AN UNSUPERVISED MACHINE LEARNING APPROACH TO DIABETIC NEUROPATHY DATA WITH INTERNAL VIEW COMPARISON AND WEIGHTING

by

Jared McCune

A thesis presented to the Department of Computer Science
and the Graduate School of the University of Central Arkansas
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science

Conway, Arkansas
August 2021

ACKNOWLEDGEMENT

I would like to thank my thesis committee chair, Dr. Olcay Kursun, for guidance and advice throughout my thesis process. He helped further my knowledge into machine learning and his continued support and patience has been invaluable to the thesis work here. He provided the initial dataset used in this thesis, as well as explanations and resources to learn more about the field of diabetic neuropathy.

I am also extremely grateful to the other members of my thesis committee, Dr. Bernard Chen and Dr. Chenyi Hu, whose feedback was instrumental in the work of this thesis. In addition, my committee members have provided me with resources that expanded my knowledge on the topics of unsupervised machine learning and given me the resources to further pursue my interest in the topic.

I would like to extend my gratitude to the entire UCA computer science faculty and staff. My understanding of computer science would be severely lacking if not for them, and my ability to effectively learn new topics and methods within computer science can be entirely attributed to them.

I extend a special thanks to Dr. Bernard Chen and Dr. Emre Celebi, who helped me discover my initial, and continued, interest in data mining and have been instrumental in my understanding of the topics used within this thesis.

Finally, I would like to thank my family and friends. The ones who have listened to and encouraged me, through good times and bad. They give me the strength and courage to succeed, which will never go unappreciated.

VITA

Jared McCune was born in Little Rock, Arkansas on July 20th, 1997. He attended school in the Faulkner County School District and graduated from Greenbrier High School in May 2015. The following August, he enrolled at the University of Central Arkansas and, in May 2019, received the degree of Bachelor of Science in Computer Science. He then enrolled in the University of Central Arkansas graduate program in August 2019 and received a Master of Science degree in Computer Science in August 2021.

ABSTRACT

As medical and technological advancements are made, newer collections of information are made available from more diverse sources. Not only have testing methods become more refined over time, but in some cases multiple tests have been developed to aid in the precision and authenticity of diagnostic processes. However, though more information is made available using multiple tests, there exists the desire to find relevant information and connections between these tests. This study took information collected from four different tests used in the diagnosis of diabetic neuropathy and, using data normalization and unsupervised learning methods, analyzed the collected information to find relevant patterns. Specifically, patterns relating to the grouping of patients based on the similarities that exist within their test results and possible redundancy detection between the tests used were expected.

To test these expectations, the unsupervised method of clustering was performed on a multi-view dataset containing test results from 40 diabetic neuropathy patients. This collected information was cleaned, and, following the data cleaning, normalized utilizing min-max normalization. Afterwards the dataset was placed through the K-Means clustering algorithm, a weighted K-means algorithm based on the validity of the K-means clusters formed in each view, and through agglomerative hierarchical clustering. Through internal validation, the optimal number of clusters for the dataset was found to be two, and the weighted method, after minor alterations to the determined weights were performed, did alter the results of the algorithm. Hierarchical clustering also revealed that in a smaller dataset, such as the diabetic neuropathy dataset, the patterns found in K-means clustering are less apparent when compared on a step-by-step level.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Data Science and Machine Learning

Data Science has evolved considerably over the last 50 years [1]. Many fields now rely on data science to sift through the large amounts of data available to find meaningful information. Many examples of this exist in day-to-day life, such as Amazon utilizing data science to suggest products that would appeal to an individual based on previous views/purchases made by said individual, video streaming services such as Hulu and Netflix showing recommended shows/movies based on what an individual has watched previously, or in video games where multiplayer compatibility between players can be decided based on the skills of all individuals involved. Modern data science utilizes computer technology to automate and manage the data processing involved. Commonality that exists in the above examples as well as in many other applications of data science is the concept of finding similarities in the data. Programs are created to learn these similarities by creating groupings or models to reference when addressing later samples of data. This is machine learning. When referencing machine learning the concept is fairly straightforward: it should create ways for computers to mimic human thought processes. For example: when someone sees a new kind of animal for the first time, they notice distinct features that make that animal different from other animals seen before or features that label the animal based on previously seen animals.

When we look at Figure 1 [2], we see that there are two distinct types of machine learning: Supervised and Unsupervised machine learning. The best way to summarize the difference between these types of machine learning is to acknowledge labels in the information. If one is given a photo album of labeled pictures of cats and dogs, their mind

now has a categorical model that can be used. Later when the animal is seen out on the side of the road or at a friend's house their mind will reference the photo album to decisively label the animal as a cat or dog (assuming it falls within these categories). This is supervised learning. It is supervised due to the fact that a dataset, in this case the photo album, is used to create a model that can be referenced in further samples of data. In contrast, unsupervised machine learning uses unlabeled data to identify patterns. Unsupervised machine learning is especially useful when a data set is available and it is inferred that patterns in the data exist but there exists no current label to corroborate this inference.

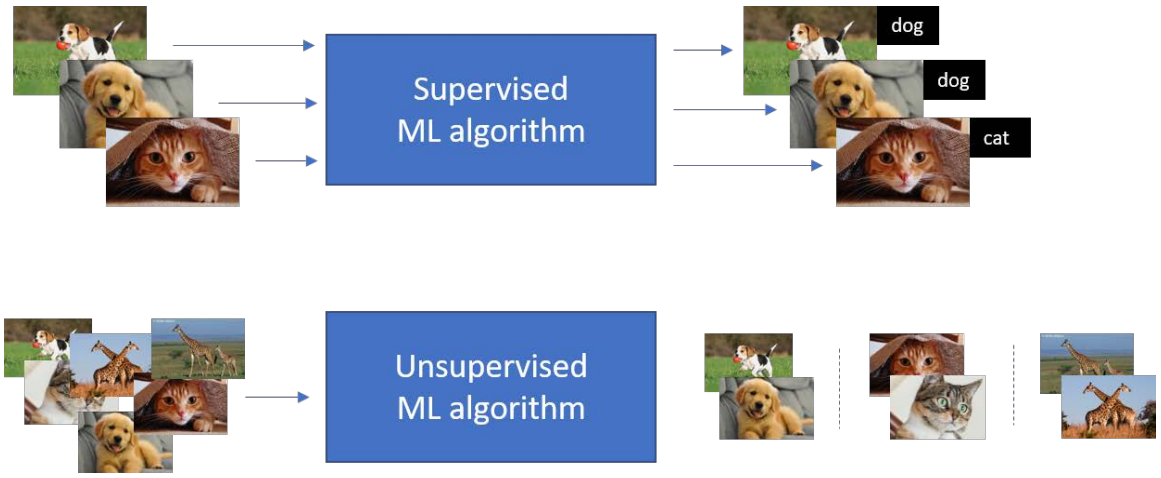

Figure 1: Machine learning applied to animals

1.2 Data Clustering

A common method of unsupervised machine learning is clustering. Clustering is the process of grouping datasets such that the groups, also called clusters, consist of objects where objects within the same cluster are more similar to each other than to objects in other clusters. Looking back to Figure 1, the unsupervised machine learning algorithm shown is an example of clustering. The animals that are grouped together share

more similarities with each other than they share with the other animal groups. Clustering can be used in cases where companies want to find patterns in their customers, or for use in the medical field where patient data is collected and finding possible patterns could assist in the efficiency of diagnostic procedure selection, such as the procedures utilized for the detection of diabetic neuropathy, which is the basis for this thesis.

1.3 Diabetic Neuropathy

Diabetes is a disease that occurs when your blood glucose, also called blood sugar, is too high [3]. Essentially this means that the body can have a built up amount of glucose and not enough insulin to send the glucose to the cells for energy. This can lead to further issues such as cardiovascular disease, skin conditions, hearing impairment, eye damage (retinopathy), kidney damage (nephropathy), or nerve damage (neuropathy). In particular, this thesis focused on diabetic neuropathy in diabetic patients. The dataset utilized was provided by a preliminary work of a more detailed project on the diagnosis and monitoring of diabetic neuropathy [4].

Diabetic neuropathy is a type of nerve damage that can occur in those who have diabetes [5, 6]. The high levels of glucose caused by diabetes can lead to injured nerves throughout the body as shown in Figure 2 [7], resulting in symptoms ranging from numbness and pain in the legs and feet to more severe problems involving the digestive system, blood vessels, heart, or the urinary tract. There exists four main types of diabetic neuropathy: peripheral neuropathy (affecting the arms, legs, feet, and hands), autonomic neuropathy (affecting the autonomic nervous system such as the heart, bladder, stomach, intestines, sex organs, and eyes), proximal neuropathy (affecting the thighs, hips, buttocks, or legs), and mononeuropathy (with two types: cranial and peripheral, which

refer to damage to a specific nerve). Given the varying types and severety of diabetes and, specifically, diabetic neuropathy, there are different tests that can be employed to diagnose the condition.

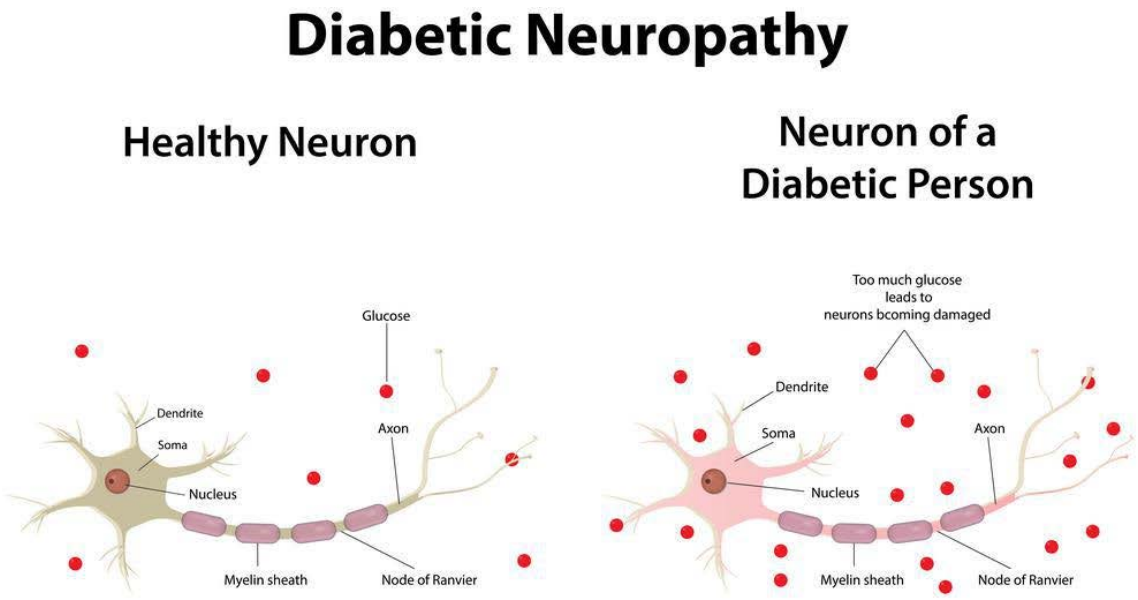

Figure 2: Glucose causes damage to neurons

Blood tests may be done in the general diagnosis of diabetes, with particular focus on the Hb1ac levels (A1C) [8]. The A1C test provides the average level of glucose over the past three months [9]. The values of this type of test are given as a percentage where values between 5.7 and 6.4 indicate prediabetes and values of 6.5 or higher indicate diabetes. Other values in a blood test can provide indication for diabetes such as the triglyceride amount (where high triglycerides can act as an indicator of diabetes or prediabetes). When moving into the diagnosis of diabetic neuropathy, further testing of blood count and glucose levels is utilized. Another test performed in the diagnosis and testing for neuropathy is the Electromyography (EMG) procedure [10]. This procedure assesses the health of muscles and the nerve cells that control them. EMG procedures

measure the responses muscles give to signals provided by the nerves by inserting a needle electrode into the muscle and prompting the patient to contract and rest the muscle. Someone who has neuropathy will not record substantial responses from the muscles as the nerve damage will result in unclear signals being sent from the nerves. EMG tests are performed on the motor and sensory nerves. Another test that can provide a closer idea of the extent of diabetic neuropathy is the Michigan Neuropathy Screening (MNS) [11]. When conducting this test, two different items are used. The first part of the MNS is a self-administered 15 item questionnaire. Following the questionnaire is the Michigan Neuropathy Screening Instrument (MNSI) Examination. This second portion of the MNS is conducted by a physician in a non-invasive procedure. The questionnaire and examination are then scored and assigned numerical values based on the answers given. The final examination utilized in this thesis is the Cortical Metrics (CM4) test [12]. This test is performed by the Brain Gauge, a device that uses the sense of touch to measure brain activity.

