雲林科技大學資訊管理系

機器學習

Department of Information Management National

Yunlin University of Science and data mining

專案作業四

M11123018 宋佩庭

M11123054 戴嘉霈

M11123063 許浴宸

指導老師：許中川

Advisor：Chung-Chain Hsu， Ph. D.

中 華 民 國 1 1 2 年 6 月

June 2023

摘要

在現今的環境下多維度資料無所不在，為了能更深層的分析與應用資料，本研究將針對多維度資料進行探討，而在本次研究中將分為兩個議題，在議題一中我們將利用多維標度將7個不同座標進行位置的標記，並將其與地圖中的實際位置進行比較分析，而在議題二中則針對我們現有的資料集進行降維並利用1-of-K 編碼和屬性值相似度兩種方法進行相似度的運算與比較最終將計算結果進行可視化，而在對兩個議題的研究下，議題一成功利用 Google 地圖 API 和 folium 庫，在 Google Map 中標註了各高鐵站的名稱並進行可視化。而議題二中除了在使用 Dash 互動式圈選資料群集下能更好的觀察每個資料群的詳細資料之外，研究顯示相似度高屬性在使用屬性質相似度時其資料的分布情況會比 1-of-K 編碼的分布情形更加的集中，也就是說當飲品為同類型時利用屬性質相似度的分布情況會使相同類型的飲品間距離較近。

關鍵字：Multidimensional Scaling、t-SNE、Attribute Similarity、Dash。

一、緒論

1.1 動機

在現今的資訊環境下，不論是電商平台或是影音串流平台都會透過進行巨量資料的分析來獲得有用的資訊，比方說蒐集用戶的興趣、愛好等資料即可很容易透過二維圖表找出較大的群聚（Cluster）組合，進而歸納出適合不同族群的電影推薦清單。但當資料維度超過三維時，就很難利用可視化圖形表示出資料之間的分佈結構及關連。若再加上巨量資料在超高維空間中分佈超稀疏的問題，難以用傳統方式聚類，因此這樣的困境常被稱為「維度災難（Curse of Dimensionality 或稱維度詛咒）」，為了讓人們容易理解高維資料的分佈情況，最常用的方式就是將資料進行降維再進行觀察，如此一來就能夠輕易地針對資料進行分析與運用。

1.2 目的

現今的資料種類多樣，若是想針對多維度的資料進行分析就需要透過降維技術來達成，因此本研究希望透過資料降維來減少資料集的特徵數量，同時最大限度地保留資料集的有用資訊，如此一來就能簡化資料集、提高計算效率、降低過度擬合的風險，同時保持重要資訊，以便更好地理解和分析資料、改善模型性能。

二、方法

本研究使用 Anaconda 整合開發環境內進行程式的撰寫，目的是為了針對現有得資料進行資料的維度降低，並針對降維所獲得的資料圖形可視化，而本研究分為兩個議題。

(1) 議題一： 在此議題下本研究將找出台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱、台東鹿野 7 個不同景點間的距離，並利用多維標度(Multidimensional Scaling, MDS)畫在 2D 平面上。

(2) 議題二： 在議題二中本研究將針對 Drink Dataset 中的資料進行資料降維，並利用相關欄位進行品項間的相似度計算，最終再利用圖形可視化將各個飲品間的相似程度進行呈現。此外，本研究利用 Plotly 和 Dash 的技術，建立了互動式圈選資料群集的功能，可以在該圖形上即時使用 Linked Brushes 技術圈選資料群，取得該群的資料點詳細資料。

三、實驗

3.1實驗資料

(1) 議題一資料

在此議題中，資料主要包含了臺灣的五個高鐵站以及兩個東部地標，分別為：台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱、台東鹿野。

(2) 議題二資料

在此議題中，本研究將使用 Drink 資料集，此資料集中共有四個特徵欄位(Drink, Rank, Amount, Quantity)及一個類別欄位(Class)：Drink 為名目型欄位，Rank、Amount、Quantity 為數值型欄位。針對每一個類別(Class)，依照 Amount 的常態分配及亂數分配 Quantity，隨機產生 Count 數量的資料筆數，資料集詳細資料如表 1所示。

