# Predictive Modeling for Type II Diabetes

Salma Chaudry and Shreya Sudini

Department of Data Science, University of Maryland Baltimore County

DATA 606: Capstone in Data Science

Dr. Tony Diana

May 1, 2024

**Abstract**

Type II Diabetes presents a critical global health epidemic, impacting millions of individuals worldwide and imposing substantial burdens on patients, healthcare providers, and medical systems. Given the increasing prevalence of this disease, this paper conducts a comprehensive study on leveraging predictive modeling techniques to enhance Type II Diabetes management. Four predictive models, Logistic Regression, CATBoost, Gradient Boosting, and Neural Networks address diabetes risks to improve medical outcomes. Model training and various data preprocessing techniques will be implemented along with thorough evaluations of performance metrics to determine each model's effectiveness in predicting diabetes risk. Results provide actionable insights for medical professionals and policymakers, streamlining decision-making processes and potentially lowering medical expenses for individuals. Graduate Students, Salma Chaudry and Shreya Sudini aim to address the Diabetes epidemic through innovative and predictive data science solutions.

## 1. Predictive Modeling for Type II Diabetes

Diabetes Mellitus is a chronic disease impacting how the body turns food into energy (CDC, 2023). Normally, during digestion, the hormone Insulin is released into the bloodstream to regulate blood sugar levels, facilitating the usage of sugars for energy (CDC, 2023). However, individuals with Diabetes cannot properly produce or utilize Insulin for blood sugar regulation. Without proper regulation, excess sugars reside in the bloodstream, potentially causing future health conditions like cardiovascular disease, kidney disease, and vision loss (Qin et al., 2022). While active health changes and treatments exist, early diagnosis is critical for effective intervention.

As a result of societal diet and lifestyle changes, the prevalence of Diabetes continues to rise. Among Diabetes types, Type II Diabetes remains the most common, impacting 90-95% of individuals with this disease (CDC, 2023). According to the International Diabetes Federation, Diabetes is expected to be the seventh greatest cause of death worldwide by 2030 (Qin et al., 2022). Moreover, by 2040, Diabetes is predicted to affect 642 million individuals, with 46.5% being undiagnosed (El Seddawy et al., 2022). Diabetes manifests without easily identifiable symptoms; therefore, many individuals develop Prediabetes or Diabetes due to unmonitored blood sugar levels, highlighting the importance of early detection and intervention (El Seddawy et al., 2022).

Diabetes is a serious epidemic affecting millions worldwide, yet many remain unaware of their susceptibility to the disease. This project aims to leverage predictive modeling to enhance Type II Diabetes management. By employing four predictive models- Logistic Regression, CATBoost, Gradient Boosting, and Neural Networks- this study strives to effectively identify Diabetes risks and improve management outcomes. We hypothesize that Neural Networks will

outperform other models due to their ability to detect non-linear relationships in data, making them adaptable for all kinds of predictive tasks (Tu, 1996). Before model training, various preprocessing methods including data cleaning, feature scaling, and class imbalance will be executed. Through intensive evaluation and comparison of model performance metrics, this study assesses each model's effectiveness in predicting diabetes risk. Findings provide actionable insights for healthcare professionals and policymakers, potentially reducing medical expenses for individuals while facilitating informed decision-making processes through knowledge discovery (El Seddawy et al., 2022).

## 2. Literature Review

This section examines various research and scholarly articles to provide a comprehensive overview of the knowledge surrounding diabetes management, predictive modeling, and machine learning techniques.

In response to the escalating prevalence of Diabetes and its detrimental health impacts, researchers Yifan Qin et al. (2022) investigated the effectiveness of machine learning models in predicting Diabetes. Obtaining 21 years of data from the National Health and Nutrition Examination Survey (NHANES) database, the study strived to improve early detection and intervention strategies for Diabetes. Evaluated algorithms included CATBoost, XGBoost, Random Forest (RF), Logistic Regression (LR), and Support Vector Machines (SVM). These five models assessed numerous performance metrics like accuracy, sensitivity, specificity, precision, F1 score, and receiver operating characteristic (ROC) curve. The dataset contained 18 diabetes-relevant factors including demographic, dietary, lab, and questionnaire response data (Qin et al., 2022). Qin and colleagues addressed class imbalance, as it impacted the accuracy of predictions. CATBoost, known for its superior accuracy compared to XGBoost, emerged as the

hypothesized best performing algorithm. Boosting algorithms like CATBoost and XGBoost notoriously resolve overfitting complications while enhancing computational efficiency. Logistic Regression, a simple and straightforward machine learning model, successfully predicts binary variables such as whether an individual has Diabetes or not (Qin et al., 2022). SVM excels in analyzing such classification challenges. Overall, hypotheses suggested that CATBoost would outperform other models, which was validated by the study's findings; CATBoost exhibited the highest accuracy of 82.1% in predicting diabetes (Qin et al., 2022). Findings considered CATBoost a reliable algorithm for diabetes prediction, highlighting the significance of leveraging machine learning models for improving healthcare outcomes in diabetes management. This article provides valuable insights into the usefulness of Logistic Regression and CATBoost algorithms for predicting diabetes.

Researchers Ahmed I. El Sedawwy et al. (2022) examined the Pima Indian Diabetes dataset from the UCI Machine Learning Repository. Given the increasing prominence of diabetes in today's society, researchers devised a Type II Diabetes prediction model using advanced predictive modeling techniques to assist in decision making and reduce medical costs for individuals. Researchers identified a binary classification problem to differentiate whether a patient suffers from diabetes or not. Throughout the study, the authors implemented various preprocessing steps for cleaning data, performing feature extraction, and instituting algorithms to predict the onset of diabetes. The dataset consisted of 768 females; 35% diabetic and 65% nondiabetic individuals (El Seddawy et al., 2022). Without considering the absence or presence of diabetes, additional data collected included demographic or clinical information. To eliminate class imbalance issues, oversampling and undersampling techniques like the Synthetic Minority Oversampling Technique (SMOTE) and Tomek-Links were integrated (El Seddawy et al., 2022).

Addressing class imbalance prevents bias, ensuring accurate predictions overall. To address outliers and missing values, researchers implemented preprocessing methodologies like imputation and interquartile range. Subsequently, various algorithms like Neural Networks, Decision Trees, Support Vector Machines, and Random Forests helped identify the most appropriate model for Type II Diabetes prediction (El Seddawy et al., 2022). Ultimately, the Neural Network emerged as the most reliable model, displaying an accuracy of 90.2% and successful performance metrics across oversampled and test datasets. Overall, this study emphasized challenges in hyperparameter optimization and model selection, affirming the necessity for proper methodologies to conserve time (El Seddawy et al., 2022). By leveraging diverse variables and handling class imbalance issues, this study provides more effective strategies for tackling the diabetes epidemic. Findings from this study support the implementation of Neural Networks for our research while highlighting the importance of handling class imbalance.

Similarly, Tasin et al. (2023) analyzed the Pima Indian diabetes dataset from the UCI Machine Learning Repository; but, additionally collected 203 samples from women employees at Rownak Textile Mills in Dhaka, Bangladesh. They managed class imbalance using SMOTE, Adaptive Synthetic Sampling (ADASYN), and hypertuning techniques, examining various machine learning algorithms for diabetes prediction including, XGBoost, Bagging, AdaBoost, Decision Trees, K-Nearest Neighbors, SVM, RF and LR. The study focused on the binary classification problem regarding the absence or presence of diabetes. Moreover, researchers assessed model performance using metrics like accuracy, precision, recall, F1 score, and AUC. Results depicted that the XGBoost classifier with the ADASYN approach outperformed with an accuracy of 81% while additionally possessing the lowest error in predicting insulin levels (Tasin

et al., 2023). The study concluded the proposed system, the XGBoost classifier with the ADASYN approach, displayed promising results for diabetes prediction, providing valuable insights for future research. Overall, this article offers valuable methodologies for the diabetes prediction project, including information on Logistic Regression and Neural Networks, along with evaluation metrics like accuracy, precision, F1 score, AUC, and recall.

