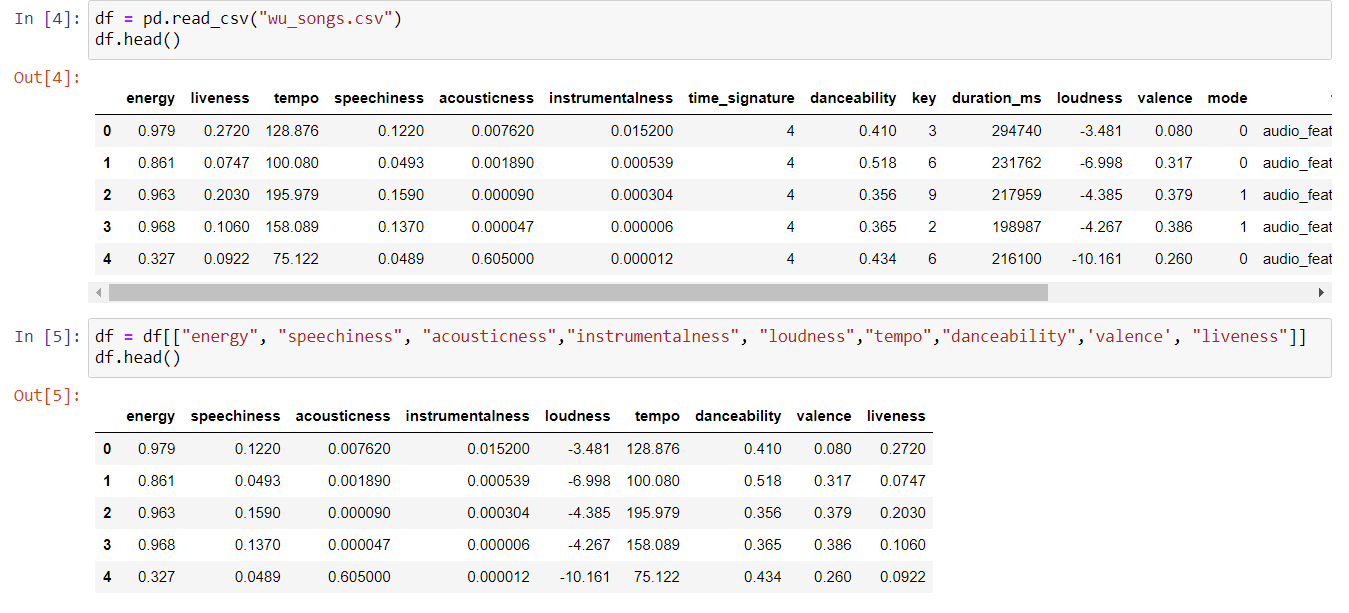
ETC\_HW4\_107403020 李泳輝

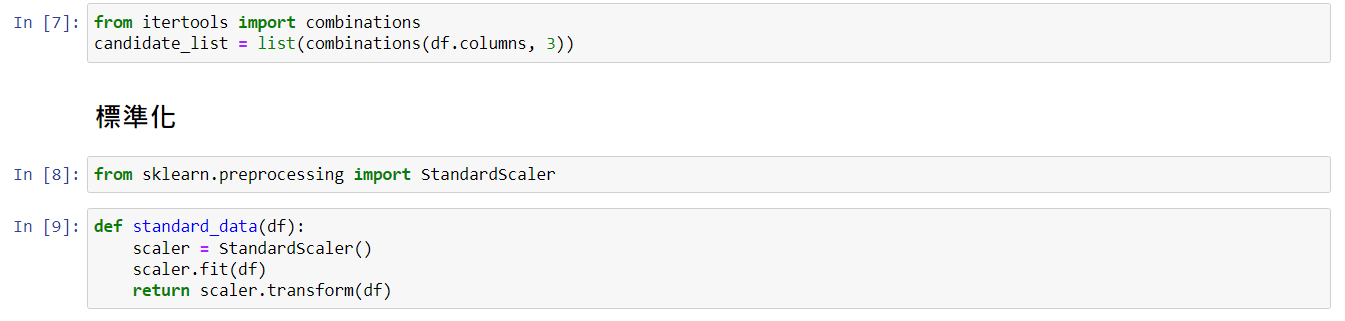
1. Python

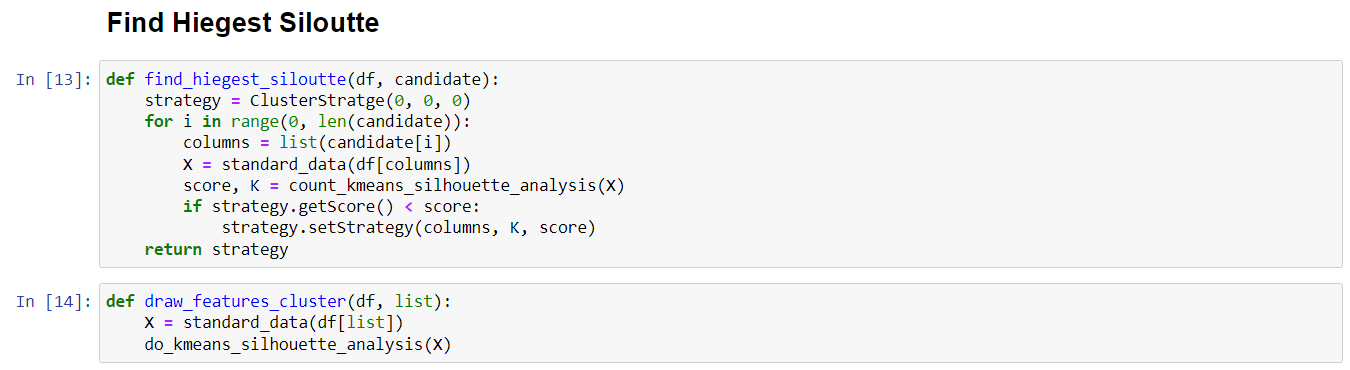
1. 載入資料並刪除除了"energy", "speechiness", acousticness","instrumentalness", "loudness", "tempo","danceability",'valence', "liveness"以外之欄位(使用 pandas dataframe) (2%)



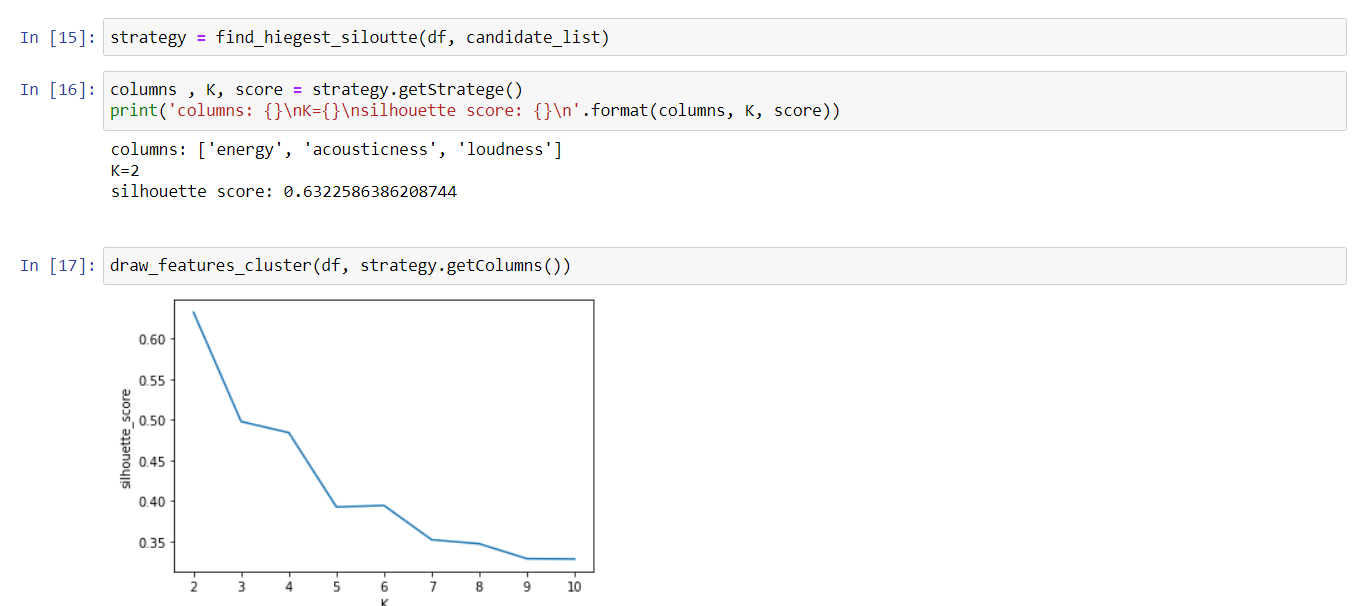
2. 將剩下的欄位做特徵篩選的動作，並使用 kmeans silhouette analysis 的方法，找出在哪三個欄位的情況下(需考慮所有組合)，分 X 群會有最高的 silhouettescore。請找出 X 與 silhouette score 還有是哪三個欄位。(20%)

