Project Report on

Detection of freezing of gait in Parkinson's disease using machine learning

submitted in partial fulfilment of the course PG-Diploma

In

Big Data Analytics

From C-DAC ACTS (Bangalore)

Guided by:

Mr. Jitendra Kumar

Presented by:

Mr. Altaf Ahmad	**PRN: 230350125008**
Mr. Mayank Jain	**PRN: 230350125044**
Mr. Swapnil Devidas Chavan	**PRN: 230350125082**
Mr. Ninad Inamdar Bhirdikar	**PRN: 230350125051**

Candidate's Declaration

We hereby certify that the work being presented in the report titled: "***Detection of freezing of gait in Parkinson's disease using machine learning*** ", in partial fulfilment of the requirements for the award of PG Diploma Certificateand submitted to the department of PG-DBDA of the C-DAC ACTS Bangalore, is an authentic record of our work carried out during the period, 1 August 2023 to 31 August 2023 under the supervision of Mr. Jitendra Kumar, C-DAC Bangalore. The matter presented in the report has not been submitted by us for the award of any degree of this or any other Institute/University.

Name and Signature of Candidate:

Mr. Altaf Ahmad	PRN: 230350125008
Mr. Mayank Jain	PRN: 230350125044
Mr. Swapnil Devidas Chavan	PRN:230350125082
Mr. Ninad Inamdar Bhirdikar	PRN: 230350125051

Counter Signed by:

CERTIFICATE

To whomsoever it may concern
This is to certify that

Mr. Altaf Ahmad
Mr. Mayank Jain
Mr. Swapnil Devidas Chavan
Mr. Ninad Inamdar Bhirdikar

Have successfully completed their project on

Detection of freezing of gait in Parkinson's disease using machine learning

Under the guidance of

Mr. Jitendra Kumar

Mr. Jitendra Kumar **Mr. N S Nathan**

(Project Guide) **(Course Co-Ordinator)**

Acknowledgement

We take this opportunity to express our gratitude to all those people who have beendirectly and indirectly with us during the competition of this project. we pay thanks to **Mr. Jitendra Kumar** who has given guidance and a light to us during this major project.His versatile knowledge about "***Detection of freezing of gait in Parkinson's disease using machine learning*** "has eased us in the critical timesduring the span of this Final Project. We acknowledge here out debt to those who contributed significantly to one or more steps. We take full responsibility for any remainingsins of mission and commission.

Students Name

Altaf Ahmad

Mayank Jain

Swapnil Devidas Chavan

Ninad Inamdar Bhirdikar

CONTENTS

Abstract

Problem statement is detection of freezing of gait in Parkinson’s disease which is involuntary, repeated oscillatory movements of a body part occur due to cell death fall in NT-dopamine in substantia Nigra a dark color part in midbrain. This disease causes slowness of movement causes oscillation in body part like hand, legs, arms, foot and truck. The frequency range of tremor in patients of Parkinson's is 3 to 20Hz where as normal movements in the frequency range of 0-2Hz

CHAPTER 1
INTRODUCTION AND OVERVIEW

The goal of this competition is to detect freezing of gait (FOG), a debilitating symptom that afflicts many people with Parkinson's disease. we will develop a machine learning model trained on data collected from a wearable 3D lower back sensor. Our work will help researchers better understand when and why FOG episodes occur. This will improve the ability of medical professionals to optimally evaluate, monitor, and ultimately, prevent FOG events.

An estimated 7 to 10 million people around the world have Parkinson's disease, many of whom suffer from freezing of gait (FOG). During a FOG episode, a patient's feet are "glued" to the ground, preventing them from moving forward despite their attempts. FOG has a profound negative impact on health-related quality of life—people who suffer from FOG are often depressed, have an increased risk of falling, are likelier to be confined to wheelchair use, and have restricted independence.