Brain Gauge devices, such as those depicted in Figure 3 [13], send vibrations to the two buttons on the device. The device is shaped similarly to a computer mouse where the patient simply places their hand on the device with their fingers placed on the buttons and, based on the type of test being conducted, different levels of vibrations are sent to the buttons for the user to identify. As diabetic neuropathy can cause numbness which can affect reflexes, this test can serve to further identify damage in the hands, arms, or the brain. As stated above, different tests can be conducted to produce the needed results. These tests include Static Threshold Detection Tests with varying thresholds (the user receives a faint vibration to one of the two fingers, followed by the user pressing the

button where the vibration occurred), Double Sided Adaptation Tests, Dynamic Threshold Detection Tests (the user receives a single, very faint vibration to one of the two fingers, then identifies which button the vibration came from), and Reaction Time Tests (the user will receive a single pulse to the right finger which will start a timer, and the user will then click the left button when they feel the vibration which will stop the timer and record the results).

Figure 3: Examples of Brain Gauge devices used

1.4 Goals of This Work

The goal of this thesis was to apply unsupervised machine learning techniques in an effort to identify existing patterns within a small set of diabetic neuropathy data. A secondary goal of this thesis was to possibly identify the necessity of the tests utilized in diabetic neuropathy diagnoses. For example: if the MNS test and CM4 test provide similar grouping results, then it may be asserted that using both tests may be unnecessary. This thesis serves as an introductory analysis of diabetic neuropathy data in an unsupervised environment as further testing should follow when a larger dataset is available. This thesis also serves as a test in multi-view data analysis and how the results gathered from different views can be compared and used without increasing the complexity of the base clustering algorithm. To address these concepts, 1025 lines of C++ code were written to perform the clustering algorithms with increased flexibility

regarding clustering multiple views, perform internal validation testing on the formed clusters, compare the resulting clusters formed based on the individual views, and perform a final clustering test with calculated weights based on the importance of each view being applied to the distance calculations.

CHAPTER 2: DATA

2.1 Multi-View Datasets

As discussed, all the tests conducted on the patients are unique. They produce results based on their own criteria. However, all the tests seem to offer a connection in terms of diabetic neuropathy information. This type of collected data is referred to as *multi-view* data. What is important about multi-view data is that the different views exhibit heterogonous features but hold potential connections [14].

What defines a view can be better illustrated by Figure 4 [15]. On the left, it shows that the news can be divided into four different sources of information. This shows that a multi-view dataset can be divided into its different views where each view has its own version of information while still containing information that can connect it to the other views. On the right, it shows how different types of information can also be used to create a multi-view dataset. Each of the possible types of information, the images and the textual information, are two distinct types of information. But in this case, they are both related to the same topic: dogs. What is important to note in these cases is that each of the different forms of data, the views, can provide sufficient information on their own. You can look at a single news source or only view pictures of dogs and you can sufficiently gather what information the news covered and what dogs are, respectively. Multi-view datasets are not used to create a basic model. Rather, they are used to reinforce a model that can be created by an individual view.

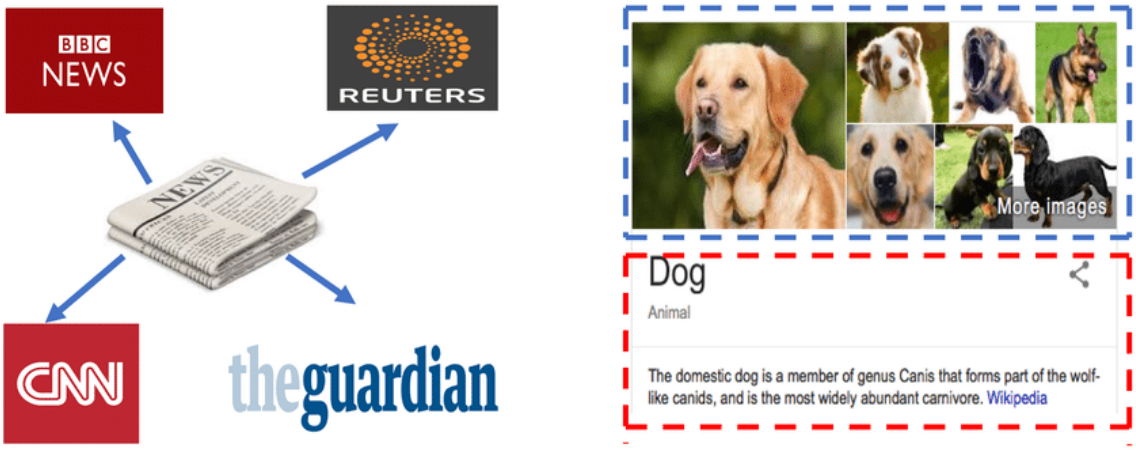

Figure 4: Examples of multi-view data

2.2 Data Collection

The data employed in the research of this thesis was collected over a three-month period (January to April 2013) from 40 diabetic patients with ages ranging from 32 to 71 (10 male and 30 female) at the Department of Endocrinology in the Medical Faculty Hospital of Bezmialem Foundation University [4]. Before the data is given for each patient, the patient is labeled with a unique ID, their age, and their gender. The data following these labels consist of four separate tests conducted for each patient: first a blood sample is taken from the patient for blood testing. Second the MNS test is applied to the subject (2 columns of information recorded based on the self-administered questionnaire and the instrument test), then five different protocols of cortical metrics tests are performed on the subject, and the last step is the data collected from EMG tests performed by a neurologist (eight columns of information recorded from this test). From Tables 1-4 it is shown that the data is separated based on the tests performed. However, it is also shown that there is missing information in two separate forms: missing individual

patient data and missing test information. As such the first step in the utilization of this information is to perform data cleaning.

MNS	
Michigan Neuropathy Screening Instrument: QUEST.	Michigan Neuropathy Screening Instrument: EXAMINATION
mnt-semp	mnt-mua
6	3
-1	-1
9	5

Table 1: Example of MNS data collection

EMG							
emg-pol binary value indicating existence of neuropathy	kps: binary value indicating the existence of karpal tunnel syndrome	MOTOR			SENSORY		
emg-pol	kps	latency	amplitude	conduction velocity	latency	amplitude	conduction velocity
1	1	3.1	6	53.5	2.6	17	60
-1	-1	-1	-1	-1	-1	-1	-1
1	1	3	8.9	57.7	2.6	27	57.1

Table 2: Example of EMG data collection

DIABETIC						
BLOOD TESTS						
hb1ac	**ldl**	**trig**	**mik**	**creatinine**	**ted**	**alt**
6.9	102	103	4.5	0.6	1	12
6.7	154	143	3.5	0.5	1	20
7.6	103	118	4.2	0.6	0.1	21

Table 3: Example of blood test data collection

CM4				
Static Threshold Detection Test	**Static Threshold Detection Test (with different threshold)**	**Double side adaptation test**	**Dynamic Threshold Detection Test**	**Reaction Time Test**
#100_1	**#100_2**	**#109**	**#713**	**#801**
10.6	96	9	29.9333	688.8
14.4	40	180	20.8	585.4
16.8	212	-1	34.2285	562.2

Table 4: Example of Cortical Metrics data collection

2.3 Data Cleaning

As stated above, the dataset utilized for this thesis is a small collection of information. In three of the 40 patients utilized in the tests, information is missing entirely for both the MNS and EMG test results. One of said patients can be observed in Tables 1 and 2. Without knowledge of the necessity of these tests, it can be assumed that these patients with missing information would be detrimental to the outcome of unsupervised data analysis. This results in the three patients being removed from the dataset for testing purposes. While removal of these patients is deemed necessary, it does cause further challenge in terms of building meaningful clusters as there is less available information to work with overall.

Another detrimental feature of the dataset is that multiple columns of test results contain missing information as well. The columns in question all relate to the Cortical Metrics tests performed using the Brain Gauge. Of the above-specified tests shown in Table 4, only two of the columns contained data for all patients. These columns are labeled as “Static Threshold Detection Test” and “Dynamic Threshold Detection Test”. The remaining three columns are removed as without a larger set of data, no safe assumptions can be made for the possible values of these tests.

2.5 Data Normalization

Feature normalization is required to approximately equalize ranges of the features and make them have approximately the same effect in the computation of similarity [16]. Min-max normalization is defined as

$$v' = \frac{v - minF}{maxF - minF}(new\_max_F - new\_min_F) + new\_min_F,$$

where *F* is the feature, *v* is the current value of feature *F*, *minF* and *maxF* refer to the overall minimum and maximum values of feature *F* respectively, and new_max_F and new_min_F refer to the new maximum and minimum values desired for feature *F*. A feature refers to a single column of information. For the sake of simplicity, the desired maximum value is set to one and the desired minimum values is set to zero.

For machine learning, every dataset does not require normalization. It is required only when features have different ranges [17]. This is important in that when you have different ranges for the columns of a dataset, you want to minimize the possible bias caused by these different ranges. This can be illustrated in Tables 5 and 6, where each column has a seemingly different scale for their respective values. In min-max normalization, the scaling of each individual column matters only regarding the proportional values that are calculated, as the minimum and maximum can be pre-defined in the program. Using the defined minimum and maximum values for Table 6, the unique columns can now be seen as appearing similar, but the actual scaling between minimum and maximum values for each column is different. The defined minimum and maximum values ensure that no column has a stronger influence in the results of machine learning algorithms, regardless of the original scale of the values. This is even more important when there are different views to consider. Different views can have a different number

of columns, and each of these individual columns can have an independent scale. The normalization of each column of each view is used to minimize a possible view bias where one or more columns of a view can have a larger effect on the result of machine learning methods. In clustering methods, this bias can affect the distance calculations utilized in the calculation of which cluster a sample best belongs to.

	Feature 1	Feature 2	Feature 3
Sample 1	-12	8	50
Sample 2	15	5	275
Sample 3	40	7	80

Table 5: Standard data before min-max normalization

	Feature 1	Feature 2	Feature 3
Sample 1	0	1	0
Sample 2	0.5192	0	1
Sample 3	1	0.6667	0.1333

Table 6: Results of min-max normalization on Table 5 with a preset minimum of 0 and a preset maximum of 1

CHAPTER 3: CLUSTERING METHODS

3.1 K-Means Clustering

K-means clustering is one of the simplest and most popular unsupervised machine learning algorithms used in unsupervised machine learning [18, 19, 20, 21]. K-means is a method of clustering that follows the concept of determining partitions of patterns into *K* groups, or clusters, such that the patterns in a cluster are more similar to each other than to patterns in different clusters [22]. These centroids are typically samples selected from the existing dataset. In most cases the Euclidean distance formula, the most commonly used distance formula, is applied for every sample to determine which centroid said sample is closest to. After the closest centroid is found, the sample is grouped, or clustered, together with that centroid. This is repeated on all samples until every sample has been clustered. Afterwards, the sum of the squared estimate of due errors (SSE) formula is calculated using the formula

$$SSE = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2,$$

where k is the number of clusters, n is the number of samples, x_i is the i^{th} sample, and c_j refers to the centroid for cluster j. The centroids are then re-calculated using the averages of the features of all samples within each cluster respectively, and all samples are re-clustered based on the new centroids. The SSE value is then re-calculated as well. This SSE value is then compared to the SSE value from the previous iteration to get a convergence value as follows:

$$Conv_{ij} = \frac{SSE_i - SSE_j}{SSE_i},$$

where SSE_i is the SSE value from the previous iteration and SSE_j is the SSE value calculated from the current iteration. If the convergence value is less than or equal to a

preset convergence threshold, then it is determined that the samples have not changed clusters significantly between these iterations and the algorithm is concluded for that run. This can be repeated for multiple runs with different, randomly selected centroids to find the lowest possible SSE value.

