機器學習 專案作業四

組員

M11221004 侯郡凌

M11223040 邱琳恩

中華民國一一三年六月二十四日

摘要

本研究旨在透過多維尺度分析（MDS）以及t-Distributed Stochastic Neighbor（t-SNE）方法實現資料集的降維任務，並搭配Dash技術進行互動式操作，以處理兩個不同的資料集。第一個資料集包含各火車站的經緯度資料，這些資料通過Google Map獲取，此資料集本研究使用MDS方法進行實作。第二個資料集為Drink dataset，則是使用t-SNE方法進行操作，該資料集包含名目資料，因此進行了兩種不同的處理：One-hot encoding和屬性相似度處理（Word2Vec）。結果顯示，MDS以及t-SNE方法皆可實現降維任務，使用獨熱編碼時，數據點呈現較為分散的分布；而採用Word2Vec編碼則更有效地保留了資料的相似性特徵，即相似度高的數據點會聚集在一起。未來的研究中，期望加入更多資料集的應用，以及不同的降維技術，並增加Dash技術的互動功能，提供更多自定義選項和分析工具，以提升數據探索過程中的體驗和效率。

關鍵字：多維尺度分析 (MDS)、t-Distributed Stochastic Neighbor（t-SNE）、One-hot encoding、Word2Vec、Dash

一、 緒論

1.1 動機

在現代資料科學領域，如何有效地將高維度數據進行降維處理，並在低維度空間中保留其結構特徵，是一個重要的研究課題。面對高維度資料集時，儲存和運算的需求增加，使得資料的可視化、分析和解釋變得困難。有效的降維技術能在保留資料主要特徵的同時，減少維度，提升分析的效率和準確性。

MDS和t-SNE是兩種常用的降維技術，分別在距離和相似度維度上有廣泛應用。本研究旨在透過兩個不同的資料集，並探討如何在降維過程中有效處理名目型（Categorical）和數值型（Numerical）屬性，並在2D平面上可視化數據，以便更直觀地理解數據內在結構。

本研究的主要動機包括：

1. 探討不同降維技術的效能：評估MDS和t-SNE在資料集維度縮減中的效果，了解其在高維資料處理中的表現。
2. 減少高維度資料的複雜性：透過降維技術減少資料維度，提升分析效率和準確性，並使資料更易於理解和解釋。
3. 提升資料可視化：利用圖像呈現資料點分布，提升資料的易讀性，幫助使用者更直觀地理解資料結構。

1.2 目的

本研究的主要目的是透過MDS和t-SNE，探討如何有效地將高維度資料進行降維處理，並在2D平面上進行可視化展示，具體目的包括：

1. 探討地理位置數據的可視化方法：使用MDS方法計算並比較台北、新竹、台中、斗六、高雄、花蓮玉里、台東知本火車站的實際距離，並在2D平面上進行可視化展示。通過Google地圖標記車站位置，驗證MDS在地理數據可視化中的應用效果。
2. 探討名目型和數值型屬性在降維過程中的處理方法：使用Drink Dataset隨機生成數據，並對數值型屬性進行常態分配及亂數分配處理。採用t-SNE方法，在2D平面上展示包含名目型和數值型屬性的數據，探討其降維效果。
3. 比較不同名目型屬性處理方法的效果：比較One-hot encoding和屬性值相似度兩種名目型屬性處理方法在t-SNE降維可視化中的效果。
4. 提供數據可視化和降維技術的應用參考：透過案例研究展示MDS和t-SNE在地理位置數據和混合屬性數據中的應用效果。
5. 資料集縮減：使用MDS和t-SNE方法對高維資料進行降維，保留數據的主要結構和特徵，以便進行後續分析和可視化。
6. 比較與評估：比較MDS和t-SNE兩種降維方法在不同資料集上的表現，包括維度縮減後的數據分佈。

