CMPSC497 Fall 2022 Programming Assignment 2.

Assigned: Friday, November 4, 2022

Due: Friday, December 2, 2022 (by midnight, submit two package of codes in .ipynb and a report in .pdf via Canvas)

Maximum: 120 points (out of 100 points)

Note: This assignment is to be done by an individual student, no team work allowed.

In this assignment, you will implement the Agglomerative Hierarchical Clustering - MIN algorithm. Additionally, you will use the Agglomerative Hierarchical Clustering algorithms with different options of cluster similarity measures (available in sklearn) to cluster the *Credit Card Clients Dataset* for comparison and perform cluster analysis.

The clustering is based on three sets of attributes/features: i) Pay Amount, ii) Bill Amount, and iii) Remaining Balance, which is a new set of features derived from Pay Amount and Bill Amount. For your implementation, you need to detail your design of some important components and show your execution results in the report. After running the Agglomerative Hierarchical Clustering software on the dataset to generate different clusterings, you need to compare them and perform a cluster analysis on the original dataset to discuss the results and report your findings.

Dataset and new features

- *Credit Card Clients Dataset*. In this assignment, we define new features, *Remaining Balance*, and add them (6 features/columns in total) to the dataset. Experimentally, you will use Remaining Balance, Bill Amount, and Pay Amount (18 features in total).

Implementation and Experiments

Task 1. Data Preprocessing [11 points]

a) Please calculate new features *Remaining Balance = Bill Amount – Pay Amount* and add them (6 features/columns in total) to the dataset. You may use excel or write code for the task. Please show the result of the first 3 records in your report. [3 points]

b) As 18 attributes are used for clustering (which is quite many), please apply *Principal Component Analysis (PCA)* from sklearn to reduce the dimensionality of attributes to 3 (i.e., n_components=3). Note that normalizing the attributes by *standardization* before PCA is a common practice. Please perform the standardization in this task. Please show the result of the first 3 records in your report. [5 points]

c) Before you do the clustering, please perform *normalization* (into [0,1]) on the three attributes used for clustering. Describe the normalization scheme you adopt and show the result of the first 3 records in your report. [3 points]

Task 2. (Your own) Implementation of the Agglomerative Hierarchical Clustering (MIN) algorithm [36 points]

Agglomerative Hierarchical Clustering algorithm is a clustering algorithm widely used for explorative analysis of (often unknown/unlabeled) data. The goal is to form a hierarchy (tree) that encodes cluster-subcluster relationship and the order in which the clusters are formed (merged). Basically, the algorithm starts with individual data points as clusters, successively merges the two "closest" (or most similar) clusters, based on the considered features/attributes of the data points, until only one cluster remains. Please perform the following subtasks and present your result in the assignment report.

a) Represent the data points in a cluster using a data structure or alternative method. The data structure should encode the hierarchical structure of data points in the cluster. Discuss/show your design and illustrate it by examples in the report. If you do not use a data structure in your design, explain your idea and show your design (with illustration by examples) to represent a cluster in your implementation. [4 points]

b) *Proximity matrix* is essential to Agglomerative Hierarchical Clustering algorithm. In the report, explain your choice of the similarity or dissimilarity measure in the proximity matrix and show/explain the module you code to calculate/construct the matrix. [4 points]

c) Find two closest clusters based on the MIN/Single Link scheme. In the report, show and explain the module of your implementation code for it. [4 points]

d) Merge two clusters into a new cluster and update the proximity matrix. In the report, show/explain the module(s) of your code for implementation of the above

operations. Also use an example (based on the data structure of your design in Task 2(a) to illustrate the merged cluster. [8 points]

e) In this task, you are asked to implement your own version of the Agglomerative Hierarchical Clustering (MIN) algorithm in python. Given a dataset and a parameter *n_clusters*, your code should return the specified number of clusters, each of which represented in the data structure of your design. Additionally, a label (i.e., 1, 2, ..., *n_clusters*) should be assigned to every data point in the dataset. In this task, please use the first 100 data points from Task 1 and *n_clusters* = 5 to generate your clustering result. [16 points]

   In addition to answering the design questions in Task 2(a)-(d) in your report, you need to submit your runnable implementation code and result in Jupyter Notebook. The code should be well formatted and well documented, without leaving any unnecessary test code or commented-out print statements in the final submission. This requirement is factored in the grading.