While researchers have multiple theories to explain when, why, and in whom FOG occurs, there is still no clear understanding of its causes. The ability to objectively and accurately quantify FOG is one of the keys to advancing its understanding and treatment. Collection and analysis of FOG events, such as with your data science skills, could lead to potential treatments.

There are many methods of evaluating FOG, though most involve FOG-provoking protocols. People with FOG are filmed while performing certain tasks that are likely to increase its occurrence. Experts then review the video to score each frame, indicating when FOG occurred. While scoring in this manner is relatively reliable and sensitive, it is extremely time-consuming and requires specific expertise. Another method involves augmenting FOG-provoking testing with wearable devices. With more sensors, the detection of FOG becomes easier, however, compliance and usability may be reduced. Therefore, a combination of these two methods may be the best approach. When combined with machine learning methods, the accuracy of detecting FOG from a lower back accelerometer is relatively high.

However, the datasets used to train and test these algorithms have been relatively small and generalizability is limited to date. Furthermore, the emphasis has been on achieving high levels of accuracy, while precision, for example, has largely been ignored.

Our work is help advance the evaluation, understanding and treatment of FOG, improving the lives of the many people who suffer from this debilitating Parkinson's disease symptom.

Understanding the Objective and Project Flow

For Machine Learning

1. Acquired the dataset form THE MICHAEL J. FOX FOUNDATION website. The data comprises lower-back 3D accelerometer data from subjects exhibiting freezing of gait episodes with labels, we tried to analyzing the data using Machine learning algorithms

Machine learning algorithms used are: -

1. Decision Trees
2. XGBoost (Extreme Gradient Boosting)
3. LightGBM (Light Gradient Boosting Machine)
4. Cat Boost(Categorical Boosting)

CHAPTER 2
DATASET

This competition dataset comprises lower-back 3D accelerometer data from subjects exhibiting freezing of gait episodes, a disabling symptom that is common among people with Parkinson's disease. Freezing of gait (FOG) negatively impacts walking abilities and impinges locomotion and independence.

Our objective is to detect the start and stop of each freezing episode and the occurrence in these series of three types of freezing of gait events: Start Hesitation, Turn, and Walking

The data series include three datasets, collected under distinct circumstances
The tDCS FOG (tdcsfog) dataset, comprising data series collected in the lab, as subjects completes a FOG-provoking protocol.
The DeFOG (defog) dataset, comprising data series collected in the subject's home, as subject completed a FOG-provoking protocol.
Dataset, comprising one week of continuous 24/7 recordings from sixty-five subjects. Forty-five subjects exhibit FOG symptoms and also have series in the defog dataset, while the other twenty subjects do not exhibit FOG symptoms and do not have series elsewhere in the data. Trials from the tdcsfog and defog datasets were videotaped and annotated by expert reviewers documented the freezing of gait episodes. That is, the start, end and type of each episode were marked by the experts. there are detecting FOG episodes for the tdcsfog and defog series.

Series from the tdcsfog dataset are recorded at 128Hz (128 timesteps per second), while series from the defog and daily series are recorded at 100Hz (100 timesteps per second).

AccV, AccML, and AccAP Acceleration from a lower-back sensor on three axes: V - vertical, ML - mediolateral, AP - anteroposterior. Data is in units of m/s^2 for tdcsfog/ and g for defog.
Start Hesitation, Turn, Walking Indicator variables for the occurrence of each of the event types.
Type Whether Start Hesitation, Turn, or Walking.
Kinetic Whether the event was kinetic (1) and involved movement, or akinetic (0) and static.

