Data Mining Programming HW2　112753204　陳品伃

1. 從 marketData 的資料中，取**女性**客戶的兩個欄位: **Age**、**Spending Score** 進行客戶分群，請使用 K-means 分群法，當 K＝2 時，請列出每一群的中心點，例如 C1 中心點: Age=33.3、Spending Score=87.1。2 個中心點的列表請根據中心點的 Age 數值升冪排序。

```
market_f = market.loc[market['Gender'] == "Female"]
```

Init = Random

程式碼

```
k_means_random = cluster.KMeans(n_clusters=2, init='random', max_iter=30, random_state=99)
k_means_random.fit(market_f_cluster)
center_random = k_means_random.cluster_centers_
center_random_df = pd.DataFrame(center_random,
                                columns=market_f_cluster.columns).sort_values(by=['Age'], ascending=True)
display(center_random_df)
```

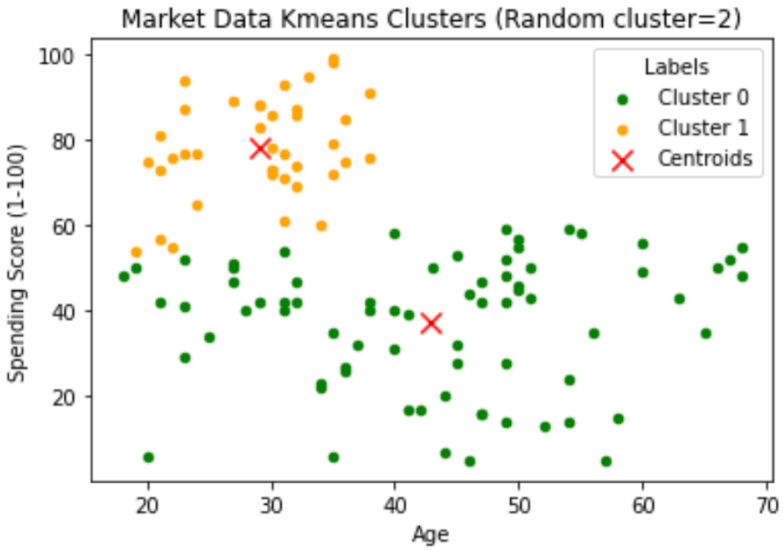
執行結果

	Age	Spending Score (1-100)
1	29.077	78.179
0	42.918	37.288

C0 中心點為：Age = 42.918 , Spending Score = 37.288

C1 中心點為：Age = 29.077 , Spending Score = 78.179

分群結果



Init = K-means++

程式碼

```
k_means_k = cluster.KMeans(n_clusters=2, init='k-means++', max_iter=30, random_state=99)
k_means_k.fit(market_f_cluster)
center_k = k_means_k.cluster_centers_
center_k_df = pd.DataFrame(center_k,
                           columns=market_f_cluster.columns).sort_values(by=['Age'], ascending=True)
display(center_k_df)
```

