Predicting Diabetes Using Machine Learning Models on Medical and Demographic Data

ISYE 6740 – Summer 2024

Project Final Report

Team 105: Parth Patel, Anthony Mazza, and Colleen Boyle

Table of Contents

Table of Figures

Table of Tables

Abstract

In this report, we will investigate the correlation between physical attributes and the likelihood of diabetes diagnosis. This will be accomplished by comparing the robustness of six classification models.

Introduction

Diabetes is a chronic disease that affects millions of people worldwide with significant impacts on health and quality of life. According to the World Health Organization, approximately 422 million people have diabetes, and it is a major cause of blindness, kidney failure, heart attacks, stroke, and lower limb amputation [1]. The prevalence of diabetes has been steadily increasing, particularly in low and middle-income countries. Given its widespread impact and the need for effective management and prevention strategies, we were motivated to work on a project involving healthcare data to explore the factors contributing to diabetes.

The dataset used in this project is sourced from Kaggle [2] and contains medical/demographic data from 100,000 patients along with their diabetes status. The data includes features such as age, gender, body mass index (BMI), hypertension, heart disease, smoking history, HbA1c level, and blood glucose level. For our project, we will be forming predictions on diabetes in patients based on their medical history and demographic information. Determining the relationship between patients' physical features to diabetes diagnoses can aid in more accurate prognosis prediction models for future patients at risk of developing diabetes. Table 1 below shows the features for the dataset and some statistics for each.

Table 1: Dataset Features and Descriptions

Feature	Description
Age Important factor - diabetes is more commonly diagnosed in older adults.	• Average age of 41.9 years with a standard deviation of 22.52 • Youngest age is 0.08 years (30 days) • Maximum age is 80 years
Gender Refers to the biological sex of the individual, impacting susceptibility to diabetes.	• 41% male, 59% female, ~ 0% other (18 patients)
Body Mass Index (BMI) A measure of body fat based on weight and height, linked to diabetes risk. Typical BMI categories: underweight (<18.5), normal (18.5-24.9), overweight (25-29.9), obese (≥30).	• Average BMI of 27.32 with a standard deviation of 6.64 • Minimum BMI of 10.01 • Maximum BMI of 95.69
Hypertension A medical condition in which the blood pressure in the arteries is persistently elevated.	• Binary value (0 = no hypertension, 1 = diagnosed with hypertension) • Average value of 0.075 with a standard deviation of 0.263
Heart Disease	• Binary value (0 = no heart disease, 1 = diagnosed with heart disease)

Feature	Description
A medical condition that is associated with an increased risk of developing diabetes.	• Average value of 0.040 with a standard deviation of 0.195
Smoking History A risk factor for diabetes and its complications.	• Categorical feature: current, former, current, never, ever, and no Info
HbA1c Level Measures average blood sugar level over the past few months. Higher levels indicate a greater risk of diabetes (≥6.5%).	• Average HbA1c level of 5.523 with a standard deviation of 1.071 • Minimum HbA1c level of 3.50 • Maximum HbA1c level of 9.00
Blood Glucose Level Refers to the amount of glucose in the bloodstream at any given time. High blood glucose levels are a key indicator of diabetes.	• Average glucose level of 138.06 with a standard deviation of 40.71 • Minimum glucose level of 80.0 • Maximum glucose level of 300.0

Methodology

Software Environment

We will be using the Python programming language along with various machine learning and statistics packages to perform our analysis. The code will be executed using a Jupyter notebook.

Data Preprocessing

To prepare the data for modeling, we first began by identifying and examining the categorical columns. We separated the dataset into features and the target variable, with *diabetes* as the target. Categorical variables, specifically *gender* and *smoking_history*, were encoded into numerical values using sklearn's LabelEncoder which is necessary for algorithms requiring numerical input.

Next, we scaled the numerical features using StandardScaler from sklearn to ensure uniformity in their contribution to the model. This step involved standardizing the features by removing the mean and scaling them to unit variance which improves the model's performance.

Outlier detection was performed using techniques such as the Z-score and Interquartile Range (IQR) methods and no significant outliers were found. After preprocessing, we split the data into 80% training and 20% testing sets. Overall, these steps standardize the input data which should greatly improve the models' ability to learn meaningful patterns and make accurate predictions.

Classification Models

The classification models we will use in analyzing our data are listed below. Each is categorized by three machine learning algorithm types – Parametric / Non-Parametric, Linearity / Non-Linearity, and Supervised / Unsupervised. These are explained in more detail following the classification model descriptions. This is also depicted in Table 2.

- KNN - The k-nearest neighbors' algorithm is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.

- SVM - A supervised machine learning algorithm that classifies data by finding an optimal line or hyperplane that maximizes the distance between each class in an N-dimensional space.
- Logistic Regression - Logistic regression is used for binary classification where we use sigmoid function, that takes input as independent variables and produces a probability value between 0 and 1.
- Random Forest - An ensemble learning method for classification and other tasks that operates by constructing many decision trees over a training time. For classification tasks, the output of the random forest is the class selected by the most trees.
- Naive Bayes – Naive Bayes is a family of linear "probabilistic classifiers" which assumes that the features are conditionally independent, given the target class.
- Neural Network - This learning model is inspired by the structure of biological neural networks in brains. Nodes are connected via edges in which they communicate via a signal. The output of each neuron is computed by some non-linear function of the sum of its inputs, called the activation function. The strength of the signal at each connection is determined by a weight, which adjusts during the learning process.

Table 2: Classification Models vs. Algorithm Type

Classification Model	Algorithm Type		
KNN	Non-Parametric	Non-Linear	Supervised
SVM	Non-Parametric	Non-Linear (with Kernel)	Supervised
Logistic Regression	Parametric	Linear	Supervised
Random Forest	Non-Parametric	Non-Linear	Supervised
Naive Bayes	Parametric	Linear	Supervised
Neural Network (simple)	Parametric	Non-Linear	Supervised

- Parametric – Summarizes a dataset of a fixed number of parameters and assumes a specific functional form for the data distribution. Non-parametric method does not have a set dataset size and does not yield a specific functional form for the data. Non-parametric is best for large datasets by creating a mapping function based on patterns within the data.
- Linearity – Assumes that the features are conditionally independent, given the target class. Linearity numerically shows the relationship between input and output data via a best fit straight line. Non-linearity is better for larger datasets with more complex relationships.
- Supervised Learning – A category of machine learning that uses labeled datasets to train algorithms to predict outcomes and recognize patterns. Unsupervised learning uses unlabeled datasets to do the same.

Analysis

The tuning, libraries and functions are explained for each model. Feature importance is explored through Gini Importance, Mean Decrease Accuracy, and Permutation Feature Importance. Performance for the models were compared using training accuracy, test accuracy, and F1 score. Further comparisons were made using confusion matrix metrics along with ROC curves. Decision boundaries were also explored for the Random Forest and Neural Network models comparing the highest scoring feature against the other features.

Exploratory Data Analysis

Before fitting any models to the data, it is crucial to perform exploratory data analysis (EDA) to understand the underlying patterns, distributions, and relationships within the dataset. This initial examination helps in identifying potential issues such as outliers, missing values, and the overall structure of the data.