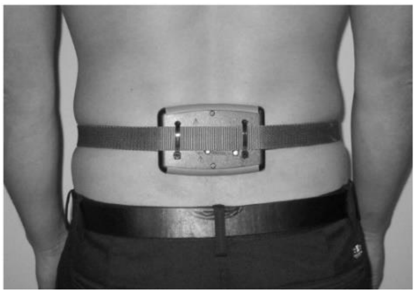


Fig (1.1)

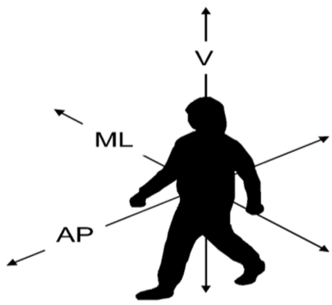


Fig (1.2)

The above figure's shown are lower back 3d sensor and the direction data is collected

CHAPTER 3
MATERIALS AND METHODS

Machin learning (ML) is the study of computer algorithms that can improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data,known as training data, in order to make predictions or decisions without being explicitly programmed to do so

- **Algorithms used for Prediction**
- **We train and test on the following Algorithms**
  - Decision Trees
  - XGBoost (Extreme Gradient Boosting)
  - LGBM (Light Gradient Boosting Machine)
  - Cat Boost(Categorical Boosting)
- **Steps followed for applying Machine learning Algorithms**

The technique used is Machine learning Algorithms which is applied on the dataset and their performance classifying the analyzed. Steps followed in applying ML on given data set,

- Create a new Environment.
- Download and import the needed packages and libraries.
- Perform data cleaning.
- Data Processing
- Extract the feature vector.
- Loading dataset for model training.
- Fitting the model.
- Test

<u>Decision Trees:</u>

Decision trees are a type of supervised machine learning algorithm used for both classification and regression tasks.

They represent a tree-like structure where each internal node represents a feature or attribute, each branch represents a decision rule, and each leaf node represents an outcome or class label. Decision trees are simple to understand and interpret. They can handle both numerical and categorical data.

XGBoost (Extreme Gradient Boosting):

XGBoost is an advanced gradient boosting algorithm designed for high performance and efficiency.

It uses an ensemble of weak learners (usually decision trees) and combines their predictions to improve accuracy.

XGBoost employs regularization techniques to prevent overfitting, and it supports parallel and distributed computing.

It has become a popular choice in machine learning competitions and real-world applications due to its robustness and effectiveness.

LGBM (Light Gradient Boosting Machine):

LightGBM is another gradient boosting framework that aims to be highly efficient and faster than traditional implementations.

It uses a histogram-based approach to split data into discrete bins, which speeds up the training process.

LightGBM employs a technique called "Gradient-based One-Side Sampling" to handle imbalanced datasets effectively.

Due to its speed and memory efficiency, LightGBM is suitable for large datasets.

Cat Boost (Categorical Boosting):

Cat Boost is a gradient boosting algorithm designed to handle categorical features without the need for extensive preprocessing.

It automatically encodes categorical features, reducing the risk of information loss.

Cat Boost also incorporates ordered boosting, which optimizes the learning rate for each boosting iteration.

It is robust to overfitting and can work well with small datasets

CHAPTER 4
MODEL BUILDING

Model building for the following algorithms: Decision Trees, XGBoost, LightGBM, and Cat Boost.

Decision Trees:

Decision Trees are a type of machine learning algorithm used for both classification and regression tasks. Here's how you build a decision tree model:

1. Data Preparation: Start by collecting and preparing your dataset. This involves cleaning the data, handling missing values, and encoding categorical variables if necessary.

2. Feature Selection: Choose the relevant features (input variables) that are most likely to impact the target variable.

3.Tree Construction: The decision tree is built in a recursive manner. At each step, the algorithm selects the best feature to split the data based on some criterion (e.g., Gini impurity for classification or mean squared error for regression).

4. Stopping Criteria: The tree-building process continues until a stopping criterion is met, such as a maximum depth for the tree, a minimum number of samples in each leaf, or if further splits do not significantly improve the model's performance.

5.Prediction: Once the tree is built, you can use it to make predictions on new data by traversing the tree based on the feature values.

6. Evaluation: Evaluate the model's performance using appropriate metrics for classification (e.g., accuracy, precision, recall) or regression (e.g., mean squared error, R-squared).

