Parkinsons model prediction using data mining methodologies

***Abstract*—Parkinson's Disease is a neurodegenerative disorder of dopamine systems. It can be treated but the effectiveness of pharmacological options faces diminishing returns and anything that might reduce the occurrence of overmedication would have strong positive impact. [PREVIOUS PAPER] collected data comparing features of speech in 52 individuals diagnosed with Parkinson's and analyzed it using deterministic linear regression. In this work, we used various Data Mining tools to prepare and analyze the data speculatively in order to discover interesting information for use in Machine Learning or other subsequent analysis. K-means and K-means++ demonstrated 93.62 and 95.57 respectively while OPTICS showed 96.62. DBSCAN was not viable due to density and resulted in 100 indicating overfitting. Validation of clustering was performed using Cross Validation. Classification after Data Mining was performed using SVM (Support Vector Machine). We also approached the problem as an Association Rule problem and discovered nothing significant though we believe further analysis is warranted.**

***Index Terms*—Data Mining, k-means, k++, OPTICS, DBSCAN, Parkinson's Disease, SVM**

I. Introduction

Parkinson's Disease is a neurodegenerative disorder. It is usually diagnosed around 60 years of age and at onset typically presents with tremors and cognitive difficulties. It is not presently known what causes the initial damage, but once the Parkinson's Disease diagnosis is made, it continues until the patient's death.

An immense amount of research has gone into understanding and treating the disorder. Postmortem studies have discovered neurological damage begins in the substantia nigra and spreads to the basal ganglia. These two areas are known to use the dopamine neurotransmitter to regulate intentional muscle movement. As the damage progresses, the patient eventually loses all ability to control muscular function.

There are pharmacological treatments that can mitigate the symptoms of the disorder. They work by increasing the amount of dopamine available in the system, or increasing the neuronal sensitivity to dopamine. This is far from an exact science and every drug comes with a limited efficacy period as the systems adapt and damage progresses. Any increase in the efficacy period would bring additional relief to hundreds of thousands. In addition, overmedication can and does occur because the time between administering the drug and the resulting effect can take weeks. This brings new dangers as increasing the available dopamine too far results in the patient expressing schizophrenic symptoms.

Because of this, patients are monitored closely and clinicians are trained to take physiological measurements that are used to determine pharmacological treatment schedules. That requires direct contact with the patient and is a significant part of the cost of treating the disorder. A Tsanas, MA Little, PE McSharry, LO Ramig (2009) investigated using deterministic regression analysis of 16 components of speech to attempt to predict the clinician's assessment. These components include things like forms of jitter and shimmer.

In this paper we use non-deterministic methods to investigate the same problem space. We used two general approaches. On one side we used clustering techniques to discover interesting points to use as training models for machine learning algorithms. On the other, we re-imagined the data as transactions and did pattern analysis.

II. Related Works

[2]The results pointed to two to five patient clusters, with similarities among the age of onset and disease duration. The studies lacked the use of existing clustering evaluation metrics which points to a need for a thorough, analysis framework, and consensus on the appropriate variables to include in cluster analysis. Accurate cluster analysis may assist with determining if PD patients' symptoms can be treated based on a subgroup of features, if personalized care is required, or if a mix of individualized and group-based care is the best approach.

[1]The statistical analysis found that the tapping features could separate participants into three severity groups. Each group has different characteristics and could represent different PD severity based on the MDS-UPDRS I-II and PDQ-8 scores. Currently, the severity assessment of a movement disorder is based on clinical observation. Therefore, it is highly dependant on the skills and experiences of the trained movement disorder specialist who performs the procedure. We believe that any additional methods that could potentially assist with quantitative assessment of disease severity, without the need for a clinical visit would be beneficial to both the healthcare professionals and patients.

