DEPT: SOFTWARE ENGINEERING

FUNDAMENTAL OF MACHINE LEARNING

NAME: FUAD SEID

ID: DBU1401283

SUBMITTED TO TEACHER DERBEW

SUBMISSION DATE: 02/05/2017

Diabetes Prediction Model Report

1. Problem Definition, Data Source, and Description

Problem Definition: Diabetes is one of the most prevalent chronic diseases globally, and early detection plays a crucial role in preventing complications. The objective of this project is to develop a machine learning model capable of predicting whether an individual is at risk of developing diabetes based on a set of health-related features. Early intervention and management are vital for diabetic patients, and this model could serve as a valuable tool in healthcare systems to identify at-risk individuals and assist in timely medical intervention.

I initiated this project due to my personal experience with my mother, a diabetic patient, and have come to recognize the importance of utilizing technology and data-driven approaches for improving health management. With the rising global prevalence of diabetes, this issue is of great personal significance to me. Through this project, I aim to help individuals detect diabetes at an early stage, improving their quality of life and preventing long-term complications.

Data Source: The dataset used for this project is the Pima Indians Diabetes Database, a widely recognized resource in healthcare machine learning tasks. It is available on GitHub at: https://github.com/praveenbharti1/diabetes_prediction/blob/main/diabetes.csv. This dataset consists of medical diagnostic data gathered from female patients of Pima Indian heritage, a population with a relatively high rate of diabetes. It includes various health-related features that can be used to predict diabetes, such as glucose levels, BMI, age, and more.

Dataset Description: The dataset contains 768 instances (observations) and 9 attributes (features). Below is a breakdown of the features:

- **Pregnancies:** Number of pregnancies the patient has had.
- **Glucose:** Plasma glucose concentration after a 2-hour oral glucose tolerance test.
- **BloodPressure:** Diastolic blood pressure measured in mm Hg.
- **SkinThickness:** Triceps skinfold thickness in mm.
- **Insulin:** 2-hour serum insulin concentration in µU/mL.
- **BMI:** Body Mass Index (weight-to-height ratio, used as an indicator of obesity).
- **DiabetesPedigreeFunction:** A function that scores the likelihood of diabetes based on family history.
- **Age:** The age of the individual.
- **Outcome:** The target variable, where 1 indicates that the patient has diabetes, and 0 indicates they do not.

The data is primarily numerical, with the Outcome column being binary. The dataset is relatively clean but contains some class imbalance, with fewer diabetic cases than non-diabetic cases.

2. Exploratory Data Analysis (EDA) Findings and Visualizations

EDA Overview: The purpose of exploratory data analysis (EDA) is to understand the underlying structure of the data, detect outliers or anomalies, identify trends, and select important features for the predictive model.

- **Missing Values:** No missing values were found, simplifying the preprocessing process.
- **Feature Distribution:** Histograms and box plots revealed skewed distributions in features such as Glucose, BMI, and Age, indicating the need for scaling.
- **Class Imbalance:** More non-diabetic cases (Outcome = 0) than diabetic cases (Outcome = 1) were observed, which may impact model bias.
- **Correlations:** A heatmap demonstrated strong correlations between Glucose, BMI, and Age with the Outcome variable.
- **Outliers:** Outliers in SkinThickness and Insulin were detected and analyzed.

Visualizations Used: A total of X visualizations were utilized to analyze the dataset, including:

1. **Histogram Plots:** To examine the distribution of numerical features.
2. **Box Plots:** To detect outliers in features like SkinThickness and Insulin.
3. **Correlation Heatmap:** To identify relationships between features.
4. **Pair Plots:** To visualize feature interactions and distributions.
5. **Count Plot:** To display the class distribution of diabetic vs. non-diabetic cases.

3. Preprocessing Steps and Choices

Data Cleaning:

- **Scaling Features:** Standardization (zero-mean, unit-variance scaling) was applied to ensure uniform feature ranges.
- **Train/Test Split:** The dataset was split into 80% training and 20% test sets.
- **Handling Class Imbalance:** Techniques such as oversampling and F1-score evaluation were employed to address class imbalance.