3.2 Agglomerative Hierarchical Clustering

While k-means clustering is a popular method, there are a few drawbacks in the method. In k-means, the number of clusters must be pre-determined, and the visualization of the results can be complex. Hierarchical clustering addresses the visualization issue by building a representative model, called a dendrogram, that shows how all samples are grouped together. Hierarchical cluster analysis forms clusters iteratively, by successively joining or splitting groups [23, 24]. Hierarchical clustering can initially be separated into two distinct types: agglomerative, which works “from the ground up”, and divisive, which works “from the top down” [25]. The main difference in these methods is that in agglomerative clustering all samples are placed into their own distinct clusters which are then combined based on their closest cluster, while divisive clustering places all sample into one large cluster which is then split in each iteration. However, for both methods, the results can be illustrated using dendrograms.

Dendrograms, such as the one in Figure 5 [26], show each step of the hierarchical clustering process. The horizontal connection lines indicate which step in the process correlates to the clustering of these points. In Figure 5, when addressed from an agglomerative approach, points E and F are the first points that are clustered, followed by points A and B. These connections are based on a distance calculation, similarly to k-means clustering. Also, similarly to k-means clustering, Euclidean distance is a common

method of distance calculation for hierarchical clustering. But following these first two steps, the next point, point D, is connected to the already formed cluster formed from points E and F. This leads to the secondary distinction in hierarchical clustering, the method of cluster linkage. There are multiple methods used to determine the closest clusters, such as single linkage, which determines the two closest clusters by finding the two closest objects within these clusters, or complete linkage, which operates similarly to single linkage but utilizes the two farthest objects as opposed to the two closest objects. There is also centroid linkage, which creates a centroid for each cluster based on all points in said cluster and calculates which two centroids are closest together for combination. While these methods are all valid, this project utilizes the average linkage method. This method uses the average distance of all points in one cluster and all points in another cluster. This method ensures that all points in each cluster are represented in the calculation for cluster combination.

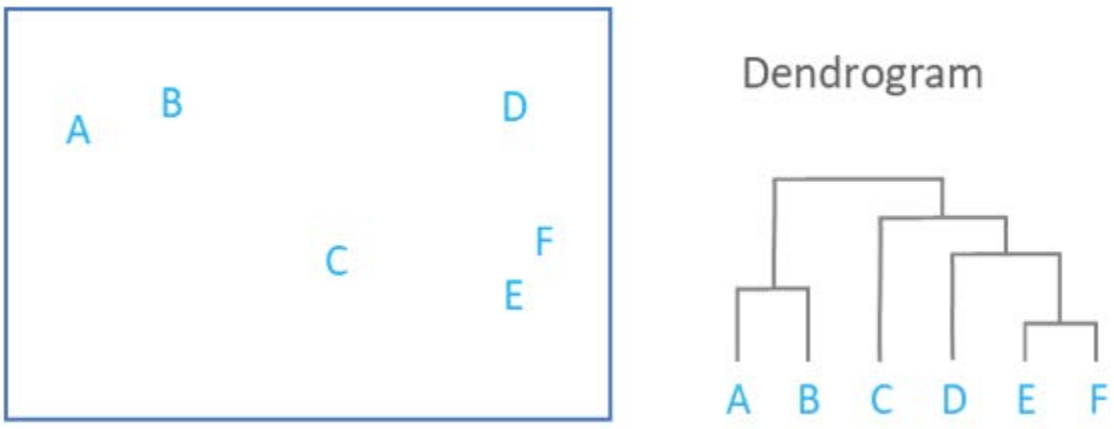

Figure 5: A simple dendrogram using graphed points

3.2 Clustering Validation Measures

When determining the goodness of clustering results, there are two main styles of validation checking: internal validation and external validation. External validation is typically based on the presence of the true partitioning of the samples, meaning that a dataset has already been optimally clustered, and the best clusters have been determined and listed. If the true partitioning is present, then external validation can be performed to estimate if the current results presented by the algorithm used are similar to the optimal results previously gathered. Without the presence of the true partitioning, internal validation must be used to estimate if the clustering results are optimal. Internal validation is useful when working with datasets that have not been fully tested and labeled, while external validation is useful to test the reliability of the created algorithm utilized for clustering, as well as comparing different clustering results.

A method of internal validation that considers the similarity of each sample to all other samples belonging to its cluster (cohesion) as well as accounting for the similarity between said sample and samples belonging to other clusters (separation) is the silhouette coefficient (also known as the silhouette score) which is used to study the separation distance between the resulting clusters [27]. The silhouette score of a single sample is defined as

$$s(i) = \frac{b(i) - a(i)}{max(b(i),a(i))},$$

where *a(i)* is the average distance of sample *i* to all points in the same cluster and *b(i)* is the average distance of sample *i* to all samples in the closest cluster. Using this information, a value is recorded for each sample falling on a range from -1 to 1. A value closer to 1 indicates that the sample in question is accurately clustered, a value close to 0

indicates possible overlap between clusters for that sample, and a value closer to -1 indicates that the sample is most likely clustered incorrectly. The main usefulness of silhouettes lies in the interpretation and validation of cluster analysis results [28]. In a singular view dataset, the silhouette values can be used to estimate the best number of clusters for a dataset, but in multi-view datasets this can be an issue when comparing views, as each view could have a different optimal number of clusters.

A standard method of external validation is the rand index. The use of the Rand index was proposed by William R. Rand as a method of comparing how well clustering methods perform [29]. This is done using

$$R = \frac{a+b}{a+b+c+d},$$

where *a* is the number of pairs of elements in the same subset in the first clustering and the number of pairs of elements in the same subset in the second clustering, *b* is the number of pairs of elements in a different subset in the first clustering and the number of pairs of elements in a different subset in the second clustering, *c* is the number of pairs of elements in the same subset in the first clustering and the number of pairs of elements in a different subset in the second clustering, and *d* is the number of pairs of elements in a different subset in the first clustering and the number of pairs of elements of elements in the same subset in the second clustering. This can also be referenced as the number of *true positives*, *true negatives*, *false positives*, and *false negatives*, respectively. A value closer to 1 indicates that the clustering results are similar, while a value closer to 0 indicates that the clustering results are different. This value can be used to test the validity of the algorithm being used when tested on a dataset where the true partition is already known.

While the rand index is often used to compare results of a clustering algorithm to the true clusters to which each sample belongs, it can also be used to compare the similarity between two separate clustering results for the same dataset without having a true partition. This can compare the results between two distinct clustering algorithms as well. In multi-view data clustering, this comparison measure can be used as a method of finding similar results between optimal clusters formed from each view.

3.3 Adapting Clustering Methods for Multi-View Data

While many clustering methods are reliable and well-documented, most of them are only suitable to single view data [14]. The brute-force method to address this without altering the actual algorithms is to concatenate all views together and treat the dataset as a single-view dataset. The issue with this method is that, although it does address all features in the dataset, it does not address the importance of each view in the dataset. In the diabetic neuropathy dataset, there are four distinct views. These views, while all relating to the same condition, are distinct in the testing method utilized. What is unknown from these different tests is the correlation between the results of each test as well as the order of importance regarding each test.

Research into the field of multi-view clustering is still young. Some methods involve more complex processes such as the creation of multiple graphs that are then combined into a graph built on the similarities between them [30], or the use of each view to create labels that are then utilized in a semi-supervised method to build the best model. Still, these methods have the disadvantages of either being very specific in the use-case scenario or that these require some more involved changes to popular clustering algorithms. However, methods have been tested that attempt to weigh the views and

perform the clustering method using these weights. This can be seen on a smaller scale already, where a single-view dataset may need to have different features counted as more important than others. These weights can either be calculated, or pre-determined by a user of the program, but the results are still expected to show some variation in result where the more important features cause the samples to cluster more similarly regarding the important features than the lesser features. The use of weights can be applied in either the distance calculation or in the new centroid calculations. But, above all, the intention of this project is to test a simplistic method of view weighting to cause some variation in the clustering of the samples without re-inventing the popular clustering algorithms.

A variation of Euclidean distance can be utilized by setting each feature distance to be multiplied by its respective weight as

$$dist(d_i, d_j) = \sqrt{\sum_K w_k (a_{i,k} - a_{j,k})^2},$$

where K is the feature number, w_k is the weight of feature K, $a_{i,k}$ refers to feature a of sample i, and $a_{j,k}$ refers to feature a of sample j. If we apply the distances as a distance between each sample and each centroid while attaching the weights of each respective view to the features within said view, we can attempt to pull the cluster results in favor of the important views without necessarily disqualifying the other views. The issue that exists in this method is the determination of view weights.

The issue of view importance is one that is continuously mentioned in multi-view data research. Assumptions can be made as to which views are important, but this can potentially harm the results of clustering without a basis for the assumptions. Assessing how important a view is can take on different meanings, such as which view(s) is/are recommended by professionals in the respective field to which the dataset belongs.

Addressing this from an unsupervised learning approach involves the assessment of which view performs best. Utilizing the silhouette value calculations mentioned above, a ranking of views based on these calculations can be made, and these rankings can be used to build the weights for the dataset. These weights can then be applied to the weighted Euclidean distance formula and, using the combination of views into a single dataset with the weights applied, the effects of view weighting can be compared. The rand index can also be used here to better detect patterns that may exist between the clusters formed in each view.

CHAPTER 4: RESULTS

4.1 K-Means Individual View Comparison

When addressing a single view, the silhouette values can be utilized in assessing the optimum K value for K-means clustering. This is another aspect of K-means that can be complicated when addressing multiple views: different views could have different optimal K values. The comparison of individual view silhouette results is given in Figures 6-15.

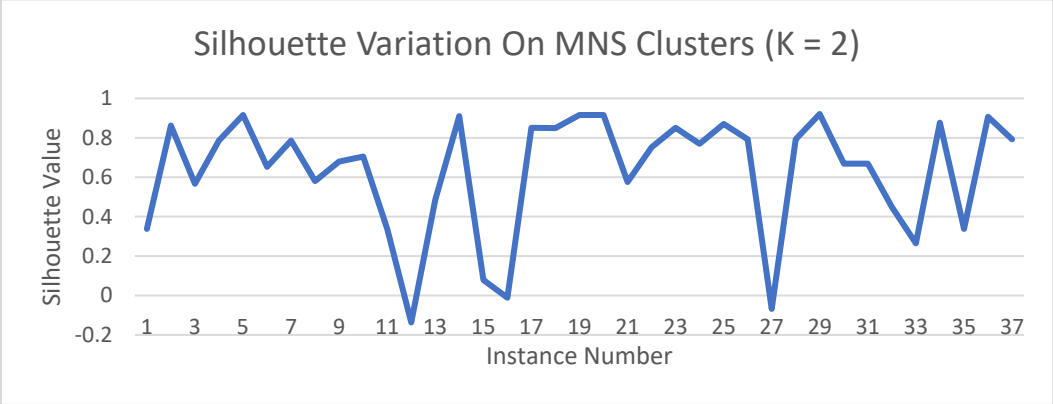

Figure 6: Variation of silhouette values based on Michigan Neuropathy Screening data when K equals 2.

