Romil Shah NUID: 001673595
Introduction to Data Mining and Machine Learning
Professor Nik Bear Brown
Final Project
Summer 2016

Parkinson's disease detection based on Speech and Gait signal analyses using Machine Learning algorithms

Introduction:

Parkinson's disease is a central nervous system disorder that affects the motion of the human body that include shaking, rigidity, slow movement and difficulty in walking [1]. As the disease advances, thinking and behavioral problems start taking place that affects a person's psychology and ultimately leads to depression. Parkinson's disease affects about 200,000 people on an average annually. It is an incurable disease but can be contained with treatment [2]. Parkinson's disease can be diagnosed on the basis of the symptoms that include muscle spasms, stiffness, movement coordination problems, dizziness, impaired speech, voice box spasms etc. Ways to diagnose it is based on the patient's medical history as well as neurological examination [3]. There is no lab test that will clearly identify it and brain tests are used to rule out other diseases. Traditional diagnosis of Parkinson's disease involves a neurological history of the patient and observation of the motor skills in various situations. Since there is no definitive laboratory test to diagnose Parkinson's disease, diagnosis is often difficult, particularly in the early stages when motor effects are not yet severe. Monitoring progression of the disease over time requires repeated clinic visits by the patient. There is no cure, but pharmacological treatment to manage the condition includes dopaminergic drugs. But all the identification is mainly dependent on the medical history. This is why I decided to work on the speech pattern and gait pattern recognition on basis of which Parkinson patients and healthy patients can be classified as the motor and speech are the parts that have the most effects.

There are several researches based on the diagnosis of Parkinson's disease based on medical instruments or based on history of the patient. The most recent is the one using neuroimaging techniques. There have been a few notable advances in the use of machine learning for the detection and classification of patients and healthy beings. One of the notable work is formation of new machine learning algorithms especially for prediction of Parkinson's [4].

One such research that has caught my attention is where they are using sensors on the patient's body and analyzing the pattern of the patient whilst walking [5]. Here they are analyzing the parts when the gait 'freezes' i.e. when the patient has problems whilst walking and suddenly stops or has a balance problem. This is when I planned to use the MYO Armband and try analyzing the gait pattern myself. They have also devised an algorithm called the FOG detection or Freezing of Gait detection.

I had devised a technique using the 'MYO Armband' and hacked it to be used for walk pattern recognition using real time data training and detection of an irregularity in the walking pattern on basis of which

Parkinson can be classified [6]. I would like to take this further and train the dataset properly to be used in classification of healthy and diseases patients.

Initially I had an idea of using the speech signals but after going through a research paper specially based on the usage of data to analyze patients, I decided to use the gait signal analysis too. There are various tools available for investigating the time-series data of the patients obtained at clinical research centers. For the linear and nonlinear analysis of signals, using linear and nonlinear dynamic parameters, several software packages such as CDA (Chaos Data Analyzer Programs), NLyzer (Nonlinear Analysis in Real Time) TISEAN (Nonlinear Time Series Analysis), WFDB Software, MATLAB Software and Physio Toolkit Software are available [7]. I am planning to use Machine Learning techniques to classify the walk pattern and speech pattern data in order to obtain a clear classification between the type of patients using the inbuilt functions and toolboxes available in MATLAB and Python. Using ANN, Fuzzy Logic, SVM, K-Means, AdaBoost etc. on the EMG and EEG Data of the patients we can classify them and do an early diagnose.

Using the methods, I expect to discover an easier and faster way of diagnosing Parkinson's disease. This way an initial way of disease identification can be done based on simple EEG and EMG data. This can save a lot of time, efforts and money. A patient's health can be analyzed by saving long-term costs. The expected results will help the research community work more on learning the various patterns that can be observed in the different data collected from sensors for different types of patients. This can help the community grow more complex algorithms to identify the difference in more efficient way. Also a web-based or mobile-application based real time data can be fetched, analyzed and prediction can be made after using the training dataset to train the models.

