Tuijin Jishu/Journal of Propulsion Technology
ISSN: 1001-4055
Vol. 47 No. 02 (2026)

A Robust and Reproducible Machine Learning Pipeline for Diabetes Prediction Using Feature Engineering and Ensemble Learning

Harun Kamau[1], Mwangi E. Karanja[1] and Jael Wekesa[1]

[1] *Department of Computing, Jomo Kenyatta University of Agriculture and Technology, Nairobi, Kenya;*

Abstract-Diabetes mellitus remains a major global health challenge, requiring accurate and early prediction to reduce the risk of severe complications and improve patient outcomes. Although machine learning techniques have been widely applied to diabetes prediction, existing studies often address key challenges such as class imbalance, feature representation, and model optimization in isolation, leading to suboptimal and non-reproducible predictive performance. This study addresses this gap by proposing a comprehensive and integrated machine learning pipeline for diabetes prediction using the Pima Indians Diabetes Dataset. The proposed framework systematically combines data preprocessing, feature engineering, dimensionality reduction, class imbalance handling, and ensemble learning within a unified pipeline. Missing values are handled using median imputation, while outliers are treated through interquartile range (IQR)-based winsorization. Three domain-informed features Glucose_per_BMI, Age_BMI, and HighPreg are introduced to capture nonlinear relationships, followed by feature selection and Principal Component Analysis (PCA). To address class imbalance, SMOTE-Tomek resampling is applied, improving minority class representation. Multiple models, including Logistic Regression, Support Vector Machines, Random Forest, XGBoost, LightGBM, and a stacked ensemble, are evaluated using cross-validation and a hold-out test set. Experimental results demonstrate that the proposed pipeline significantly improves predictive performance, with accuracy increasing from 74.03% to 82.11%, F1-score from 53.33% to 71.11%, and balanced accuracy from 0.6893 to 0.7780. Notably, recall improved substantially across models, enhancing sensitivity to diabetic cases. Additionally, recall improved substantially, with Logistic Regression increasing from 0.6267 to 0.8000 and SVM from 0.5467 to 0.7467. LightGBM and the stacked ensemble achieved the best overall performance. These findings highlight the effectiveness of integrating preprocessing, feature engineering, and ensemble learning into a unified framework for robust and reliable diabetes prediction.

Keywords: *Diabetes Prediction, Machine Learning Pipeline, Feature Engineering, Ensemble learning*

1. Introduction

Diabetes mellitus is a chronic metabolic disorder characterized by persistent hyperglycemia resulting from defects in insulin secretion, insulin action, or both. It represents one of the most significant global health challenges of the 21st century, with rapidly increasing prevalence across both developed and developing regions. According to the International Diabetes Federation, approximately 11.1% (589 million adults) were living with diabetes in 2025, a number projected to rise to 853 million, a 46% increase (1 in 8 adults) by 2050 [1]. The disease is associated with severe long-term complications, including cardiovascular diseases, nephropathy, neuropathy, and retinopathy, which collectively contribute to increased mortality and healthcare costs. Consequently, early detection and effective management of diabetes remain critical priorities for healthcare systems worldwide.

The importance of early diagnosis lies in its ability to enable timely intervention and reduce the risk of severe complications. Traditional diagnostic approaches, such as fasting plasma glucose (FPG), oral glucose tolerance tests (OGTT), and glycated hemoglobin (HbA1c) measurements, are widely used in clinical practice. However,

these methods often require specialized laboratory infrastructure, are time-consuming, and may not be suitable for large-scale screening, particularly in resource-limited settings. As healthcare systems increasingly shift toward preventive care, there is a growing demand for scalable, efficient, and cost-effective tools capable of identifying individuals at high risk of developing diabetes using readily available clinical data.

Recent advances in machine learning (ML) have provided promising alternatives for disease prediction and risk stratification. ML algorithms, including Logistic Regression, Support Vector Machines (SVM), Random Forests, k-Nearest Neighbors (k-NN), and Gradient Boosting models, have been extensively applied to diabetes prediction tasks. These techniques are capable of learning complex patterns from multidimensional datasets and generating predictive models with high accuracy. Studies such as those by Ganie et al. [2] and Almutairi et al [3] demonstrate the effectiveness of ML approaches in analyzing clinical attributes such as glucose levels, body mass index (BMI), age, and insulin levels. The availability of benchmark datasets, particularly the Pima Indians Diabetes Dataset, has further facilitated comparative analysis and methodological advancements in this domain.

Despite these advancements, several limitations persist in existing studies. One major challenge is class imbalance, where the number of non-diabetic instances significantly exceeds diabetic cases, leading to biased model predictions and reduced sensitivity. Additionally, real-world datasets frequently contain missing values and outliers, which can negatively impact model performance if not properly handled. Another critical issue is the limited ability of traditional models to capture nonlinear relationships and complex feature interactions inherent in medical data. While techniques such as Synthetic Minority Over-sampling Technique (SMOTE), feature scaling, and data imputation have been employed to address some of these issues, they are often implemented in isolation rather than as part of a unified framework. Furthermore, relatively few studies have systematically explored the combined impact of feature engineering and ensemble learning strategies within a reproducible and end-to-end pipeline.

To address these gaps, this study proposes a comprehensive and reproducible machine learning pipeline for predicting diabetes onset using the Pima Indians Diabetes dataset. The proposed approach integrates data preprocessing, feature engineering, and ensemble modeling into a unified framework. Specifically, the pipeline incorporates robust techniques for handling missing values, treating outliers, and mitigating class imbalance using SMOTE-TOMEK. In addition, novel feature engineering methods are introduced to capture meaningful interactions between clinical variables, thereby enhancing predictive performance. Multiple machine learning models are trained and combined using ensemble techniques, including stacking, to leverage complementary strengths and improve overall model generalization.

The main contributions of this study are summarized as follows:

1. We design an end-to-end machine learning pipeline for diabetes prediction that integrates data preprocessing, and feature engineering, under a unified framework, enabling reproducible experimentation on clinical datasets.
2. We introduce domain-informed feature engineering techniques that improve both model interpretability and predictive accuracy.
3. We conduct a systematic comparative evaluation of ensemble learning methods, including Random Forest, XGBoost, and LightGBM, using stratified cross-validation and multiple performance metrics (ROC AUC, PR AUC, F1-score).
4. We demonstrate that LightGBM achieves the best predictive performance on the PIMA dataset and analyze its behavior in terms of feature importance, model stability, and generalization limitations.

