

**Applied Integrative Analytics Capstone Project**

Elderlies Loneliness Reduction Capstone Project

Tianyi Wu[[1]](#footnote-2)\*

Ontario Tech University, 2000 Simcoe Street North, Oshawa, Ontario L1G 0C5, Canada

Abstract

Loneliness has become a widespread issue globally, affecting many individuals to varying degrees. It is now considered to be on par with other significant concerns like obesity and smoking. To put it into perspective, the negative health impact of loneliness is comparable to smoking 15 cigarettes a day (UneeQ, 2023). Older adults are particularly vulnerable to loneliness, and with the increasing aging population worldwide, there is a growing need to address and alleviate loneliness among the elderly. Clustering algorithms are usually used to cluster datasets based on the similarity of data points. The objective of this project is to apply different algorithms to the generated elderly dataset to match elderly individuals into suitable groups based on their basic information data. By facilitating connections and communication among them, the aim is to alleviate loneliness to some extent. Creating opportunities for shared experiences and social interactions among elderly individuals can help combat the negative effects of loneliness and improve their overall well-being. © 2001 Elsevier Science. All rights reserved.

*Keywords:* Clustering Algorithms; Similarity Measurement; Loneliness; Elderly

1. Introduction

The primary objective of this project is to identify relevant characteristics of elderly individuals and utilize algorithms to assist them in finding others who share similar features, thereby facilitating their grouping. To achieve this goal, a dataset containing information about elderly individuals needs to be employed. However, it is crucial to consider ethical concerns, such as the potential exposure of personal privacy information. Furthermore, accessing existing datasets may be challenging due to limitations in publicly available sources. As a result, a mock dataset has been generated specifically for this project. Moreover, the project is focused on helping elderly individuals make new friends and reduce loneliness. To accomplish this, clustering algorithms like DBSCAN have been adjusted and utilized within the project. These algorithms aim to group individuals based on their shared characteristics, enabling potential connections and interactions. Finally, it is essential to test the algorithms and evaluate the efficiency and representativeness of the results.

To produce an initial dataset for the project, a mock dataset has been generated based on authoritative reports on the elderly. The dataset variables were selected following certain rules. These rules ensure that the statistics in the dataset are generated from real-world collections, are correct and clear, and are reasonable representations of elderly characteristics. In the process of setting up the data, reasonable guesses and modifications were made based on the statistics from the reports. The goal was to create a usable dataset for the project. By analysing the dataset, several patterns about the elderly can be observed. For example, when analysing the data related to gender and age, it appears that women have a higher possibility of living beyond 90 years old. This suggests a higher likelihood of longevity for women, as there are more female age outliners above 95 years old compared to males.

In the project, various matching applications and clustering algorithms have been explored and applied. Matching applications such as dating apps, mentorship platforms, and healthcare matching platforms have been considered as potential models to adapt for the elderly grouping and connection purposes. In terms of clustering algorithms, several algorithms have been discussed: DBSCAN, K-means, Mini Batch K-means, GMM algorithms and Hierarchy clustering. These algorithms offer different approaches to clustering data based on their characteristics and objectives. By applying these selected clustering algorithms to the generated dataset, the aim is to group elderly individuals with similar characteristics and facilitate meaningful connections among them. The choice of algorithms allows for flexibility in handling different types of data variables, enhancing the accuracy and effectiveness of the clustering process.

To evaluate the performance of clustering algorithms and assess the quality of the clustering results in this project, several measurements are necessary. One commonly used measure is the Silhouette Score, which quantifies the compactness and separation of clusters. Additionally, the t-SNE (t-Distributed Stochastic Neighbor Embedding) dimension reduction method has been employed for visualization purposes.

1. Literature Review
   1. Real-World Applications

There are numerous matching applications in the real world, with dating apps being the most widely used. Among them, there are two popular types of dating platforms: Swipe-Based and Algorithm-Based Dating Apps. Algorithm-based dating apps, which are more relevant to this project, employ a series of questions and algorithms to match users with compatible individuals. This process resembles collecting datasets, with more personal and private information involved. These apps adopt a more scientific approach to match, utilizing algorithms to identify compatibility and assist users in finding partners who are a good match for them. Prominent examples of existing dating apps, such as eHarmony and OkCupid, show the utilization of such algorithms for effective matching (ANDERSON, 2023). For instance, OkCupid uses users' preferences and calculates the weight of their answers, presenting potential matches with the most in common, along with a match percentage (Awa-Abuon, 2022). One advantage of the algorithm-based approach, as exemplified by OkCupid, is that it focuses more on compatibility rather than appearance, which the latter is main consideration of swipe-based algorithms. This makes it more suitable for those who are searching for long-term relationships. An obvious disadvantage is that it requires time and patience to use since users must provide a lot of answers.

Another application of matching is seen in matching seniors with suitable healthcare services. HomeHero specifically focuses on helping seniors who require healthcare but struggle to find agencies that can provide the most suitable services. Their algorithms prioritize individualization to address the unique demands of each senior client. HomeHero employs the K-means algorithm to achieve this objective. When a user completes their application form, the algorithm ranks healthcare service providers in HomeHero's database based on a normalized set of quantitative attributes, such as personality and compatibility. To simplify the calculations, variables are normalized, and different weights are assigned to each variable in the algorithm. In HomeHero's algorithm, each variable is assigned a score ranging from 0 to 100. A higher score indicates a greater likelihood for a healthcare service provider to appear on the recommendation list (Hill, 2015).

Another example of an existing matching application is Together, a mentor matching platform. This platform utilizes a pairing algorithm to connect users with mentors who can offer guidance and advice to help them achieve their goals. Before the matching process begins, users are required to complete a questionnaire. This information is used to match them with mentors from existing HRIS databases. The platform allows users to customize their matching preferences, enabling them to set different priorities for each match. Together offers two methods for matching users and mentors. The first method is Admin Led, where a project manager utilizes the matching algorithm to pair each user and simultaneously sends the match results to both the users and mentors. The second method is Mentee Led, which empowers users to choose their own mentors from a recommended list. In the Mentee Led approach, users select one mentor from three recommended options. Mentors, in turn, have the option to request or accept the matching request. If a user's request is rejected, the process moves on to the next mentor in the circle and repeats until a final match is made (Together, 2023).

* 1. Matching Algorithms
     1. Gale-Shapley Algorithm

One example of an existing matching algorithm is the Gale- Shapley Algorithm, which is employed by the dating app Hinge. The Gale-Shapley Algorithm, an older algorithm, focuses on quickly establishing stable marriage relationships. In the context of Hinge, this algorithm is utilized to pair individuals who are likely to share a mutual interest in each other. Hinge uses the Gale-Shapley Algorithm to consider users' engagement and preferences, matching them with individuals who have similar preferences. This approach increases the likelihood of a mutual attraction between users. By leveraging this algorithm, Hinge aims to facilitate meaningful connections based on shared interests and compatibility factors. (Scanlon, 2020). However, this method has its limitations as it does not fully consider individuals' genuine emotional needs. It follows a predefined set of rules for romance without fully addressing users' unique emotional requirements. As an alternative approach, Alvin Roth proposed combining the Gale-Shapley Algorithm with the NRMP (National Resident Matching Program) matching algorithm to enhance the stability of matches in real-world systems (University of California, 2016).

* + 1. Collaborative filtering

Collaborative filtering is another algorithm mentioned by researchers for matching algorithms. This algorithm predicts matches based on both the user's personal preferences and the collective opinion of the majority. It operates on the principle that users who have similar preferences in the past are likely to have similar preferences in the future (Data&Tech, 2023). Collaborative filtering is commonly utilized in recommendation systems that rely on users' historical behavior datasets, particularly for movie or product recommendations. Industry giants like Amazon and Facebook have extensively employed this algorithm in their platforms. Collaborative filtering can be classified into two categories: user-based and item-based. User-based collaborative filtering involves measuring the similarity between the targeted user and other users based on their past behaviour. On the other hand, item-based collaborative filtering focuses on measuring the similarity between items themselves (Kurama, 2022).

Apart from the matching algorithm, this project could also be regarded as a clustering problem. The objective is to merge individuals who possess similar features. Clustering entails grouping objects based on similarities and determining the groups using unlabelled data. It primarily relies on the specific user and the scenario (CR, 2023). However, before clustering the dataset, we need to calculate the similarities between data points. These data points can then be used for further cluster analysis.

