K-means clustering

SSE & Silhouette coefficient

10927207 蒲品憶

1. 步驟流程

```python
# 生成資料
X, y = make_blobs(n_samples=300, centers=4, random_state=42)


# 繪製散點圖
plt.scatter(X[:, 0], X[:, 1], c=y, cmap='viridis', edgecolors='k')
plt.title('Generated Data with make_blobs(by kiwi_tech)')
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.show()
```

```python
# 使用手肘法找到最佳的集群數
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300, n_init=10, random_state=0)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)

# 繪製手肘法圖
plt.plot(range(1, 11), wcss)
plt.title('Elbow Method(by kiwi_tech)')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')  # WCSS代表群內平方和
plt.show()
```

```python
# 使用輪廓分析法找到最佳的集群數
silhouette_scores = []
for i in range(2, 11):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300, n_init=10, random_state=0)
    kmeans.fit(X)
    silhouette_scores.append(silhouette_score(X, kmeans.labels_))

# 繪製輪廓分析法圖
plt.plot(range(2, 11), silhouette_scores)
plt.title('Silhouette Analysis(by kiwi_tech)')
plt.xlabel('Number of clusters')
plt.ylabel('Silhouette Score')
plt.show()
```

1. 使用 make_blobs()函式隨機生 300 個點，設定分成四群
2. SSE(sum of the squared error) / elbow methods / 手肘法
   i. 使用 kmeans.inertia_函式找出群中心，計算每一群中的每一個資料點到群中心的距離，**找出 SSE 相對平緩的資料點作為 elbow point**，並以此 elbow point 選為群數。

$$SSE = \sum_{i=1}^{k} \sum_{p \in C_i} |p - mi|^2$$

誤差平方和（sum of the squared errors, SSE）

3. Silhouette coefficient / Silhuette scores / 輪廓係數法

i. 使用 silhoutte _score_函式找出群中心，目的是找出同一群的資料點內最近(凝聚度越小的值)，不同群越分散(**分離度越高的值**)，用來滿足集群主要的目標。也就是說，S 越大，K 值越符合。

$$S(i) = \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)}$$

, S 介於 [-1, 1]

