Machine Learning Methods for Diabetes Classification:

Evaluating Logistic Regression, SVM, Decision Trees, Random Forests, and AdaBoost

Hamza Alshamy

April 30, 2024

Pre-processing and Exploratory Data Analysis (EDA)

Pre-processing and exploratory data analysis (EDA) were conducted in three main steps to prepare the dataset for predictive modeling.

Data Recoding and Renaming

Initially, the variable *Sex* was recoded from being represented as 1 (male) and 2 (female) to a binary format, 0 for female and 1 for male, for consistency across the dataset. This variable was subsequently renamed to *Male* to clearly reflect the binary gender representation.

One-hot Encoding and Multicollinearity Avoidance

Four categorical variables, namely *AgeBracket*, *EducationBracket*, *IncomeBracket*, and *Zodiac*, were subjected to one-hot encoding to transform them into a format suitable for machine learning models. To prevent multicollinearity—a condition where one variable can be linearly predicted from the others with a substantial degree of accuracy—one category from each of the newly created dummy variables was dropped. This step ensures that the models remain identifiable and their parameters estimable.

Standardization of Interval Variables

Three interval variables, *BMI*, *PhysicalHealth*, and *MentalHealth*, were standardized to have a mean (μ) of zero and a standard deviation (σ) of one. Standardization is crucial for models sensitive to variable scales, such as Logistic Regression and Support Vector Machine (SVM), facilitating a more reliable interpretation of model coefficients.

Train/Test Split and Imbalance Addressing

The dataset was split into training and testing sets with an 80% to 20% ratio to evaluate the models' generalizability. Additionally, it was noted that the dataset is imbalanced with respect to the outcome variable, diabetes. Specifically, out of 253,680 observations, 218,334 are non-diabetic, and only 35,346 are diabetic. This imbalance implies that a naive model predicting every individual as non-diabetic would achieve an accuracy rate of approximately 86.07%, highlighting the necessity for metrics that take the imbalance into account.

1 Logistic Regression

Using the prescribed dataset from the pre-processing stage, a logistic regression model using `sklearn` was built to classify diabetes. The metrics used to assess the model's performance and generalizability are AUC score, training and test accuracy, Matthews Correlation Coefficient (MCC), and Precision & Recall. As explained in the EDA section, since the dataset is imbalanced, accuracy in itself is not a good

metric; however, we can learn from accuracy if the model is overfitting or if it is generalizable. Further, to address the imbalance, the parameter `class_weight` was adjusted to "balanced". We see from Table 1 below that the training and test accuracy are similar, which indicates good generalizability. Notice that the moderate recall for diabetes cases comes at the cost of precision and accuracy.

Model	AUC	Training Accuracy	Test Accuracy	MCC	Precision (Diab.)	Recall (Diab.)
Logistic	0.828	0.731	0.730	0.363	0.31	0.78

Table 1: Logistic Regression Model Performance Metrics

The logistic regression model achieved an AUC of 0.828 and an MCC of 0.3627, indicating a good ability to distinguish between the classes but with moderate correlation between the observed and predicted classifications. Figure 1 visualizes the Area under the ROC curve.

