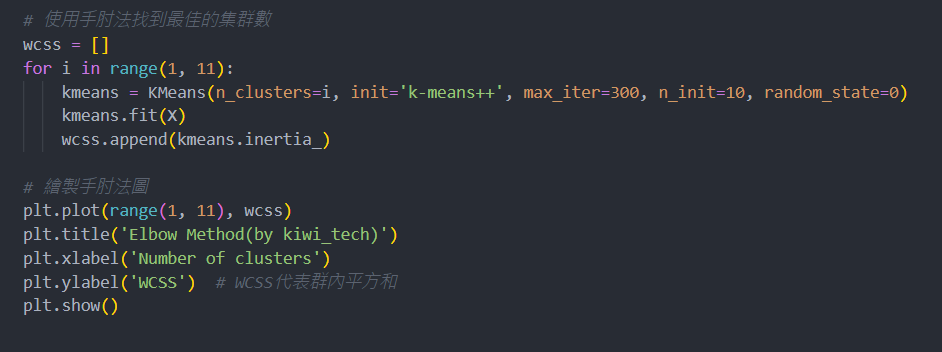
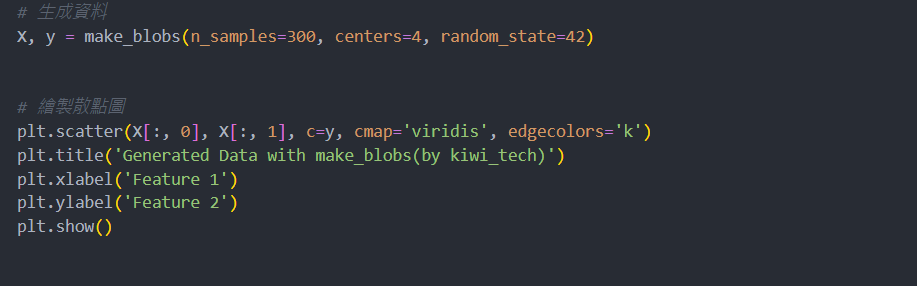
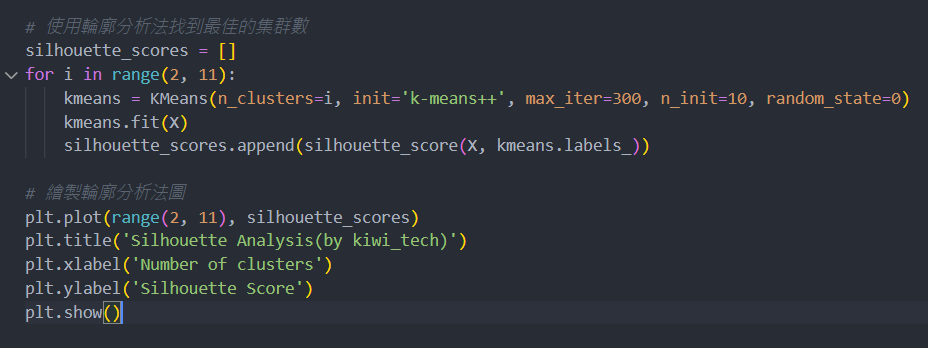
K-means clustering

SSE & Silhouette coefficient

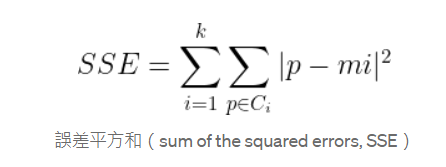
10927207 蒲品憶

1.步驟流程

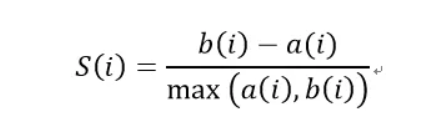




1. 使用make\_blobs()函式隨機生300個點，設定分成四群
2. SSE(sum of the squared error) / elbow methods / 手肘法
   * 1. 使用kmeans.inertia\_函式找出群中心，計算每一群中的每一個資料點到群中心的距離，**找出 SSE 相對平緩的資料點作為elbow point**，並以此elbow point選為群數。



1. Silhouette coefficient / Silhuette scores / 輪廓係數法
   * 1. 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 介於 [-1, 1]