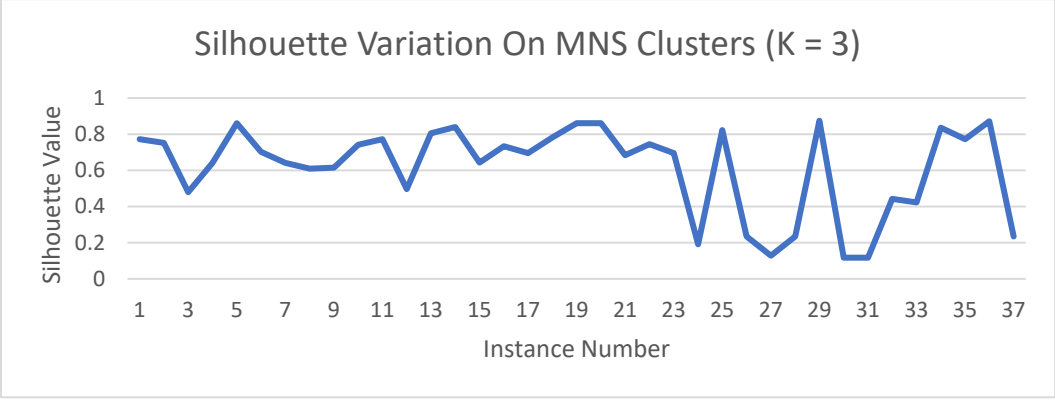

Figure 7: Variation of silhouette values based on Michigan Neuropathy Screening data when K equals 3.

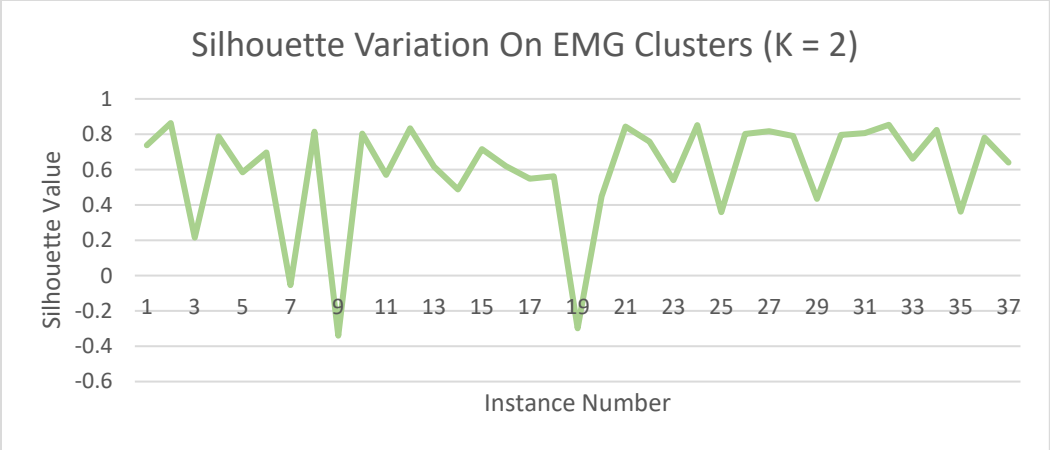

Figure 8: Variation of silhouette values based on Electromyography data when K equals 2.

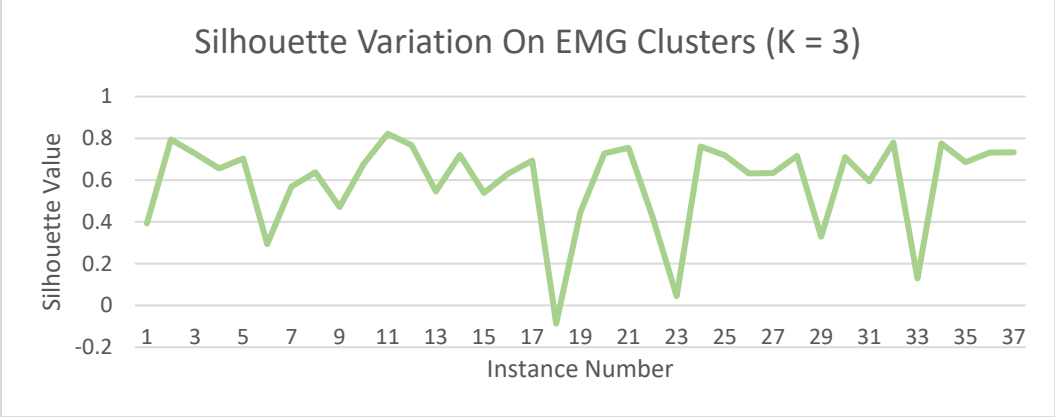

Figure 9: Variation of silhouette values based on Electromyography data when K equals 3.

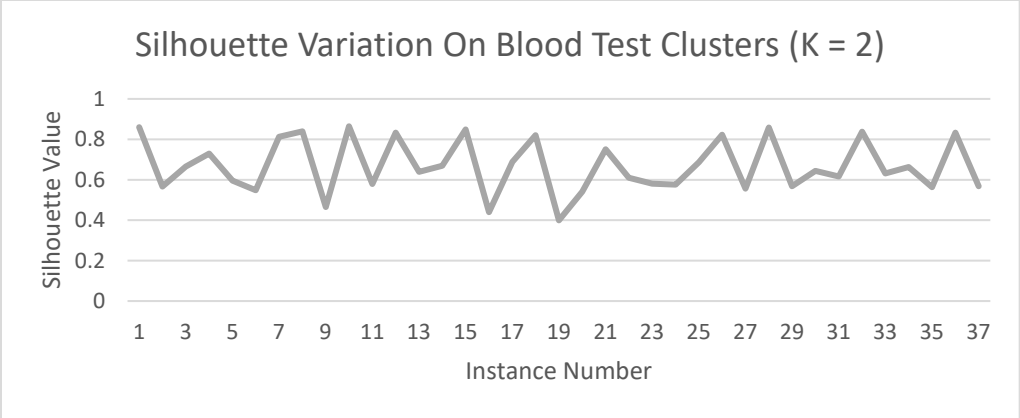

Figure 10: Variation of silhouette values based on Blood Test data when K equals 2.

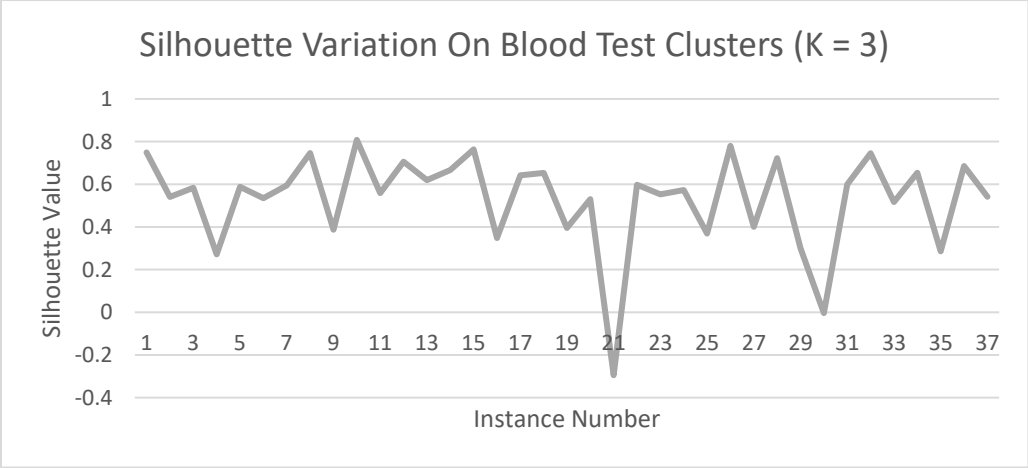

Figure 11: Variation of silhouette values based on Blood Test data when K equals 3.

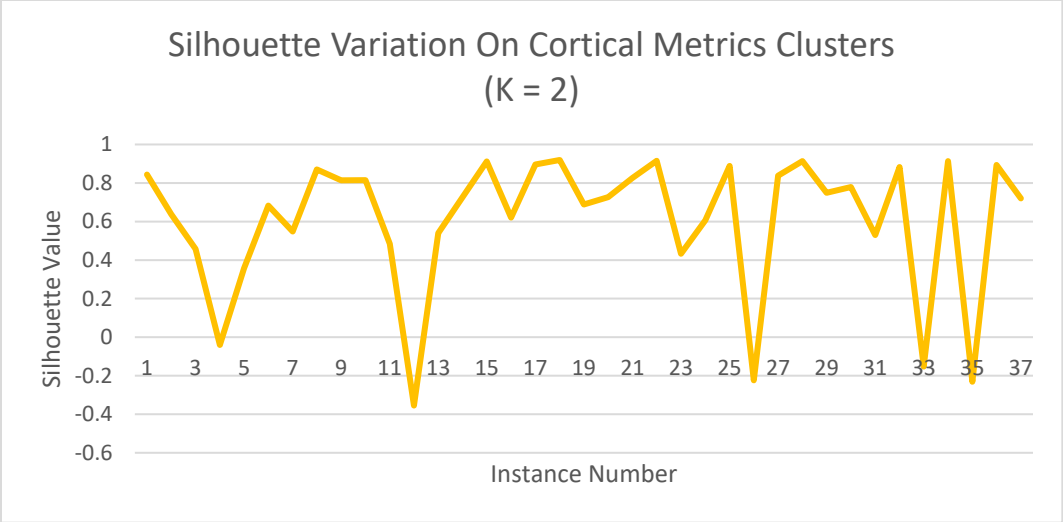

Figure 12: Variation of silhouette values based on Cortical Metrics data when K equals 2.

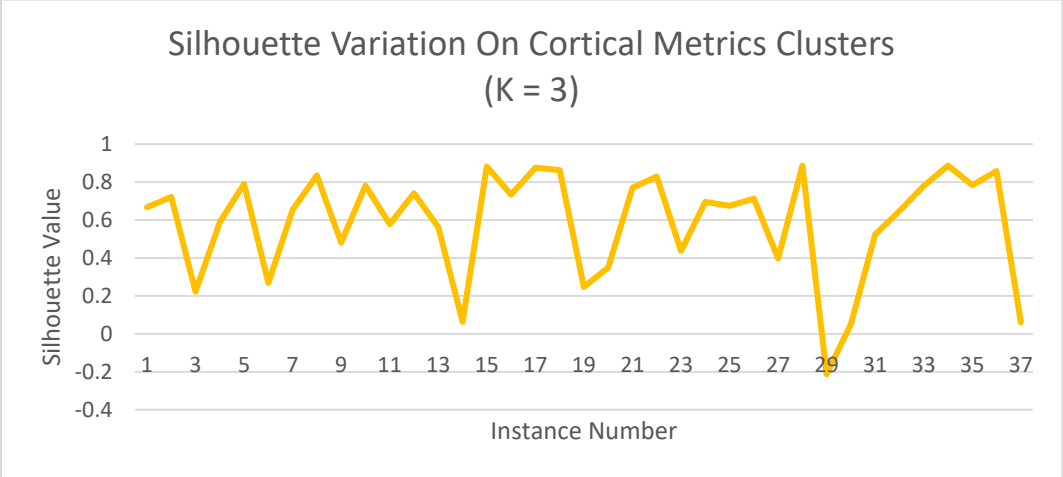

Figure 13: Variation of silhouette values based on Cortical Metrics data when K equals 3.

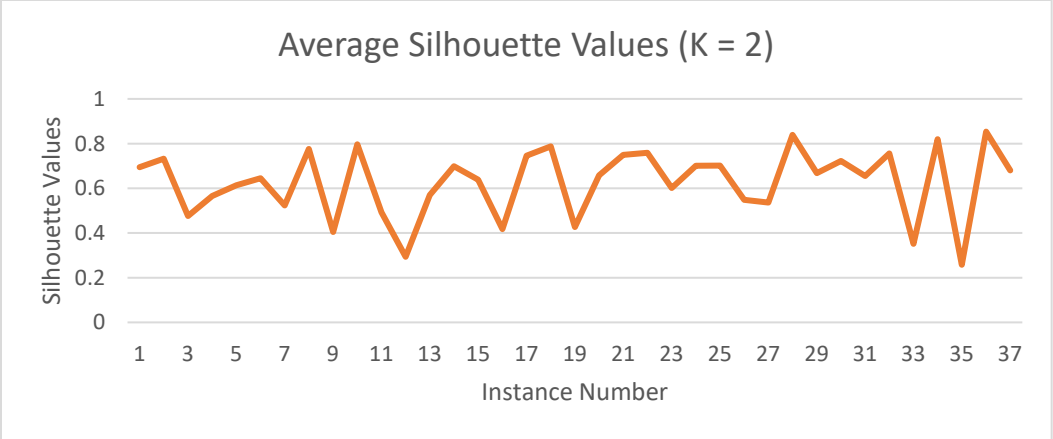

Figure 14: Average silhouette values from all tests when number of clusters (K) equals 2.