The remainder of this paper is organized as follows. Section 2 presents a review of related work in diabetes prediction using machine learning techniques. Section 3 describes the dataset, data preprocessing, machine learning models and experimental setup. Section 4 presents and discusses the results of the study. Finally, Section 5 concludes the paper and suggests directions for future research.

2. Related Work

Machine learning-based approaches for diabetes prediction have been extensively studied, with the Pima Indians Diabetes Dataset serving as a widely adopted benchmark. Existing literature reveals a progression from traditional classification models toward more sophisticated hybrid and ensemble-based frameworks. However, despite substantial advancements, several methodological and practical limitations remain unresolved.

Early studies on diabetes prediction primarily employed conventional machine learning algorithms such as Logistic Regression, Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), Random Forests, and Gradient Boosting. These models are typically evaluated using standard preprocessing techniques, including feature scaling, missing value imputation, and cross-validation, to ensure generalization performance [4], [5]

While these approaches provide a solid baseline, their effectiveness is often constrained by assumptions about data distribution and feature relationships. Linear models such as Logistic Regression struggle to capture nonlinear interactions among clinical variables, whereas distance-based methods like k-NN are sensitive to feature scaling and noise. Tree-based models, including Random Forests and Gradient Boosting, have demonstrated superior performance due to their ability to model complex feature interactions. However, their performance is highly dependent on data quality and preprocessing consistency, which is not uniformly addressed across studies.

To overcome the limitations of single classifiers, recent research has increasingly focused on ensemble learning techniques. Studies by Ganie et al.[2] and Abnoosian et al. [6] demonstrate that combining multiple models can significantly improve predictive stability and reduce variance. Boosting algorithms, in particular, have shown strong performance in handling complex decision boundaries and improving classification accuracy.

Despite these improvements, most ensemble-based approaches are implemented in isolation, without integrating complementary preprocessing and feature engineering strategies into a unified framework. Furthermore, while ensemble models often achieve higher accuracy, they may introduce increased computational complexity and reduced interpretability, limiting their practical applicability in clinical settings. Recent work by Rafie et al. [7] attempts to address this issue by integrating explainable AI techniques with XGBoost, highlighting the growing importance of interpretability in medical decision-making.

Class imbalance remains one of the most persistent challenges in diabetes prediction. In datasets such as PIMA, the minority diabetic class is often underrepresented, leading to biased models that favor the majority class. Various studies have explored resampling techniques to mitigate this issue. For instance, Hama Saeed [8] applies up-sampling methods to improve classification sensitivity, while Mujahid et al. [9] demonstrate that combining oversampling with feature optimization yields improved generalization performance. Advanced techniques such as SMOTE, ADASYN, and hybrid methods like SMOTE-ENN have shown promise in generating synthetic samples while reducing noise and class overlap.

However, a critical limitation in existing work is that class balancing techniques are often applied as standalone solutions. Without integration into a broader modeling pipeline, these methods may lead to overfitting or fail to generalize effectively across unseen data. Moreover, many studies prioritize accuracy over clinically relevant metrics such as recall and F1-score, which are more appropriate for imbalanced medical datasets.

Recent literature increasingly emphasizes the importance of feature representation in improving predictive performance. Abdollahi et al.[10] introduce a deep learning approach coupled with genetic algorithm-based feature selection, demonstrating the promise of hybrid evolutionary learning in diabetes classification tasks. The broader medical domain also benefits from methodological innovations. Bouqentar et al. [11] and Raza et al.[12], show how tailored feature engineering enhances early heart disease prediction, leverage ensemble-based feature engineering to assess maternal health risks during pregnancy. Studies such as Khan et al. [13] highlight that robust feature selection methods including recursive feature elimination, mutual information, and regularization techniques can significantly enhance model accuracy and interpretability.

Similarly, Tukhtaev et al. [14] demonstrate that domain-specific feature engineering, including interaction features and nonlinear transformations, leads to improved performance in medical prediction tasks. These findings suggest that model performance is not solely dependent on algorithm choice but is heavily influenced by the quality and structure of input features.

Nevertheless, many studies treat feature engineering as a secondary step rather than a central component of the modeling pipeline. In addition, there is limited exploration of how engineered features interact with ensemble models, representing a gap in current research.

Beyond predictive accuracy, recent research has shifted toward improving model transparency and automation. Explainable AI techniques, such as SHAP and LIME, have been integrated into predictive models to provide insights into feature importance and decision-making processes. For example, Chen et al. [15] utilize SHAP-based explanations in clinical prediction tasks, demonstrating the importance of interpretability in healthcare applications.

In parallel, Automated Machine Learning (AutoML) frameworks, as highlighted by Yuan et al. [16], are gaining traction for their ability to optimize model selection and hyperparameters. While these approaches improve scalability and reproducibility, they often operate as black-box systems and may lack domain-specific customization required in medical applications.

A critical analysis of existing literature reveals several gaps. First, many studies focus on isolated improvements such as model selection, resampling techniques, or feature engineering without integrating these components into a cohesive and reproducible pipeline. Second, limited attention has been given to the interaction between feature engineering and ensemble learning, despite evidence that both significantly influence model performance. Third, while class imbalance handling is widely studied, its integration with feature optimization and ensemble strategies remains inconsistent. Finally, issues of reproducibility and systematic evaluation across multiple modeling stages are often inadequately addressed.

To address these limitations, the present study proposes a comprehensive machine learning pipeline that unifies data preprocessing, feature engineering, class imbalance handling, and ensemble modeling within a single framework. Unlike prior approaches that treat these components independently, this study systematically integrates them to enhance predictive accuracy, robustness, and reproducibility. By evaluating multiple models and incorporating engineered features alongside advanced resampling techniques, this work contributes a structured and extensible approach to diabetes prediction using the PIMA dataset.

3. Material and Methods

This section discusses the setting of the experiments, datasets, and classification algorithms and their evaluation metrics. This study proposes a comprehensive machine learning pipeline for the prediction of diabetes using the Pima Indians Diabetes Dataset. The methodology integrates systematic data preprocessing, feature engineering, class imbalance handling, dimensionality reduction, and ensemble learning techniques to improve predictive performance and model robustness. The overall workflow of the proposed system is illustrated in Figure 1. The methodology consists of six major stages: dataset acquisition, data preprocessing, feature engineering, class imbalance correction, model training, and model evaluation.