* 建立相關function





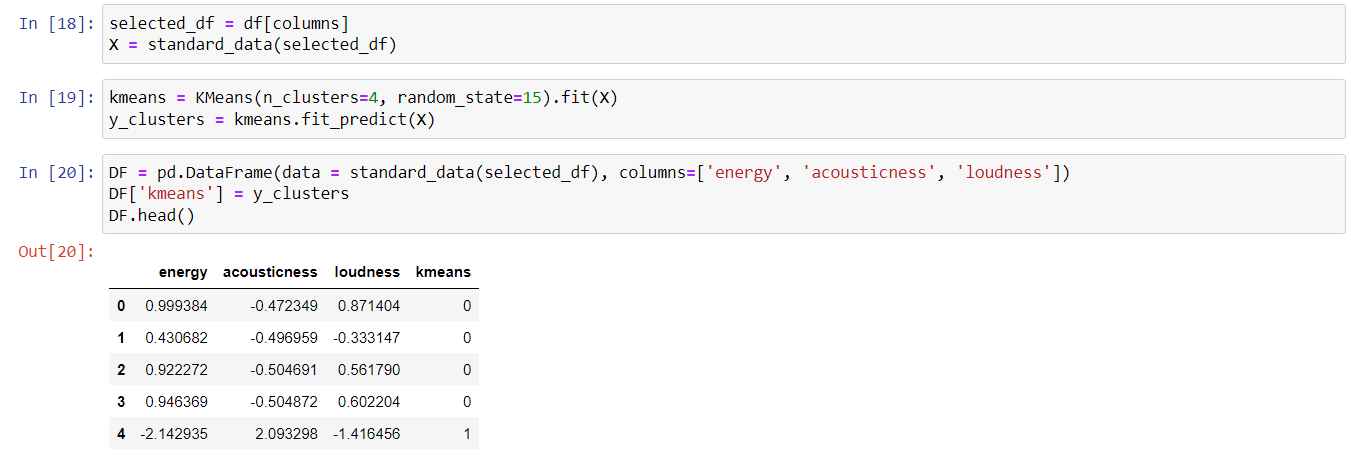
* 執行

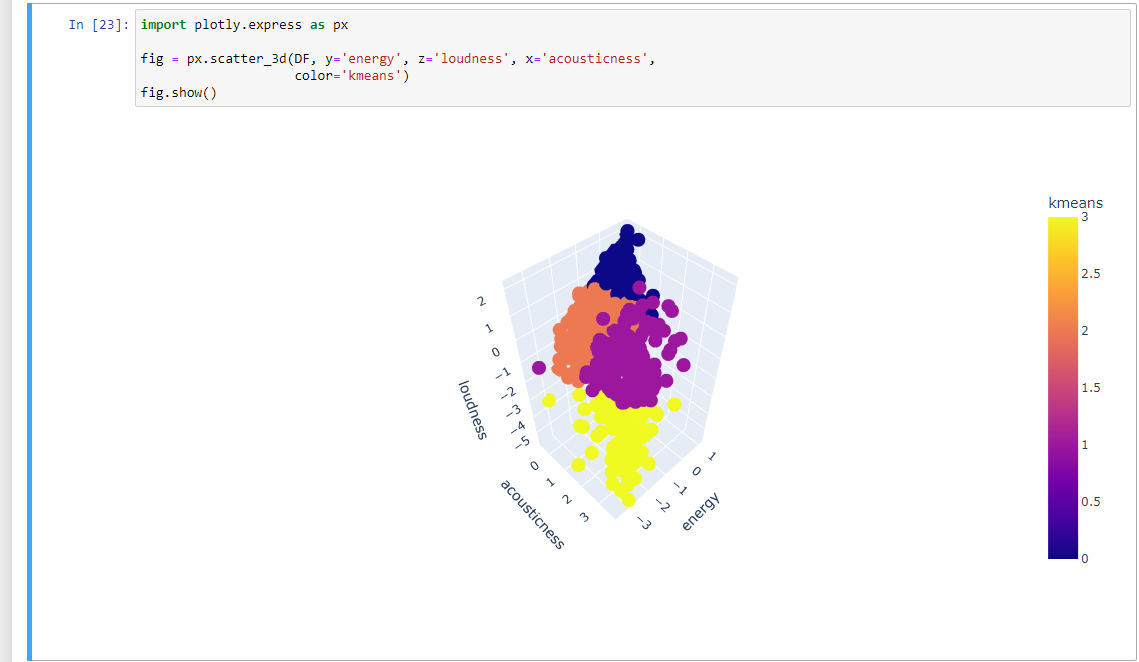


請解釋 silhouette 分析法 與 elbow 轉折判斷法的差別(3%)

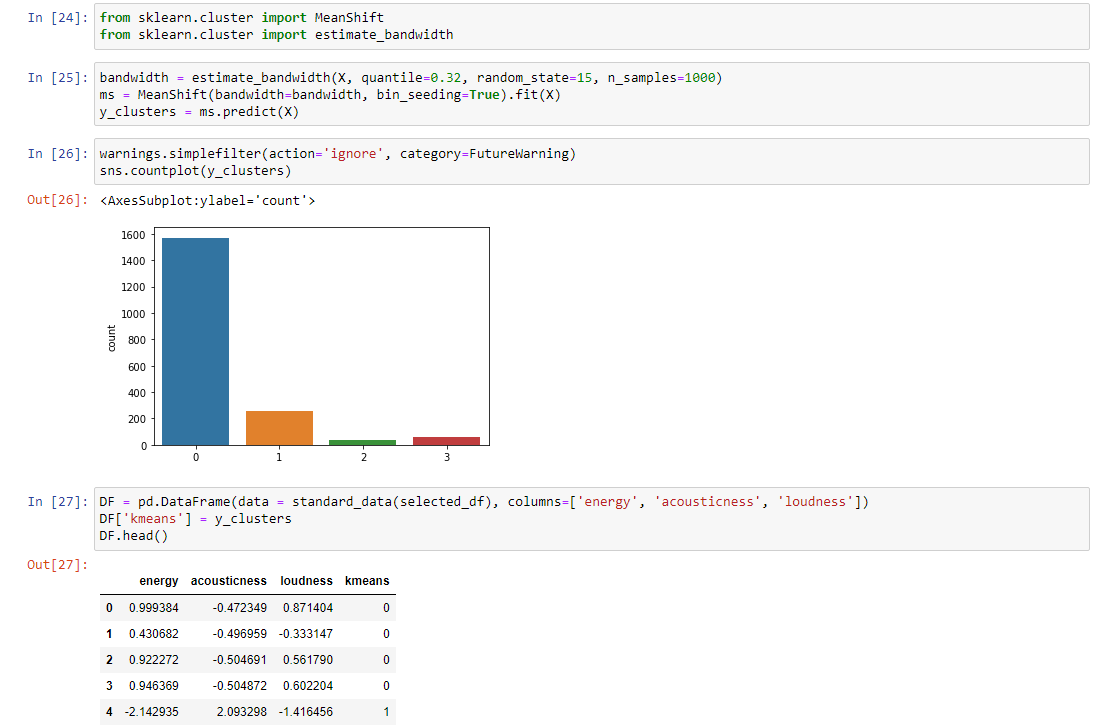
* elbow 轉折判斷法是針對不同K的分群，計算樣本點到各自分群中心距離的總和。當分群的數目增加，距離的總和自然會下降。當K值大於真正的K值後，下降就不會那麼明顯。這樣正確的K值就會在這個轉折點上。但若遇到某些特例，像是下降幅度趨緩的圖，就不適合用elbow判斷方式。
* silhouette分析法會衡量物件與所屬cluster的相似度，即內聚性。silhouette的分數越高代表著物件與所屬的cluster有越密切的關聯。silhouette的缺點是他的計算複雜度為O(n^2)，若資料量上到百萬，計算算量會非常巨大。

