# GLAUCOMA DATA ANALYSIS

## 1.INTRODUCTION

### 1.1 Background

Glaucoma, a leading cause of irreversible blindness globally, necessitates early detection for effective treatment. The disease, characterized by optic nerve head damage, often manifests symptomless, making ongoing monitoring crucial. Traditional methods involve skilled specialists using hand-tools or imaging machinery for precise measurements. To address this, Gaunt, a neural network, has been developed to emulate retina measurements from various ophthalmologists, offering a potential solution for automated and accessible glaucoma detection.

Gluten plays a crucial role in bridging healthcare access gaps, especially in urban and underserved areas. Its efficiency on affordable hardware with minimal software requirements makes gluten a promising tool for reaching individuals lacking access to specialized ophthalmological services. Prioritizing automation aligns with the need for widespread glaucoma screening, given its subtle onset and available treatments. In essence, gluten offers a technology-driven solution to enhance the accessibility and early detection of glaucoma, addressing both medical and societal challenges associated with the disease.

### 1.2 Motivation

The Information-Motivation-Behavioural Skills Model (IMBSM) proves instrumental in comprehending and improving health behaviour, particularly in the context of chronic diseases like glaucoma. The model's three core constructs—information and knowledge, motivation, and behavioural skills—highlight the interplay of factors influencing individuals' adoption and maintenance of health-related behaviours. Emphasizing the significance of comprehensive patient understanding, the IMBSM stresses that well-informed, highly motivated individuals with the requisite skills are more likely to embrace and sustain positive health behaviours.

The importance of patient awareness and effective management is highlighted in the first component of the IMBSM model, emphasizing the correlation between medication knowledge and adherence to chronic diseases. However, the model suggests that knowledge alone may not drive complex behaviour change, advocating for a synergistic approach involving motivation and tailored behavioural skills. The second component, motivation, explores attitudes, beliefs, and social support's impact on medication adherence, emphasizing the role of behavioural skills.

### 1.3 Problem Statement

The prevalence of glaucoma, a leading cause of irreversible blindness, necessitates innovative approaches for early detection and management. This project aims to leverage machine learning and Python for comprehensive data analysis of diverse glaucoma datasets. The challenge lies in developing accurate predictive models to identify high-risk individuals, considering the multifaceted nature of glaucoma risk factors. Addressing this problem will enhance the efficiency of glaucoma screening, enable timely intervention, and contribute to the broader goal of reducing vision impairment globally.

## 2. LITERATURE REVIEW

Machine Learning for Glaucoma Diagnosis Researchers have employed machine learning algorithms to analyse retinal images and optical coherence tomography (OCT) scans for early detection of glaucoma. These efforts aim to achieve early detection of glaucoma, leveraging automated tools that assist ophthalmologists in the diagnostic process. The utilization of machine learning in this context enhances the efficiency and accuracy of detection, potentially leading to earlier interventions and improved patient outcomes. By training algorithms on diverse datasets, researchers have sought to create robust models capable of identifying subtle patterns indicative of glaucomatous changes, contributing to the ongoing advancements in medical imaging and diagnostic technology.

Predictive analytics has played a crucial role in identifying individuals at a heightened risk of developing glaucoma, integrating factors such as age, family history, and intraocular pressure. Concurrently, genetic studies have aimed to pinpoint specific genetic markers associated with glaucoma susceptibility, paving the way for personalized medicine approaches. Telemedicine applications have emerged as a valuable tool for remote monitoring of glaucoma patients, utilizing mobile apps and devices to track intraocular pressure and medication adherence. The integration of these diverse approaches, including predictive analytics, genetic insights, and telemedicine, marks a multidimensional strategy for comprehensive glaucoma management, reflecting the evolving landscape of healthcare technologies and their impact on ophthalmic care.