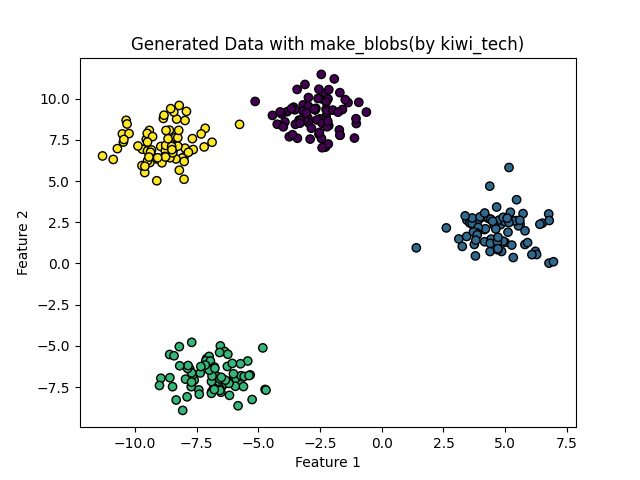
1. 繪製資料集圖、SSE圖、Silhuette scores圖

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

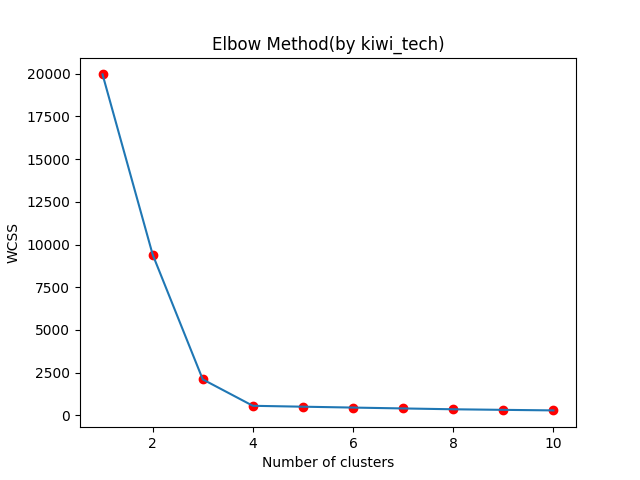
2.Good Example

* 使用資料集

各特徵群之間界線分明



* SSE(sum of the squared error)



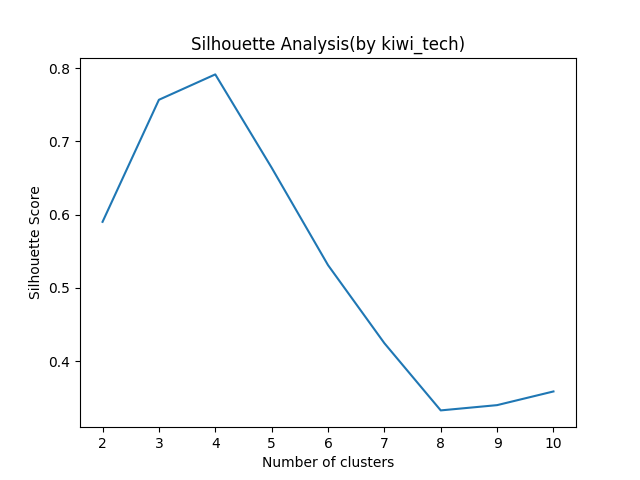
從圖中可以看出，inertia/WCSS急遽下降直到K=4後趨於平緩，可得知，elbow point為 k = 4

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…

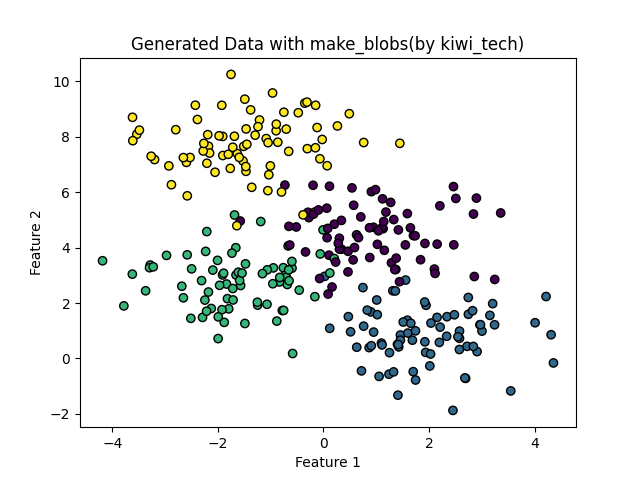
答案: 此資料集在k=4時有最佳解法



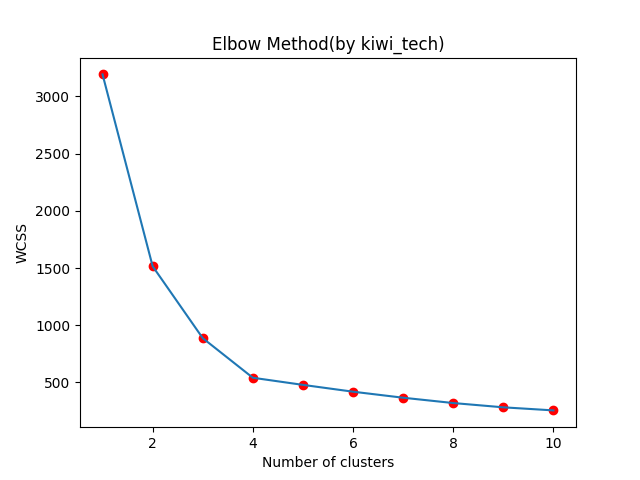
3.Bad Example

* 使用資料集

各特徵群之間界線不分明



* SSE(sum of the squared error)

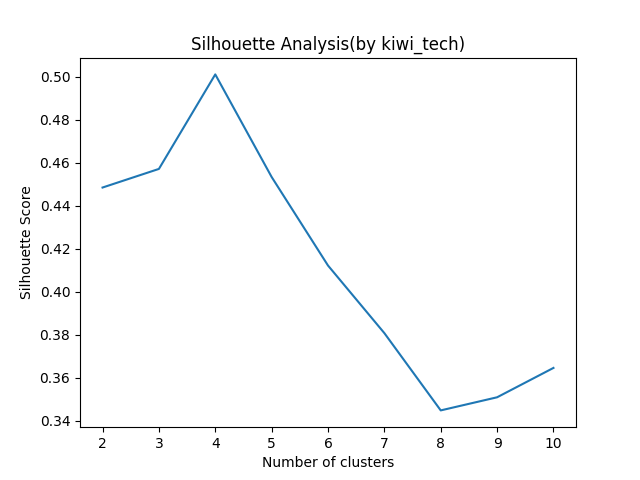


從圖中可以看出，inertia/WCSS在k=3、4、5、6、7產生相對不明確的結果，因此很難選擇合適點。

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