Esra Eryılmaz
171044046

CSE 454 – Data Mining (Fall 2021)
Assignment #1
Report

How parameters effect the results?

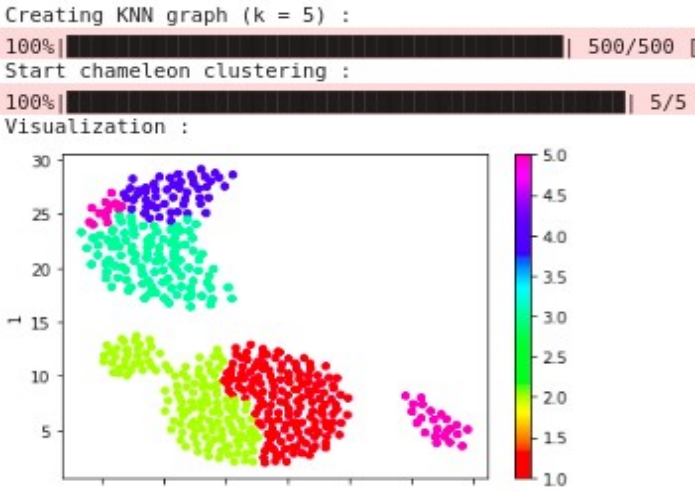
Clusters:

While performing, I used popular dataset : aggregation dataset, but I reduced it a little bit.

Parameters → knn, c1, c2, alpha

Change knn value :

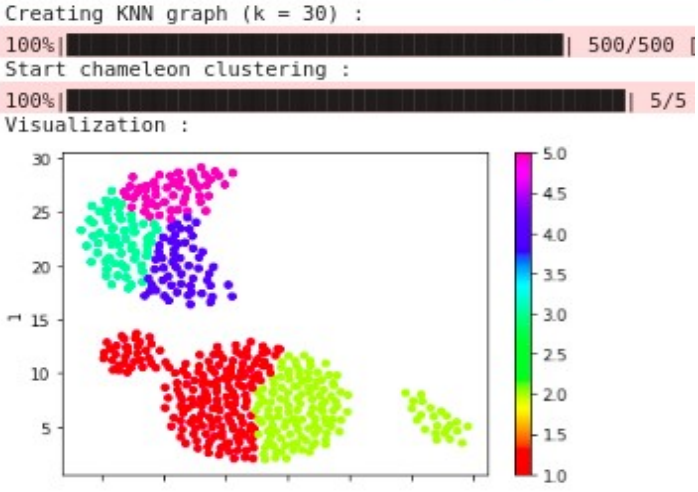
knn : 5 → 30 → 50
c1 : 5
c2 : 20
alpha : 2

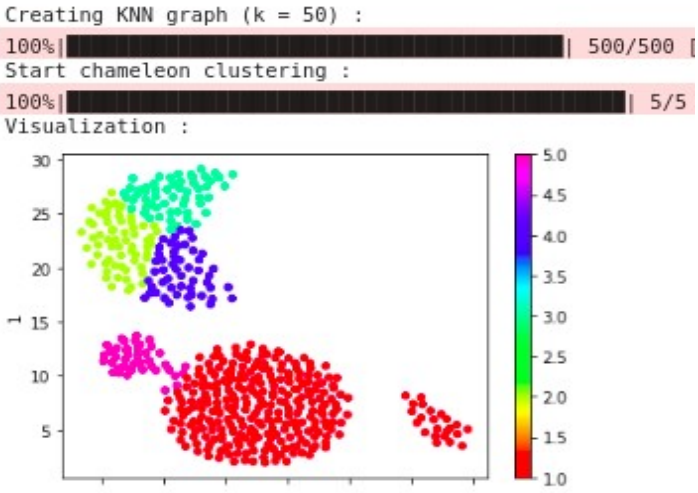


knn : A small value of k means that noise will have a higher influence on the result and a large value make it computationally expensive.

While choosing k value if I use emprical method than k should be √ 500/2 = 15

So I think bigger k value gives better results.