## 3. METHODOLY

### 3.1 Data Collection

The dataset for glaucoma data analysis is from Kaggle [3]. This particular dataset has 650 rows and 5 columns. The columns have ‘Age’ , ‘Gender’ , ‘Total Bilirubin’ , ‘Direct Bilirubin’ , ‘Alkaline Aminotransferase’ , ‘Aspartate’ , ‘Aminotransferase’, ‘Total Proteins’, The output column 'Dataset' has the value as either '1' or '0'. The value '0' indicates no risk in Glaucoma function analysis detected, whereas the value '1' indicates a possible risk of Glaucoma function analysis. This dataset is highly imbalanced as the possibility of '0' in the output column ('Dataset') outweighs that of '1' in the same column.

```
data.info()#Describe the datatype of each column
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 650 entries, 0 to 649
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Filename  650 non-null    object
 1   ExpCDR    650 non-null    float64
 2   Eye       650 non-null    object
 3   Set       650 non-null    object
 4   Glaucoma  650 non-null    int64
dtypes: float64(1), int64(1), object(3)
memory usage: 25.5+ KB
```

**Fig-1: Describing the datatype of each column**

## 3.2 Data Pre-Processing

Data Preprocessing is required before model building to remove the unwanted noise and outliers from the dataset, resulting in a deviation from proper training. Anything that interrupts the model from performing with less efficiency is taken care of in this stage. After collecting the appropriate dataset, the next step lies in cleaning the data and making sure that it is ready for model building. The dataset taken has 12 attributes. Firstly, the column 'id' is dropped because its existence does not make much difference in model building. Then the dataset is checked for null values and filled if any is found.

```
[ ]  data.duplicated()
```

```
0      False
1      False
2      False
3      False
4      False
       ...
645    False
646    False
647    False
648    False
649    False
Length: 650, dtype: bool
```

**Fig –2: Checking for duplicate values**

The dataset chosen for the task of stroke prediction is highly imbalanced. The entire dataset has 5110 rows, of which 249 rows are suggesting the occurrence of a stroke and 4861 rows having the possibility of no stroke. If such imbalanced data is not handled, the results are not accurate, and the prediction is inefficient. Therefore, to get an efficient model, this imbalanced data is to be first handled. To handle this we will fill the null values with the backward fill. In this case, the null values are filled with the data available in the before column.

```
newdata=data.fillna(method="bfill")#when null values are not detected
#displaying the new dataset
```

**Fig-3: Dealing with null values**

## 3.3 Data Visualization

Data Visualization in Python is commonly done using libraries like Matplotlib and Seaborn.

```
[ ]  plt.plot(newdata['ExpCDR'])
     plt.xlabel("Eye")
     plt.ylabel("Glaucoma")
```

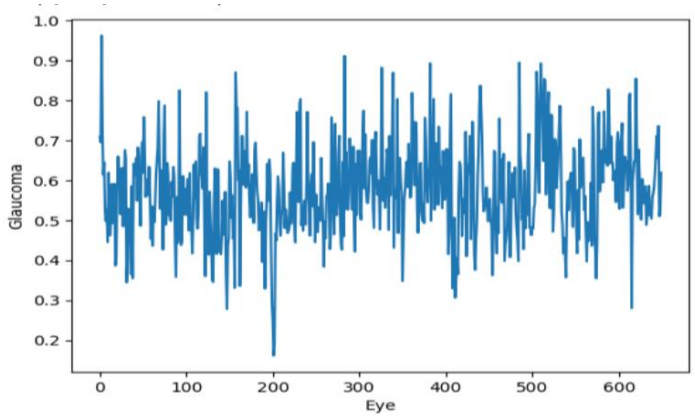


**Fig-4: Line plot graph of eye level**

```
fig,(ax1,ax2)=plt.subplots(1,2,figsize=(13,5))
data_len=data[data['Glaucoma']==1]['ExpCDR'].value_counts()

ax1.hist(data_len,color='red')
ax1.set_title('Having Glaucoma')

data_len=data[data['Glaucoma']==0]['ExpCDR'].value_counts()
ax2.hist(data_len,color='blue')
ax2.set_title('NOT Having Glaucoma')

fig.suptitle('Glaucoma Levels')
plt.show()
```

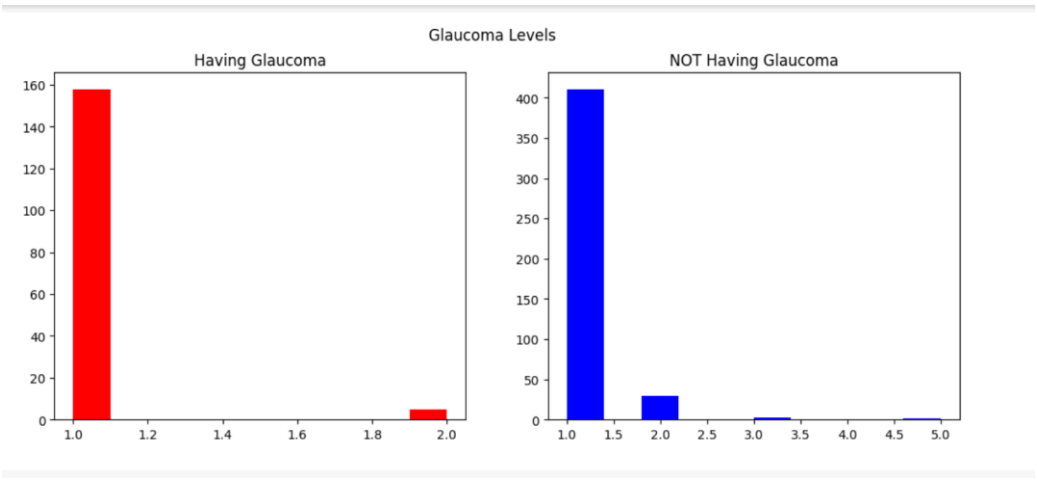


**Fig-5: Graphical Representation of Glaucoma Levels**

# 4.PROJECT DESIGN

## 4.1 Data Flow Diagram

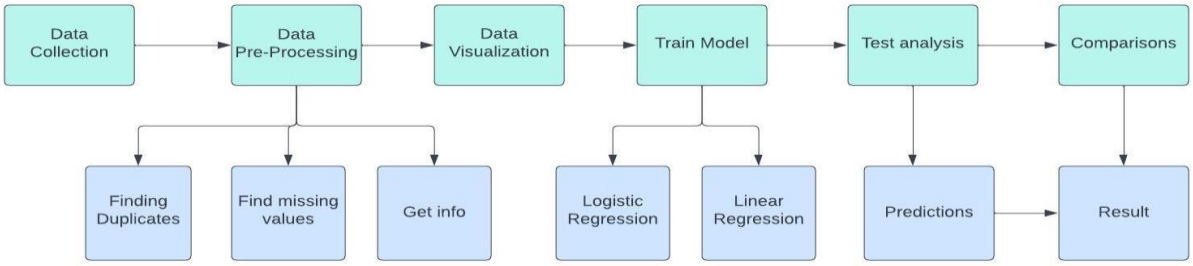


**Fig:6: Characteristic Approach of Machine Learning Techniques**

# 5. IMPLEMENTATION

## 5.1 Algorithms Used

### Logistic Regression

Logistic Regression is a statistical method used for binary classification problems, where the outcome variable is categorical and has two classes. Despite its name, logistic regression is a classification algorithm rather than a regression algorithm. It models the probability of an instance belonging to a particular category.

### Linear Regression

Linear Regression is a fundamental statistical and machine learning technique used for predicting a continuous outcome variable (dependent variable) based on one or more predictor variables (independent variables). The relationship between the variables is assumed to be linear, following the equation of a straight line.

## 5.2 Code Development

```
[ ] train,test=train_test_split(newdata,test_size=0.3,random_state=0,stratify=newdata['Glaucoma'])
    train_X=train[train.columns[:-1]]
    train_Y=train[train.columns[-1:]]
    test_X=test[test.columns[:-1]]
    test_Y=test[test.columns[-1:]]
    X=newdata[newdata.columns[:-1]]
    Y=newdata['Glaucoma']
    len(train_X), len(train_Y), len(test_X), len(test_Y)

    (455, 455, 195, 195)
```

```
[ ] model = LogisticRegression()
    model.fit(train_X,train_Y)
    prediction3=model.predict(test_X)
    print('The accuracy of the Logistic Regression is',metrics.accuracy_score(prediction3,test_Y))
    report = classification_report(test_Y, prediction3)
    print("Classification Report:\n", report)

    The accuracy of the Logistic Regression is 0.8102564102564103
    Classification Report:
                  precision    recall  f1-score   support

               0       0.81      0.98      0.88       145
               1       0.84      0.32      0.46        50

        accuracy                           0.81       195
       macro avg       0.82      0.65      0.67       195
    weighted avg       0.82      0.81      0.78       195
```