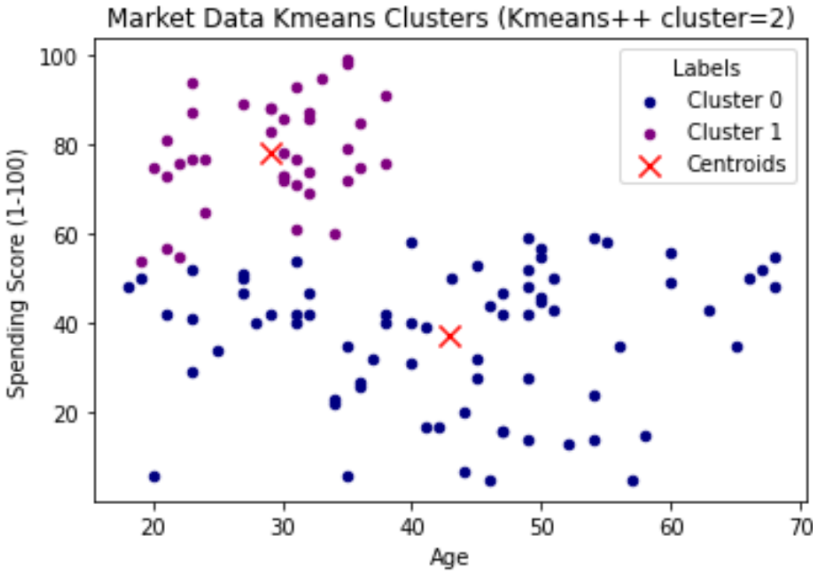
執行結果

	Age	Spending Score (1-100)
1	29.077	78.179
0	42.918	37.288

C0 中心點為：Age = 42.918 , Spending Score = 37.288

C1 中心點為：Age = 29.077 , Spending Score = 78.179

分群結果



Q1 Summary

在第一題使用了 random 和 Kmeans++兩種選擇群中心的方法，對於此資料集來說，兩種方法的結果皆相同。在一開始選擇群心會影響到後面的分類結果，從結果看來，兩群分佈較不平均 (Cluster 0 樣本較多)，但有有效切分兩群之效果。

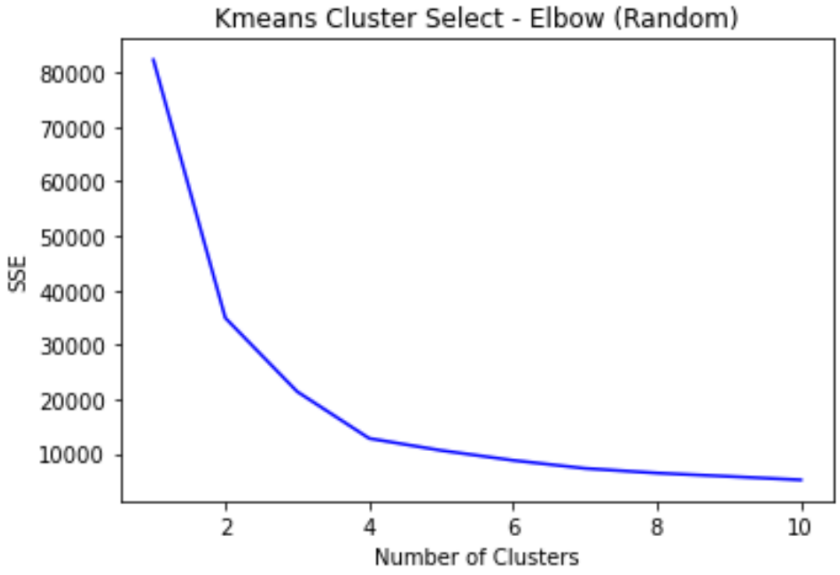
2. 承第 1 題，請利用 Elbow 方法找出 K 應該設置多少？ 請提供參考圖如下圖一。

Init = Random

程式碼

```
clusters = [num for num in range(1,11)]
SSE = []

for k in clusters:
    k_means = cluster.KMeans(n_clusters=k, init='random')
    k_means.fit(market_f_cluster)
    SSE.append(k_means.inertia_)

plt.plot(clusters, SSE, color = "blue")
plt.title('Kmeans Cluster Select - Elbow (Random)')
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
```

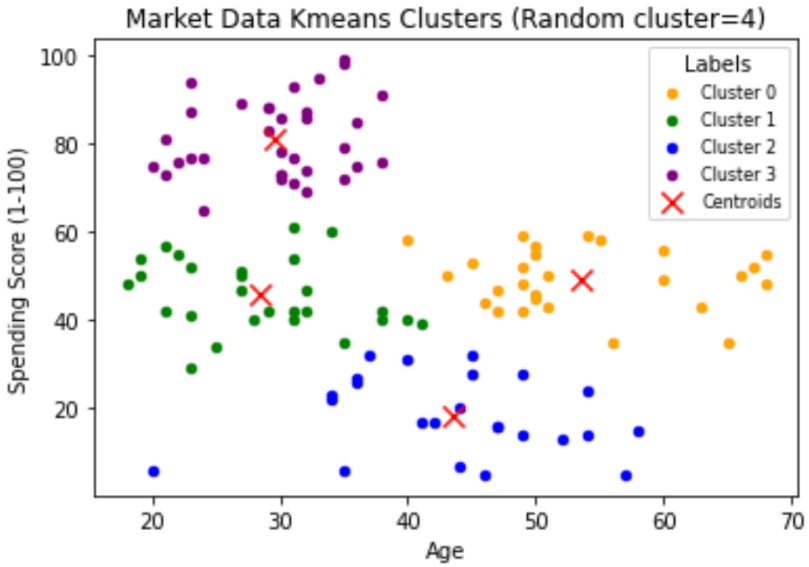
執行結果



選擇最佳分群(Cluster=4)