表 1 Drink 資料集

Table 1. Drink Dataset					
Class	**Drink**	**Rank**	**Amount($N(\mu,\sigma)$)**	**Quantity**	**Count**
A	7Up	7	(100, 200)	Random(500, 1000)	300
B	Sprite	6	(200, 10)	Random(500, 1000)	150
C	Pepsi	5	(200, 10)	Random(500, 1000)	150
D	Coke	4	(400, 100)	Random(500, 1000)	300
E	Cappuccino	3	(800, 10)	Random(1, 500)	150
F	Espresso	2	(800, 10)	Random(1, 500)	150
G	Latte	1	(900, 400)	Random(1, 500)	300

$N(\mu,\sigma)$：Normal Distribution

3.2 前置處理

3.2.1 Drink 資料集

(1) 定義資料特徵

本研究將定義資料特徵，包括飲品種類（drinks）、等級（ranks）、金額平均值（amounts_mean）、金額標準差（amounts_std）、數量最小值（quantities_min）、數量最大值（quantities_max）以及每個特徵對應的資料點數量（counts），並根據定義的特徵，使用對應的均值、標準差、最小值和最大值來生成隨機數值，並重複生成資料以達到指定的數量。

(2) 特徵欄位標準化

本研究針對數值行欄位（Rank、Amount、Quantity）進行標準化，透過計算平均值和標準差使其數值縮小到一定的範圍。

(3) 可視化數據

本研究使用 t-SNE 算法將未正規化和正規化後的特徵欄位降低維度至 2 維空間，以便查看原始數據的分布情況。如下圖所示，圖 1 展示了未正規化的特徵空間的 t-SNE 結果，而圖 2 展示了正規化後的特徵空間的 t-SNE 結果。