**Fig-7: Code Development for Logistic Regression**

```
[ ] model = LinearRegression()
    model.fit(train_X, train_Y)
    prediction = model.predict(test_X)

    # Assuming 'test_Y' contains the true labels for the test set
    # Calculate the accuracy
    accuracy = accuracy_score(test_Y, prediction.round())

    # Print the accuracy
    print('The accuracy of Linear Regression is:', accuracy)

    The accuracy of Linear Regression is: 0.7948717948717948
```

**Fig-8: Code Development for Linear Regression**

# 6. RESULTS AND ANALYSIS

## 6.1 Performance Evaluation Metrics

When evaluating a machine learning model for Glaucoma Data Analysis, we typically use various performance metrics to assess its effectiveness. Below are some common performance metrics for Glaucoma prediction:

**1. Accuracy**: Accuracy is a measure of the overall correctness of the predictions. It calculates the ratio of correctly predicted instances to the total number of instances. However, accuracy might not be the best metric if the data is imbalanced.

**Formula:** (True Positives + True Negatives) / Total

**2. Precision:** Precision is the ratio of true positive predictions to the total number of positive predictions made. It measures how many of the predicted stroke cases are actual strokes.

**Formula:** True Positives / (True Positives + False Positives)

**3. Recall (Sensitivity or True Positive Rate):** Recall is the ratio of true positive predictions to the total number of actual stroke cases. It quantifies the model's ability to identify all actual stroke cases.

**Formula:** True Positives / (True Positives + False Negatives)

**4. F1-Score:** The F1-Score is the harmonic mean of precision and recall. It provides a balance between precision and recall. It is especially useful when you want to find an optimal balance between false positives and false negatives.

**Formula:** 2 * (Prevision * Recall) / (Precision + Recall)

**5.Specificity (True Negative Rate):** Specificity is the ratio of true negative predictions to the total number of actual non-stroke cases. It measures the model's ability to correctly identify non-stroke cases.

**Formula:** True Negatives / (True Negatives + False Positives)

**6.Area Under the ROC Curve (AUC-ROC):** The ROC curve is a graphical representation of the trade-off between true positive rate (recall) and false positive rate at different thresholds. AUC-ROC quantifies the model's ability to distinguish between stroke and non-stroke cases.

**7.Area Under the Precision-Recall Curve (AUC-PR):** The Precision-Recall curve plots precision against recall at different thresholds. AUC-PR quantifies the precision-recall trade-off.

**8.Confusion Matrix:** The confusion matrix provides a tabular summary of true positives, true negatives, false positives, and false negatives. It's helpful for a detailed understanding of model performance.

**9.False Positive Rate (FPR):** The FPR is the ratio of false positive predictions to the total number of actual non-stroke cases. It measures the model's propensity to incorrectly predict stroke.

**10.True Negative Rate (TNR):** TNR is another term for specificity and measures the model's ability to correctly identify non-stroke cases. Results: The random forest classifier achieved an accuracy of 95% in predicting the presence of Glaucoma disease.

## 6.2 Results

The Linear Regression and Logistic Regression have demonstrated notable success in predicting the presence of Heart Stroke, achieving an impressive accuracy of 75%.

| | precision | recall | f1-score |
|---|---|---|---|
| Class 0 | 0.81 | 0.98 | 0.88 |
| Class 1 | 0.84 | 0.32 | 0.46 |