To begin our EDA, we first used the *describe()* function on the Pandas data frame that contains the dataset. This provides a summary of the dataset's numerical features and metrics such as the minimum, maximum, mean, and standard deviation. These statistics give us a foundational understanding of the dataset's characteristics and can be seen in Table 3.

Table 3: Statistical Metrics of Each Feature

	Age	Hypertension (binary)	Heart Disease (binary)	BMI	HbA1c level	Blood glucose level	Diabetes (binary)
count	100,000	100,000	100,000	100,000	100,000	100,000	100,000
mean	41.886	0.075	0.039	27.321	5.528	138.058	0.085
std	22.517	0.263	0.195	6.637	1.071	40.708	0.279
min	0.080	0.000	0.000	10.010	3.500	80.000	0.000
25%	24.000	0.000	0.000	23.630	4.800	100.000	0.000
50%	43.000	0.000	0.000	27.320	5.800	140.000	0.000
75%	60.000	0.000	0.000	29.580	6.200	159.000	0.000
max	80.000	1.000	1.000	95.690	9.000	300.000	1.000

To understand the prevalence of diabetes within our dataset, we wanted to determine what proportion of the dataset is classified as diabetic. This is illustrated as a pie chart in Figure 1.

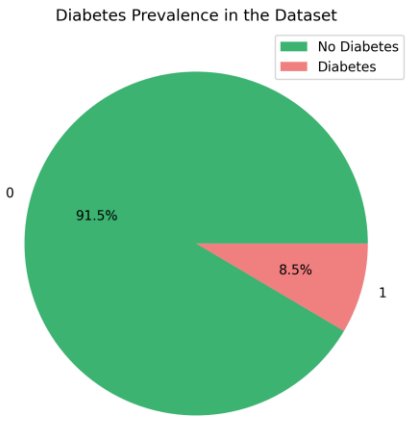

Figure 1: Diabetes Prevalence in the Dataset

As seen from the pie chart above, only 8.5% of the data is diabetic which is a small fraction of our dataset and indicates an imbalance of the target variable.

The distribution of the numerical features can also be visualized as histograms (Figure 2) and box and whisker plots (Figure 3) to better understand the data.

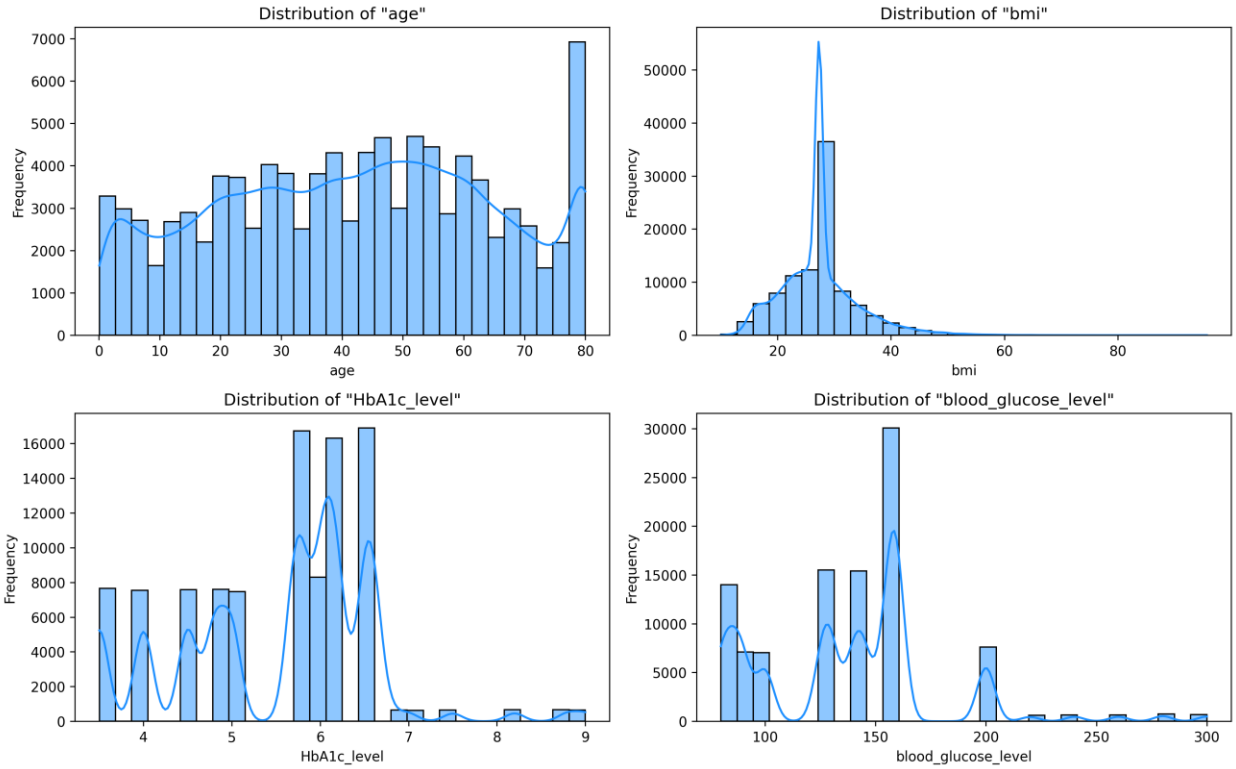

Figure 2: Dataset Histograms

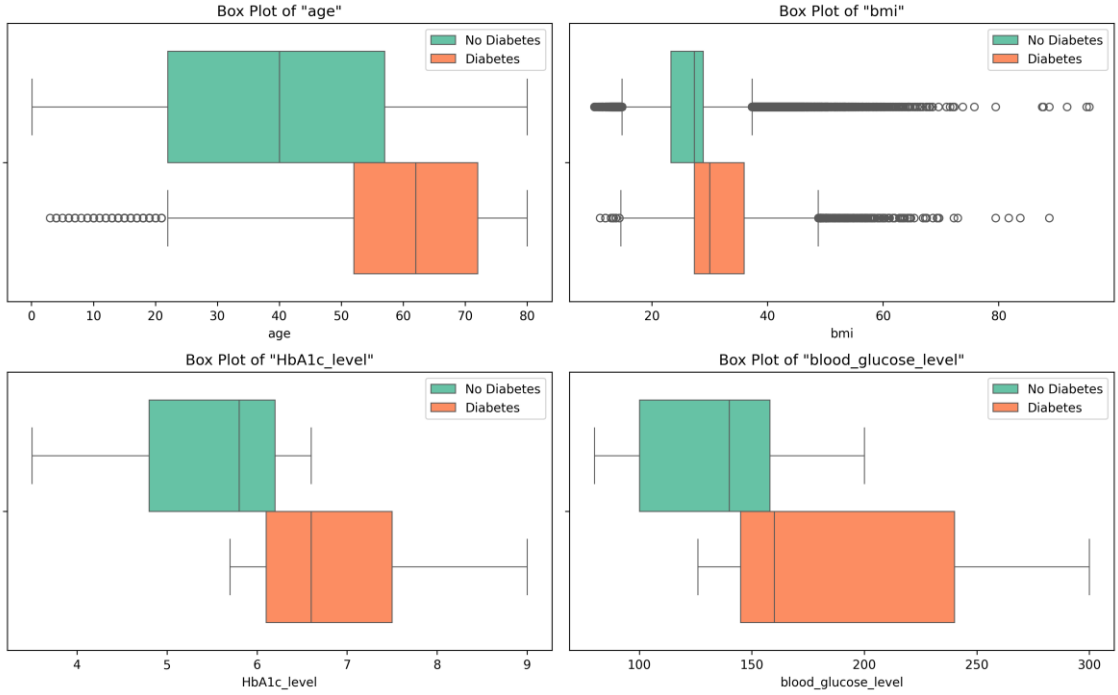

Figure 3: Dataset Whisker Plots

Additionally, we can examine bar plots of the binary features, *hypertension* and *heart_disease*, to see the proportion of the data that is diagnosed with each condition. This can be seen in Figure 4.