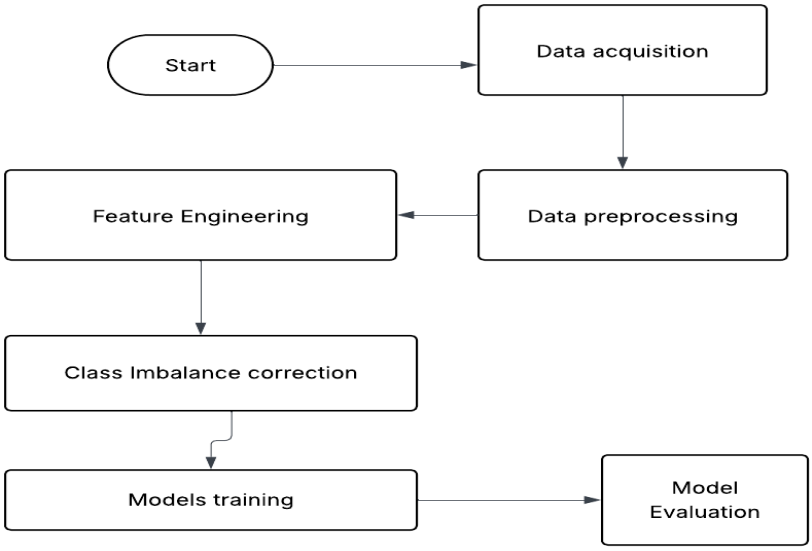

Figure 1: Proposed Machine Learning Pipeline

3.1 Dataset Description

The experiments in this study utilize the Pima Indians Diabetes Dataset, which is widely used in medical machine learning research for evaluating diabetes prediction models. The dataset contains 768 patient records collected from female patients of Pima Indian heritage aged 21 years or older. Each record consists of eight clinical predictor variables and a binary outcome variable indicating whether the patient was diagnosed with diabetes. The features are described in table 1.

Table 1: Dataset description

Feature	Description	Value range
Pregnancies	Number of pregnancies	0–17
Glucose	Plasma glucose concentration	0–199
BloodPressure	Diastolic blood pressure (mm Hg)	0–12
SkinThickness	Triceps skin fold thickness (mm)	0–99
Insulin	2-hour serum insulin (mu U/ml)	(0–84)
BMI	Body mass index (weight in kg/(height in m)^2)	0–6
DiabetesPedigreeFunction	Genetic diabetes likelihood score	0.07–2.4
Age	Age of the patient	21–81
Outcome	Diabetes diagnosis (0 = No, 1 = Yes)	0 or1

The target variable is Outcome, indicating the presence (1) or absence (0) of diabetes. Notably, several attributes contain zero values representing missing measurements. The dataset exhibits class imbalance, where the number of non-diabetic instances is higher than diabetic instances. Proper preprocessing and resampling techniques are therefore required to ensure balanced model learning.

3.2 Data Preprocessing

Data preprocessing is a critical step in machine learning pipelines, as medical datasets often contain missing values, noisy observations, and inconsistent data distributions. In this study, several preprocessing techniques were applied to improve data quality.

3.2.1 Handling Missing Values

Zero values in key physiological variables were replaced with NAN to represent missing data. Imputation was performed using the median strategy within a preprocessing pipeline to maintain consistency between training and test sets. Specifically, columns (Glucose, BloodPressure, SkinThickness, Insulin, BMI) were treated as missing values and imputed with median values which is robust to outliers and preserves the distribution of the data [17].

3.2.2 Feature Engineering and Dimensionality Reduction

Feature engineering was guided by a domain-informed transformation strategy, focusing on capturing nonlinear relationships and interactions between clinical variables. Specifically, three types of transformations were applied: ratio-based scaling, interaction modeling, and threshold-based discretization. The first feature, *Glucose_per_BMI*, is a ratio capturing relative glucose levels adjusted for body composition:

$$\text{Glucose\_per\_BMI} = \frac{\text{Glucose}}{\text{BMI}+\epsilon} \tag{1}$$

where ϵ is a small constant to avoid division by zero.The second feature, *Age_BMI*, models the interaction between age and BMI:

$$\text{Age\_BMI} = \text{Age} \times \text{BMI} \tag{2}$$

This captures the compounded effect of aging and obesity on diabetes risk. The third feature, *HighPreg*, is a binary variable derived through threshold-based discretization:

$$\text{HighPreg} = \begin{cases} 1 & if\ Pregnancies > 2 \\ 0 & otherwise \end{cases} \tag{3}$$

This transformation reflects increased diabetes risk associated with higher pregnancy counts while reducing noise and improving interpretability. Overall, these transformations enhance feature representation by improving class separability and enabling models to capture nonlinear dependencies. Following feature engineering, Principal Component Analysis (PCA) was applied to reduce dimensionality, eliminate redundancy, and improve model generalization. The transformation is expressed as:

$$Z = XW \tag{4}$$

where

X is the standardized data matrix,
W represents the eigenvector matrix, and
Z denotes the transformed feature space.

By retaining the principal components with the highest variance, PCA helps reduce noise and multicollinearity in the dataset.

3.2.3 Outlier Handling

Medical datasets frequently contain extreme observations due to measurement errors or rare physiological conditions. To begin with, all records that had zero (0) values for Blood pressure, BMI and Glocose were discarded before any other preprocessing was done. To mitigate the impact of extreme physiological observations, the IQR-based winsorization technique was applied. Winsorization limits extreme values by replacing them with percentile-based threshold values. In this study, values below Q1 - 1.5*IQR or above Q3 +

1.5*IQR were clipped to the respective thresholds accordingly. This approach reduces the influence of extreme observations while retaining the overall data distribution. IQR is calculated as shown in equation 1.

$$IQR=Q3-Q1 \tag{5}$$

where

Q3 is the 75 percentile

Q1 is the 25 percentile

3.2.4 Feature Transformation and Selection

Polynomial features (degree=2) and PowerTransformer were applied to enhance linear separability [18]. Numerical features were imputed and scaled using StandardScaler. Mutual information-based feature selection (SelectKBest, k=8) was applied, followed by PCA to retain 95% variance [14]. This was aimed at reducing overfitting, training time,noise and tree splits for tree based classifiers.

3.2.5 Feature Scaling

Since machine learning algorithms are sensitive to the scale of input features, standardization was applied to normalize the features. Standardization transforms the data such that each feature has a mean of zero and a standard deviation of one. The standardized feature value z is computed as:

$$z = \frac{xi-\mu}{\sigma} \tag{6}$$

Where:
x_i represents the original feature value,
μ denotes the mean of the feature, and
σ represents the standard deviation.

