ML 無監督式學習

(Machine Learning, Unsupervised Learning)

利用 KMeans 將花(Iris)分成三種聚類

point fit 到模型裡面後，scatterplot 呈現出結果

script.py — Light Mode

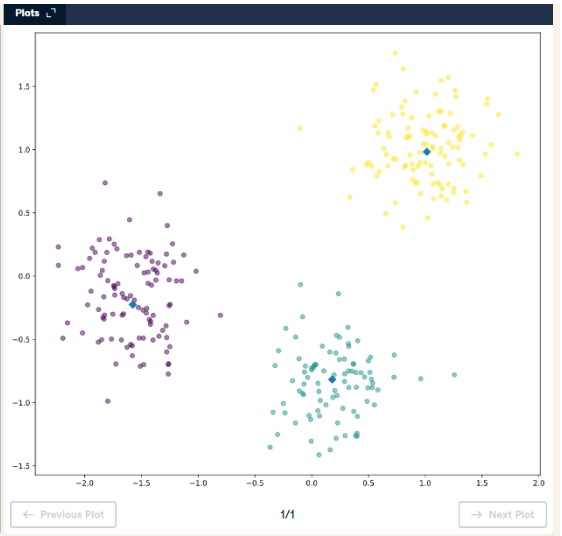
```
# Import KMeans
from sklearn.cluster import KMeans

# Create a KMeans instance with 3 clusters: model
model = KMeans(n_clusters=3)

# Fit model to points
model.fit(points)

# Determine the cluster labels of new_points: labels
labels = model.predict(new_points)

# Print cluster labels of new_points
print(labels)

```

Run Code | Submit Answer

script.py — Light Mode

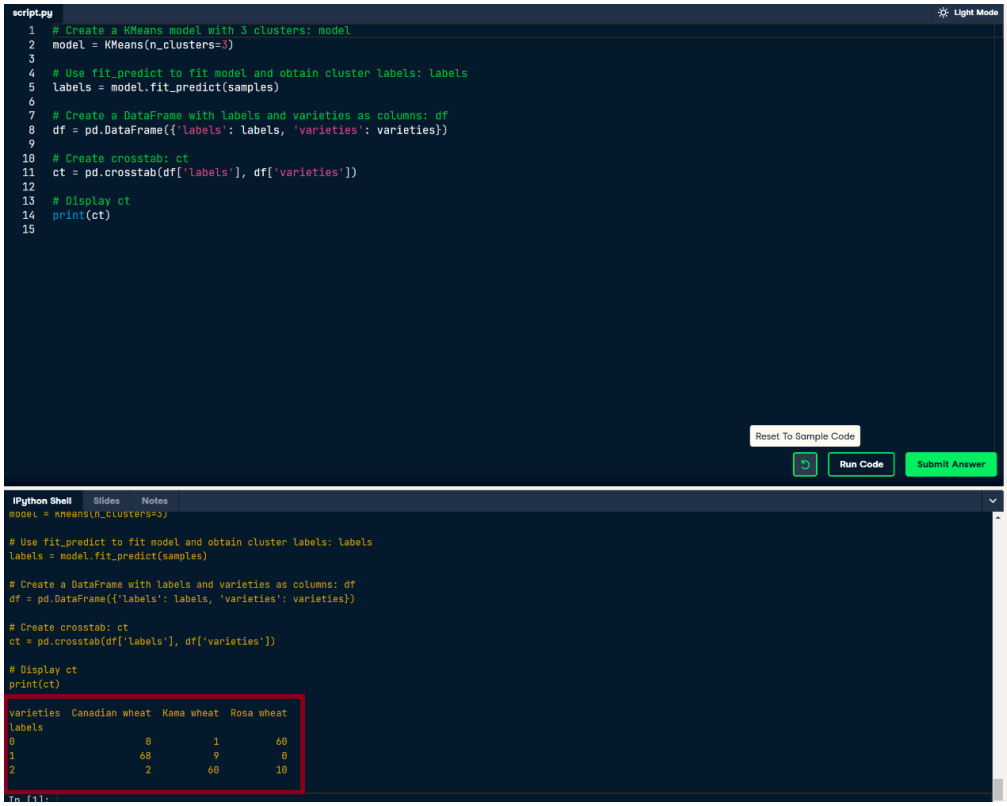
```
# Import pyplot
from matplotlib import pyplot as plt

# Assign the columns of new_points: xs and ys
xs = new_points[:,0]
ys = new_points[:,1]

# Make a scatter plot of xs and ys, using labels to define the
colors
plt.scatter(x= xs, y= ys, c=labels, alpha=0.5)

# Assign the cluster centers: centroids
centroids = model.cluster_centers_

# Assign the columns of centroids: centroids_x, centroids_y
centroids_x = centroids[:,0]
centroids_y = centroids[:,1]

# Make a scatter plot of centroids_x and centroids_y
plt.scatter(x= centroids_x, y=centroids_y, marker='D', s=50)
plt.show()