* 1. Clustering Algorithms
     1. Hierarchical Algorithms

For clustering algorithms, there are various types available. One such example is the Hierarchical Clustering algorithm. Hierarchical clustering usually contains two types, one is agglomerative type, and another is divisive type. The difference between those two hierarchical clustering algorithms is that agglomerative type first assumes each data point as a cluster, and then cluster the similar clusters/points until the previous requirements are satisfied. Divisive type first assumes all the data points as one cluster, then it divided the different points into different clusters until the previous requirements are satisfied. When clustering, hierarchical clustering algorithms calculate the distance between two data points and create a distance matrix, usually it will calculate the Euclidean distance. Then after new points join in or separated, the distance will be calculated again until the pre-defined requirements being satisfied (Karabiber, Hierarchical Clustering, 2023).

* + 1. DBSCAN

The DBSCAN clustering algorithm can be modified to be more suitable for mixed data types, especially when dealing with large datasets. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups data points based on their density. Unlike other clustering algorithms, DBSCAN does not require the initialization of a specific number of clusters, and it is robust to outliers. Instead, DBSCAN relies on two parameters that need to be determined beforehand: epsilon and minPoints. Epsilon represents the radius of the data circle used for density checks, while minPoints denotes the minimum number of data points required inside the data circle around a single data point. If a data point has enough data points within its specified radius, it is labeled as a core point, indicating its membership in a cluster (Sharma, 2020).

* + 1. K-means and other relevant algorithms

K-means is a clustering algorithm designed to divide numerical data points into groups, based on a predefined number of clusters. Typically, each group will contain an equal volume of data. This algorithm calculates the Euclidean distance, and within each group, each data point will have the minimum Euclidean distance to the respective k- centroid point. However, one disadvantage of K-means is its inability to handle categorical data. To address this limitation, alternative algorithms such as K-Modes and k-Prototype have been introduced. K-Modes is a variation of K-means that is specifically designed for categorical data. and k-Prototype is an extension algorithm of K-Modes that can handle mixed data types, including both numerical and categorical data (Kumar, 2021). Similarly, to K-means, the k-Prototype algorithm also calculates the Euclidean distance between numerical data points. The difference lies in the way it measures the distance between categorical data points. In the case of categorical data, k-Prototype utilizes the number of matching categories as a metric. Ultimately, the k- Prototype algorithm computes the actual distance between data points by combining the metrics for both numerical and categorical data points, allowing for clustering of mixed data types (Aprilliant, 2021). They’re going to have problems when the dataset is too large, so based on K-means, mini-batch k-means has been raised. It is more efficient than K-means since the mini-Batch k-means use small, random, and fixed data batch first store the data, then iterate the collected dataset sample (Kharwal, 2021).

* + 1. Gaussian Mixture Models (GMM)

K-means is calculated the mean, there are also another algorithm better than k-means by calculating data points’ not only the mean, but also variance and covariance, the GMM algorithm (Gaussian Mixture Models). GMM is a clustering algorithm based on distribution. Each cluster distribution has its own mean and variance, which is how GMM is calculated. This algorithm is particularly useful when dealing with cluster shapes that are "elliptical" (Dobilas, 2021). At the beginning, we will consider the dataset contains several normal distribution data clusters, which is also being called Gaussian Distribution here. For each Gaussian Distribution, the algorithm will analysis each data points in the dataset and decided which Gaussian Distribution best suitable for them. To achieve that, the algorithm will calculate the probability of points for each of the distributions. For each data points, after testing and comparing the probability being in different distributions, data points belong to the distribution which has higher probability (Santos, 2022). To utilize the algorithm effectively, it is important to decide on several parameters beforehand. Two key parameters are the init\_params and covariance\_type. The init\_params, which used to initialize weights, means and the precisions, it defaults set as k-means, and the covariance\_type, which used to describe the used type of covariance parameters.

* 1. Similarity Measurement
     1. Gower Distance

There are several metrics that can be utilized to measure similarity, and one of them is the Gower Distance. This metric is particularly useful for assessing similarities between data points that contain a mixture of different data types. The Gower Distance calculates the distance between each pair of data points using various methods based on the type of data being compared. For instance, when dealing with numerical data, it employs the normalized Manhattan Distance, whereas for categorical data, it converts the data into binary columns and then calculates the Dice Coefficient. The general form of the coefficient is represented by the function below, where Sijk represents the contribution provided by the Kth variable, and Wijk is either 0 or 1, depending on whether the comparison is valid for the Kth variable or not (Kesh, 2020).

A picture containing font, text, line, white

Description automatically generated

* + 1. Jaccard Similarity Coefficient

Another measurement for calculate similarity is Jaccard similarity coefficient (J). Jaccard similarity coefficient can compare the similarity between two objects, usually its two sets of data. It marks the same data as 1 and 0 if not the same, then it calculates the percentage of similarity between two sets by dividing the same data counts by the total set data number. The calculate function is J(A,B) = |A∩B|/|A∪B| (Karabiber, Jaccard Similarity, 2023).

* + 1. Silhouette Coefficient Score

Another method for calculating metrics is called Silhouette Samples, which provides a silhouette score for each data point within a cluster. This metric is commonly used to evaluate the results of clustering analysis. The silhouette score ranges from -1 to 1. When the score is closer to 1, it indicates that the data points have been well- clustered and can be clearly distinguished within their respective clusters. Conversely, a score closer to -1 suggests that the data points have been incorrectly clustered. The calculation of the silhouette score is done using the formula: Silhouette Score = (b - a)/ max(a, b), where "a" represents the average distance between data points within the same cluster, and "b" represents the average distance between all clusters. (Bhardwaj, 2020). The Silhouette Samples method can also be used to determine the number of clusters, typically represented by "k." The idea is to calculate the Silhouette score for different cluster results generated by varying values of k. The cluster result with the highest score is considered the best, and the corresponding value of k is chosen as the optimal number of clusters (Scikit learn, 2023). For K-means and its extension algorithms, it is crucial to set an initial value for k at the beginning. This number plays a vital role as it determines the number of clusters that the algorithms will generate.

* 1. Elbow Method

Another Method to decide the best cluster number and commonly used is the Elbow Method. It first calculates the Within-Cluster Sum of Square (wcss), which represents the total square distance between the points with the centroid within one cluster. This value will decrease when the cluster number increased. The elbow method is going to visualize the decreased wcss and find the rapidly changed part. That rapidly changed at one point, that point is the best number of clusters. The curve looks like elbow and that is how the name came. After that point, the curve will start to parallelly close to the x-axis and gradually stay stable (Saji, 2021). Elbow method with calculating wcss usually only for k-means algorithm, for other algorithms such as GMM, wcss cannot be used, [Bayesian Information Criterion](https://en.wikipedia.org/wiki/Bayesian_information_criterion) (BIC) could be calculated instead and following with the elbow method to find the best number of cluster (Sarkar, 2019).

* 1. Dimension reduction
     1. t-SNE

Moreover, dimension reduction could also be considered, especially when dealing with a high number of variables. One possible method for dimensionality reduction is t-SNE. t-SNE is a technique that is commonly used for visualization as it effectively reduces the dimensionality of data. It is particularly useful for large real-world datasets (DataTechNotes, 2020). By combining t-SNE with dimensionality reduction, it can be further employed in conjunction with other clustering algorithms such as GMM.

* + 1. Principal Component Analysis (PCA)

Another dimension reduction method is Principal component analysis (PCA), it is suitable for large-scale, high-dimensional dataset and during the reduction, this method can minimize the data loss. PCA turns the original variables into a set of uncorrelated linear variables, which are called principal component. By finding the maximum variance directions in the principal component, then projecting data onto these directions, the dimension reduction will be achieved. Then through sort the variance explained by the principal component, it can tell which data is more important for the dataset and can be utilized for different further analysis. The overall dataset direction is decided by the eigenvectors, which eigenvectors number is decided according to the chosen principal components. The size of dataset direction is decided by the eigenvalues (avcontentteam, 2023).

* 1. Latin Hypercube Sampling (LHS)

LHS is a statistical method for randomly sampling numbers from a distribution with the goal of achieving a more uniform distribution. It involves dividing the sample space into equally sized intervals and selecting one point from each interval with the same probability. For instance, when sampling from a normal distribution, traditional random sampling tends to select points closer to the mean (µ) with higher probability, while neglecting the lower probability regions in the tails of the distribution. In contrast, LHS considers each interval in the sample space, ensuring that points from lower probability density regions are also included in the sample. LHS is particularly useful for datasets with multiple dimensions. For example, in two-dimensional LHS, each variable (x and y) is divided into n equally sized regions, and one point is randomly selected from each region. This process generates a two-dimensional random sample set that covers a wider range of the distribution. (Zach, 2020).