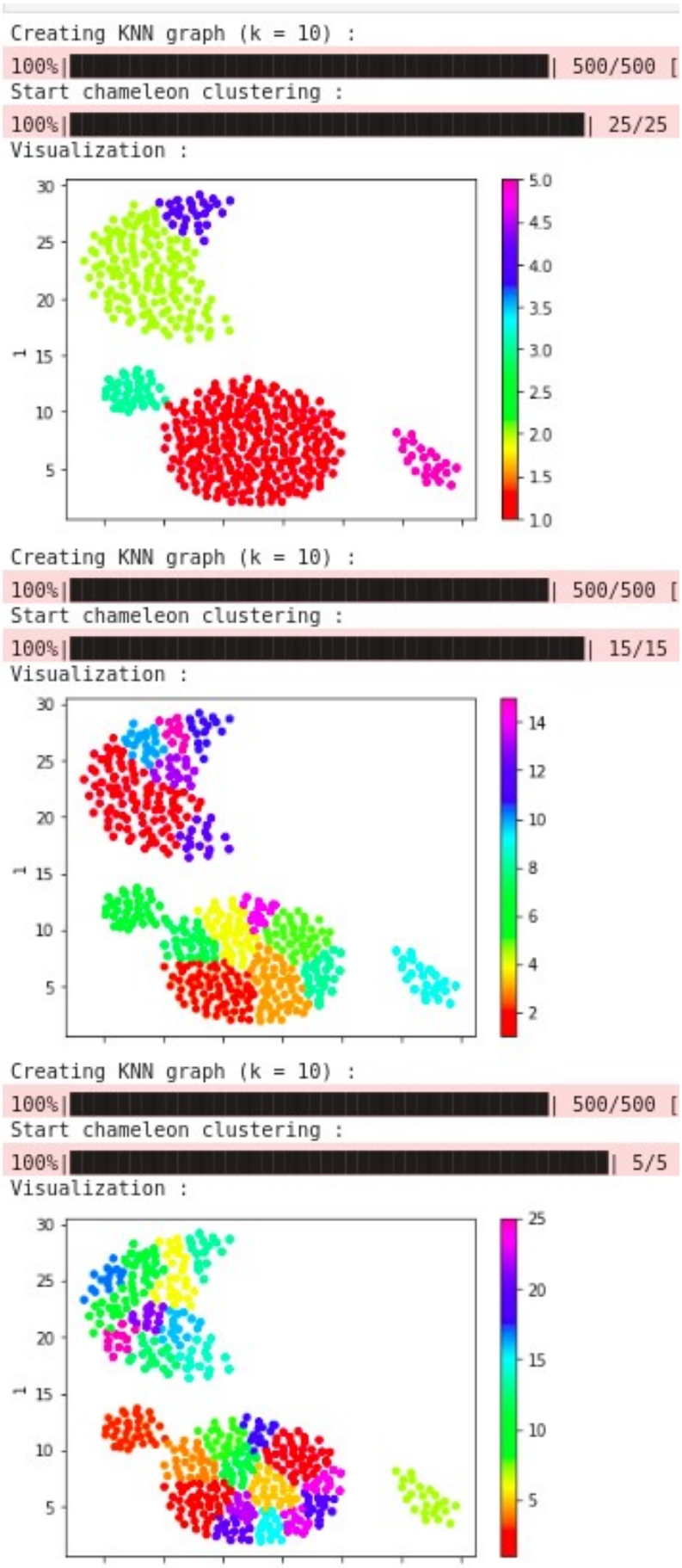
Change c1 :

knn : 10
c1 : 5 → 15 → 25
c2 : 30
alpha : 2

c1 : Parameter c1 is the cluster numbers used in the merge_partition(graph, dataset, alpha, c1) function.

It is second phase for combining the relative inter-connectivity and relative closeness.

Higher c1 values gives us smaller clusters and high number of clusters. So I think bigger c1 value gives not good results.