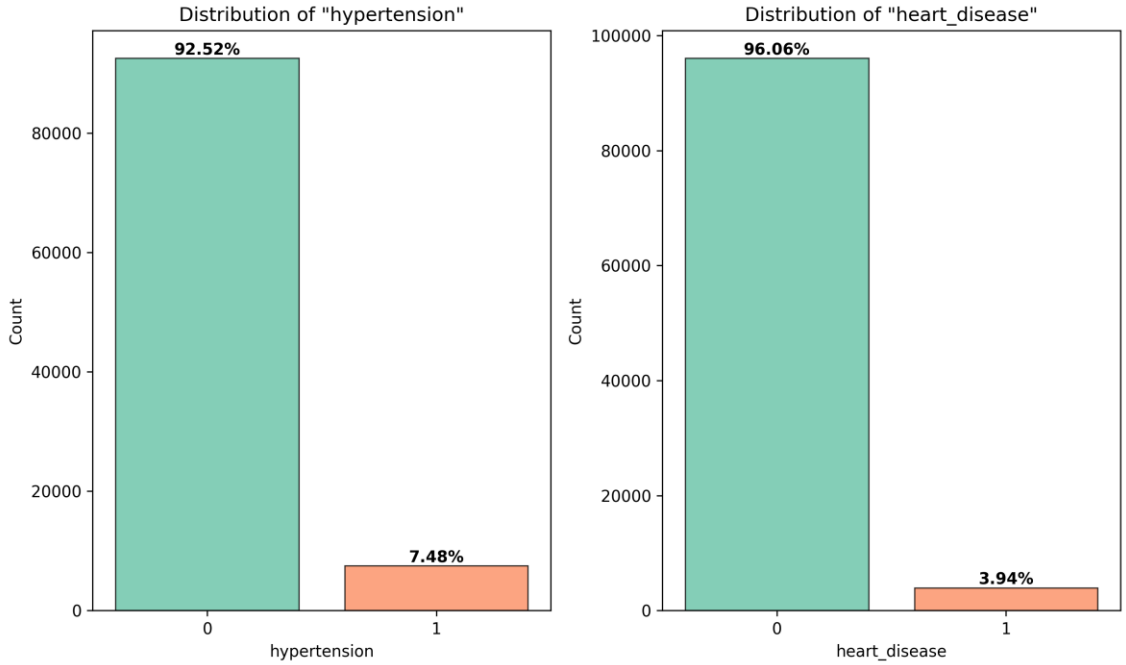

Figure 4: Bar Graphs for Hypertension and Heart Disease Binary Features as a Portion of the Dataset

Similarly, we can create bar plots for the categorical features, *smoking_history* and *gender*, to visualize their distributions within the dataset, as illustrated in Figure 5.

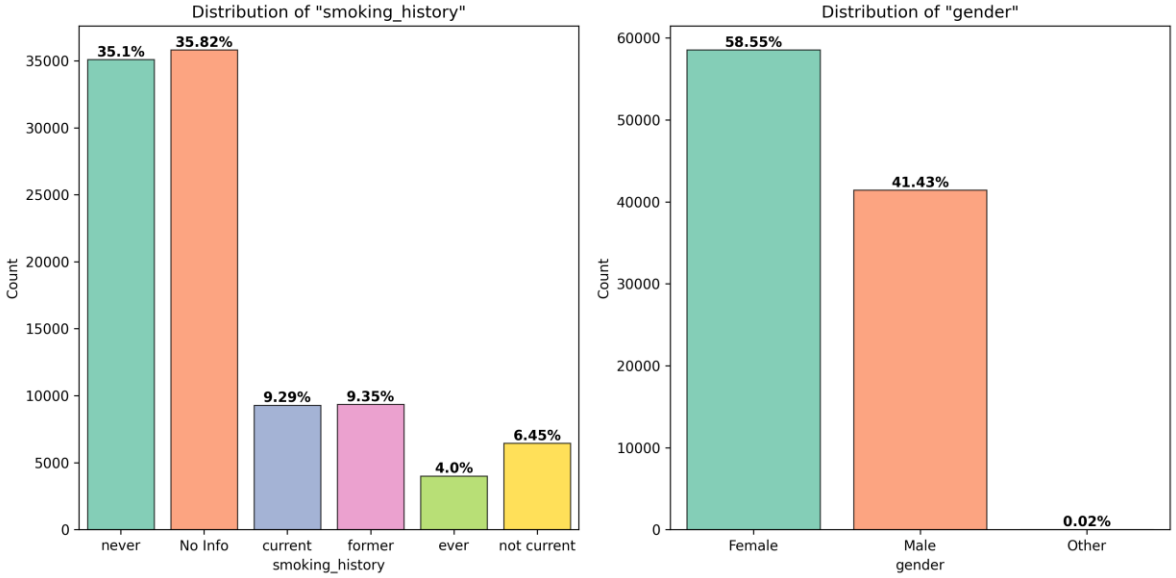

Figure 5: Bar Graphs for Smoking History and Gender Categorical Features as a Portion of the Dataset

Another crucial step in data exploration is examining the relationships between various features and the target variable, *diabetes*. We can examine bar plots of the binary features, *hypertension* and

heart_disease, to better understand the distribution of diabetic cases in relation to these conditions. This is depicted in Figure 6.

