Lab 6.2 KMeans Clustering (3/6/2022)

ລະຫັດນັກສຶກສາ: 205Q0010.19

ຊື່ ແລະ ນາມສະກຸນ: ທ້າວ ນູຊ໋ວ ເຮີ

ຈົ່ງຕອບຄຳຖາມຕໍ່ໄປນີ້ໃຫ້ສຳເລັດດ້ວຍການນຳໃຊ້ຄຳສັ່ງຂອງ Python:

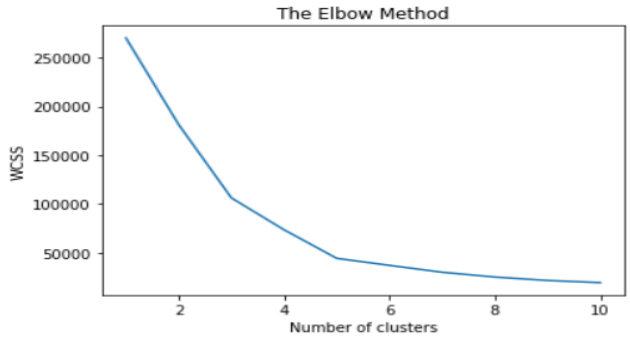
ພາກທີ 1

1.1. ຈາກຊຸດຂໍ້ມູນ Mall_Customers.csv. ຈົ່ງທຳການເລືອກຄຸນລັກສະນະ (Features): Age ແລະ Annual Income (k$).

```
dataset = pd.read_csv('Mall_Customers.csv')
X = dataset.iloc[:, [3, 4]].values
```

1.2 ຈົ່ງທຳການກຳນົດຄ່າຂອງ K ດ້ວຍການນຳໃຊ້ sklearn.cluster ແລະ Kmeans ເພື່ອຄິດໄລ່ຄ່າ wcss ພ້ອມແຕ້ມເສັ້ນສະແດງຂອງ Elbow Method

```
from sklearn.cluster import KMeans
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```



1.3. ຈົ່ງສ້າງແບບຈຳຮອງ Kmeans ເພື່ອແບ່ງກຸ່ມຊຸດຂໍ້ມູນດ້ວຍຕາມຄ່າຂອງ K ໃນຂໍ້1.2.

```
kmeans = KMeans(n_clusters = 5, init = 'k-means++', random_state = 42)
```

1.4 ຈົ່ງແບ່ງກຸ່ມຊຸດຂໍ້ມູນໃນຂໍ້ທີ 1.1 ດ້ວຍແບບຈຳຮອງ (Algorithm) ຂໍ້ 1.3

```
y_kmeans = kmeans.fit_predict(X)
```

1.5. ຈົ່ງທຳການສຳຫຼວດ (visualizing) ກຸ່ມຂໍ້ມູນທີ່ຖືກແບ່ງດ້ວຍ scatter plot ພ້ອມອະທິບາຍຜົນ.

```
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'red', label = 'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'blue', label = 'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'green', label = 'Cluster 3')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'cyan', label = 'Cluster 4')
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'magenta', label = 'Cluster 5')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c = 'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```

- ແມ່ນກຸ່ມທີ່ມີອາຍຸສູງແຕ່ມີລາຍໄດ້ຂະຫນາດປານກາງຫາສູງ
- ແມ່ນກຸ່ມທີ່ມີລາຍໄດ້ນ້ອຍໄປຫາຫຼາຍໃນໄວອາຍຸຍັງນ້ອຍ
- ແມ່ນກຸ່ມຄົນທີ່ມີອາຍຸລໍາດັບກາງແຕ່ມີລາຍໄດ້ຂະຫນາດນ້ອຍໄປຫາຫຼາຍ
- ແມ່ນກຸ່ມຄົນທີ່ມີລາຍໄດ້ຫຼາຍ ໃນໄວອາຍຸນ້ອຍ
- ແມ່ນກຸ່ມຄົນທີ່ມີລາຍໄດ້ນ້ອຍ ໃນໄວອາຍຸປານກາງ
- ແມ່ນກຸ່ມລະດັບສະເລ່ຍ

ພາກທີ 2

2.1 ຈາກຊຸດຂໍ້ມູນ Mall_Customers.csv. ຈົ່ງທຳການເລືອກຄຸນລັກສະນະ (Features): Age, Annual Income(k$) ແລະ Spending Score (1-100)

```
col = ['Age', 'Annual Income (k$)','Spending Score (1-100)']
```