```

Run Code | Submit Answer



評估聚類 Evaluating a Clustering

慣性(Inertia)評估最佳聚類數量(KMeans)
慣性越小聚類越密集，而 KMeans 開始緩降為最佳解(紅框)
下圖，利用交叉表(cross-tabulation)評估聚類分布狀況(紅框)
評估花及品種是否有對應



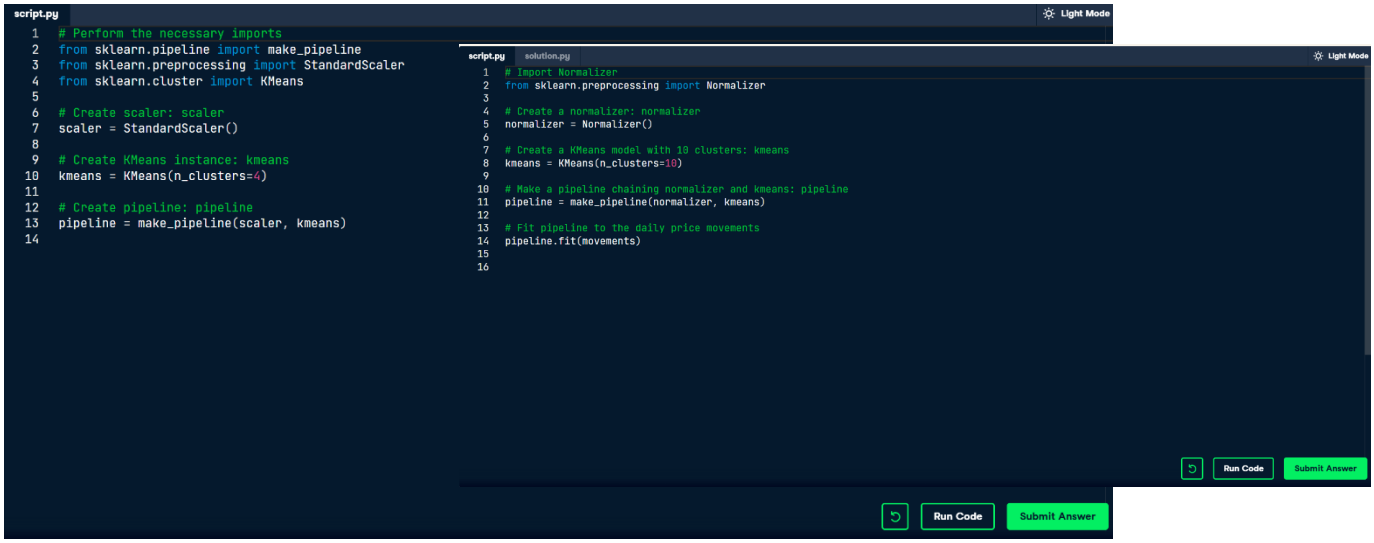
優化調整類組 Transforming Features for Better Clusterings