二、 方法

本研究將透過兩種方法分別使用採用火車站經緯度資料及Drink Dataset來探討高維數據的降維處理和可視化方法，具體研究方法如下：

**針對MDS方法：**

首先，使用Google地圖API獲取台北火車站、新竹火車站、台中火車站、斗六火車站、高雄火車站、花蓮玉里站、台東知本站的經緯度資訊。接著，利用Haversine公式計算這些火車站之間的實際地理距離，形成距離矩陣。然後，採用多維尺度分析（MDS）方法將距離矩陣轉換為2D平面坐標。最後，使用Dash框架繪製互動式圖表，展示MDS降維結果。圖表中標記各火車站的位置，並添加互動功能，使使用者可以放大、縮小和移動圖表，便於詳細觀察和分析。

**針對t-SNE方法：**

首先，根據Drink Dataset，以DataFrame形式隨機生成指定數量的數據，包含四個特徵欄位（Drink, Rank, Amount, Quantity）和一個類別欄位（Class）。對名目型欄位Drink進行One-hot encoding編碼和屬性值相似度處理。在進行距離計算之前，對所有特徵進行正規化，以確保不同特徵的值處於相同的尺度上。接著，利用t-SNE方法將處理後的高維數據降至2D平面，分別展示One-hot encoding和屬性值相似度處理後的數據在t-SNE降維後的可視化效果。最後，通過比較兩種處理方法在t-SNE降維可視化中的表現，分析它們對最終結果的影響及其優劣。所有可視化結果將透過Dash框架呈現，提供互動式展示。

三、 實驗

3.1 資料集

本研究使用兩個不同的資料集進行降維處理：火車站經緯度資料集和Drink Dataset。火車站經緯度資料集包含了台北火車站、新竹火車站、台中火車站、斗六火車站、高雄火車站、花蓮玉里、台東知本的具體經緯度訊息，如表1所示。另一方面，Drink Dataset 包含7種不同品牌的飲料，其中包含Class、Drink、Rank、Amount、Quantiy等特徵，如表2所示。

表1、火車站經緯度dataset

|  |  |  |
| --- | --- | --- |
| 火車站 | 經度 | 緯度 |
| Taipei | 121.5170416140462 | 25.048144340471694 |
| Hsinchu | 120.97218465041708 | 24.803620504236306 |
| Taichung | 120.68502265782085 | 24.136965842698647 |
| Douliu | 120.54100869869137 | 23.71194174320373 |
| Kaohsiung | 120.30263663949066 | 22.63967061761657 |
| Hualien | 121.31177017259222 | 23.331693127455807 |
| Taitung | 121.06074805573594 | 22.710408169564232 |

表2、Drink dataset

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Class** | **Drink** | **Rank** | **Amount**  **(*N*(*μ*,*σ*))** | | **Quantity** | **Count** |
| A | 7Up | 7 | (100, 200) | Random(500, 1000) | | 100 |
| B | Sprite | 6 | (200, 10) | Random(500, 1000) | | 200 |
| C | Pepsi | 5 | (200, 10) | Random(500, 1000) | | 100 |
| D | Coke | 4 | (400, 100) | Random(500, 1000) | | 400 |
| E | Cappuccino | 3 | (800, 10) | Random(1, 500) | | 400 |
| F | Espresso | 2 | (800, 10) | Random(1, 500) | | 200 |
| G | Latte | 1 | (900, 400) | Random(1, 500) | | 100 |

3.2 前置處理

本研究在MDS、t-SNE方法上進行了不同的前置處理，以下將針對兩種方法進行說明。

**針對MDS方法：**

首先，定義車站的經緯度資訊後，將城市名稱和經緯度分別存儲在cities和coords中，接著透過Haversine方法計算城市之間的地理距離，並建構距離矩陣以供後續的MDS方法以及Dash應用，如表3所示。

**針對t-SNE方法：**

在t-SNE方法中，採用了兩種不同的名目屬性處理方式，分別使用One-hot encoding（獨熱編碼）及Word2Vec處理。

在One-hot encoding處理部分，首先，定義了一個生成單個數據點的函數 generate\_single\_data\_point，用於生成具有特定Class、Drink、Rank、Amount的平均值和標準差、Quantiy的範圍。接著，將Amount及Quantiy這兩列拆分成單獨的特徵列，進行獨立處理。隨後，使用One-hot encoding方法對Drink欄位進行編碼，且將其轉換為二進制(0或1)。最後，將獨熱編碼後的各個特徵合併，編碼結果如表4所示。