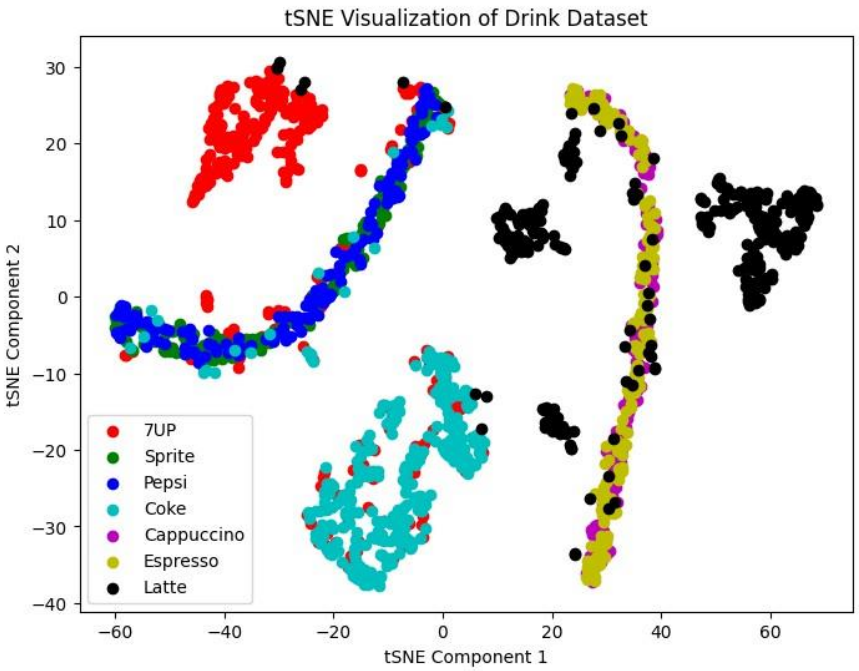

圖 1 未正規化的特徵空間的 t-SNE 結果

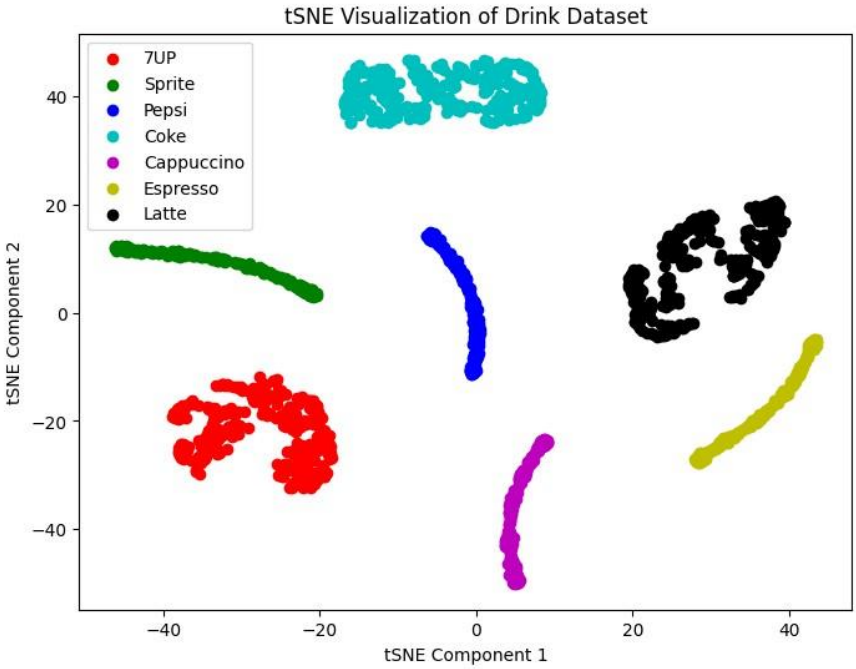

圖 2 正規化後的特徵空間的 t-SNE 結果

3.3實驗設計

3.3.1 議題一

MDS（Multidimensional Scaling）是一種將高維數據降維到較低維度的空間中的方法(Kruskal & Wish, 1978)。目標是將數據的距離保持在降維後的空間中。通過計算每對物體之間的距離（如歐氏距離）來構建距離矩陣。然後依據矩陣構建一個低維坐標，使得在此坐標下的物體間的距離盡可能接近原始的數據，並進行可視化分析。本研究將使用 MDS，將台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱、台東鹿野間的距離畫在二維數據上，並在 Google Map 上標記這些地點位置。

(1) **獲取距離資料**

我們首先使用 Google Maps API，找到台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱和台東鹿野的座標，並利用 Google Maps Distance Matrix API 獲取各站之間的距離資料。

(2) **MDS 降維**

將獲取的距離資料轉換為陣列形式。使用 scikit-learn 的 MDS 演算法，設定 n_components 為 2（二維平面），並使用 precomputed 作為距離矩陣的類型，將計算好的距離矩陣作為輸入，最後使用 fit_transform() 方法，將距離陣列轉換為二維平面上的座標。

(3) **Google Map 標記位置**