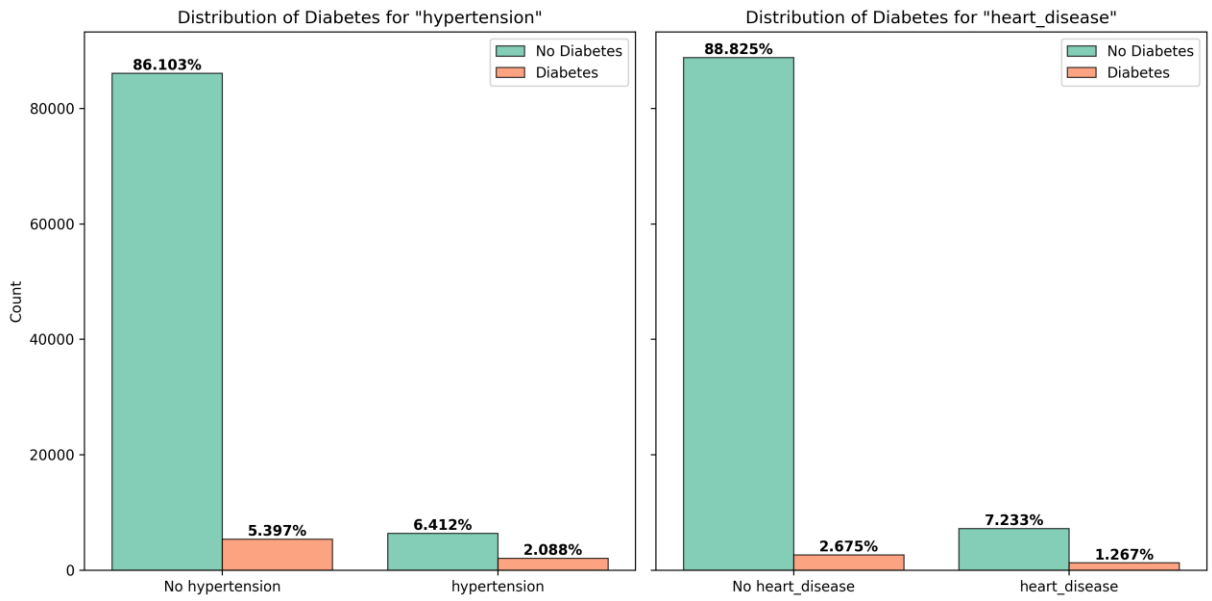

Figure 6: Bar Graphs for Hypertension and Heart Disease Binary Features vs Diabetes Diagnosis

Similarly, we can visualize the proportion of each gender that is diabetic in the dataset (see Figure 7).

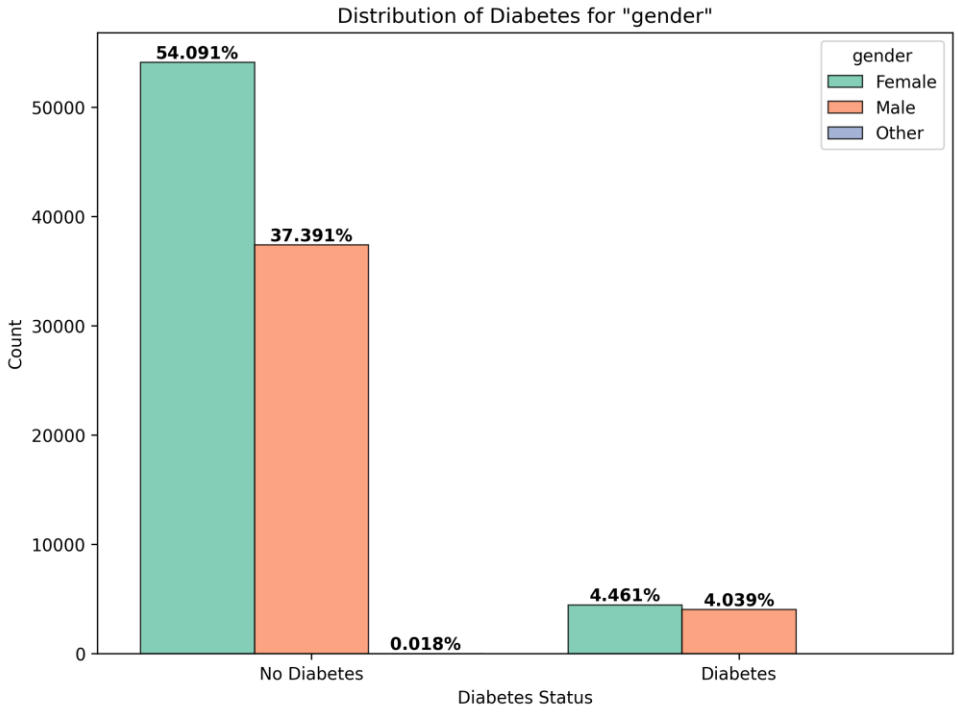

Figure 7: Bar Graphs for Smoking History and Gender Categorical Features vs Diabetes Diagnosis

Another important step in our exploration is to analyze the relationship between age and various numerical features to observe how conditions like heart disease, blood glucose levels, and hypertension

vary with age (shown in Figure 8). This will help us verify whether the dataset aligns with expectations, as it is commonly observed that the risk of these conditions tends to increase with age.

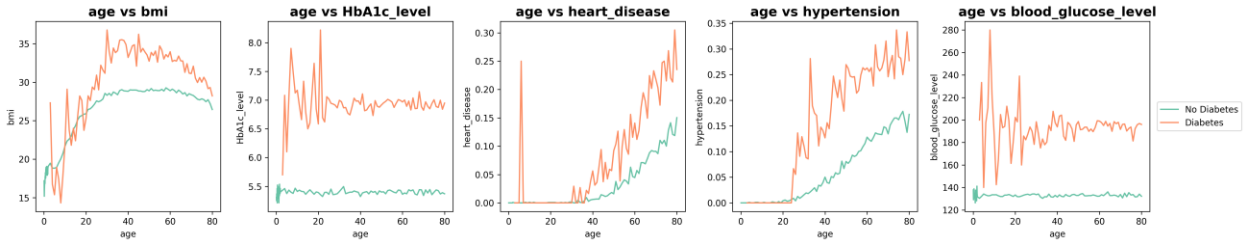

Figure 8: Comparison Between Age and Various Features vs Diabetes Diagnosis

As evident from the plots above, the risk of conditions like heart disease, hypertension, elevated HbA1c levels, and high blood glucose levels tends to be higher for diabetic patients. Additionally, these conditions increase at a significantly greater rate with age for those diagnosed with diabetes compared to those without diabetes.

Finally, the most important step in our exploration is constructing a correlation matrix to analyze the relationships between variables. To do this, we created a copy of the data where we used one-hot encoding to convert categorical features into numerical form and then created a heatmap to identify and visualize the most significant correlations among the features.