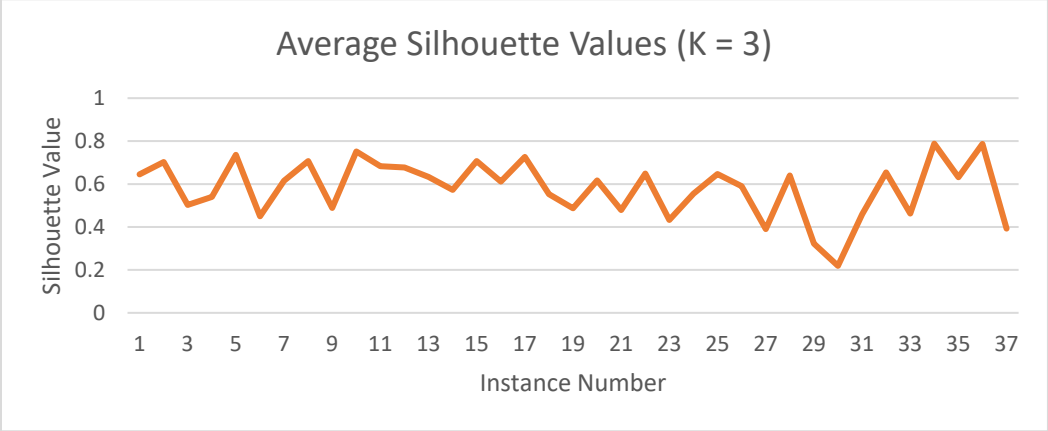

Figure 15: Average silhouette values from all tests when number of clusters (K) equals 3.

What is interesting within these results is that there are few similarities between the views, regardless of the number of clusters. Another interesting pattern that exists is that, despite cases such as Figures 8-9 and 12-13 where the silhouette values are more dynamic when K equals 2 rather than 3, the overall average silhouette values from each view implies that 2 is the optimal number of clusters overall. When addressing Figures 14-15, the overall average silhouette values per point follow a similar trend of wavering near the 0.6 range, but Figure 14 displays sharper contrasts in the values than in Figure 15. Given this information one could make an argument for either value for the optimal number of clusters, but from the perspective of a program, the silhouette values support 2

as the optimal number of clusters regardless of using all or one view to make the decision. A higher average silhouette score, coupled with samples showing higher silhouette scores in cases where K = 2 (Figures 6, 8, 10, and 12) support this case. Knowing the optimal number of clusters is useful in looking at the individual views, but it does not necessarily indicate that the clusters that are formed follow the same pattern. While the clusters formed from the EMG test data could suggest clusters based on severity, the clusters formed from the blood test data could be based on a different criterion. Knowing that the optimal number of clusters is the same between each test, further comparison testing can be performed between the cluster results from each view to find if the optimal number of clusters reveals any patterns between the results.

While the silhouette values can be used to show how well the samples are clustered as well as identifying the optimal number of clusters, it is not the most efficient method to find similarities between the cluster results. To further test the patterns provided by K-means clustering on each view, the rand index provides a better measure of the similarities between results on each view.

When comparing cluster results with a true partition, a value close to 1 indicates that the results are close to identical to the clusters they should be. But when comparing two separate cluster results with neither of them being a true partition, the result of the rand index values indicates similar patterns in the results. When the optimal cluster results in each of the diabetic neuropathy tests are compared, as shown in Table 7, the results show something interesting. Additionally, the rand index comparisons of the combined views method of K-means clustering are given, which shows that combining

the views without using any weighting method produces results identical to the blood test view results.

	Combined	MNS	EMG	Blood	CM4
Combined	1	0.474	0.563	1	0.513
MNS	0.474	1	0.506	0.474	0.543
EMG	0.563	0.506	1	0.563	0.587
Blood	1	0.474	0.563	1	0.513
CM4	0.513	0.543	0.587	0.513	1

Table 7: Comparison of cluster results (K = 2) using rand index values

Only one comparison result gives a value greater than 0.6, and that result is a 1. While this does give great insight into the importance, or the bias, that blood test data provides, it does not show very significant patterns. A rand index value of 0.5 indicates that the cluster results being compared are similar for half of the samples. Given the limited number of samples in the diabetic neuropathy dataset, this would indicate that 18 or 19 of the samples are clustered similarly in most cases. This can still be useful when the results are compared to each other rather than compared to expecting ideal matches. Given the data provided, it is shown that clusters formed from the Cortical Metrics dataset are most similar to the cluster results formed from the Electromyography test. This is the second highest rand index value in all comparisons, but this can be further extended to find the best possible combinations of views according to the most similar results gathered. Regarding the results in Table 7, this indicates that Michigan Neuropathy Screening cluster results are most similar to Cortical Metrics cluster results, Electromyography cluster results are most similar to Cortical Metrics cluster results, Blood Test cluster results are most similar to Electromyography cluster results, and Cortical Metrics cluster results are most similar to Electromyography cluster results. Using this information, further work within the field can build on the possible

relationships between these different views, or the connections can be revisited once more samples become available.

4.2 Testing Influence of View Weights on K-Means Clustering

Given the silhouette scores from the individual results of each view in the diabetic neuropathy dataset, a ranking of the views can be created based on the average silhouette score for each view. The rankings provided show that if the goodness of the clusters is utilized as the measure for the importance of a view, then the rankings from most to least important are Blood Test, Michigan Neuropathy Screening, Cortical Metrics, and Electromyography. Using these rankings, weights can be determined for each view. A simple way to address these weights is to assign the weights as the opposite of the ranking for the views. So, Blood Test, while being rank number one, would have a weight value of four applied. MNS would be assigned a weight of three. Essentially, any weight higher than one indicates a larger amount of influence given to that view in the clustering result. What is interesting about these determined weights, due to either the limited number of samples or the reflection of the silhouette scores, is that the results of weighting the views based on these ranks showed identical results to the combined-view approach. Even if the view rankings were swapped to allow for the least-represented view to have the most pull on the results (Electromyography being given the weight of four, Cortical Metrics given three, etc.), the resulting silhouette scores were still identical. If a multiplier is applied to the weight scheme, however, the results are impacted. Figure 16 shows the silhouette score results when the original rank is used as a simple weighting metric, the results when this simple weight metric is swapped, and when the swapped metric is utilized with a 10-times multiplier added to amplify the weight effects.

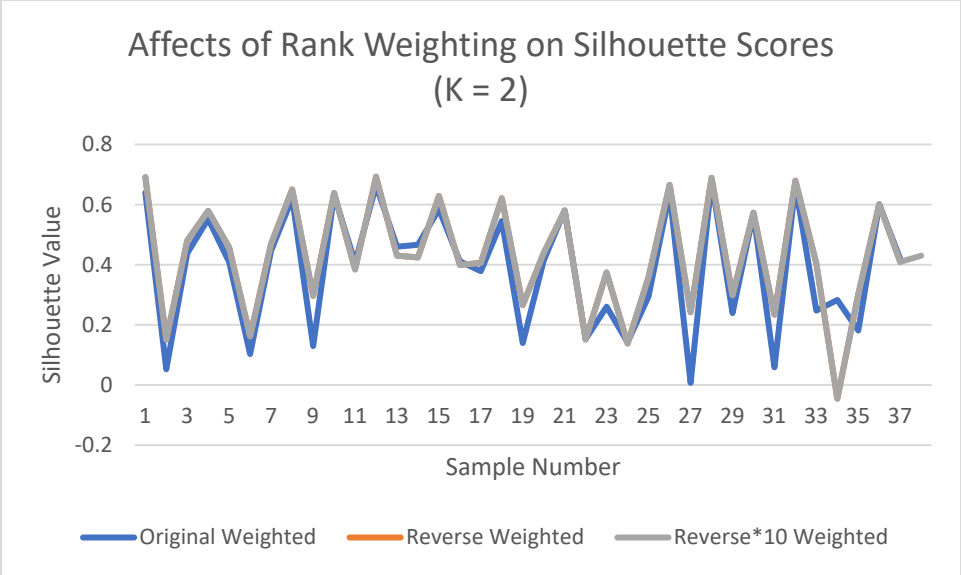

Figure 16: Comparison of weight results on silhouette scores when K = 2

Although the original and reverse weighted methods are shown, the results overlap entirely so only one appears to be visible. When the multiplier is given to the reverse scheme, however, the results are altered. In most samples the patterns remain similar, but the multiplier method produces slightly more dynamic results for most points. This shows that a simple weighting method can be applied to alter the results of this dataset, but the differences in smaller datasets require an increase in the size of the applied weights to show any significant effects on the clusters formed using K-means clustering.

4.3 Agglomerative Hierarchical Clustering View Results

Hierarchical clustering provides the opportunity to view cluster formations at the individual level. While K-means clustering relies on the predetermination of the number of desired clusters as well as having results that vary based on the centroids selected, hierarchical clustering benefits from providing a result that can show how each sample relates to all others. Utilizing the average linkage method mentioned above, all samples in each formed cluster will be used to find the best combinations. When applying hierarchical clustering to multi-view datasets, not many methods currently exist due to the time complexity of $O(n^3)$ when applying hierarchical clustering on standard datasets. However, in datasets with fewer samples, this increase in time complexity is not significant when compared to the results that can be gathered.

Where comparisons between samples in K-means clustering can be difficult to visualize, dendrograms created from hierarchical clustering analysis can be used in the diabetic neuropathy dataset to visualize the distinct patterns created from each view. Using the formed dendrograms shown in Figures 17-21, these patterns can be visualized in a step-by-step fashion.

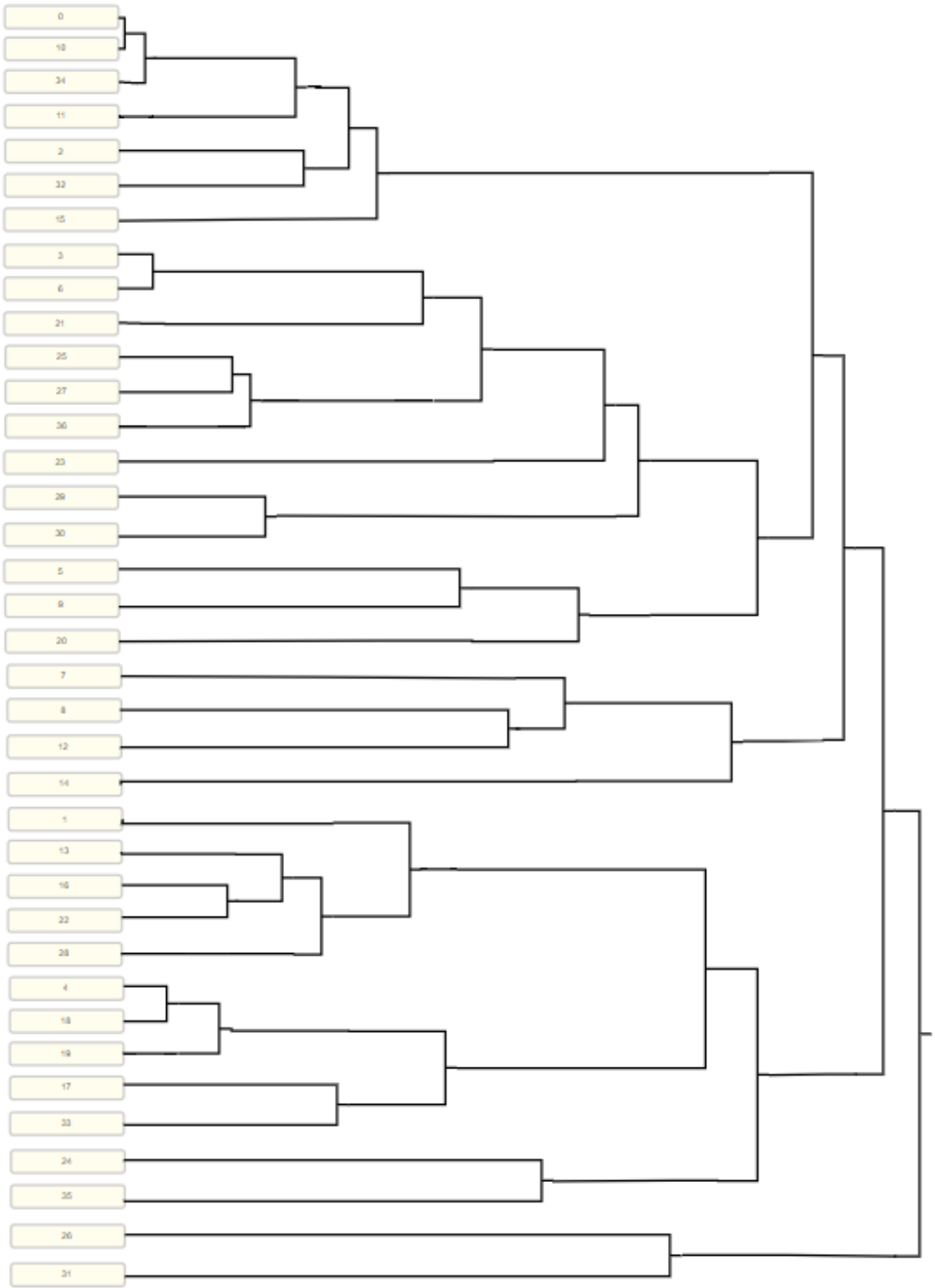

Figure 17: Dendrogram formed from Michigan Neuropathy Screening hierarchical clustering.