Aishwarya Mujumdar and Dr. Vaidehi proposed a study that outperforms existing models regarding accuracy. By leveraging datasets containing features like BMI, age, insulin, glucose levels, and various external factors, the study employed modeling algorithms like Decision Trees, Logistic Regression, Gradient Boost Classifier, AdaBoost, SVM, KNN, and Bagging algorithms (Mujumdar & Vaidehi, n.d.). The authors implemented a pipeline to provide high accuracy. Metrics used to determine the best algorithm for diabetes prediction include the confusion matrix, accuracy, F1 score, precision, and recall. The study successfully concluded that the LR model, known for efficient binary classification tasks, outperformed other models with an accuracy score of 96%. The LR confusion matrix illustrated that the model predicted the majority of the outcomes. However, with the pipeline implementation, where a linear sequence of transformers is connected for modeling, produced different results. The AdaBoost Classifier and Gradient Boost Classifier, leveraging various weak learners, notably decision trees, achieved higher accuracies of 98.8% and 98.1% respectively. This study additionally compared the accuracies of the PIMA dataset and the current dataset, demonstrating that more information about a patient's health effectively predicts the possibility of an individual having diabetes. Overall, this study suggests how Logistic Regression and Boosting algorithms work in predicting diabetes, emphasizing the utility of the confusion matrix for determining the accuracy of predicted outcomes (Mujumdar & Vaidehi, n.d.).

Amani Yahyaoui et al. (2020) compared the performance of conventional machine learning models and deep learning methods to predict diabetes in its early stages for timely treatment. The authors utilized the PIMA Indian Diabetes dataset, composed of 768 instances with eight features for the evaluation. The features included pregnant count, plasma glucose concentration, diastolic blood pressure, skin thickness, serum insulin, body mass index, diabetes pedigree function, and age. SVM, RF, and Convolutional Neural Networks (CNN) consisted of the applied machine learning models. To eliminate the possibility of system bias, the experiment was performed ten times, with the final overall accuracy calculated as the average of those ten repetitions (Yahyaoui et al., 2020). Applied performance metrics included overall accuracy, kappa coefficient, precision, recall, and f-measure. Results indicated that Random Forest, which combines the decisions of different decision trees, outperformed other models, obtaining an overall accuracy of 83.67%. Moreover, the CNN model generated predictions with 76.81% accuracy. The proposed CNN model comprised three layers: convolutional, pooling, and fully connected layers. The convolutional layer processes input via kernels and activation functions. The pooling layer reduces dimensionality and increases robustness; classification occurs at the output layer. Due to lower accuracy within the CNN model, the authors recommend improving performance through feature extraction using automatic deep feature extraction techniques . Overall, this study highlights the use and functionality of CNN, which we plan to incorporate into the project, offering valuable insights into diabetes prediction (Yahyaoui et al., 2020).

Tejas N. Joshi and Professor Pramila M. Chawan (2018) developed a system that detects diabetes in its early stages with higher accuracy by showing the results of supervised learning methods, including Support Vector Machines, Logistic Regression, and Artificial Neural Networks (ANN). The study explains the two major types of diabetes: Type I (T1D) and Type II

Diabetes (T2D), where T2D appears to be the most common form with 90% of diabetic patients; nonetheless, T2D remains the focus of our diabetes research. The main causes for T2D include lifestyle factors, physical activity, dietary habits, and genetics; T1D originates from an auto immunological destruction of cells. The authors used a dataset with seven attributes including Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, and Age. The authors proposed a defined methodology with six different phases, Data Extraction, Preprocessing, SVM Based Learning, LR Based Processing, Post Processing, and Result Analysis. SVM provides discriminative power for classification problems with a large number of features; this algorithm consequently delivers a higher performance. LR performed best in classifying the dependent variable having binary classes with high efficiency and transparency. In the results, SVM and LR produced similar accuracies, 79%, and 78% respectively. This study emphasizes the importance of LR for binary classification problems like diabetes prediction.

Shahid Mohammad Ganie and Majid Bashir Malik (2022) proposed a new ensemble learning based framework for early prediction of Type II Diabetes. The study employed ensemble learning techniques like Bagging, Boosting, and Voting techniques. The dataset comprised individuals from different geographical regions and age groups, with a balanced male-to-female ratio, as recommended by medical experts like endocrinologists and diabetologists. The dataset consisted of 1,939 records and 10 independent lifestyle parameters, including age, sex, family history, smoking, drinking, thirst, urination, height, weight, and fatigue. Additionally, it included a dependent variable that determines whether the patient is diabetic or not. The authors meticulously applied imputation, resampling, discretization, and various data transformation techniques for data preprocessing. SMOTE handled data imbalance while Correlation Coefficient Analysis (CCA) determined the optimum set of lifestyle features

(Ganie & Malik, 2022). K-Fold Cross-Validation removed biases in the data to achieve realistic results. For feature engineering, information gain and correlation methods were employed. The study concluded that Bagged Decision Trees performed best with a testing accuracy score of 99.14%. Furthermore, Stochastic Gradient Boosting follows closely with a testing accuracy of 98.45%. The Voting classifiers, Logistic Regression, Decision Trees, and SVM came in last place with 89.51% accuracy. Performance metrics used included precision, recall, specificity, confusion matrix, and F1-score. Overall, this study provides detailed insights into various preprocessing techniques such as resampling and employing correlation analysis and information gain for feature engineering. These techniques will assist in handling large datasets like the one under examination for this project.

## 3. Data Collection and Preprocessing

To perform a proper data science analysis, Google Colab was employed on the T4 Graphic Processing Unit (GPU) to enhance collaboration efforts and provide efficient computation for larger datasets. The dataset derives from the Behavioral Risk Factor Surveillance System (BRFSS) which is a survey conducted by the CDC each year; this dataset was accessed through the data science platform, Kaggle (Teboul, 2022). The survey collects responses from over 400,000 Americans on health behaviors, chronic health conditions, and the use of preventative services (Teboul, 2022). The survey questions have been converted as features for this dataset. The dataset originally consisted of three CSV files from 2015; however, for this project, two of the CSV files containing data for two diabetes classes (0: no diabetes, 1: diabetes) were selected instead of the remaining CSV file with three classes (Teboul, 2022). This decision streamlines the analysis process and minimizes confusion or errors. Overall, we chose the two files with two diabetes classes, with one as the training set and the other as the testing

set. The already balanced dataset, named “Train.csv,” was selected for training to eliminate bias in this binary classification problem. The testing dataset, “Test.csv,” required cleaning to remove inconsistencies before machine learning implementation. Testing on an imbalanced dataset helps evaluate model performance in real-world settings where class distributions are generally skewed (Teboul, 2022). Additionally, incorporating separate files for training and testing improves performance and reduces the possibility of overfitting (Teboul, 2022). Both datasets contain 21 features, displayed in Table 1.

**Table 1. Dataset Features**

| Feature | Description | Data Type |
|---|---|---|
| Diabetes_binary | If the patient is diabetic (1) or not (0) | float64 |
| HighBP | Whether patient has high blood pressure or not | float64 |
| HighChol | Whether patient has high cholesterol or not | float64 |
| CholCheck | Whether the patient had a cholesterol check in 5 years or not | float64 |
| BMI | Body Mass Index value | float64 |
| Smoker | Has the patient smoked at least 100 cigarettes in their entire life? | float64 |
| Stroke | Has the patient had a stroke? | float64 |
| HeartDiseaseorAttacks | Has any coronary heart disease (CHD) or myocardial infarction or not? | float64 |
| PhysActivity | Had any physical activity in the past 30 days? | float64 |
| Fruits | Consume fruits one or more times per day? | float64 |
| Veggies | Consume veggies one or more times per day? | float64 |

| | | |
|---|---|---|
| HvyAlcoholConsump | Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week?) | float64 |
| AnyHealthCare | Any kind of healthcare coverage, including health insurance, prepaid plans such as HMO, etc | float64 |
| NoDocbcCost | Was there a time in the past 12 months when you needed to see a doctor not because of cost? | float64 |
| GenHlth | Rate general health on a scale of 1-5 | float64 |
| MentHlth | Scale of patient's mental health | float64 |
| PhysHlth | Scale of patient's physical health, including physical illness and injury in the past 30 days | float64 |
| DiffWalk | Difficulty in walking or climbing stairs? | float64 |
| Sex | Patient Gender | float64 |
| Age | 13-level age category | float64 |
| Education | Education level on a scale of 1-6 | float64 |
| Income | Income level on a scale of 1-8 | float64 |

Most of the columns in the dataset contain binary values ranging from zero to one, but some columns possess different values on different scales such as age, education, or income level (Teboul, 2022). Columns, "BMI," "MentlHlth," "PhysHlth," "Age," "Education," and "Income" have fluctuations, suggesting that multiple values are present in the dataset (Teboul, 2022).