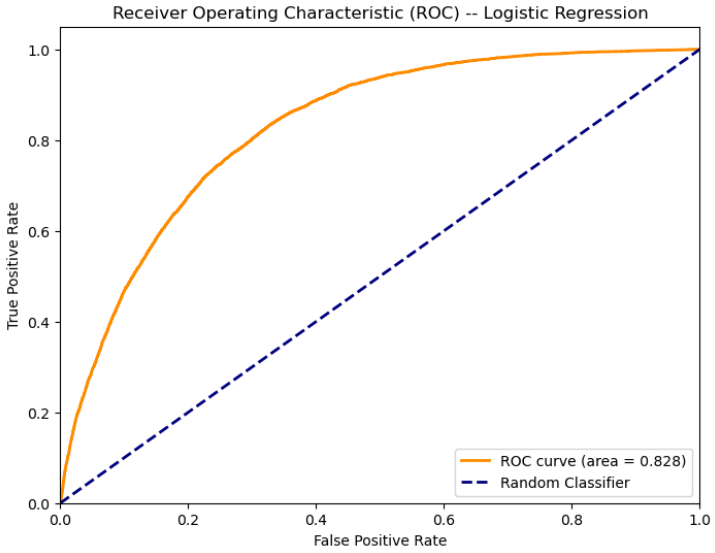


Figure 1: Receiver Operating Characteristic (ROC) – Logistic Regression

The impact of each feature on the model's AUC was explored by systematically dropping each feature and recalculating the AUC to identify the most significant predictors. That is, the variable associated with the most significant drop in AUC is considered the best predictor. Observe from Figure 2 that the best predictor from the Logistic model is General Health followed by BMI and High Blood Pressure.

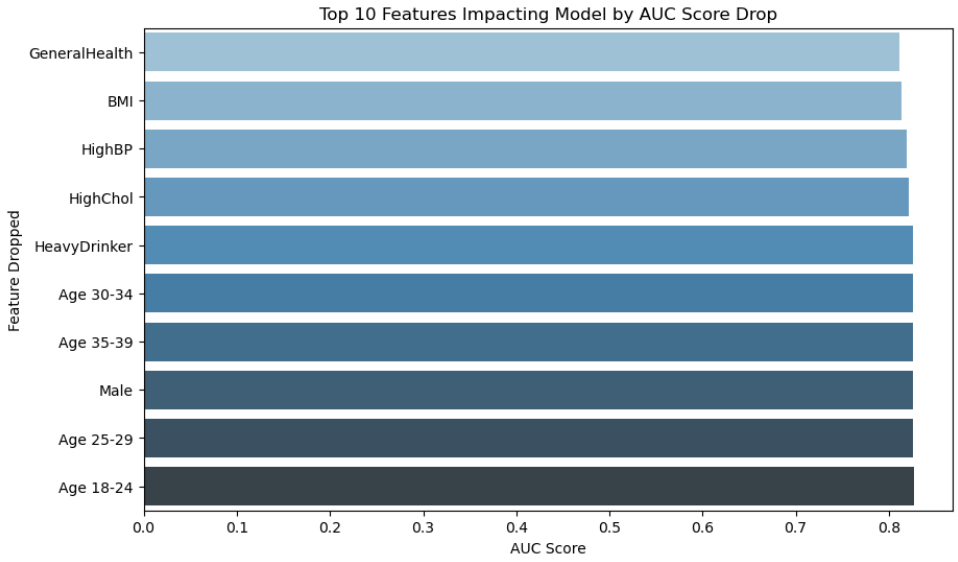


Figure 2: Top 10 Features Impacting Logistic Regression Model by AUC Score Drop

The Logistic Regression model's relatively high AUC suggests it is capable of distinguishing between patients with and without diabetes effectively. However, the moderate MCC indicates room for improvement in model precision and recall. The importance of the top predictors aligns with medical understanding that general health status, body mass index, and blood pressure as key indicators of diabetes risk.

2 Support Vector Machine (SVM)

Using the Support Vector Machine (SVM) model with the `LinearSVC` class from `sklearn`, we conducted hyperparameter tuning to find the optimal value of the regularization parameter C which controls the model's tolerance for misclassification. The optimal C value, determined to be 10, was based on maximizing the area under the receiver operating characteristic (ROC) curve (AUC score) through cross-validation. To address the imbalanced nature of the dataset, we computed class weights using the `compute_class_weight` function from `sklearn.utils.class_weight`. This method assigns a higher weight to the underrepresented class, thereby increasing the penalty for misclassifying the minority class during the training process. As observed from Table 2, the similarity between the training and test accuracy scores indicates the model's good generalizability to unseen data. The moderate recall score is achieved at the cost of lower precision, however, since this prediction is concerned with the medical field, being able to flag diabetics (recall) and start early medical intervention takes priority over precision. The recall without taking class weight into account was 0.07; therefore, addressing the imbalance was necessary for reasonable classification.

