Predictive Modelling for Diabetes Mellitus with Respect to Basic Medical History

Patrick Purta, Aryan Mishra, Vishal Reddy Vadde, Ruthvika Bojjala, Gopichand Jagarlamudi and Bonaventure Chidube Molokwu[a]

Department of Computer Science, College of Engineering and Computer Science, California State University, Sacramento, U.S.A.

Keywords: Diabetes Mellitus, SMOTE, SVM-SMOTE, SMOTE-ENN, Predictive Modeling, Risk Assessment, Clinical Decision Support, Medical History.

Abstract: In our work herein, we observed how three (3) common oversampling techniques - SMOTE, SMOTE-ENN, and SVM-SMOTE - affect the performance of Machine Learning (ML) models applied towards predicting diabetes risk with reference to the Pima-Indian (Akimel O'odham) Diabetes dataset. Our aim was to figure out if using these methods to mitigate class imbalance, in a medical dataset, might cause the ML models to overfit - in other words, they tend to do very well on the training data but lose fitness and accuracy on new data. Our project began from a simple question: "Can oversampling fix class imbalances, with respect to a given dataset, without hurting the model's ability to generalize?" Previous studies have shown that oversampling can help balance target-classes within a dataset, but these studies do not always address the risk of overfitting. To answer this, we combined each oversampling technique via three (3) ensemble methods - Extra Trees, Gradient Boosting, and Random Forest - and compared their performances via cross-validation objective functions. Our results reveal that, although each method improves the results or metrics on the training data, they tend to under-perform slightly on unseen test or sample data. This suggests that while oversampling is a useful strategy, it must be applied with caution to avoid overfitting. These insights are important for refining predictive models, especially in healthcare contexts where reliable performance is critical.

1 INTRODUCTION

Working on this project has been like untangling a knot in a routine we all know too well, balancing the scales when it comes to data. In healthcare, especially with something as crucial as diabetes diagnosis, having a dataset where the positive cases are much fewer than the negatives is common; and it often leaves our models struggling to learn the right patterns. Imagine trying to hear a soft whisper in a loud room; the less frequent instances get drowned out.

We started exploring this issue because we noticed something interesting with oversampling techniques like Synthetic Minority Over-sampling Technique (SMOTE), SMOTE-ENN (SMOTE + Edited Nearest Neighbors), and SVM-SMOTE (Support Vector Machine SMOTE). These methods essentially "create" more examples from the minority class so that every voice in the dataset gets a chance to be heard. However, while these techniques boosted performance during training, our models - built with Extra Trees, Gradient Boosting, and Random Forest - sometimes became too focused on the training data, showing signs of overfitting when it came time to predict new cases.

This raised a pressing question for our team: "Can we use oversampling to balance the dataset without inadvertently making the models overconfident with reference to the training examples?" With the Pima-Indian Diabetes dataset (Nnamoko and Korkontzelos, 2020) as our test case, we set out to find an answer because, in healthcare, even the smallest error(s) can yield big consequences. The goal is not just to record high objective-function scores during training, but to build tools that work reliably when it really counts.

To address this issue, we employed a thorough hands-on approach. Individually, we applied each oversampling technique separately to our training data; thereafter, we carefully evaluated how different ensemble models performed using a cross-validation strategy. By comparing training and testing results,

[a] https://orcid.org/0000-0003-4370-705X

Purta, P., Mishra, A., Vadde, V. R., Bojjala, R., Jagarlamudi, G. and Molokwu, B. C.
Predictive Modelling for Diabetes Mellitus with Respect to Basic Medical History.
DOI: 10.5220/0013692100003985

In *Proceedings of the 21st International Conference on Web Information Systems and Technologies (WEBIST 2025)*, pages 343-349
ISBN: 978-989-758-772-6; ISSN: 2184-3252

we aimed at a better understanding of where our models were getting it right and where they were losing their grip.