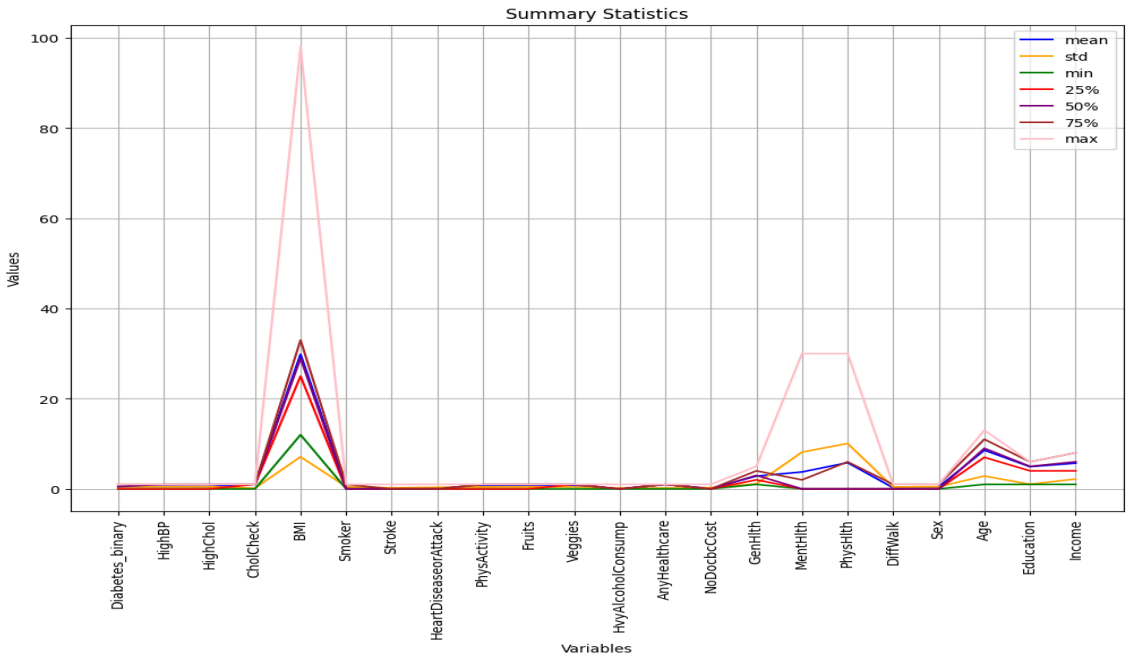



**Fig 1. Summary Statistics for Training Dataset**

Figure 1 shows the summary statistics of the columns in the training dataset. It shows what the statistics of different columns look like. The identification of columns with unique values can be concluded by looking at the graph. While some columns are useful, others increase the complexity of diabetes prediction and can compromise the performance of the model thus proving their irrelevance in the predictions. The solution for this is to remove some columns to make a path for better predictions with relevant predictions.

Along with the hypothesis of predictive modeling of diabetes management, we will be exploring socioeconomic factors such as age, education and income and BMI affecting diabetes, which will be helpful in drawing insights for policymakers and health management. For this reason, the socioeconomic indicators cannot be removed from the dataset. Therefore, from all the columns, “MentlHlth”, “PhysHlth”, “GenHlth”, “DiffWalk”, “AnyHealthcare”, and ““NoDocbcCost” contribute less to the diabetes prediction; as a result, we are dropping these columns. There is always a chance of the dataset having duplicates, which are also dropped from

the dataset. After dropping the duplicates and the irrelevant columns, the final transformed dataset is shown in Table 2 with a total of 16 features.

**Table 2. Revised Features**

| Feature | Description | Data Type |
|---|---|---|
| Diabetes_binary | If the patient is diabetic (1) or not (0) | float64 |
| HighBP | Whether patient has high blood pressure or not | float64 |
| HighChol | Whether patient has high cholesterol or not | float64 |
| CholCheck | Whether the patient had a cholesterol check in 5 years or not | float64 |
| BMI | Body Mass Index value | float64 |
| Smoker | Has the patient smoked at least 100 cigarettes in their entire life? | float64 |
| Stroke | Has the patient had a stroke? | float64 |
| HeartDiseaseorAttacks | Has any coronary heart disease(CHD) or myocardial infarction or not? | float64 |
| PhysActivity | Had any physical activity in the past 30 days? | float64 |
| Fruits | Consume fruits one or more times per day? | float64 |
| Veggies | Consume veggies one or more times per day? | float64 |
| HvyAlcoholConsump | Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week?) | float64 |
| Sex | Patient Gender | float64 |
| Age | 13-level age category | float64 |
| Education | Education level on a scale of 1-6 | float64 |

| | | |
|---|---|---|
| Income | Income level on a scale of 1-8 | float64 |

To find which of the columns have more correlation with diabetes, we employed correlation analysis. Correlation Analysis is mainly used to find the relationship between the features in the dataset. The heatmap for the correlation analysis is shown in Figure 2.

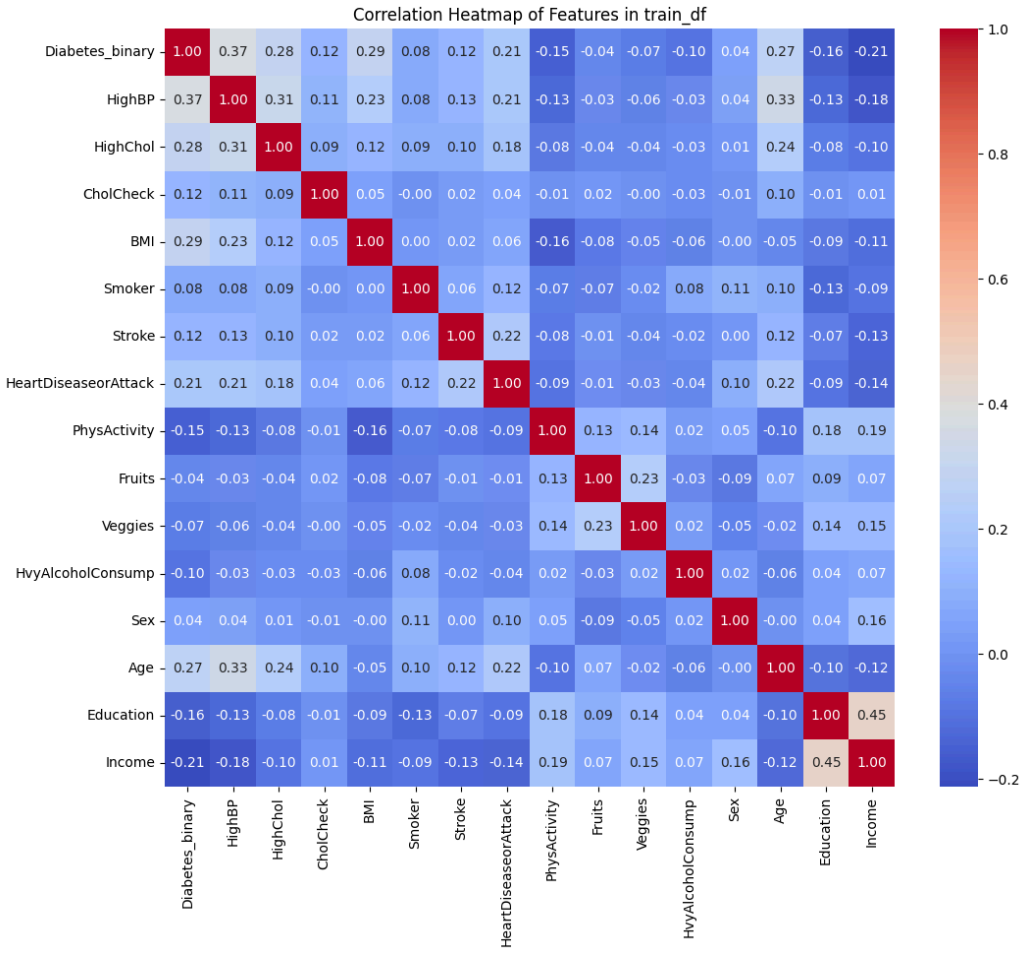
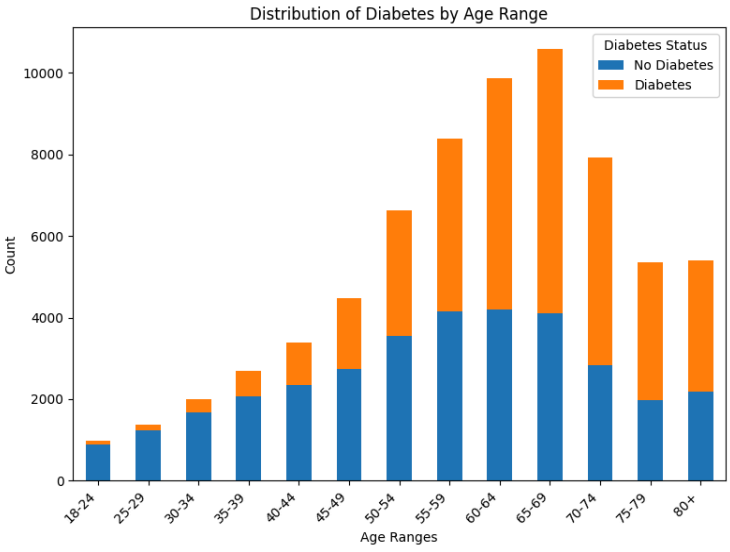



**Fig 2. Correlation Heatmap of Features in Training Dataset**

From Figure 2, it is apparent that the most affected, the people who are at more risk of getting diabetes are the ones with high blood pressure (HighBP). The other columns that have more a positive correlation are Body Mass Index (BMI), HighChol (high cholesterol), and Age.

