**K-means相關的分群演算法研究**

**The study of K-means related**

**clustering algorithm**

組員

資工 大四 B043040003 范真瑋

資工 大四 B043040020 張哲魁

資工 大四 B043040039 周家池

資工 大四 B043040044 吳俊忻

1. **簡介**

K-means是劃分方法中較經典的分群演算法之一，由於該演算法的效率高，所以在對大規模資料進行分群時被廣泛應用，目前許多演算法均圍繞著該演算法進行擴展和改進。

優點：對於大型資料集是簡單高效，時間、空間複雜度低。

缺點：

* 最重要是資料集大時結果容易區域最佳
* 需要預先設定K值
* 對最先的K個點選取很敏感
* 對noise和outlier非常敏感
* 只用於numerical類型資料
* 不能解決非凸（non-convex）資料

K-means對初始值的設置很敏感，所以有了K-means++。

K-means對noise和outlier非常敏感，所以有了K-medoids和K-medians。

K-means不能解決非凸（non-convex）資料，所以有了Kernel K-means。

1. **使用方法介紹**

* K-means++

對選取初始中心點的方法做改進，希望K個群的中心互相越遠越好。

1. 從資料集中隨機選一個作為初始中心
2. 計算每個樣本與目前已有的中心的最短距離，再計算每個樣本被選為下一個中心的機率，最後以該機率隨機選出下一個中心
3. 重複第二步直到共選出K個中心

* K-medoids

對 update 進行改進，由原本的算平均來更新新的群心，改為新的中心點就是與其他點的距離和最小的那一個，其餘步驟皆與 K-means 相同。

時間複雜度較高，為 ，因為 update 時還要去算每個點的距離，不像是 K-means 直接取平均值。

* K-medians

與K-means不同的地方

* 計算群心是利用中位數而不是平均值，可避免outlier造成影響
* 計算距離則是用Manhattan distance

優點

* 若要計算的距離為Taxicab metric，則適合用K-medians
* 計算距離的時間複雜度為O(KN)，K為群心數、N為資料數
* Kernel K-means

為解決凸集合分群問題，將原有的變數投影在高維度空間進行傳統K-means

的複雜度過高，顯然不切實際

但在K-means中，我們只需要求各點與群集的距離排序，目標函數為各點向量與群心之歐幾里得距離 ，Kernel K-means則為，由於，利用解特定映射函數，來迴避掉 直接得，解，這裡令 ，即為所謂的Kernel function，求得高維度映射空間內積。

1. **實作方式**

* K-means++

算出每個樣本被選為下一個中心的機率後，再計算累積機率。接著隨機產生一個0~1之間的數，判斷屬於哪個區間，並選出對應的點作為下一個中心。

* K-medoids

同使用方法介紹。

* K-medians

同使用方法介紹。

* Kernel K-means

由於放棄了的映射，所以與傳統K-means不同無法求群心在利用群心去計算點與各群的距離，在此處必須求點與比較群所有點之距離之平均來代表點與群之距離，因此複雜度從提升至，由於每次疊代都必須計算次的，此處我利用先行算好所有來加速計算

1. **實驗結果**

Iris (K = 3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **K-means** | **K-medoids** | **K-medians** | **KK-means** |
| **Random / RBF** | 78.855666 | 88.39000 | 81.69000 | 78.855664 |
| **K-means++ / POLY** | 78.851441 | 84.63000 | 81.69000 | 97.399080 |

※KK-means : Kernel K-means (SSE)

※RBF與POLY為Kernel K-means的kernel funciton

Abalone (K = 3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **K-means** | **K-medoids** | **K-medians** | **KK-means** |
| **Random / RBF** | 271.367966 | 282.429737 | 297.190680 | 271.373551 |
| **K-means++ / POLY** | 271.367966 | 282.429737 | 297.190680 | 293.903649 |

※KK-means : Kernel K-means (SSE)

※RBF與POLY為Kernel K-means的kernel funciton

Abalone (K = 4)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **K-means** | **K-medoids** | **K-medians** | **KK-means** |
| **Random / RBF** | 173.298094 | 208.781891 | 199.204657 | 173.297991 |
| **K-means++ / POLY** | 173.298094 | 199.313582 | 199.204657 | 189.645400 |

※KK-means : Kernel K-means (SSE)

※RBF與POLY為Kernel K-means的kernel funciton

Abalone (K = 7)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **K-means** | **K-medoids** | **K-medians** | **KK-means** |
| **Random / RBF** | 78.254143 | 92.963095 | 89.644346 | 78.255884 |
| **K-means++ / POLY** | 78.254143 | 83.255043 | 89.276590 | 89.401630 |

※KK-means : Kernel K-means (SSE)

※RBF與POLY為Kernel K-means的kernel funciton

Abalone (K = 28)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **K-means** | **K-medoids** | **K-medians** | **KK-means** |
| **Random / RBF** | 28.146570 | 38.776020 | 32.525794 | 29.859349 |
| **K-means++ / POLY** | 27.333311 | 33.747686 | 28.668192 | 36.843252 |

※KK-means : Kernel K-means (SSE)

※RBF與POLY為Kernel K-means的kernel funciton

1. **難題與解決**

* Time cost過大難以進行變數測量
* 用額外記憶體空間換取計算速度(Kernel K-means)
* 利用numpy的陣列計算取代迴圈
* Abalone dataset性別欄位不為數值
* Encoding (0, 1, 2)
* Drop column
* Abalone dataset使用K-means相關演算法分群效果不佳
* 嘗試不同的K值(3, 4, 7, 28)
* 更換其他類型演算法
* Abalone dataset使用K-means相關演算法分群發現性別影響太大
* 捨棄性別欄位
* 資料前處理(原本編碼的方式不恰當)

1. **結論**

在 Iris dataset 中，點的數量少，且點的分布相對於其他的dataset來說單純很多，分布的很廣且分離，只有一部分的點較為密集，而各種方法就會在此產生些微的差距。另外，已知資料的群數為3群，若以此為 K ，與原始資料比對其所屬的各個群組，則各個方法差距微小，而且結果與原始分群相近。

而在 abalone dataset 中，不但屬性很多，資料量也很大。若以年齡作為分群目標，實驗後效果均不佳，推測有以下原因：樣本的年齡分佈不均、資料集中的數據可能與年齡較無太大關聯、性別欄位的處理不恰當、K-means相關的演算法不適合處理此資料集…。

沒有一種 K-means 的演算法是萬用的，針對不同的資料集，需要看情況去決定要使用哪一種的演算法，而學會所謂的「看情況」是我們要學習的重點。

1. **參考資料**

Arthur, D. & VassilvitsKii, S. (2007). K-means++: The Advantages of Careful Seeding.

Kaufman, L. & Rousseeuw, P.J. (1987). Clustering by means of Medoids.

Jain, A. K., & Dubes, R. C. (1988). Algorithms for clustering data. Englewood Cliffs, NJ: Prentice-Hall.

Ardian Umam. (2017, December 15). Machine-Learning-From-The-Scratch. Retrieved from <https://github.com/ardianumam/Machine-Learning-From-The-Scratch>.

Welling, M. (2013). Kernel K-means and Spectral Clustering.

Wolper, D. H. & Macrea, W. G. (1996). No Free Lunch Theorems for Optimization.

MayuKh, H. (2010). Age of Abalones using Physical Characteristics:A Classification Problem.

廖韋雅、詹祥麟、薛友仁(2007)。集群方法與集群指標關係之研究。