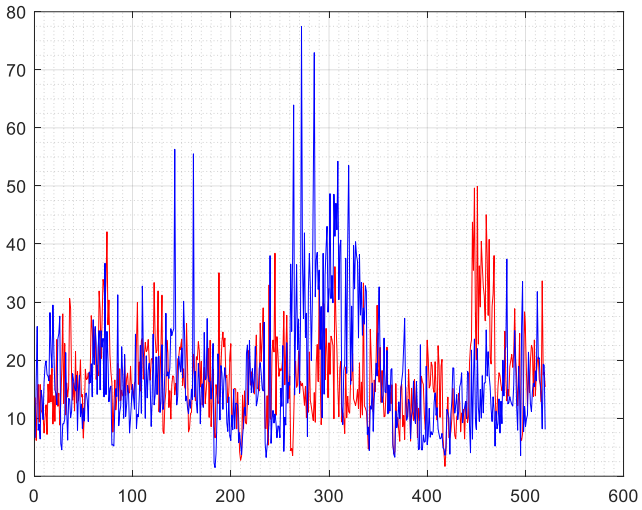
Whilst working on the dataset there might be several complications that may arise due to various factors. For example, multiple datasets for a given disorder often exist, collected from different sources and using slightly different features [8]. Combining them in some effective way into a large, cohesive dataset would result in a more robust and well-trained learner.

I am planning to use the available datasets on Parkinson's disease based on walking patterns [9] and that of the EMG [10], EEG and speech signals [11]. I would test the different algorithms on the available dataset and test them on the live data. I am also planning to make a web application that would help in taking in real time data from around the world to be analyzed and as more data rolls in, the better it would be as the accuracy of the classification and prediction would increase.

Comparison of different classifiers for Speech Data:

There are various classifiers that help train and learn a model. To decide upon a machine learning algorithm to choose a classification problem, the best bet is to test out a couple of different algorithms. The final decision comes upon as a trade-off between speed, complexity and accuracy.

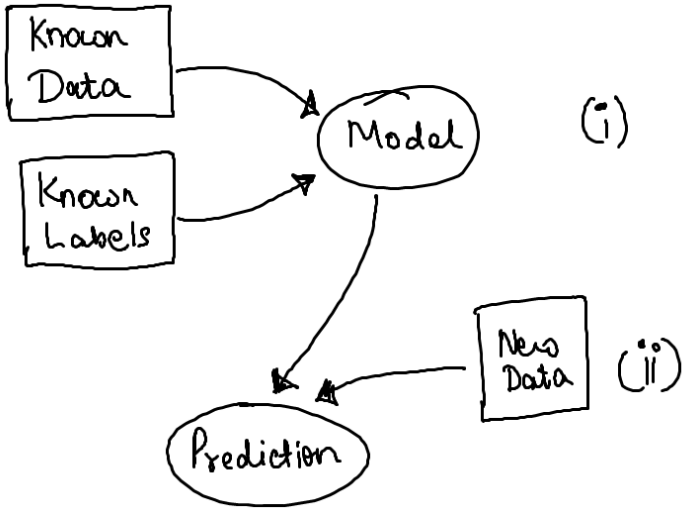
If the training data set is small, high bias/low variance classifiers are a better bet than low bias/high variance classifiers as the latter might end up overfitting. As the training dataset becomes more then the use of latter is better.



Time series of Feature 4: Jitter (rap)
RED: Parkinson's Patient
BLUE: Healthy

A. Supervised Learning:

The task of supervised learning is building of a model for predictions based on evidence. The adaptive algorithms learn the pattern based on the observations and identifies the methods. There are 2 basic processes involved in this:



Supervised learning is based upon training of a dataset based on known labels and learning the pattern that is concerned with the dataset. Based on this when a new data is obtained [14]. Thus supervised data is split into two categories:

(i) Classification – where a label is assigned to each of the classes which is usually true or false based

(ii) Regression – where the aim is to fit the model to the trained model in order to predict newer observations based on data

The different algorithms inside supervised learning are:

a. Support vector machine –

This algorithm constructs a hyperplane or set of hyperplanes where the classification and regression are carried out. The training data is specified as linear or nonlinear classifier.

