**Determining Diabetes Diagnosis Through Machine Learning**

Omar Abdel-Kerem

Department of Software Engineering

Data Science Graduate Program

University of St. Thomas, St. Paul, MN

In Collaboration with:

Sarah Preese

Brandon Osvold

Theon Masters

Michael Tiede

**Description of the problem and predicted answer**

According to the Centers for Disease Control and Prevention, diabetes affects 37.3 million Americans, which is about 11.3% of the US population. Of this 37.3 million, the CDC estimates that about 8.5 million are undiagnosed, meaning they have diabetes but do not know. In addition, there are about 96 million Americans with prediabetes, which is more than one-third of the total US population. These numbers are staggering, which is why diabetes is the focus of this machine learning project. It is important to try to determine what the biggest health predictors are in diagnosing diabetes, so health care providers can learn early whether a patient may develop diabetes in the future. This project attempts to answer the question: what are the important predictors for diagnosing diabetes? Are some health factors more important than others?

**Description of data source**

The data set was obtained from Kaggle. The author is Rahul. Below is the link to the website where the dataset can be downloaded as a csv file.

Link: [*https://www.kaggle.com/datasets/rahulsah06/machine-learning-for-diabetes-with-*](https://www.kaggle.com/datasets/rahulsah06/machine-learning-for-diabetes-with-python) *python*

**Description of predictors**

This dataset consists of nine total columns: eight predictors and one target. There are 768 total records. The predictors are Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, and Age. The target column is Outcome. Each of these predictors and the outcome will be explained below.

The ‘Pregnancies’ predictor is a measure of the number of pregnancies a person has had to date. The ‘Glucose’ predictor is the plasma glucose concentration (mg/dL) during an oral glucose tolerance test. This test measures a patient’s tolerance to sugar over two hours. Less than 140 mg/dL is normal. 140-199 mg/dL indicates prediabetes. Greater than 199 mg/dL suggests the patient has diabetes. The ‘Blood Pressure’ predictor the measure of the patient’s diastolic blood pressure (mmHg). Less than 80mmHg is normal. More than 140mmHg indicates high blood pressure. The ‘Skin Thickness’ predictor is the measure of the patient’s skin fold thickness (mm) of their triceps.

The ‘Insulin’ predictor is a measure of the patient’s serum insulin (mu U/mL) measured in 2 hours. The ‘BMI’ predictor is the patient’s body mass index for weight in kg and heigh in m. The ‘Diabetes Pedigree Function’ is a function that scores the likelihood of a diabetes diagnosis based on family history. Lastly, the ‘Age’ predictor is the patient’s age in years. This dataset does not include children, only 21 years and older. The target column is the ‘Outcome,’ which means the patient either has diabetes (1) or does not have diabetes (0).

**General statistics of the dataset**

Prior to standardization, general statistics were calculated for each of the predictors. For ‘Pregnancies,’ it was found that the mean was 3.85 and the standard deviation was 3.37. The values ranged from 0 to 17. There were no missing values, so data cleaning was not needed. For ‘Glucose,’ the mean was calculated to be 121 with a standard deviation of 32. A higher standard deviation indicates that the values for this predictor are spread out. However, some of this “spread” may be due to zeros being present in the column. Although there were no missing values, these zeros were considered “missing” because a patient cannot have 0 mg/dL plasma glucose as they would be dead or close to dead. These zeros were replaced with the column mean during data cleaning.

For ‘Blood Pressure,’ the mean was calculated to be 69.1 with a standard deviation of 19.1. Again, this larger standard deviation indicates that the values for this predictor are spread out instead of being clustered close together. Although there are no null or missing values, it was found that this column contained 35 zeros. Like ‘Glucose,’ these zeros were considered “missing” because a patient cannot have 0 mmHg diastolic blood pressure without being dead or close to dead. These zeros were replaced with the column mean during data cleaning.

For ‘Skin Thickness,’ it was found that the mean was 20.5 and the standard deviation was 15.9. Like ‘Glucose’ and ‘Blood Pressure,’ there were many zeros present in this column, which does not make clinical sense, so the column mean was substituted. The ‘Insulin’ predictor had a column mean of 79.8 and a standard deviation of 115. This column had 374 zeros present, which could account for this very large standard deviation. As pointed out during the presentation, it is possible the health care professionals did not bother entering a value for ‘Insulin’ because there is a high negative correlation with ‘Glucose,’ therefore that column could be eliminated from the dataset without effecting the model. However, this was not considered at the time of the model building, so again, the zeros were substituted with the column mean.