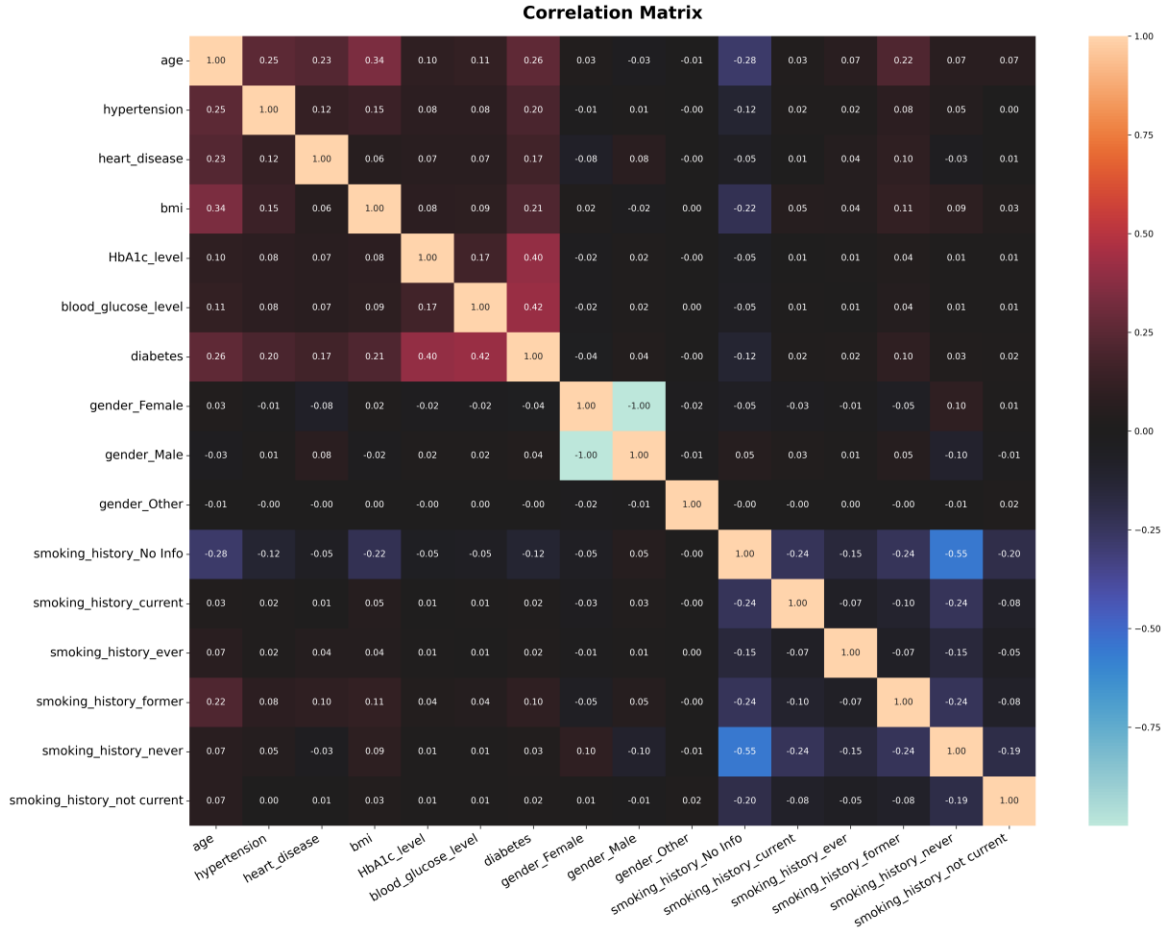

Figure 9: Correlation Matrix

As seen from the heatmap in Figure 9, there are many expected correlations between the features. Some features are positively correlated as expected such as age with BMI, hypertension, and heart disease, while others are slightly negatively correlated, such as the female gender with heart disease. Some notable correlations are with the diabetes feature itself which is shown more clearly in the plot on Figure 10.

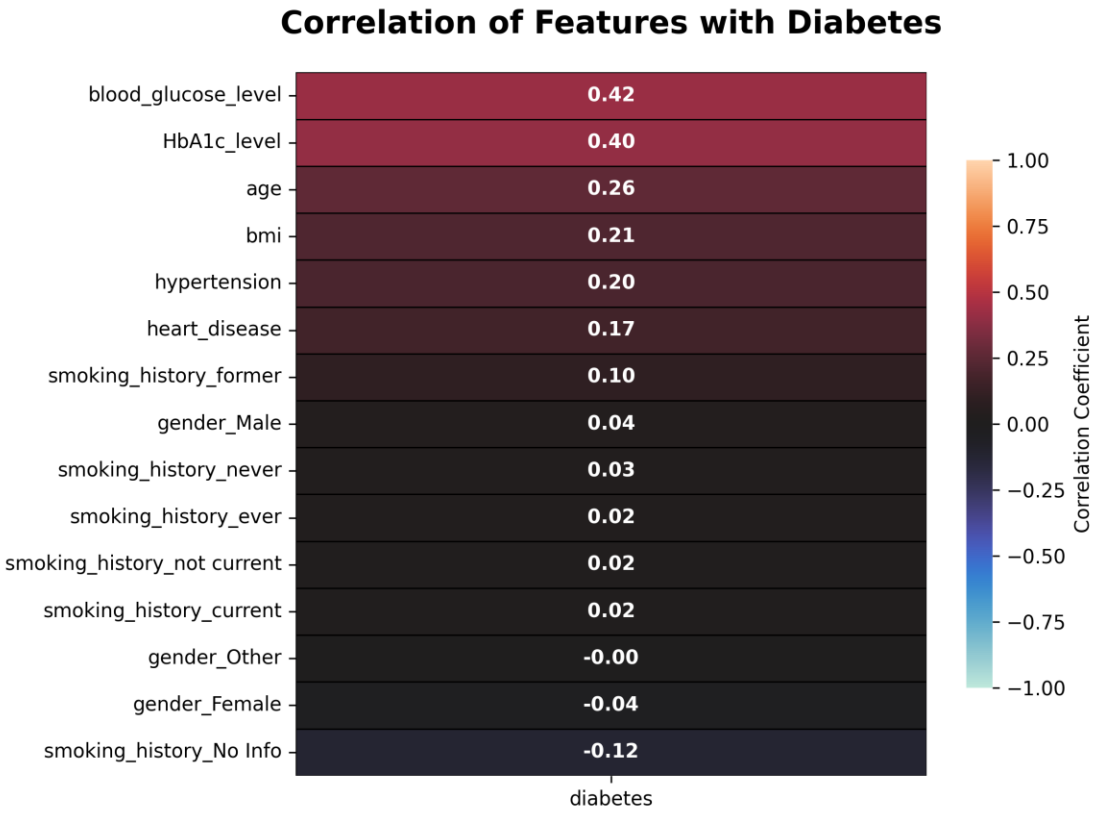

Figure 10: Heatmap of Correlation Between Feature and Diabetes Diagnosis

As expected, diabetes shows a strong positive correlation with several factors including blood glucose levels, HbA1c levels, age, BMI, hypertension, heart disease, and a history of smoking. The correlation of diabetes with gender reveals a slight positive association with males whereas females show a negative correlation. Additionally, individuals with no prior smoking history tend to show a negative correlation with diabetes. These observed correlations align with established medical knowledge and offer valuable insights into our dataset.

Analysis and Results

The tuned models with their parameters are displayed in Table 4 along with the libraries used. These are the models used for performance comparison. Model performance was compared using training accuracy, test accuracy, and F1 score. Performance was further measured using confusion matrix metrics. Feature importance was explored using Gini Importance, Mean Decrease Accuracy, and Permutation Feature Importance; these indicators compare which features contribute most to the results.

Table 4: Library/Function of Each Model

Model	Library / Function
Random Forest	Function: RandomForestClassifier(max_depth=9, max_features='log2', max_leaf_nodes=6, n_estimators=50) found using RandomizedGridSearchCV Libraries: from sklearn.ensemble import RandomForestClassifier from sklearn.model_selection import RandomizedSearchCV
Neural Net	Function: MLPClassifier(alpha=0.01, hidden_layer_sizes=(500,), learning_rate='adaptive', max_iter=500) found using GridSearchCV Libraries: from sklearn.neural_network import MLPClassifier from sklearn.model_selection import GridSearchCV
Logistic Regression	Function: LogisticRegression(max_iter=200, solver='liblinear') Libraries: from sklearn.linear_model import LogisticRegression
Naïve Bayes	Function: GaussianNB(var_smoothing = 1e-3) Libraries: from sklearn.naive_bayes import GaussianNB
K-Nearest Neighbor (KNN)	Function: KNeighborsClassifier(n_neighbors=5) Libraries: from sklearn.neighbors import KNeighborsClassifier
Support Vector Machine (SVM)	Function: SVC(probability=True, kernel='linear', random_state=42) Libraries: from sklearn.svm import SVC

Model Performance

We compared model performance using training accuracy, test accuracy, F1, and confusion matrix metrics. These performance indicators allow us to compare the models across a variety of perspectives.