Considering xi as the d-dimensional vector and y_i as the labels associated with them with '-1' and '+1' for simple model and more labels for multi class. Here w is the normal vector to the hyperplane and b is the offset along the vector.

Considering Si variable where

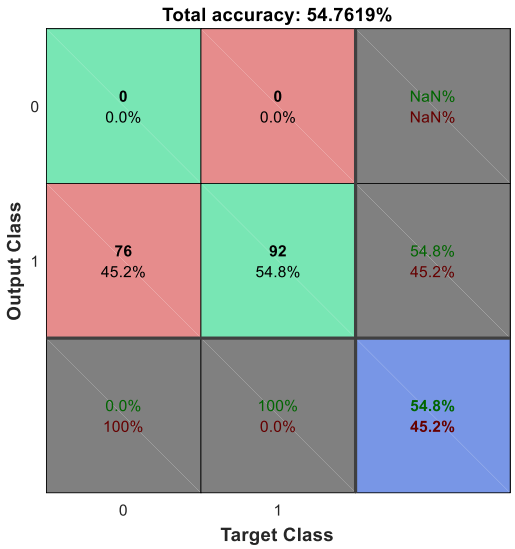
Si = max (0, 1 – y_i (w.x_i + b)), iff S_i is smallest nonnegative number with:

y*i* * (w.x_i + b) >= 1 – S_i

Thus the equation reduces to:

$$\min \frac{1}{n} \sum_{i=1}^{n} (Si + ||w||^2)$$

$$\text{s.t.} \sum_{i=1}^{n} ci * yi = 0$$



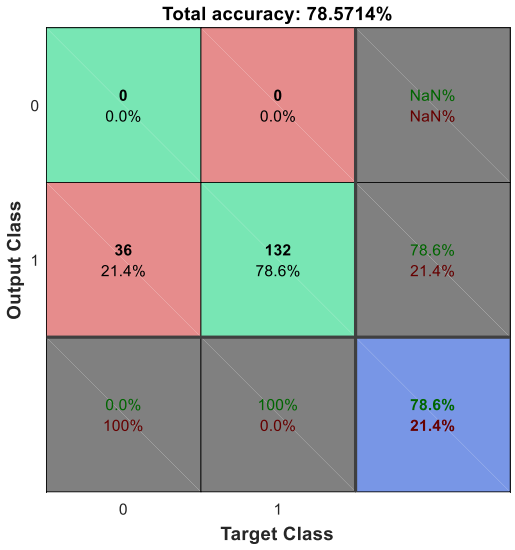
b. Naïve Bayes Classifier –

A conditional probability model where it assigns probabilities: $p(C_k \mid x_1, x_2 .. x_n)$.

Thus the conditional probabilities are mentioned as:

$$p(Ck \mid x1, x2..xn) = \frac{1}{Z} p(Ck) \prod_{i=1}^{n} p(xi \mid Ck)$$

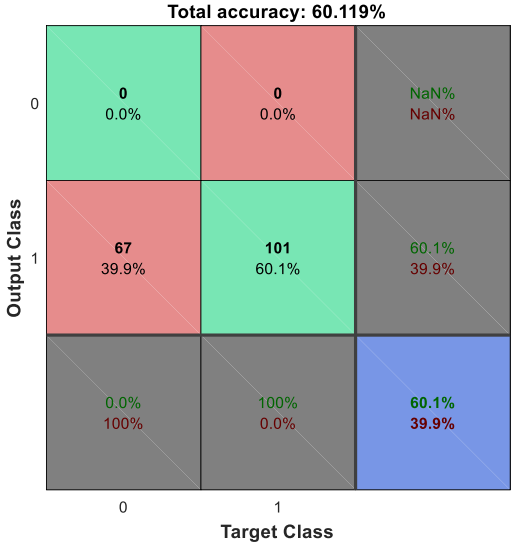
Where Z = p(x) is a scaling factor for k possible classes. Naïve Bayes is particularly useful for supervised learning where each distribution can be independently estimated as a one dimensional distribution thus removing the curse of dimensionality.



c. k-NN (k nearest neighbors) –

In k-NN the classification is done on basis of a majority vote by its neighbors. The regression gives the output as the property value of the object.

