Group number 832 : Predictive Modeling of Diabetes Using Machine Learning Techniques

Your submissions:

- Group number_Report.pdf
- Group number_Codes.ipynb (with necessary comments)
- Group number_Codes.html (converted from the ipynb file above)
- Group number_Slide.pdf (your presentation slide)

Notes

- No extension to the deadline
- Each team can only submit one copy by a single member, just list all of your members in the report
- use RED font for the parts that you revised according to the feedbacks in your presentation

First Name	Last Name	Email address
Vijay Krishna	Konatham	vkonatham@hawk.iit.edu
Sree Datta	Pusala	spusala@hawk.iit.edu
Subrahmanya Tejaswi	Vadlamani	svadlamani2@hawk.iit.edu
Durga Sai Sri	Bommi	dbommi@hawk.iit.edu

Table of Contents

1. Introduction

Diabetes is a chronic and prevalent condition that affects millions of people worldwide and is defined by abnormal blood sugar balance. If left untreated, it might lead to major health complications. Early detection and prediction are critical for improved patient outcomes. Because of its ability to evaluate enormous amounts of patient data and create exact predictions, machine learning techniques, a subset of artificial intelligence, have grown in popularity in the healthcare business. These models can assist in identifying those who are at risk of developing diabetes, allowing for preventative healthcare regimens and personalized therapies.

The fundamental incentive for machine learning-based predictive modeling for diabetes is the possibility of early detection and intervention. Predicting diabetes onset, even in the prediabetic stage, provides a critical window of opportunity for early treatment. This enables medical practitioners and patients to begin targeted drug therapy and lifestyle adjustments, reducing the risk of complications and increasing the quality of life for those at risk or currently suffering from the condition. Personalized healthcare is being driven forward by customized prediction models that consider each patient's unique medical history, genetics, lifestyle, and surroundings. These solutions have the potential to considerably improve therapy efficacy and patient outcomes. The huge healthcare data repository provides a once-in-a-lifetime chance to investigate the complicated network of variables that lead to diabetes onset.

2. Data

It appears that we have access to some fascinating health data! The Behavioral Risk Factor Surveillance System (BRFSS) is a treasure trove of information for studying health-related behaviors and illnesses. Its annual reach of approximately 400,000 Americans since 1984 lends it exceptional depth.
Let's dive into the specifics:
Data Source: The data comes from the BRFSS survey conducted by the CDC in 2015. You can find the dataset on Kaggle via this link.
https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset
Data Size: The original dataset contains responses from 441,455 individuals. For your specific projects, you're working with three different subsets:
The clean dataset with 253,680 survey responses for the Diabetes_012 target variable.
Another clean dataset with 70,692 responses for the Diabetes_binary target variable (balanced 50-50 split).
And a larger set of 253,680 responses for the Diabetes_binary target variable (imbalanced).
Features: The datasets are rich, boasting 330 features in the original set. For your project, you're using 21 feature variables across all three datasets. These features range from direct participant responses to calculated variables based on individual inputs.
Target Variables: For the first dataset (Diabetes_012), you have a multi-class target variable with three classes: 0 (no diabetes or only during pregnancy), 1 (prediabetes), and 2 (diabetes).
The second dataset (Diabetes_binary 50-50 split) has a binary target variable (0 for no diabetes, 1 for prediabetes or diabetes) and is balanced.
The third dataset (Diabetes_binary imbalanced) also has a binary target variable but is not balanced, so there's some class imbalance to be mindful of.

3. Problems and Solutions

Multi-class Classification for Diabetes Types:

Problem: Develop a predictive model to classify individuals into three diabetes types (0 for no diabetes or only during pregnancy, 1 for prediabetes, and 2 for diabetes) using the Diabetes_012 dataset.

Solution: Make use of multi-class classification methods like Random Forest, KNN, Decision tree or Logistic regression.

Plan: Apply these algorithms to the Diabetes_012 dataset, cross-validate them, and fine-tune the hyperparameters. Model performance can be measured using metrics like accuracy, precision, recall, and F1-score.

Binary Classification for Diabetes Risk:

Problem: Create a predictive model to classify individuals as either having no diabetes (0) or having prediabetes/diabetes (1) using the balanced Diabetes_binary 50-50 split dataset.

Solution: Apply binary classification algorithms like Logistic Regression, Decision tree, or KNN.

Plan: Train models on the balanced Diabetes_binary 50-50 split dataset, employ techniques like Grid Search for hyperparameter tuning, and assess model performance. Use metrics like ROC-AUC, precision recall curve, and confusion matrix.

Addressing Class Imbalance in Binary Classification:

Problem: Tackle the challenge of class imbalance in the imbalanced Diabetes_binary dataset and build a predictive model.

Solution: Explore techniques such as oversampling (SMOTE), undersampling, or using advanced algorithms designed for imbalanced datasets (e.g., XGBoost, LightGBM).

Plan: Experiment with different imbalance handling methods on the imbalanced Diabetes_binary dataset. Evaluate and compare results to identify the most effective approach.

Feature Importance and Selection:

Problem: Identify and select the most influential features in predicting diabetes outcomes from the 21 available features.