At its heart, our project is more than just statistics - it is about making a realistic difference in how we use technology in healthcare. By delving deeper into the balance between correcting data imbalances and avoiding overfitting, we do hope that our work can guide the development of more robust as well as trustworthy diagnostic tools that put patient-care first.

2 RELATED LITERATURE

In our effort to improve how machines predict diabetes, we dug into a lot of previous work to see how others have handled similar challenges. This review granted us insights into the past approaches and/or methodologies.

2.1 Background of the Work

Researchers have long struggled with imbalanced data, especially in healthcare, where the number of positive cases (diabetes diagnosis) is often much smaller than the negative cases. One of the earliest breakthroughs came with the introduction of SMOTE (Chawla et al., 2002), which set the stage for creating synthetic examples to balance the dataset. Since then, numerous projects have applied these methods - especially to datasets like the Pima-Indian-Indian Diabetes dataset - to better understand how boosting the minority class can help the overall model. However, these projects also revealed a common downside: while oversampling improves training results, it sometimes leads the models to perform less effectively when faced with new data.

2.2 Building a Strong Knowledge Base

A number of articles have reinforced the idea that oversampling techniques such as SMOTE, SMOTE-ENN, and SVM-SMOTE can help level the playing field during model training - by providing more examples from the under-represented class. For example, SVM-RBF: Support Vector Machine (SVM) with Radial Basis Function (RBF), Decision Tree, Naive Bayes, and RIPPER learning algorithms were coupled with SMOTE to improve classification performance on the Pima-Indian dataset. Studies that compared these techniques have shown that although they improve the class balance during training, they can also make a model too "comfortable" with the training data, resulting in overfitting(Santos et al., 2018). In other words, the model learns the training data so well that it struggles to adapt when shown data it has not seen before. Researchers (Poornima and R., 2024) have also experimented with ensemble methods - e.g. Extra-Trees, Random Forest(Olisah et al., 2022), Gradient Boosting, etc. - highlighting both the benefits and the challenges of combining these oversampling techniques with robust classifiers(Zhang et al., 2024).

2.3 Theoretical Support and Methodological Framework

What ties all these together is the well-known trade-off between reducing bias and increasing variance. Oversampling helps reduce bias by making sure the model gets trained on sufficient examples of the minority class, but it can also lead to higher variance - meaning that the model might not perform well on unseen data. Several literature supports using cross-validation as a way to tackle this problem. Cross-validation (CV) helps check whether a model's good performance on the training data transcends to when it encounters new data. Many studies (Santos et al., 2018) have used this approach to ensure that the benefits of oversampling are not lost due to overfitting. This blend of theory and practical provides the framework for our work herein - giving us guidance on how to set up our experiments and interpret our results.

By examining the aforementioned past efforts and work — early from the development of SMOTE to recent studies on ensemble methods — we have built a solid foundation for our work herein. Our review of related literature not only highlights the strengths and weaknesses of existing methods, but it also highlights areas where improvements can be made. Thus, this background supports our goal towards striking an ideal balance between fixing data imbalances and keeping our models adaptable in real-world healthcare domains (Mooney, 2018; Lugat, 2021).

3 PROPOSED FRAMEWORK AND METHODOLOGY

3.1 Formalism with Respect to the Problem Statement

Given a medical dataset, $D = (x_i, y_i)$, for $i = 1 \ldots n$, and with input features, $x_i \in \mathbb{R}^d$, and binary outcomes, $y_i \in \{0, 1\}$, where class imbalance exists such that:

$$\sum_{i=1}^{n} y_i << \sum_{i=1}^{n} (1 - y_i)$$

i.e. the number of diabetic cases, y_i, is much smaller than non-diabetic cases, $1 - y_i$; the goal is to develop a robust classification function: $f(x) : \mathbb{R}^d \rightarrow \{0,1\}$, such that $f(x)$ accurately predicts the presence of diabetes (outcome = 1) using ensemble classifiers, while reducing the effects of class imbalance via oversampling techniques. This leads to the following optimization objective: Minimize over $\big(f \in F : L(f(x),y)\big)$ subject to the training set being balanced via an oversampling method, O, where $O \in \{\text{SMOTE, SMOTE-ENN, SVM-SMOTE}\}$.