* 1. Potential Utilization

Regarding the literature review, there are several potential algorithms that could be employed in this project. Firstly, the Gale-Shapley Algorithm could be considered. One approach is to modify this algorithm by incorporating other algorithms to overcome its limitations. For instance, we can introduce additional factors specific to elderly individuals, such as health conditions, sleep patterns, or interests related to romance. By incorporating these factors and tailoring the Gale-Shapley Algorithm accordingly, we can enhance the matching process and address the mental well-being of elderly individuals more effectively. Secondly, hierarchical algorithms can be utilized. To integrate it into this project, hierarchical clustering can be used to group individuals based on one feature at a time. For example, individuals with the same age can be clustered, and within that cluster, individuals with the same sleep condition can be further clustered. This iterative process continues until the desired initial small group size is achieved. Thirdly, the Gower Distance can be calculated, and the distance values can be used as inputs for clustering algorithms to perform further clustering. If the initial cluster results are not satisfactory, additional clustering methods can be applied until more desirable clusters are obtained. Alternatively, if other clustering algorithms are not desired, the data variables can be adjusted based on different priorities and then clustered at different levels using the same algorithm. By considering these different algorithms and approaches, we can explore various strategies to achieve better matching and clustering outcomes for the project. Fourthly, since the dataset contains more categorical data type, it can be converted to 0 and 1 array to present as the input. This could reference by the concept of feature extension. So, for the same data point under the same features, it could mark 1 to represent, otherwise mark 0. For example, if the data point interest is Reading, then under Reading column it will be marked as 1, if the data point interest is Listening to music, then under Reading column it will be marked as 0. Then the dataset will contain only 0 and 1. Next is to apply algorithms using the adjusted dataset.

1. Data Description
   1. Methodology:

Due to the unavailability of public resources for this study, the data mainly relies on restoring data from reports and data files on the elderly and making reasonable inferences and estimations based on basic information and relevant data from the report. The main report has been used is the Aging and Chronic Diseases Report from Public Health Agency of Canada (Public Health Agency of Canada, 2020) and 2011 Federal Disability Report (Health Canada, 2011). The specific data generation process is as follows:

1. Quoting Data from the Report: Directly quoting data from the report was the initial step. This includes data such as the top ten diseases and their prevalence among the elderly.
2. Making Reasonable Guesses: For data with unknown references, moderate and reasonable guesses were made. For example, the report mentioned that elderly individuals in Canada predominantly reside in densely populated areas, i.e., urban areas, and that there is a higher percentage of women (81%) compared to men (77.6%). Although the male-to-female ratio cannot be confirmed as the ratio of elderly individuals living in urban areas (since the sum of the data does not equal one), it can be reasonably inferred that the 81% of females and 77.6% of males mentioned in the report refer to the numerical ratio of the elderly living in rural areas. Therefore, when restoring the original dataset, 81% of females were assigned to live in urban areas, while the remaining 19% lived in rural areas. For males, 77.6% lived in urban areas, while the remaining 22.4% lived in rural areas.
3. Predicting and Filling Missing Data: Predictions and fillings were made based on unmentioned but predictable data. For example, the report mentioned that about 46.8% of the elderly have occasional difficulty falling asleep, with 54.3% being female and 38.8% being male. However, there was still a 6.9% gap in the data. To simplify the data design, only gender was considered as a variable. Therefore, for the 6.9% missing data, proportional split filling was performed during data restoration. This means that 6.9% was allocated as 5% for women and 1.9% for men based on the ratio of male-to-female values provided. Thus, when restoring the dataset, the values of 54.3% and 38.8% mentioned in the report were not directly used, but the values of 59.3% and 40.7% were used instead.

By following these steps, the dataset was generated, which aims to simulate the actual situation of the elderly based on available data, inferences, and reasonable estimations.

After collecting the relevant data values, the project utilized Python and Excel tools to generate the final raw dataset and perform data cleaning procedures. During the data cleaning process, certain delimiters were present in the strings of the original dataset, which required attention. With the proficiency of Python tools and Excel, modifications were made to ensure that the generated dataset accurately reflected the impact of age and gender on disability data. Specifically, in the disability column, instead of simply indicating "Yes" or "No," the generated dataset included descriptive text such as "women aged 75 to 84 with disabilities." To address this, Excel was employed to remove the delimiters and eliminate descriptive words that should not appear. In doing so, the relevant entries were appropriately changed to "Yes" to maintain consistency. The resulting cleaned data was then stored in an Excel document. This data cleaning phase, involving the use of Python and Excel, was essential to ensure the accuracy and reliability of the dataset for further analysis and processing. One thing to mention is that the disability is generated from the 2011 Federal Disability Report published by Health Canada (Health Canada, 2011).

The dataset consists of a total of 13 columns, each containing specific information about the seniors. Here is a description of each column:

1. Gender: This column indicates the gender of the seniors, with two options - female and male.
2. Age: The age column represents the age of the seniors.
3. Ethnicity: This column categorizes seniors into different ethnicities, with eight categories and corresponding percentages assigned to each ethnicity (e.g., 86.29% white).
4. Interests: The interests column includes 16 different interests that have been generated using an AI tool (e.g., gardening, knitting, reading). The distribution of interests follows a normal distribution pattern in the final version of the dataset.
5. Health Issues: This column lists 10 different health issues along with their corresponding percentages, which have been provided by the Canada Health Report (e.g., Hypertension - 65.7%, Periodontal Disease - 52%).
6. Residence: The residency column indicates the ratio of where seniors usually live, with two options - cities or rural areas.
7. Housing Condition: This column represents whether seniors live in their own house or in a different setting, such as a nursing home.
8. Living Alone: The living alone column indicates whether seniors live by themselves or with others.
9. Sleep Hours: This column represents the average sleep duration of seniors. It categorizes the data into three groups: exceeding the recommended sleep hours (7.3 hours), meeting the recommended sleep hours, or having less sleep than recommended.
10. Sleep Quality: This column describes the sleep quality of seniors and includes three variables: occasional difficulty falling asleep, difficulty staying awake during the day, and no difficulties in falling asleep.
11. Social Condition: The social condition column indicates whether seniors feel lonely or not. It includes three variables: feeling lonely, occasionally feeling lonely, and not feeling lonely.
12. Marital Status: This column represents the marital status of seniors and includes four variables: married or living together, widowed, separated, or divorced, and single or never married.

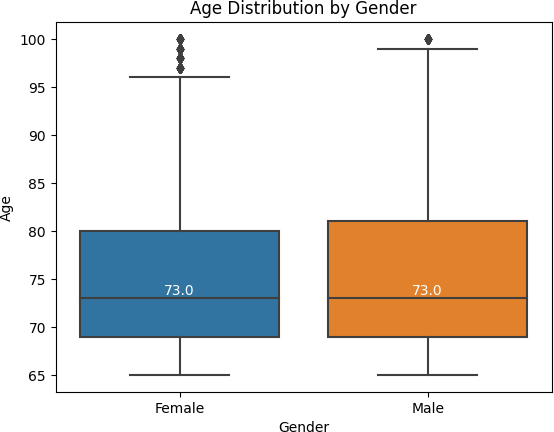


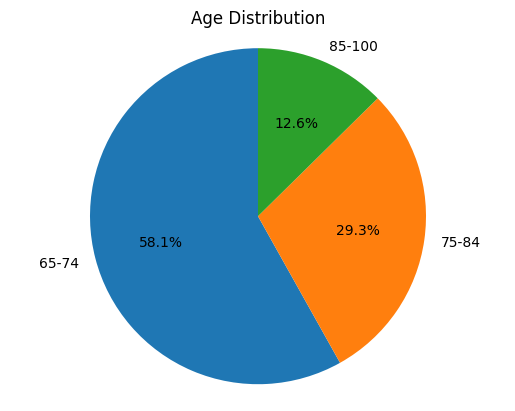
Fig. 2. EDA Analysis - Age Distribution by Gender

1. Disability: The disability column indicates whether seniors have a disability or not.

These columns provide comprehensive information about the seniors, covering demographics, interests, health conditions, living arrangements, sleep patterns, social well-being, and marital status.

* 1. EDA Analysis:

Fig. 1. EDA Analysis - Age Distribution



The dataset contains 13 columns with Age as the only numerical values and others as categorical values. The average age for this dataset is 75.4 and there are more people from the age group 65 to 74 which occupy 58.1%, the other 29.3% are aged from 75 to 84 and the rest 12.6% are aged from 85 to 100, which 100 is the maximum age. The age feature for females exhibits a higher number of outliers compared to males. This finding suggests that females may have a higher likelihood of living longer than males. The median age for both men and women in the dataset is 73 years. This indicates that the middle value of the age distribution for both genders is around 73. Among all the individuals in the dataset, approximately 54.96% are female, while 45.04% are male. It is observed that most of the age outliers in the dataset are attributed to individuals aged above 95. This could be because there are relatively fewer individuals in the dataset who are older than 95, leading the system to consider them as outliers based on the distribution of the age feature. The third quartile (Q3) of the age distribution for males is higher than that of females. This implies that, within this dataset, the higher age groups have a greater representation of males.

