Diabetes Prediction

Abstract

This study aimed to predict diabetes outcomes using data mining techniques and the Behavioral Risk Factor Surveillance System (BRFSS) 2014 dataset from Kaggle. A range of machine learning models was explored, including logistic regression, decision trees, gradient boosting, bagging classifiers, and deep neural networks (DNNs). Model performance was evaluated using F1-scores, precision, and recall. After further analysis, the gradient boosting model was identified as the best-performing model due to having the highest weighted F1-score.

Introduction

Diabetes is a chronic disease with significant health and economic impacts. Predicting the onset of diabetes, particularly type 2 diabetes, is vital for early intervention. Type 2 diabetes is influenced by lifestyle and demographic factors such as obesity, physical inactivity, and age. This project used survey data from BRFSS to build predictive models that identify at-risk individuals and explored the relationships between various risk factors and diabetes outcomes.

Methodology

Before training models, it was important to explore the dataset using data visualization techniques. This helped understand the relationships between variables and detect potential issues such as class imbalance, outliers, and correlations. Correlation analysis was conducted to identify redundant features that could introduce model bias. A correlation heatmap was generated using Pearson's correlation coefficient. BMI (Body Mass Index) is a key health indicator, but the distribution in the dataset was right skewed with extreme values (outliers). BMI was scaled using Z score normalization. Ordinal variables, such as mental and physical health, were binned into three categories to decrease computational workload.

Nearly 85% of the dataset was of responses from people not having diabetes, approximately 14% with diabetes, and 1.8% with prediabetes. The dataset was then split into 70% training and 30% testing for evaluation. To mitigate class imbalance effects, the training set was sampled using random splitting, under-sampling, and a hybrid approach combining both oversampling (SMOTE) of prediabetes and under-sampling of no diabetes and diabetes. The testing data remained the same from the original train/test split.

The models will be evaluated using F1-scores, which represent the harmonic mean of precision and recall. This metric effectively balances Type I (false positive) and Type II (false negative) errors, ensuring a trade-off between identifying positive cases and minimizing false alarms.

Logistic Model:

Variance inflation factors for all predictor variables were evaluated and the highest values were iteratively removed until all values were less than ten in preparation for the logistic regression model. The logistic model used was the multinomial logistic regression model, a multivariate logistic regression model with the Stats model library. The SoftMax function was applied to the probability-based predictions to get the predicted class. The logistic regression model was fitted and variables with high p-values, greater than .05, were iteratively removed from the highest until all p-values were lower than .05. The number of variables left in the model was eight, not including the constant, which is not overly complex for the number of observations it was trained and tested on. The model's coefficients are negative for the constant, heavy alcohol consumption, Income, and number of days exercised in a month. The variables with positive coefficients are high blood pressure(binary), high cholesterol (binary), general health, age, and standardized BMI. The logistic regression model performed well when trained on the original training set but suffered from lower validation metrics (accuracy, F1-score, and recall) when trained on the under-sampled and SMOTE-enhanced datasets.

Decision Tree

Each training set was optimized using GridSearchCV, with hyperparameters tuned based on the macro F1-score. The hyperparameters used were a criterion, max depth, min samples per leaf, and minimum sample split. Class weight balance was used as a hyperparameter for these models which can handle class imbalance. All variables in the dataset were used in the models. All the training sets were used, and the results were less-than-ideal models. The best decision tree was trained on the under-sampled smote hybrid training data with an F1-score of .8 for 'No Diabetes' and .43 for 'Diabetes'.

Gradient boosting

For each training set, the gradient boosting model was optimized using Optuna, with hyperparameters tuned to maximize accuracy. The hyperparameters used were the number of estimators, learning rate, max depth, and subsample. All the variables were used in training the model. The resulting model of the original training set was the best model with an F1-score of .91 for 'No Diabetes' and .28 for 'Diabetes'.

Random Forest

Randomized Search Cross-Validation was used to optimize hyperparameters with the macro F1-score as the evaluation metric. The hyperparameters tuned were the number of estimators, minimum sample leaf, minimum sample split, maximum depth, and bootstrapping. All of the variables from the dataset were used. Based on the weighted f1-score the model trained with the original dataset performed the best and even predicted a few prediabetes observations correctly without suffering a significant loss of accuracy. The best model was trained on the original data and had an F1-score of .8 for 'No Diabetes', .01 for ‘Prediabetes’, and .43 for 'Diabetes',

DNN:

Two types of models were created a deep neural network and a two-layer neural network with hyperparameter tuning. The TensorFlow library was used to create neural network models. The deep neural network consisted of a 64-unit dense layer, followed by a dropout rate of 0.1, another 64-unit dense layer with a 0.1 dropout, and a 32-unit dense layer, all preceding the final SoftMax output layer. Hyperparameter tuning was conducted using the Keras Tuner, optimizing the model for the macro F1 score for the number of neurons and learning rate. The under-sampled hyperparameter-tuned model performed the best with an increased f1-score in diabetes predictions. After hyperparameter tuning, the DNN achieved an F1-score of .91 for 'No Diabetes' and .28 for 'Diabetes', matching the gradient boosting model's performance.