在Word2Vec處理部分，首先，設置隨機數種子並定義Drink dataset，接著計算每種飲料的平均值數據，並針對Rank、Amount、Quantity，Count欄位進正規化處理，最後透過Word2Vec轉換飲料名稱為向量，並計算飲料之間的距離矩陣，以利於後續的t-SNE方法以及Dash應用，如表5所示。

表3、火車站經緯度之距離矩陣

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Taipei | Hsinchu | Taichung | Douliu | Kaohsiung | Hualien | Taitung |
| Taipei | 0.00 | 61.30 | 131.69 | 178.46 | 294.91 | 191.99 | 264.05 |
| Hsinchu | 61.30 | 0.00 | 79.62 | 129.02 | 250.09 | 167.26 | 232.93 |
| Taichung | 131.69 | 79.62 | 0.00 | 49.48 | 171.00 | 109.94 | 163.19 |
| Douliu | 178.46 | 129.02 | 49.48 | 0.00 | 121.70 | 89.24 | 123.38 |
| Kaohsiung | 294.91 | 250.09 | 171.00 | 121.70 | 0.00 | 128.81 | 78.18 |
| Hualien | 191.99 | 167.25 | 109.94 | 89.24 | 128.81 | 0.00 | 73.71 |
| Taitung | 264.05 | 232.93 | 163.19 | 123.38 | 78.18 | 73.71 | 0.00 |

表4、應用獨熱編碼於Drink dataset結果

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7Up | Sprite | Pepsi | Coke | Cappuccino | Espresso | Latte |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

表5、應用Word2Vec於Drink Dataset之距離矩陣

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7Up | Sprite | Pepsi | Coke | Cappuccino | Espresso | Latte |
| 7Up | 0.000000 | 1.017510 | 1.071149e+00 | 3.007474 | 4.336312 | 3.974350 | 4.514016 |
| Sprite | 1.017510 | 0.000000 | 9.728307e-01 | 2.003688 | 3.555650 | 3.425595 | 4.097352 |
| Pepsi | 1.071149 | 0.972831 | 4.214685e-08 | 2.547805 | 3.745725 | 3.168252 | 3.644110 |
| Coke | 3.007474 | 2.003688 | 2.547805e+00 | 0.000000 | 2.404297 | 3.007392 | 3.887165 |
| Cappuccino | 4.336312 | 3.555650 | 3.745725e+00 | 2.404297 | 0.000000 | 1.713040 | 2.660979 |
| Espresso | 3.974350 | 3.425595 | 3.168252e+00 | 3.007392 | 1.713040 | 0.000000 | 1.082886 |
| Latte | 4.514016 | 4.097352 | 3.644110e+00 | 3.887165 | 2.660979 | 1.082886 | 0.000000 |

3.3 實驗設計

本研究採用MDS以及t-SNE方法實現資料集維度縮減任務，並搭配Dash技術進行互動式操作。

**針對MDS方法：**

首先，蒐集目標的七個火車站經緯度資訊並進行定義，以cities、coords變數儲存城市名稱和經緯度，接著引入sklearn.manifold.MDS、Dash、plotly.graph\_objs、json等函式庫來進行後續操作，透過Haversine計算出城市之間的地理距離，並構建距離矩陣，並使用MDS將距離矩陣轉換為二維坐標，以便可視化，且為了增加互動式操作，使用Dash技術建立網頁應用，並定義回調函數，呈現在圖表上移動、點擊、選擇時顯示的數據，提升與資料點的互動性。

**針對t-SNE方法：**

首先，分為兩種不同的名目屬性處理方式進行：在One-hot encoding處理部分，首先，使用不同類別的飲料特徵值及數值範圍生成了一組虛擬的資料集，且使用隨機種子確保每次生成資料一致，以便之後的分析。接著，使用One-hot encoding將飲料類別進行編碼，並將編碼後的特徵與其他數值特徵結合。然後，應用t-SNE演算法將多維特徵降維到二維，以便後續的可視化。最後，在Dash應用的設置方面，包含一個散布圖用於顯示t-SNE降維後的結果，以及三個用於顯示懸停、點擊和選擇互動數據的區域。