3.2 Methodology Overview

1. Data Preparation: We began by cleaning the dataset — replacing zero-values in medically critical fields (like glucose level, blood pressure, BMI, etc.) with `NaN`-values, and then imputing the missing values using the KNN-Imputer method. This helps ensure the model is not biased via invalid misleading inputs.
2. Train-Test Split: The dataset was split into training and testing sets using stratified sampling in a bid to preserve the proportion of diabetic and non-diabetic cases in both sets.
3. Oversampling the Minority Class: We tested three (3) widely-used oversampling techniques: `SMOTE`: Generates synthetic samples for the minority class based on k-nearest neighbors. `SMOTE-ENN`: Combines `SMOTE` with Edited Nearest Neighbors to also clean noisy samples. `SVM-SMOTE` (Demidova and Klyueva, 2017): A variant of `SMOTE` that uses an SVM to better define the border of the minority class.
4. Feature Scaling and Standardization: After imputation, we applied a Quantile Transformer to standardize the data and reduce skewness. This step is crucial before training, especially when using models that are sensitive to large-scale data values.
5. Model Training: For each oversampling technique, we trained three (3) models — Random Forest, Extra Trees, and Gradient Boosting — on the resampled training data.
6. Model Evaluation: Finally, we evaluated each model using multiple metrics such as Accuracy, Precision, Recall, F1-score, AUC (Area Under the Curve), and MCC (Matthews Correlation Coefficient). This helped us understand not just how accurate the models were, but how well they handled both classes (Kaliappan et al., 2024).

3.3 Formal Algorithm

The formal algorithm for the Oversampling and Model-Training Pipeline begins with the input of a dataset, D, an oversampler, O, and a classifier, C, with the aim of producing a trained model, M, as the output. Initially, the dataset, D, is split into features, X, and labels, y. Any missing values in the features are handled via a KNN-Imputer. Subsequently, the features, X, are standardized via a Quantile Transformer to ensure a uniformly scaled distribution. The dataset is then further divided into Training and Testing sets: `X_train`, `X_test`, `y_train`, `y_test`. The oversampler, O, is applied to the Training data to generate: `X_train_oversampled` and `Y_train_oversampled`. After oversampling, the classifier, C, is trained on the oversampled Training data. Finally, the trained model, M, is returned.

4 PROPOSED SYSTEM ARCHITECTURE AND SETUP

The framework of our proposed system is illustrated via Figure 1; while Table 1 represents the description of our benchmark dataset. Table 2 showcases a handful of the hyperparameter configurations for our experimental setup. Table 3 highlights the objective functions (or performance metrics) employed herein to evaluate how well each benchmark model performed - especially in terms of identifying diabetic patients (the minority class). Table 4 lists the baseline models we have employed toward benchmarking and comparative analyses.

Table 1: Description of Dataset.

Property	**Description**
Name	Pima-Indian Diabetes Dataset
Source	UCI Machine Learning Repository / Kaggle
Instances	768
Features	8 input features + 1 binary output (Outcome)
Target Variable	Outcome (1 = diabetic, 0 = non-diabetic)
Missing Values	Handled using KNN-Imputer

5 EXPERIMENT AND RESULTS

We used the Pima-Indian diabetes dataset, a widely recognized dataset in medical and healthcare re-