	Age	Spending Score (1-100)
1	28.370	45.704
3	29.618	81.235
2	43.583	18.500
0	53.630	49.296

分群結果



Init = K-means++

程式碼

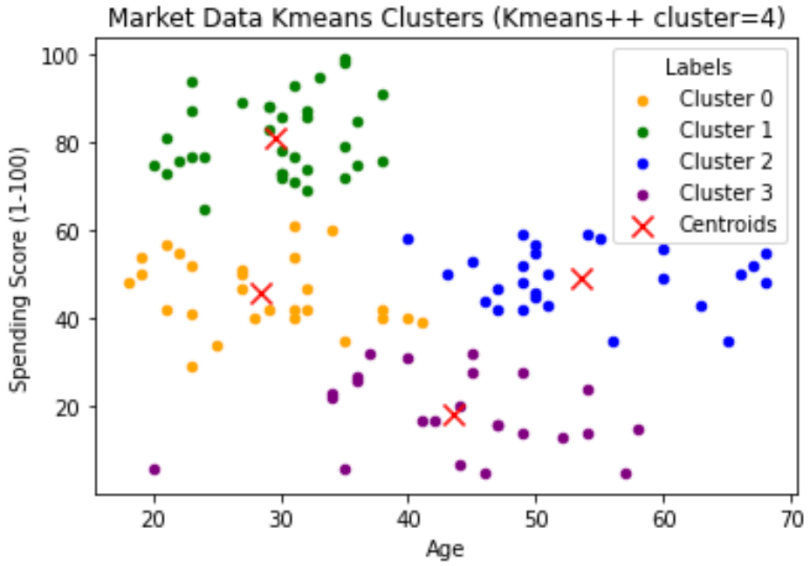
```
clusters = [num for num in range(1,11)]
SSE = []

for k in clusters:
    k_means = cluster.KMeans(n_clusters=k, init='k-means++')
    k_means.fit(market_f_cluster)
    SSE.append(k_means.inertia_)

plt.plot(clusters, SSE, color = "green")
plt.title('Kmeans Cluster Select - Elbow (Kmeans++)')
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
```

執行結果

選擇最佳分群(Cluster=4)

	Age	Spending Score (1-100)
0	28.370	45.704
1	29.618	81.235
3	43.583	18.500
2	53.630	49.296

分群結果



Q2 Summary

這題同樣也用了 random 和 Kmeans++來選群中心，兩種方法並無差異，效果相同。而根據 Elbow 方法，此資料集切分成 4 群為最理想狀態，比起前面所設的兩群 (k=2)，分成四群 (k=4) 後，每群之間分佈較均勻，且樣本數差不多，也有效切分四群樣本。

3. 從 marketData 的資料中，取所有客戶的三個欄位: Age、Annual Income、Spending Score 進行客戶分群，請使用 K-means 分群法，當 K＝3 時，請列出每一群的中心點，例如 C1 中心點: Age=33.3、Annual Income=87.1、Spending Score=88.1。3 個中心點的列表請根據中心點的 Age 數值升冪排序。

Init = Random

程式碼

```
k_means_random = cluster.KMeans(n_clusters=3, init='random', max_iter=30, random_state=99)
k_means_random.fit(market_all_cluster)
center_random = k_means_random.cluster_centers_
center_random_df = pd.DataFrame(center_random,
                                columns=market_all_cluster.columns).sort_values(by=['Age'], ascending=True)
display(center_random_df)
```

執行結果

	Age	Annual Income (k$)	Spending Score (1-100)
2	32.692	86.538	82.128
1	40.325	44.154	49.829
0	40.395	87.000	18.632