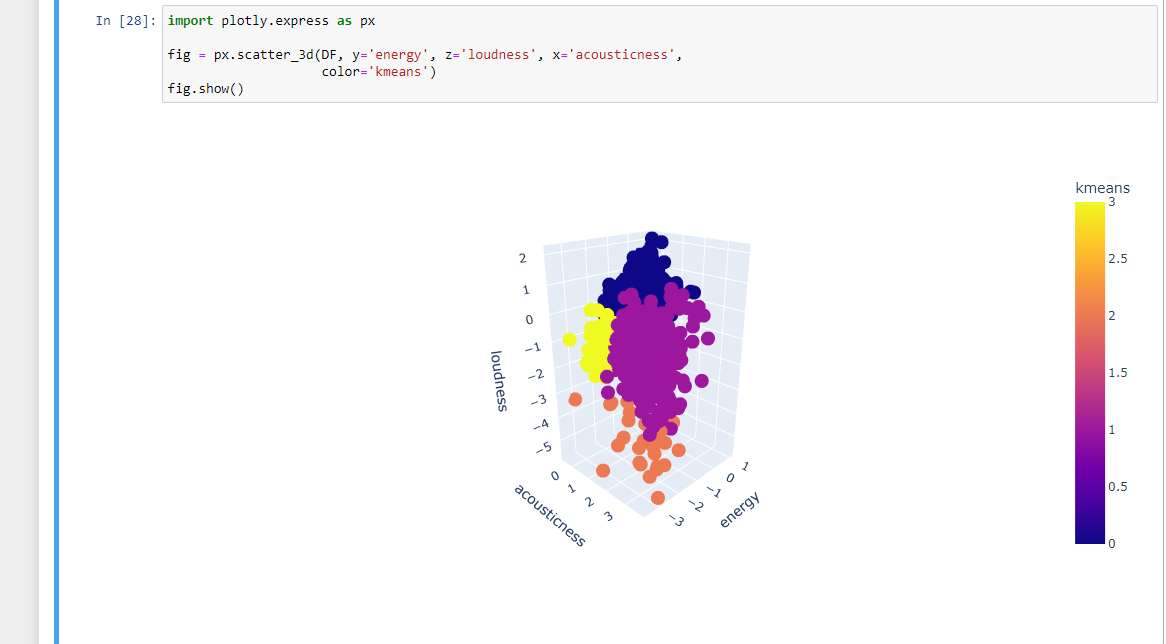
# 3. 使用剛剛找出來的欄位用 k-means 做分群。超參數設定為 n\_cluster=4 ,random\_state=15。並使用 plotly 繪製出 3d 圖形如以下所示(15%)：