The highest negative correlation is with Income, Education, and PhysActivity (Physical Activity), suggesting that less income, education, and physical activity can contribute to a person having diabetes.

As discussed earlier, the majority of the columns in the dataset have binary values, except columns like “Age”, “Education”, “Income”, and “BMI”. BMI possesses a wide range of unique values, which is clear from Figure 1. The other three columns have a considerably low range of unique values, but since we are looking at how the socioeconomic factors affect diabetes, here are some visualizations of age ranges, education groups, and income levels, who are more affected by diabetes.



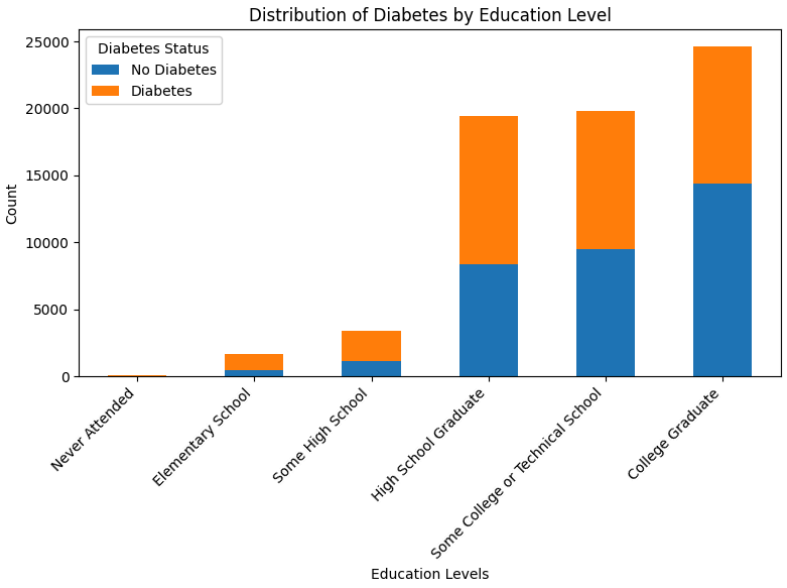




**Fig 3. Distribution of Diabetes by Age**

**Fig 4. Distribution of Diabetes by Education**

From Figure 3, the majority of people in the dataset are in the age range 65-69 and are the ones who are mostly diabetic out of the other age ranges. Identification of the age ranges can have significant importance in forming decisions for health management or generalizing diabetes tests for these age groups.

A similar plot is done for the education group which is shown in Figure 4. From Figure 4, we can conclude that the majority of individuals are college graduates and there is an equal distribution of high school graduates who attended some college or technical school. Based on this visual, high school graduates exhibit the highest prevalence of diabetes, making this demographic a high-risk group. Knowing this information will assist healthcare and insurance providers in targeted medical interventions for patients.

Regarding "Income," there are eight unique values present; but, the collected data categorizes income only into three groups. Consequently, we grouped the values into three distinctive groups: Less than $10,000, Less than $35,000, and More than $75,000. The insights gathered regarding the relationship between income and diabetes are depicted in Figure 5.

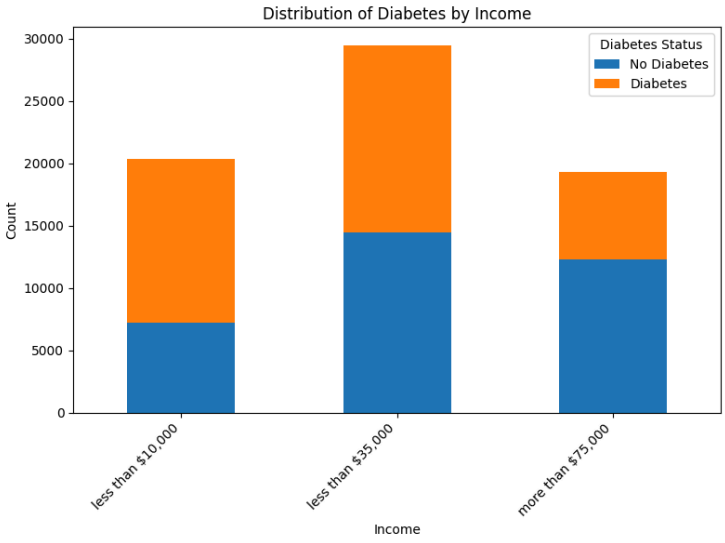
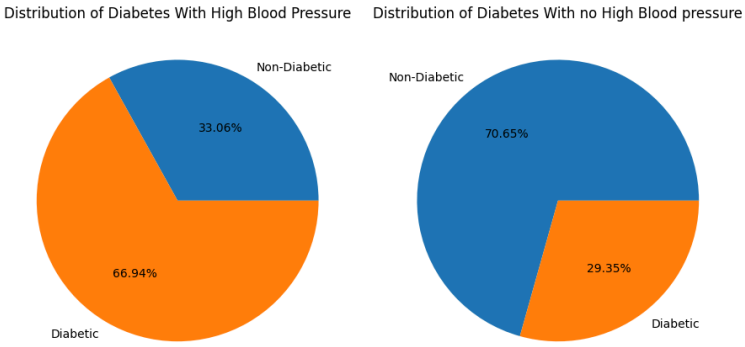



**Fig 5. Distribution of Diabetes by Income**

According to Figure 5, the majority of the people who participated in the survey make less than $35,000. But, those who have diabetes are in the income group who make less than 10,000.

Looking at the conclusions of the socioeconomic factors, it is justified from the correlation heatmap. Age shows a positive correlation with diabetes, suggesting that as age increases, the risk of diabetes rises. This explains why age ranges of 65-69 have a higher diabetic population. Similarly, there is a negative correlation between income and education, indicating that lower income or education levels are associated with a higher risk of diabetes.

After concluding that people with high blood pressure are likely to have more diabetes according to the correlation analysis shown in Figure 2, we further examine the distribution of diabetes concerning blood pressure (BP) through pie charts shown in Figure 6.



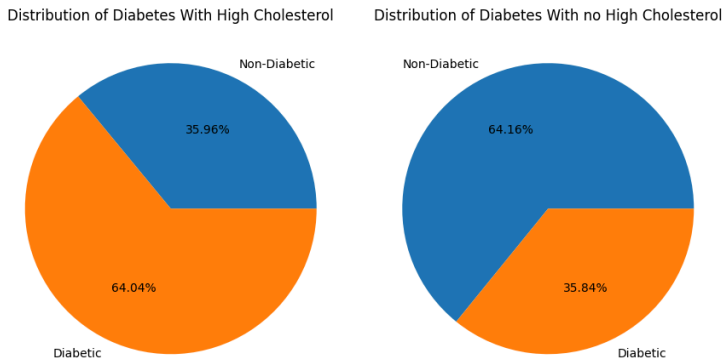




**Fig 6. Distribution of Diabetes by BP** **Fig 7. Distribution of Diabetes by Cholesterol**

As shown in Figure 6, approximately 67% of individuals with high blood pressure suffer from diabetes; moreover, 30% of those without high blood pressure suffer from diabetes.

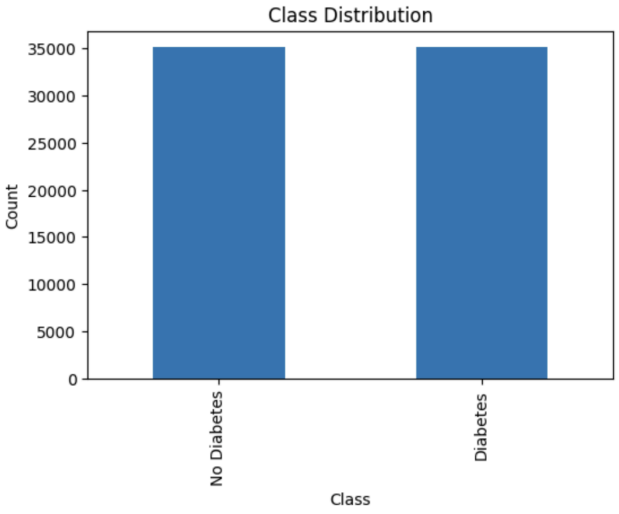
Another variable showing a high correlation with diabetes is Cholesterol, as shown in Figure 7. According to the figure, about 64% of individuals with high cholesterol are diabetic; furthermore, about 36% of those without high cholesterol have diabetes.

