.cluster import KMeans
import seaborn as sns

df = pd.read_csv("iris.csv")
df.columns = ['x1', 'x2', 'x3', 'x4', 'y']
df = df.drop(['x3', 'x4'], 1)
print(df.head(10))

kmeans = KMeans(n_clusters = 3)
X=df.values[:,0:2]
kmeans.fit(X)
df['pred'] = kmeans.predict(X)
df.head(10)

sns.lmplot('x1', 'x2', scatter=True, fit_reg=False, data=df, hue='pred')
df = pd.read_csv("iris.csv")
df.columns = ['x1', 'x2', 'x3', 'x4', 'y']
df = df.drop(['x3', 'x4'], 1)
print(df.head(10))
kmeans = KMeans(n_clusters = 3)
```

```
X=df.values[:,0:2]
kmeans.fit(X)
df['pred'] = kmeans.predict(X)
df.head(10)
sns.lmplot('x1','x2',scatter=True,fit_reg=False,data=df,hue='pred')
```

Output