XGBoost (Extreme Gradient Boosting):

XGBoost is an ensemble learning algorithm based on gradient boosting. It's known for its high performance and flexibility. Here's how to build an XGBoost model:

1. Data Preparation: Similar to Decision Trees, start by preparing and cleaning your data.

2. Feature Selection: Select relevant features and encode categorical variables if needed.

3.Model Initialization: Initialize an XGBoost model, specifying hyperparameters like learning rate, maximum depth of trees, and the number of boosting rounds.

4.Gradient Boosting: XGBoost builds multiple decision trees sequentially. Each new tree corrects the errors made by the previous ones.

5.Regularization: XGBoost includes regularization techniques to prevent overfitting, such as L1 and L2 regularization on the tree weights.

6.Prediction: Make predictions by combining the predictions from all the individual trees.

7.Evaluation: Evaluate the model's performance using appropriate metrics. XGBoost often includes built-in evaluation during training to monitor the model's progress.

LightGBM (Light Gradient Boosting Machine):

LightGBM is another gradient boosting framework, designed to be efficient and scalable. Here's the model building process for LightGBM:

1.Data Preparation: As usual, prepare and clean your dataset.

2.Feature Selection: Select relevant features and handle categorical variables. LightGBM has a feature that can work directly with categorical data.

3.Model Configuration: Set hyperparameters like learning rate, tree depth, number of leaves, and bagging fraction.

4. Leaf-wise Tree Growth: Unlike depth-wise growth, LightGBM uses a leaf-wise growth strategy, which focuses on expanding the leaf with the highest gain in terms of the target.

5.Gradient-Based Learning: Similar to XGBoost, LightGBM uses gradient boosting to iteratively refine the model.

6. Data Parallelism: LightGBM can handle large datasets by utilizing data parallelism during training.

7. Prediction and Evaluation: Make predictions and evaluate the model's performance.

Cat Boost (Categorical Boosting):

Cat Boost is specifically designed to handle categorical features effectively. Here's how to build a Cat Boost model:

1.Data Preparation: Clean and preprocess your data, paying special attention to categorical variables.

2.Categorical Handling: Cat Boost can handle categorical variables directly without extensive preprocessing. It uses a technique called "ordered boosting" to work with categorical data.

3. Model Initialization: Initialize a Cat Boost model, specifying hyperparameters such as learning rate, tree depth, and number of iterations.

4.Ordered Boosting: CatBoost's ordered boosting strategy takes advantage of the order of categorical variables, resulting in improved accuracy.

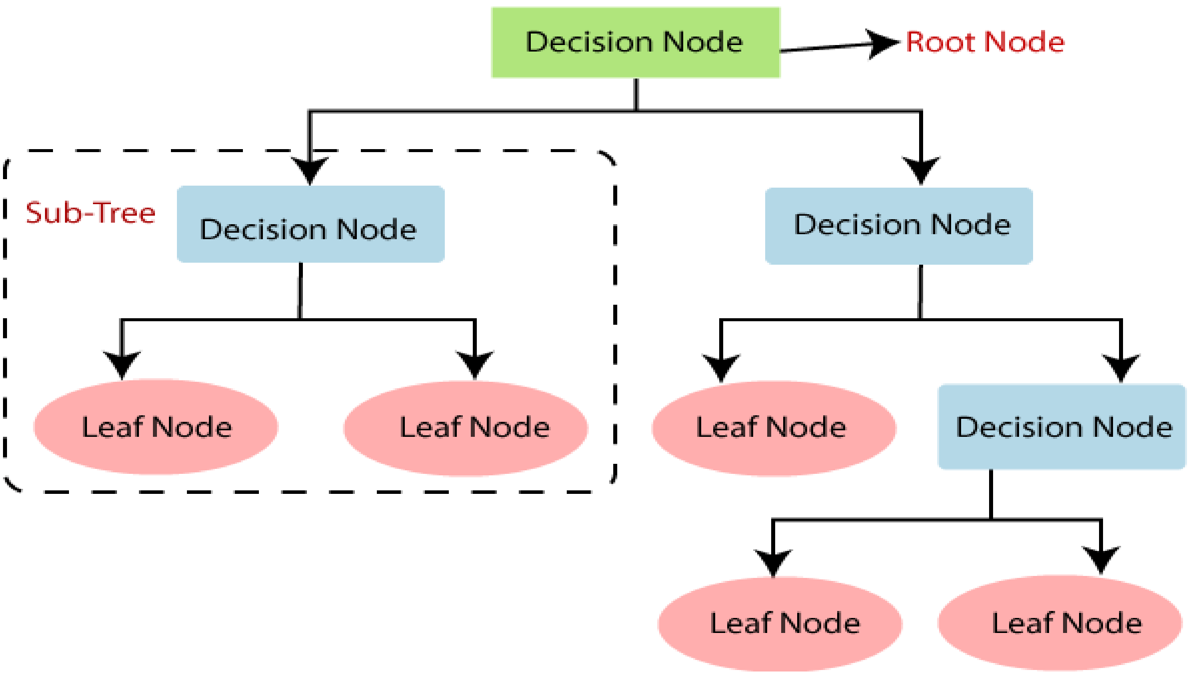
5. Regularization and Bayesian Optimization: Cat Boost includes built-in L2 regularization and employs Bayesian optimization to search for optimal hyperparameters.