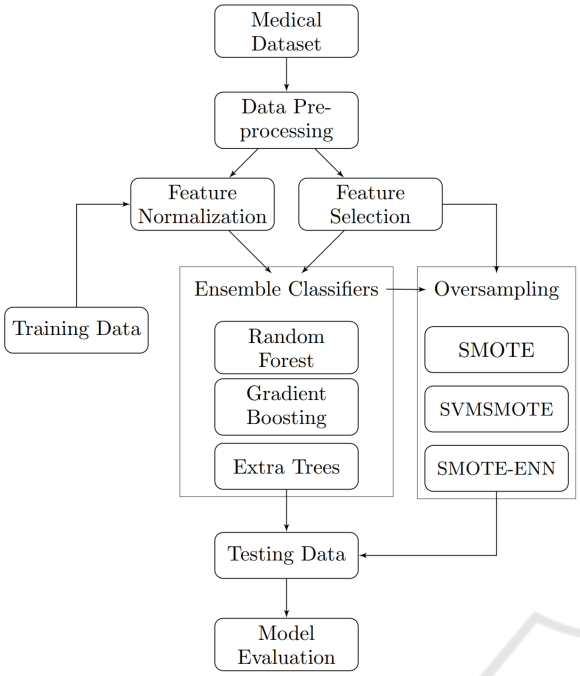

Figure 1: Architectural overview of our proposed system.

Table 2: Hyperparameters Configuration/Setting.

Component	Parameter	Value(s) Tested
SMOTE	k_neighbors	5
SVMSMOTE	k_neighbors	5
Random Forest	n_estimators	100
Extra Trees	n_estimators	100
Gradient Boosting	learning_rate	0.1
	n_estimators	100
KNNImputer	n_neighbors	5
Quantile Transformer	output_distribution	'normal'

Table 3: Objective Functions (Performance Metrics).

Metric	Purpose
Accuracy	Measures overall correctness
Precision	Ratio of True Positives to predicted Positives
Recall	Sensitivity to positive cases
F1-score	Harmonic mean of Precision and Recall
ROC-AUC	Area under ROC curve
MCC	Balanced measure for imbalanced datasets

search, to thoroughly test the effectiveness of three (3) robust ML algorithms: Random Forest, Extra Trees, and Gradient Boosting. Each of these algorithms was carefully selected based on their known strengths in handling classification tasks, especially with medical datasets. To ensure our models effectively manage data imbalance — a common issue in medical diagnosis — we employed multiple sampling techniques (SMOTE, SVM-SMOTE, and SMOTE-ENN) as well as "No Sampling" technique.

Table 4: Baseline Models for Comparison.

Model	Description
Logistic Regression	Simple linear classifier
Naive Bayes	Probabilistic classifier
Decision Tree	Base model for tree ensembles
Dummy Classifier	Majority class predictor (baseline)

Table 5: Accuracy scores (Train/Test).

Model	No Sampling	SMOTE	SVM-SMOTE	SMOTE-ENN
Random Forest	0.759/ 0.721	0.819/ 0.766	0.814/ 0.753	0.961/ 0.721
Extra Trees	0.756/ 0.714	0.823/ 0.734	0.833/ 0.734	0.960/ 0.727
Grad-Boost	0.754/ 0.760	0.785/ 0.740	0.788/ 0.760	0.941/ 0.766

Table 6: F1-scores (Train/Test).

Model	No Sampling	SMOTE	SVM-SMOTE	SMOTE-ENN
Random Forest	0.755/ 0.716	0.820/ 0.750	0.804/ 0.726	0.961/ 0.726
Extra Trees	0.748/ 0.710	0.822/ 0.735	0.833/ 0.736	0.960/ 0.733
Grad-Boost	0.752/ 0.754	0.785/ 0.745	0.787/ 0.764	0.941/ 0.771

Table 7: Matthews Correlation Coefficient, MCC, scores (Train/Test).

Model	No Sampling	SMOTE	SVM-SMOTE	SMOTE-ENN
Random Forest	0.456/ 0.420	0.631/ 0.457	0.606/ 0.490	0.893/ 0.427
Extra Trees	0.452/ 0.420	0.639/ 0.462	0.644/ 0.401	0.926/ 0.463
Grad-Boost	0.467/ 0.455	0.571/ 0.390	0.583/ 0.468	0.886/ 0.508

6 DISCUSSION

Table 8 showcases the experiment results with respect to our Extra-Trees model, and Table 9 represents the experiment results with respect to our Gradient-

Boosting model.