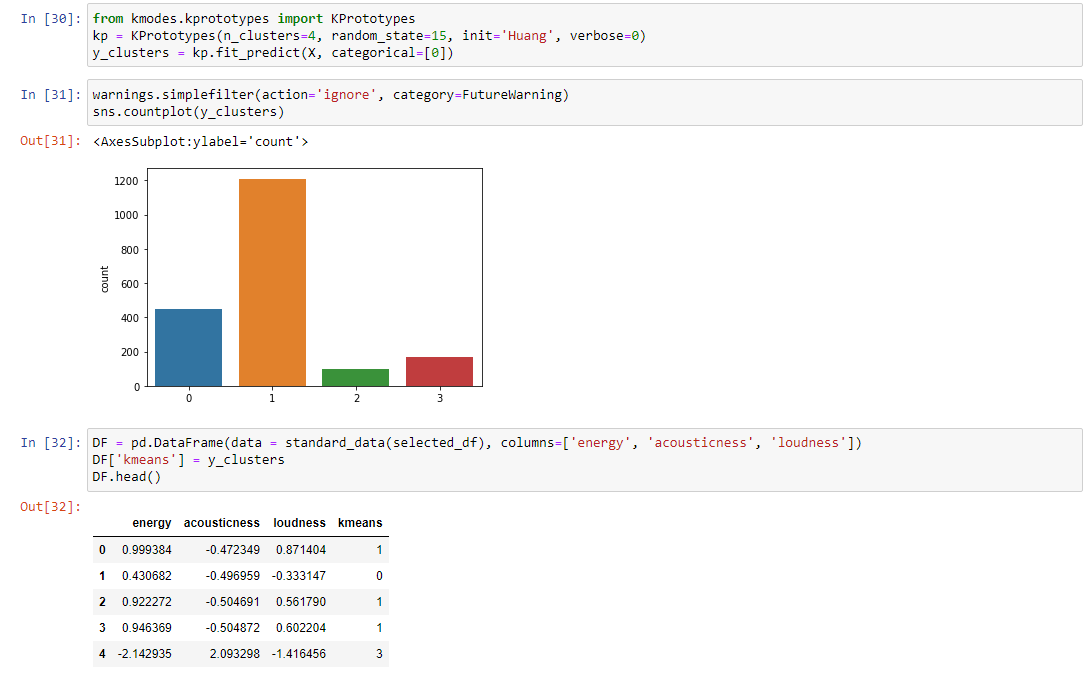


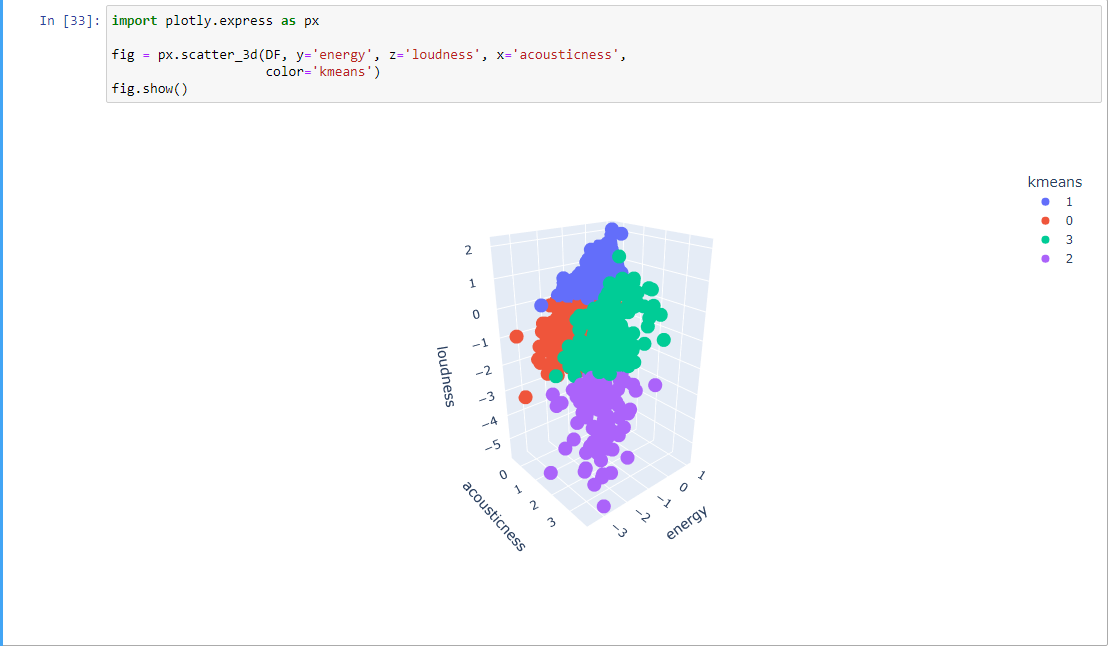
4. 使用剛剛找出來的欄位用 Meanshift 做分群(15%) 請找出最佳的 estimate\_bandwidth. 超參數設定為 random\_state=15,quantile=0.32, n\_samples=1000