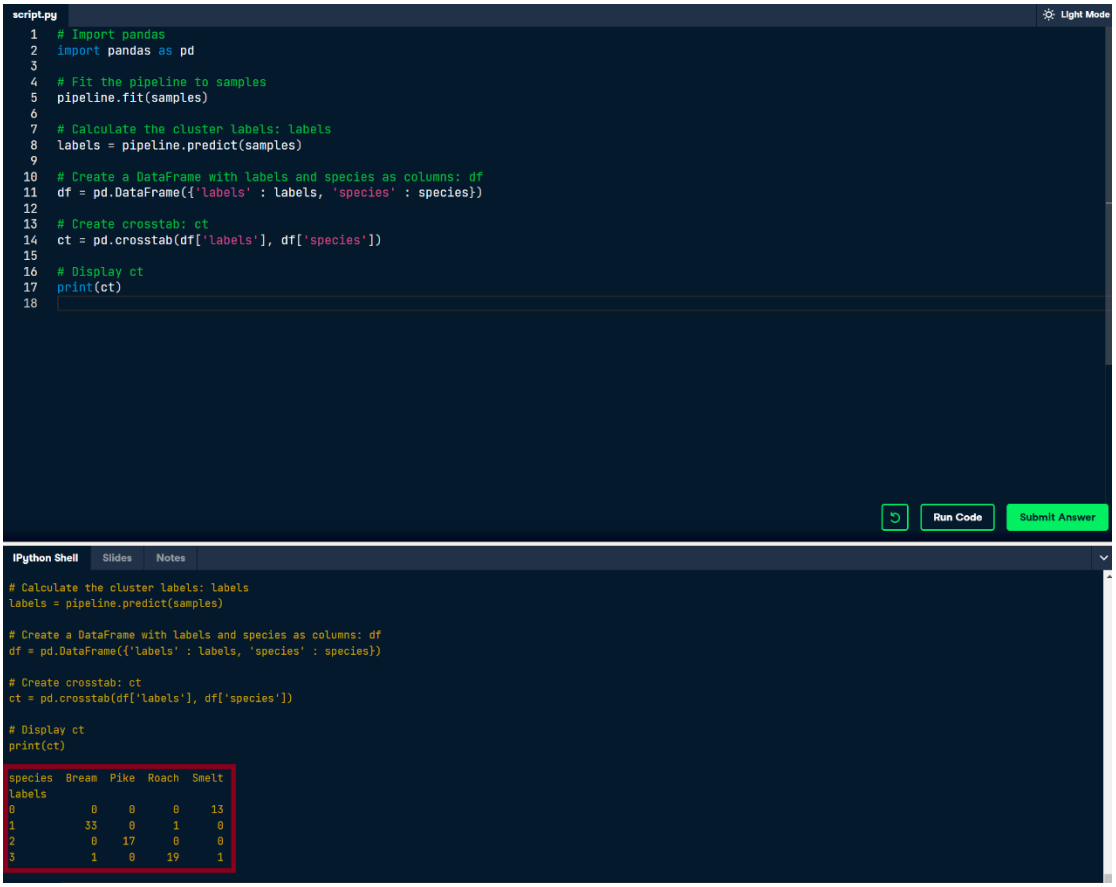
假設交叉表分布沒有對應

先用 StandarScaler 或 Normalizer 與 KMeans 建構一個 pipeline

即可將資料標準化在分組，調整資料

下圖為優化後的聚類，交叉表(魚的品種)相互對應



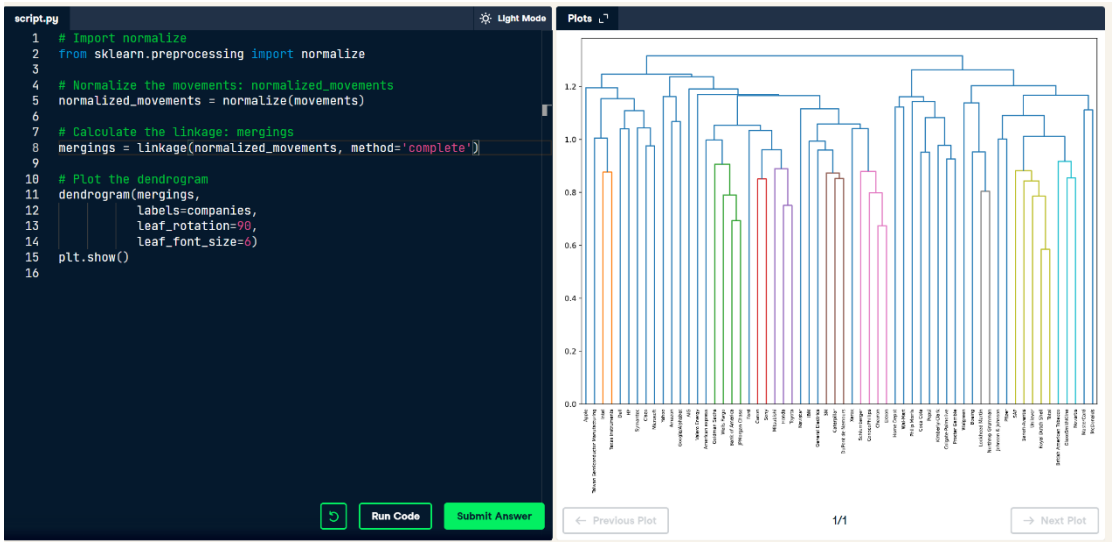


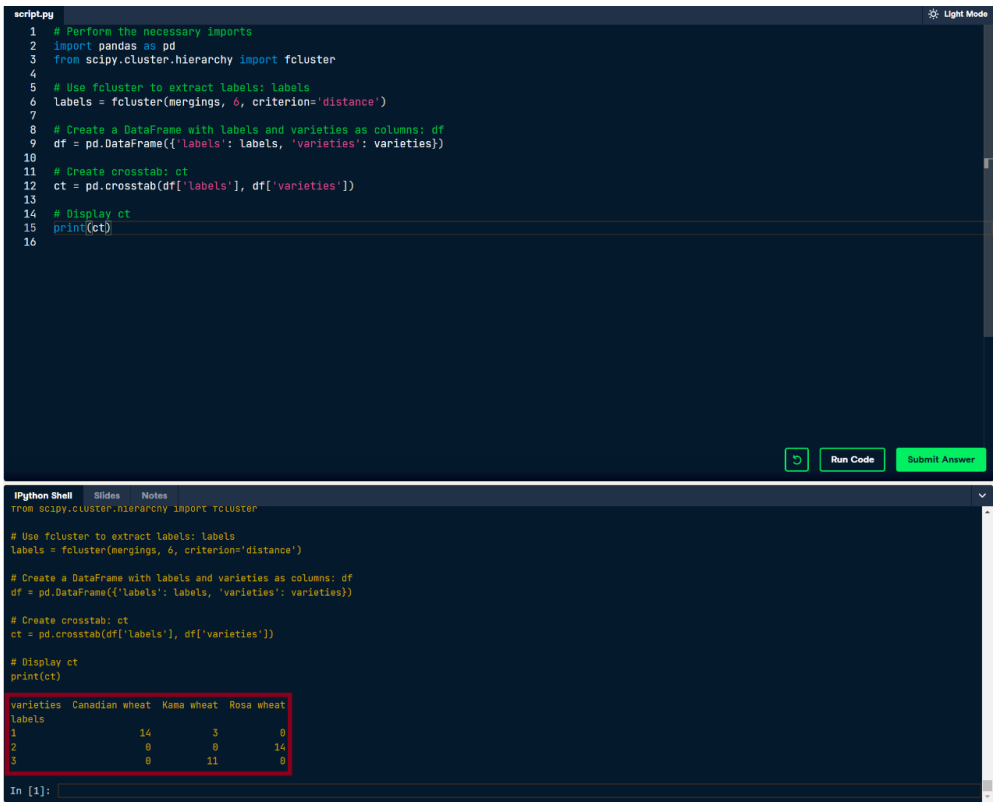
t-SNE 與層次聚類視覺化

Visualization with Hierarchical Clustering and t-SNE

Normalize 資料後用 dendrogram 視覺化

可用 fcluster 提取特定高度的聚類(下圖高度為 6)





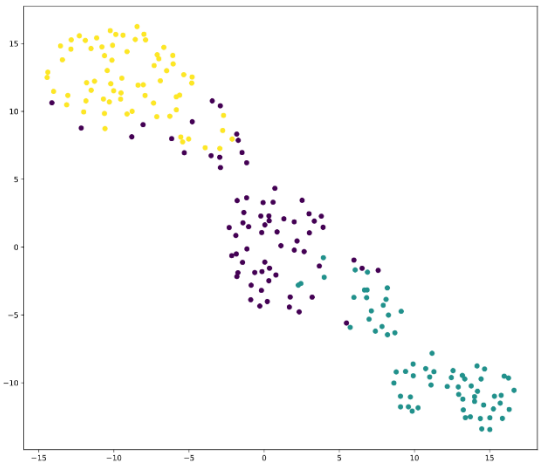
t-SNE 視覺化

script.py solution.py Light Mode

```python
# Import TSNE
from sklearn.manifold import TSNE

# Create a TSNE instance: model
model = TSNE(learning_rate=200)

# Apply fit_transform to samples: tsne_features
tsne_features = model.fit_transform(samples)

# Select the 0th feature: xs
xs = tsne_features[:,0]

# Select the 1st feature: ys
ys = tsne_features[:,1]