[3]They evaluate our method on a large PD dataset and present the results. The results showed that the proposed method is effective in predicting PD progression by improving the accuracy and computation time of the disease diagnosis. The method can be implemented as a medical decision support system for real-time PD diagnosis when big data from the patients is available in the medical datasets.

[4]The UPDRS (Unified Parkinson's Disease Rating Scale) has been used by taking into account both the motor and total

labels, and the best results have been obtained using a mixed multi-layer perceptron (MLP) that classifies and regresses at the same time and the most important features of the data obtained are taken as input, using an autoencoder. A success rate of 99.15 % has been achieved in the problem of predicting whether a person suffers from severe Parkinson's disease or non-severe Parkinson's disease. In the degree of disease involvement prediction problem case, a MSE (Mean Squared Error) of 0.15 has been obtained. Using a full deep learning pipeline for data preprocessing and classification has proven to be very promising in the field Parkinson's outperforming the state-of-the-art proposals.

[5] This technique is applied on a hospital database and analyzes the performance successfully and so is useful in giving accurate results and making an effective decision by the Ministry of Health in Iraq and related parties to find appropriate solutions.

[2] The existing literature highlights the necessity for a comprehensive analysis framework in Parkinson's disease (PD) research, particularly in the context of patient clustering. The studies discussed underscore the need for standardized clustering evaluation metrics to enhance the reliability of clustering outcomes. Moreover, there is a consensus on the importance of identifying appropriate variables for cluster analysis, which could contribute to a more accurate understanding of PD patient subgroups. The potential implications of such analyses extend to personalized care strategies, subgroup-specific symptom treatment, and optimizing a balance between individualized and group-based care approaches.

III. Proposed Solution

A. Data Normalization

After we have the dataset ready, we go for the normalization step. Normalization is a preprocessing technique used to scale numeric features within a specific range, typically between 0 to 1. The main aim behind normalizing data it guarantees that the data is appropriately scaled for machine learning algorithms, improving the accuracy and effectiveness of the subsequent analytical steps.

In our project, we have used Min-Max Normalization for scaling of the data. Also, to get the meaningful data we have excluded the first 3 columns from our consideration. This adjustment ensures that the resulting dataset only contains the relevant, normalized numeric features. This step is crucial for preparing the data for subsequent analysis, improving the performance of clustering and regression algorithms.

B. Clustering methods and Visualization Clustering Results:

- K-Means
  Initialization: In K-Means, the algorithm starts with an initial guess for the cluster centroids, typically chosen randomly from the data points. Random initialization, however, can result in suboptimal convergence and sensitivity to the initial placement of centroids.
  - Algorithm Steps:
    - Randomly initialize K centroids.
    - Assign each data point to the nearest centroid.
    - Update the centroids based on the mean of the data points assigned to each cluster.
    - Repeat steps 2 and 3 until convergence.