C0 中心點為：Age＝40.395

Annual Income＝87.000

Spending Score＝18.632

C1 中心點為：Age＝40.325

Annual Income＝44.154

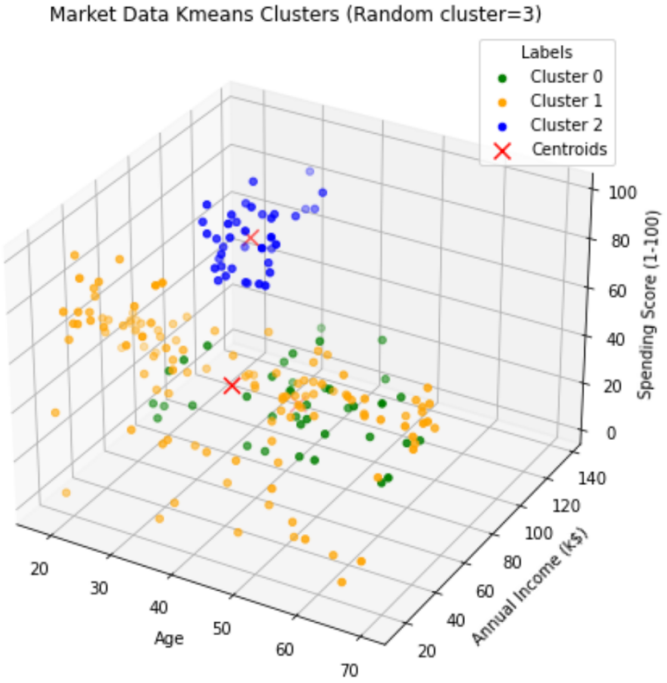
Spending Score＝49.829

C2 中心點為：Age＝32.692

Annual Income＝86.538

Spending Score＝82.128

分群結果



Init = K-means++

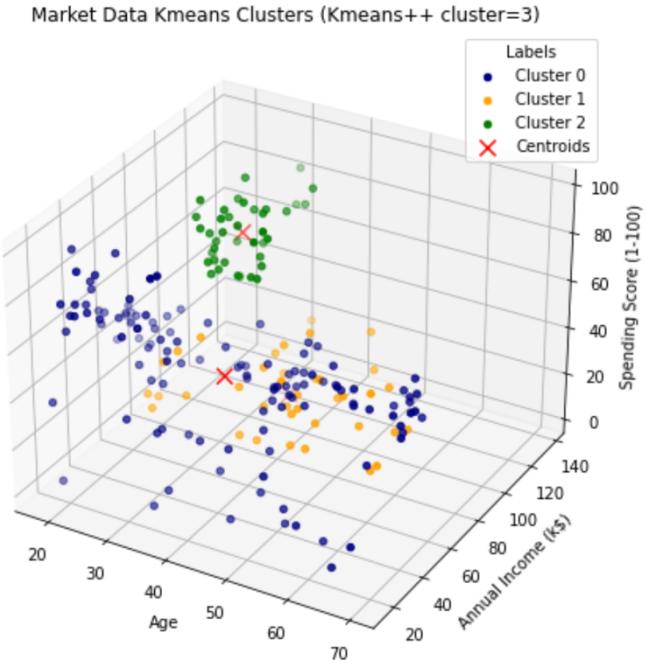
程式碼

```
k_means_k = cluster.KMeans(n_clusters=3, init='k-means++', max_iter=30, random_state=99)
k_means_k.fit(market_all_cluster)
center_k = k_means_k.cluster_centers_
center_k_df = pd.DataFrame(center_k,
                           columns=market_all_cluster.columns).sort_values(by=['Age'], ascending=True)
display(center_k_df)
```

執行結果

	Age	Annual Income (k$)	Spending Score (1-100)
2	32.692	86.538	82.128
0	40.325	44.154	49.829
1	40.395	87.000	18.632

分群結果



Q3 Summary

可以從圖中看到，分為 3 群時，Cluster0 擁有較多的樣本數，每群之間的樣本數分佈較不均勻。

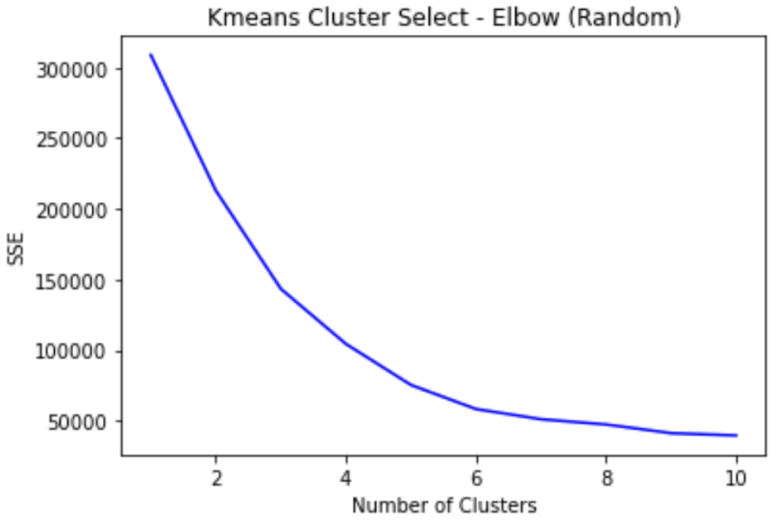
4. 承第 3 題，請利用 Elbow 方法找出 K 應該設置多少？ 請提供參考圖如圖一。