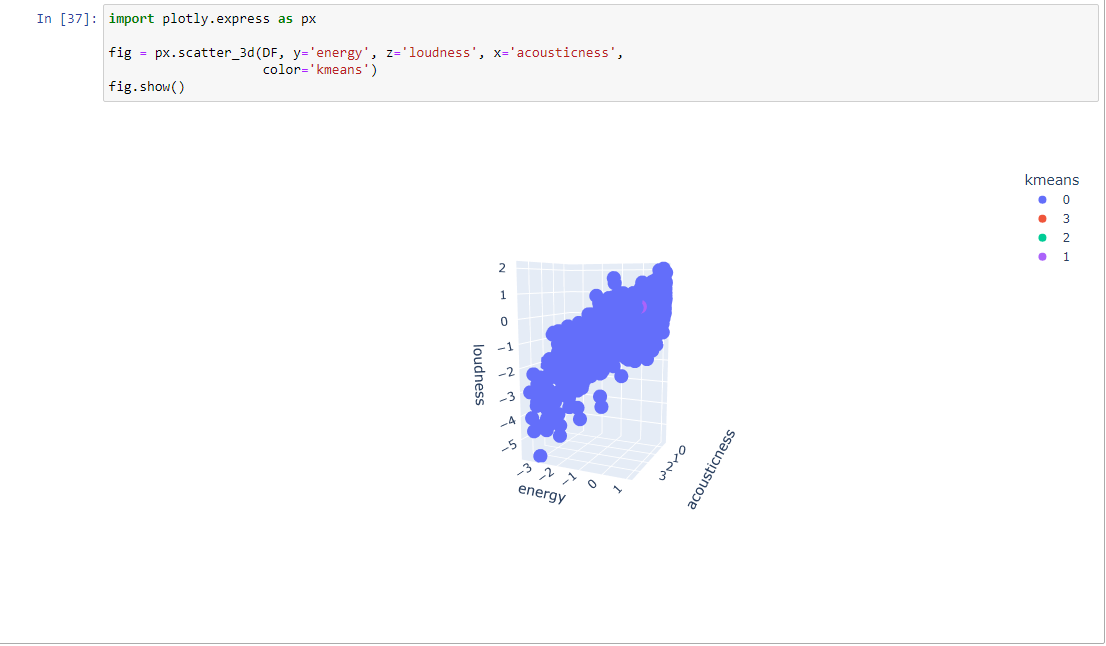
5. 使用剛剛找出來的欄位用 k-prototypes 做分群。超參數設定為 n\_cluster=4, random\_state=15,init=’Huang’,verbose=0。並使用 plotly 繪製出 3d 圖形如第三題的圖(15%)





6. 使用剛剛找出來的欄位用 k-modes 做分群。超參數設定為 n\_cluster=4,random\_state=15,init=’Huang’,verbose=0。並使用 plotly 繪製出 3d 圖形如第三題的圖(15%)