Model	**AUC**	**Training Acc.**	**Test Acc.**	**MCC**	**Precision (Diab.)**	**Recall (Diab.)**
SVM	0.8275	0.7272	0.7265	0.361	0.31	0.78

Table 2: Support Vector Machine Model Performance Metrics

The model, retrained with the best C value ($C = 10$), achieved an AUC score of 0.823 and an MCC of 0.3614. Based on the AUC score and test accuracy, we can conclude the data is not linearly separable. Given the performance metrics discussed in Table 2 and the AUC in Figure 3, the model generalizes well to unseen data, takes the imbalance into account, and is able to flag diabetics relatively well.

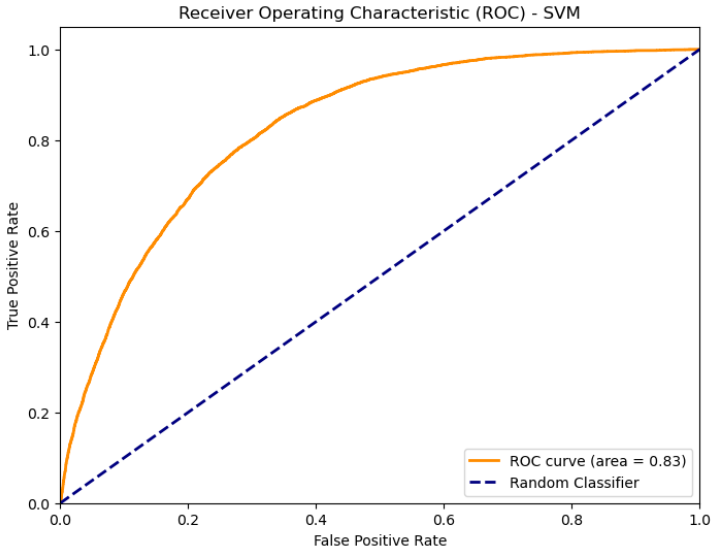


Figure 3: Receiver Operating Characteristic (ROC) – SVM

Similar to what was done with logistic regression, to find the best predictor, each feature was dropped iteratively, and the feature associated with the largest drop in the AUC score was considered the best predictor. Figure 4 below visualizes the top 10 best predictors.

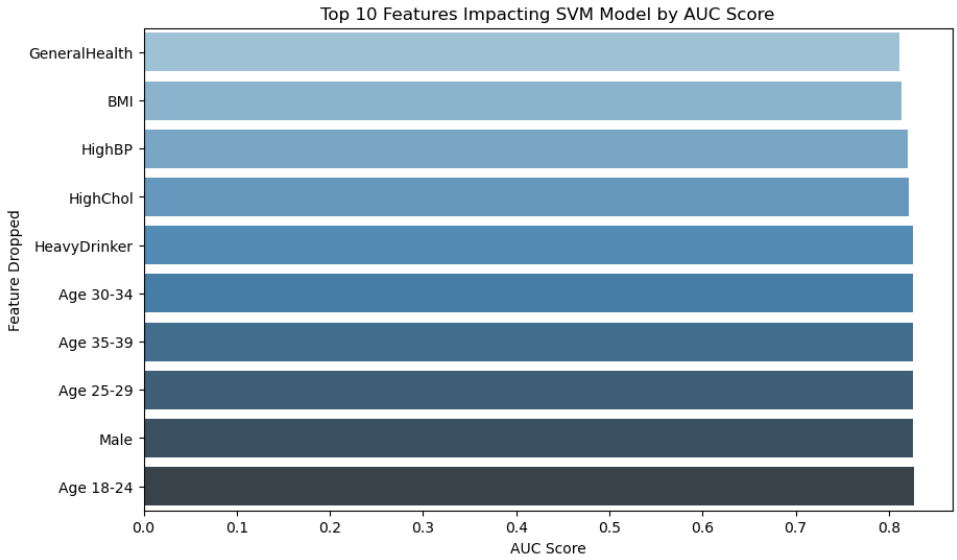


Figure 4: Top 10 Features Impacting SVM Model by AUC Score Drop

Similar to Logistic Regression, the best predictor remains General Health followed by BMI and High Blood pressure.