本研究使用個人的 Google Maps API 金鑰。定義要標記的位置，以高鐵站名稱為鍵，對應的地址為值。創建一個 folium 地圖，設定初始經緯度和縮放級別。使用 geocode() 函數，獲取台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱和台東鹿野的經緯度座標。再使用 folium.Marker() 函數，將位置的經緯度座標加入地圖中。最後再儲存地圖成網頁的形式以及在程式上視覺化地圖。

3.3.2 議題二

t-SNE（t-distributed stochastic neighbor embedding，t-隨機鄰近嵌入法）是一種非線性的機器學習降維方法，由 Laurens van der Maaten 和 Geoffrey Hinton 於 2008 年提出 (Maaten & Hinton, 2008)，由於 t-SNE 降維時保持局部結構的能力十分傑出，因此成為近年來學術論文與模型比賽中資料視覺化的常客。本研究將

使用 Drink 資料集，透過 tSNE 方法處理名目型屬性的數據，針對 1-of-K 編碼和屬性值的相似度兩種方法比較兩者之間的差異。

(1) **1-of-K**

1-of-K 的優點是它將名目型屬性轉換成了數值型變數，使 tSNE 能夠處理這些變數並計算它們之間的距離，它可以捕捉到不同類別之間的差異，並在降維過程中保持一定的分離性。在一般實務上使用 t-SNE 有以下參數可以設置，分別為 n_components(嵌入空間的維度)、perplexity(困惑度)、random_state(隨機種子)、verbose(運算資訊)，perplexity 表示 t-SNE 運算過程中考慮的鄰近點有多少，verbose 表示是否顯示運算訊息，包括鄰近點數量、時間、KL 散度、誤差等。本研究使用 t-SNE 參數設定如表 2所示：