Task 3. Hierarchical Clustering [34 points]

In the task, you are asked to perform hierarchical clustering using different options of the agglomerative hierarchical clustering algorithms available in sklearn for exercise and comparison. You may also use scipy instead of sklearn. Of course, you should feel free to use your own implementation as part of the tasks, as you see appropriate.

a) Perform the agglomerative hierarchical clustering (MIN) to cluster the data obtained from Task 1. Our goal is to extract five largest clusters for further analysis later. To visually decide those clusters, please draw *dendrograms* (with sufficient levels) for inspection. Based on the decision made from your inspection, re-run the clustering to obtain the clustering. You may use the parameters *distance_threshold* or *n_clusters* in *sklearn.cluster.AgglomerativeClustering* to decide the clustering, but note that *n_cluster = 5* does not necessarily give you the five largest clusters.

   In the report, please show (i) explanation of your decision, with illustration and justification based on the dendrogram(s) shown. If you decide to extract an alternative number of clusters (instead of 5), please explain explicitly with clear justification; (ii) a list of clusters extracted. For each of those clusters, please show the first 5 data points and the total number of data points in the cluster. [15 points]

b) Repeat Task 3(a) using the agglomerative hierarchical clustering (MAX). [7 points]

c) Repeat Task 3(a) using the agglomerative hierarchical clustering (WARD). [7 points]

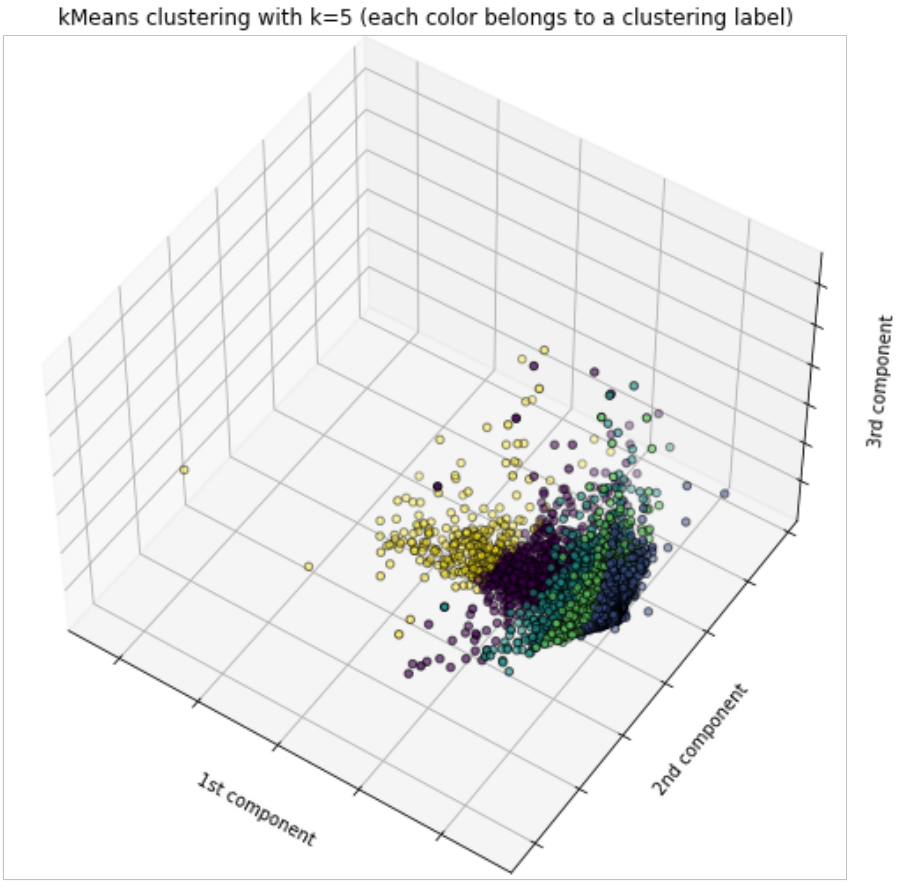
d) Compare and report what you find via the process of performing Task 3(a)-(c). Which clustering looks better than others based on what you observed so far? Note that it's all right if your findings at this point is different from your later conclusion. [5 points]