在Word2Vec處理部分，先針對Drink Dataset進行定義，且為了確保結果相同，設置固定隨機數種子為40。接著進行Amount和Quantity數據計算，並針對Rank、Amount、Quantity，Count欄位進正規化處理，最後透過Word2Vec轉換飲料名稱為向量，並計算飲料之間的距離矩陣，接著使用t-SNE方法，降維到二維空間，並將perplexity超參數值設為2，更好地捕捉每種飲料之間的局部相似性。在Dash應用部分，定義回調函數包含，圖像上移動、點擊、選擇，提升與資料點的互動性和資料易讀性。

3.4 實驗結果

**針對MDS方法：**

如圖1所示，在圖像上進行移動、點擊、選擇，皆可與圖像產生互動，且根據本研究比對發現，資料點的位置與Google Map上實際的車站位置相同，因此在火車站經緯度dataset上，成功實現MDS降緯度任務以及Dash的應用。

**針對t-SNE方法：**

在One-hot encoding的名目資料處理部分，如表6所示為最終結果座標，及圖2所示，最終結果顯示了降維後的數據分布情況，由圖可知，各數據點的分布相對較為鬆散，表示使用獨熱編碼進行降維的效果並不理想，因為它未能很好地捕捉和保留數據點之間的關係。

在Word2Vec名目屬性處理部分，如圖3所示，可以發現相似度高的品項會聚集在附近，成功實現t-SNE降緯度任務，且因為同樣使用了Dash方法，因此在圖像上進行移動、點擊、選擇，皆可與圖像產生互動。

在表7部分以選擇為例，使用Box Select工具進行框選動作，被框選的資料點情報會呈現在下方的Selection Data資訊中，包含資料點的名稱、顏色、(x,y)等。