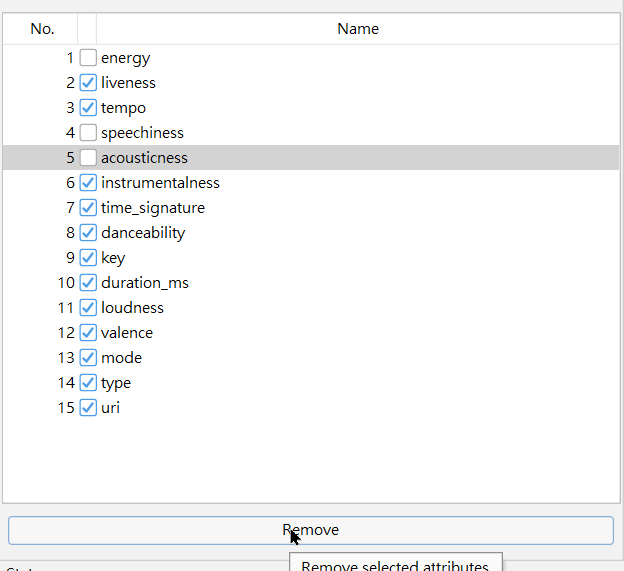


7. 請比較說明上述四種分群法的差異(5%)

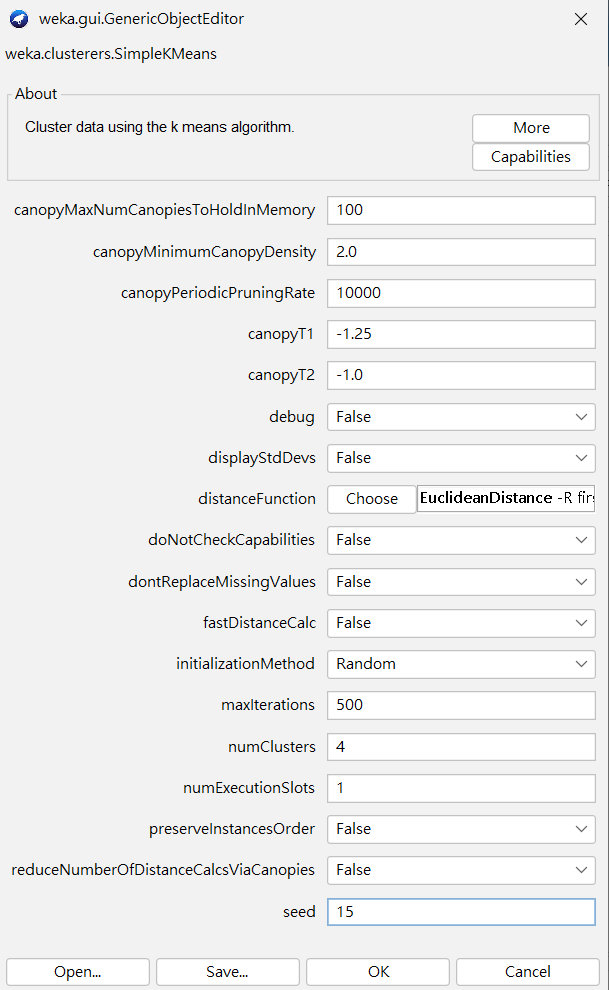
* K-Means演算法中，最終的cluster結果容易受到初始中心的影響。
* Mean Shift演算法與K-Means一樣是基於Cluster中心的演算法，但Mean Shift演算法不需要事先制定類別個數K。
* K-modes演算法可以看做K-means算法在非數值屬性集合的版本，將原本K-Means使用的歐基里德距離替換成相異程度的算法。
* K-Propertypes可以對離散與數值屬性兩種混合的數據進行clustering，其結合了K-Means與K-modes算法。

1. Weka

* 只留下要用的三個欄位



* 設定SimpleKMeans超參數



* 執行結果