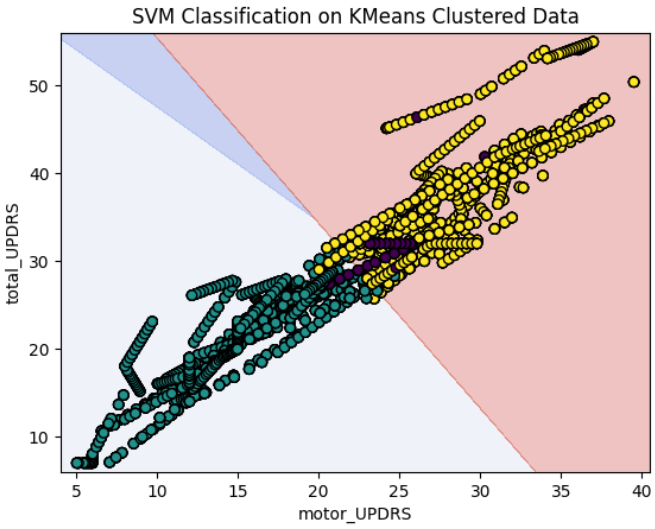

Fig. 1. K-means scatter plot

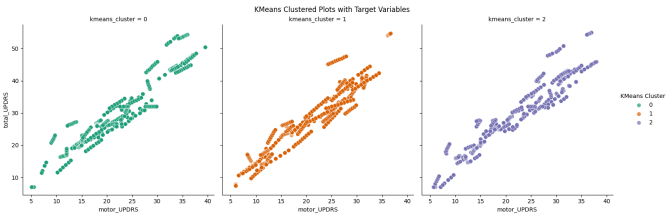

Fig. 2. K-Means scatter plot for target variables

- Kmeans++
  Initialization: K-Means++ improves upon the initialization step by selecting centroids that are more likely to be distant from each other, resulting in better convergence and more robust solutions.
  - Algorithm Steps:
    - Randomly select the first centroid from the data points.
    - . For each subsequent centroid, choose the next one from the remaining data points with probability proportional to the square of the distance from the point to the nearest existing centroid.
    - Assign data points to the nearest centroid.
    - Update centroids based on the mean of the data points assigned to each Repeat steps 3 and 4 until convergence. cluster.
    - Repeat steps 3 and 4 until convergence.
  - Comparision:
    K-Means++ tends to converge faster and provides more accurate and stable results compared to K-Means, especially when the number of clusters (K)

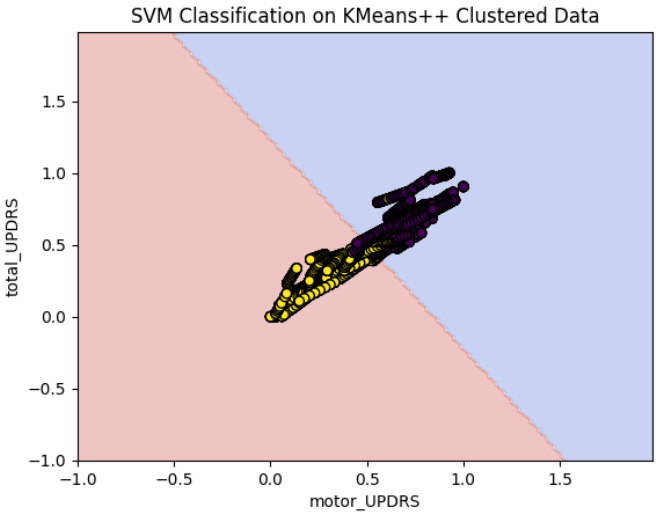

Fig. 3. K-Means++ scatter plot

is relatively large. K-Means++ reduces the risk of converging to a local minimum, making it more robust for various datasets. While K-Means++ is generally preferred, K-Means might still be suitable for smaller datasets or when computational resources are limited.

DBSCAN:
DBSCAN is a density-based clustering algorithm that divides a dataset into groups based on the density of data points in the feature space. Unlike partitioning methods like K-Means, DBSCAN does not require the number of clusters as an input and can discover clusters of arbitrary shapes. It is particularly effective in identifying clusters embedded in noise and handling varying cluster shapes and sizes.Algorithm Steps:

* · Parameter Definition:
  Epsilon(): The maximum distance between two data points for one to be considered in the neighbourhood of the other.
  MinPts: The minimum number of data points required to form a dense region (including the data point itself).
  · Initialization: Randomly select a data point that has not been visited.
  · Density-Based Exploration:If the number of data points in the () - neighborhood of the selected point is greater than or equal to MinPts, form a dense region and label all points within the () - neighborhood as part of the same cluster. If a point has fewer than MinPts neighbors but is within the () - neighborhood of another point that is part of a dense region, the point is considered a border point and assigned to that cluster. If a point has fewer than MinPts neighbors and is not within the () - neighborhood of any dense region point, it is labelled as noise.
  · Expand Clusters: Repeat the process for unvisited data points until all data points have been visited.