Table 8: Extra-Trees model (Train/Test).

Sampling Method	Accuracy (Train/Test)	F1-score (Train/Test)	Interpretation
No Sampling	0.756/ 0.714	0.748/ 0.710	Minor overfitting, ∼5% variance
SMOTE	0.823/ 0.734	0.822/ 0.735	Better scores but overfitting by 7-10%
SVM-SMOTE	0.833/ 0.734	0.833/ 0.736	Better scores but overfitting by 7-10%
SMOTE-ENN	0.960/ 0.727	0.960/ 0.733	Major overfitting, up to 25%

Table 9: Gradient-Boosting model (Train/Test).

Sampling Method	Accuracy (Train/Test)	F1-score (Train/Test)	Interpretation
No Sampling	0.754/ 0.760	0.752/ 0.754	No overfitting, ∼1% variance
SMOTE	0.785/ 0.740	0.785/ 0.745	Minor overfitting, 1-5% variance
SVM-SMOTE	0.788/ 0.760	0.787/ 0.764	Minor overfitting, 1-5% variance
SMOTE-ENN	0.941/ 0.766	0.941/ 0.771	Major overfitting, up to 20%

6.1 Performance Analysis

The experimental results reveal that the performance of the ensemble classifiers — Random Forest, Extra Trees, and Gradient Boosting — varied significantly depending on the oversampling technique applied. Random Forest consistently achieved the highest training scores across all oversampling methods, with training Accuracy reaching as high as 0.961 and training F1-score also at 0.961 when using SMOTE-ENN. However, these impressive training metrics did not carry over to the Test set, where the corresponding Accuracy and F1-score dropped to 0.721 and 0.726, respectively. This large disparity suggests significant overfitting, where the model learns the Training data too closely and fails to generalize well to new, unseen data. A similar pattern was observed in the Extra-Trees classifier, where training Accuracy and F1-score under SMOTE-ENN were both 0.960, but Test set values fell to 0.727 and 0.733, respectively. This reinforces the conclusion that aggressive oversampling techniques like SMOTE-ENN can lead to overly optimistic training performance while compromising real-world applicability.

In contrast, Gradient Boosting demonstrated a more balanced performance between the training and testing phases. The differences between the training and test metrics were much narrower, especially with SMOTE and SVM-SMOTE. For example, when paired with SVM-SMOTE, Gradient Boosting achieved a training Accuracy of 0.788 and a test Accuracy of 0.760, with F1-scores of 0.787 and 0.764, respectively. This indicates only a minor degree of overfitting, suggesting that Gradient Boosting may be better at generalizing from the Training data while still benefiting from oversampling. These findings are further supported by the Matthews Correlation Coefficient (MCC) score, which reflects the quality of binary classifications in imbalanced datasets. Although, Random Forest and Extra-Trees yielded very high MCC values on the Training data (up to 0.926); their test MCC scores were notably lower. In contrast, Gradient Boosting with SMOTE-ENN achieved the highest test MCC at 0.508, indicating a stronger ability to maintain balanced performance across both classes.

6.2 Interpretation of Results

The primary performance indicators used to evaluate the models were Accuracy, F1-score, and Matthews Correlation Coefficient (MCC); which together provide a multidimensional view of the classification quality. Accuracy captures the overall correctness of the model, while F1-score accounts for the trade-off between Precision and Recall, making it particularly useful in imbalanced medical datasets. MCC offers a single summary metric that reflects the balance of both True and False Positives as well as negatives - making it especially relevant when the dataset is skewed.