A similar transformation of dropping duplicates and the irrelevant on the testing dataset as well as the training and testing dataset needs to be the same for the model to perform well.

Then, we proceed to check for any null values in the dataset. In both the training and testing datasets, there are no null or NaN values, so we did not apply any imputations to the data.

**Handling Missing Values**

Initially, the training dataset was balanced, but with the removal of duplicates, it is now imbalanced. The training data must be balanced to ensure the model is trained well to provide better results with the testing dataset. For the balancing, we are employing SMOTE (Ganie & Malik, 2022). Since we are not keen on removing the values and adding duplicates to the dataset, we are using SMOTE, as it adds a synthetic value to the minority class based on the feature similarity with nearest neighbors. The balanced class distribution after applying SMOTE is shown in Figure 8. It is important to note that the testing dataset has a considerable imbalance in the diabetes classes. Figure 9 shows that people are not diabetic.



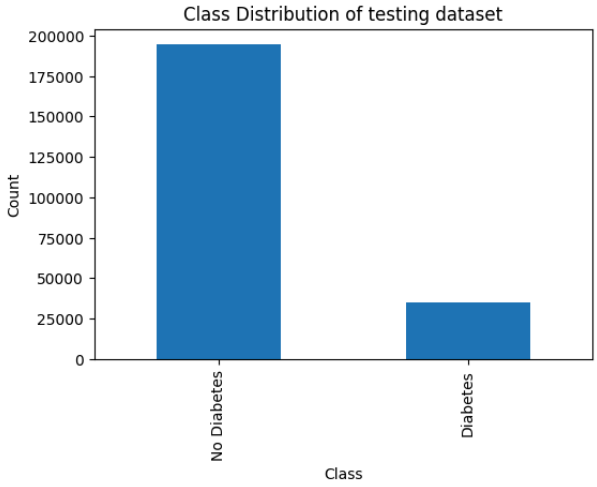




**Fig 8. Class Distribution after SMOTE**

**Fig 9. Distribution of Testing Dataset**

**Feature Engineering- Scaling**

There was a conversion of the Diabetes_binary column into a string for ease in visualization, but for the modeling, all the columns need to be numeric. For this conversion, we

mapped the values 0 and 1 for non-diabetic and diabetic respectively. This is done both on training and testing datasets. There is a categorical column “Sex” in the dataset, which is converted to numeric using Label Encoder. Label encoding is simple and easy to implement as it assigns a unique integer to each category. After all the columns are converted into numeric values, we scaled the dataset, since there are columns with many unique values. Scaling is important as it will converge the values into a similar range which will be useful for faster computation in the case of boosting algorithms, to decrease the dominance of columns with many values on the predictions, and ultimately increase the performance of the models. We used MinMaxScaler to scale the values down to 0 and 1 since most of the columns have binary values.

**Splitting the Training and Testing Data**

After the data is cleaned, balanced, and scaled down, we proceed to split the training and testing data into data and target variables. “Diabetes_binary” is the target column, which shows if the patient is diabetic or not. Training data helps train the model while testing data assesses its performance on unseen instances. Overall, this process helps evaluate the model’s accuracy and its ability to formulate new data, guaranteeing reliable predictions.

## 4. Methods

### A. Logistic Regression

Logistic Regression (LR) is a renowned supervised classification algorithm that commonly assists in predicting binary classification problems (Qin et al., 2022). In binary classification, a variable can only have two categories; in this project, the binary classification task is determining whether an individual possesses diabetes or not based on the provided features (Qin et al., 2022). Binary outcomes are measured by incorporating a logistic function. LR was selected for its simplicity, ease of implementation, and frequent use in predicting critical

binary variables like diabetes prediction, heart disease prediction, spam detection, and cancer detection (Qin et al., 2022). Nevertheless, its popularity and usage in the medical industry deemed this a reliable algorithm to incorporate. Included below are steps for integrating Logistic Regression.

***Implementation of Logistic Regression:***

**Step 1:** Import required libraries and performance metrics like scikit-learn, accuracy_score, classification_report, and confusion_matrix

**Step 2**: Initialize an instance of the LR model

**Step 3**: Train the model by fitting it on the training dataset

**Step 4:** Make predictions on the test dataset using the trained LR model

**Step 5:** Evaluate the performance of the LR model using metrics such as accuracy, precision, recall, f1 score, and confusion matrix. This provides insights into the model's predictive performance and its ability to properly predict instances.

**Step 6:** Calculate the training accuracy to evaluate how well the model performs on training set

**B. Gradient Boosting**

Gradient Boosting (GB) is an ensemble learning technique that combines a series of weak learners into a single strong learner, with each new model learning to minimize the loss from the previous model using gradient descent (GeeksforGeeks, 2023). The core idea of this model is for a weak learner to learn from the loss of the previous model, essentially "boosting" the performance with iterations. GB is more robust, as it updates the weights based on gradients, which are less sensitive to outliers (GeeksforGeeks, 2023). Using the GB machine for diabetes prediction is quite useful; diabetes prediction has complex relationships between various factors such as BMI, high blood pressure, high cholesterol, and the risk of being diabetic. This algorithm

learns from the previous model sequentially and aims to decrease the loss, capturing these complex relationships and providing accurate predictions (GeeksforGeeks, 2023). Medical data often contains outliers which can affect model performance. GB is robust and less sensitive to extreme values due to the use of gradients to update weights, hence proving to be useful for diabetes prediction. Additionally, this algorithm yields high predictive accuracy due to its iterative learning, ensemble learning, and use of regularization techniques to reduce overfitting and improve generalization ability. Displayed below are the steps for integrating the Gradient Boosting Classifier.

***Implementation of Gradient Boosting Classifier:***

**Step 1:** Import necessary libraries

**Step 2**: Initialize the Gradient Boosting Classifier with specified hyperparameters such as the number of decision trees in the ensemble, maximum depth of individual trees, and learning rate.

**Step 3**: Train the model by fitting it on the training dataset.

**Step 4:** Make predictions on the test dataset using the trained GB model

**Step 5:** Evaluate the performance of the GB model using metrics such as accuracy, precision, recall, f1 score, and confusion matrix. This provides insights into the model's predictive performance and its ability to properly predict instances.

**Step 6:** Calculate the training accuracy to evaluate how well the model performs on training set

In this paper, we took decision trees as the weak learners for the GB classifier. For the hyperparameters, the number of decision trees we took is (n_estimators) 100, the depth of each decision tree (max_depth) is 3 and the learning rate, the step size shrinkage to update the weights is taken as 0.1, we chose a small value to reduce overfitting. This model is fitted onto the scaled training dataset for training and the predictions are made on the scaled testing dataset. Further,

the performance metrics such as training accuracy, testing accuracy, confusion matrix, precision, and recall are calculated for evaluation.

**C. CATBoost**

CATBoost, or Categorical Boosting, is an open-source library designed for regression or classification with a large number of features (GeeksforGeeks, 2024). CATBoost is a variant of the Gradient Boosting algorithm that handles both categorical and numerical features without requiring feature encoding techniques to convert categorical features into numerical ones (GeeksforGeeks, 2024). Moreover, embedded algorithms within CATBoost automatically handle missing values to minimize overfitting and enhance overall prediction performance (GeeksforGeeks, 2024). Results are achieved without requiring parameter tuning and imputation, conveniently saving time and simplifying the data preparation process (GeeksforGeeks, 2024). Despite this algorithm demanding extensive memory consumption and training time, CATBoost's fast, robust, and scalable resources attract many developers due to its ease and time efficiency (GeeksforGeeks, 2024). In addition to its benefits, we considered CATBoost a reputable algorithm as it outperformed four unique models, LR, RF, SVM, and XGBoost in the diabetes prediction project conducted by Qin et al. (2022). The researchers determined that CATBoost provided a higher accuracy compared to other algorithms extensively utilized in the medical industry; moreover, the advantages of not requiring hyperparameter tuning and providing reduced chances of overfitting deemed this a good algorithm to implement into our diabetes predictive modeling project. Included below are steps for integrating CATBoost into our project.

***Implementation of CATBoost:***

**Step 1:** Import required libraries and performance metrics like CATBoost classifier, accuracy_score, classification_report, and confusion_matrix

**Step 2**: Initialize an instance of the CATBoost model with specified hyperparameters if needed

**Step 3**: Train the CATBoost model by fitting it on the training dataset

**Step 4:** Make predictions on the test dataset using the trained CATBoost model

**Step 5:** Evaluate the performance of the CATBoost model using metrics such as accuracy, precision, recall, f1 score, and confusion matrix. This provides insights into the model's predictive performance and its ability to properly predict instances.