This transformation ensures that all features contribute equally during model training.

3.4 Handling Class Imbalance

The dataset exhibits moderate class imbalance (268 positive cases vs. 500 negative cases). To address this, we adopted a hybrid resampling strategy combining SMOTE and Tomek links, selected for its ability to both increase minority class representation and improve class boundary quality. SMOTE was applied to generate synthetic minority class samples by interpolating between existing instances and their nearest neighbors. This approach was preferred over simple oversampling because it reduces overfitting by introducing new, plausible samples rather than duplicating existing ones. However, SMOTE alone may introduce noisy or overlapping samples near class boundaries. Given a minority class sample x_i a new synthetic sample is generated as:

$$x_{new} = x_i + \lambda(x_{nn} - x_i) \tag{7}$$

where

x_{nn} represents a randomly selected nearest neighbor, and λ is a random value between 0 and 1. This method improves class balance and allows the classifier to learn decision boundaries more effectively.

3.5 Model Development

Four machine learning models were selected based on a representative modeling strategy, aimed at capturing diverse learning behaviors and ensuring a comprehensive evaluation of predictive performance. Rather than selecting models arbitrarily, the study includes algorithms from different methodological categories: linear, kernel-based, tree-based, and ensemble learners. Specifically, Logistic Regression was chosen as a baseline linear model due to its interpretability and widespread use in medical prediction tasks. Support Vector Machines (SVM) were included to capture nonlinear decision boundaries through kernel functions. Random Forest was selected as a tree-based ensemble method, capable of modeling complex feature interactions and handling noisy data. Finally, Gradient Boosting-based models (e.g., XGBoost or LightGBM) were included due to their state-

of-the-art performance in structured/tabular datasets and their ability to iteratively improve weak learners. This selection ensures a balanced comparison across fundamentally different algorithmic paradigms, allowing the study to assess how preprocessing, feature engineering, and class imbalance handling affect models with varying assumptions and learning capacities. Additionally, these models have been consistently reported in prior studies like [2], [8] enabling meaningful benchmarking while extending the analysis through a unified and optimized pipeline. The four were evaluated to determine the most effective model for diabetes prediction.

i.Logistic Regression

Logistic regression (LR) models the probability that a patient belongs to the diabetic class. The probability function is defined as:

$$P(y = 1 \mid x) = \frac{1}{1+e^{-(\beta 0+\beta x)}} \quad (8)$$

where

β0 represents the intercept and
β denotes the feature coefficients.

ii.Random Forest

Random Forest (RF) is an ensemble learning algorithm that constructs multiple decision trees during training and combines their predictions using majority voting. This approach reduces variance and improves generalization.

iii.XGBoost

Extreme Gradient Boosting (XGBoost) is a powerful ensemble method that builds sequential decision trees using gradient boosting optimization. Each new tree attempts to correct the errors made by previous trees.

iv.LightGBM

Light Gradient Boosting Machine (LightGBM) is an efficient gradient boosting framework that uses histogram-based learning and leaf-wise tree growth. It is particularly suitable for large datasets and provides high predictive accuracy with reduced training time.

Additionally, a Stacked Ensemble used LR, RF, and XGB as base learners with LR as a meta-learner. Hyperparameters were optimized empirically. Each model was integrated into an imbalanced-learn pipeline encompassing feature engineering, outlier handling, transformation, and SMOTE-Tomek resampling.

3.6 Model Evaluation

To evaluate the predictive performance of the models, 5-fold cross-validation was used in combination with a 20% hold-out test set. The dataset was divided into five subsets. During each iteration, four subsets were used for training while the remaining subset was used for validation. This process was repeated five times to ensure robust performance estimates. The final model performance was evaluated on the unseen test dataset.

3.7 Evaluation Metrics

Several evaluation metrics were used to assess model performance:

Accuracy

$$\textbf{Accuracy} = \frac{\textbf{TP+TN}}{\textbf{TP+TN+FP+ FN}} \quad (9)$$

Precision

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (10)$$

Recall

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (11)$$

F1 Score $$\mathbf{F1} = 2x\frac{\mathbf{Precision\ x\ Recall}}{\mathbf{Precision+Recall}}$$ (12)

Specificity $$\frac{\mathbf{TN}}{\mathbf{TN+FP}}$$ (13)

Balanced Accuracy $$\frac{\mathbf{Specificity\ +\ Recall}}{\mathbf{2}}$$ (14)

ROC-AUC**:** The Receiver Operating Characteristic (ROC) curve measures the trade-off between sensitivity and specificity, while the Area Under the Curve (AUC) summarizes the model's classification ability across different thresholds.

PR-AUC: The Precision-Recall Area Under the Curve computes the area under the curve plotting Precision vs. Recall across various probability thresholds

Confusion Matrix: This tool was used for visualizing true positives, false positives, etc.

3.8 Experimental Setup

All experiments were implemented in the Anaconda data analysis platform. The platform provides open-source programming languages like Python for large-scale data processing, predictive analytics, and scientific computing [19], [20], [21]. The experiments were run on a computing machine with a 3.0 GHz processor and 8 GB RAM.The primary libraries used include: Scikit-learn for machine learning algorithms, XGBoost for gradient boosting models, LightGBM for efficient tree-based learning,Pandas and NumPy for data manipulation, Matplotlib and Seaborn for visualization.

3.9 Hyperparameters

The dataset was split with a ratio of 70:30, where 70% of the data was employed for training the algorithms and 30% was used to Test and validate their efficacy [22]. In each of them, evaluation was based on the hyper parameters described in Table 2.

Table 2: Model hyper parameters

Algorithm	Hyper parameters
Logistic regression(LR)	max_iter=1000, solver="liblinear", random_state=42
Support Vector Machine(SVM)	probability=True, kernel='linear', random_state=42
Random Forest (RF)	n_estimators=100, random_state=42
XGBoost(XGB)	use_label_encoder=False, eval_metric='logloss', random_state=42
LightGBM(LGBM)	random_state=42
Stacked Ensemble	-

4. Results

This section presents the results obtained from the implementation of the machine learning pipeline described in the previous section. The evaluation focuses on assessing the performance of the developed models using appropriate metrics, including accuracy, precision, recall, F1-score, and, balanced accuracy, ROC-AUC, PR-AUC. Comparisons are also made between different models to determine the most effective approach for the given dataset.