The dataset shows a higher proportion of senior individuals belonging to the white race (86.34%). This could be indicative of Canada being a predominantly white country, as reflected in the demographic composition of the dataset. In addition to the white race, there is a notable representation of senior individuals from Asia in the dataset. This suggests a significant presence of seniors with Asian backgrounds in Canada. Most seniors in Canada experience hypertension, with a prevalence rate of 22.1%. The main health issues among seniors include hypertension, periorbital disease, and osteoarthritis. Together, these three conditions account for more than half of the reported health issues among seniors.

Generally, seniors tend to prefer living in urban areas (78.9%) rather than rural areas (21.1%). What’s more, seniors typically reside in private residences (92%) rather than nursing homes or other similar institutions (8%) for the elderly. Among the elderly, more than a quarter of people are living alone. The same parameter has also been found for elderly who occasionally feel lonely (29.9%), in fact, combining people who feel lonely often (16.5%), it is almost half (53.6%) of the elderly will feel lonely.

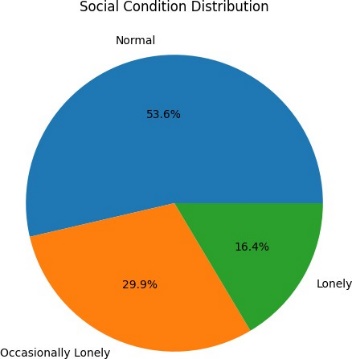


Fig. 3. EDA Analysis – Social Condition Distribution

* 1. Limitations:

Since the report used statistical data about the elderly in Canada only, the results of this data study cannot represent the situation of the elderly worldwide. Besides, not every country has the same elderly situations, and the focus points when the related department conducts research may vary. Therefore, the variables in this dataset may not encompass all elderly characteristics. What’s more, some data contains biased data, for example, there are too many white races in the Ethnicity column, and this could lead to an overlapping dataset which result in lower accuracy score.

1. Prediction Model
   1. Gower Distance Input

The first attempted algorithm involves calculating the Gower Distance and using the resulting values as inputs for all potential algorithms to cluster the data points.

To improve the accuracy and obtain more precise results, the Health Issues column will undergo feature extension before initiating the clustering process. This extension aims to incorporate additional features related to health issues for each data point.

Before calculating the Gower Distance result, it is necessary to convert the dataset variables to dummy variables since most of them are in string format. All variables, except for Age, have been dummied. Once the dummy variables have been created using the 'get\_dummy' process, the next step is to calculate the 'Dice Distance' for each data point. This calculation can be performed using the 'DistanceMetric' package. For the Age column, which is already in numerical format, the 'Manhattan Distance' should be calculated. After that, it should be normalized by dividing it by the highest value, which is 1 (Kesh, 2020). The final step is to calculate the Gower Distance. To calculate the Gower Distance, the weights of the variables for the distance calculation are also required. For simplicity, let's assume that all variables are equally weighted, so each variable will have a weight of 1. The Gower Distance can be obtained by summing the distance values of all variables and then dividing the shape of dataset.

The next step is to apply the Gower Distance that was calculated earlier to the DBSCAN algorithm. When setting the parameters for the algorithm, considering the size of the dataset, the "min\_samples" parameter has been set to 100, indicating that each group should have a minimum of 100 data points. The "eps" parameter has been tested with different values, such as 0.3 and 0.5, to find the best result. Different values of "eps" have shown variations in the DBSCAN clustering results. After clustering, the cluster labels will be added to the original dataset as numeric values. Data points with the same numeric label will be grouped together in a cluster. If a data point has a numeric label of -1, it indicates that it is considered an outlier, meaning that it does not belong to any suitable cluster (Lasaosa, 2021).

If the clustering results are not satisfactory, such as having an uneven distribution or too many outliers, additional steps can be taken to improve the clustering. For example, if there are more than 2000 data points classified as outliers or if a single group contains many data points, it may be necessary to extract the outliers and the data points from that group from the original dataset. The previous steps can then be repeated, including calculating the Gower Distance and applying the DBSCAN algorithm, to obtain new clustering results. The iteration process continues until the clustering results appear to be more organized and meet the desired criteria. Typically, a satisfactory level is reached when the number of outliers is less than 500 data points and there is only one group with approximately 100 clustered data points. By iteratively refining the clustering process and adjusting the thresholds for outliers and group size, it is possible to achieve a more desirable and meaningful clustering outcome.

Other clustering algorithms that can be considered in addition to DBSCAN are K- means, mini batch K-means, GMM and Hierarchy clustering. Unlike DBSCNE, those algorithms require determining the number of clusters in advance. The optimal number of clusters can be determined through Elbow method or by utilizing the Silhouette Coefficient through manually test the score with different number of clusters. Considering with the dataset size (10,000 rows), the number of clusters will be limited in range from 100 to 1,000. To visualize the algorithms input datasets, dimension reduce will be processed and to use the t-SNE technique to achieve this goal.

When setting parameters for each of the algorithms, since they are similar algorithms, so the basic parameters will consider using the same. For k-means, the parameter n\_init will be set as “auto”. For mini batch k-means, it needs to set batch\_size and max\_iter parameters. Considering about the dataset size, the batch\_size has been set as 1000 and the max\_iter has been set as 100. The n\_init parameters has been set to “auto” which is newly added in the sklearn version 1.2 (Sklearn, 2023). For GMM, covariance\_type has been set as “full” and other parameters are not set but following the default setting. For hierarchical clustering, the affinity metric will be set as 'euclidean', and the linkage metric will be set as 'ward'.

* 1. 0&1 Dataset Input

Instead of using the Gower Distance as the input, another approach is to adjust the original dataset by converting it into a 0 and 1 array using the feature extension method. This method involves representing the presence or absence of a specific variable with 1 or 0, respectively. The get\_dummies function can be utilized to achieve this for most of the columns in the dataset. However, for the Age, Interests, and Health Issues columns, a slightly different approach is taken.

For the Age column, it is divided into different age ranges, and for each data point, it is checked to which age range it belongs. If the age matches the age range, it is represented by 1, otherwise, it is represented by 0. For the Interests column, a list of seven predefined interests is provided. The original Interest column is examined, and if a data point corresponds to any of the listed interests, it is considered a match and represented by 1. If it does not match any of the listed interests, it is categorized as "others" and assigned a value of 0 to complete the feature extension process. Regarding the Health Issues column, it is first split into different new columns based on the presence of each health issue. The original Health Issues column is then removed, and the newly created columns undergo the feature extension process. Once the dataset has been transformed into the desired 0 and 1 array format, it can be applied to clustering algorithms to continue the clustering process.

* 1. 0&1 Dataset selection Input

To increase the accuracy of algorithms, the dataset could select columns with higher variance in case the dataset contains biased data and does not have good characteristics to represent. The columns with the top 30% of variances could be selected and applied to all the algorithms. More data sources could also be added to the original dataset to improve the feature selection efficiency.

* 1. LHS Input

To test the accuracy of algorithms, this project has explored various methods to assess their effectiveness. In addition to using dataset similarity and transforming the dataset into a binary format, the project also incorporates the Latin Hypercube Sampling (LHS) technique. In this section, a new dataset will be generated using LHS based on the original dataset consisting of 0s and 1s. This newly created dataset will then be utilized in the k-means algorithm. The objective of this approach is to evaluate the silhouette score, and the clustering results will not be presented for convenience. The analysis will consider different numbers of clusters, specifically 100, 300, 500, and 1000. The silhouette score of the original dataset will also be included for comparison purposes.