3 Decision Tree

The Decision Tree model was constructed using the `DecisionTreeClassifier` from `sklearn`, with splits determined by the Gini Criterion aimed at minimizing leaf impurity. Similar to what was done in Logistic Regression, the parameter `class_weight` was adjusted to "balanced" to address the imbalanced dataset. Further, the model's ability to classify diabetes was optimized by tuning the `max_depth` parameter, which controls the complexity of the decision tree. The criterion for selecting the best depth was based on the accuracy score.

A `GridSearchCV` approach was employed to identify the optimal depth within a range of 1 to 20. The search determined that a `max_depth` of 2 was optimal, indicating that depths beyond 2 lead to overfitting on the training set, thereby diminishing the model's generalizability to unseen data, such as the test set. For instance, setting the depth to 100 resulted in nearly perfect accuracy on the training set but significantly reduced the accuracy on the test set to approximately 0.5.

Table 3 below presents the performance metrics of the Decision Tree model in predicting the diabetes class. The close alignment between the training and test set accuracies suggests that the model generalizes well to unseen data. Observe that recall for the Decision Tree is lower than of the *generalized* linear models such as Logistic Regression and SVM and while precision remains the same. That means that the model has weaker predictive power when classifying diabetics.

Model	**AUC**	**Training Acc.**	**Test Acc.**	**MCC**	**Precision (Diab.)**	**Recall (Diab.)**
Decision Tree	0.7574	0.7632	0.7639	0.316	0.32	0.62

Table 3: Decision Tree Model Performance Metrics

Figure 5 below illustrates the Receiver Operating Characteristic (ROC) curve for the Decision Tree model, achieving an Area Under the Curve (AUC) score of 0.757. Without addressing the imbalance in the dataset, the AUC score was 0.813 which can be seen as more misclassification to increase recall. Nonetheless, the AUC score reflects the model's ability to discriminate between the positive (diabetic) and negative (non-diabetic) classes with a moderate degree of accuracy.

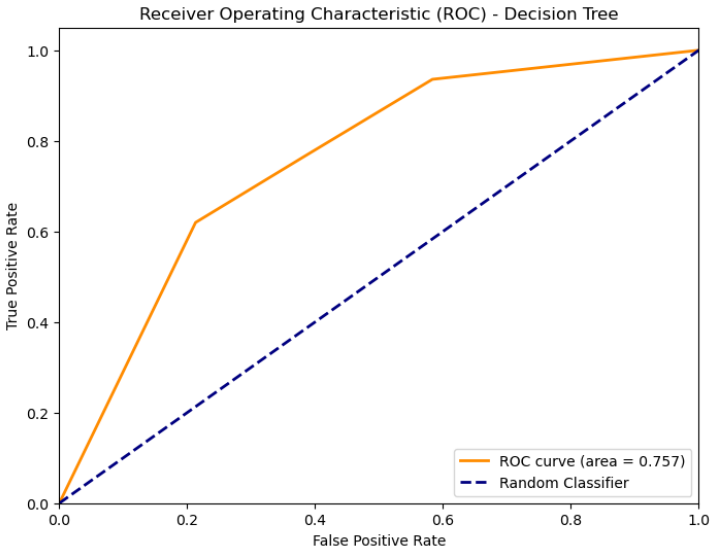


Figure 5: Receiver Operating Characteristic (ROC) – Decision Tree

Lastly, to identify the most significant predictor for diabetes within the Decision Tree model, we utilized the `feature_importances_` attribute provided by `sklearn`. This attribute ranks each feature by its influence on the model's decisions, allowing us to pinpoint which variables play the most critical role in predicting diabetes. The utilization of the `feature_importances_` attribute, as opposed to the feature dropping method applied in the logistic regression and SVM analyses, was primarily due to computational efficiency. Given the extensive number of dummy variables generated during the pre-processing phase, as detailed earlier, employing the feature dropping method would have significantly increased computational time and complexity