# Scatter plot, coloring by variety_numbers
plt.scatter(xs, ys, c=variety_numbers)
plt.show()
```

Run Code Submit Answer

Plots



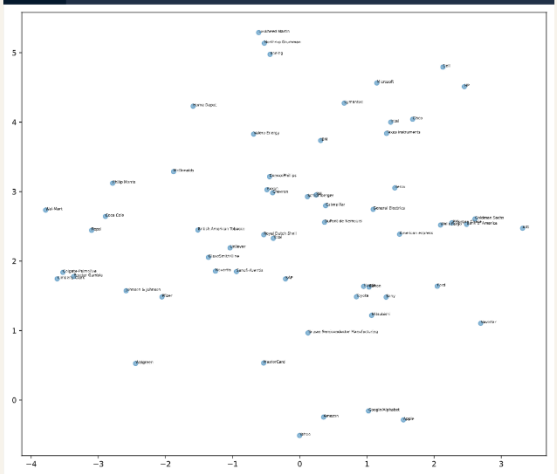
Previous Plot 1/1 Next Plot

script.py Light Mode

```python
# Import TSNE
from sklearn.manifold import TSNE

# Create a TSNE instance: model
model = TSNE(learning_rate=50)

# Apply fit_transform to normalized_movements: tsne_features
tsne_features = model.fit_transform(normalized_movements)

# Select the 0th feature: xs
xs = tsne_features[:,0]

# Select the 1th feature: ys
ys = tsne_features[:,1]

# Scatter plot
plt.scatter(xs, ys, alpha=0.5)

# Annotate the points
for x, y, company in zip(xs, ys, companies):
    plt.annotate(company, (x, y), fontsize=5, alpha=0.75)
plt.show()