表 2 1-of-K 之 t-SNE 參數設定

參數	值
n_components	2
perplexity	30
random_state	42
verbose	1

(2) **屬性值相似度**

考慮屬性值的相似度方法可以地利用名目型屬性的相似度信息，透過建立相似度矩陣，將這些相似度作為 tSNE 的輸入，保留了原始屬性值之間的關聯性。此方法能夠更好地捕捉到不同屬性值之間的相似性和差異性。本研究使用 t-SNE 參數設定如表 3所示：

表 3 屬性值相似度之 t-SNE 參數設定

參數	值
n_components	2
perplexity	20
random_state	42
verbose	1

3.4實驗結果

3.4.1 議題一

圖 3為使用 MDS 將台北高鐵站、苗栗高鐵站、雲林高鐵站、台南高鐵站、高雄高鐵站、花蓮豐濱、台東鹿野間的距離畫在二維數據上的結果，x 軸為經度距離，單位為公尺；y 軸為緯度距離，單位為公尺。圖 4為以苗栗高鐵站為例，利用 Google Maps API 在 Google Map 上標記這些地點位置，並能在地圖內使用滑鼠進行移動、查看的可視化結果，。

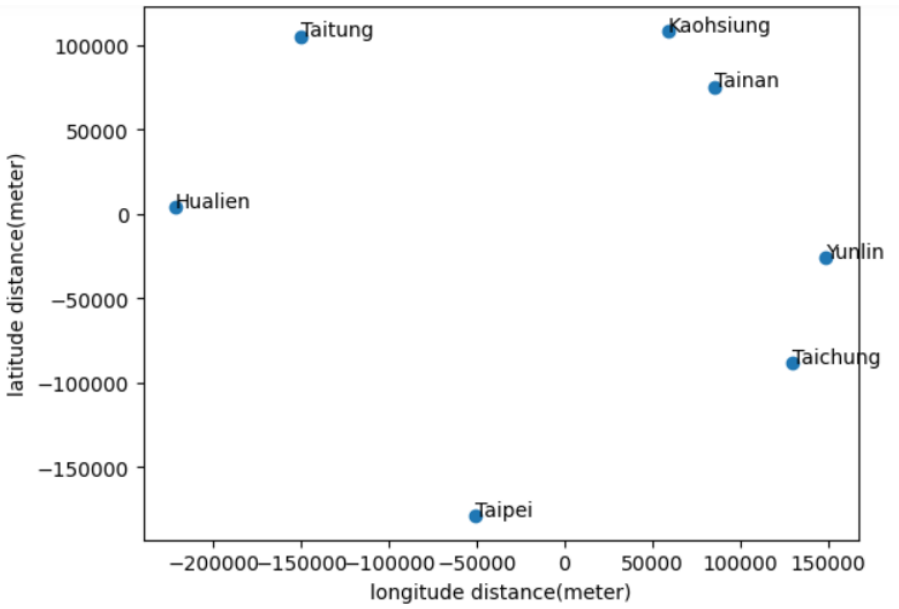

圖 3 MDS 二維化結果

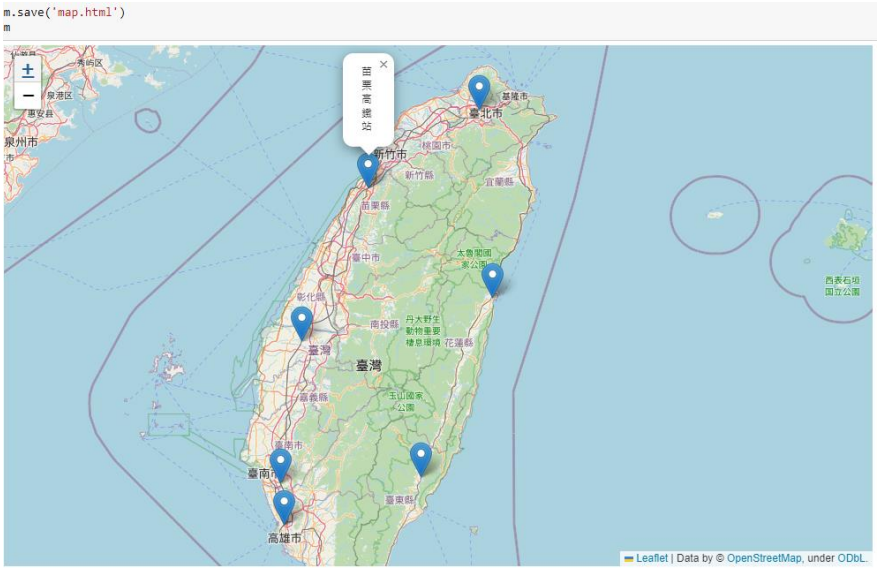

圖 4 Google Map 可視化結果

3.4.2 議題二

實驗結果顯示，1-of-K 使用了91個最近鄰居值，索引了1500個樣本並計算了最近鄰的時間為0.015秒，而計算條件概率的時間為0.002秒，平均標準差為0.00，且250次迭代後的 KL 散度為11.448116，1000次迭代後的 KL 散度為-7.212219，如圖 5所示。而1-of-K 結合 Plotly 和 Dash 建立的互動式圈選資料群集的功能如圖 6所示；屬性值相似度使用了61個最近鄰居值，索引了1500個樣本並計算了最近鄰的時間為0.026秒，而計算條件概率的時間為0.004秒，平均標準差為0.00，且250次迭代後的 KL 散度為20.515785，1000次迭代後的 KL 散度為-5.099223，如圖 7所示。而屬性值相似度結合 Plotly 和 Dash 建立的互動式圈選資料群集的功能如圖 8所示。