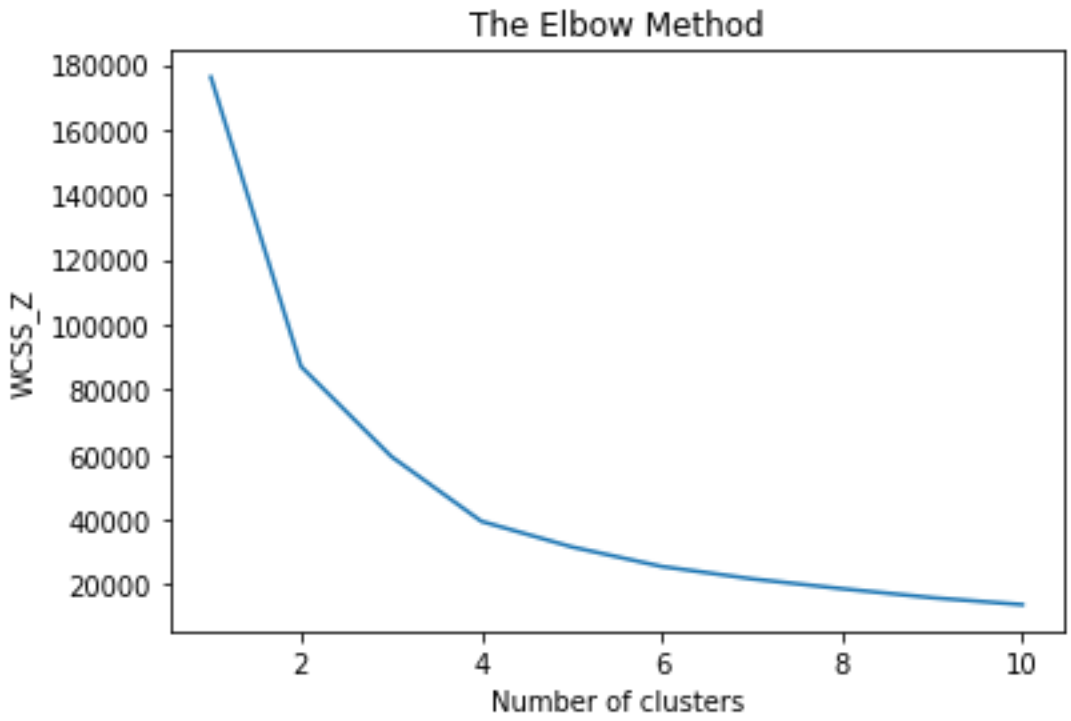
2.2. ຈົ່ງທຳການກະກຽມຂໍ້ມູນດ້ວຍ preprocessing.StandardScaler() ຂອງຄຸນລັກສະນະຂອງຂໍ້ມູນໃນຂໍ້2.1.

```
from sklearn import preprocessing
scaler = preprocessing.StandardScaler() # create StandardScaler instance
z=scaler.fit_transform(dataset[col]) # calc z-score
z[:5].round(4) # 2D numpy array

array([[-1.4246, -1.739 , -0.4348],
       [-1.281 , -1.739 ,  1.1957],
       [-1.3528, -1.7008, -1.7159],
       [-1.1375, -1.7008,  1.0404],
       [-0.5634, -1.6627, -0.396 ]])
```

2.3 ຈົ່ງທຳການກຳນົດຄ່າຂອງ K ດ້ວຍການນຳໃຊ້ sklearn.cluster ແລະ Kmeans ເພື່ອຄິດໄລ່ຄ່າ wcss ພ້ອມແຕ້ມເສັ້ນສະແດງຂອງ Elbow Method

```
from sklearn.cluster import KMeans
wcss_z = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters = i, init = 'k-means++', random_state = 42)
    kmeans.fit(X)
    wcss_z.append(kmeans.inertia_)
plt.plot(range(1, 11), wcss_z)
plt.title('The Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS_Z')
plt.show()
```



2.4. ຈົ່ງສ້າງແບບຈຳຮອງ Kmeans ເພື່ອແບ່ງກຸ່ມຊຸດຂໍ້ມູນດ້ວຍຕາມຄ່າຂອງ K ໃນຂໍ້2.3

```
kmeans_z = KMeans(n_clusters = 4, init = 'k-means++', random_state = 42)
```