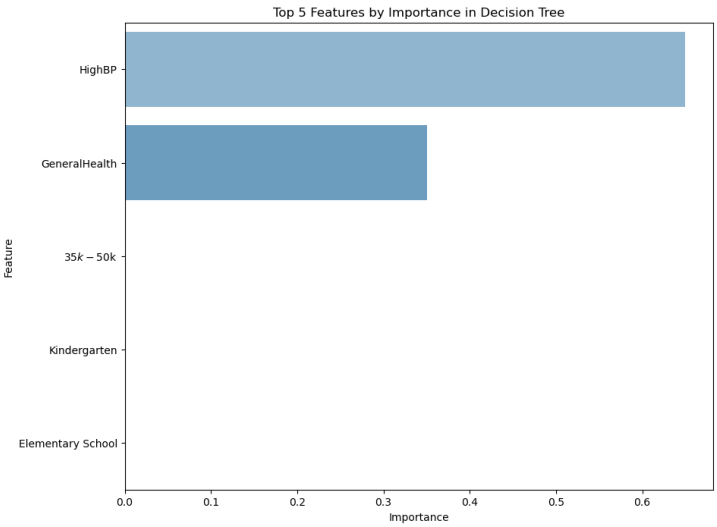


Figure 6: Top 5 Features by Importance in Decision Tree

Figure 6 above ranks the top 5 most important features in the Decision Tree model. The best predictor is High Blood Pressure followed by General Health. Despite minor differences in ranking, it seems that there is a consensus between the Logistic Regression Model, SVM Model, and Decision Tree Model in the importance High Blood Pressure and General Health for classifying diabetes. Also, the reason why there are only two features of importance in this model is because it is only one decision tree with `max_depth = 2`. That is, the single decision tree is only looking at the two most important feature. Figure 7 illustrates the decision tree for better understanding.

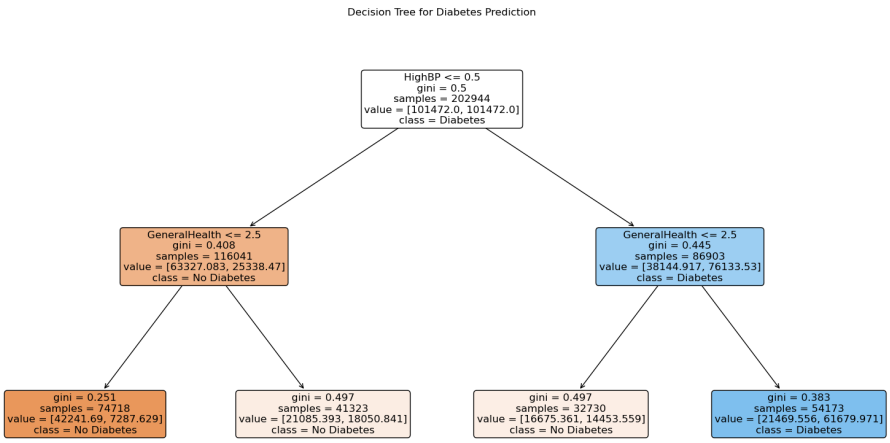


Figure 7: Decision Tree for Diabetes Prediction

4 Random Forest

For the Random Forest, a `RandomForestClassifier` from `sklearn` was implemented with specified parameters aimed at improving the model's performance. The `class_weight` parameter was set to "balanced" to address the imbalanced data, a moderate number of estimators was chosen (`n_estimators = 500`) to balance between computational efficiency and model performance, and `max_depth` of 10 was specified to prevent the trees from becoming to deep and overfitting on the training data.

Table 4 below presents the performance metrics of the Random Forest model in predicting the diabetes class. The similarity in the training and test set accuracy implies good generalizability to unseen date. Moreover, similar to the previous models, there is a trade-off between precision and recall – the increased recall in this model is achieved by increasing the false positive rate.