Task 4. Cluster Analysis [39 points]

After the clusterings are decided, an important task is to perform data analysis on data objects in each cluster. The goal is to characterize and understand better the clients in different clusters. For objective comparison, a frequently used internal metric is *sum of squared errors (SSE)*. Additionally, we may apply external metrics, such as *Entropy* or *GINI Index*, on some of the attributes, e.g., the *default status* (i.e., the class label used in the classification task), to help assess the clusterings and their clusters.

To carry out a cluster analysis, you may explore the attributes/features to characterize the clusters. You may examine the attributes of the medoid (i.e., the representative client) in each cluster or calculate the statistics of user attributes in each cluster for comparison. For example, you may calculate the Impurity (GINI Index) of the customers' default status (and gender, marital status, etc.) or calculate the average payment (and age, credit, etc) for each cluster. You may also analyze the distributions of various attribute values to see whether you can capture a pattern. There is various interesting information about customers in different clusters that you can explore (i.e., the more the better) and make comparison. Please show your findings in the follow-up subtasks/discussions.

a) Given the three hierarchical clusterings (i.e., from MIN, MAX and WARD, compute and report their *SSE (sum of squared errors)*. Additionally, show the three clustering results using matplotlib. Note that cluster labels are generated when you run sklearn.cluster.AgglomerativeClustering, so the plotting should be easy. An example figure with k=5 is shown below.



In the report, show the SSE and scatterplots of the three clusterings. Explain (i) which clustering is the best based on SSE; (ii) the shapes of clusters based on your inspection/observation of the plots, with an explanation linking to the different cluster similarity measures (i.e., MIN, MAX and WARD) used; iii) whether SSE favors one of the measures and why. [8 points]

b) Apply external measures, specifically GINI Index, entropy, and percentage of defaults, on the *default status* to assess the quality of clusterings. Which measure is a better tool? Which measure can reveal more interesting information under the context of this application? In the report, show your result and your findings with explanation. [8 points]

c) Upon the clustering selected based on the best SSE, perform data analysis of clusters for comparison, e.g., in terms of the characteristics and distribution of data points in those clusters. In the report, show your result and findings with explanation. Illustrations with tables and figures are excellent.

Note that, in addition to the class label, i.e., default status, other attributes, e.g., marriage, may also be interesting. Please exercise the concepts and skills you

learned in the course in the comparison. Simply completing the comparison mentioned, i.e., default status and marriage, is insufficient. This task will be graded based on both of the effort you made on the analysis and the findings. Please make as many analyses as you can and report your findings. [15 points]

d) In Task 1(c), the data points are normalized. Is it really better than clustering without preprocessing with normalization? Please use data without normalization to find the best clustering. Compared with the normalized data, present the plots to make comparison. In the report, show the comparison and present your findings. (Note: you are not required to perform clustering analysis in this task, but you are welcome to do so.) [8 points]

Deliverables

- Please submit the report (.pdf) and two Jupyter Notebook (.ipynb) on Canvas. One Jupyter Notebook contains your implementation of the Agglomerative Hierarchical Clustering (MIN) algorithm and the other for the rest of the tasks.
- The report should contain the experimental results for various tasks in the assignment.
- The code should be well formatted and well documented, without leaving any unnecessary test code or commented-out print statements in the final submission. This requirement is factored in the grading.