Feature Importance and Selection: Solution: Employ techniques like feature importance from tree-based models, recursive feature elimination, or LASSO regression.

Plan: Implement feature importance methods on the 21 features of the datasets. Select a subset of the most relevant features and assess the impact on model performance.

Evaluation Metrics Selection:

Problem: Choose appropriate evaluation metrics for each classification task, considering the specific nuances of predicting diabetes types and risk.

Solution: Tailor metrics based on the specific goals of each classification task. For example, emphasize sensitivity and specificity for healthcare applications.

Plan: Evaluate models using task-specific metrics, ensuring a comprehensive understanding of their performance. Compare results with different metrics to provide a holistic assessment.

Handling Missing or Inconsistent Data:

Problem: Address challenges related to missing or inconsistent data within the features.

Solution: Employ imputation techniques such as mean, median, or advanced methods like K-nearest neighbors imputation.

Plan: Address missing or inconsistent data in the features, compare imputation methods, and analyze the impact on model performance. Ensure robustness by evaluating models with and without imputation.

4. KDD

4.1. Data Processing

The initial phase involves data preprocessing, encompassing tasks such as data cleaning, handling missing values, and scaling features. Feature engineering might be employed to extract meaningful insights from raw data, ensuring the datasets are ready for model training.

4.2. Data Mining Methods and Processes

Utilizing a range of machine learning algorithms for modeling purposes will include techniques like Random Forest, Decision trees, Logistic Regression, KNN, Naive Bayes and advanced algorithms tailored for imbalanced datasets. The model selection, training, validation, and hyperparameter tuning processes are detailed under this section.

5. Evaluations and Results

5.1. Evaluation Methods

Multi-class Classification for Diabetes Types:

Evaluation Strategy: Use Hold-out evaluation with 80-20 split to ensure robust performance assessment.

Metrics: Evaluate using accuracy, precision, recall, and F1-score for each class, consider using macro and micro-average for a comprehensive overview.

Binary Classification for Diabetes Risk: -

Evaluation Strategy: Use Hold-out evaluation with 50-50 split to ensure robust performance assessment.

Metrics: Use metrics like ROC-AUC, precision-recall curve, and confusion matrix. Sensitivity and specificity are vital in healthcare scenarios to understand the model's ability to identify true positives and negatives.

Addressing Class Imbalance in Binary Classification:

Evaluation Strategy: Use Hold-out evaluation with 80-20 split with a focus on strategies to handle class imbalance (e.g., SMOTE).

Metrics: Apart from standard metrics, pay attention to metrics suitable for imbalanced datasets, such as the F1-score, area under the precision-recall curve (AUC-PR), and specificity.

Feature Importance and Selection:

Evaluation Strategy: Utilize cross-validation to ensure robustness in feature selection.

Metrics: Assess the impact of feature selection on model performance using metrics like accuracy, precision, and recall. Additionally, observe changes in computational efficiency.

5.2. Results and Findings

Here's a summary of the evaluation metrics for different models you've tried:

For diabetes _ 012 Dataset:

Random Forest Classifier:

Accuracy: 84.12%

Decision Tree Classifier:

Accuracy: 76.73%

K-Nearest Neighbors (kNN):

Accuracy: 82.00%

Logistic Regression:

Accuracy: 84.83%

For Diabetes_binary Dataset:

Logistic Regression:

Accuracy: 73.99%

Decision Tree Classifier:

Accuracy: 65.32%

k-Nearest Neighbors (kNN):

Accuracy: 73.77%

For diabetes _ binary imbalance Dataset:

k-Nearest Neighbors (kNN):

Accuracy: 70%

Naive Bayes:

Accuracy: 72%

Decision Tree Classifier:

Accuracy: 79%

Logistic Regression:

Accuracy: 73%

Based on the accuracy and other evaluation metrics, the Random Forest Classifier for 1st dataset and Logistic Regression for 2nd dataset and decision tree for 3rd dataset models seem to perform relatively better across different datasets and classes. These models exhibit higher accuracy and better precision, recall, and F1-score compared to other models like kNN.

6. Conclusions and Future Work

6.1. Conclusions

The Random Forest Classifier for the first dataset, Logistic Regression for the second dataset, and Decision Tree for the third dataset models appear to perform comparatively better across different datasets and classes based on assessment metrics and accuracy. When compared to other models like kNN, these models show improved recall, F1-score, accuracy, and precision.

After implementing feature reduction for the decision tree model, there was a notable increase in its accuracy. This improvement is evident across different datasets and classes, with the decision tree outperforming other models such as kNN. The results suggest that the applied feature reduction strategy contributed positively to the decision tree's predictive capabilities, making it a preferred choice for the given datasets and classification tasks.

6.2. Limitations

Acknowledging the constraints and limitations encountered during the analysis, such as data quality issues, model complexities, or constraints imposed by the datasets.

6.3. Potential Improvements or Future Work

Suggesting avenues for further research and improvement, which might include exploring different algorithms, refining data preprocessing techniques, incorporating more recent datasets, or focusing on interpretability and explainability.