4. Model Selection and Training Details

The project uses **K-Nearest Neighbors (KNN)** to predict diabetes:

- **KNN Classifier:** A non-parametric algorithm used for classification, where the class of a sample is determined by the majority class of its nearest neighbors. It is a simple yet effective model for classification tasks.
    - **K:** Represents the number of nearest neighbors considered.
    - **Distance Metric:** Euclidean distance is typically used, but it can be customized.

You can experiment with different values of K to identify the optimal model for this dataset.

5. Model Evaluation Metrics and Discussion

Several metrics were used to evaluate the model's performance:

- **Accuracy:** The proportion of correct predictions.
- **Precision:** The proportion of positive predictions that are actually correct.
- **Recall:** The proportion of actual positive cases that are correctly identified.
- **F1-score:** The harmonic mean of precision and recall.
- **Confusion Matrix:** A table used to describe the performance of a classification model.

6. Interpretation of Results

- **Feature Importance:** Glucose, BMI, and Age were identified as the top three most influential factors in predicting diabetes. Higher glucose levels and BMI showed a strong correlation with the likelihood of having diabetes.
- **Model Insights:** The K-Nearest Neighbors (KNN) model was used for prediction. While KNN performed well, its performance depends on the choice of K and the distance metric. It is sensitive to imbalanced data and may require feature scaling for optimal results.

7. Deployment Details and Instructions

Cloud Deployment: The model has been deployed on Render Cloud, making it accessible via an API at: https://machine-learning-1-yl7v.onrender.com/docs

API Usage: You can send a POST request to the API with patient data to receive a diabetes prediction. The API processes the input and returns a prediction based on the trained K-Nearest Neighbors (KNN) model.

Frontend Web Application: To enhance user interaction, I developed a React-based web application that provides a user-friendly interface for diabetes prediction. This allows users to input their medical data easily and receive real-time predictions without interacting directly with the API.

The website can be accessed here: https://ml-t4w1.vercel.app/

Using Postman:

1. Open Postman.
2. Select POST as the request type.
3. Enter the URL: https://machine-learning-1-yl7v.onrender.com/predict
4. Navigate to the Body tab and select raw > JSON format.
5. Enter the following JSON payload:

```
{
    "Pregnancies": 2,
    "Glucose": 120,
    "BloodPressure": 70,
```

```
  "SkinThickness": 20,
  "Insulin": 85,
  "BMI": 25.3,
  "DiabetesPedigreeFunction": 0.5,
  "Age": 30
}
```

6. Click Send, and you will receive a JSON response with the predicted outcome.

Using cURL:

```
curl -X POST "https://machine-learning-1-yl7v.onrender.com/predict" \
-H "Content-Type: application/json" \
-d '{"Pregnancies": 2, "Glucose": 120, "BloodPressure": 70, "SkinThickness": 20,
"Insulin": 85, "BMI": 25.3, "DiabetesPedigreeFunction": 0.5, "Age": 30}'
```

8. Potential Limitations and Future Improvements

- **Class Imbalance:** Further exploration of SMOTE (Synthetic Minority Over-sampling Technique) for synthetic minority data generation to balance diabetic and non-diabetic cases.
- **Feature Engineering:** Enhancing predictions by incorporating lifestyle factors such as physical activity and diet.
- **Model Complexity:** Exploring interpretable AI techniques to improve model explainability.
- **Model Generalization:** Testing on different datasets to enhance reliability across diverse populations.
- **Deployment Performance:** As the model is hosted on the free version of Render Cloud, the initial data fetching may be slow. The first request may show a fetching error, but after pressing "predict" 4-5 times, it will eventually display results. Once loaded, it operates smoothly and quickly. I attempted to troubleshoot this issue through tutorials and expert consultations, but the free version of Render inherently suffers from slower speeds. The only way to resolve this is by upgrading to a paid plan, so I have left it as is.

This report provides a comprehensive overview of the diabetes prediction model, from problem definition to deployment, offering insights into its strengths, limitations, and future enhancements.