Overfitting:

Cross-validation was done with five folds with the best testing macro F1 scores models. Cross-validation results demonstrated that the models were robust, with mean accuracies closely matching test accuracies and exhibiting low variance. To reduce overfitting, dropout layers (0.1) were added to the neural network, and L1/L2 regularization was tested.

Results

The gradient-boosted model and the deep neural network had nearly the same evaluation metrics on the testing set. With f1-scores of .92 for no diabetes, 0 for prediabetes, and .28 for diabetes on the testing data.

Further analysis

To improve performance, the SoftMax approach in multinomial logistic regression could be modified to prioritize recall for underrepresented classes. A one-vs-all logistic regression model could allow different predictors for 'Diabetes' and 'Prediabetes.' Additionally, ensemble methods such as stacking could improve overall classification by first distinguishing 'No Diabetes' vs. 'Diabetes/Prediabetes' and then refining the prediction between the latter two categories

Sourcing new variables, which are not highly correlated with the variables. These new variables could be genetic marker inferences such as asking if parents have diabetes. Various dummy variables could be used to identify class patterns but could lead to high complexity, which is why this method was avoided in the analysis. A better approach would be to try methods to segment the categorical variables such as k-mode clustering to create a new variable similar to principal component analysis. Additionally, Stacking or ensemble models with the pre-existing or new models could be created. Stacking or ensemble models could be created, but cross-validation would be crucial for this. Other sampling techniques could be tried such as adaptive synthetic sampling.

Conclusion

The severe class imbalance (84.2% No Diabetes, 1.8% Prediabetes, 13.9% Diabetes) made it difficult for models to detect the minority class (Prediabetes). To address this, different resampling techniques were tested which included class weighting within models, under-sampling of the majority class, and under-sampling combined with oversampling. Despite these adjustments, prediabetes recall remained low across models, indicating the need for additional feature engineering, or sourcing additional variables to better identify prediabetes. Gradient boosting and deep neural networks achieved the highest F1 scores for 'Diabetes,' while logistic regression remained the most interpretable model. However, all models struggled to detect 'Prediabetes,' indicating a need for improved feature selection and resampling techniques. Future work includes improving sensitivity for underrepresented outcomes and exploring additional features for better generalization.

Index

DNN evaluation metrics on testing set:

```
Accuracy: 0.8508

Confusion Matrix:

 [[62754     0  1350]

 [ 1216     0   132]
 [ 8658     0  1994]]
Classification Report:
              precision    recall  f1-score   support

         0.0     0.8640    0.9789    0.9179     64104
         1.0     0.0000    0.0000    0.0000      1348
         2.0     0.5736    0.1872    0.2823     10652

    accuracy                         0.8508     76104
   macro avg     0.4792    0.3887    0.4001     76104
weighted avg     0.8081    0.8508    0.8127     76104
```

Gradient Boosting evaluation metrics on testing set:

```
Accuracy: 0.8508
Confusion Matrix:
 [[62715     3  1386]
 [ 1215     1   132]
 [ 8621     1  2030]]
Classification Report:
              precision    recall  f1-score   support

         0.0     0.8644    0.9783    0.9179     64104
         1.0     0.2000    0.0007    0.0015      1348
         2.0     0.5722    0.1906    0.2859     10652

    accuracy                         0.8508     76104
   macro avg     0.5455    0.3899    0.4018     76104
weighted avg     0.8117    0.8508    0.8132     76104
```

Logistic Regression:

```
Confusion Matrix:
 [[62652     0  1452]
 [ 1242     0   106]
 [ 8935     0  1717]]
Classification Report:
              precision    recall  f1-score   support

         0.0       0.86      0.98      0.92     64104
         1.0       0.00      0.00      0.00      1348
         2.0       0.52      0.16      0.25     10652

    accuracy                           0.85     76104
   macro avg       0.46      0.38      0.39     76104
weighted avg       0.80      0.85      0.81     76104
```