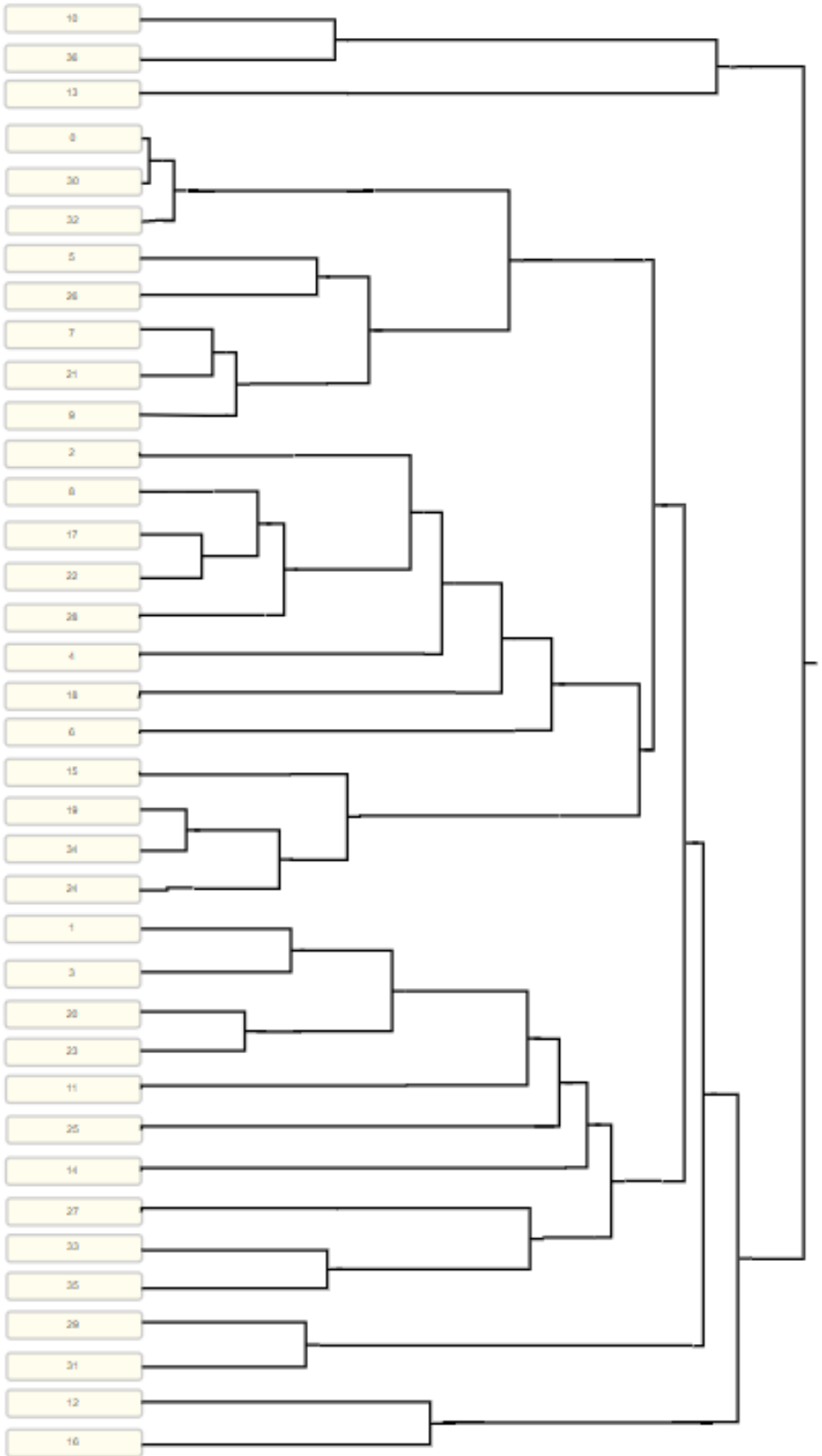

Figure 18: Dendrogram formed from Electromyography hierarchical clustering.

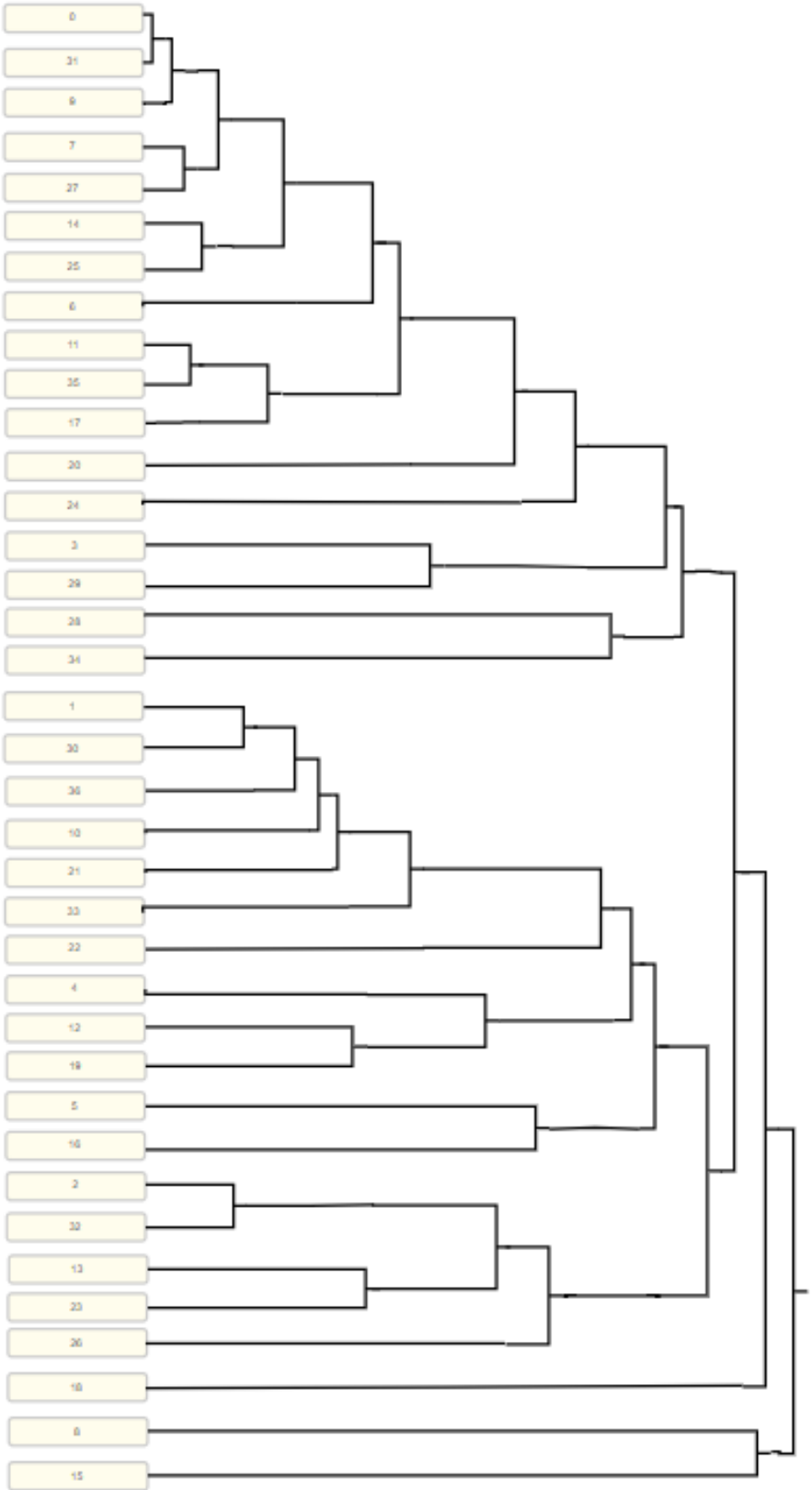

Figure 19: Dendrogram formed from Blood Test hierarchical clustering.

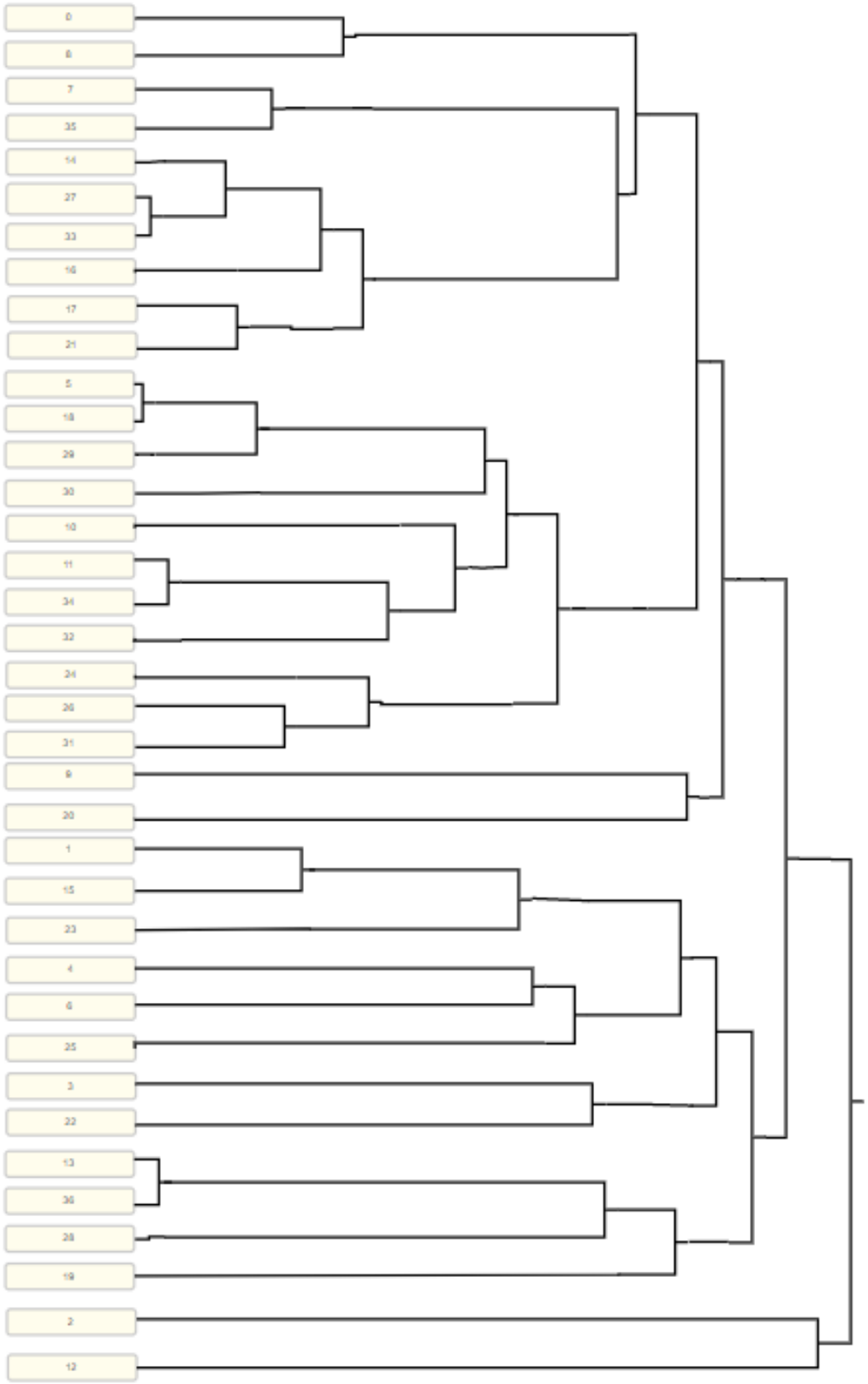

Figure 20: Dendrogram formed from Cortical Metrics hierarchical clustering.