Model	**AUC**	**Training Acc.**	**Test Acc.**	**MCC**	**Precision (Diab.)**	**Recall (Diab.)**
Random Forest	0.8255	0.7564	0.7531	0.367	0.33	0.74

Table 4: Random Forest Model Performance Metrics

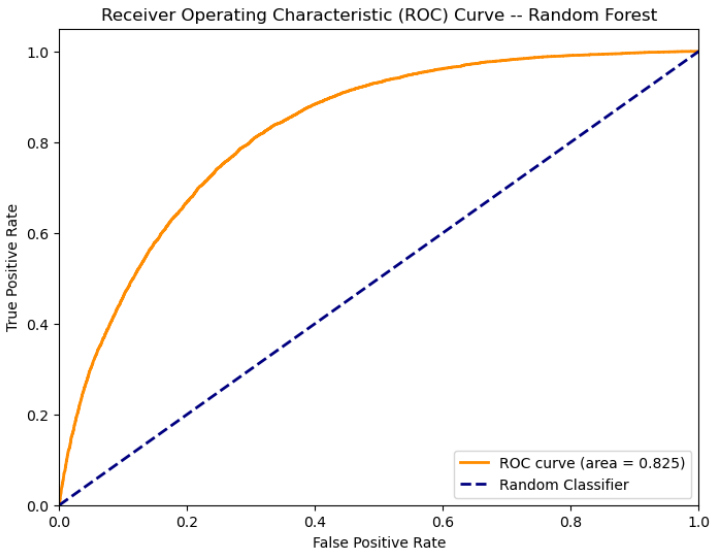


Figure 8: Receiver Operating Characteristic (ROC) – Random Forest

Figure 8 above illustrates the AUC score of 0.825 of the Random Forest model. Given the metrics,

including the AUC score and MCC, the model preforms well in classifying diabetes.

Lastly, similar to Decision Trees, feature importance was assessed by using `feature_importances_` attribute provided by `sklearn`. From Figure 9 below, we can see that General Health is the best predictor (the same for Logistic Regression and SVM) followed by High Blood Pressure and and BMI.

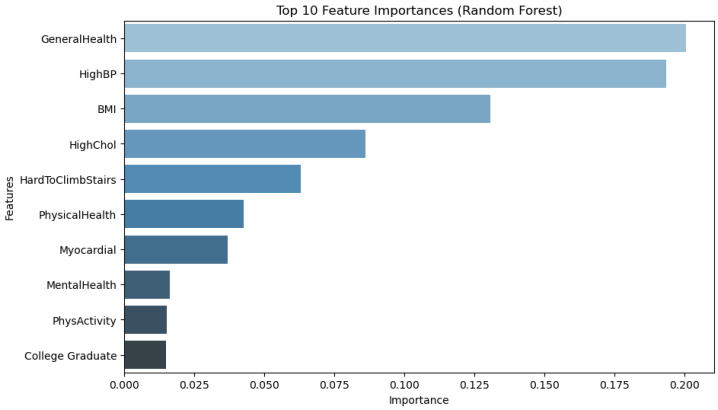


Figure 9: Top 5 Features by Importance in Random Forest

5 AdaBoost

The AdaBoost model was implemented using the `AdaBoostClassifier` from `sklearn`, with specified parameters to optimize performance. Similar to all previous models, in response to the dataset's imbalance, the `class_weight` parameter was adjusted to "balanced". This adjustment ensures that the model pays more attention to the minority class during training. A `max_depth` of 4 was selected for the base estimator to provide sufficient model complexity without risking overfitting. Furthermore, the learning rate was set to 1 to control the contribution of each classifier, and the number of estimators was set to a modest 200. The chosen parameters aim to enhance the AdaBoost model performance.

Model	**AUC**	**Training Acc.**	**Test Acc.**	**MCC**	**Precision (Diab.)**	**Recall (Diab.)**
AdaBoost	0.7919	0.7543	0.7558	0.345	0.32	0.69

Table 5: AdaBoost Model Performance Metrics

Table 5 shows the model's performance metrics. The similarity between the training and test accuracy imply good generalizability of the model to unseen data. On the other hand, the moderate recall is achieved at the cost of lower precision. As explained in previous sections, since the issue at hand is classifying diabetes (a medical concern), a higher priority has been placed on recall. The moderate AUC and MCC indicate reasonable ability to distinguish between classes. Figure 10 shows the AUC score.