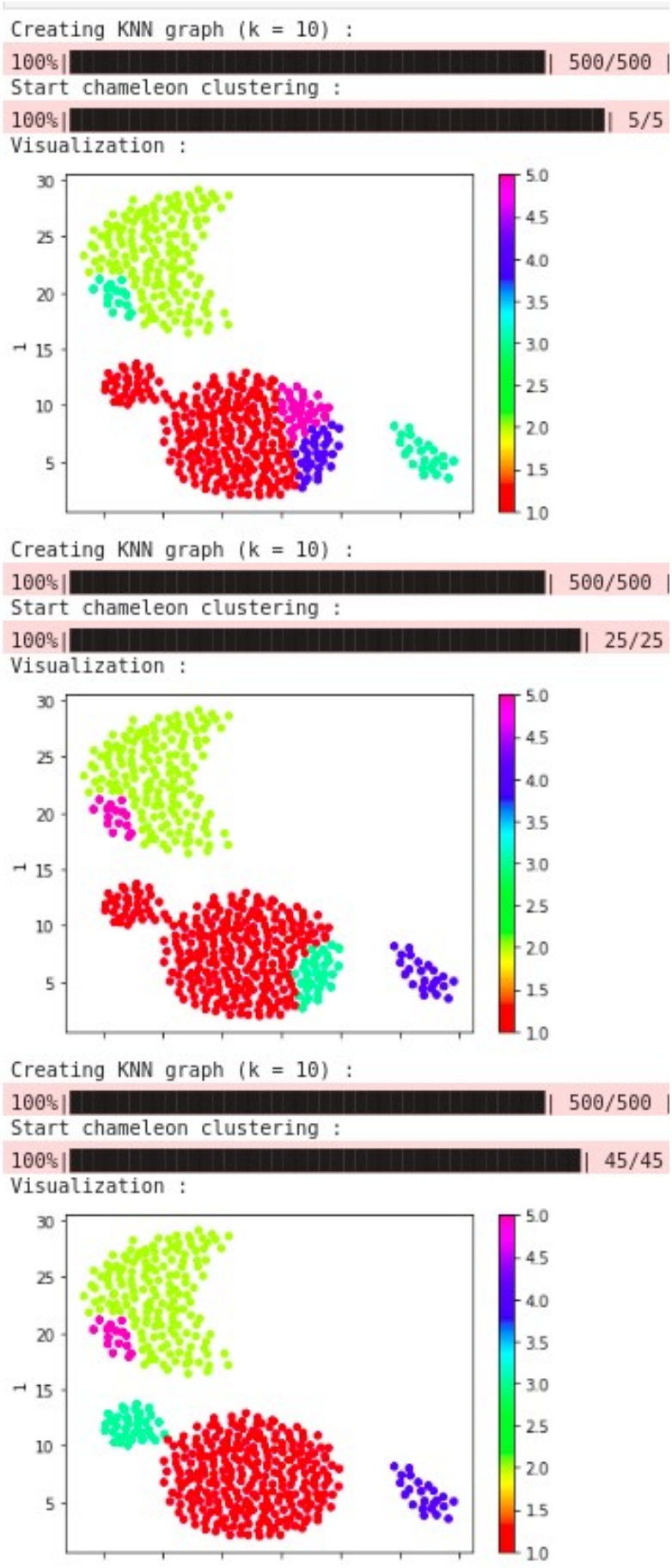
Change c2 :

knn : 10
c1 : 5
c2 : 10 → 30 → 50
alpha : 2

c2 : Parameter c2 is the cluster numbers used in the partition_graph(graph, c2, dataset) function.

It is initial sub-cluster numbers used in the first phase.

I think bigger c2 value gives good results. But I think it changes with the dimensionality of dataset.



Change alpha :