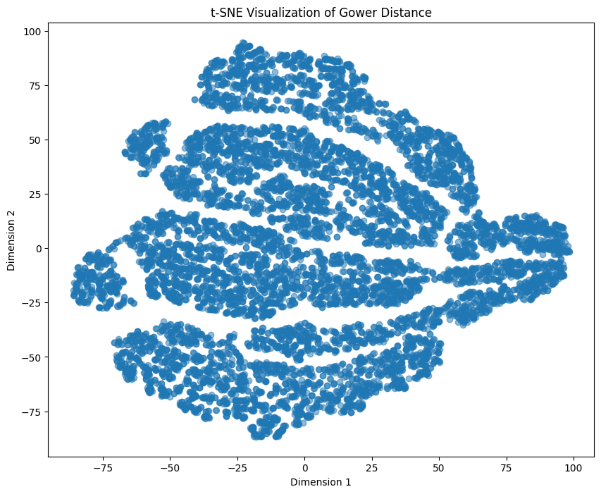


Fig. 5. Gower Distance Dimension Reduction Visualization

1. Experimental Results

Some rules have been considered before applying clustering algorithms. Since the k is limited from 100 to 1,000, it is possible to have the silhouette score increased when k also increases and never shows the highest score. In this case, the silhouette score when k = 1,000 will be considered since it represents the highest score. Additionally, the situations where k = 300 for all algorithms will also be considered for comparison purposes.

* 1. Gower Distance Input
     1. Gower Distance Result:



Fig. 4. Gower Distance Result

* + 1. T-SNE:

To visualize the clusters, it is necessary to do the dimension reduction first. The method involves utilizing the t-SNE method. By reducing the dimensionality of the Gower Distribution, a scatter plot has been generated, as depicted below.

* + 1. DBSCNE

Unfortunately, it appears that the DBSCAN algorithm with the provided parameters (eps and min\_samples) and the feature extension for the Health Issues column did not yield satisfactory results. Despite attempting different values of eps (0.2, 0.3, 0.5, 0.8) and min\_samples (20, 50, 100, 200), all data points were labeled as 0 which represents all the data points clustered as one group.

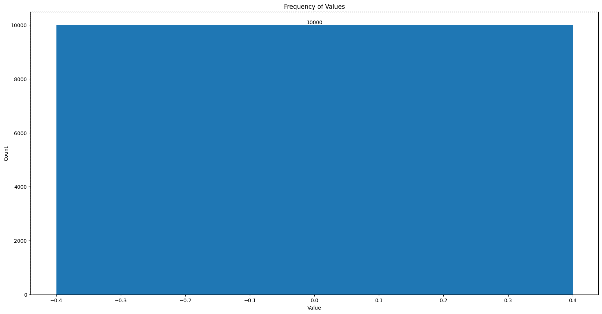


Fig. 6. DBSCNE Clustering Result

* + 1. K-means

When applying the elbow method to find the best k, the method keeps running instead of giving a result. As a result, the silhouette scores have been manually tested with different cluster numbers (k), and the details can be checked in the table below. Within the limited k range, the silhouette score keeps increasing as ‘k’ increases.

Table 1

K-means Silhouette Scores results for Gower Distance

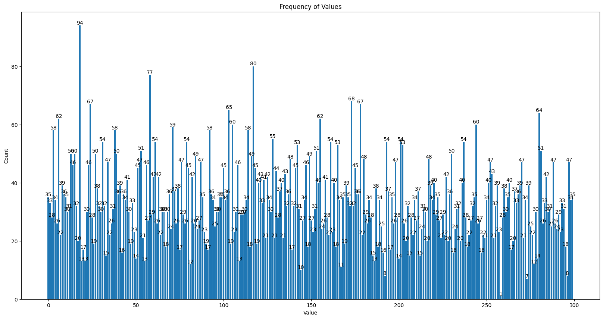


Fig. 8. Mini Batch K-means clustering result when having 300 clusters

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.081 |
| 200 | 0.082 |
| 300 | 0.082 |
| 500 | 0.087 |
| 800 | 0.094 |
| 1000 | 0.098 |

Among the 300 clusters, the minimum number of data points in one cluster is 14 and the maximum number of data points in one cluster is 77. The average number of data points is 33.3. The silhouette score for k = 300 is 0.082.

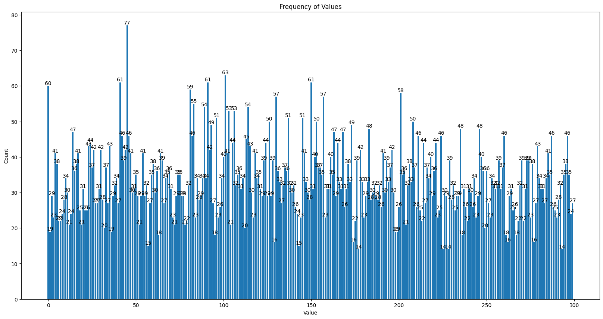


Fig. 7. K-means clustering result when having 300 clusters

* + 1. Mini Batch K-means

Like K-means, Mini-Batch K-means also didn't yield a result when applying the elbow method to find the best k. Therefore, the silhouette scores have been manually tested with different cluster numbers (k), and the details can be checked in the table below. The highest silhouette score achieved is 0.074 when k = 300.

Table 2

Mini Batch K-means Silhouette Scores results for Gower Distance

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.072 |
| 200 | 0.074 |
| 300 | 0.074 |
| 500 | 0.070 |
| 800 | 0.066 |

Among the 300 clusters, the minimum number of data points in one cluster is 1 and the maximum number of data points in one cluster is 94. The average number of data points is 33.3. The silhouette score for k = 300 is 0.074.

* + 1. GMM

When apply Gower Distance to GMM algorithm, the system breakdown happened and processing time prolonged.

* + 1. Hierarchy

When apply Gower Distance to GMM algorithm, the system breakdown happened and processing time prolonged.

* + 1. Conclusion

Only K-means and Mini-Batch K-means have shown the cluster results when applying Gower Distance to the clustering algorithms. K-means outperforms Mini-Batch K-means, as it achieves a higher silhouette score compared to Mini-Batch K-means when the number of clusters is the same. For example, when k = 300, K-means demonstrates a higher silhouette score than Mini-Batch K-means.

* 1. Adjusting Dataset Input:
     1. Adjusted Dataset:

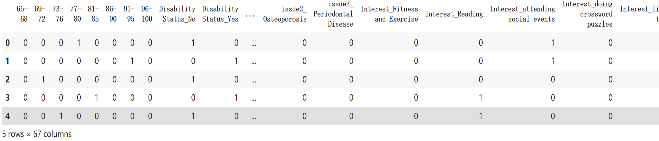


Fig. 9. Adjusted Dataset result

The dataset column number has extended to 67 columns after the adjustment.

* + 1. T-SNE:

By reducing the dimensionality of the 0&1 array dataset, a scatter plot has been generated, as depicted below.

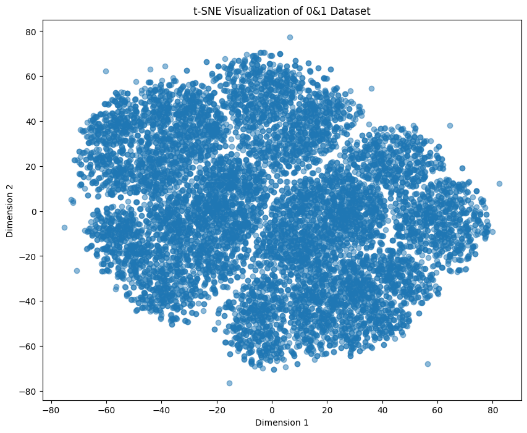


Fig. 10. Adjusted Dataset Dimension Reduction Result

* + 1. DBSCNE

Unfortunately, even with the changed 0 & 1 dataset, this algorithm labeled all the data points as outliers with labeled -1.

* + 1. K-means

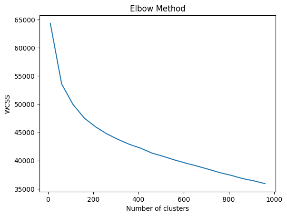
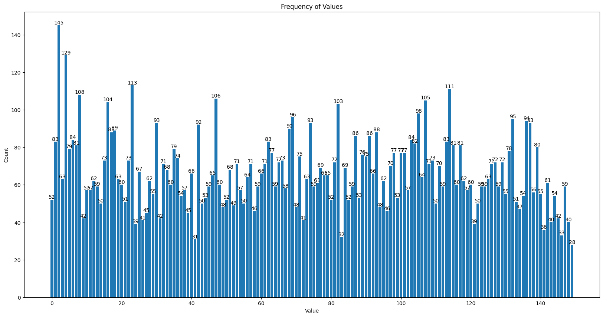
The Elbow method has been applied to K-means as well as the silhouette score with testing different k. The best silhouette score is 0.043 when k = 150. The minimum number of data points in one cluster is 28 and the maximum number of data points in one cluster is 145. The average number of data points is 66.7.

Table 3

K-means Silhouette Scores results for 0&1 Dataset.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.043 |
| 150 | 0.043 |
| 200 | 0.041 |
| 300 | 0.040 |

Fig. 11. K-means elbow method and best silhouette score cluster result for 0&1 dataset.



