

Chapter 11 - Demo Iris - 4 Features

Cho dữ liệu iris.xls

1. Chuẩn hóa dữ liệu
2. Áp dụng Elbow tìm k
3. Áp dụng thuật toán K-Means để giải bài toán phân cụm theo K
4. So sánh giữa kết quả phân cụm với kết quả hiện có.
5. Cho $X_{\text{test}} = \text{np.array}([[4.7, 3.2, 1.5, 0.4], [4.8, 3.5, 4.5, 1.6], [6.1, 3.5, 5.7, 2]])$, cho biết những mẫu này thuộc cụm nào?
6. Vẽ hình, xem kết quả. Nhận xét kết quả.

```
In [1]: # from google.colab import drive
# drive.mount("/content/gdrive", force_remount=True)
```

```
In [2]: %cd '/content/gdrive/My Drive/LDS6_MachineLearning/practice/Chapter11_Kmeans/'
```

```
In [3]: from sklearn.cluster import KMeans
from sklearn import metrics
from scipy.spatial.distance import cdist
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

```
In [4]: iris = pd.read_excel("Iris.xls")
iris.shape
```

Out[4]: (150, 5)

```
In [5]: iris.head(3)
```

Out[5]:

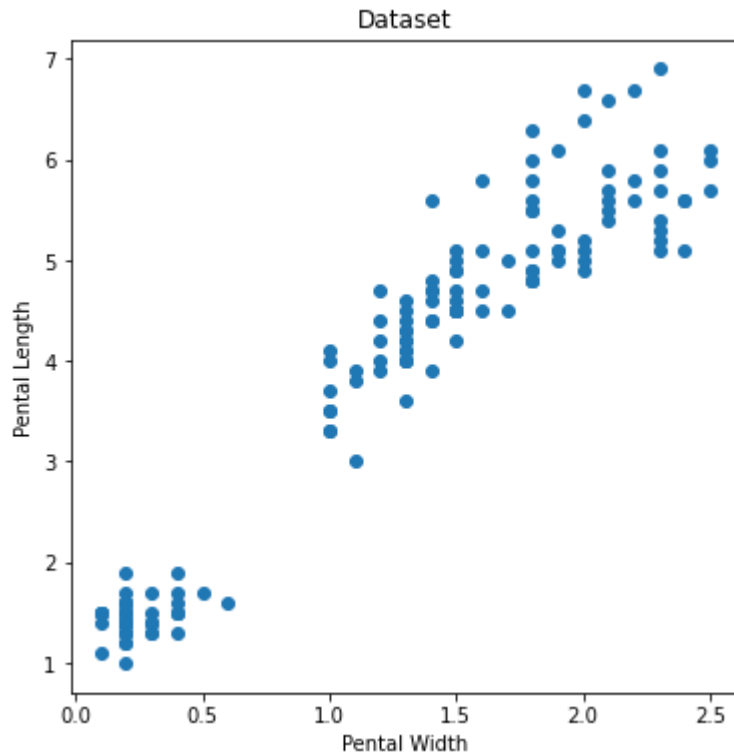
	sepalength	sepalwidth	petallength	petalwidth	iris
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa

```
In [6]: iris.groupby('iris').petallength.count()
```

Out[6]:

```
iris
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
Name: petallength, dtype: int64
```

```
In [7]: plt.figure(figsize=(6,6))
plt.scatter(iris.petalwidth, iris.petallength)
plt.title('Dataset')
plt.xlabel("Pental Width")
plt.ylabel("Pental Length")
plt.show()
```



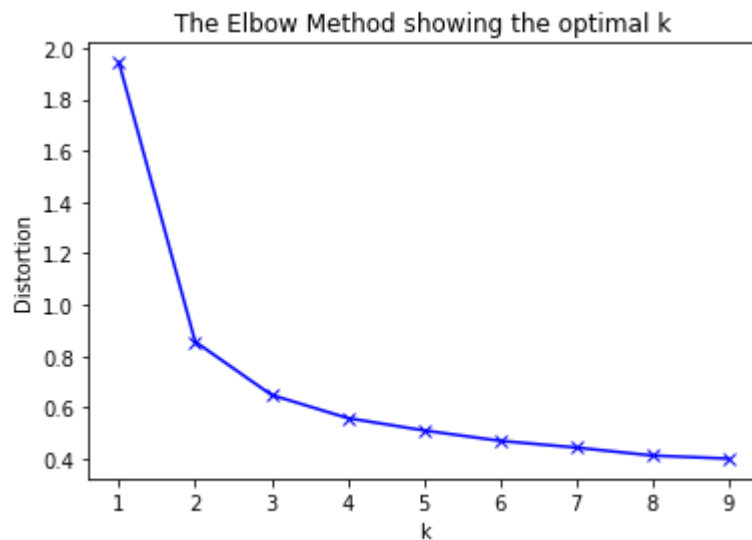
```
In [8]: X = iris.drop('iris', axis=1) # phân cum theo 4 thuộc tính
X.head(3)
```

Out[8]:

	sepalength	sepalwidth	petallength	petalwidth
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

```
In [9]: # k means determine k
distortions = []
K = range(1,10)
for k in K:
    kmeanModel = KMeans(n_clusters=k).fit(X)
    kmeanModel.fit(X)
    distortions.append(sum(np.min(cdist(X, kmeanModel.cluster_centers_,
                                      'euclidean'), axis=1)) / X.shape[0])

# Plot the elbow
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```



[illegible]

```
In [11]: X['iris'] = pd.Series(labels)
```

```
In [12]: X.head(3)
```

Out[12]:

	sepalength	sepalwidth	petallength	petalwidth	iris
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

```
In [13]: X_test = np.array([[4.7, 3.2, 1.5, 0.4],
                             [4.8, 3.5, 4.5, 1.6],
                             [6.1, 3.5, 5.7, 2]])
pred = kmeans.predict(X_test)
pred
```

```
Out[13]: array([0, 1, 2])
```

```
In [14]: plt.figure(figsize=(8,8))
plt.scatter(centroids[:, 3],centroids[:, 2], marker = "x", s=150, color='r')
plt.scatter(X.petalwidth, X.petallength, c=X.iris)
plt.scatter(X_test[:,3], X_test[:,2], marker="s", c='b')
plt.xlabel("Pental Width")
plt.ylabel("Pental Length")
plt.title("K-Means Cluster Iris", color="red")
plt.show()
```

