 國立雲林科技大學資訊管理系

機器學習-作業三

Department of Information Management

National Yunlin University of Science & Technology Assignment

Banana 資料集和Size3 資料集分群演算法分析

Banana Datasets and Size3 Dataset Cluster Analysis

楊欣蓓、陳怡君、鄭皓名、陳郁云

指導老師：許中川 博士

Advisor: Chung-Chian Hsu, Ph.D.

 中華民國112年12月

December 2023

**摘要**

本實驗對Banana資料集和size3資料集進行分群分析，因這兩種資料集的資料分佈密度及資料群聚的形狀都不同，所以可以去觀察分群演算法適合哪些型態的資料集。

對的績效分別為何，使用K-means、階層式分群、DBSCAN三種分群演算法進行分群分析，，

關鍵字：hours-per-week、early stopping、patient、overfitting、learning rate、generalization gap、BatchNormalization

**一、緒論**

1. **研究動機**

在本課程中學習了許多分群演算法，本研究主要是在眾多的群聚分析中進行分析，因其資料集的資料雜亂，所以需要資料前處理才能進行分群分析，而選擇Banana、size3的主要原因是各個資料集的資料分佈不同，使得本研究可以使用不同的分群演算法（例如:K-means、階層式分群、DBSCAN）來進行實驗，通過比較這些分群演算法表現，可以評估它們在處理不同形狀和密度的分群時的績效。

本研究發現 Banana、size3的二維結構使得群集可視化相對容易，並且可以使用不同的分群演算法將數據點劃分到群集中，以及可視化這些群集的分佈，這有助於理解不同算法的群集結果，由於這些資料集包含不同形狀和密度的群集，所以可以嘗試調整不同群集算法的參數，以查看參數對結果的影響如何。 如果這個資料集與某個實際應用相關，例如顧客購買行為、產品銷售模式等，使用 K-means、階層式分群、DBSCAN 可以幫助本研究識別和理解樣本之間的相似性和差異性，並且為業務決策提供有價值的資訊。總上所述，本研究進行 K-means、階層式分群、DBSCAN 分群演算法是為了從數據中挖掘有效資訊，理解潛在的結構和模式，並且比較不同算法在這個情境下的效能表現。

1. **研究目的**

本研究欲以 Banana 資料集和 size3 資料集以各種分群來進行比較 K-means、 階層式分群、DBSCAN 三種分群所花費的時間，將此分群以 SSE、Accuracy、 Entropy 為此衡量指標，並劃分出這三種分群所呈現的結果，其中本研究想以 DBSCAN 分群來進行不同的參數設定，此研究透過各種參數設定來進行比較後， 藉由比較後的參數設定上分析出以 Banana 資料集和 size3 資料集中的試驗以達成 Banana 資料集和 size3 資料集的最佳設定值，也了解每種分群在不同情況下的優缺點及此限制，在碰到資料集中存在雜訊時，該如何使用各種分群 來解決資料集的問題，並透過 K-means、階層式分群、DBSCAN 分群來分析資料 集的 SSE、Accuracy、Entropy 的結果。

**二、實驗方法**

1. **實作說明**

本研究使用K-means、階層式分群、DBSCAN三種分群演算法對Banana、size3進行分群實驗。在資料前處理的部分，對資料刪除了離群值，使分群結果不被離群值影響、對資料作正規化，使不同屬性的數值差距不會甚多；調整了分群演算法的參數，利用SSE、Accuracy、Entropy的結果去調整參數，使分群結果更加完確。此外本次實驗為分群分析，並透過Dendrogram和scatter的資料視覺化，去觀察不同演算法對資料的分群效果，進而去比較各個演算法的分群效果為何，最後記錄各個演算法的執行時長及分群結果並做出研究分析。

1. **操作說明**

本研究執行環境皆採用Python3.10.10，以Visual Studio Code作為開發工具，利用K-means、階層式分群、DBSCAN三種演算法進行分群實驗，並使用Pandas、Numpy、Scikit-learn、Matplotlib等函式庫來讀取資料、分析分群結果及資料視覺化呈現。於資料前處理，利用skewness和kurtosis去檢視資料離群值並刪除離群值、用MinMaxScaler將資料數值之間的差距縮至0至1之間，上述對資料的操作可以使資料在做分群時分得更完整、提高分群績效。