**Step 6:** Calculate the training accuracy to evaluate how well the model performs on training set

**D. Neural Networks**

Neural Networks are models which have similar functionality as the brain which is becoming popular in predictive modeling (GeeksforGeeks, 2024). Neural Networks are made up of interconnected neurons that take input and learn from it for classifications to make predictions. A Neural Network is organized into layers with an input layer, one or more hidden layers, and an output layer, which are made up of multiple neurons joined by weighted connections (GeeksforGeeks, 2024). The input layer takes the raw data, where each neuron represents a feature in the dataset. The neurons present in the hidden layers simply take the input terms, add them up, and pass them through an activation function to produce an output (GeeksforGeeks, 2024). The model learns to make accurate predictions by adjusting its weights and biases using the loss values iteratively. The output layer makes the predictions based on the processed input.

We chose Neural Networks to be one of the models because they can learn the complex and non-linear relationships in the data, with features like Body Mass Index (BMI), age, and blood pressure in the data (Tu, 1996). They are also good at processing high dimensional data. The dataset we have has 15 features; using a neural network seemed plausible to extract predictions with good accuracy. Moreover, Neural Networks are useful for finding patterns in the data; they can learn from the individual patterns in the dataset, which will help generate personalized predictions for diabetes risk (Tu, 1996). Neural Networks are additionally scalable and can improve their performance continuously, which can be helpful as the diabetes data is ever growing and Neural Networks can learn from new data by retraining the model. The Neural Network employed for this project is a feedforward neural network, where the data moves in a sequence of input layer, hidden layer, and reaches the output layer without any feedback loops (GeeksforGeeks, 2024).

***Implementation of Neural Networks:***

**Step 1:** Imported necessary libraries for building the layers

**Step 2**: Define the hidden layers and output layer by defining the number of neurons and the activation function for each layer.

**Step 3**: Compiled the model and determined the optimizer, loss function, and evaluation metric.

**Step 4**: Trained the model on the scaled dataset by giving the epochs, batch size, and validation split.

**Step 5**: Evaluated the model on scaled test data for accuracy and loss.

**Step 6**: Made predictions on the scaled test data

**Step 7**: Converted the predictions into binary values of 0 and 1 based on the threshold

**Step 8**: Calculate the evaluation metrics like precision, recall, and confusion matrix for the performance of the model.

For the implementation, we used TensorFlow and Keras to build the Neural Network (UC Business Analytics, n.d.). This is a sequential model, with an input layer, two hidden layers, and an output layer. We used Dense layers, which are fully connected layers, where each neuron in one dense layer is connected to every neuron in the previous layer (UC Business Analytics, n.d.). The scaled trained data is passed to the input layer, which is given to the first hidden layer that is initialized with 64 neurons and ReLU (Rectified Linear Unit) which introduces a non-linearity in the layer, allowing the model to learn complex relationships in the data. The output of this is given to the second hidden layer which has 32 neurons with ReLU as its activation function. The output layer is also a dense layer with only one neuron, with Sigmoid as its activation function, which compresses the values to the (0,1) range (UC Business Analytics, n.d.). The model is then compiled, where the Adam optimizer is given for optimization because of its adaptive learning rate to update the weights of the network (Vishwakarma, 2024). Since we are employing binary classification, binary cross-entropy is chosen for the loss function (Saxena, 2023). Finally, "accuracy" is selected as an evaluation metric to measure its performance.

After the model is initialized and compiled, predictions are made on the scaled testing dataset, which are stored as probabilities. Furthermore, these probabilities are converted into binary values of (0,1) by specifying a threshold of 0.5. What we are doing here, classifying the outcomes with a probability less than 0.5 as class 0 (non-diabetic) and greater than probability 0.5 into class 1 (diabetic). Once we obtained this, we calculated the metrics precision, recall, and confusion matrix for evaluating the performance of the model.

## 5. Results

The metrics we used for determining the best models are accuracy, precision, recall, F1 score, and confusion matrix. The training and testing accuracies for the models are shown in the table below.

**Table 3. Summary of Model Accuracies**

| Model | Training Accuracy | Testing Accuracy |
|---|---|---|
| Logistic Regression | 72.88% | 69.63% |
| CATBoost | 73.44% | 69.08% |
| Gradient Boosting Machine | 73.78% | 68.99% |
| Neural Network | 62.48% | 72.79% |

From Table 3, we concluded that Neural Networks performed the best with a testing accuracy of 72.79%; the Gradient Boosting Machine has the lowest testing accuracy of all other models at 68.77%. However, the training accuracy of Neural Networks is less than all the models, suggesting that it does not overfit on the training data and generalizes well on the unseen test data compared to other models. The other three models performed with similar testing accuracies, with Logistic Regression having the highest accuracy of 69.63%. The Boosting algorithms have higher training accuracies with CATBoost at 73.44% and Gradient Boosting Machine at 73.78%, but show a decrease in testing accuracy suggesting some degree of overfitting and not being able to generalize well on the unseen data.

In this paper, we evaluated the models on other metrics such as precision, recall, and F1 score, which are discussed for each model below with the help of tables.

Precision measures the accuracy of positive predictions done by the model. It is calculated as the ratio of true positive predictions to the total number of positive predictions (Acharya, 2024). High precision means it makes only a few false positive predictions (Acharya, 2024).

Recall, also known as sensitivity, is the ability of the model to predict positive cases (Acharya, 2024). It is calculated as the ratio of true positives to the sum of true positives and false negatives (Acharya, 2024). In our project, recall indicates the proportion of correctly identified diabetic cases among all the positive ones. A high recall value indicates the model captures most of the positive cases in the dataset.

F1 score is the harmonic mean of both precision and recall. It provides a metric which combines both precision and recall into a single value, to give a balance between the two (Acharya, 2024). It is particularly useful for our case since there is a class imbalance between diabetic and non-diabetic cases in the test dataset. Using the F1 score, it would be easy to compare between the models (Acharya, 2024).

Support is the number of occurrences in each of the classes in the dataset (Acharya, 2024).

It is important to note that the testing dataset used has an imbalance between the diabetic and non-diabetic classes. There are more non-diabetic individuals in the testing dataset.

**Table 4. Logistic Regression Performance Metrics**

Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.94 | 0.69 | 0.79 | 194377 |
| 1 | 0.30 | 0.75 | 0.43 | 35097 |
| accuracy | | | 0.70 | 229474 |
| macro avg | 0.62 | 0.72 | 0.61 | 229474 |
| weighted avg | 0.84 | 0.70 | 0.74 | 229474 |

In Table 4, the precision for class 0 (non-diabetic) is 94%, and for class 1 (diabetic) is 30%. Among all the samples, the model predicted as not having diabetes, about 94% do not have diabetes, and among those predicted to have diabetes, about 30% have diabetes.

The recall for class 0 is approximately 69%, and for class 1, it's approximately 75%. This means that of all the actual samples that do not have diabetes, the model correctly identified about 69%, and of all the actual samples with diabetes, it correctly identified about 75%.

The F1 score for class 0 is approximately 79% indicating there is a balance between the model's ability to correctly identify non-diabetic patients and capture all non-diabetic cases. For class 1 is approximately 43%, which shows that the model's ability to correctly identify diabetic cases is higher compared to avoiding misclassifying non-diabetic cases.

**Table 5. CATBoost Performance Metrics**

Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.94 | 0.68 | 0.79 | 194377 |
| 1 | 0.30 | 0.77 | 0.43 | 35097 |
| accuracy | | | 0.69 | 229474 |
| macro avg | 0.62 | 0.72 | 0.61 | 229474 |
| weighted avg | 0.84 | 0.69 | 0.73 | 229474 |

From Table 6, the precision for class 0 (non-diabetic) is approximately 94%, and for class 1 (diabetic) is approximately 30%. Among the samples the model predicted as not having diabetes, about 94% do not have diabetes, and among those predicted to have diabetes, about 30% have diabetes.

The recall for class 0 is approximately 68%, and for class 1, it is approximately 77%. This means that of all the actual samples that do not have diabetes, the model correctly identified about 68%, and of all the actual samples with diabetes, it correctly identified about 77%.

The F1 score for class 0 is approximately 79% indicating there is a balance between the model's ability to correctly identify non-diabetic patients and capture all non-diabetic cases. For class 1 is approximately 43%, which shows that the model's ability to correctly identify diabetic cases is higher compared to avoiding misclassifying non-diabetic cases.