The Mahalanobis distance is computed and the labeled examples are ordered by increasing distance. Optimal 'k' is found for the nearest neighbors.



B. Unsupervised Learning:

This is a type of machine leaning algorithm where the labels are unknown and the dataset is analyzed on basis of that.

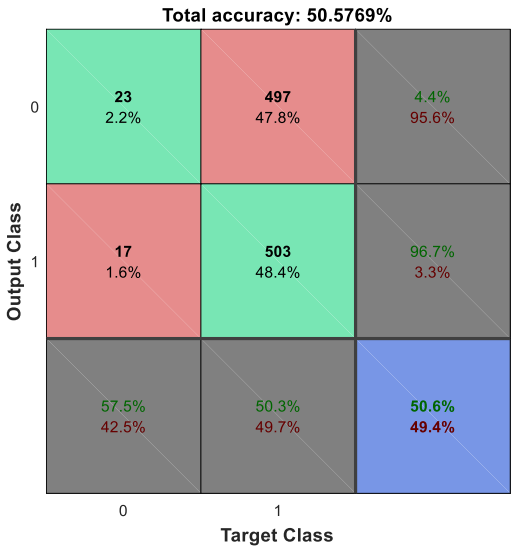
a. K-Means –

A set of observations with each being d-dimensional vector, this algorithm focuses on partitioning the n-observations into k<=n sets.

Assume the observations {x1, x2, …, xn} and the k-sets S = {S1, S2, …, Sk} so as to minimize the SSE within each cluster [15].

Thus the objective is as follows:

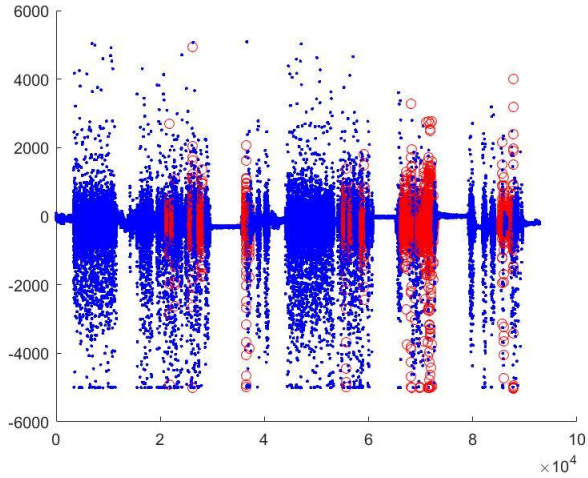
$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x \to Si} \left| |x - ui| \right|^2$$



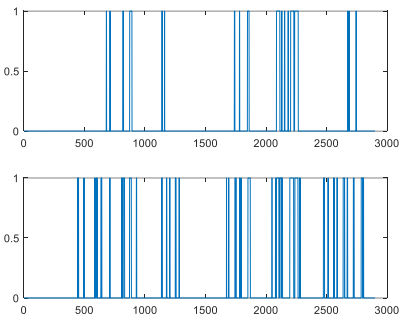
Comparison of different classifiers for Gait Data

The gait dataset is a large dataset with unknown labels. I am trying to obtain the performance with k-NN and k-Means in order to compare the supervised and unsupervised learning algorithm respectively.

The high peaks show the freezing data i.e. the time when the patient freezes during the walk. These data are obtained from sensors across ankles, legs and thighs [13].