**三、實驗設計**

1. **資料集**

名稱: Banana 資料集

資料筆數:4811筆

**表1** Banana datasets欄位介紹

|  |  |  |
| --- | --- | --- |
| 欄位 | 屬性 | 內容 |
| 0 | x | numeric |
| 1 | y | numeric |
| 2 | class | nominal |

名稱: size3 資料集

原始資料筆數:1000筆

刪除離群值後的資料筆數:820筆

**表2** size3 datasets欄位介紹

|  |  |  |
| --- | --- | --- |
| 欄位 | 屬性 | 內容 |
| 0 | x | numeric |
| 1 | y | numeric |
| 2 | class | nominal |

1. **資料前處理**
2. **Banana資料集**

* 分析資料集：
* 透過Skewness和Kurtosis所繪製出來的分佈圖，可得知此資料集為非常態分佈，因此利用IQR方法去檢視資料集是否有離群值，將資料做好排序後，將第三四分位數減去第一四分位數可得IQR，當有資料數值落在第一四分位數-1.5\*IQR以下或第三四分位數+1.5\*IQR以上範圍，則該筆資料為離群值應刪除，最後從資料筆數前後對比以及IQR箱型圖得知該資料集沒有離群值。
* 從scatter圖看出資料數值皆落在0到1之間，所以無需做正規化。

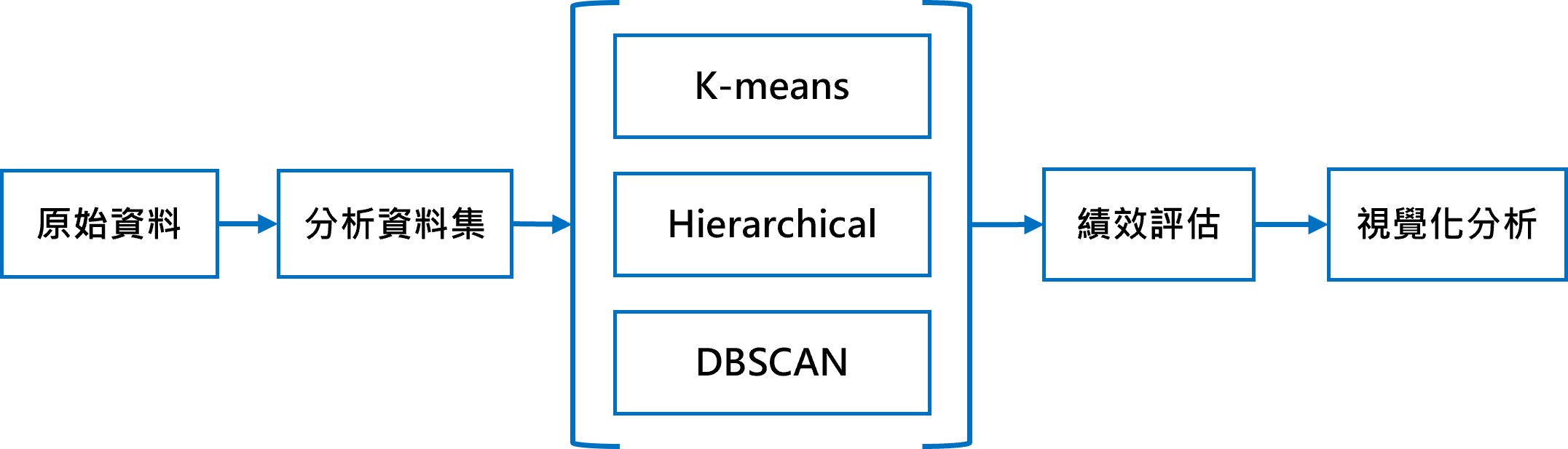
1. **Size3資料集**

* 資料前處理：
  + 透過IQR方法檢查資料前後筆數及IQR箱型圖確認資料是否存在離群值，並一一將離群值做刪除，使資料分群結果不受離群值影響。
  + 從scatter圖看出若干筆資料數值未落在0至1之間，為了不讓數值之間的差距影響分群結果，使用正規化將資料數值落在0到1之間。

**表4** 部分資料處理後的Size3資料集

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 資料  特徵 | No.0 | No.1 | No.2 | No.3 | No.4 | No.5 |
| **X** | 0.60260 | 0.78808 | 0.56988 | 0.59167 | 0.78194 | 0.70639 |
| **Y** | 0.87058 | 0.48759 | 0.68918 | 0.76108 | 0.50162 | 0.62674 |