Table 5 depicts the results for each model where training accuracy, test accuracy, and the F1 score (weighted) were determined. These are ranked from best performance to worst according to the highest value for the weighted F1 score. The F1 score is a measure that combines both precision and recall and is very useful in the case of an imbalanced dataset because it accounts for both false positives and false negatives. In our case, the dataset is imbalanced since only 8.5% of the data is diabetic. The test accuracy could also be used to determine the best performance as it reflects how well the model generalizes to new, unseen data. Training accuracy is useful for determining if the model is overfitted but is not suitable for ranking model performance. We can see from Table 5 that the best performing model in terms of the three performance metrics is Neural Networks.

Table 5: Model vs. Performance Metrics

Ranking	Model	Train Accuracy	Test Accuracy	F1 (weighted)
1	Random Forest	0.9721	0.9709	0.9682
2	Neural Network	0.9724	0.9706	0.9680
3	KNN	0.9701	0.9617	0.9587
4	Logistic Regression	0.9606	0.9590	0.9559
5	SVM	0.9612	0.9597	0.9555
6	Naïve Bayes	0.9034	0.9024	0.9099

Other performance metrics that are typically useful in a confusion matrix are precision, sensitivity (recall), specificity, negative predictive value, and overall accuracy. Table 6 depicts the confusion matrix metrics for each model. They are ranked according to best performance for overall accuracy. This gives insight into the overall general performance of the model. This paired with the other metrics give a more complete understanding of the model performance. In cases such as many false positives/negatives or imbalance in the data, the other metrics would be more important to assess and have influence in ranking the best performing model. In the case of our data, there were not many instances of false positives/negatives or imbalance in the data. Therefore, it is acceptable to rank performance based on total accuracy of each model.

Table 6: Model vs. Confusion Matrix Metrics

Ranking	Model	Precision	Sensitivity (Recall)	Specificity	Negative Predictive Value	Accuracy
1	Random Forest	1.0000	0.9709	1.0000	0.9692	0.9709
2	Neural Network	0.9863	0.9706	0.9991	0.9696	0.9706
3	KNN	0.8934	0.9617	0.9929	0.9661	0.9617
4	SVM	0.9189	0.9597	0.9951	0.9620	0.9597
5	Logistic Regression	0.8647	0.9590	0.9908	0.9652	0.9590
6	Naïve Bayes	0.4540	0.9024	0.9266	0.9652	0.9024

The performance of the models varied due to the inherent characteristics of each algorithm and their ability to handle different aspects of the data. The Random Forest model achieved the highest accuracy and F1 score due to its ensemble nature, combining predictions from multiple decision trees to produce a robust and generalized model capable of capturing non-linear relationships and feature interactions. Neural Networks also performed well due to its ability to learn intricate patterns through multiple layers of neurons.

Logistic Regression and SVM, both linear models, had high accuracies. Logistic Regression as a simple linear model struggled to capture non-linear relationships and this may have also been true for SVM as it relies on a linear kernel. Regardless, both models are highly interpretable and computationally efficient, making them useful for understanding underlying patterns. KNN performed adequately but its sensitivity to the number of neighbors and computational inefficiency with large datasets likely affected the precision and therefore the overall results. Naive Bayes had the lowest performance, and this might be

explained by the feature independence assumption which is rarely true in real-world data and particularly in medical datasets where features often interact.

ROC Curve

An ROC Curve is a graph used to represent the True Positive Rate (TPR: Sensitivity / Recall) vs. False Positive Rate (FPR: 1-Specificity). This helps evaluate the performance of a binary classification model by assessing the trade-offs between sensitivity and specificity. AUROC is the area under the ROC curve. The best AUROC value for model prediction performance would be 1. This indicates perfect classification (the model correctly identifies all positive and negative instances without any mistakes). Therefore, the goal is to have an AUROC value closest to 1 [5].

Each model was first processed by setting the number of samples to 80000 and setting random seed and *random_state* to 42 to ensure reproducibility and control randomness. Cross-validation was then utilized to hypertune and reduce overfitting. Once complete, the ROC Curves were determined. All are illustrated for comparison in Figure 11 below. The AUROC values are indicated in the figure's key. From this, we can conclude that the Neural Networks model was the most accurate at distinguishing between classes.

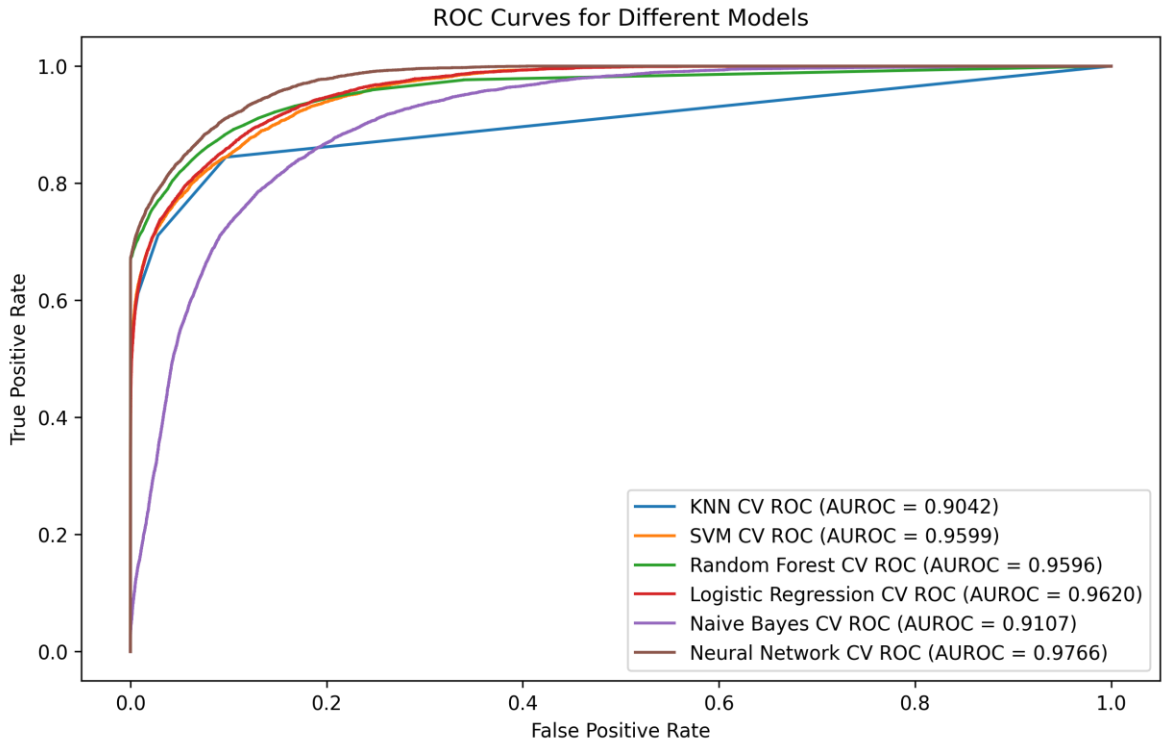

Figure 11: ROC Curve Comparison for All Models

Feature Importance

Feature importance refers to techniques that assign a score to input features based on how useful they are at predicting the target outcome variables. These scores help us understand how each feature contributes to the model's prediction performance. Each model type has different ways of determining feature importance. For example, KNN, SVM, Neural Networks, Naive Mayes, and Random Forest can use permutation importance. This is done by randomly shuffling each feature and observing the change in

model performance, you can estimate the importance of each feature. The idea is that if a feature is important, shuffling its values should significantly decrease the model's performance. For Logistic Regression, an example of feature importance calculation is by standardizing coefficients. Standardizing the features before fitting the model, you can directly compare the coefficients to assess the relative importance of each feature.