* + 1. Mini Batch K-means

The elbow method has been applied to Mini Batch K-means as well as the silhouette score with testing different k. The best silhouette score is 0.038 when k = 100. The minimum number of data points in one cluster is 28 and the maximum number of data points in one cluster is 145. The average number of data points is 100.

Table 4

Mini Batch K-means Silhouette Scores results for 0&1 Dataset.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.038 |
| 150 | 0.033 |
| 200 | 0.026 |
| 300 | 0.025 |

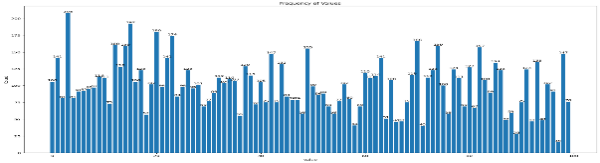
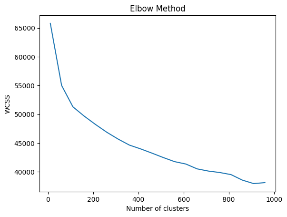


Fig. 12. Mini Batch K-means elbow method and best silhouette score cluster result for 0&1 dataset.

* + 1. GMM

The elbow method has not been applied to GMM but only testing silhouette score with different k manually. The best silhouette score is 0.04 when k = 200. The minimum number of data points in one cluster is 19 and the maximum number of data points in one cluster is 109. The average number of data points is 50.

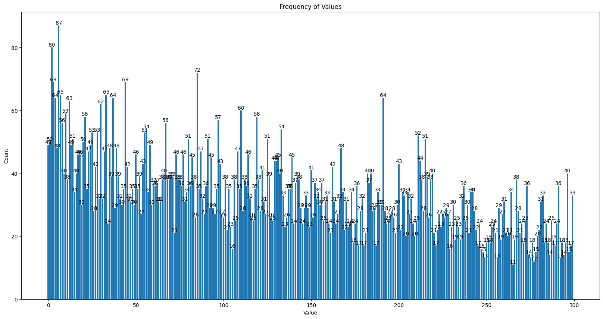


Fig. 14. Hierarchy Silhouette Scores with different cluster number for 0&1 Dataset.

Fig. 13. GMM Silhouette Scores with different cluster number for 0&1 Dataset.

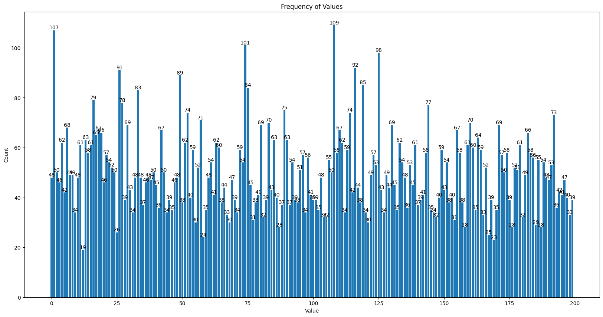


Table 5

GMM Silhouette Scores results for 0&1 Dataset.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.036 |
| 150 | 0.040 |
| 200 | 0.040 |
| 300 | 0.037 |

* + 1. Hierarchy

The elbow method has not been applied to Hierarchical clustering. Instead, the silhouette score has been manually tested with different values of k. Within the limited range of k, the silhouette score keeps increasing as ‘k’ increases. Among the 300 clusters, the minimum number of data points in one cluster is 11 and the maximum number of data points in one cluster is 87. The average number of data points is 33.3. The silhouette score for k = 300 is 0.01.

Table 6

|  |  |
| --- | --- |
| k | Silhouette Score |
| 150 | 0.002 |
| 200 | 0.004 |
| 300 | 0.010 |
| 400 | 0.016 |
| 500 | 0.019 |
| 800 | 0.033 |
| 1000 | 0.041 |

Hierarchy Silhouette Scores results for 0&1 Dataset.

* + 1. Conclusion

When applying the dataset that has been converted to a binary (0 & 1) array, all the algorithms have shown cluster results. Among the algorithms, K-means performs the best as it achieves the highest silhouette score. Furthermore, when the number of clusters is the same, for example, when k = 300, K-means also has the highest silhouette score.

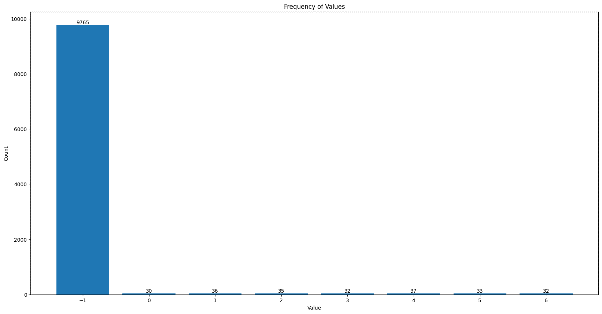


Fig. 16. DBSCNE clustering result for selected 0&1 dataset

* 1. 0&1 array dataset with selected columns

All the potential testing procedures mentioned above are not showing the expected high silhouette score. One possible explanation for this is that the dataset itself is biased. For example, if the 'race' attribute contains predominantly 'white' race, it will result in every data point being the same. Consequently, the data points overlap with each other, leading to a lower silhouette score. To address this issue, feature selection has been considered. The approach taken is to calculate the variance for each column across the entire dataset. Then, the columns with the top 30% variance among all the columns are selected. There are a total of 20 columns that have been selected and subsequently, the algorithms are applied to the modified dataset.

* + 1. T-SNE:

By reducing the dimensionality of the 0&1 array selected dataset, a scatter plot has been generated, as depicted below.

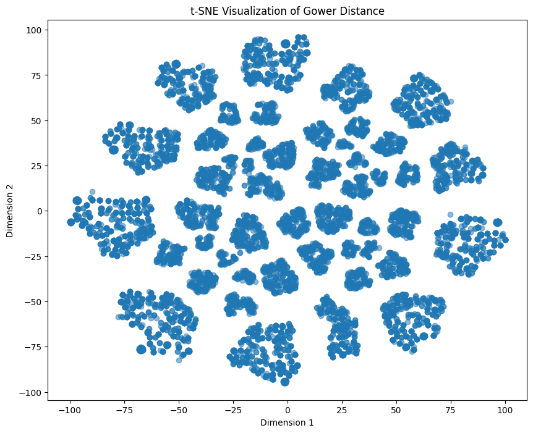


Fig. 15. Selected columns Dimension Reduction Result

* + 1. DBSCNE

Even though DBSCAN has clustered some data points, there are still mainly 97% of data points that have been labeled as outliers. Even after attempting to sub-cluster the outliers, they continue to be identified as outliers.

* + 1. K-Means

With k increase, the silhouette score also increased. Among the 300 clusters, the minimum number of data points in one cluster is 10 and the maximum number of data points in one cluster is 98. The average number of data points is 33.3. The silhouette score for k = 300 is 0.218.

Table 7

K-means Silhouette Scores results for 0&1 Dataset selected.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.168 |
| 200 | 0.195 |
| 300 | 0.218 |
| 500 | 0.275 |
| 800 | 0.363 |
| 1000 | 0.416 |

* + 1. Mini Batch K-means

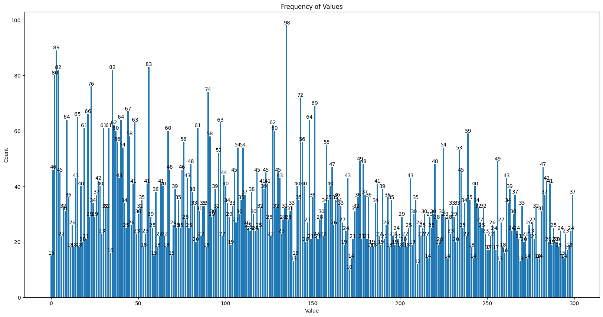
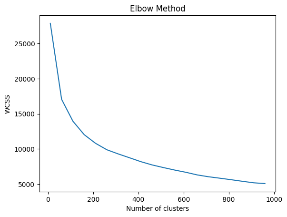


Fig. 17. K-means Silhouette Scores with different cluster number for selected 0&1 dataset.

With k increase, the silhouette score also increased. Among the 300 clusters, the minimum number of data points in one cluster is 4 and the maximum number of data points in one cluster is 95. The average number of data points is 33.3. The silhouette score for k = 300 is 0.215.