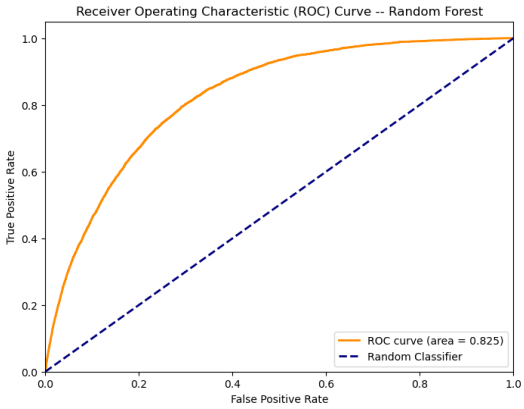


Figure 10: Receiver Operating Characteristic (ROC) – AdaBoost

Unlike a single decision tree or random forests, the `feature_importance_` attribute cannot be straightforwardly applied to the AdaBoost model. Therefore, to find the best predictor, multiple models were built in which only one feature was dropped, recalculated the AUC, and the variable associated with the largest drop in AUC was considered as the best predictor (same method used with Logistic Regression and SVM). Figure 11 shows the top 10 predictors for the AdaBoost model.

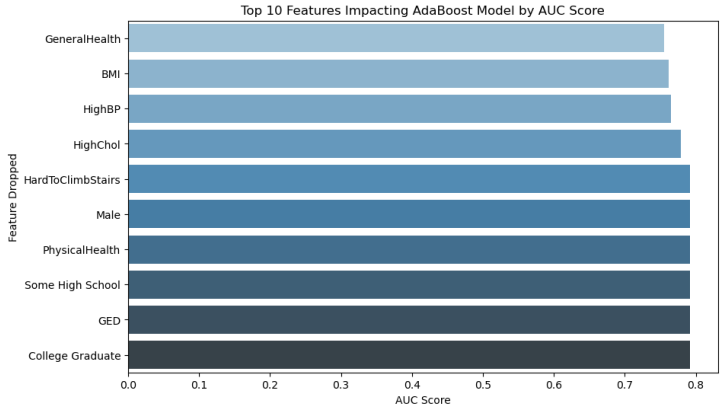


Figure 11: Top 10 Features Impacting AdaBoost Model by AUC Score Drop

The best predictor from the AdaBoost is General Health followed by BMI then High Blood Pressure. All models, with the exception of the decision tree imply that General Health is best feature for classifying diabetes.

Models Comparison

To assess what the best model is, a definition for good performance must be defined. Since the outcome variable, diabetes, is a medical concern, it is of utmost priority to flag diabetics as such to start early intervention. That is, by our standard, we would allow some false positives (lower precision) to increase recall. Therefore, a priority is placed on recall. Further, the AUC provides a holistic overview of the models' performances while the MCC indicates how well the model provides given that the dataset is imbalanced. Tables 6 and 7 summarize the performance of all five models.

Model	**AUC**	**Training Acc.**	**Test Acc.**	**MCC**	**Precision (Diab.)**	**Recall (Diab.)**
Logistic	0.8287	0.7311	0.7303	0.363	0.31	0.78
SVM	0.8275	0.7272	0.7265	0.361	0.31	0.78
Decision Tree	0.7574	0.7632	0.7639	0.316	0.32	0.62
Random Forest	0.8255	0.7560	0.7529	0.367	0.33	0.74
AdaBoost	0.7919	0.7543	0.7558	0.345	0.32	0.69

Table 6: Models Performance Metrics

Model	**Best Predictor**	**2nd Best Predictor**	**3rd Best Predictor**
Logistic	GeneralHealth	BMI	HighBP
SVM	GeneralHealth	BMI	HighBP
Decision Tree	HighBP	GeneralHealth	None (`max_depth = 2`)
Random Forest	GeneralHealth	HighBP	BMI
AdaBoost	GeneralHealth	BMI	HighBP