– OPTICS Algorithm (Ordering Points To Identify the Clustering Structure)

  * Introduction:
    OPTICS is a density-based clustering algorithm that extends the concept of DBSCAN. It aims to identify clusters of arbitrary shapes and sizes in a dataset while addressing some limitations of DBSCAN. OPTICS introduces the concept of reachability and produces an ”ordered” list of points, providing a more flexible representation of the cluster structure.

    · Algorithm Steps:
    · Parameter Definition:
    Epsilon(): The maximum distance between two data points for one to be considered in the neighborhood of the other. MinPts: The minimum number of data points required to form a dense region (similar to DBSCAN).
    · Initialization: Initialize an empty priority queue to store the reachability distances.
    · Core Distance: For each data point, calculate the core distance, which is the distance to the MinPts-th nearest neighbor. This reflects the local density around the point.
    · Reachability Distance: For each data point, calculate the reachability distance for every point in its ()-neighborhood. The reachability distance is the maximum of the core distance of the data point and the distance between the data point and the considered neighbor.
    · Building Reachability Plot: Populate the priority queue with the reachability distances, creating a reachability plot that reflects the density-based structure of the dataset.
    · Cluster Extraction: Traverse the reachability plot, identifying valleys in the plot as potential cluster boundaries. Determine clusters based on the reachability distances and core distances.

C. Classification

– SVM Classification on KMeans Clustered Data:

  * The SVM classifier is trained on the features (‘motor_UPDRS‘ and ‘total_UPDRS‘) and labels (‘Cluster‘) obtained from KMeans clustering
  * The accuracy and confusion matrix are computed to evaluate the performance of the SVM classifier.

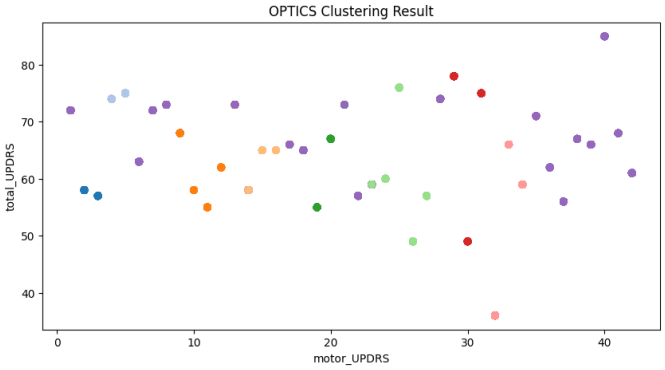

Fig. 4. Optics scatter plot

  - A scatter plot is generated with the SVM decision boundary to visualize the classification results.

- SVM Classification on KMeans++ Clustered Data: Similar to the first case, this performs SVM classification on data clustered using KMeans++.
- Random Forest is an ensemble learning algorithm that operates by constructing a multitude of decision trees during training and outputting the mode of the classes for classification problems. It builds each tree using a random subset of the features and introduces randomness in both the data and the feature selection, leading to a more robust and accurate model. Random Forest is capable of handling complex relationships in data and is less prone to overfitting.
  The provided code implements Random Forest Classification within distinct clusters identified by DBSCAN. Initially, the dataset is divided into subsets corresponding to unique cluster labels assigned by DBSCAN. Subsequently, within each cluster, a binary target variable is created by applying a threshold to the 'motor_ UPDRS' variable specific to that cluster. Features and the binary target variable are then meticulously prepared, excluding any irrelevant columns. The dataset is further partitioned into training and testing sets to facilitate model training and subsequent evaluation. Following this, a Random Forest Classifier is instantiated and trained on the designated training set within each individual cluster. The trained model is then employed to make predictions on the corresponding test set, and the resulting classification accuracy is computed and printed for each distinct cluster. This approach allows for a tailored assessment of the classification model within the context of different clusters, offering valuable insights into the model's performance across various subsets of the data.
- SVM Classification on OPTICS Clustered Data: We used Support Vector Regression (SVR) to predict 'motor_ UPDRS' values based on a set of features. SVR is a type of Support Vector Machine adapted for regression tasks, and in this context, it seeks to find a regression function that best fits the data. The model is trained on a subset of the data, and its performance is evaluated using metrics such as Mean Squared Error and R-squared. SVR is particularly useful when dealing with non-linear relationships and complex patterns in the data. Adjusting the kernel type and regularization parameter allows for customization to different data characteristics.