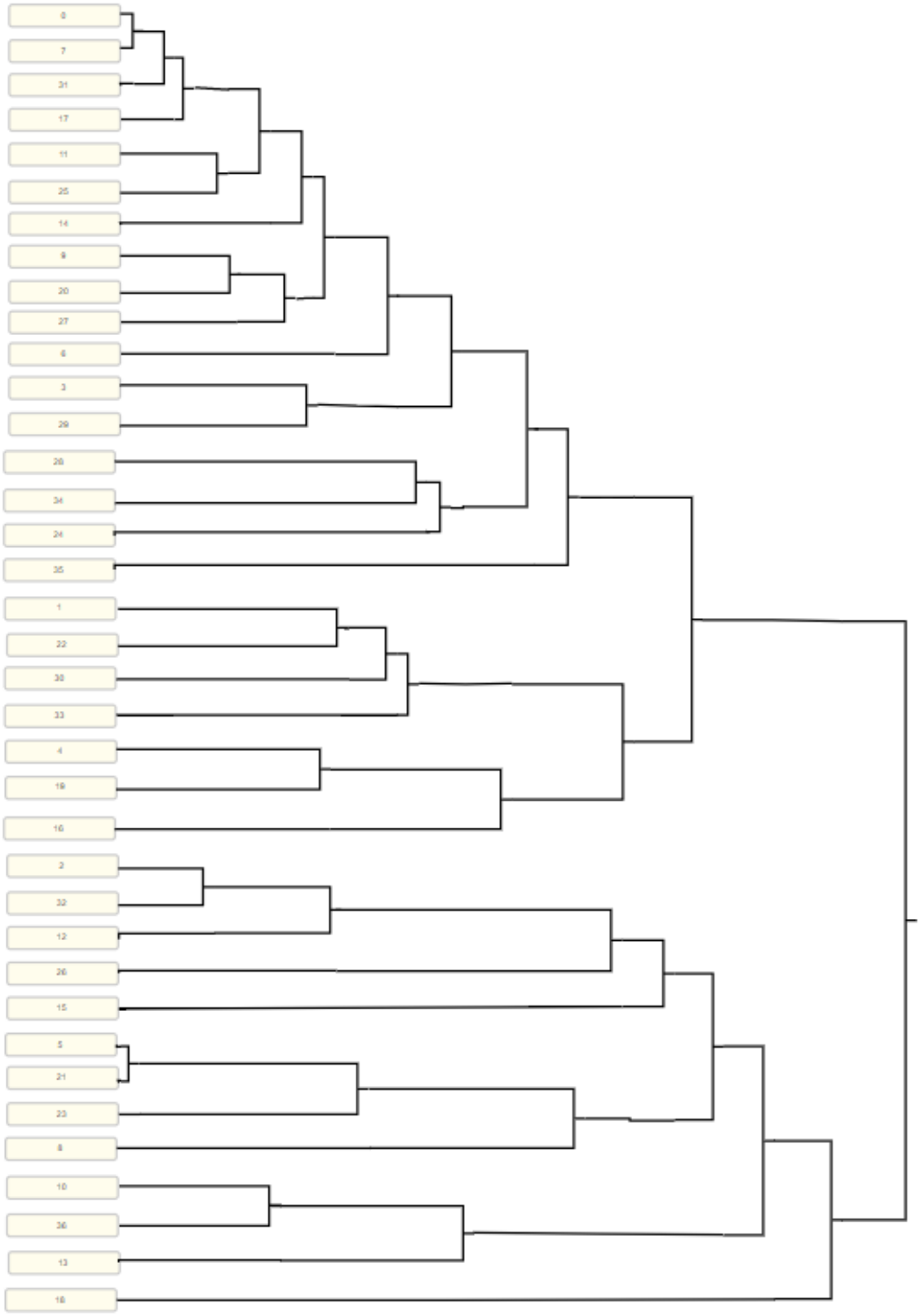

Figure 21: Dendrogram formed from combined-views approach hierarchical clustering.

The resulting clusters formed from the Michigan Neuropathy Screening data shown in Figure 17 provide a strong correlation to the conditions of the patients involved. The data from the MNS tests is formed in two columns, with one being based on the patient's own perspective of their condition and the other being based on the observations of a physician. These scores do not have extreme variations, causing the cases such as in samples 0, 10, and 34, or in samples 4, 18, and 19, where the scores were exactly the same. Given that these scores vary on a scale of 0-10, the results would suggest that the clusters formed from the MNS examinations could be highly useful in comparing the conditions of patients. However, given that the first part of the MNS examination is self-administered, user error is much more likely to affect the results.

Looking into Figure 18, the clusters formed do not follow the same pattern as in Figure 17. Specifically, where the MNS examination results are general scores that have a higher chance of being equivalent, the results of the electromyography test have much more diverse measurements and more recorded columns of information. As is the nature of hierarchical clustering, all features are linked to those that are the most similar, but these similarities are not as strong compared to the MNS results. In cases such as the cluster formed from samples 10, 36, and 13, the samples are grouped not due to how similar they are to each other, but how different they are from all other samples. Given the larger amount of information gathered from this examination, it is still important to note that the more immediate connections found, such as the cluster formed from samples 0, 30, and 32, arguably show more revealing connections between patients. The electromyography tests are administered by professionals and are less likely to contain

user error. These tests also measure the nerves more directly, so the connections found can indicate a similar level of nerve damage caused by diabetes.

K-means results performed on the blood test information indicated that the blood test clusters formed were the most internally valid clusters out of all tests in the dataset. Similar to the electromyography tests, there are multiple columns of diverse information involved in the blood tests, and no two patients had the exact same result. Even the most similar clusters shown in Figure 19 (0, 31 and 6, 7 and 27, or 11 and 35) had more diverse values in each feature than in any of the clusters formed from the other tests performed on the patients. Similar to the electromyography test, the blood tests are less likely to be affected by user error. Given the diversity of the clusters formed, even on an individual level, the individual connections and lower-level clusters may be less useful in further testing than observations on the larger clusters formed later in the clustering process.

Cortical Metrics examinations group the patients together based on their reaction time. Knowing this, the clustering of the patients is best addressed as how similar the reaction times between the patients are. Similar to the EMG and blood tests, the values are diverse, causing no two patients to have exactly the same values across all recorded features. However, similar to the MNS tests, only two columns of usable information are present. From these observations, the resulting clusters in Figure 20 mimic the closeness of the MNS results where two different patient results have the capacity to be extremely similar (such as samples 5 and 18, 27 and 33, or 13 and 36) while still not containing identical values. Cortical Metrics examinations are newer forms of examination and do involve the chance for user error, but also provide the chance to better measure the

severity of the neuropathy condition without invasive procedure. Data cleaning resulted in three of the five examinations being excluded from testing due to lack of available information, so the current results should be revisited when more material from the excluded tests becomes available.

Comparing Figures 17-20, it is most noticeable that no two dendrograms follow the same pattern. The only true similarities to note are that in each dendrogram, outliers become apparent when only two distinct clusters remain. However, these outliers do not contain the same patients in each dendrogram. When combining all views into a singular view for clustering, as shown in Figure 21, the results appear more evenly distributed when 2 clusters remain. When compared to the K-means clustering performance, the distribution of samples into clusters based on the combined approach in hierarchical clustering does not match the fact that the Blood Test view had the exact same clusters formed as the combined method K-means clusters. Further comparisons between Figure 21 to Figures 17-20 show that no one view perfectly matches the results of the combined approach. On the perspective of multi-view clustering this indicates that the unique properties from each individual neuropathy test did have a significant influence on the clusters formed in each step of the hierarchical clustering process. This lends to the credibility of each test, suggesting that each test provides results that, though relating to the same overall diagnosis, gives unique information and the exclusion of any test could be detrimental to any comparison procedure.

4.4 Comparing K-Means and Hierarchical Results

While K-means clustering identified the optimal number of clusters and the possible patterns between the tests results, and hierarchical clustering helped to visualize

the connections between samples on an individual scale, it is also important to note the similarities between the cluster results in each method. K-means relies on certain presets, such as the convergence threshold value and the number of desired centroids, and this can cause variation in the results gathered from the clustering algorithm. Hierarchical clustering tends to be more consistent and does not heavily rely on presets, but it does vary depending on the linkage method utilized and tends to be more time complex than the standard K-means algorithm. Given the results from both clustering algorithms, a comparison between the methods can be used to test if the results remain consistent across methods.

The results gathered are based on the observation that two is the optimal number of clusters according to K-means. The dendrogram shown in Figure 22 shows the direct comparison between the clusters formed in K-means clustering and the results formed in hierarchical clustering applied to the combined-view approach. The clusters formed in both methods are similar, but seven samples are grouped in different clusters when different methods are used. What is unique about these seven samples is that they are all clustered together before being combined with the first cluster. This observation can mean several things. This could indicate that the K-means algorithm clustered this group of patients incorrectly, but the average silhouette scores for these samples indicate that the clusters formed in K-means are clustered correctly, although no average value exceeded 0.8, indicating that no sample is clustered perfectly in K-means. This could also indicate that these samples are overlapping and could belong in either cluster. Finally, the results in Figure 22 could be varied based on the differences within the clustering method.