2.5. ຈົ່ງແບ່ງກຸ່ມຊຸດຂໍ້ມູນໃນຂໍ້ທີ 2.2 ດ້ວຍແບບຈຳຮອງໃນຂໍ້ 2.4

```
z_kmeans = kmeans_z.fit_predict(X)
```

2.5. ຈົ່ງຊອກຫາເມັດກາງ cluster_centers_ ຂອງການແບ່ງກຸ່ມຊຸດຂໍ້ມູນ

```
kmeans_z.cluster_centers_.round(4)
```

```
array([[ 31.9589,  72.9589],
       [ 55.8148,  51.7778],
       [ 39.    , 106.5   ],
       [ 30.3469,  29.2653]])
```

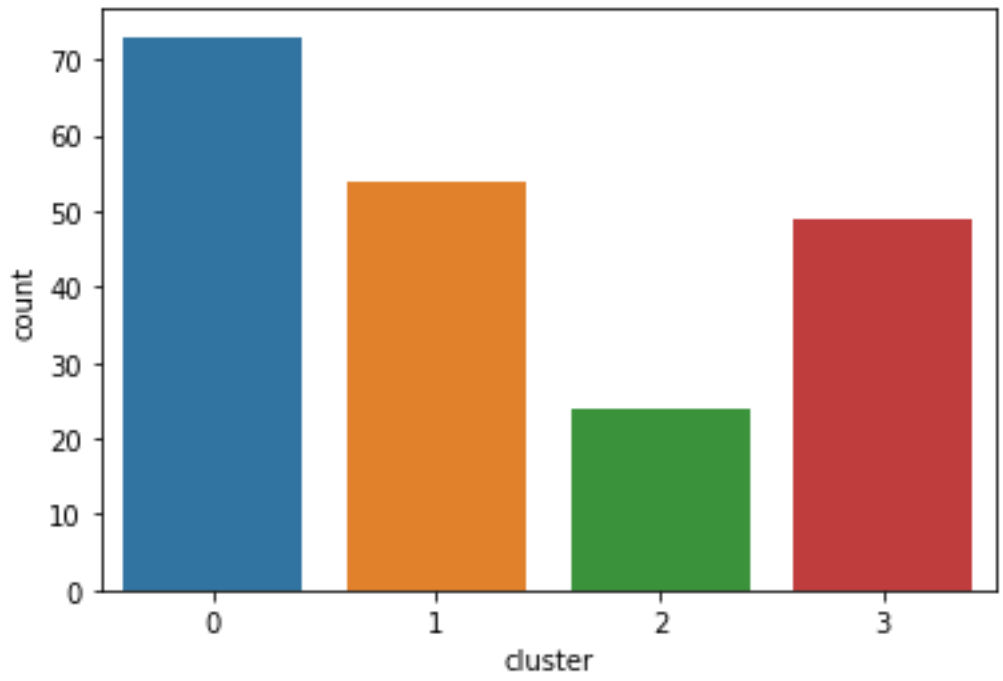
2.6. ຈົ່ງສະແດງນຳເບີກຸ່ມ (labels_) ໃສ່ໃນຊຸດຂໍ້ມູນໃນຮູບແບບ dataframe

```
dataset['cluster']=kmeans_z.labels_
dataset.head()
```

CustomerID	Gener	Age	Annual Income(K$)	Spending Score(1-100)
1	Male	19	15	39
2	Male	21	15	39
3	Fmale	20	16	6
5	Fmale	23	16	77

2.7. ຈົ່ງສະແດງກຸ່ມຂໍ້ມູນດ້ວຍ sns.countplot(), ພ້ອມອະທິບາຍຜົນ.