D. Frequent Itemsets and Association Rules

For the analysis of the data as Association Rules, first the data needed to be re-imagined. After normalization, the data was put through a second encoding process. Three methods for encoding were used. The first was Equal Size indicating equal splits of 0.2 in the normalized date. The second was Bell Curve Like meaning the ratios more closely represented standard deviations. The third was based on how extreme the data was. Anything above or below a small threshold was considered outlier and anything outside of one standard deviation different. This was calculated based on magnitude. The data was encoded separately from the source of truth. Different combinations of the two, as well as various threshold values were performed.

DATASET

The data used for our analysis came from STUDY. It consists of 22 columns. Two are identifiers to distinguish patient and specific trial. Two of the data columns are demographical data: age as an integer and binary representation of sex where 1 is female. Two columns are clinician measurements of symptoms and are considered source of truth. The remaining 16 columns encode jitter, shimmer, and 5 other numeric values which describe characteristics of voice thought to be related to Parkinson's Disease symptom severity. These values are continuous positive real numbers. All values were normalized to [0..1]. The data contains approximately 200 trials per patient for 42 patients with no null values, 5875 total records. The clinician measurements were gathered using the NIH endorsed process for measuring Parkinson's symptoms, the Unified Parkinson's Disease Rating Scale (UPDRS). Trained and certified providers use its three general categories to produces continuous values conducive to normalization.

The special-built device used to analyze patient speech gathers 16 distinct features. Five describe Jitter and 6 describe Shimmer. These measure potentially interesting fundamental variations in speech patterns, frequency and amplitude, respectively. NHR and HNR are a ratio if noise to various tonal components. RPDE, DFA, and PPE are additional measurements which are also potentially interesting.

The data set was downloaded from UCI Machine Learning Repository in November of 2023. They were originally collected by Athanasios Tsanas, Max A. Little, Patrick E. McSharry, Lorraine O. Ramig for Accurate telemonitoring of Parkinson's disease progression by non-invasive speech tests published in 2009 by IEEE Transactions on Biomedical Engineering. Any use or transferal of the data should include the above citation and reference.

IV. Cross Validation

We used cross validation for evaluating the SVM classification models which is applied on clustering methods (Kmeans, Kmeans++, Optics, DBScan).
Cross-validation is a statistical technique used in machine learning and model training to assess how well a model will generalize to an independent dataset. The basic idea is to partition the dataset into multiple subsets, train the model on some of these subsets, and then evaluate its performance on the remaining subset. This process is repeated multiple times, and the performance metrics are averaged over the different folds to provide a more robust estimate of the model's performance.
The most common form of cross-validation is k-fold cross-validation, where the dataset is divided into k subsets or folds. The model is trained k times, each time using k-1 folds for training and the remaining fold for validation. This process ensures that every data point is used for both training and validation exactly once.
Here's a step-by-step overview of the k-fold cross-validation process:

- Data Splitting:
  - The dataset is randomly partitioned into k equally sized folds.
- Model Training and Validation:
  - The model is trained on k-1 folds (training set).
  - The trained model is validated on the remaining fold (validation set).
- Performance Evaluation:
  - Performance metrics (e.g., accuracy, precision, recall) are recorded for each fold.
- Iteration:
  - • The dataset is randomly partitioned into k equally sized folds.
- Performance Summary:
  - Steps 2 and 3 are repeated k times, with a different fold used as the validation set in each iteration.