4.1. Baseline Model Performance

This subsection presents the performance of baseline models trained on the raw dataset without any preprocessing or optimization techniques. A summary of model performances is as shown in table 3.

Table 3: Baseline model performance with raw data

Model	Accuracy	Precision	Recall	F1_score	Bal_accuracy	ROC-AUC	PR-AUC
LR	0.7359	0.6724	0.4815	0.5612	0.6774	0.8297	0.7335
SVM	0.7446	0.7200	0.4444	0.5496	0.6756	0.8212	0.6872
RF	0.7446	0.6719	0.5309	0.5931	0.6954	0.8268	0.7350
XGB	0.7229	0.6232	0.5309	0.5733	0.6788	0.8049	0.6702
LGBM	0.7359	0.6429	0.5556	0.5960	0.6944	0.8027	0.6720
Stacked Ensemble	0.7403	0.6667	0.5185	0.5833	0.6893	0.8383	0.7409

4.2 Feature Engineering

This subsection evaluates the effect of feature engineering techniques, including scaling, feature creation, and outlier removal on model performance. Table 4 provides a summary of model performances before and after feature engineering.

Table 4: Model performance with feature engineering

	Model	Accuracy	Precision	Recall	F1_score	Bal_accuracy	ROC-AUC	PR-AUC
Before	LR	0.8073	0.7705	0.6267	0.6912	0.7644	0.8765	0.7622
	SVM	0.7798	0.7368	0.5600	0.6364	0.7276	0.8499	0.7486
	RF	0.7752	0.7031	0.6000	0.6475	0.7336	0.8534	0.7315
	XGB	0.7706	0.6984	0.5867	0.6377	0.7269	0.8115	0.6707
	LGBM	0.7569	0.6833	0.5467	0.6074	0.7069	0.8344	0.7029
	Stacked Ensemble	0.8073	0.7895	0.600	0.6818	0.7580	0.8736	0.7507
After	LR	0.8073	0.7705	0.6267	0.6912	0.7644	0.8781	0.7667
	SVM	0.7844	0.7593	0.5467	0.6357	0.7279	0.8298	0.7222
	RF	0.7798	0.7368	0.5600	0.6364	0.7276	0.8629	0.7337
	XGB	0.7752	0.6970	0.6133	0.6525	0.7367	0.8176	0.6970
	LGBM	0.7798	0.7143	0.6000	0.6522	0.7371	0.8276	0.7266
	Stacked Ensemble	0.8211	0.8000	0.6400	0.7111	0.7780	0.8769	0.7616

The application of feature engineering techniques resulted in consistent improvements across most evaluation metrics, including accuracy, precision, F1-score, and balanced accuracy. These improvements indicate that the transformed feature space enhanced the models' ability to learn meaningful patterns from the data. However, the impact on recall was relatively modest, as feature engineering does not directly address class imbalance. Recall showed a substantial improvement across the models, particularly for Logistic Regression (**0.6267** to **0.8000**) and SVM (**0.5467** to **0.7467**), with a moderate increase observed in LightGBM (**0.6000** to **0.6667**). This

indicates that the models became significantly more effective at correctly identifying positive (minority class) instances, demonstrating the effectiveness of the applied technique in enhancing sensitivity to the minority class. The stacking ensemble model demonstrated the greatest benefit, achieving the highest performance across multiple metrics. It enhanced model learning without introducing major trade-offs, leading to more stable and slightly improved performance. Overall, the results suggest that feature engineering contributes to improved model stability and predictive performance, without introducing major trade-offs therefore serving as an essential step in the machine learning pipeline.

4.2 Feature Engineering with SMOTE-TOMEK

Table 5: Model performance with feature engineering and SMOTE

	Model	Accuracy	Precision	Recall	F1_score	Bal_accuracy	ROC-AUC	PR-AUC
Before SMOTE- Tomek	LR	0.8073	0.7705	0.6267	0.6912	0.7644	0.8781	0.7667
	SVM	0.7844	0.7593	0.5467	0.6357	0.7279	0.8298	0.7222
	RF	0.7798	0.7368	0.5600	0.6364	0.7276	0.8629	0.7337
	XGB	0.7752	0.6970	0.6133	0.6525	0.7367	0.8176	0.697
	LGBM	0.7798	0.7143	0.6000	0.6522	0.7371	0.8276	0.7266
	Stacked Ensemble	0.8211	0.8000	0.6400	0.7111	0.7780	0.8769	0.7616
After SMOTE- Tomek	LR	0.7706	0.6316	0.8000	0.7059	0.7776	0.8776	0.7630
	SVM	0.7706	0.6437	0.7467	0.6914	0.7649	0.8462	0.7097
	RF	0.7661	0.6667	0.6400	0.6531	0.7361	0.8642	0.7464
	XGB	0.7477	0.6250	0.6667	0.6452	0.7284	0.8154	0.6858
	LGBM	0.7615	0.6494	0.6667	0.6579	0.7389	0.8381	0.7138
	Stacked Ensemble	0.7706	0.6667	0.6667	0.6667	0.7459	0.8600	0.7432

The application of SMOTE- TOMEK resulted in a significant improvement in recall across all models, indicating enhanced detection of minority class instances. Accuracy showed a slight improvement across several models, increasing from (**0.7752 to 0.7798)** for Random Forest, (**0.7798 to 0.7844)** for SVM, and (**0.8073 to 0.8211)** for the Stacked Ensemble, indicating a modest enhancement in overall predictive performance after feature engineering. Although a reduction in accuracy and precision was observed, this trade-off is expected in imbalanced classification problems. The F1-score and balanced accuracy demonstrated overall improvement, confirming a better balance between sensitivity and specificity. Furthermore, the stability of ROC-AUC and PR-AUC suggests that the discriminative ability of the models was preserved as shown in figure 2 and figure 3 respectively. Among the evaluated models, Logistic Regression showed the most notable improvement, while the stacking ensemble exhibited limited gains, likely due to sensitivity to noise introduced during resampling.

