Comparative Analysis of Machine Learning Models for Diabetes Risk Prediction

Berk Sancak
Department of Computer Engineering
Eskisehir, Turkey
berksancak6161@gmail.com

Eray Birel
Department of Computer Engineering
Eskisehir, Turkey
eraybirel@gmail.com

***Abstract*—In this study, we explored the application of machine learning to predict diabetes risk using the BRFSS 2015 dataset. As part of our Pattern Recognition course project, we implemented Support Vector Machines (SVM), Random Forest, Gradient Boosting, Logistic Regression, and a Voting Classifier ensemble. We evaluated these models using accuracy, AUC, and F1-score metrics. Our findings highlight the Gradient Boosting model as the top performer, achieving an accuracy of 0.7536 and an F1-score of 0.7628. To make our work accessible, we developed a Gradio interface, which was a challenging yet rewarding experience.**

***Index Terms*—diabetes prediction, machine learning, ensemble learning, support vector machines, random forest**

I. Introduction

Diabetes is a global health concern, impacting millions with its long-term effects. During our Pattern Recognition course, we became interested in leveraging machine learning to aid early detection. After exploring available datasets, we chose the BRFSS 2015 dataset from Kaggle, which offered a rich set of health indicators. Our goal was to build a predictive model and create a user-friendly tool, despite our initial limited experience with ensemble methods and interface design.

II. Methods

A. Dataset

We sourced our data from the BRFSS 2015 dataset on Kaggle [1], which includes 70,692 samples with a balanced 50/50 split of diabetic and non-diabetic cases. Features like BMI, HighBP, and Age caught our attention, and we spent time understanding their relevance to diabetes risk.

B. Preprocessing

We standardized the features using StandardScaler, a step that took us some trial and error to get right. We split the data into 80% training and 20% testing sets, ensuring a fair evaluation of our models.

C. Models

As beginners, we decided to start with default hyperparameters for the following models:

- Support Vector Machine (SVM) with a linear kernel.
- Random Forest with 100 trees.
- Gradient Boosting with 100 trees.
- Logistic Regression.
- Voting Classifier, which combined the above models and was our attempt to improve results.

Choosing these models was a learning process, as we debated their suitability based on course lectures.

D. Evaluation Metrics

We assessed our models using accuracy, AUC, and F1-score, which we selected after discussing their importance in imbalanced data scenarios. We also generated confusion matrices to better visualize our predictions.

III. Results

Table I shows the performance of our models. The Gradient Boosting model stood out with an accuracy of 0.7536 and an F1-score of 0.7628. Figure 1 displays the confusion matrices, which helped us identify where our models struggled most.

TABLE I
Performance Metrics of the Models

Model	Accuracy	AUC	F1-Score
SVM	0.7485	0.7486	0.7586
Random Forest	0.7380	0.7382	0.7484
Gradient Boosting	0.7536	0.7537	0.7628
Logistic Regression	0.7484	0.7485	0.7531
Voting Classifier	0.7507	0.7507	0.7541

Fig. 1. Confusion Matrices of the Models

IV. Discussion

We were surprised to see Gradient Boosting outperform others, especially since we initially favored Random Forest due to its popularity. The F1-score of 0.7628 suggested it handled the balanced dataset well, though we noticed some misclassifications in the confusion matrices, particularly with false positives. Random Forests lower accuracy (0.7380) frustrated us at first, but we realized it might be overfitting to noisy features like MentHlth. The Voting Classifiers balanced performance (0.7507) was a relief, showing our ensemble idea had merit. A key challenge was using default hyperparameters—next time, we'll tune them to see if we can push the accuracy higher.

V. Conclusion

This project taught us how to apply machine learning to a real-world problem like diabetes prediction, with Gradient Boosting emerging as the best model. Moving forward, we plan to experiment with hyperparameter tuning and explore additional datasets to refine our approach.

Acknowledgment

We extend our gratitude to our instructors for their guidance and to our peers for their feedback during this project.

References

[1] A. Teboul, Diabetes Health Indicators Dataset, Kaggle, 2021. [Online]. Available: https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset

[2] F. Pedregosa et al., Scikit-learn: Machine Learning in Python, *J. Mach. Learn. Res.*, vol. 12, pp. 28252830, 2011.

[3] A. Abid et al., Gradio: Hassle-Free Python Web App Development, 2020. [Online]. Available: https://gradio.app/

[4] W. McKinney, Data Structures for Statistical Computing in Python, in *Proc. 9th Python Sci. Conf.*, 2010, pp. 5661.

[5] C. R. Harris et al., Array Programming with NumPy, *Nature*, vol. 585, pp. 357362, 2020.

[6] J. D. Hunter, Matplotlib: A 2D Graphics Environment, *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 9095, 2007.