Common values for k are 5 or 10, but the choice can depend on the size of the dataset and the specific requirements of the analysis.
Cross-validation helps to provide a more accurate estimate of a model's performance compared to a single train-test split. It is particularly useful when the dataset is limited, and a reliable assessment of model performance is crucial. Cross-validation helps detect issues like overfitting or underfitting and allows for better hyperparameter tuning.

- K-Fold Cross-Validation:
  - **Description:** The dataset is divided into k subsets or folds. The model is trained k times, each time using k-1 folds for training and the remaining fold for validation.
  - **Advantages:** Provides a good balance between computational efficiency and reliable performance estimation
- Stratified K-Fold Cross-Validation:
  - **Description:** Similar to k-fold cross-validation, but it ensures that each fold maintains the same class distribution as the original dataset. This is particularly useful when dealing with imbalanced datasets.
  - **Advantage:** Ensures that each fold represents the overall class distribution.
  - **Disadvantages:** Can be computationally more expensive

Results & analysis

In our dataset evaluation, K-means and K-means++ demonstrated 93.62 and 95.57 respectively, with K-means++ exhibiting improved results attributed to its better initialization. Conversely, DBSCAN unexpectedly yielded 100% accuracy, indicating overfitting. This arises from DBSCAN's inability with varying density datasets, as it employs a fixed radius for cluster formation.
The inherent challenges of DBSCAN in handling data with diverse density patterns prompted our exploration of an alternative. OPTICS, with its emphasis on minimum points and adaptive density, outperformed DBSCAN. This aligns with expectations, showcasing OPTICS as a robust solution for datasets characterized by fluctuations in density. The study underscores the importance of algorithm selection, emphasizing the need for methods tailored to the dataset's specific characteristics.

Models	Accuracy	Cross Validation	Stratified Cross validation
KMeans	93.62%	0.92	0.92
KMeans++	95.57%	0.95	0.96
Optics	96.62%	0.96	0.97
DBSCAN	100%	Nan	Nan

TABLE I
Clustering Evaluation Results

These findings offer valuable insights into the nuances of clustering algorithms and emphasize the significance of adapting methods to the inherent complexities of the data at hand. Such considerations are crucial in ensuring accurate and meaningful clustering results.
The Association Rules analysis did not provide any useful results though it is plausible that more analysis could

yield potentially valuable results. We recommend further analysis.

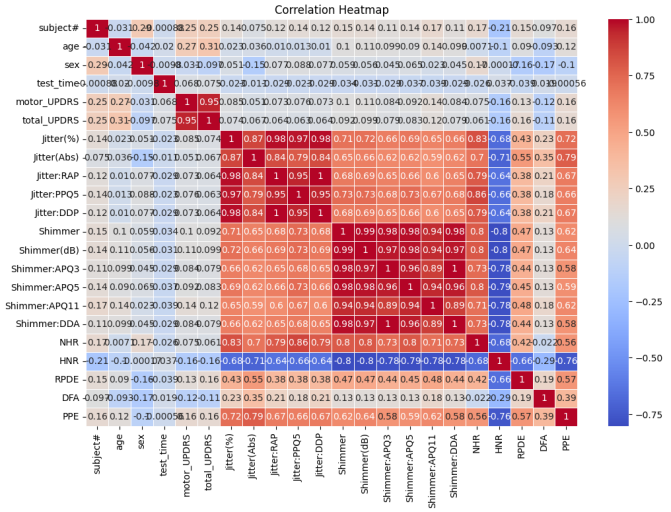

Fig. 5. Correlation Graph

V. Conclusion

We conclude that there is value in performing analysis using Data Mining techniques. We believe that a larger data set would provide valuable insight into individualized treatment. Additional clustering analysis should focus on OPTICS and related algorithms. We also believe that the novel approach of re-imaging the data as transactions and using an Association Rule approach should take priority over clustering techniques. Using that framework, additional analysis could be performed by considering it from a longitudinal perspective where time trial x0 relates to time trial x1.