1. **實驗設計**
2. **Banana資料集**

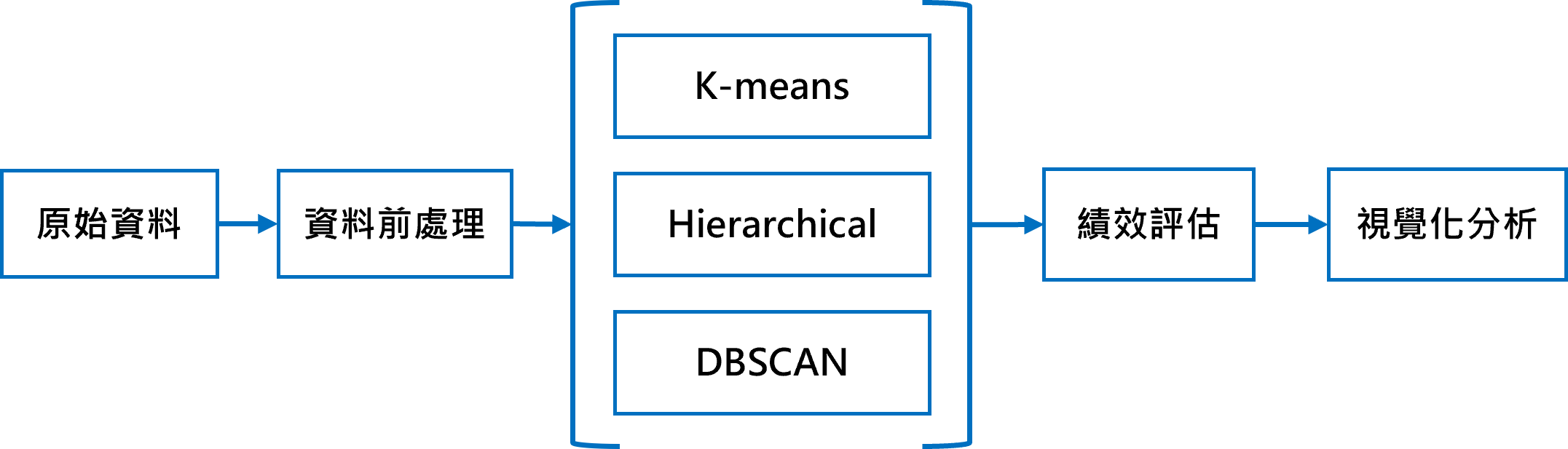


**圖 1**

*Banana資料集實驗設計流程圖*

Banana資料集的實驗設計如上圖1。做資料分群實驗前，先對Banana資料集分析處理，本研究透過IQR方法得知此資料集中未存在離群值且利用資料散佈圖得知數值範圍落在0至1之間而無需正規化。進行資料分析後，使用K-means、Hierarchical、DBSCAN演算法分群，從SSE、Accuracy、Entropy三種績效指標去調整參數，讓分群結果更加良好。在本實驗中Banana資料集被分成兩群，為將資料視覺化本研究利用Dendrogram和scatter去呈現各個演算法分群結果，其中scatter將各兩群的標籤分別標為「+」及「O」可以透過圖去更清楚地觀察分群的結果，而Dendrogram的部分可觀察群內兩兩資料之間合併的結果和群與群之間合併的過程，藉由上述去分析在不同度量距離的參數下，分群結果會產生多少變化，進而去挑選出最適當的分群方式。

1. **Size3資料集**

****

**圖 1**

*size3實驗設計流程圖*

size3資料集實驗設計如上圖1。於資料前處理，本研究利用IQR方法檢視資料集並刪除離群值、對資料集做正規化，將數值範圍落在0至1之間，於分析資料時，也是透過K-means、Hierarchical、DBSCAN演算法分群，並從SSE、Accuracy、Entropy三種績效指標去調整參數。在本實驗中size3資料集被分成四群，於scatter將分群標籤分別標為「1」、「2」、「3」、「4」使資料可被更清楚地檢視，於Dendrogram也有比較不同度量距離的參數，去評估各個分群結果為何。