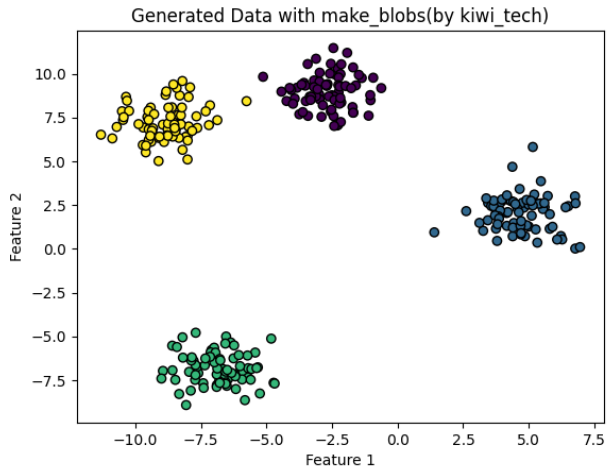
4. 繪製資料集圖、SSE 圖、Silhuette scores 圖

✧ 額外補充: Silhouette coefficient 為 SSE 的加強改進版。

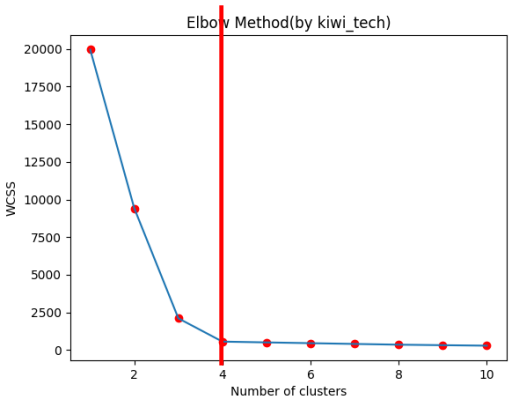
2. Good Example

➢ 使用資料集

各特徵群之間界線分明



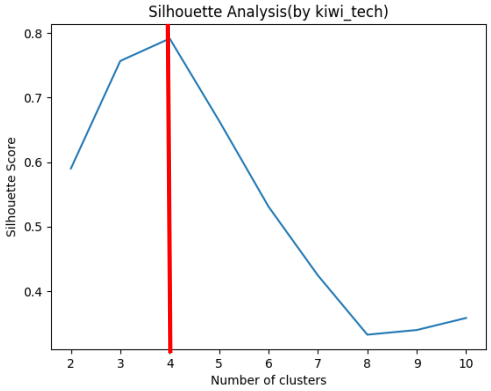
➢ SSE(sum of the squared error)



從圖中可以看出，inertia/WCSS 急遽下降直到 K=4 後趨於平緩，可得知，elbow point 為 k = 4

答案: 此資料集在 k=4 時有最佳解法

➢ Silhouette coefficient



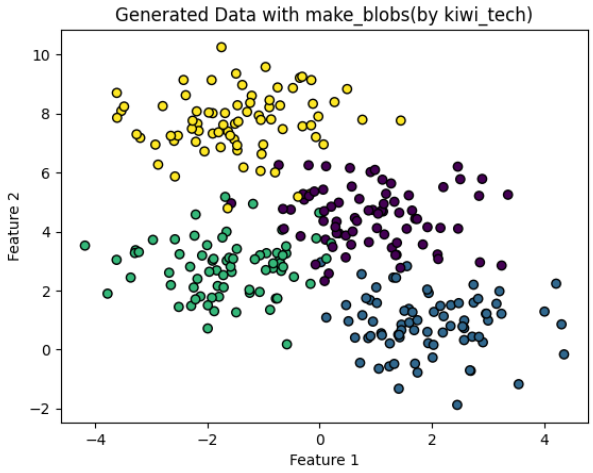
從圖中可以看出，Silhouette score 在 k=4 時到達臨界值，因此可知...

答案: 此資料集在 k=4 時有最佳解法

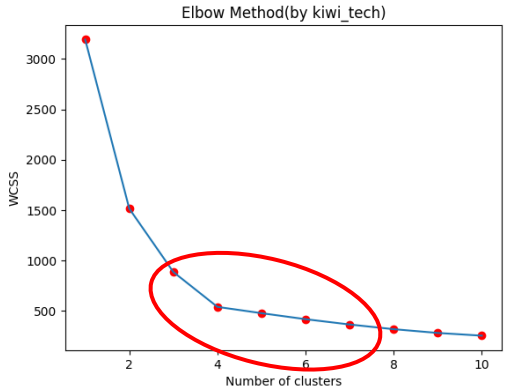
3. Bad Example

➢ 使用資料集

各特徵群之間界線不分明



➢ SSE(sum of the squared error)



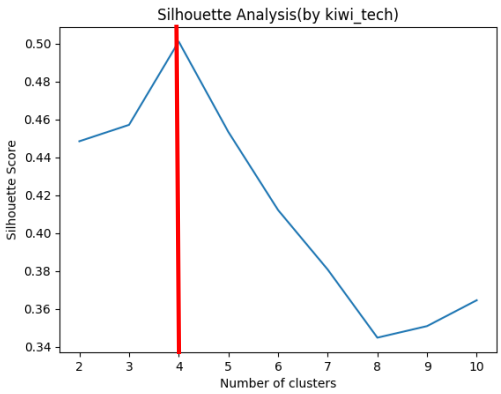
從圖中可以看出，inertia/WCSS 在 k=3、4、5、6、7 產生相對不明確的結果，因此很難選擇合適點。

答案: Bad Example 難以判斷何時為最佳解法

✧ 因此我才上下都有做 Silhouette coefficient 方法比較兩者差異，並讓大家明白 Silhouette coefficient 有改善 SSE 會因為各特徵群之間界線不分明而導致結果不明確的問題，即使

遇到各特徵群之間界線不分明的問題，還是能判斷出找出最佳解。

- Silhouette coefficient



從圖中可以看出，Silhouette score 在 k=4 時到達臨界值，因此可知…

答案: 此資料集在 k=4 時有最佳解法