References

[1] Surangsrirat D, Sri-Iesaranusorn P,Chaiyaroj A, Vateekul P, Bhidayasiri R. Parkinson's disease severity clustering based on tapping activity on mobile device.

[2] Hendricks, R. M., & Khasawneh, M. T. (2021).A Systematic Review of Parkinson's Disease Cluster Analysis Research.

[3] Mehrbakhsh Nilashi, Othman Ibrahim, Sarminah Samad, Hossein Ahmadi, Leila Shahmoradi, Elnaz Akbari, An analytical method for measuring the Parkinson's disease progression: A case on a Parkinson's telemonitoring dataset, Measurement,https://doi.org/10.1016/j.measurement.2019.01.014.

[4] García-Ordás, María & Benítez-Andrades, José & Aveleira, Jose & Alija-Perez, Jose & Benavides, Carmen. (2023). Determining the severity of Parkinson's disease in patients using a multi task neural network. Multimedia Tools and Applications. 10.1007/s11042-023-14932-x.

[5] Israa Ali Alshabeeb, Nidaa Ghalib Ali, Saba Abdulameer Naser, Wafaa M. R. Shakir. A Clustering Algorithm Application in Parkinson Disease based on kmeans Method

[6] Hendricks, R., & Khasawneh, M. (2021). Cluster Analysis of Categorical Variables of Parkinson's Disease Patients. Brain sciences, 11(10), 1290. https://doi.org/10.3390/brainsci11101290.

[7] Ram Deepak Gottapu, Cihan H Dagli,Analysis of Parkinson's Disease Data,Procedia Computer Science,Volume 140,2018,Pages 334-341,ISSN 1877-0509, https://doi.org/10.1016/j.procs.2018.10.306.

[8] Tucker, C., Han, Y., Nembhard, H. B., Lewis, M., Lee, W. C., Sterling, N. W., & Huang, X. (2015). A data mining methodology for predicting early stage Parkinson's disease using non-invasive, high-dimensional gait sensor data. IIE transactions on healthcare systems engineering, 5(4), 238–254.

[9] Dinov, I. D., Heavner, B., Tang, M., Glusman, G., Chard, K., Darcy, M., Madduri, R., Pa, J., Spino, C., Kesselman, C., Foster, I., Deutsch, E. W., Price, N. D., Van Horn, J. D., Ames, J., Clark, K., Hood, L., Hampstead, B. M., Dauer, W., & Toga, A. W. (2016). Predictive Big Data Analytics: A Study of Parkinson's Disease Using Large, Complex, Heterogeneous, Incongruent, Multi-Source and Incomplete Observations. PloS one, 11(8), e0157077. https://doi.org/10.1371/journal.pone.0157077

[10] Zhang, J. Mining imaging and clinical data with machine learning approaches for the diagnosis and early detection of Parkinson's disease. npj Parkinsons Dis. 8, 13 (2022). https://doi.org/10.1038/s41531-021-00266-8.

We declare that we have completed this assignment completely and entirely on our own, without any consultation with others. We have read the UAB Academic Honor Code and understand that any breach of the Honor Code may result in severe penalties.

We also declare that the following percentage distribution ***faithfully*** represents individual group members' contributions to the completion of the assignment

Name	Overall Contribution (%)	Major work items completed by me	Signature or initials	Date
Veeresh Kondapaneni	20%	Kmeans, DBscan, Report	VK	12-03-2023
Parth Bhatt	20%	Kmeans++, SVM, Report	PB	12-03-2023
Briyan Lear	20%	Data analysis , Dataset selection , data preprocessing , Report, PPt	BL	12-03-2023
Priya Dobariya	20%	Cross validation, PPT,Report	PD	12-03-2023
Srushti Nayak	20%	Cross validation, Optics ,Report	SRN	12-03-2023