6.Prediction and Evaluation: Make predictions and evaluate the model's performance using appropriate metrics.

CHAPTER 5
ARCHITECTURE

Decision Trees algorithm Work

In a decision tree, for predicting the class of the given dataset, the algorithm starts from the root node of the tree. This algorithm compares the values of root attribute with the record (real dataset) attribute and, based on the comparison, follows the branch and jumps to the next node. For the next node, the algorithm again compares the attribute value with the other sub-nodes and move further. It continues the process until it reaches the leaf node of the tree. The complete process can be better understood using the below algorithm:
Step-1: Begin the tree with the root node, says S, which contains the complete dataset.
Step-2: Find the best attribute in the dataset using Attribute Selection Measure (ASM).
Step-3: Divide the S into subsets that contains possible values for the best attributes.
Step-4: Generate the decision tree node, which contains the best attribute.
Step-5: Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.



Decision Tree Terminologies:

- Root Node: Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
- Leaf Node: Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
- Splitting: Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.
- Branch/Sub Tree: A tree formed by splitting the tree.
- Pruning: Pruning is the process of removing the unwanted branches from the tree.
- Parent/Child node: The root node of the tree is called the parent node, and other nodes are called the child nodes.

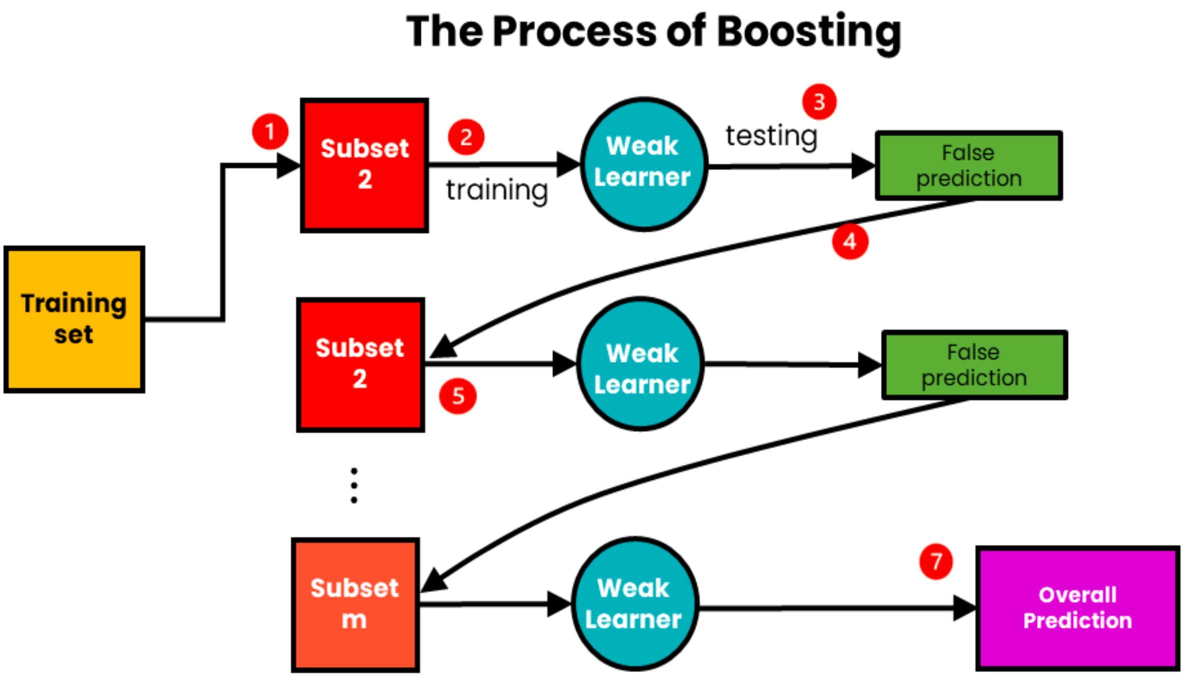
Boosting Algorithms:

- **Extreme Gradient Boosting Machine (XGBM)**
- **Light GBM**
- **CatBoost**

Boosting aims to produce a model with a lower bias than that of the individual models. Like in bagging, the weak learners are homogeneous.

Boosting involves sequentially training weak learners. Here, each subsequent learner improves the errors of previous learners in the sequence. A sample of data is first taken from the initial dataset. This sample is used to train the first model, and the model makes its prediction. The samples can either be correctly or incorrectly predicted. The samples that are wrongly predicted are reused for training the next model. In this way, subsequent models can improve on the errors of previous models.Unlike bagging, which aggregates prediction results at the end, boosting aggregates the results at each step. They are aggregated using weighted averaging.

Weighted averaging involves giving all models different weights depending on their predictive power. In other words, it gives more weight to the model with the highest predictive power. This is because the learner with the highest predictive power is considered



	CatBoost	LightGBM	XGBoost
Developer	Yandex	Microsoft	DMLC
Release Year	2017	2016	2014
Tree Symmetry	Symmetric	Asymmetric Leaf-wise tree growth	Asymmetric Level-wise tree growth
Splitting Method	Greedy method	Gradient-based One-Side Sampling (GOSS)	Pre-sorted and histogram-based algorithm
Type of Boosting	Ordered	-	-
Numerical Columns	Support	Support	Support
Categorical Columns	Support Perform one-hot encoding (default) Transforming categorical to numerical columns by border, bucket, binarized target mean value, counter methods available	Support, but must use numerical columns Can interpret ordinal category	Support, but must use numerical columns Cannot interpret ordinal category, users must convert to one-hot encoding, label encoding or mean encoding
Text Columns	Support Support Bag-of-Words, Naïve-Bayes or BM-25 to calculate numerical features from text data	Do not support	Do not support
Missing values	Handle missing value Interpret as NaN (default) Possible to interpret as error, or processed as minimum or maximum values	Handle missing value Interpret as NaN (default) or zero Assign missing values to side that reduces loss the most in each split	Handle missing value Interpret as NaN (tree booster) or zero (linear booster) Assign missing values to side that reduces loss the most in each split

Model

The model that was used for detection event of Freezing of Gait using versions machine learning algorithms. The data set consist of test taken in home and lab

We used 3 algorithms for each data set that is home and lab (defog & tdcsfog)

The algorithms used on defog(home) are :- Decision Trees, Cat Boost(Categorical Boosting) &XGBoost(Extreme Gradient Boosting)

The algorithms used on tdcsfog(lab) are :- Decision Trees, LightGBM (Light Gradient Boosting Machine) &XGBoost (Extreme Gradient Boosting)