While training Accuracy for all models frequently exceeded 90%, such high values were not replicated on the Test set where most accuracy values hovered between 71% and 76%. This discrepancy underscores the presence of overfitting, particularly in models like Random Forest and Extra Trees when trained with SMOTE-ENN. F1-scores on the Test set ranged between 0.71 and 0.77, with Gradient Boosting paired with SMOTE-ENN yielding the highest test F1-score of 0.771. This suggests that the combination allowed for the most effective balance between detecting true diabetic cases and avoiding False Positives. Meanwhile, the MCC results offered a more nuanced interpretation of model balance. Gradient Boosting with SMOTE-ENN once again led with a test MCC of 0.508, indicating strong classification consistency. Random Forest with SVM-SMOTE also performed

welll with a test MCC of 0.490.

These results suggest that while oversampling methods can enhance model performance by mitigating class imbalance, they must be applied with caution to avoid overfitting. Among the classifiers tested, Gradient Boosting demonstrated the most robust and reliable generalization, especially when combined with SMOTE-ENN or SVM-SMOTE. These combinations improved the model's ability to identify diabetic cases effectively without significantly compromising performance on the majority class, which is essential in high-stake medical prediction tasks.

6.3 Implications, Benefits, and Contribution to Humanity

The results from this work and project have several important real-world implications, viz:

(a) *Early and accurate detection of diabetes:* Our models can help identify positive diabetes cases earlier, thereby enabling timely interventions that can prevent serious complications, improve patient outcomes, and significantly reduce healthcare costs.

(b) *Deployment in real-world scenarios:* Credits to the robust performance after advanced preprocessing (especially SVM-SMOTE and SMOTE-ENN). This enabled the models to be confidently deployed in critical fields such as healthcare or finance, where imbalanced data is common and high-stake decisions are made.

(c) *Broader applicability:* The techniques and workflow we demonstrated herein, combining strong ensemble models with smart preprocessing techniques, can be applied to other imbalanced problems like cancer detection, fraud detection, or predicting rare diseases.

7 CONCLUSIONS AND FUTURE

7.1 Limitations

Despite the promising outcomes herein, our research has several limitations. Primarily, our evaluation was limited to the Pima-Indian dataset which predominantly includes data from women of (Indigenous) Pima-Indian heritage; thus potentially restricting the model's applicability to broader populations and genders. Additionally, advanced Deep Learning architectures and a wider range of preprocessing methods were not explored.

7.2 Summary of Model

Our developed models are effective analytical tools capable of ingesting and processing medical and clinical data. The model is capable of taking tabular numeric data as input, where each row is an instance of a patient's medical profile and each column represents features such as pregnancy count, glucose levels, insulin levels, etc. In the initial phase of data transformation, missing and/or invalid values were identified in the dataset with respect to features like Glucose level, BloodPressure, SkinThickness, Insulin, and BMI. These features contained invalid zero entries, which were later transformed into NaN-values, and imputed using KNN-Imputer to maintain data integrity.

Preprocessing stage consists of several steps to prepare the data for modeling. Firstly, we tackled missing values by filling them up, then we used EDA (Exploratory Data Analysis): using histograms to show that the distribution of features has been corrected, and confirming there are no more zero-entries. Feature selection is done with the help of SelectKBest and ANOVA F-test (f_classif), so that we can choose the most relevant features which correlate with the target for our model. The dataset is then cross-validated with train_test_split.

Data oversampling is performed using SMOTE, SVMSMOTE, or SMOTE-ENN, depending on which sampling method the user chooses to include in the model. The Testing data should not have any knowledge of the Training data. Our oversampling methods create synthetic data points via interpolation between two (2) original data points with respect to a given dataset. If the Test set includes these synthetic data points, then it has some knowledge of the Training data, and this may cause bias in the resultant model's learning (Zhang et al., 2024). Therefore, oversampling is performed only on the Training data. Subsequently, the features are standardized using QuantileTransformer with a normal-distribution output to reduce skewness and ensure that the model sees uniformly distributed input features.

Upon completion of preprocessing, model training is done robustly with the help of k-fold cross validation so that the performance of the model can be evaluated effectively. To properly evaluate the model's generalization capability, key performance metrics such as Accuracy and F1-score are calculated for both Training data and Testing data which we have explained in detail in the previous section. The model yields its results with respect to standard objective functions/metrics for classification tasks. The primary output is whether a patient is diabetic or not

diabetic. This research is not just limited to only medical data, it can be adapted to any type of numeric data requiring a classification task.