Table 8

Mini Batch K-means Silhouette Scores results for 0&1 Dataset selected.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.158 |
| 200 | 0.193 |
| 300 | 0.215 |
| 500 | 0.260 |
| 800 | 0.336 |
| 1000 | 0.367 |

* + 1. GMM

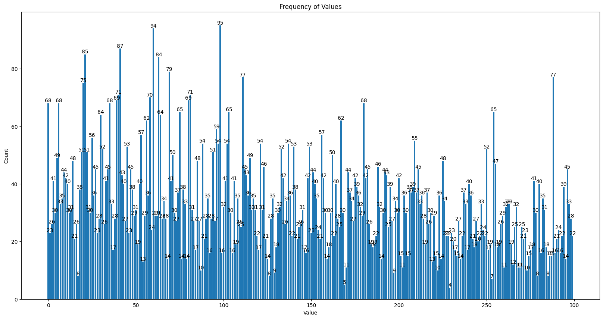
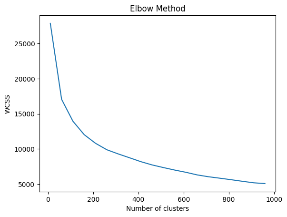


Fig. 18. Mini Batch K-means Silhouette Scores with different cluster number for selected 0&1 dataset.

With k increase, the silhouette score also increased. Among the 300 clusters, the minimum number of data points in one cluster is 2 and the maximum number of data points in one cluster is 113. The average number of data points is 33.3. The silhouette score for k = 300 is 0.211.

Table 9

GMM Silhouette Scores results for 0&1 Dataset selected.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.167 |
| 200 | 0.186 |
| 300 | 0.211 |
| 500 | 0.265 |
| 800 | 0.354 |
| 1000 | 0.409 |

* + 1. Hierarchy

With k increase, the silhouette score also increased. Among the 300 clusters, the minimum number of data points in one cluster is 14 and the maximum number of data points in one cluster is 69. The average number of data points is 33.3. The silhouette score for k = 300 is 0.213.

Table 10

Hierarchy Silhouette Scores results for 0&1 Dataset selected.

|  |  |
| --- | --- |
| k | Silhouette Score |
| 100 | 0.166 |
| 200 | 0.185 |
| 300 | 0.213 |
| 500 | 0.268 |
| 800 | 0.364 |
| 1000 | 0.421 |

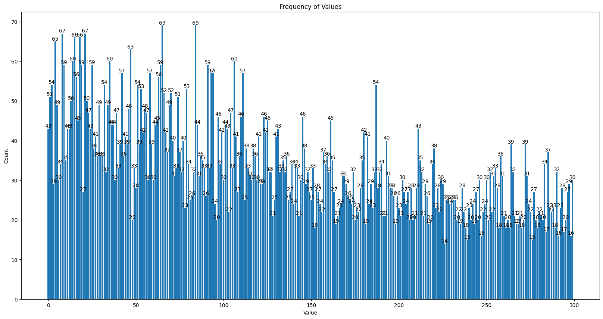


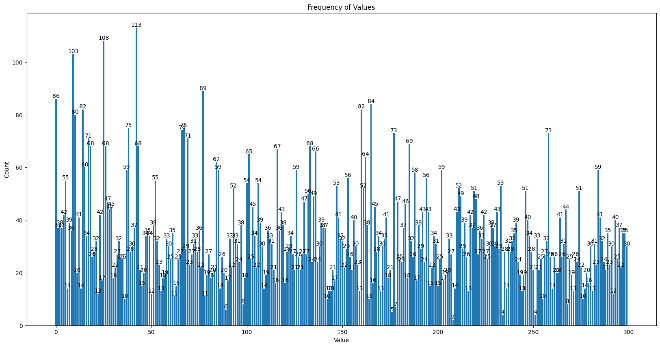
Fig. 20. Hierarchy Silhouette Scores with different cluster number for selected 0&1 dataset.

* + 1. Conclusion

When applying the selected 0&1 array dataset, all the algorithms have shown cluster results, and the silhouette score has increased as expected. As ‘k’ increases, the silhouette score also increases. Therefore, for comparison under equal conditions, k equal to 300 has been selected as an example for all algorithms. Among all the algorithms, K-means achieves better results regardless of whether k is smaller or larger. Additionally, when k is large, the Hierarchy clustering algorithm also performs better.

1. Conclusion

Fig. 19. GMM Silhouette Scores with different cluster number for selected 0&1 dataset.



This project aims to adapt clustering algorithms to achieve the goal of grouping people. Initially, real-world authorized reports were used to generate a usable dataset. Subsequently, different clustering algorithms were applied to cluster individuals into distinct groups using this dataset. And Silhouette Score has been applied to test different algorithms’ accuracy and through comparing with the score, which algorithm has doing better could be decided. To visualize the cluster results, a dimension reduction method was employed, incorporating the t-SNE technique. This technique effectively reduces the dimensionality of the data, allowing for visualization of the clustered data points.

* 1. Key steps and cluster results

Prior to applying any algorithms, the Gower Distance was calculated and used as input for potential clustering algorithms. Furthermore, the original dataset was transformed into a binary (0 and 1) array and utilized as input for clustering algorithms as well. Additionally, the columns with higher variances (top 30% among all columns) were filtered and applied to the algorithms.

For the Gower Distance, initially, the DBSCAN algorithm was attempted but failed to effectively cluster the data, resulting in every data point being assigned to a single cluster. Subsequently, the K-means, Mini Batch K-means, Gaussian Mixture Models (GMM), and Hierarchical clustering algorithms were directly applied to the Gower Distance. However, when applying GMM and Hierarchical clustering, they took longer than expected and did not yield any results. On the other hand, both K-means and Mini Batch K-means successfully clustered the data points into distinct groups. In the case of the K-means algorithm using the Gower Distance, within the range of k values from 100 to 1,000, the silhouette score increased as ‘k’ increased. When k was set to 300, the minimum number of data points in a single cluster was 14, the maximum number was 77, and the corresponding silhouette score was 0.082. In contrast, for the Mini Batch K-means algorithm, the highest silhouette score achieved was 0.074 when k was set to 300. In this case, the minimum number of data points in a single cluster was 1, while the maximum number was 94.

Regarding the 0 and 1 array dataset, the DBSCAN algorithm still failed to produce the desired outcome, labeling every data point as outliers. For the K-means algorithm using the 0 and 1 dataset, the best silhouette score achieved was 0.043 when k was set to 150. Within a single cluster, the minimum number of data points was 28, and the maximum number was 145. In the case of the Mini Batch K-means algorithm, the best silhouette score obtained was 0.038 when k was set to 100. The minimum number of data points in a single cluster was 16, while the maximum number was 208. Interestingly, the GMM algorithm successfully clustered the data using this input, resulting in a single cluster. The best silhouette score achieved was 0.04 when k was set to 200. The minimum number of data points in a single cluster was 19, and the maximum number was 109. Similarly, the Hierarchy algorithm successfully clustered the data using this input, resulting in a single cluster. Within the k range from 100 to 1,000, the silhouette score increased as ‘k’ increased. When k was set to 300, the minimum number of data points in a single cluster was 11, the maximum number was 87, and the silhouette score was 0.01.

Regarding the selected dataset transformed into a 0 and 1 array, the silhouette score for all the algorithms increases as k increases within the specific range. However, when compared to the best silhouette scores achieved with other inputs, the increase is close to 0.5, which is significant enough to demonstrate the efficiency. Generally, K-means is doing better than other algorithms no matter with smaller k or larger k.

* 1. Silhouette Score Tables

The two tables have shown the merged silhouette scores results when k = 300 and when the algorithms with the best silhouette score.

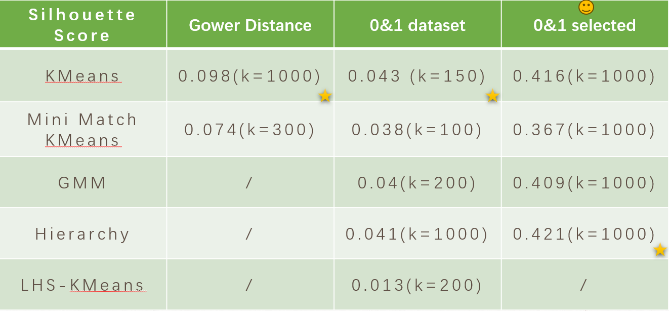


Fig. 22. Best Silhouette Scores for all algorithms.