After train and test we conclude the best model we chose is are Decision Trees & XGBoost on Turing(attribute)

Results and graphs

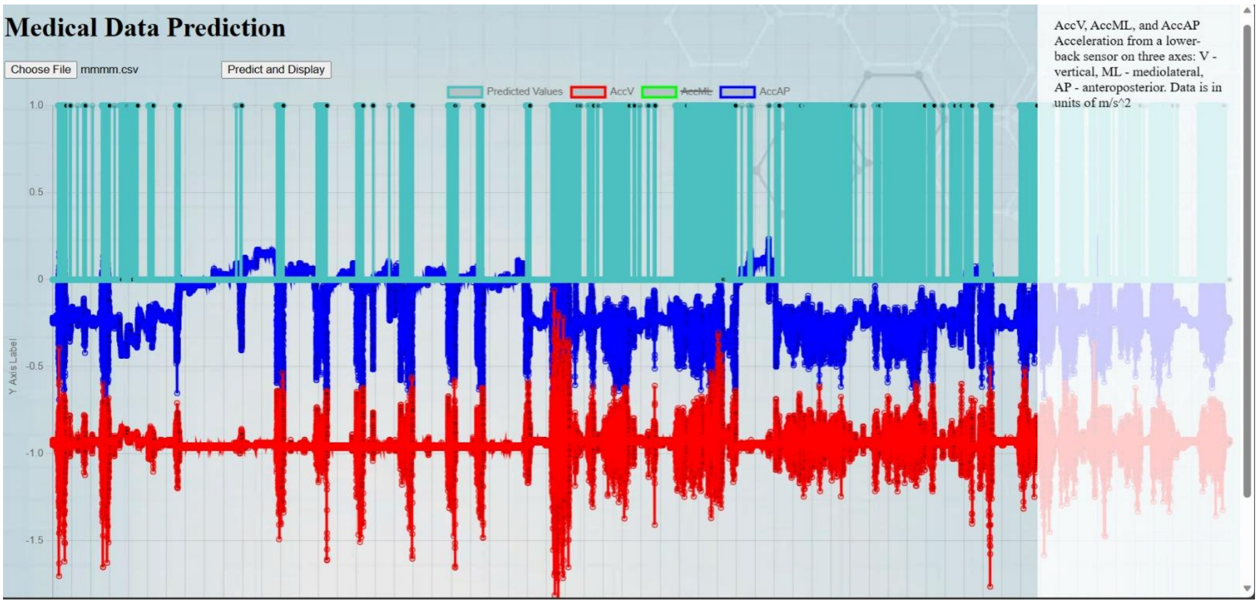


Fig of balanced data

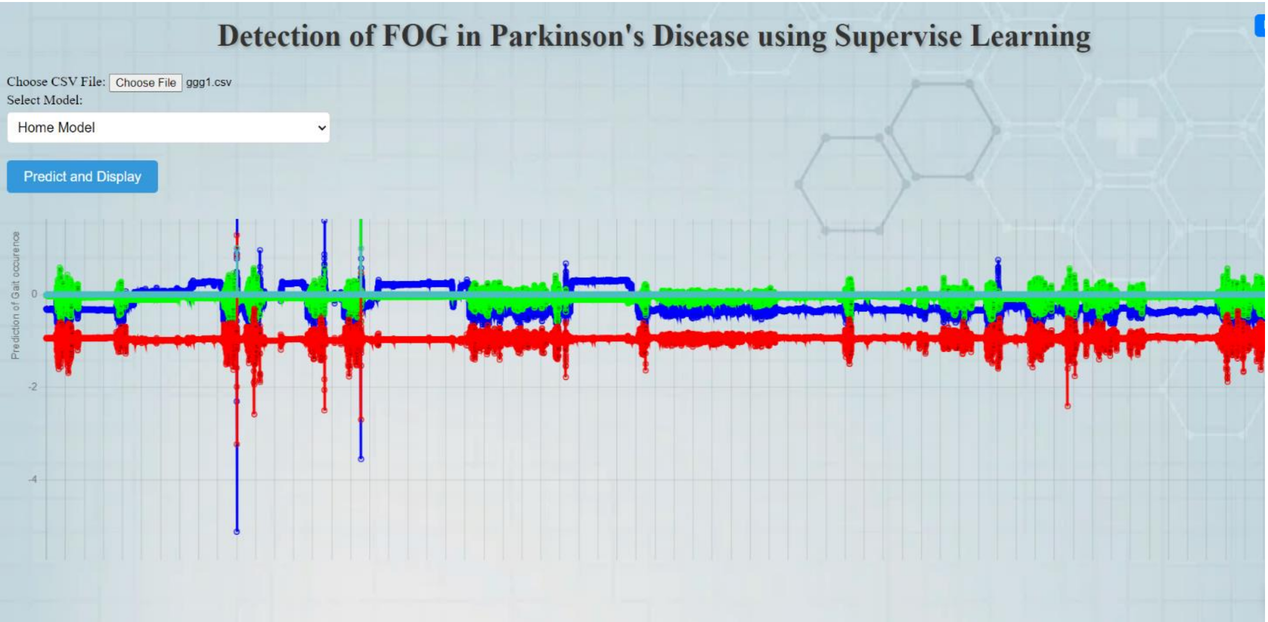


Fig of unbalance data

The change in accelaratoin in graph has Fog event

CHAPTER 6
APPLICATION

This application is used by doctors or Doman experts to analyze freezing of gate

It will make effort to understand Parkinson's disease

- ## System and Software Requirements

Hardware Specifications

Machine: Desktop/Laptop

Operating system: Windows 11 (or above) or Linux 18 LTS

(or above)Processors: Intel core i5 or AMD 5 series

(minimum)

Memory: 16GB RAM or

above Hard Disk(SSD):

250 GB or more

Video Card (optional): Intel Integrated Graphics (suggested – 4 GB graphicscard - NVIDIA)

Network: Ethernet / Wi-Fi with 25 Mbps Speed Connection (UL/DL)

Software Specifications

Language: Python 3.7 or above (stable

build) Platform: Anaconda Latest

stable build Notebooks:Jupyiter and

Google Colab

- Libraries: Pandas, NumPy, Matplotlib, Seaborn, Scikit-Learn, **XGBM, LightGBM, Cat Boost** glob.

CHAPTER 7
CONCLUSION

In conclusion, this project has successfully addressed the problem of Detection of FOG in Parkinson's disease by developing and implementing a machine learning model.