```

Run Code Submit Answer

Plots

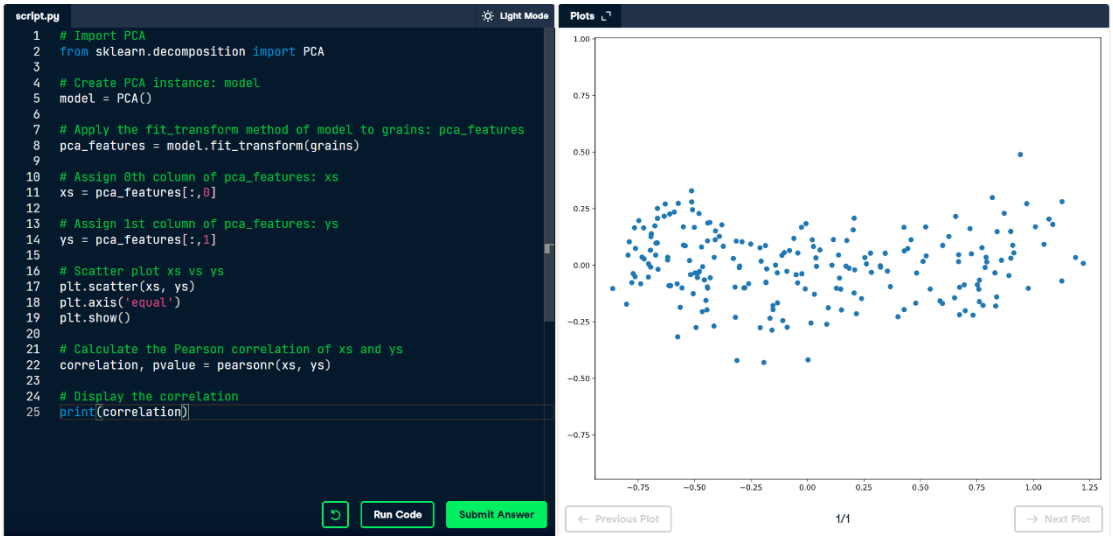


Previous Plot 1/1 Next Plot

PCA 降維 Decorrelating Data and Dimension Reduction

先用皮爾森(Pearson Correlation)做 scatter plot

接著 PCA 降維(下圖)



內在維度 Intrinsic Dimension

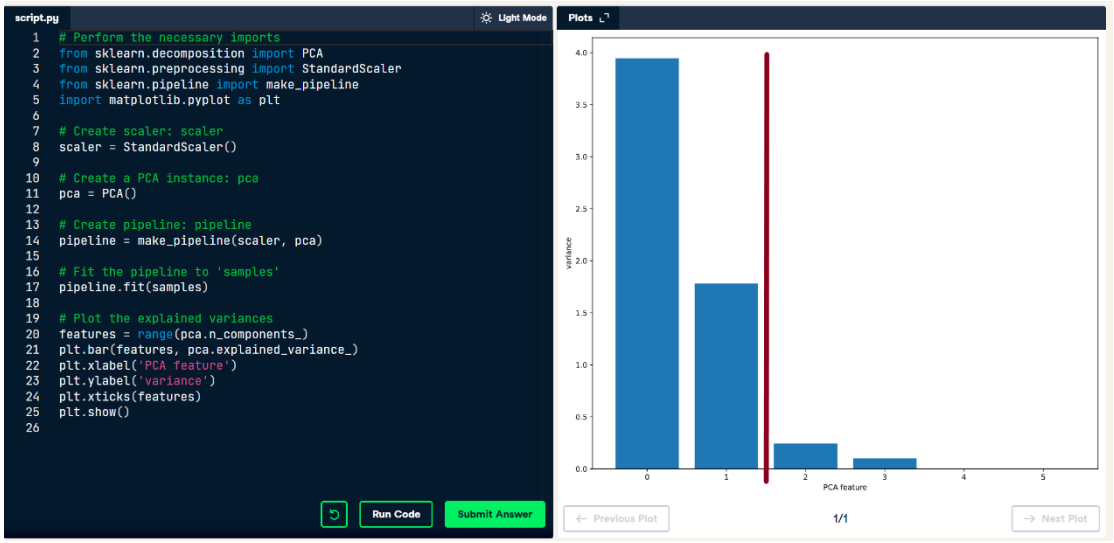
資料先 PCA(箭頭為 Pearson 的值>0)