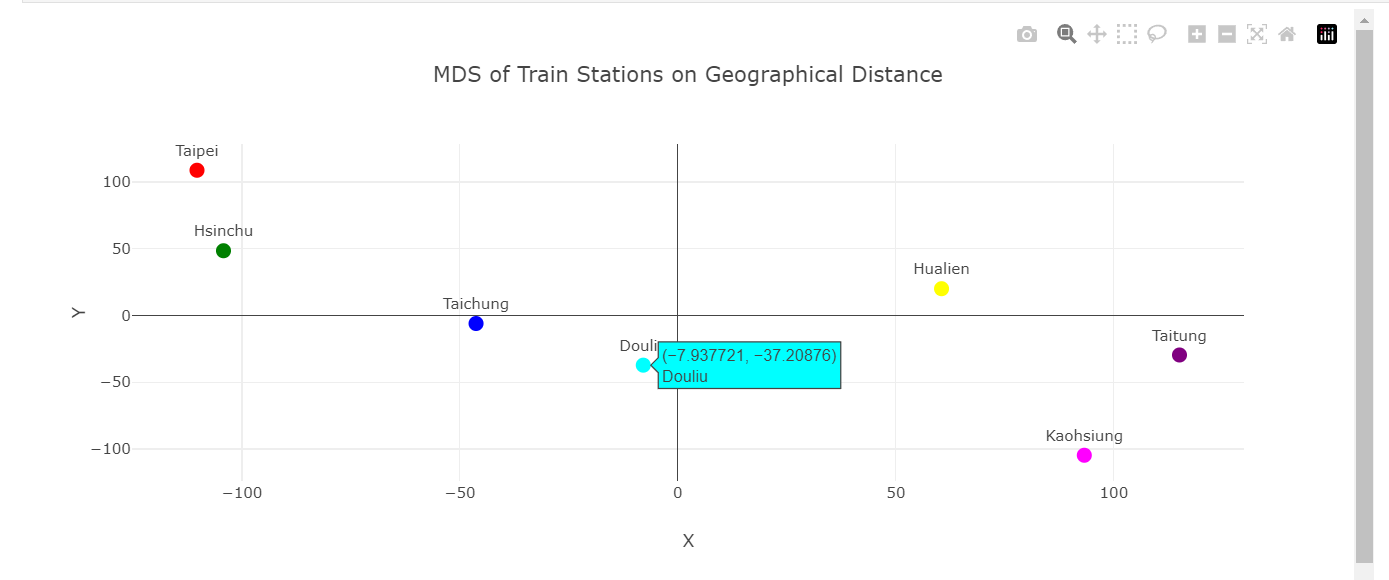


圖1、Dash圖像:火車站經緯度dataset

表6、One-hot encoding編碼：t-SNE降維度結果

| Drink | T-SNE1 | T-SNE2 |
| --- | --- | --- |
| 7Up | -29.594 | 60.112 |
| Sprite | -45.997 | 11.892 |
| Pepsi | 20.369 | 70.019 |
| Coke | 53.929 | 31.704 |
| Cappuccino | -12.437 | -26.423 |
| Espresso | 3.966 | 21.798 |
| Latte | 37.524 | -16.518 |