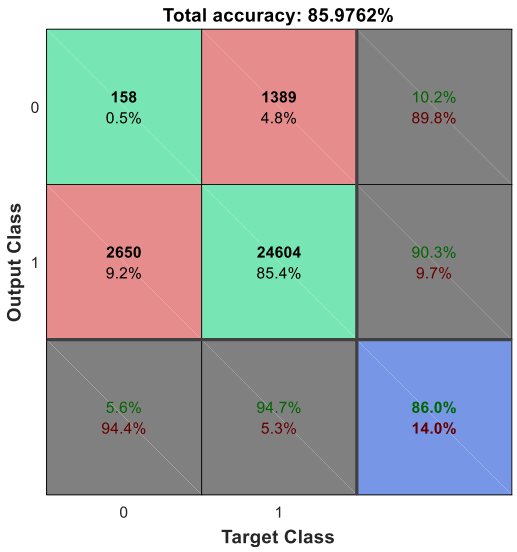
Time series of Feature 1: Ankles (horizontal acceleration)
RED: Freezing points
BLUE: Normal walk



The main aim to understand from this is how the two kinds of algorithm perform when the labels are absent and also the data is large. I have tried implementing k-NN and k-Means and comparing between them.

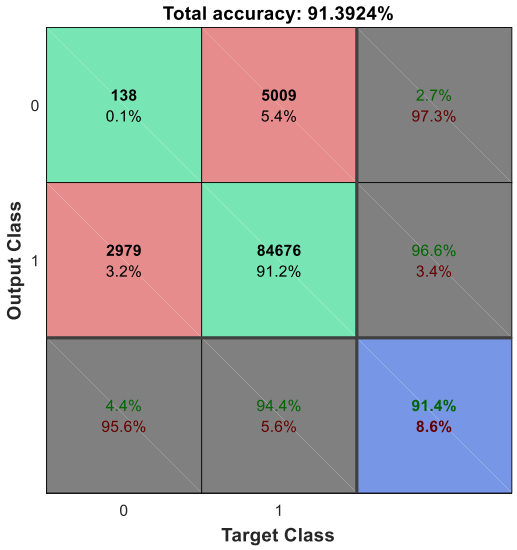
A. Supervised Learning:

a. k-NN –



B. Unsupervised Learning:

a. K-Means –



Here I also tried corrupting the data using 'NaN' at random places and then trying to find the confusion matrix using k-Means ONLY and using k-Means after applying PCA to recover the unknown values. Thus PCA is used to recover missing data after which the k-Means can be performed and results can be improved substantially.

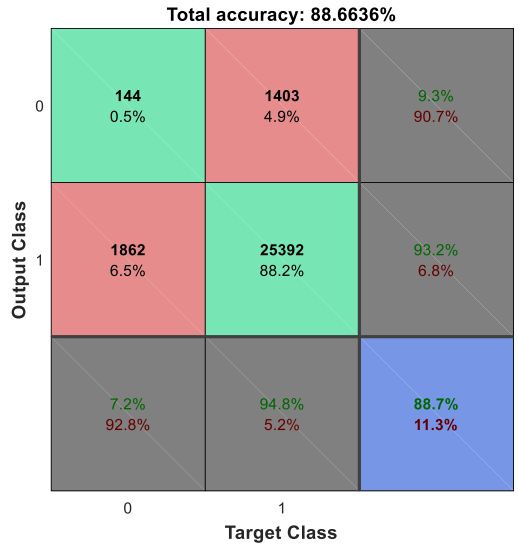
PCA: The most common derivation of PCA is in terms of a standardized linear projection which maximizes the variance in the projected space. For a set of observed d-dimensional data vectors $\{y_i\}$, i = {1,2..n}, the k principal axes $\{w_j\}$, j = {1,2..n} are those orthonormal axes onto which the retained variance under

projection is maximal. It can be shown that the vectors w_j are given by the k dominant eigenvectors (i.e. those with the largest associated eigenvalues l_j) of the sample covariance matrix:

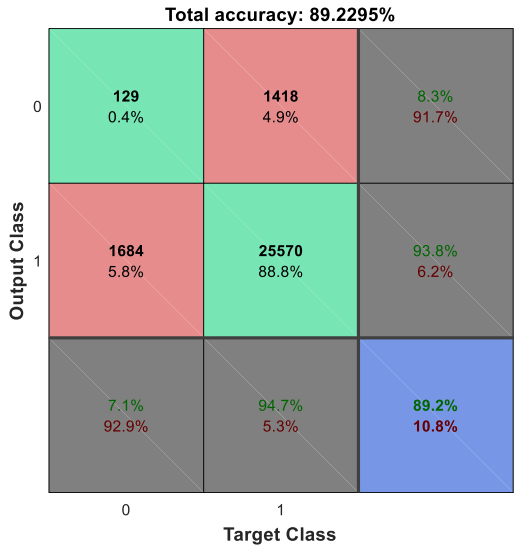
$$\mathbf{S} = \sum_{i=1}^{n} \frac{(\boldsymbol{y}_i - \bar{\boldsymbol{y}})(\boldsymbol{y}_i - \bar{\boldsymbol{y}})^\top}{n}$$

The property of PCA is that it minimizes the squared reconstruction error which is the property that is exploited here.

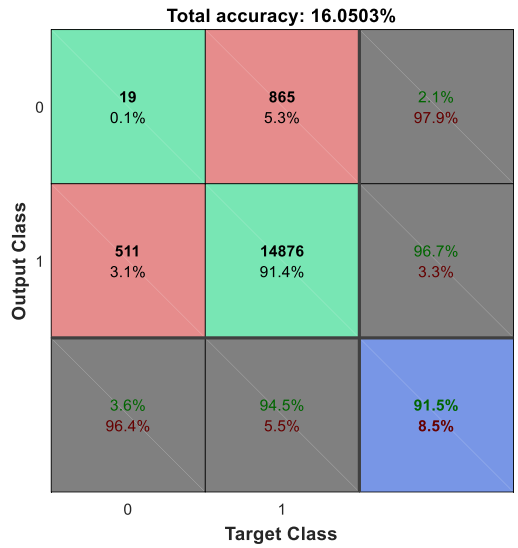
Missing Data k-NN



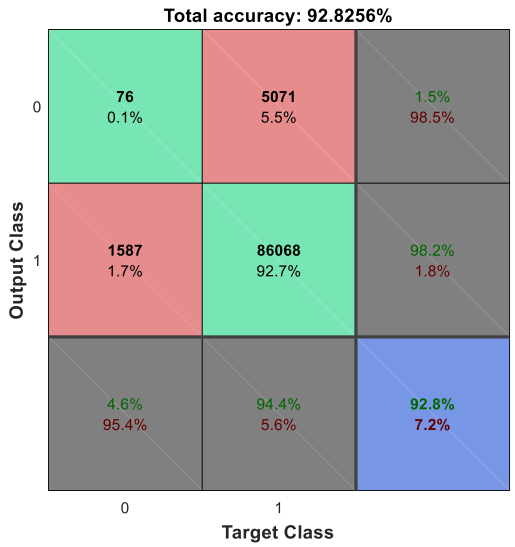
Recovered Data k-NN



Missing Data k-Means



Recovered Data k-Means



Conclusion:

Whilst using the smaller and labeled dataset, the supervised learning methods performed better than the unsupervised learning. This is because the labels and the features were already given and the algorithms had to learn them based upon the mathematical formulation. Hence the confusion matrix efficiency was high for supervised learning especially for Naïve.

Using unlabeled and large dataset, the unsupervised learning performed better in terms of confusion matrix and time complexity. The supervised learning used more time to compute and give results.

Furthermore, I corrupted the data by 30% and tried performing the algorithms. After that I tried recovering the missing data using PCA and then performing the different algorithms. It was found that PCA improved the efficiency manifold before applying any learning algorithm. PCA helps in recovering missing data points. Moreover, in such cases also the unsupervised learning performed better than supervised learning.