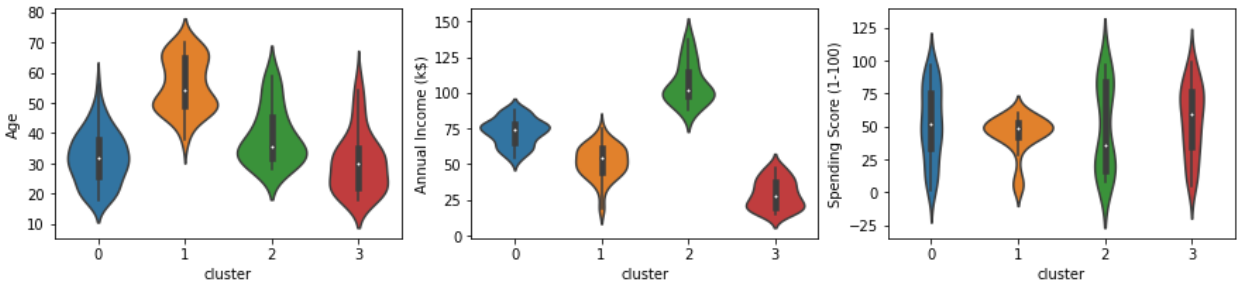
```
sns.countplot(x='cluster', data=dataset);
```



➔ ໃນຮູບຈະສະແດງໃຫ້ເປັນເຖິງການຂະຫຍາຍຕົວຂອງກຸ່ມວ່າເປັນຊະນິດທີ່ມີຈຳນວນລາຍຮັບຫຼາຍຫຼື ນ້ອຍແລະ ຈັດຢູ່ໃນລຳດັບໃດ

2.8. ຈົ່ງສະແດງກຸ່ມຂໍ້ມູນດ້ວຍ sns.violinplot(), ພ້ອມອະທິບາຍຜົນ.

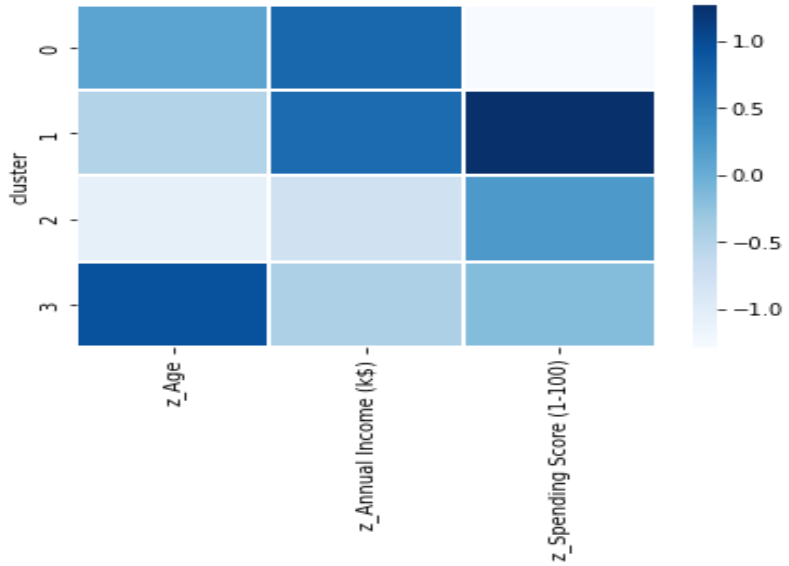
```
cols=['Age', 'Annual Income (k$)', 'Spending Score (1-100)']
fig, ax = plt.subplots(nrows=1, ncols=3, figsize=(15, 3))
ax=ax.ravel()
for i, col in enumerate(cols):
    sns.violinplot(x='cluster', y=col, data=dataset, ax=ax[i])
```



- ⇨ ໃນຮູບທີ1 : ແມ່ນຈະສະແດງເຖິງອາຍຸ
- ⇨ ໃນຮູບທີ2 : ແມ່ນຈະສະແດງເຖິງລາຍໄດ້ຂອງແຕ່ລະກຸ່ມ
- ⇨ ໃນຮູບທີ3 : ແມ່ນຈະສະແດງເຖິງຄະແນນການໃຊ້ຈ່າຍແຕ່ລະກຸ່ມ

2.9. ຈົ່ງສະແດງກຸ່ມຂໍ້ມູນດ້ວຍ sns.heatmap(), ພ້ອມອະທິບາຍຜົນ.

```
sns.heatmap(dx.groupby('cluster').median(), cmap="Blues", linewidths=1);
```



ໃນຮູບນີ້ແມ່ນຈະສະແດງແຕ່ລະແຖວໂດຍລະອຽດເຊິ່ງວ່າຖ້າແຖວໃດມີສີທີ່ແຈ້ງແມ່ນກຸ່ມທີ່ມີຈຳນວນຫຼາຍໃນກຸ່ມນັ້ນຕາມລຳດັບລົງມາ