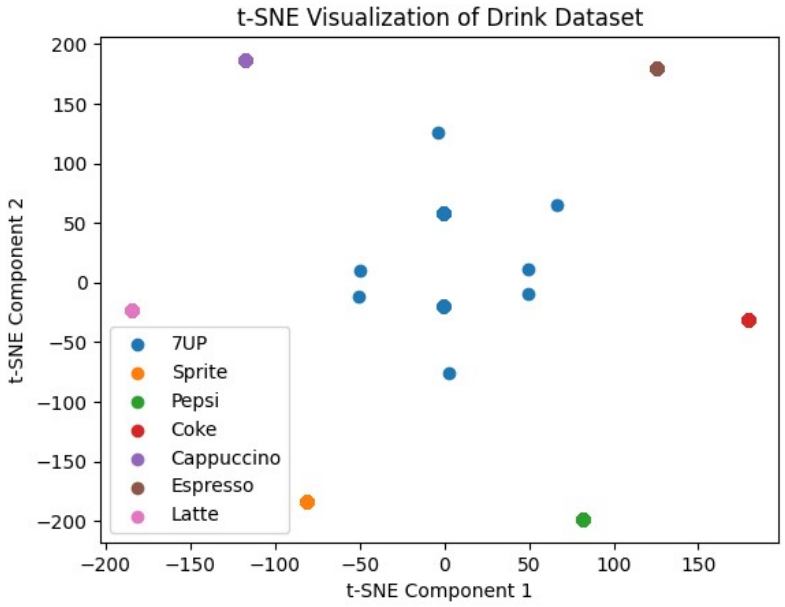

圖 5 1-of-K 結果

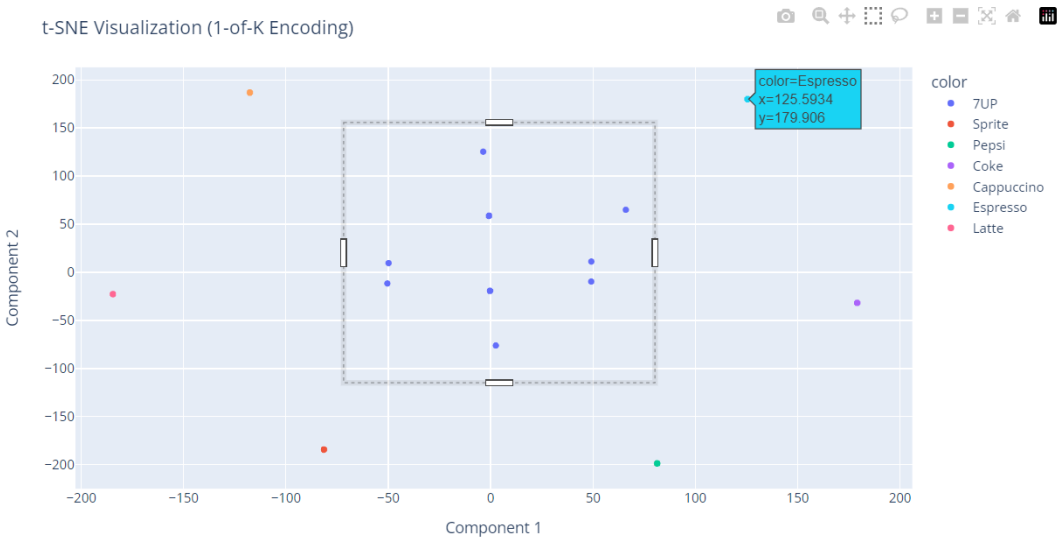

圖 6 1-of-K 結合 Dash 互動式圈選資料群集功能結果

本研究使用 t-SNE 方法來將數據進行降維，而1-of-K 會將每個屬性值都轉換為一個維度，並將其表示為1或0，並將這些二進制編碼的屬性值作為特徵，顯示到二維平面上，目標是保持原始空間中樣本之間的局部關係，在降維後的空間中仍然讓相似的樣本保持靠近，如圖 5、圖 6所示，在降維後的二維平面上，具有相似屬性值的樣本會聚集在一起，而具有不同屬性值的樣本會分散開來。