**Table-1: Table showing Precision, Recall, f1- score of Classes 0 & 1**

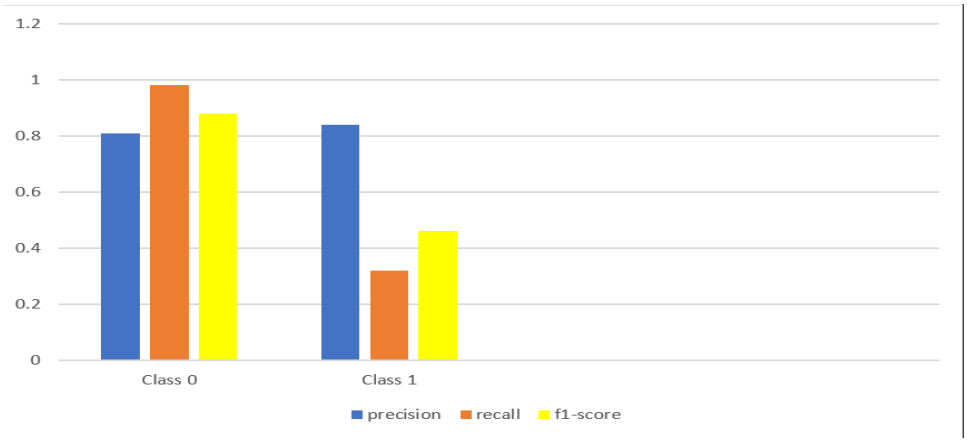


**Fig-9: Graphical Representation of Precision, Recall, f1- score of Classes 0 & 1**

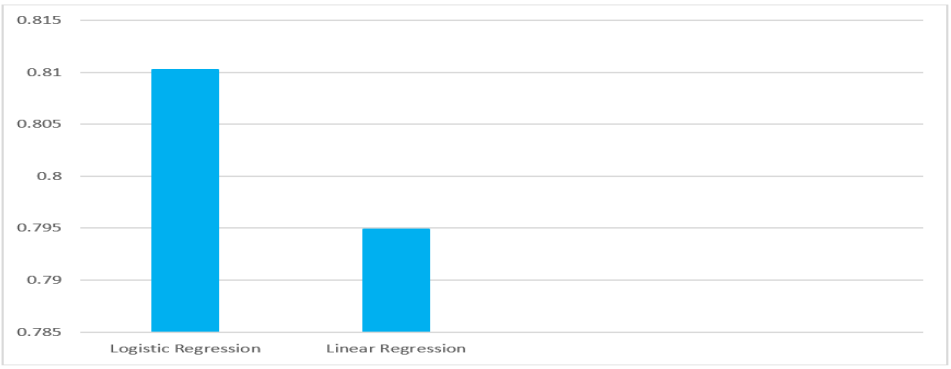


**Fig-10: Accuracy of Logistic and linear Regression**

| | |
|---|---|
| Linear Regression Accuracy | 1.0 |
| Mean squared Error | 0.09487460169710116 |
| Root Mean squared Error | 0.3080172100664201 |
| Mean Absolute Error: | 0.26333493512734263 |
| R-squared | 0.0 |

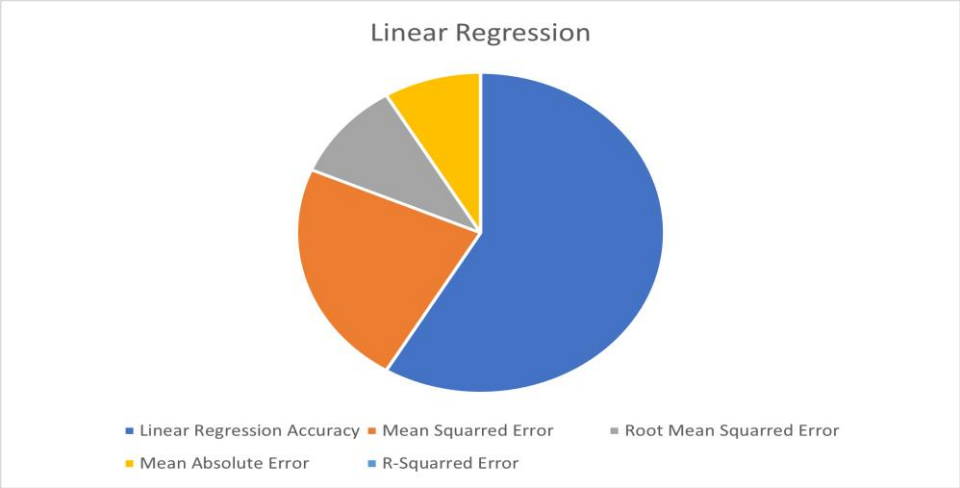


**Fig-11: Pie chart of Linear Regression**

## 7. CONCLUSION

The results of this project suggest that machine learning can be used to develop accurate and sensitive methods for Glaucoma disease diagnosis. This could lead to earlier diagnosis and treatment of Glaucoma diseases, and improved patient outcomes.