knn : 10

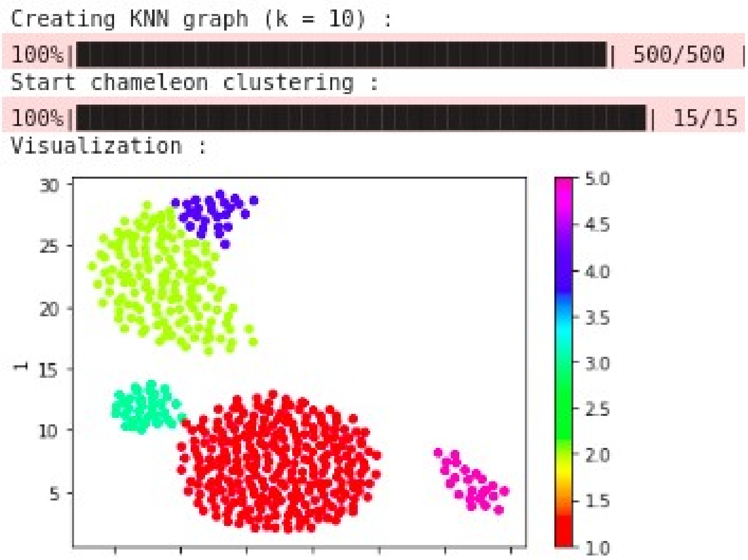
c1 : 5

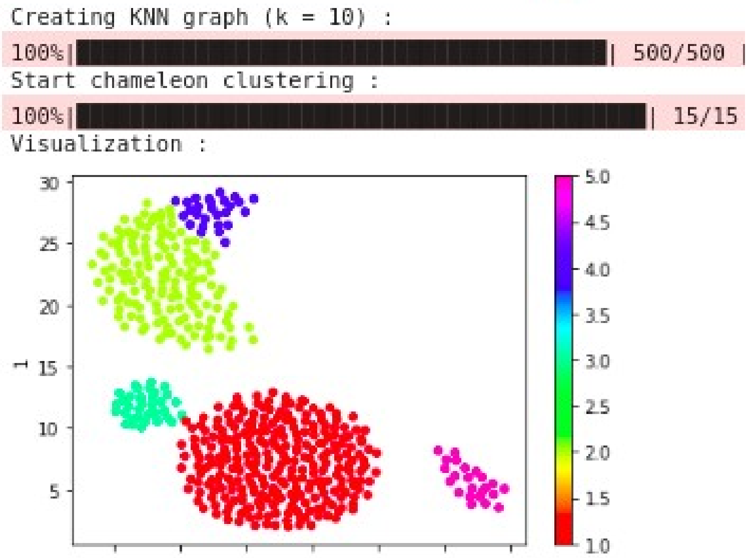
c2 : 20

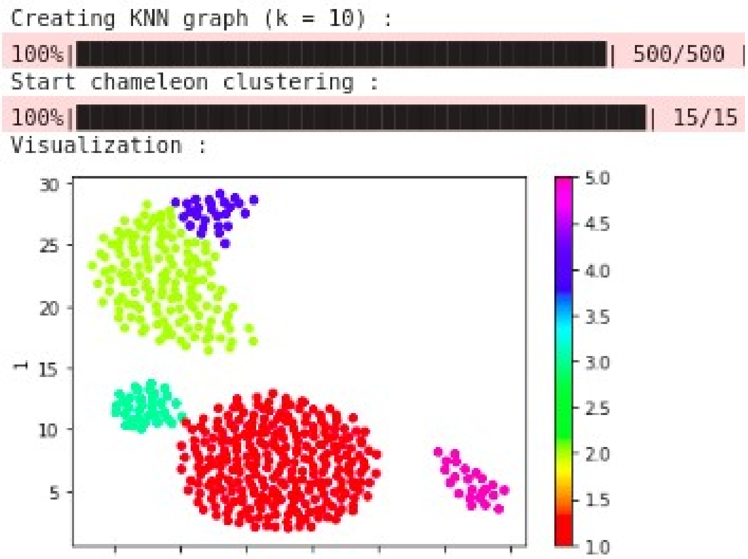
alpha : 1 → 50 → 100

alpha : It does not effect the results , for taking power of relative_closeness() and alpha.

It is like constants.







What are the advantages and disadvantages of the algorithm? Compare it with other clustering techniques.

Advantages :

It is efficient for low dimensional space.

hMETIS quickly produce high-quality partitionings for a wide range of unstructured graphs and hypergraphs.

Chameleon good at finding clusters of arbitrary shape other than other clustering algorithms. For example BIRCH good at performing with spherical data not arbitrary shape data.

Chameleon works on data of all attributes. But for example STING mainly good with numerical values. K-means works only on numerical data.

Disadvantages :

Worse case complexity O(n^2) in high dimensional space. For example comparing to the dbscan or optics algorithms, chameleon complexity is bad.

Chameleon has problems when the initial sparsification and graph partitioning process does not produce subclusters, as is often the case for high-dimensional data.

What is the time complexity of chameleon? How it is comparing to the other clustering techniques?

Chameleon, at first, divides the original data into clusters with smaller size based on the nearest neighbor graph, and then the clusters with small size are merged into a cluster with bigger size, based on agglomerative algorithm, until satisfied.

The overall computational complexity of chameleon depends on the amount of time it requires to construct the k-nearest neighbor graph and the amount of time it requires to perform the two phases of the clustering algorithm. It depends on the dimensionality of the under-lying dataset.

Not very good in terms of time complexity. It gives good results but takes long time in large datasets.

The overall complexity of chameleon two-phase clustering algorithm is :
O($n + nlogn + m^2logm$) = O(n^2)

Chameleon : O(n^2)

Birch : O(n)

Cure : O(n^2logn)

K-means : O(n)

K-medoids : O(n^2)

DBSCAN : O($nlogn$)

OPTICS : O($nlogn$)

STING : O(n)

Chameleon may give better results on small datasets. But in large datasets, because of the complexity another clustering technique can be used.

Complexity is bad comparing to the birch, k-means, dbscan, optics and sting clustering techniques.