7.3 Future Work

Future research efforts will substantially broaden the scope of our investigation. We plan to incorporate more diverse and extensive datasets, encompassing varied populations to enhance the generalizability and robustness of the predictive models herein. Additionally, further exploration into advanced Deep Learning methods such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and hybrid Deep Learning models is anticipated. These sophisticated architectures could uncover deeper patterns within complex medical data, potentially offering substantial improvements in predictive performance. Moreover, detailed preprocessing techniques including advanced feature engineering, dimensionality reduction techniques such as Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE), will be rigorously examined to optimize the feature set for enhanced predictive Accuracy scores.

Addressing the interpretability of these models will also be an essential aspect of our future work, aiming to produce models that are not only accurate but also easily interpretable by healthcare professionals. In addition to technical advancements, we intend to deploy our model on accessible and user-friendly platforms, such as interactive web applications or mobile apps. This approach aims to facilitate real-time diabetes risk assessment tools for healthcare professionals and patients alike, promoting proactive health management.

Collaborations with clinical institutions and healthcare providers will also be pursued to validate our model further through extensive real-world clinical trials and implementations, ensuring practical relevance and efficiency in varied clinical domains.

ACKNOWLEDGMENT

The authors would like to thank the Department of Computer Science at California State University, Sacramento, for providing the necessary resources and guidance throughout the course of this project. We also extend our gratitude to the faculty members and mentors who offered valuable insights and support during the research and experimentation phases. Special thanks to the creators and managers of the Pima-Indian Diabetes Dataset for making the dataset publicly available, which served as a critical foundation for our study. Finally, we would like to acknowledge the contributions of all team members whose dedication, collaboration, and shared problem-solving efforts were instrumental in the successful completion of this project.

REFERENCES

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Demidova, L. and Klyueva, I. (2017). Svm classification: Optimization with the smote algorithm for the class imbalance problem. In *2017 6th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–4.

Kaliappan, J., Kumar, I. J. S., Sundaravelan, S., Anesh, T., Rithik, R. R., Singh, Y., Vera-Garcia, D. V., Himeur, Y., Mansoor, W., Atalla, S., and Srinivasan, K. (2024). Analyzing classification and feature selection strategies for diabetes prediction across diverse diabetes datasets. *Frontiers in Artificial Intelligence*, 7:1421751.

Lugat, V. (2021). Pima indians diabetes - eda and prediction (0.906). https://www.kaggle.com/code/vincentlugat/pima-indians-diabetes-eda-prediction-0-906. Accessed: 2025-05-18.

Mooney, P. T. (2018). Predict diabetes from medical records. https://www.kaggle.com/code/paultimothymooney/predict-diabetes-from-medical-records. Accessed: 2025-05-18.

Nnamoko, N. and Korkontzelos, I. (2020). Efficient treatment of outliers and class imbalance for diabetes prediction. *Artificial intelligence in medicine*, 104:101815.

Olisah, C. C., Smith, L. N., and Smith, M. L. (2022). Diabetes mellitus prediction and diagnosis from a data preprocessing and machine learning perspective. *Computer methods and programs in biomedicine*, 220:106773.

Poornima, V. and R., R. (2024). A hybrid model for prediction of diabetes using machine learning classification algorithms and random projection. *Wirel. Pers. Commun.*, 139:1437–1449.

Santos, M. S., Soares, J. P., Abreu, P. H., Araújo, H., and Santos, J. A. M. (2018). Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [research frontier]. *IEEE Computational Intelligence Magazine*, 13:59–76.

Zhang, Z., Ahmed, K. A., Hasan, M. R., Gedeon, T., and Hossain, M. Z. (2024). A deep learning approach to diabetes diagnosis. In *Asian Conference on Intelligent Information and Database Systems*, pages 87–99. Springer.