For ‘BMI,’ it was found that the mean was 32 and the standard deviation was 7.88. Like with many other predictors in this dataset, there were quite a few zeros present in this column. Having a BMI of 0 is impossible, so the zeros were treated as missing values and the column mean was substituted. For ‘Diabetes Pedigree Function,’ the means was found to be 0.47 with a standard deviation of 0.33. This low standard deviation demonstrates the values were clustered closer together ranging from 0.08 to 2.42. There were no missing values, so data cleaning was not needed. Lastly for ‘Age,’ it was found that the mean was 33.2 and the standard deviation was 11.8. The values ranged from 21 to 81. There were no missing values, so data cleaning was not needed.

**Tools and methods utilized**

Throughout this project, various tools and methods were used including data cleaning, Z-score standardization, many Python libraries, regularization, and Support Vector Machine RBF. Although there were no true missing or null values, as mentioned in the earlier section, many fields contained a zero where it did not make clinical sense. To clean the data, these values were replaced by the column mean.

Once the data was cleaned, Z-score standardization was applied to all predictor columns. This is important to ensure the columns with larger numbers or ranges do not dominate or skew the model. For example, Diabetes Pedigree Function has values that range from 0.08 to 2.42 whereas Insulin values were seen as high as 846. Without standardization, the model would skew toward Insulin being more important than Diabetes Pedigree Function because the values are so much larger. This would produce an inaccurate model and therefore the predictions would likely be wrong. Standardizing the data puts all the predictors at even playing field despite original values or units.

To successfully build an SVM model for this dataset, numerous Python libraries, NumPy, Pandas, Sklearn, and Matplotlib, were imported into Google Collab, the coding environment chosen for this project. NumPy was needed to work with arrays. Pandas was used for data cleaning and manipulation. Sklearn was crucial for building the SVM model used to predict the outcome, a diabetes diagnosis. Lastly, Matplotlib was used for plotting graphics, such as an ROC curve.

In order to minimize the error (hinge loss) and control the model complexity to ensure the model is not overfit, regularization was performed by adjusting the box constraint (C) and kernel scale (gamma). Various box constraints were tested ranging from 0.001 to 5.0 and numerous gamma values were tested ranging from 0.05 to 3.0. It was found that the optimal model occurred with a box constraint value of 1.0 and a gamma value of 0.8. Through regularization, it was found that the important predictors for determining a diabetes diagnosis were Glucose, Diabetes Pedigree Function, and BMI.

An SVM RBF model was built and trained with the dataset to classify the records as either 0 (no diabetes) or 1 (diabetes). SVM was the chosen machine learning model because the risk of error is reduced by maximizing the margin between the two classes. Originally, the entire dataset was used for training and testing. This produced a model with high precision, recall, and f1 score for both classes. However, feedback during the presentation suggested the dataset should be split for training and testing to get a more authentic real-world model. The model quality metrics after the split were not as high as the previous report, but this model is more realistic because it did not previously see the records during training. For class 0, it was found that the precision was 0.75, recall was 0.89, and the f1 score was 0.81. For class 1, it was found that the precision was 0.68, recall was 0.46, and the f1 score was 0.55. A new ROC was also generated with these new changes.

**Summary of Presentation**

A more detailed explanation of the project was given in the above sections, so this will be a brief overview of what was discussed during the class presentation. The presentation started with an introduction of diabetes including the shocking statistics mentioned in the “Description of the problem and predicted answer” section of this paper. Then, an overview of the dataset was provided, including a description of the predictors and target column. Next, an explanation for why this dataset was chosen was given, which was to provide insight for healthcare professionals about the important factors in determining a diabetes diagnosis. This knowledge will hopefully lead to the promotion of overall health and wellbeing of individuals that are susceptible to a future diabetes diagnosis.

Then, a summary of the various tools and methods used throughout the project were explained. Details of these tools and methods can be found under the “Tools and Methods Utilized” section of this paper. The model quality was then discussed using an image of the generated classification report and ROC curve from the code. However, as discussed previously, changes were made to this part of the code because the dataset was not originally split into a training and testing set. Lastly, an analysis of the results from the model was explained. This included pitfalls, a crucial one being that over 10% of the total fields in the dataset were substituted with column mean. In addition, the dataset was unbalanced with almost double the records with an outcome of 0. After the presentation, feedback was provided (mentioned earlier in this paper), some of which we incorporated into the final code.