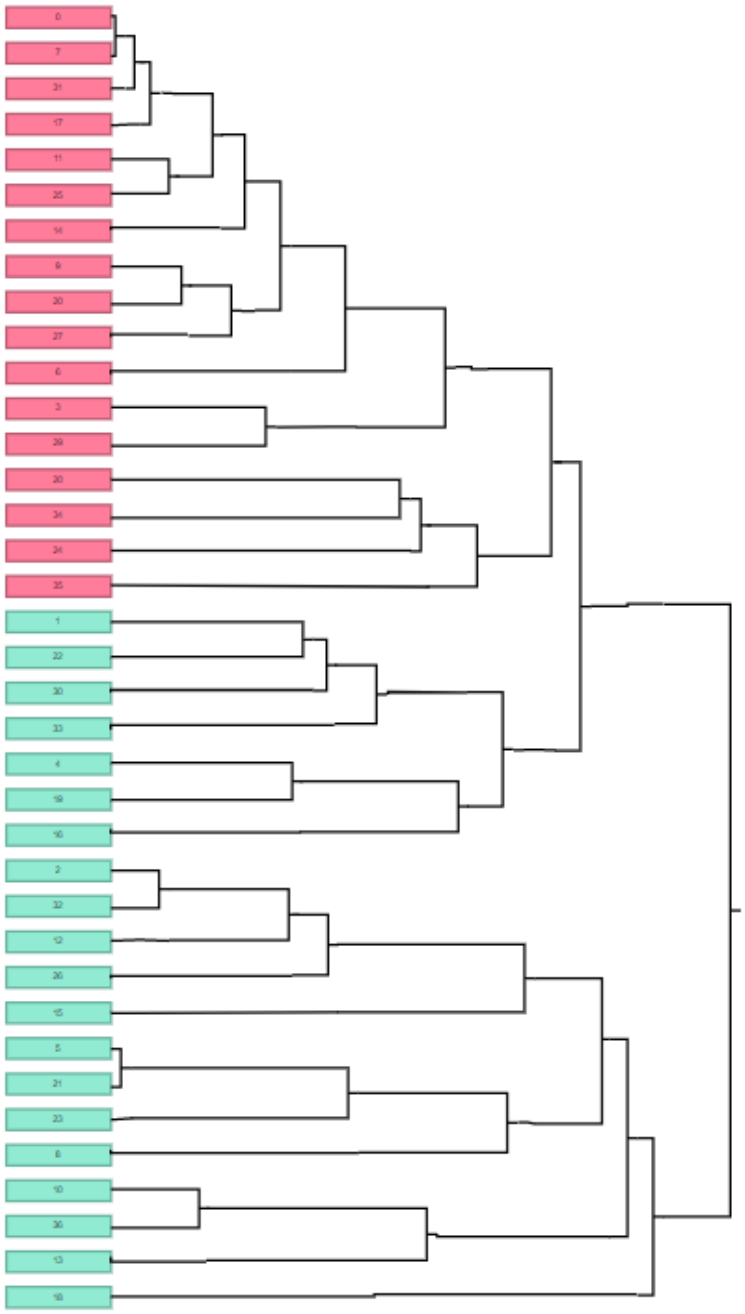

Figure 22: Cluster results regarding the combined-view clustering approach, different colors indicate the two clusters formed in K-means clustering

In K-means clustering centroids are pre-determined initially and recalculated in each iteration, but in agglomerative hierarchical clustering all distances between samples are considered with each iteration. K-means clustering also relies on a convergence measurement to estimate when all samples are clustered completely, while agglomerative hierarchical clustering runs until either one cluster remains or a predetermined cut-off point is reached for the number of clusters. While these seven samples are interesting to note, what is also interesting is that the clusters are otherwise identical. All samples belonging to the same cluster in K-means clustering, specifically the red cluster, also belong to the same cluster in hierarchical clustering. Seven of the twenty samples in the green cluster are placed with the red cluster, but thirteen samples are still clustered similarly. Overall, the resulting similarity between these results shows that 81% of the samples are clustered similarly. This shows sufficient agreement between the cluster results, so the desired method of clustering will not significantly change the result.

CHAPTER 5: CONCLUSIONS

The unsupervised learning methods applied to the diabetic neuropathy data did not reveal any definitive results in regard to the possible connections that exist between each test. However, comparisons between the clustering results of K-means clustering on each view reveal that possible similarities exist between results gathered from the Electromyography tests and the Cortical Metrics test. When more data becomes readily available, this similarity can be further investigated utilizing the cluster comparisons used in this thesis. Another insight is gained from the comparison of how well-formed the clusters are when based on each view and different cluster numbers are utilized. Based on each view, the internal validation indicates that two clusters provide the best average result for the samples overall.

If the agglomerative hierarchical clustering algorithm is applied, the clusters formed by the different views do not show any significant similarities. The results do, however, show the smaller patient groupings formed when different tests are emphasized. These groupings are less significant in assessing the overall view importance, but they are significant in possibly finding what significant groups are formed within each test conducted. These groups can possibly be applied in a supervised manner based on the individual tests to predict the group for which a newly introduced patient is most likely to belong.

The use of a more simplistic weighting method for views in a multi-view clustering algorithm did not prove to show significant changes to the resulting K-means clusters when compared to concatenating the views into a singular dataset. This could be based on the simplistic method of weighting, as the multiplier applied to the weights did

provide some more notable changes to the silhouette scores. Noting that the use of weights based on the internal validation of each view can affect the outcome when the weight values are intensified, this can be a simpler start to addressing multi-view datasets than the methods that either involve major alterations to the algorithm or alterations to the dataset being tested.

As described, the method of view importance used in this thesis relies on the internal validation of the individual views as opposed to the combination of results between views. While the research in the topic of multi-view datasets may be complex in terms of clustering algorithm variation, these methods can be applied to diabetic neuropathy data when more samples become available. This thesis sought to find a less intensive method of combining cluster results between views, seeking to alter the popular clustering algorithms very little and focusing, instead, on the validity of the views to reinforce clustering results. Other methods of multi-view clustering tend to focus on creating a fusion graph of all cluster results, which also implies the alteration of the dataset to fit into a two-dimensional space to allow for this, or by delving deeper into the weighting of views through cluster-weighting [30] and self-weighting schemes [31]. Another common issue addressed in multi-view research is the increased time complexity, as most research intends to adapt the view importance calculation to be applied to large datasets with as minimal a change in computation time as possible. In the Diabetic Neuropathy dataset in this thesis, the time complexity proves inconsequential as all view clusters are calculated in seconds. Testing the simple weighting method on large datasets does prove to be very time consuming, and this will need to be addressed if the use of internal validation is to be applied in further testing on view importance.

REFERENCES

[1] U. W. D. S. Team, "A Modern History of Data Science," *University of Wisconsin Data Science Degree*, 10-Jul-2017. [Online]. Available: https://datasciencedegree.wisconsin.edu/blog/history-of-data-science/.

[2] A. Samnani, "An Introduction to Supervised and Unsupervised Learning," *Medium*, 06-Dec-2019. [Online]. Available: https://aahil-samnani.medium.com/an-introduction-to-supervised-and-unsupervised-learning-4e5d2cdcec19.

[3] "Diabetes," *National Institute of Diabetes and Digestive and Kidney Diseases*. [Online]. Available: http://www.niddk.nih.gov/health-information/diabetes#:~:text=Diabetes%20is%20a%20disease%20that,prevent%20diabetes%20or%20manage%20it.

[4] Olcay Kursun, M. Muzaffer Ilhan, Ahmet Cinar, M. Erdem Isenkul, C. Okan Sakar, A. Esra Gursoy, Ertugrul Tasan, Oleg V. Favorov, "Analyzing Relations Among Measurements of Diabetic Neuropathy," at the International Conference on Applied Informatics for Health and Life Sciences, Kuşadası, Turkey, Oct. 19-22, 2014.

[5] "Diabetic neuropathy," *Mayo Clinic*, 03-Mar-2020. [Online]. Available: http://www.mayoclinic.org/diseases-conditions/diabetic-neuropathy/symptoms-causes/syc-20371580#:~:text=Diabetic%20neuropathy%20is%20a%20type,in%20your%20legs%20and%20feet.

[6] Z. T. Bloomgarden, "Diabetic Neuropathy," *Diabetes Care*, vol. 30, no. 4, pp. 1027–1032, 2007.

[7] *Pain and Sleep: Diabetic Neuropathy*, 2017. [Online]. Available: http://www.soundsleephealth.com/pain-and-sleep-diabetic-neuropathy/.

[8] D. E. Olson, M. K. Rhee, K. Herrick, D. C. Ziemer, J. G. Twombly, and L. S. Phillips, "Screening for Diabetes and Pre-Diabetes with Proposed A1C-Based Diagnostic Criteria," *Diabetes Care*, vol. 33, no. 10, pp. 2184–2189, 2010.

[9] "The A1C Test & Diabetes," *National Institute of Diabetes and Digestive and Kidney Diseases*. [Online]. Available: http://www.niddk.nih.gov/health-information/diagnostic-tests/a1c-test#:~:text=The%20A1C%20test%20measures%20the,level%20is%20below%205.7%20percent.

[10] M. F. Rabbi, K. H. Ghazali, O. Altwijri, M. Alqahtani, S. M. Rahman, A. Ali, K. Sundaraj, Z. Taha, and N. U. Ahamed, "Significance of Electromyography in The Assessment of Diabetic Neuropathy," *Journal of Mechanics in Medicine and Biology*, vol. 19, no. 03, p. 1930001, 2019.

[11] A. Moghtaderi, A. Bakhshipour, and H. Rashidi, "Validation of Michigan neuropathy screening instrument for diabetic peripheral neuropathy," *Clinical Neurology and Neurosurgery*, vol. 108, no. 5, pp. 477–481, 2006.

[12] "Take control of your brain health," *corticalmetrics*. [Online]. Available: http://www.corticalmetrics.com/.

[13] B. Troia, "Brain Gauge: A Cognitive Assessment Tool to Measure and Monitor Brain Health," *Quantified Bob*, 22-Apr-2020. [Online]. Available: https://www.quantifiedbob.com/brain-gauge-cognitive-measurement-tool/.

[14] Y. Yang and H. Wang, “Multi-view clustering: A survey,” *Big Data Mining and Analytics*, vol. 1, no. 2, pp. 83–107, 2018.

[15] M. Najafi, L. He, and P. S. Yu, “Error-robust multi-view clustering,” *2017 IEEE International Conference on Big Data (Big Data)*, 2017.

[16] S. Aksoy and R. M. Haralick, “Feature normalization and likelihood-based similarity measures for image retrieval,” *Pattern Recognition Letters*, vol. 22, no. 5, pp. 563–582, 2001.

[17] T. A. I. Team, “How, When, and Why Should You Normalize / Standardize / Rescale Your Data?,” *Towards AI - The Best of Tech, Science, and Engineering*, 29-May-2020. [Online]. Available: https://towardsai.net/p/data-science/how-when-and-why-should-you-normalize-standardize-rescale-your-data-3f083def38ff.

[18] D. M. J. Garbade, “Understanding K-means Clustering in Machine Learning,” *Medium*, 12-Sep-2018. [Online]. Available: https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1.

[19] K M. Khanum, T. Mahboob, W. Imtiaz, H. Abdul Ghafoor, and R. Sehar, “A Survey on Unsupervised Machine Learning Algorithms for Automation, Classification and Maintenance,” *International Journal of Computer Applications*, vol. 119, no. 13, pp. 34–39, 2015.

[20] J. J. Armstrong, M. Zhu, J. P. Hirdes, and P. Stolee, “K-Means Cluster Analysis of Rehabilitation Service Users in the Home Health Care System of Ontario: Examining the Heterogeneity of a Complex Geriatric Population,” *Archives of Physical Medicine and Rehabilitation*, vol. 93, no. 12, pp. 2198–2205, 2012.

[21] S. Thompson, M. E. Celebi, and K. H. Buck, “Fast color quantization using MacQueen’s k-means algorithm,” *Journal of Real-Time Image Processing*, vol. 17, no. 5, pp. 1609–1624, 2019.

[22] A. K. Jain and R. C. Dubes, “Clustering Methods and Algorithms,” in *Algorithms for clustering data*, Englewood Cliffs, NJ: Prentice-Hall, 1988, pp. 55–142.

[23] J. A. Bunge and D. H. Judson, “Data Mining,” *Encyclopedia of Social Measurement*, pp. 617–624, 2005.

[24] V. Cohen-Addad, V. Kanade, F. Mallmann-Trenn, and C. Mathieu, “Hierarchical Clustering: Objective Functions and Algorithms,” *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 378–397, 2018.

[25] M. Roux, “A Comparative Study of Divisive and Agglomerative Hierarchical Clustering Algorithms,” *Journal of Classification*, vol. 35, no. 2, pp. 345–366, 2018.

[26] T. Bock, “What is a Dendrogram?,” *Displayr*, 09-Dec-2020. [Online]. Available: http://www.displayr.com/what-is-dendrogram/.

[27] G. Ogbuabor and U. F. N, “Clustering Algorithm for a Healthcare Dataset Using Silhouette Score Value,” *International Journal of Computer Science and Information Technology*, vol. 10, no. 2, pp. 27–37, 2018.

[28] P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 1987.

[29] W. M. Rand, “Objective Criteria for the Evaluation of Clustering Methods,” *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

[30] H. Wang, Y. Yang, and B. Liu, “GMC: Graph-Based Multi-View Clustering,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 6, pp. 1116–1129, 2020.

[31] J. Liu, F. Cao, X.-Z. Gao, L. Yu, and J. Liang, “A Cluster-Weighted Kernel K-Means Method for Multi-View Clustering,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 4860–4867, 2020.