1. **實驗結果**
2. **Banana資料集實驗結果**

葡萄酒品質資料集的實驗分為兩種：類別預測與數值預測。

1. **Size3資料集實驗結果**

進行模型訓練時，調整各個演算法的超參數，調整後的各個最好績效分別R2:RandomForest、RMSE:RandomForest、MAPE:RandomForest、RunTime(s):KNN。透過上述綜合結果RandomForest為表現最好的模型。

**表12**  KNN模型設定參數

|  |  |  |
| --- | --- | --- |
| **模型** | **n\_neighbors** | **leaf\_size3** |
| **KNN** | 25 | 35 |

於

**表12**  RandomForest模型設定參數

|  |  |  |  |
| --- | --- | --- | --- |
| **模型** | **min\_samples\_split** | **max\_depth** | **max\_features** |
| **Random**  **Forest** | 25 | 30 | ‘sqrt’ |

**表12**  SVR模型設定參數

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模型** | **C** | **max\_iter** | **cache\_size3** | **gamma** |
| **SVR** | 200 | 1000000 | 1000 | ‘auto’ |

**表12** 各模型績效表現

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模型** | **R2** | **RMSE** | **MAPE** | **Run Time(s)** |
| **KNN** | 0.24 | 10.87 | 0.31 | 0.0179 |
| **Random**  **Forest** | 0.25 | 10.83 | 0.31 | 1.79 |
| **XGBoost** | 0.22 | 11.21 | 0.32 | 133.11 |
| **SVR** | 0.20 | 11.15 | 0.33 | 2.45 |

1. **結論**

葡萄酒品質資料集中，分別預測了酒的品質(類別預測)以及酒成分中總二氧化硫含量多寡(數值預測)，實驗過程嘗試了不同模型參數組合並觀察各自的績效優劣。在預測葡萄酒品質時，發現了當資料集各類別資料分布有些懸殊時，容易使測試資料在預測上的績效較差，例如切割訓練資料時，第1、7類的資料只有少數兩筆，因此，測試資料的預測績效就會較差，而為了解決上述的問題，在未來的應用上可以使用Oversampling的技術，使各類別資料分布可趨於平均。此外，在這兩個實驗中，發現增加模型深度確實可以縮小generalization gap，但當深度較深時，可能會降低泛化能力，以本實驗為例，加入BatchNormalization後，確實有提高泛化能力。另外，值得我們注意的是early stopping 設定的patient以及batch\_size3的設定，在兩個實驗中，不約而同，當patient設定較大時，模型在訓練與驗證集上的績效是組合中最好的，但在測試資料上的表現卻差強人意，可能模型產生過度擬合，以至於績效較差；而在batch\_size3設定較小時，權重更新較為頻繁，模型在訓練以及用驗證資料評估的過程，雖然績效稍微差了一點，但在測試資料上反而績效為最佳，意味著當batch\_size3設定太大的時候，權重更新較不頻繁，使模型在測試資料上的績效較為不好，而較小的batch\_size3，能讓模型在預測陌生的資料上能夠更準確。

在成人資料集中，主要是透過KNN、RandomForest、XGBoost、SVR四種演算法去各別預測hours-per-week的績效為何，在調整各個演算法的超參數後，本研究的結果發現RandomForest是表現最好的模型，其中R2、 RMSE、MAPE績效分別達到0.25、10.83、0.31。

**參考文獻**

Ronny Kohavi、Barry Becker (1996)。成人數據集。

<https://archive.ics.uci.edu/ml/datasets/adult>

Paulo Cortez (2009)。葡萄酒質量數據集。

<https://archive.ics.uci.edu/ml/datasets/Wine+Quality>

Dario Radečić (2021)。如何使用 TensorFlow 優化學習率——比你想像的要容易。

[How to Optimize Learning Rate with TensorFlow — It’s Easier Than You Think | by Dario Radečić | Towards Data Science](https://towardsdatascience.com/how-to-optimize-learning-rate-with-tensorflow-its-easier-than-you-think-164f980a7c7b)

Ryan Lu (2018)。Preprocessing Data : 類別型特徵\_OneHotEncoder & LabelEncoder 介紹與實作。

[https://medium.com/ai%E5%8F%8D%E6%96%97%E5%9F%8E/preprocessing-data-onehotencoder-labelencoder-%E5%AF%A6%E4%BD%9C-968936124d59](https://medium.com/ai反斗城/preprocessing-data-onehotencoder-labelencoder-實作-968936124d59)