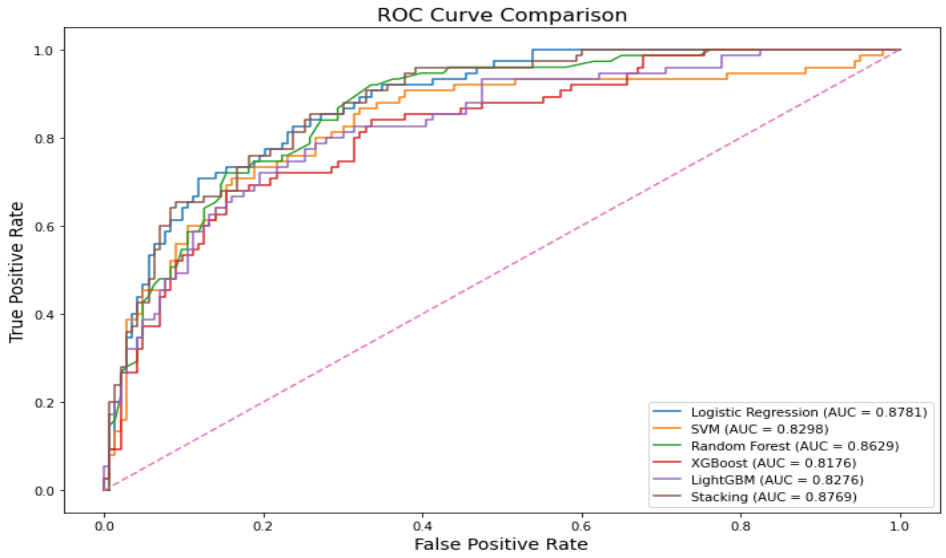

Figure 2: ROC curve before feature Engineering

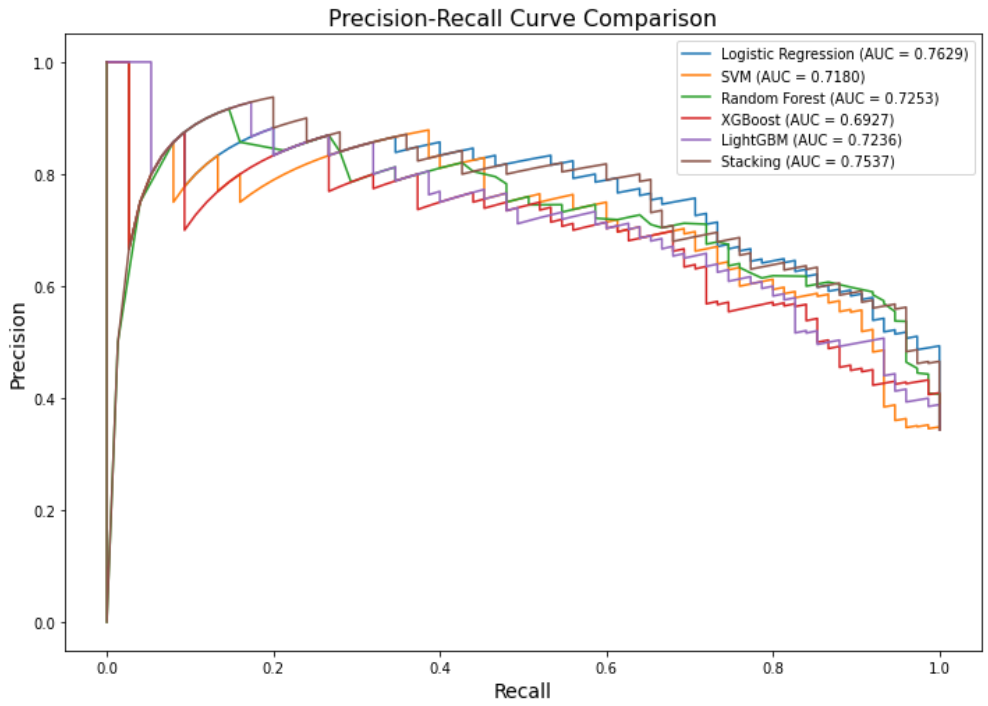

Figure 3: Precision recall curve before feature engineering

4.2 Feature Importance and Selection

The objective was to evaluate the contribution of both original and engineered features in predicting diabetes outcomes. Feature importance analysis was conducted using permutation importance to provide a model-agnostic assessment of predictor relevance. Table 5 provides a detailed feature importance ranking performance across all the models. The results reveal a strong consistency in feature importance rankings across all models. In particular, Glucose emerged as the most influential predictor in every model. This aligns with clinical knowledge, as glucose concentration is a primary indicator in diabetes diagnosis. Additionally, the engineered feature Age_BMI consistently ranked as the second most important feature across all models, demonstrating that combining age and body mass index provides enhanced predictive power compared to using these variables independently. The consistency of feature rankings across ensemble models such as Random Forest, XGBoost,

and LightGBM further validates the robustness of these predictors. Tree-based models (RF, XGB, LGBM) and linear models (LR, SVM) exhibited similar ranking patterns, reinforcing the robustness of the identified important features. However, differences in importance magnitude were observed due to variations in how models compute importance scores. The Stacking model, evaluated using permutation importance, confirmed that Glucose and Age_BMI remain dominant predictors. Notably, some features such as Insulin and Age exhibited slightly negative importance values, suggesting that they may introduce noise or redundancy in the ensemble model. These findings align with the observed improvements in classification performance, particularly in Recall and F1-score.

Table 6: Feature Importance Ranking Performance

Feature	LR	SVM	RF	XGB	LGBM	Stacked
Glucose	1.929320	1.437265	0.200559	0.246556	368.4	0.081610
Age_BMI	0.673944	0.921085	0.167051	0.139607	442.0	0.029852
Glucose_per_BMI	0.641912	0.448590	0.098776	0.063249	297.6	0.007193
BMI	0.152744	0.275754	0.114765	0.083850	343.4	0.006531
Age	0.288022	0.485040	0.094620	0.077167	188.6	-0.001756
Pregnancies	0.561481	0.452660	0.042336	—	161.6	-0.000383
DiabetesPedigreeFunction	0.327471	0.257646	0.085226	0.067522	422.0	0.003736
HighPreg	0.308298	0.282472	—	0.069704	—	-0.001513
Insulin	0.131626	0.095143	0.071890	0.081448	165.4	-0.003309
BloodPressure	0.124527	0.105306	0.053921	0.055851	231.6	0.001530
SkinThickness	—	—	0.062155	0.063649	202.4	—

Note: A dash (—) indicates that the feature was not among the top-ranked variables for the respective model. Feature importance values are model-specific and therefore not directly comparable across different models. For the LightGBM model, importance values are based on split/gain metrics and are not normalized. For the stacking model, feature importance was computed using permutation importance, representing the change in model performance when a feature is randomly shuffled.