**Table 6. Gradient Boosting Classifier Performance Metrics**

Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.94 | 0.67 | 0.79 | 194377 |
| 1 | 0.30 | 0.77 | 0.43 | 35097 |
| accuracy | | | 0.69 | 229474 |
| macro avg | 0.62 | 0.72 | 0.61 | 229474 |
| weighted avg | 0.84 | 0.69 | 0.73 | 229474 |

From Table 6, the precision for class 0 (non-diabetic) is approximately 94% and for class 1 (diabetic) is approximately 30%. Among the samples the model predicted as not having diabetes, about 94% do not have diabetes, and among those predicted to have diabetes, about 30% have diabetes.

The recall for class 0 is approximately 67%, and for class 1, it's approximately 77%. This means that of all the actual samples that do not have diabetes, the model correctly identified about 67%, and of all the actual samples with diabetes, it correctly identified about 77%.

The F1 score for class 0 is approximately 79%, indicating a balance between correctly identifying non-diabetic cases and capturing all non-diabetic cases. For class 1 is approximately 43%, which shows that the model's ability to correctly identify diabetic cases is higher compared to avoiding misclassifying non-diabetic cases, which is less compared to class 0.

**Table 7. Neural Network Performance Metrics**

Classification report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.93 | 0.73 | 0.82 | 194377 |
| 1 | 0.32 | 0.71 | 0.44 | 35097 |
| accuracy | | | 0.73 | 229474 |
| macro avg | 0.63 | 0.72 | 0.63 | 229474 |
| weighted avg | 0.84 | 0.73 | 0.76 | 229474 |

From Table 7, the precision for class 0 (non-diabetic) is approximately 93%, and for class 1 (diabetic) is approximately 32%. Among the samples the model predicted as not having diabetes, about 93% do not have diabetes, and among those predicted to have diabetes, about 32% have diabetes.

The recall for class 0 is approximately 73%, and for class 1, it is approximately 71%. This means that of all the actual samples that don't have diabetes, the model correctly identified about 73%, and of all the actual samples with diabetes, it correctly identified about 71%.

The F1 score for class 0 is approximately 82%, indicating a balance between correctly identifying non-diabetic cases and capturing all non-diabetic cases. For class 1 is approximately

44%, which shows that the model's ability to correctly identify diabetic cases is higher compared to avoiding misclassifying non-diabetic cases, which is less compared to class 0.

**Overall Metric Analysis**

The precision, recall, and F1 scores display similarity across all the models for both classes, demonstrating consistent performance in predicting both class outcomes. Notably, LR, CATBoost, and GB algorithms demonstrate similar precision and F1 scores. However, Neural Networks show a slightly better performance regarding recall for Class 1, individuals with diabetes. This implies that Neural Networks successfully identifies diabetic individuals compared to other models, suggesting a possible advantage in healthcare related applications.

**Confusion Matrix for all Models**

Confusion Matrix is a matrix that displays the number of correct and incorrect instances predicted by the model (T, 2021). It is a matrix that displays True Positives, accurate predictions of positive values, True Negatives, accurate predictions of negative values, False Positives, inaccurate predictions of positive values, False Negatives, and inaccurate predictions of negative values (T, 2021). These matrices provide a summary of the performance of the model.

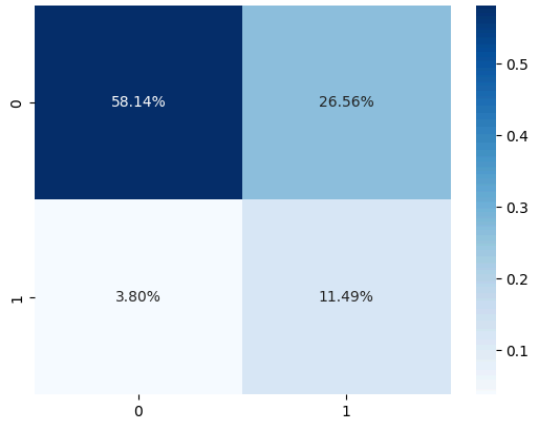



**Fig 10. LR Confusion Matrix**

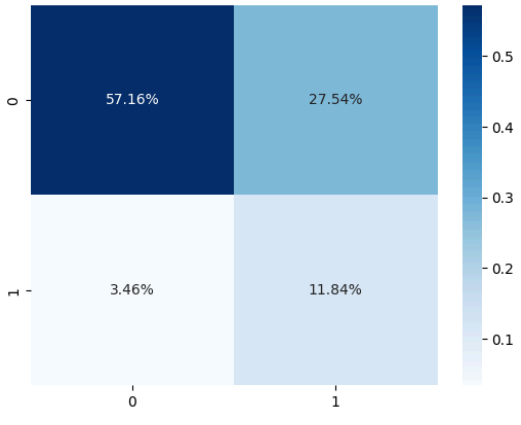
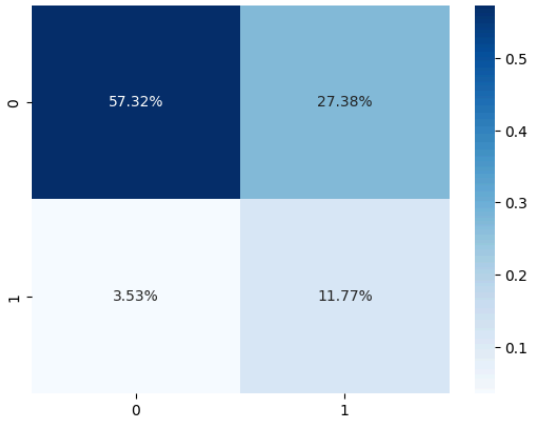



**Fig 11. GB Confusion Matrix**



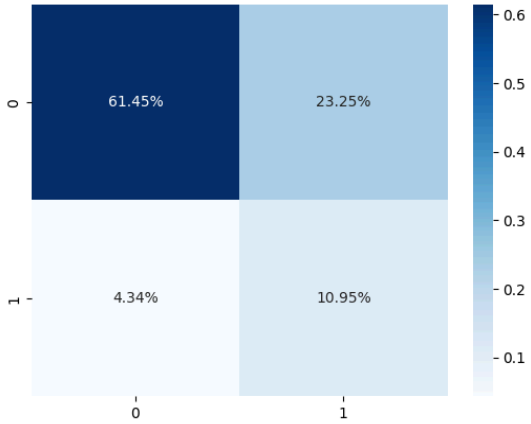




**Fig 12. CATBoost Confusion Matrix** **Fig 13. Neural Networks Confusion Matrix**

From Figure 10, we can see that about 58.14% of the LR instances are correctly classified as non-diabetic, and approximately 11.49% of LR instances are classified as diabetic. Approximately 26.56% are incorrectly classified as non-diabetic while 3.80% of the instances are incorrectly classified as diabetic.

In Figure 11, 57.16% of the GB instances are correctly classified as non-diabetic while approximately 11.84% of GB instances are classified as diabetic. Approximately 27.54% of GB instances are incorrectly classified as non-diabetic while 3.46% of the instances are incorrectly classified as diabetic.

From Figure 12, we can see that about 57.32% of the CATBoost instances are correctly classified as non-diabetic, and approximately 11.77% of CATBoost instances are classified as diabetic. Approximately 27.38% are incorrectly classified as non-diabetic while 3.53% of the instances are incorrectly classified as diabetic.

In Figure 13, we can see that about 61.45% of the Neural Network instances are correctly classified as non-diabetic, and approximately 10.95% of Neural Network instances are

classified as diabetic. Approximately 23.25% are incorrectly classified as non-diabetic while 4.34% of the instances are incorrectly classified as diabetic.

From the confusion matrices, we saw how different models predicted the values correctly and incorrectly. All the models showed a similar pattern in terms of value predictions. The Neural Networks predicted the highest number of true positives out of all models, indicating that it is effective in predicting people who are diabetic. However, it also has a high number of false positives which may lead to undetected cases of diabetes. All of the models predicting a high number of true positives may result from the imbalance in the testing data.

## 6. Discussion

Seeing the results, the skewness in the data is an important factor to consider while thinking about using this methodology for other disease predictions. This methodology is useful for diabetes predictions with the features set similar to this. It cannot be said with confidence, how this will perform on different datasets for  different disease predictions.

The models we employed are chosen according to the features and size of the dataset, other algorithms can be employed like Random Forest, which is known to perform well on high dimensional data such as diabetes data and is robust to overfitting (Joshi & Chawan, 2018). Support Vector Machines can also be used as one of the potential algorithms as it is known to be deterministic in binary classifications (Joshi & Chawan, 2018). Apart from the conventional machine learning models, there can be other deep learning models such as Conventional Neural Networks (CNN) and Recurrent Neural Networks that can be employed to learn complex data such as this one, because we have implemented simple feedforward networks.