All these models can use different methods as well to determine feature importance. Determining the best method for feature importance depends on the specific goals, the nature of the data, and the type of model being used. In our case, we want to use Random Forest's Gini Importance (also known as mean decrease impurity – MDI) method as it is particularly useful in measuring feature importance in a way that captures interactions between features and handles both numerical and categorical data effectively. It calculates how often a randomly chosen element from the set would be incorrectly labeled if it were randomly labeled according to the distribution of labels in the subset. Its equation is shown below in Figure 12 [3]. The Gini Importance for the features of our dataset were calculated and are shown in Table 7. It was determined that HbA1c is the most significant feature in our dataset. The second most is the blood glucose level followed by age.

$$Imp(X_m) = \frac{1}{N_T} \sum_{T} \sum_{t \in T: v(s_t) = X_m} p(t) \Delta i(s_t, t)$$

Figure 12: Gini Importance Equation [3]

Table 7: Random Forest Gini Importance

Feature	Gini Importance
HbA1c_level	0.666942
blood_glucose_level	0.320800
age	0.012258
bmi	0
hypertension	0

Mean Decrease Accuracy (MDA) measures the average reduction in model accuracy across all out-of-bag cross-validated predictions when a particular feature undergoes permutation after training but before making predictions. It offers a comprehensive evaluation of a feature's significance in influencing the model's performance. MDA inherently adopts a global perspective, taking into account the entire dataset to gauge the importance of individual features [7]. MDA was computed for the Random Forest model and is depicted in Table 8.

Table 8: Random Forest Mean Decrease Accuracy

Feature	Decrease in Accuracy
HbA1c_level	0.05615
blood_glucose_level	0.04790

Permutation feature importance is a model inspection technique that measures the contribution of each feature to a fitted model's statistical performance on a given tabular dataset. This technique is particularly useful for non-linear or opaque estimators and involves randomly shuffling the values of a single feature and observing the resulting degradation of the model's score [1]. By breaking the relationship between

the feature and the target, we determine how much the model relies on such a particular feature [6]. This can be seen performed for Neural Networks in Table 9.

Table 9: Neural Network – Permutation Feature Importance

Feature	Permutation Feature Importance
HbA1c_level	0.057 +/- 0.000
blood_glucose_level	0.044 +/- 0.000
age	0.001 +/- 0.000
bmi	0.001 +/- 0.000
heart_disease	0.001 +/- 0.000

HbA1c_level was the most important feature from all three methods with *blood_glucose_level* being second.

Decision Boundaries

Since every metric for feature importance indicated that HbA1c was the most significant, this will be compared to the other features in our decision boundary visualizations.
Figure ***13*** and
Figure ***14*** illustrate the decision boundary for two model types. Neural Networks and Random Forest were selected to illustrate the effects of decision boundaries on the dataset. Here, blue indicates a positive diabetes diagnosis while red indicates a negative diabetes diagnosis.

A decision boundary compares data and separates them by class. This aids in visually understanding how a classifier differentiates between classes in a dataset which can lend insight into how the model can perform on unseen data. In our cases, we will be using complex decision boundaries [4]. To avoid overfitting the data cross validation was utilized. We can see in the decision boundary figures that HbA1c allows for good separation across the variables indicating its importance relative to other features.