4.3 Statistical significance

A Wilcoxon signed-rank test was conducted to compare model performance across different pipeline stages, including baseline, feature engineering, and feature engineering with SMOTE-Tomek. The results indicated that the application of the full pipeline led to statistically significant improvements in F1-score and recall for several models ($p < 0.05$). The Friedman test was conducted to determine whether there are statistically significant differences in model performance across the three pipelines (Baseline, Feature Engineering, and Feature Engineering + SMOTE-Tomek) using F1-score, Recall, and ROC-AUC as evaluation metrics. The overall results indicate that:

i. **F1-score**:A statistically significant difference was observed ($\chi^2 = 6.3333$, $p = 0.0421$). This suggests that at least one pipeline configuration leads to a meaningful improvement or degradation in F1-score across the evaluated models.

ii. **Recall**:A stronger statistically significant difference was found ($\chi^2 = 9.0$, $p = 0.0111$). This indicates that the pipeline transformations, particularly techniques such as class balancing, have a notable effect on the models' ability to correctly identify positive cases.

iii. **ROC-AUC**:No statistically significant difference was observed ($\chi^2 = 4.3333, p = 0.1146$).This implies that, despite variations in preprocessing and resampling, the overall ranking ability of the models (i.e., discrimination between classes) remains relatively stable across pipelines.

iv. **ROC-AUC** :No statistically significant difference was observed ($\chi^2 = 5.3333$, $p = 0.0695$).This implies that there is no statistically significant difference in precision-recall performance across the evaluated pipelines at the 5% significance level. However, the p-value is close to the threshold, suggesting a marginal trend toward significance. This implies that while preprocessing and resampling techniques may influence PR-AUC, their effect is not consistent across all models. The findings further suggest that the applied pipeline improvements primarily enhance recall and F1-score rather than significantly improving the precision-recall trade-off.

4.4 Model comparison with literature

Table 7: Model performance of similar studies

Model performance for similar studies on PIMA Indian Diabetes dataset are as described in table 7.

Study	Dataset	Model	Accuracy
Hama Saeed 2023	PIMA	SVM	0.78
Rafie et al. 2025	PIMA	XGBoost	0.82
Our Study	PIMA	LightGBM	0.82.11

5. Results and Discussion

The experimental analysis demonstrates that the applied preprocessing and resampling strategies exert a statistically significant influence on classification performance, particularly with respect to Recall and F1-score. In contrast, the effect on ROC-AUC was not significant, suggesting that the improvements are primarily realized in class-specific predictive performance rather than overall model discrimination. Furthermore, the Friedman test revealed significant differences among the evaluated pipeline configurations, providing evidence for the efficacy of the proposed preprocessing and feature engineering approaches.

Key observations from the study are as follows:

i. **Feature Engineering:** The introduction of interaction terms and domain-specific features improved model sensitivity, enabling more accurate identification of minority class instances.

ii. **Outlier Handling:** Winsorization of extreme values enhanced model robustness by stabilizing the influence of outliers.

iii. **Nonlinear Transformations:** Polynomial and power transformations facilitated the capture of complex interactions by tree-based models, although careful parameter tuning was necessary to mitigate overfitting risks.

iv. **Feature Selection and Dimensionality Reduction:** Feature selection based on mutual information, combined with Principal Component Analysis (PCA), reduced multicollinearity and improved generalization across models.

v. **Class Imbalance Correction:** The SMOTE-Tomek resampling strategy effectively increased recall for the minority class while maintaining precision, highlighting its utility in imbalanced clinical datasets.

vi. **Ensemble Learning:** Stacked ensembles generally enhanced predictive stability; however, LightGBM alone outperformed the stacked configuration, indicating its inherent ability to model nonlinear relationships efficiently.

5.1 Limitations

Despite the observed improvements, several limitations should be acknowledged:

i. **Dataset Size:** The Pima Indians Diabetes dataset is relatively modest (768 instances), which may constrain the generalizability of the findings to broader populations.

ii. **Feature Scope:** The study focused exclusively on structured clinical features, excluding lifestyle, behavioral, and genomic factors that may further influence diabetes risk.

iii. **Hyperparameter Optimization:** Hyperparameters were tuned empirically; more rigorous optimization techniques, such as Bayesian or grid search, could potentially enhance predictive performance and robustness.

Overall, the findings demonstrate that integrating feature engineering, robust preprocessing, resampling, and ensemble learning can substantially improve sensitivity and F1-score in diabetes prediction tasks, particularly for imbalanced datasets, while maintaining stable overall model performance.

6. Conclusion

Our comprehensive pipeline demonstrates that the applied preprocessing and resampling strategies have a statistically significant impact on classification performance, especially for Recall and F1-score. However, their effect on ROC-AUC is not significant, suggesting that improvements are primarily realized in class-specific predictive performance rather than overall model discrimination. Furthermore, the Friedman test revealed a significant overall difference among the evaluated pipeline configurations, confirming the effectiveness of the proposed preprocessing techniques.

Future Work

Future enhancements of the proposed pipeline include exploring deep learning architectures, such as neural networks and convolutional models, for capturing complex nonlinear relationships in patient data. Additionally, integrating longitudinal patient records and multi-modal data, including genomic and lifestyle features, can enhance predictive accuracy and clinical relevance. Advanced hyperparameter optimization strategies, such as Bayesian optimization or genetic algorithms, can be employed to automatically fine-tune model parameters. Further, deploying explainable AI techniques (SHAP, LIME) will improve interpretability and trust in clinical settings.

Reproducibility and Implementation

The entire pipeline is implemented in Python using scikit-learn, XGBoost, LightGBM, and imbalanced-learn libraries. All steps, from feature engineering to model evaluation, are modular and reusable. Researchers can replicate the study by downloading the PIMA Indian Diabetes dataset and running the pipeline with provided hyperparameters. The inclusion of cross-validation and multiple metrics ensures reliable performance assessment.

Implications for Clinical Practice

The pipeline demonstrates that ML models, when properly engineered and tuned, can serve as decision support tools for early diabetes detection. Clinical adoption can lead to improved patient outcomes through timely interventions. Importantly, the use of interpretable features and feature importance plots facilitates understanding of model decisions by healthcare practitioners.

Ethical Considerations

All analyses are performed on publicly available, anonymized datasets, ensuring patient privacy. Deployment of ML models in healthcare must consider fairness, bias, and equity. Models should be evaluated on diverse populations to prevent health disparities. Clinical decisions should remain under the oversight of qualified healthcare professionals.

Data availability

The dataset is readily available from the following repository. https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

Acknowledgments
We acknowledge the PIMA Indian Diabetes dataset contributors and the open-source Python ML libraries, which facilitated reproducible research.