From the different methodologies that we have seen in the literature review section, some references had a CNN showing less accuracy than RF and SVM (Yahyaoui et al., 2020). But, in

our project, employing a similar neural network, resulted in it performing best out of all the other algorithms. It also proves the study that showed that Neural Networks are one of the best performing models (El Seddawy et al., 2022). Another one of the papers concluded that boosting algorithms perform the best out of bagging and voting classifiers (Ganie & Malik, 2022). But, according to our methodology, boosting algorithms produced the lowest accuracy out of all of the models employed. Based on our findings, CATBoost does not perform that well, contradicting one of the references that had CATBoost giving out the best accuracy (Qin et al., 2022). However, after the Neural Networks, Logistic Regression exhibits better performance at around 70%, agreeing with the references that prove that Logistic Regression predicts a diabetic patient with around 70% accuracy (Joshi & Chawan, 2018).

Regarding future changes for diabetes, a recent study was published on how the early detection of diabetes is time consuming and difficult. A team of international researchers headed by an Associate Professor at the University of Bochum in Germany developed mathematical calculations, which, using only two values taken from blood samples, provided a reliable and inexpensive diabetes diagnosis at early stages ("Simple and Reliable Early Prediction of Diabetes," 2024). The study had already tested on different samples from people of different countries, which yielded good results. Seeing the results and the easy implementation compared to the rigorous steps of data processing, cleaning, and model training, this study provides a foundation for accessible, economical methods for early diabetes prediction.

## 7. Conclusion

This project leveraged four predictive modeling algorithms to identify risks and enhance Type II Diabetes management through predictive data science solutions. Successful and thorough implementation of data preprocessing methods, visualizations, machine learning models, and

performance metrics provided actionable insights for healthcare organizations, researchers, providers, and policymakers. Out of all algorithms, our hypothesis of Neural Networks outperforming all algorithms was validated, encouraging healthcare researchers and organizations to prioritize further investigation of Neural Network models.

Although not discussed in this paper, we originally intended to incorporate the Lazy Predict model, a Python library that seamlessly expedites the model selection and evaluation process by providing a list of the performance of all machine learning algorithms on a given dataset. Ideally, this model would have saved time, allowing us to focus on the best four performing models instead of choosing algorithms solely based on research from literature reviews. However, memory posed a significant challenge in implementing this model. Our interface, Google Colab, demanded additional memory at increased costs; moreover, we struggled to find a way to have the model fully generate results, as a blank result appeared despite referencing reliable websites and Python notebooks from our professor. Ultimately, we shifted our focus to implementing two successful predictive boosting algorithms instead of Lazy Predict. If the project deadline had been longer, we would have had more time to troubleshoot Lazy Predict; but, with the need for the paper and other steps to be completed, we had to alter plans. We highly recommend that researchers and other healthcare organizations investigate and apply Lazy Predict in their assessments to accelerate the model selection process.

While we aimed for our model accuracies to be higher, they still provide insights for future investigations. However, it is important to recognize that the testing dataset has an imbalance between the diabetic and non-diabetic classes; there are more non-diabetic individuals in the testing dataset which ultimately skews prediction outcomes.

Nonetheless, our project still informs healthcare organizations, providers, researchers, and policymakers of suggested algorithms to implement or preprocessing techniques to consider in their investigations. Additionally, insights from our visualizations assist in insurance or policy development and help healthcare providers prioritize resources and interventions for high-risk individuals, optimizing healthcare delivery and outcomes. Overall, our project facilitates informed decision-making processes through knowledge discovery and assists the medical community in devising proper data science methodologies for successful diabetes research projects.

Besides encouraging Neural Network implementation, other promising algorithms include Random Forest or Support Vector Machines. While this project does not examine these models, numerous research articles support these algorithms in detecting diseases like diabetes (Qin et al., 2022). However, once again, we highly suggest adding the Lazy Predict model to determine the best performance and assess from there. Conducted by graduate students Salma Chaudry and Shreya Sudini, we believe this project successfully addressed the Diabetes epidemic through innovative and predictive data science solutions.

## 8. References

Acharya, N. (2024, February 15). Understanding Precision, Recall, F1-score, and Support in Machine Learning Evaluation. *Medium*. https://medium.com/@nirajan.acharya666/understanding-precision-recall-f1-score-and-support-in-machine-learning-evaluation-7ec935e8512e#:~:text=It's%20like%20a%20balance%20between,of%20 instances%20in%20 each%20 class.

CDC. (2023, September 5). *What is Diabetes?* Centers for Disease Control and Prevention. Retrieved March 21, 2024, from https://www.cdc.gov/diabetes/basics/diabetes.html

El Seddawy, A. I., Karim, F. K., Hussein, A. M., & Khafaga, D. S. (2022). *Predictive Analysis of Diabetes-Risk with Class Imbalance*. National Library of Medicine. Retrieved March 21, 2024, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9578843/

Ganie, S. M., & Malik, M. B. (2022). *An Ensemble Machine Learning Approach for Predicting Type-II Diabetes Mellitus Based on Lifestyle Indicators*. Science Direct. Retrieved April 26, 2024, from https://www.sciencedirect.com/science/article/pii/S2772442522000399

GeeksforGeeks. (2024, April 29). *CatBoost in Machine Learning*. Retrieved April 29, 2024, from https://www.geeksforgeeks.org/catboost-ml/

GeeksforGeeks. (2023, March 31). *Gradient Boosting in ML*. GeeksforGeeks. https://www.geeksforgeeks.org/ml-gradient-boosting/

GeeksforGeeks. (2024, January 3). *What is a Neural Network?* GeeksforGeeks. https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/

Joshi, T. N., & Chawan, P. M. (2018). *Diabetes Prediction Using Machine Learning Techniques*. Ijera. Retrieved April 26, 2024, from https://www.ijera.com/papers/Vol8_issue1/Part-2/C0801020913.pdf

Mujumdar, A., & Vaidehi, V. (n.d.). *Diabetes Prediction using Machine Learning Algorithms*. Science Direct. Retrieved March 21, 2024, from https://www.sciencedirect.com/science/article/pii/S1877050920300557

Qin, Y., Wu, J., Xiao, W., Wang, K., Huang, A., Liu, B., Yu, J., Li, C., Yu, F., & Ren, Z. (2022). *Machine Learning Models for Data-Driven Prediction of Diabetes by Lifestyle Type*. National Library of Medicine. Retrieved March 21, 2024, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9690067/

Saxena, S. (2023, September 13). *Binary Cross Entropy/Log loss for binary classification*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/03/binary-cross-entropy-log-loss-for-binary-classification/#:~:text=Binary%20Cross%20 Entropy%2C%20 also%20 known,binary%20 labels%20of%20a%20 dataset.

*Simple and Reliable Early Prediction of Diabetes*. (2024, January 19). ScienceDaily. Retrieved April 30, 2024, from https://www.sciencedaily.com/releases/2024/01/240119122708.htm#:~:text=Josef%20Hospital%20in%20%20Bochum%2C%20%20Germany,on%20%20January%2C%202%2C%202024

T, D. (2021, December 11). Confusion Matrix Visualization - Dennis T - Medium. *Medium*. https://medium.com/@dtuk81/confusion-matrix-visualization-fc31e3f30fea

Tasin, I., Nabil, T. U., Islam, S., & Khan, R. (2023). *Diabetes Prediction Using Machine Learning and Explainable AI Techniques*. National Library of Medicine. Retrieved March 21, 2024, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10107388/

Teboul, A. (2022). *Diabetes Health Indicators Dataset*. Kaggle. Retrieved April 26, 2024, from https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset/data

Tu J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, *49*(11), 1225–1231. https://doi.org/10.1016/s0895-4356(96)00002-9

UC Business Analytics. (n.d.). *Feedforward Deep Learning Models*. UC Business Analytics R Programming Guide. Retrieved April 30, 2024, from https://uc-r.github.io/feedforward_DNN

Vishwakarma, N. (2024, April 26). *What is Adam Optimizer?* Analytics Vidhya. https://www.analyticsvidhya.com/blog/2023/09/what-is-adam-optimizer/#:~:text=The%20Adam%20 optimizer%2C%20short%20 for,Stochastic%20 Gradient%20 Descent%20with%20 momentum.

Yahyaoui, A., Jamil, A., Rasheed, J., & Yesiltepe, M. (2020, January 23). *A Decision Support System for Diabetes Prediction Using Machine Learning and Deep Learning Techniques*. IEEE Xplore. Retrieved March 21, 2024, from https://ieeexplore.ieee.org/abstract/document/8965556