Labeled Small Dataset size: 1040 x 26

Unlabeled Large Dataset size: 92803 x 3

The computation time taken by each on different type of dataset is as follows:

Computation Time	Labeled Small Dataset	Unlabeled Large Dataset
SVM	45.0857	
Naïve	1.4973	
k-NN	0.6692	1.3866
k-Means	0.4790	0.8486
PCA + Naïve		1.5694
PCA + k-Means		1.4219

Noise – Accuracy relation for different type of dataset is as follows:

(A) Labeled small dataset: Speech Dataset

Noise	0%	1%	2%	5%
SVM	54.7619	38.0852	73.2143	63.3333
Naïve	78.5714	79.1667	76.7857	67.2619
k-NN	60.119	60.119	60.119	57.1429
k-Means	50.5769	50.3846	48.5577	41.3462

(B) Unlabeled large dataset: Gait Signals

Noise	0%	1%	2%	5%
k-NN	85.9762	85.8651	85.9901	84.6984
k-Means	91.3924	88.65	86.1037	75.9703
PCA + Naïve	89.2295	85.6602	86.1567	85.6637
PCA + k-Means	92.8256	91.401	91.4851	87.0423

It is clearly seen from the above tables that for labeled dataset, SVM performs better when noise is present upto 2-5% compared to others. But the performance if k-NN is almost steady upto 2% noise. As far as computational time is concerned, k-Means and k-NN outperform others.

For the unlabeled dataset, the PCA enabled Naïve and k-Means perform better than the others and from them PCA enabled k-Means performs way much better even when the noise is upto 5%. In terms of computational time, we might see that k-Means is fastest but doesn't perform well in presence of noise. Here PCA enabled k-Means performs better.

References:

[1] https://en.wikipedia.org/wiki/Parkinson%27s_disease
[2] https://www.gstatic.com/healthricherkp/pdf/parkinson_s_disease.pdf
[3] "Parkinson's disease: clinical features and diagnosis", http://www.ncbi.nlm.nih.gov/pubmed/18344392
[4] "New machine-learning algorithms for prediction of Parkinson's disease", https://www.researchgate.net/publication/234131546_New_machine-learning_algorithms_for_prediction_of_Parkinson's_disease
[5] "Wearable assistant for Parkinson's disease patients with the freezing of gait symptom", http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5325884
[6] http://devpost.com/software/myowalk
[7] "Data Processing for Parkinson's Disease: Tremor, Speech and Gait Signal Analysis", http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6150330
[8] "A Machine Learning Approach to Diagnosis of Parkinson's Disease", http://scholarship.claremont.edu/cgi/viewcontent.cgi?article=1784&context=cmc_theses
[9] https://physionet.org/physiobank/database/gaitndd/
[10] https://physionet.org/physiobank/database/tremordb/
[11] https://archive.ics.uci.edu/ml/datasets/Parkinsons
[12] https://archive.ics.uci.edu/ml/datasets/Daphnet+Freezing+of+Gait
[13] http://www.ius.edu.ba/sites/default/files/articles/51-145-1-PB.pdf
[14] http://www.mathworks.com/help/stats/supervised-learning-machine-learning-workflow-and-algorithms.html
[15] http://www.mathworks.com/discovery/unsupervised-learning.html

Progress Highlights:

1. Problem identified properly with the main focus being around Gait and Speech data signals.
2. We have small and large datasets with labels as well as no labels.
3. Applied supervised and unsupervised learning methods on both types of dataset and obtained results.
4. Results look satisfactory and the comparison of various models is also shown in a tabular form.
5. Added noise to the data of various levels and tried to predict and obtain accuracy by comparing with true labels. A tabular form is obtained that shows the comparison between different methods and which one can be more/less useful during the presence of noise.
6. Computational time is also shown that can help us identify a proper method to be used when.
7. All the programming is done in MATLAB and using statistical packages.

Further:

1. Progress looks satisfactory as of now and is mostly finished.
2. I will try making all the code for R.
3. Will try getting more dataset to work with so that the models can be compared with various datasets.
4. Have to furnish the report with details as discussed with Professor via email.