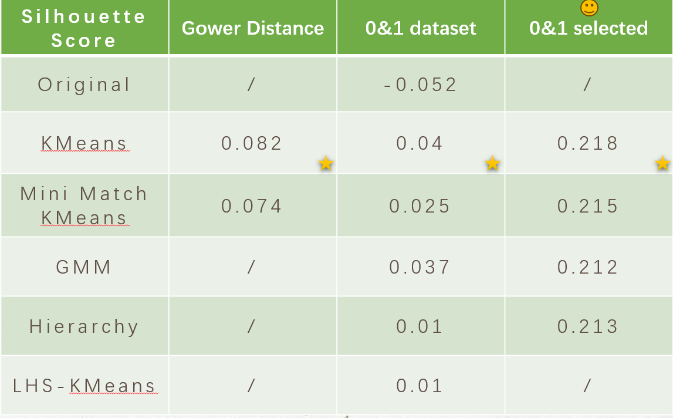


Fig. 21. Silhouette Scores for all algorithms when k = 300.

* 1. Future steps

When applying the algorithms and calculating the silhouette scores, the only situation that yields better scores as expected is when selecting columns with higher variance. Therefore, the next step to improving accuracy could be adding more data to the original dataset. However, it is important to note that the quality of the data should be considered. Data with lower variance may be meaningless and not contribute effectively to the analysis.

What's more, in this project focused on clustering data points, we have successfully identified and grouped elderly individuals into clusters, along with the number of data points in each cluster. However, we have not determined the specific reasons why they were clustered together. Although clustering is based on similarities, we have not explored the original context of the data to identify the exact similarities. For example, while we know that elderly ID 100 and elderly ID 360 have been clustered together, we have not identified the specific similarities between them. This is an area for potential future projects to investigate the original dataset and determine the precise similarities that contributed to the clustering results. Achieving this could be done through the utilization of classification algorithms.

* 1. Potential applications

For potential applications, the results of this project can be utilized in other clustering systems, such as roommate matching or mentor-assistant matching. For example, universities often have initiatives to connect international students with local students who can help in adapting to the new culture, environment, and even language practice. The clustering system developed in this project can be employed to help institutions identify suitable matches between international students and local helpers. This would facilitate a smoother transition and create a more comfortable school life for international students who are studying away from their home country.

References

1. ANDERSON, A. (2023, 1 29). MUO. Retrieved from MUO: https://www.makeuseof.com/swipe-based-vs-algorithm-based-dating-apps/
2. Aprilliant, A. (2021, 1 17). The k-prototype as Clustering Algorithm for Mixed Data Type (Categorical and Numerical). Retrieved from Medium: https://towardsdatascience.com/the-k-prototype-as-clustering-algorithm-for-mixed-data-type-categorical-and-numerical-fe7c50538ebb
3. avcontentteam. (2023, 4 26). PCA | What Is Principal Component Analysis & How It Works? (Updated 2023). Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2016/03/pca-practical-guide-principal-component-analysis-python/
4. Awa-Abuon, J. (2022, 5 18). What Is OkCupid: The Dating App Explained. Retrieved from Trending: https://www.makeuseof.com/what-is-okcupid-dating-app/?newsletter\_popup=1
5. Bhardwaj, A. (2020, 5 26). Silhouette Coefficient. Retrieved from Medium: https://towardsdatascience.com/silhouette-coefficient-validating-clustering-techniques-e976bb81d10c
6. Carrasco, O. C. (2019, 6 3). Gaussian Mixture Models Explained. Retrieved from Medium: https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95
7. CR, A. (2023, 4 25). MLOps Blog. Retrieved from neptune.ai: https://neptune.ai/blog/clustering-algorithms
8. Data&Tech. (2023). RewireMag. Retrieved from RewireMag: https://rewire.ie.edu/dating-apps-darkest-secret-algorithm/
9. DataTechNotes. (2020, 11 26). TSNE Visualization Example in Python. Retrieved from DataTechNotes: https://www.datatechnotes.com/2020/11/tsne-visualization-example-in-python.html
10. Dobilas, S. (2021, 5 22). GMM: Gaussian Mixture Models — How to Successfully Use It to Cluster Your Data? Retrieved from Medium: https://towardsdatascience.com/gmm-gaussian-mixture-models-how-to-successfully-use-it-to-cluster-your-data-891dc8ac058f
11. Health Canada. (2011). 2011 Federal Disability Report. Retrieved from Government of Canada: https://www.canada.ca/en/employment-social-development/programs/disability/arc/federal-report2011/section1.html#s1c
12. Hill, K. (2015, 10 23). Revealing HomeHero’s Matching Algorithm for Senior Care. Retrieved from Medium: https://medium.com/@kaleazy/revealing-homehero-s-matching-algorithm-for-senior-care-2a376b1a32a5
13. Karabiber, F. (2023). Hierarchical Clustering. Retrieved from LearnDataSci: https://www.learndatasci.com/glossary/hierarchical-clustering/#:~:text=a%20real%20dataset-,What%20is%20Hierarchical%20Clustering%3F,hierarchical%20tree%20called%20a%20dendrogram.
14. Karabiber, F. (2023). Jaccard Similarity. Retrieved from learndatasci: https://www.learndatasci.com/glossary/jaccard-similarity/
15. Kesh, S. (2020, 5 19). How to calculate Gower’s Distance using Python. Retrieved from Medium: https://medium.com/analytics-vidhya/concept-of-gowers-distance-and-it-s-application-using-python-b08cf6139ac2
16. Kharwal, A. (2021, 9 10). Mini-batch K-means Clustering in Machine Learning. Retrieved from Thecleverprogrammer: https://thecleverprogrammer.com/2021/09/10/mini-batch-k-means-clustering-in-machine-learning/
17. Kumar, S. (2021, 5 7). Clustering Algorithm for data with mixed Categorical and Numerical features. Retrieved from Medium: https://towardsdatascience.com/clustering-algorithm-for-data-with-mixed-categorical-and-numerical-features-d4e3a48066a0
18. Kurama, V. (2022, 10 11). Data Science. Retrieved from builtin: https://builtin.com/data-science/collaborative-filtering-recommender-system
19. Lasaosa, J. M. (2021, 5 29). Clustering on numerical and categorical features. Retrieved from Medium: https://towardsdatascience.com/clustering-on-numerical-and-categorical-features-6e0ebcf1cbad
20. M. Venkat Reddy el at. (2017). Divisive Hierarchical Clustering with K-means and Agglomerative Hierarchical Clustering. IJCST, Volume 5 Issue 5.
21. Public Health Agency of Canada. (2020). Aging and Chronic Diseases. Ottawa: Public Health Agency of Canada.
22. Saji, B. (2021, 1 20). Elbow Method for Finding the Optimal Number of Clusters in K-Means. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2021/01/in-depth-intuition-of-k-means-clustering-algorithm-in-machine-learning/
23. Santos, G. (2022, 1 26). A Simple Introduction to Gaussian Mixture Model (GMM). Retrieved from https://towardsdatascience.com/a-simple-introduction-to-gaussian-mixture-model-gmm-f9fe501eef99: https://towardsdatascience.com/a-simple-introduction-to-gaussian-mixture-model-gmm-f9fe501eef99
24. Sarkar, T. (2019, 10 1). Clustering Metrics Better Than the Elbow Method. Retrieved from KD nuggets: https://www.kdnuggets.com/2019/10/clustering-metrics-better-elbow-method.html
25. Scanlon, K. (2020, 8 31). Dating Data: An Overview of the Algorithm. Retrieved from Medium: https://medium.com/swlh/dating-data-an-overview-of-the-algorithm-afb9f0c08e2c
26. Scikit learn. (2023). Selecting the number of clusters with silhouette analysis on KMeans clustering. Retrieved from Scikit learn: https://scikit-learn.org/stable/auto\_examples/cluster/plot\_kmeans\_silhouette\_analysis.html
27. Sharma, S. (2020, 9 8). How to Master the Popular DBSCAN Clustering Algorithm for Machine Learning. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/
28. Sklearn. (2023). mini batch kmeans. Retrieved from Sklearn: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html
29. Together. (2023). together. Retrieved from together: https://www.togetherplatform.com/
30. UneeQ. (2023, 2 22). AI Ethics. Retrieved from UNEEQ: https://www.digitalhumans.com/blog/loneliness-and-the-role-of-conversational-ai-companions
31. University of California. (2016, 2 10). How a matchmaking algorithm saved lives. Retrieved from Medium: https://medium.com/@UofCalifornia/how-a-matchmaking-algorithm-saved-lives-2a65ac448698
32. Zach. (2020, 9 13). What is Latin Hypercube Sampling? Retrieved from Statology: https://www.statology.org/latin-hypercube-sampling/

1. \* Corresponding author. Tel.: +1-365-688-5701; e-mail: tianyi.wu@ontariotechu.net. [↑](#footnote-ref-2)