再做 StandardScaler 跟 PCA 的 pipeline

可看出 X=2 (下圖紅線右邊)PCA 特徵沒有顯著差異

表示內在維度為 2

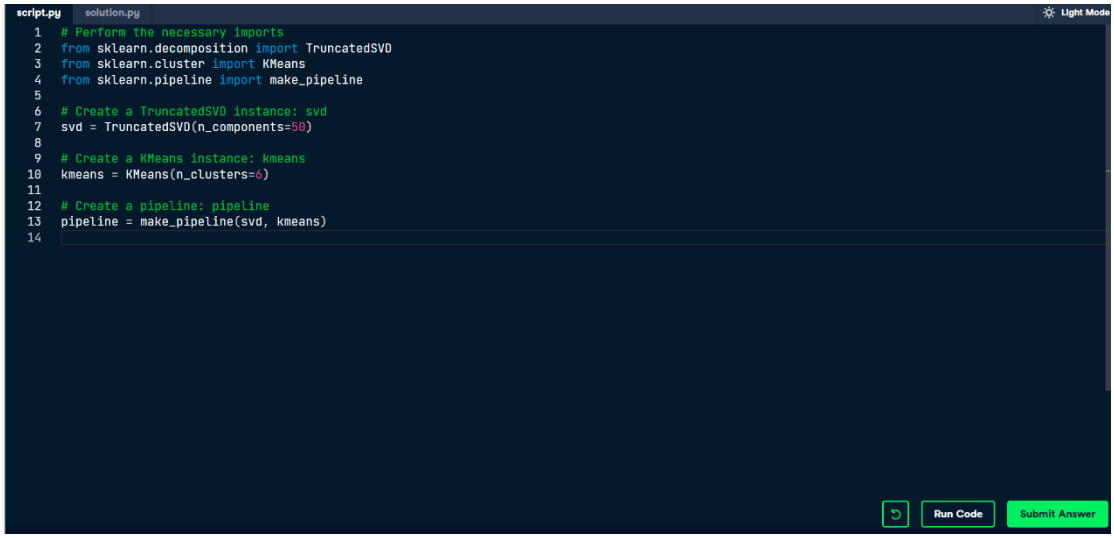
X=2,3 沒有顯著差異，可用 PCA 降維至 2 (PCA(n_components=2))

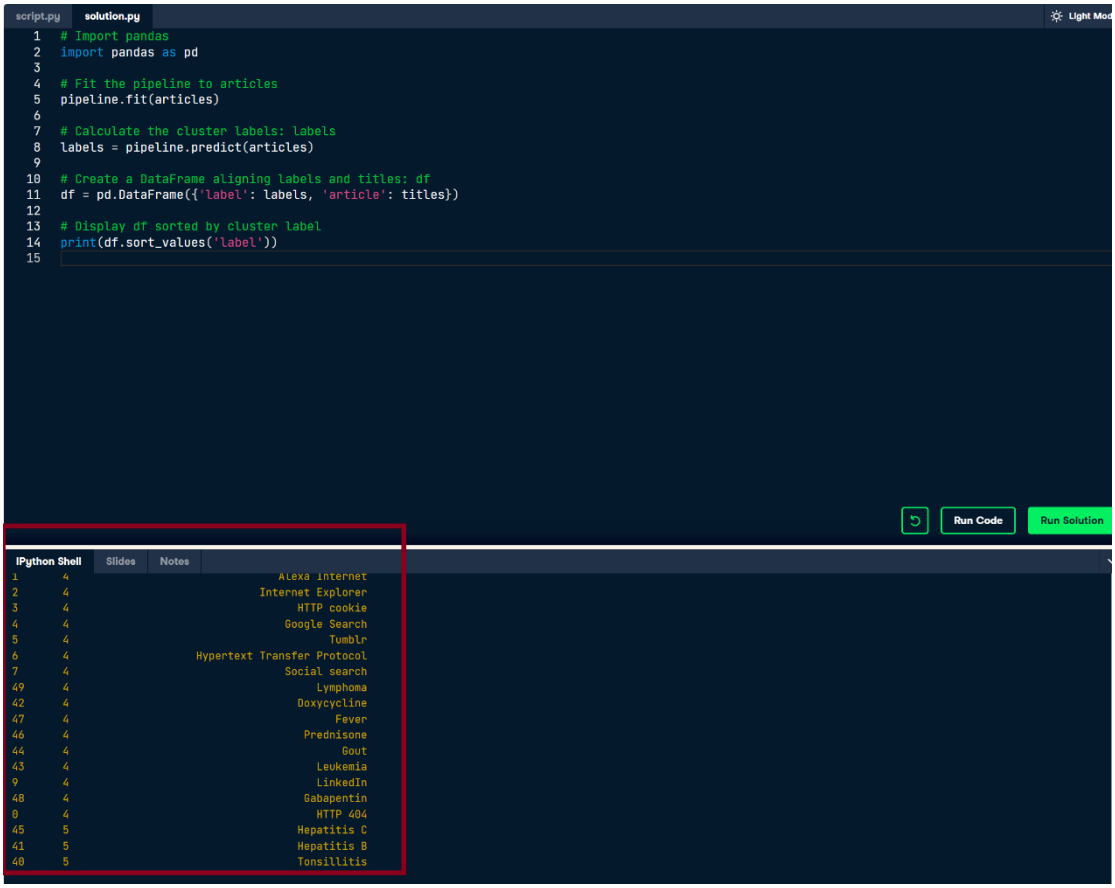


PCA 降維，做 Pipeline (TruncatedSVD & KMeans)