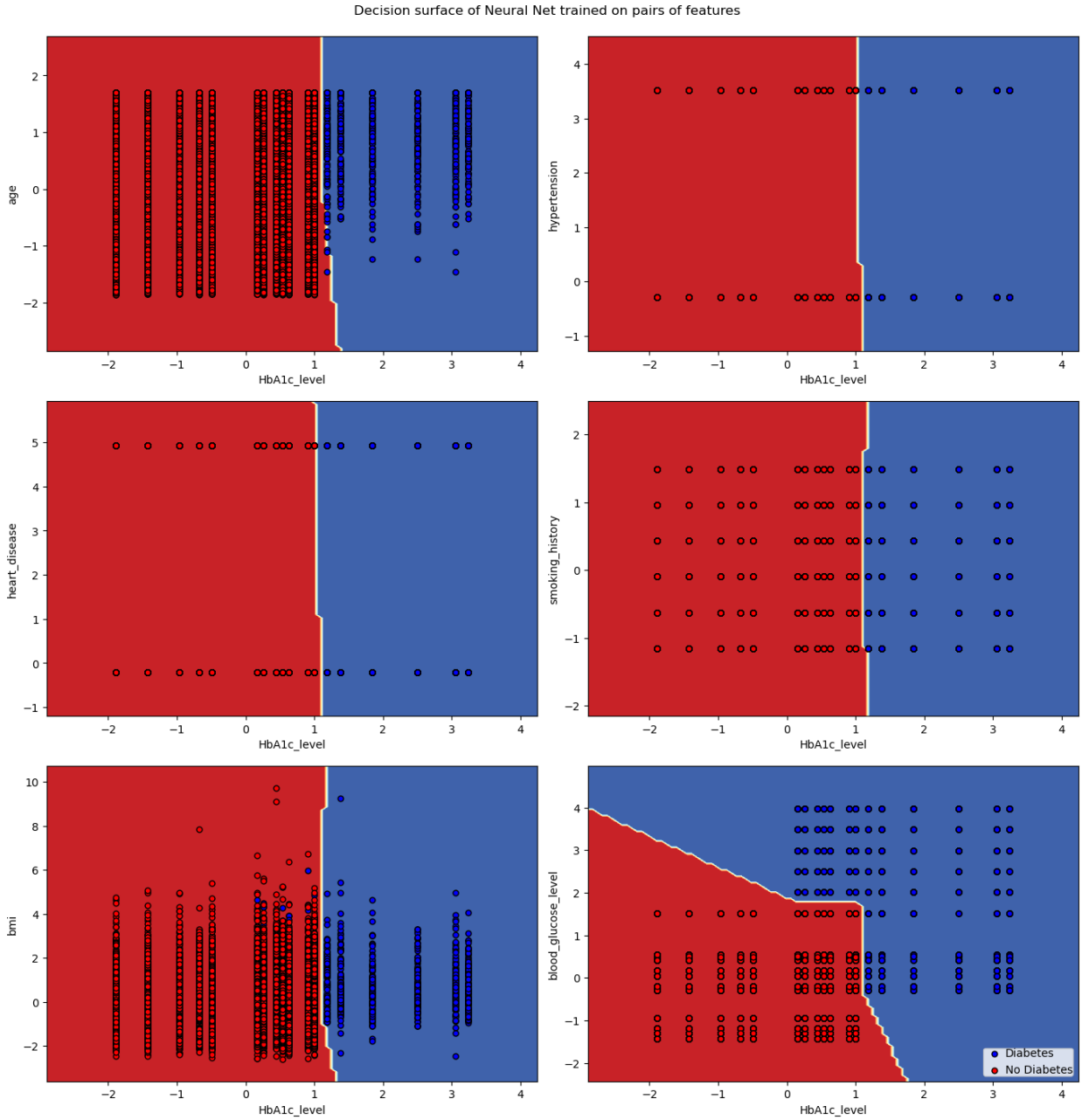

Figure 13: Neural Networks Decision Boundary HbA1c Level vs. Diabetes

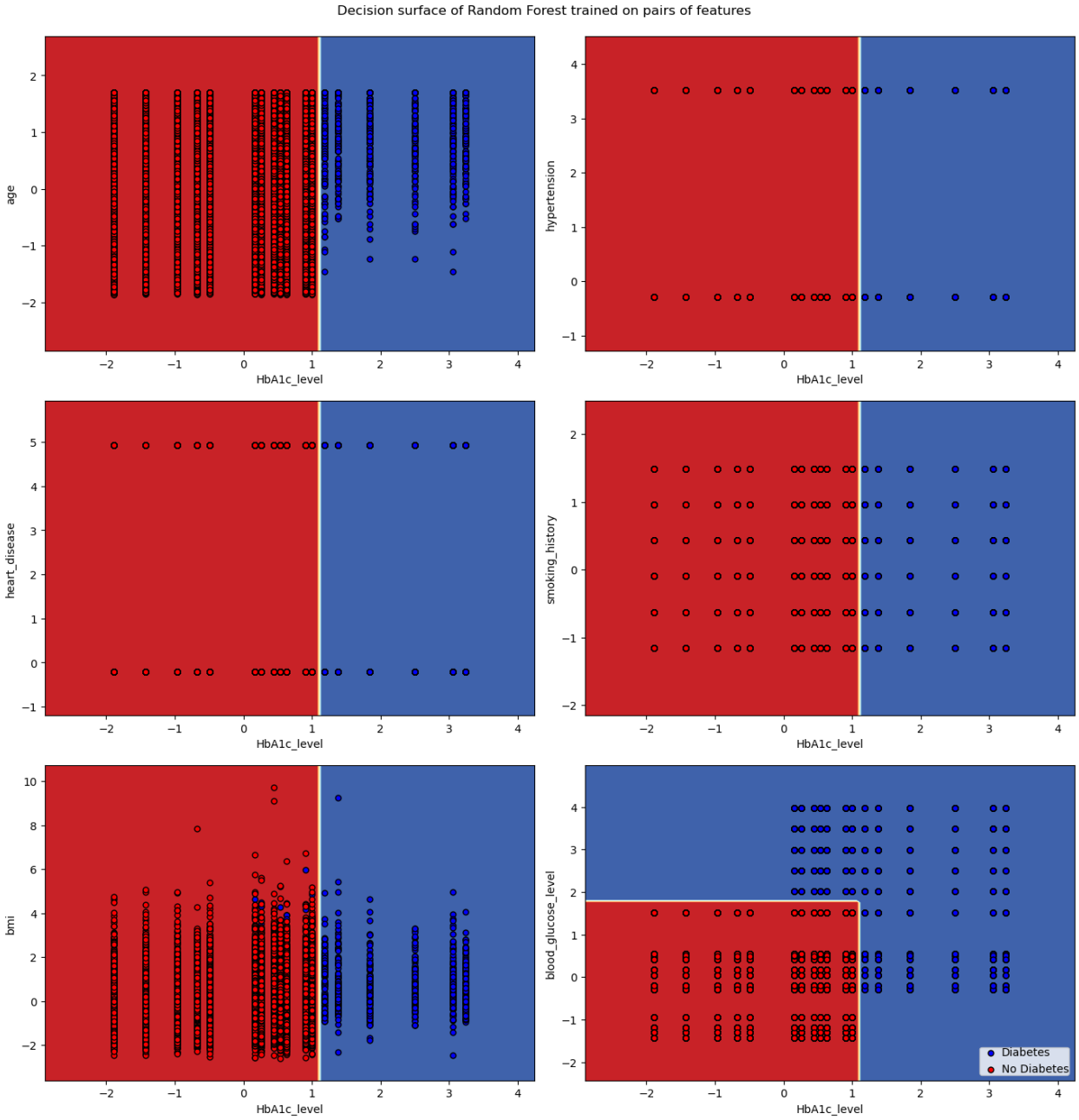

Figure 14: Random Forest Decision Boundary HbA1c Level vs. Diabetes

Conclusion

Through our analysis, *HbA1c_level* and *blood_glucose_level* are the two most significant features for predicting diabetes. Further investigation into these features could involve examining their correlations or creating a composite feature to assess its predictive power against diabetes.

We employed multiple methods to rank model performance. Each provided unique insights into the models' predictive abilities and comparing gives a comprehensive view of the model performance. Prioritization of these metrics is based on several factors. No performance metric is sufficient by itself in determining the best model. Each needs to be investigated to understand the overall performance of the data. This is especially important for dataset characteristics such as imbalance, outliers, false positives, etc. Another factor is the end goal of the data analysis. Some metrics may highlight different aspects of model performance relevant to a specific initial question about the data. Therefore, this would hold more importance.

In our analysis of the dataset, our goal was to understand which model had the highest overall prediction performance accuracy. After investigating the various metrics for performance, Random Forest demonstrated the best performance. Even in the cases where Random Forest did not rank first (such as for AUROC), it was within a few tenths of a percentage point different from the best performer in that metric. Thus, we can conclude that Random Forest model is the most effective predictive model for understanding the relationship between diabetes and the given features.

Roles and Responsibilities

Project Task	Team Member Responsible
Proposal, modeling logistic regression and naïve bayes, final report (introduction, data processing, EDA, model analysis)	Parth Patel
Proposal, modeling neural net and random forest, final report (feature importance, decision boundary plots, Gini calculations, model analysis)	Anthony Mazza
Proposal, modeling KNN and SVM, final report (ROC Curve, model analysis, conclusion, citations and references)	Colleen Boyle

Bibliography

[1] World Health Organization. (2021). Diabetes.
https://www.who.int/news-room/fact-sheets/detail/diabetes
[2] Mohammad, Mustafa. Kaggle. Diabetes Prediction Dataset.
https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset/data.
[3] Black, Sam. Medium. 28 October 2020. *Calculating a Feature's Importance with Gini Importance.*
https://sam-black.medium.com/calculating-a-features-importance-with-xgboost-and-gini-impurity-3beb4e003b80.
[4] Sahu, Suchismita. Medium. 6 September 2021. *Decision Boundary For Classifiers: An Introduction*.
https://medium.com/analytics-vidhya/decision-boundary-for-classifiers-an-introduction-cc67c6d3da0e.
[5] Dash, Shailey. Medium. 19 October 2022. *Understanding the ROC and AUC Intuitively*.
https://medium.com/@shaileydash/understanding-the-roc-and-auc-intuitively-31ca96445c02.
[6] *Permutation feature importance*
https://scikit-learn.org/stable/modules/permutation_importance.html
[7] *Feature Importance with Random Forests*
https://www.geeksforgeeks.org/feature-importance-with-random-forests/