Init = Random

程式碼

```
clusters = [num for num in range(1,11)]
SSE = []

for k in clusters:
    k_means = cluster.KMeans(n_clusters=k, init='random')
    k_means.fit(market_all_cluster)
    SSE.append(k_means.inertia_)

plt.plot(clusters, SSE, color = "blue")
plt.title('Kmeans Cluster Select - Elbow (Random)')
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
```

執行結果

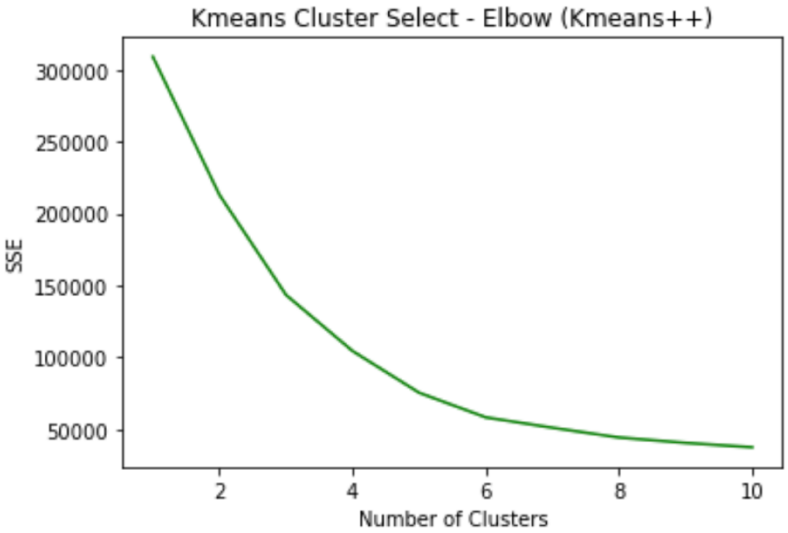


Init = K-means++

程式碼

```
clusters = [num for num in range(1,11)]
SSE = []

for k in clusters:
    k_means = cluster.KMeans(n_clusters=k, init='k-means++')
    k_means.fit(market_all_cluster)
    SSE.append(k_means.inertia_)

plt.plot(clusters, SSE, color = "green")
plt.title('Kmeans Cluster Select - Elbow (Kmeans++)')
plt.xlabel('Number of Clusters')
plt.ylabel('SSE')
```

執行結果



Q4 Summary

根據 Elbow 方法，這題並無明顯的最佳解，因此，透過上圖我選擇了分為 3 群作為最佳解，同樣比較了 random 和 K-means++的找初始群新方法，並無明顯差異。

5. 承第 4 題的 K 值設置，假設現在有一個行銷活動，請問你要怎麼透過 K-means 分群結果進行篩選，選擇一群目標客群，請列出此群的中心點，並解釋你的理由。

程式碼

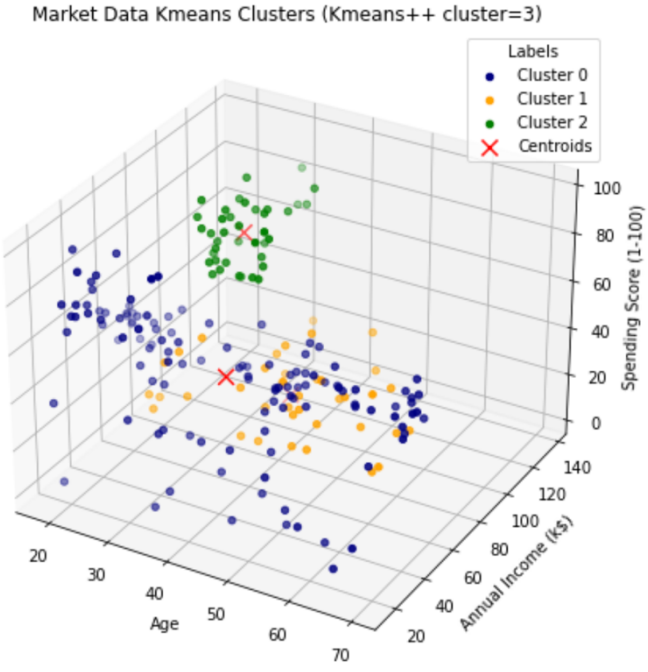
```
k_means_k = cluster.KMeans(n_clusters=3, init='k-means++', max_iter=30, random_state=99)
k_means_k.fit(market_all_cluster)
center_k = k_means_k.cluster_centers_
center_k_df = pd.DataFrame(center_k, columns=market_all_cluster.columns)
display(center_k_df)
```