再帶入得到結果(下圖)

PCA 維度降到 50，總共 6 組 (資料為維基百科熱門標題文章)



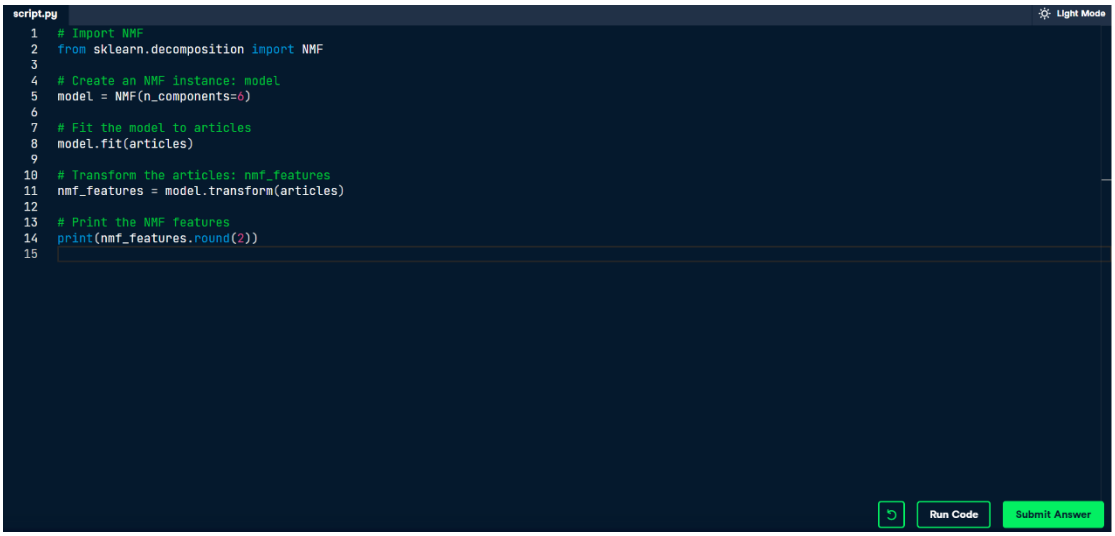


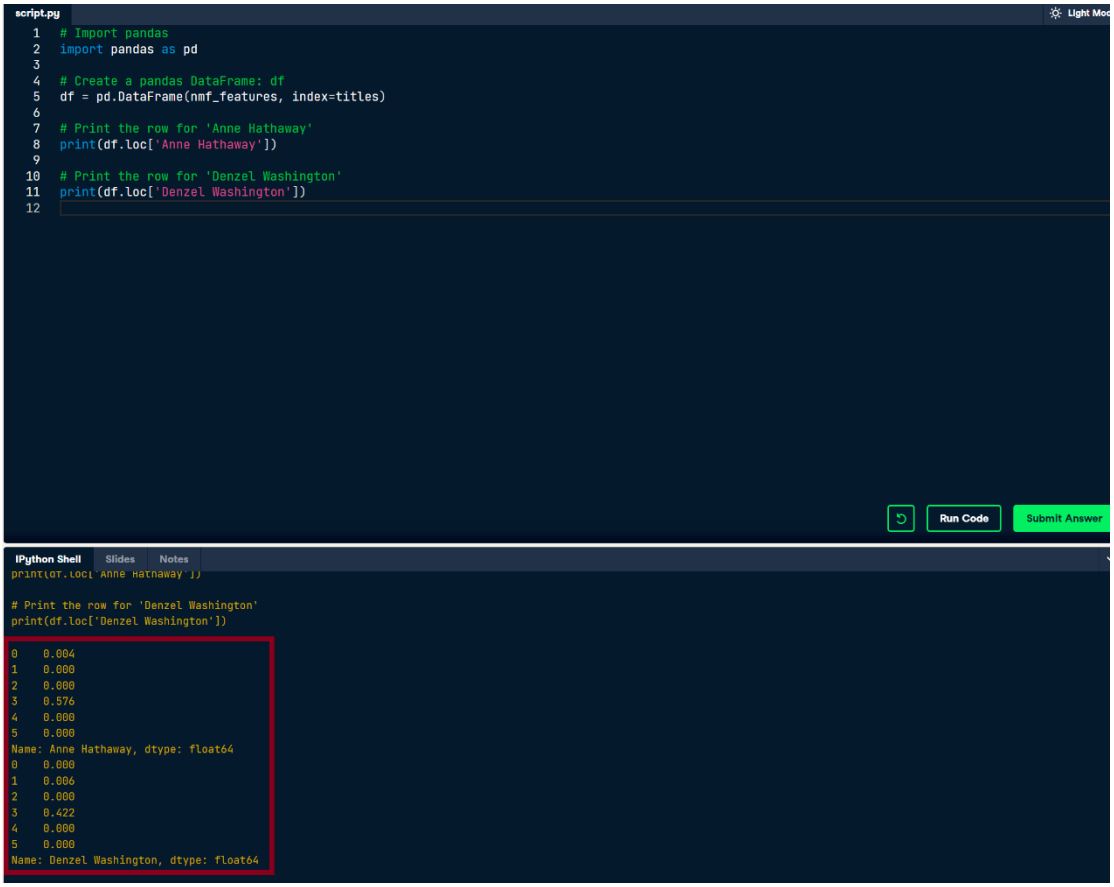
非負矩陣降維 NMF

資料為調查安海瑟薇跟丹佐華盛頓的文章特性

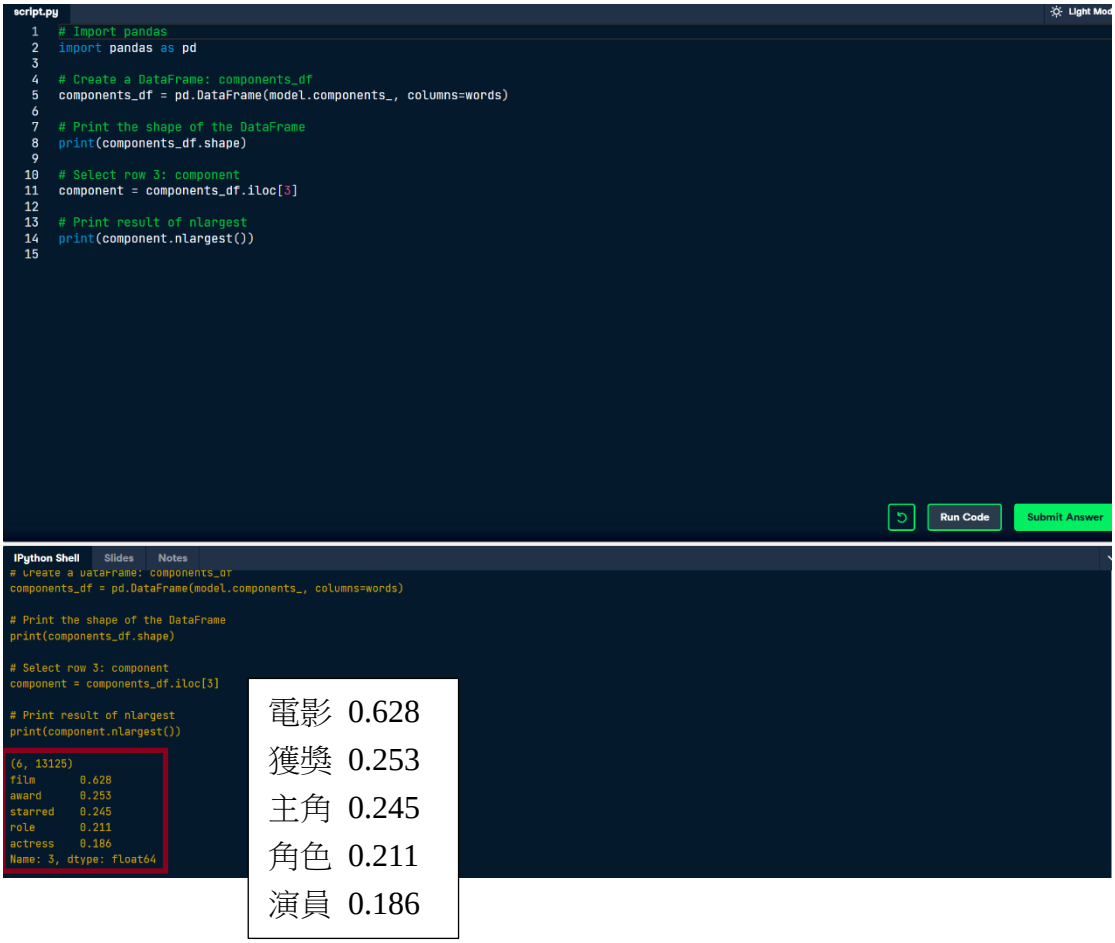
下圖紅框可看出 3 值最高

也就是 3 為主要重建(construct)文章的部分





用 iloc 把 3 的資料提取出來
發現安海瑟威跟丹佐華盛頓的文章
主要都具有以下特徵 (紅框)



電影 0.628
獲獎 0.253
主角 0.245
角色 0.211
演員 0.186

利用 NMF 建置推薦系統

(資料為先前使用的熱門維基百科文章)

資料標準化後，利用.dot (餘弦定理)找出跟西羅有相關的文章(紅框)

下兩圖概念相同

根據喜愛的音樂家，利用.dot，推薦有相關聯的音樂家