In conclusion, this project has successfully addressed the problem of Detection of FOG in Parkinson's disease by developing and implementing a machine learning model.

Our work will help researchers better understand when and why FOG episodes occur. This will improve the ability of medical professionals to optimally evaluate, monitor, and ultimately, prevent FOG events

CHAPTER 8
REFERENCES

Research paper referred:

Machine learning for large-scale wearable sensor data in Parkinson disease: concepts, promises, pitfalls and futures
https://movementdisorders.onlinelibrary.wiley.com/doi/abs/10.1002/mds.26693

Online detection of freezing of gait with smartphones and machine learning techniques
https://ieeexplore.ieee.org/abstract/document/6240371

Deep learning for freezing of gait detection in Parkinson's disease patients in their homes using a waist-worn inertial measurement unit
https://www.sciencedirect.com/science/article/abs/pii/S0950705117304859

Real-time detection of freezing of gait in Parkinson's disease using multi-head convolutional neural networks and a single inertial sensor
https://www.sciencedirect.com/science/article/abs/pii/S0933365722002111

Increasing Robustness in the Detection of Freezing of Gait in Parkinson's Disease
https://www.mdpi.com/2079-9292/8/2/119

Towards a wearable system for predicting freezing of gait in people affected by Parkinson's disease
https://ieeexplore.ieee.org/abstract/document/8894855