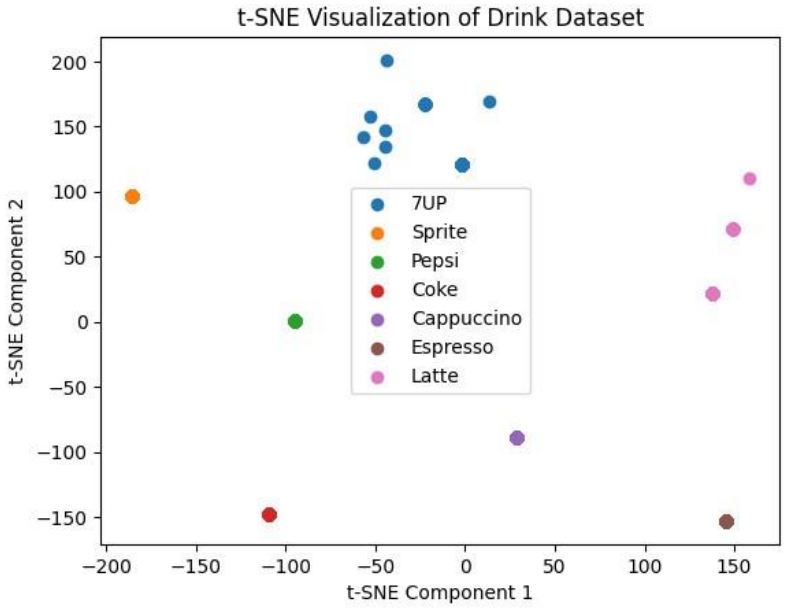

圖 7 屬性值相似度結果

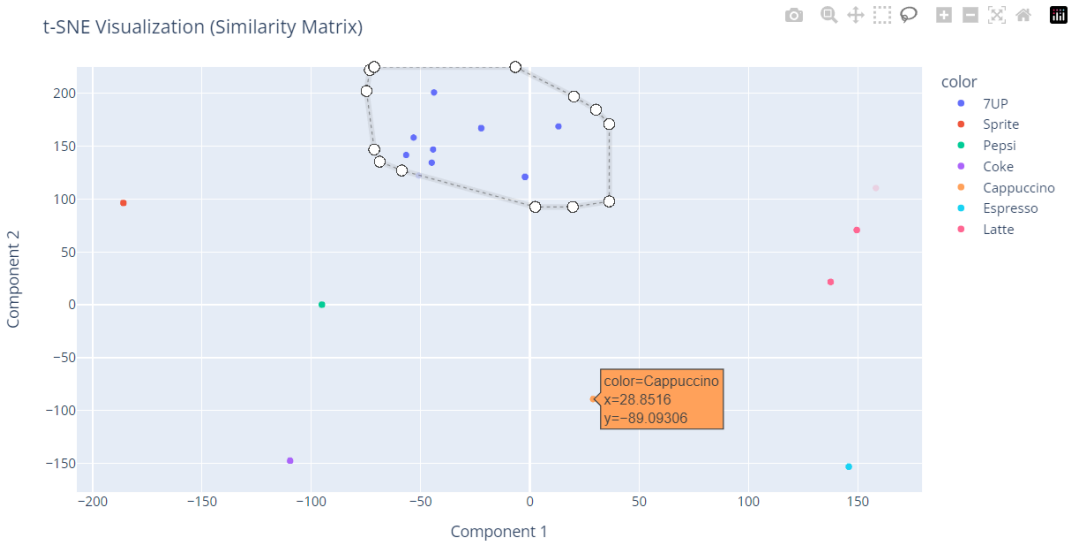

圖 8 屬性值相似度結合 Dash 互動式圈選資料群集功能結果

本研究使用 t-SNE 方法來將數據進行降維，而屬性值相似度方法使用了屬性值之間的相似度矩陣作為輸入，在這種方法中，相似度矩陣使用不同的方式來表示屬性值之間的相似度或不相似度，然後，t-SNE利用這個相似度矩陣來進行降維，保持相似的屬性值在嵌入空間中靠近，不相似的屬性值則遠離，如圖

7、圖 8中的右下部分，Cappuccino、Espresso、Latte 都為同類別的飲料，故位置和方向較相近。

四、結論

在議題一中，本研究使用 Google Maps Distance Matrix API 獲取了台灣高鐵站之間的距離資料，利用 MDS 模型進行了距離資料的降維，將其轉換為二維平面上的座標。最後成功利用 Google 地圖 API 和 folium 庫，在 Google Map 中標註了各高鐵站的名稱並進行可視化。在議題二中首先將 Drink 資料集中進行前處理，其中包含了資料特徵的定義、標準化等，最後利用兩種不同的 t-SNE 方法進行處理與比較，第一種方法是使用 1-of-K 編碼此方法能將名目型屬性轉換成了數值型變數，並將數據點的類別訊息表示為向量。這樣做可以使降維過程中保持一定的分離性，如此一來就能在視覺化上更好地展示不同類別之間的關係和分佈。第二種方法是屬性值相似度此方法透過建立相似度矩陣已呈現出原始屬性值之間的關聯性。最後將這兩種方法加入互動式圈選資料群集功能以提供更精準、更人性化的資料呈現。研究顯示第二種方法相較於第一種方法此方法能夠更好地捕捉到不同屬性值之間的相似性和差異性。相似度高屬性在使用屬性質相似度時其資料的分布情況會比 1-of-K 編碼的分布情形更加的集中，也就是說當飲品為同類型時利用屬性質相似度的分布情況會使相同類型的飲品間距離較近。

參考文獻

Admin. (2023). *[實用技巧]Python 和 Google Maps API：完美組合，輕鬆獲取您喜愛的地點*. GeoLab. https://www.spatialgeolab.com/python-google-map-api/

Csdn_inside. (2019). MDS 降维 详细推导 及 Python 实现. CSDN. https://blog.csdn.net/csdn_inside/article/details/86004733

Kruskal, J. B., & Wish, M. (1978). *Multidimensional scaling* (Vol. 11). Sage.

Omnixri. (2020). *如何應用高維資料可視化一眼看穿你的資料集*. AI HUB. https://aihub.org.tw/platform/specialist/article/3e34f5fc-ddcb-11ea-a936-0242ac120002

Star. (2023). *Plotly Express in Python*. plotly. https://plotly.com/python/plotly-express/

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, *9*(11).