Conflict of Interest Statement
The authors declare no conflict of interest.

Funding
This study did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] IDF, "Facts & figures," International Diabetes Federation. Accessed: Mar. 30, 2026. [Online]. Available: https://idf.org/about-diabetes/diabetes-facts-figures/

[2] S. M. Ganie, P. K. D. Pramanik, M. Bashir Malik, S. Mallik, and H. Qin, "An ensemble learning approach for diabetes prediction using boosting techniques," *Front. Genet.*, vol. 14, Oct. 2023, doi: 10.3389/fgene.2023.1252159.

[3] E. Almutairi, M. Abbod, and Z. Hunaiti, "Prediction of Diabetes Using Statistical and Machine Learning Modelling Techniques," *Algorithms*, vol. 18, no. 3, Mar. 2025, doi: 10.3390/a18030145.

[4] M. S. Alzboon, M. Al-Batah, M. Alqaraleh, A. Abuashour, and A. F. Bader, "A Comparative Study of Machine Learning Techniques for Early Prediction of Prostate Cancer," in *2023 IEEE Tenth International Conference on Communications and Networking (ComNet)*, Nov. 2023, pp. 1–12. doi: 10.1109/ComNet60156.2023.10366703.

[5] I. Murere, B. Ndlovu, S. Dube, M. Muduva, and F. Jacqueline Kiwa, "Comparative Analysis of Machine Learning Techniques for Predicting Diabetes," in *Proceedings of the International Conference on Industrial Engineering and Operations Management*, Augsburg (Greater Munich), Germany: IEOM Society International, Jul. 2024. doi: 10.46254/EU07.20240073.

[6] K. Abnoosian, R. Farnoosh, and M. H. Behzadi, "Prediction of diabetes disease using an ensemble of machine learning multi-classifier models," *BMC Bioinformatics*, vol. 24, no. 1, p. 337, Sep. 2023, doi: 10.1186/s12859-023-05465-z.

[7] Z. Rafie, M. S. Talab, B. E. Z. Koor, A. Garavand, C. Salehnasab, and M. Ghaderzadeh, "Leveraging XGBoost and explainable AI for accurate prediction of type 2 diabetes," *BMC Public Health*, vol. 25, no. 1, p. 3688, Oct. 2025, doi: 10.1186/s12889-025-24953-w.

[8] M. A. Hama Saeed, "Diabetes type 2 classification using machine learning algorithms with up-sampling technique," *J. Electr. Syst. Inf. Technol.*, vol. 10, p. 8, Dec. 2023, doi: 10.1186/s43067-023-00074-5.

[9] M. Mujahid *et al.*, "Data oversampling and imbalanced datasets: an investigation of performance for machine learning and feature engineering," *J. Big Data*, vol. 11, no. 1, p. 87, Jun. 2024, doi: 10.1186/s40537-024-00943-4.

[10] J. Abdollahi, B. NouriMoghaddam, and A. MIRZAEI, *Diabetes Data Classification using Deep Learning Approach and Feature Selection based on Genetic*. 2023. doi: 10.21203/rs.3.rs-2855804/v1.

[11] M. A. Bouqentar *et al.*, "Early heart disease prediction using feature engineering and machine learning algorithms," *Heliyon*, vol. 10, no. 19, p. e38731, Oct. 2024, doi: 10.1016/j.heliyon.2024.e38731.

[12] A. Raza, H. U. R. Siddiqui, K. Munir, M. Almutairi, F. Rustam, and I. Ashraf, "Ensemble learning-based feature engineering to analyze maternal health during pregnancy and health risk prediction," *PLOS ONE*, vol. 17, no. 11, p. e0276525, Nov. 2022, doi: 10.1371/journal.pone.0276525.

[13] S. Khan *et al.*, "Advanced Feature Selection Techniques in Medical Imaging—A Systematic Literature Review," *Comput. Mater. Contin.*, vol. 0, no. 0, pp. 1–10, 2025, doi: 10.32604/cmc.2025.066932.

[14] A. Tukhtaev, D. Turimov, J. Kim, and W. Kim, "Feature Selection and Machine Learning Approaches for Detecting Sarcopenia Through Predictive Modeling," *Mathematics*, vol. 13, no. 1, p. 98, Jan. 2025, doi: 10.3390/math13010098.

[15] L. Chen *et al.*, "Dual-radiomics based on SHapley additive explanations for predicting hematologic toxicity in concurrent chemoradiotherapy patients," *Discov. Oncol.*, vol. 16, Apr. 2025, doi: 10.1007/s12672-025-02336-2.

[16] H. Yuan, K. Yu, F. Xie, M. Liu, and S. Sun, "Automated machine learning with interpretation: A systematic review of methodologies and applications in healthcare," *Med. Adv.*, vol. 2, pp. 205–237, Aug. 2024, doi: 10.1002/med4.75.

[17] S. T. H. Rizvi, M. Y. Latif, M. S. Amin, A. J. Telmoudi, and N. A. Shah, "Analysis of Machine Learning Based Imputation of Missing Data," *Cybern. Syst.*, vol. 56, no. 6, pp. 818–832, Aug. 2025, doi: 10.1080/01969722.2023.2247257.

[18] A. Abu-Shareha, M. M. Abualhaj, M. A. Alsharaiah, A. Al-Saaidah, and A. Achuthan, "Diabetes Prediction Through Classification Using Pima Dataset: Survey and Evaluation," *J. Soft Comput. Data Min.*, vol. 6, no. 1, pp. 1–20, Jun. 2025.

[19] A. Kadiyala and A. Kumar, "Applications of Python to evaluate environmental data science problems," *Environ. Prog. Sustain. Energy*, vol. 36, no. 6, pp. 1580–1586, 2017, doi: 10.1002/ep.12786.

[20] D. Rolon-Mérette, M. Ross, T. Rolon-Mérette, and K. Church, "Introduction to Anaconda and Python: Installation and setup," *Quant. Methods Psychol.*, vol. 16, no. 5, pp. S3–S11, May 2020, doi: 10.20982/tqmp.16.5.S003.

[21] A. K. Singh and A. K. Jain, "Automating Monte Carlo simulation data analysis using Python in Anaconda environment," in *Signal Processing with Python: A practical approach*, IOP Publishing, 2024. doi: 10.1088/978-0-7503-5929-0ch10.

[22] H. Bichri, A. Chergui, and M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, 2024, doi: 10.14569/IJACSA.2024.0150235.