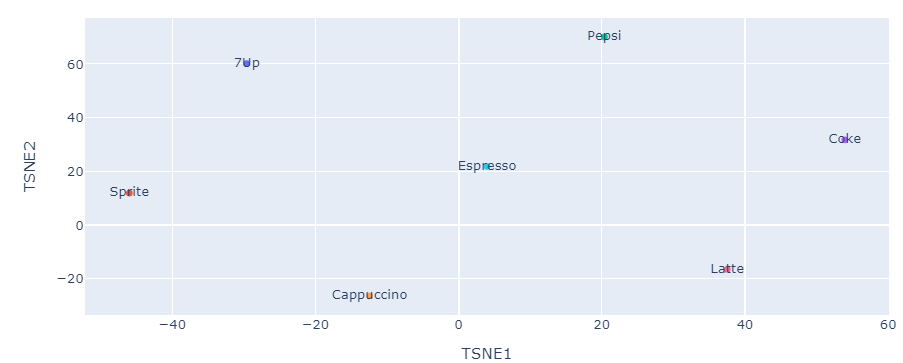


圖2、應用獨熱編碼與t-SNE降維的結果可視化

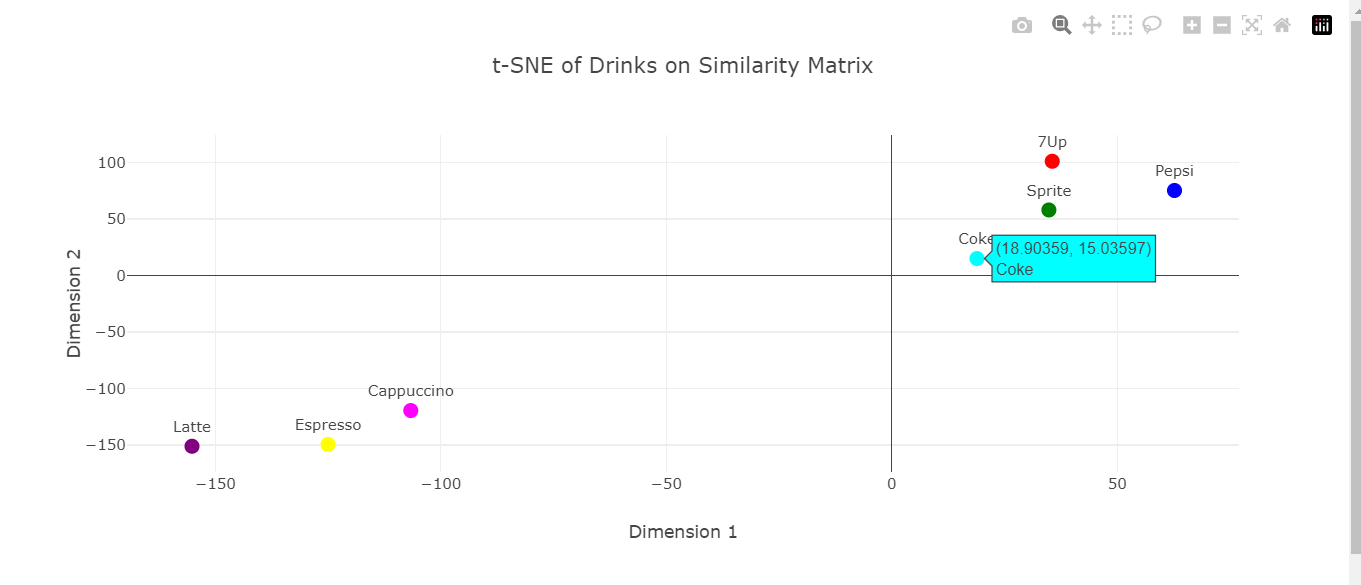


圖3、應用Word2Vec編碼與t-SNE降維的結果可視化

表7、Dash 框選：Drink dataset

|  |
| --- |
| **框選Sprite、Coke:** |
|  |
| **Selection Data 資訊:** |
|  |

四、 結論

本研究透過MDS以及t-SNE方法進行資料集維度縮減任務，皆將火車站經緯度dataset、Drink dataset降至二維，且在t-SNE方法部分，進行了不同名目屬性處理方式的比較，分別使用One-hot encoding以及Word2Vec。

在One-hot encoding部分，從圖2中可以得出結論，獨熱編碼在某些情況下可能並不是最佳的特徵表示方法，尤其是在需要保持數據點之間關係的降維和可視化任務中。因此，使用獨熱編碼處理名目資料，會使各數據點的資料分布更分散。隨著數據量的增加，獨熱編碼的高維度特性可能會導致處理時間顯著延長，進一步降低了其在大規模數據集上的效率和實用性。

在Word2Vec部分則是呈現相似度高的品項聚在附近的現象，能夠有效地捕捉屬性之間的語義相似度，從而使t-SNE在降維後能夠更好地保持這些語義關係。

透過不同的處理方式比較，對t-SNE方法有了更多的認識，發現不同的名目屬性處理方式對結果有顯著影響。且本研究搭配Dash技術，進行互動式操作，如圖像上移動、點擊、選擇，使得使用者能夠更直觀地理解和探索數據。

使用者可以通過互動界面放大、縮小圖像，點擊特定點以查看其具體資訊，或框選多個點進行資訊比較，這樣的互動操作提升了數據分析的可視化效果，在未來的研究中，期望加入更多資料集的應用，以及嘗試不同的降緯度技術，並提升Dash技術的互動功能，加入更多自定義選項和分析工具，提升在數據探索過程中的體驗和效率。

參考文獻

|  |
| --- |
| chwang. (2024, January 13). *Data Visualization資料視覺化- Python -Plotly進階視覺化 — Dash教學(一).* Medium. https://chwang12341.medium.com/data-visualization%E8%B3%87%E6%96%99%E8%A6%96%E8%A6%BA%E5%8C%96-python-plotly%E9%80%B2%E9%9A%8E%E8%A6%96%E8%A6%BA%E5%8C%96-dash%E6%95%99%E5%AD%B8-%E4%B8%80-c087c0008b78 |
| hustqb. (2018, June 9). *T-SNE实践——Sklearn教程.* CSDN博客. https://blog.csdn.net/hustqb/article/details/80628721 |
| lyn5284767. (2018, August 8). 机器学习-降维算法(MDS算法). CSDN博客. https://blog.csdn.net/hustqb/article/details/80628721 |
| Python: *How to Get Data from Linked Brushes in Mlpd3, Bokeh, Plotly?* (n.d.). Stack Overflow. https://stackoverflow.com/questions/44531241/python-how-to-get-data-from-linked-brushes-in-mlpd3-bokeh-plotly |
| Stack Overflow. (n.d.). *Python: How to get data from linked brushes in mlpd3, Bokeh, Plotly?* Retrieved June 12, 2024, from https://stackoverflow.com/questions/44531241/python-how-to-get-data-from-linked-brushes-in-mlpd3-bokeh-plotly |