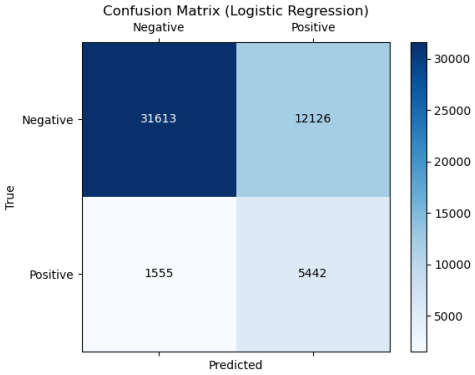
Table 7: Top Predictors for Each Model

Out of the two *generalized* linear models, Logistic Regression preforms slightly better on all metrics than SVM. As for tree-based models, the Random Forest model preforms the best based on the metrics while the single Decision Tree preforms the worst. From table 7, observe that all models list the same top three best predictors which supports the models' predictive powers. Both Logistic Regression and

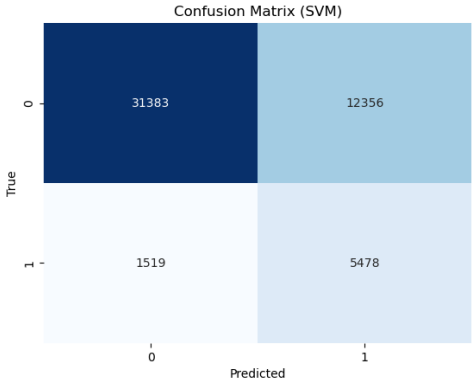
Random Forest models perform similarly across all metrics and predictors, however, an advantage of the Random Forest model is that it can capture non-linear relationships better than Logistic Regression. Therefore, we argue that the best model is Random Forest.

Appendix

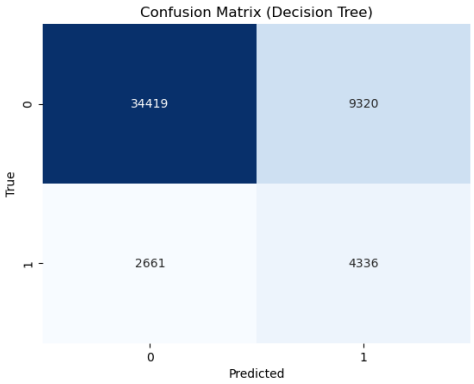
Confusion Matrices



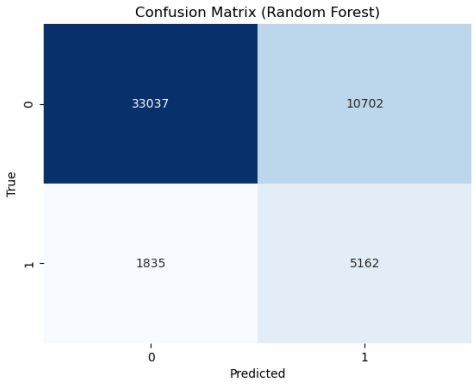
(a) Logistic Regression



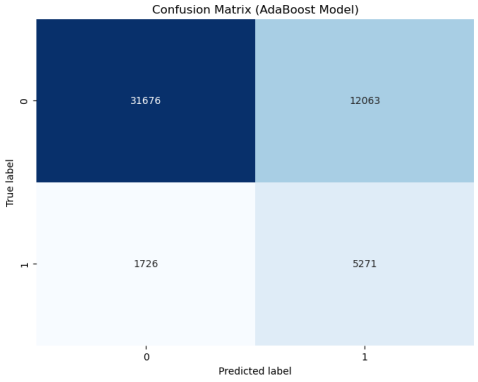
(b) SVM



(c) Decision Tree



(d) Random Forest



(e) AdaBoost

Figure 12: Confusion matrices for different models