執行結果

	Age	Annual Income (k$)	Spending Score (1-100)
0	40.325	44.154	49.829
1	40.395	87.000	18.632
2	32.692	86.538	82.128

分群結果

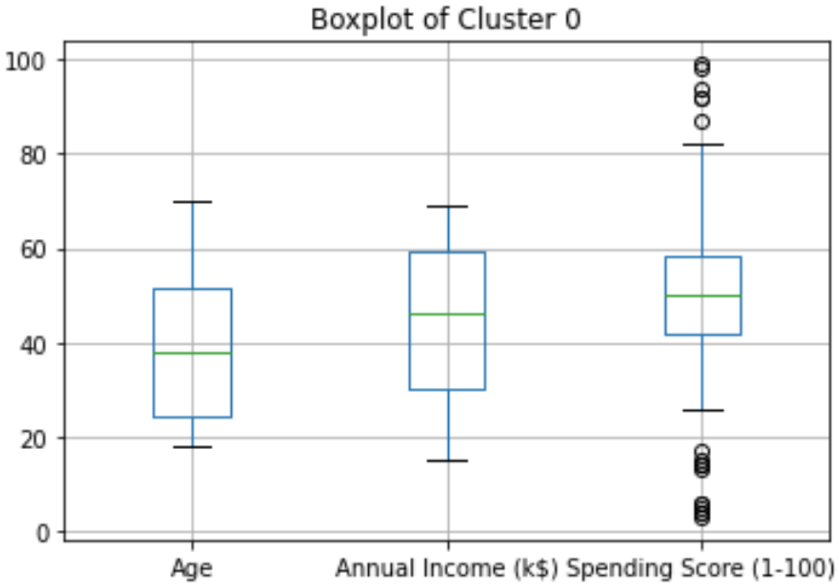
```
Counter({0: 123, 2: 39, 1: 38})
```



各群資訊

Cluster 0:

	Age	Annual Income (k$)	Spending Score (1-100)
count	123.000	123.000	123.000
mean	40.325	44.154	49.829
std	16.114	16.038	19.694
min	18.000	15.000	3.000
25%	24.500	30.000	42.000
50%	38.000	46.000	50.000
75%	51.500	59.500	58.500
max	70.000	69.000	99.000



Cluster 1:

	Age	Annual Income (k$)	Spending Score (1-100)
count	38.000	38.000	38.000
mean	40.395	87.000	18.632
std	11.377	16.271	10.916
min	19.000	70.000	1.000
25%	34.000	76.250	10.250
50%	41.500	80.000	16.500
75%	47.000	96.000	26.750
max	59.000	137.000	40.000

Cluster 2:

	Age	Annual Income (k$)	Spending Score (1-100)
count	39.000	39.000	39.000
mean	32.692	86.538	82.128
std	3.729	16.312	9.364
min	27.000	69.000	63.000
25%	30.000	75.500	74.500
50%	32.000	79.000	83.000
75%	35.500	95.000	90.000
max	40.000	137.000	97.000

Q5 Summary

根據第四題的 Elbow 方法，最後選擇分 3 群作為最佳解，可以從上面的資訊看到 Cluster0 有 123 個樣本，Cluster1 有 38 個，而 Cluster2 有 39 個樣本。深入每一群內來看，我會選擇 Cluster2 作為我的目標客群。Cluster2 顧客主要集中在 27-40 歲的顧客，可針對該年齡層的人來設計他們所流行的商品。另外，Cluster2 的顧客 Annual Income 也落在 69k - 120k，比起 Cluster0 的顧客，較能針對 Cluster2 的顧客來設計擁有較高品質且較高單價的商品。雖然從 Annual Income 來看，Cluster1 的顧客與 Cluster2 差不多，但就 Spending Score(顧客購買行為和物品分數)來看 Cluster2 是優於 Cluster1 的。最後，因 Cluster0 三項因素分佈較廣泛，也較難鎖定某一特定族群來做銷售設計，因此，最終選擇 Cluster2 來作為目標客群。