Multiclass Diabetes Detection Using Random Forest Classification

Amjed Al-Mousa
Computer Eng. Dept.
Princess Sumaya University for Technology
Amman, Jordan
a.almousa@psut.edu.jo

Laith AlKhdour
Computer Eng. Dept.
Princess Sumaya University for Technology
Amman, Jordan
LAI20190130@std.psut.edu.jo

Hatem Bishawi
Computer Eng. Dept.
Princess Sumaya University for Technology
Amman, Jordan
HAT20190377@std.psut.edu.jo

Fares AlShubeliat
Computer Eng. Dept.
Princess Sumaya University for Technology
Amman, Jordan
FAR20190137@std.psut.edu.jo

2023 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT) | 979-8-3503-2405-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/JEEIT58638.2023.10185679

***Abstract*— Detecting diabetes at an early stage can help save lives and improve the patients' quality of life significantly. Diabetes can be detected with the assistance of information regarding the patient's lifestyle and health. This work aims to predict diabetic patients using different machine-learning classification algorithms and a dataset about diabetic and healthy patients. The work employs a data balancing technique to handle the data imbalance issue, as well as using cross-validation. In addition, it compares these machine-learning algorithms according to several performance indicators like accuracy, precision, recall, and F1-score. Accordingly, the Random Forest classifier proved to produce the best results with accuracy, precision, recall, and an F1-score, all equal to 89%.**

Keywords*— *Machine Learning; multiclass; Classification; Diabetes; Diabetes Health Indicator Dataset.

I. Introduction

Diabetes is a lifelong health condition that affects the glucose level in the blood to become higher than usual [1]. There are two types of diabetes; type 1 Diabetes and type 2 Diabetes. Type 1 diabetes is a chronic health condition when the pancreas releases little insulin [2]. Type 2 diabetes is an impairment in how the blood uses sugar to fuel the body, resulting in excess sugar circulating in the blood's circulation [3]. According to [4], prediabetes is considered a series health condition due to the high increase in blood sugar levels. This increase, however, is not high enough to be diagnosed as diabetic.

The dataset presented in this paper will used to train and test several Machine Learning (ML) classifiers to detect whether a patient is non-diabetic, pre-diabetic, or diabetic [5]. The dataset was produced by the Center for Disease Control (CDC) [6].

The use of Machine learning techniques in the medical field is growing in different specializations. In [7], several ML classifiers were used in detecting heart diseases; Stochastic Gradient Descent (SGD) classifier, K-nearest neighbor (KNN) classifier, Random Forest Classifier, Logistic Regression Classifier, and Hard voting Ensemble Method. The ML voting ensemble method was found efficient in detecting heart diseases, getting an accuracy of 90%. ML was also used to predict strokes in [8]

In [9], different Machine Learning classification models were used for Diabetes detection. Logistic Regression, Linear Support Vector Machine (SVM), polynomial kernel SVM, voting classifier, Random Forest Classifier, and Linear Discriminant Analysis (LDA) were built and tested for this purpose. Random Forest Classifier performed the best compared to the others, with an 82% accuracy score.

The author in [10] built several models and used the same dataset as this paper via the RapidMiner Auto Model tool. The work used Logistic Regression, Random Forest, SVM, Naïve Bayes, Decision Tree, AdaBoost, KNN, and Multilayer Perceptron, with the X classifier performing the best of them, achieving an 86.61% accuracy score.

It is worth noting that this work presents a different approach in which data was processed to handle the class imbalance in the dataset. Notably, the dataset holds records from 2011 – 2015, which aids in learning long-term trends, seasonal patterns, and forecasting.

II. Data Pre-processing

The dataset used has 21 attributes and the target value of the dataset used. The attributes represented several health conditions related to each patient. The dataset also includes information about each patient's lifestyle and wealth status. Table 1 illustrates the attributes. Before using the dataset, the data must be prepared and cleaned to enhance the data quality and give the best possible results. Note that each attribute's type has been changed to either Boolean or integer, depending on the attribute description.

A. Data duplicates

Data duplicated in the dataset has been removed from the dataset to ensure that no instances are found in both the training and test sets when splitting the dataset, which led to a better and more efficient testing strategy.

B. Feature Engineering

Figure 1 shows the heatmap of the correlation between attributes, which is needed to decide which attribute(s) may be dropped. Highly correlated attributes are the ones targeted in this approach. However, the heatmap shows that the highest correlation between the two attributes is "GenHtlh" and "PhysHlth", with a correlation of 0.52 between them. This led to keeping the attributes without dropping or extracting any attributes. Please note the heatmap was plotted after changing the attribute's type to either Boolean or integer (based on the metadata that describes them). Please note that the diagonal was set to 0.

Table 1: Description of the attributes

Attribute	Range	Description
Diabetes_012	0-2	0 non-diabetic, 1 pre-diabetic, 2 diabetic
HighBP	0/1	Whether the patient has High Blood Pressure or not
HighChol	0/1	Whether the patient has High Blood Pressure or not
CholCheck	0/1	Whether the patient had any cholesterol check in 5 years or not
BMI	15-52	Body mass index
Smoker	0/1	Whether the patient is a smoker or not
Stroke	0/1	Whether the patient had any strokes before or not
HeartDiseaseorAttack	0/1	Whether the patient has coronary heart disease (CHD) or myocardial infarction or not
PhysActivity	0/1	Physical activity in the last 30 days
Fruits	0/1	If the patient consumes fruit 1 or more times per day
Veggies	0/1	If the patient consumes fruit 1 or more times per day
HvyAlcoholConsump	0/1	Men: more than 14 alcoholic drinks per week Women: more than 7 alcoholic drinks per week
AnyHealthcare	0/1	Whether the patient has any kind of health care coverage
NoDocbcCosts	0/1	If there was a need to see the doctor in the past 12 months, but did not happen due to costs
GenHlth	1-5	General health on a scale of 1-5 1 being excellent and 5 being poor
MentHlth	0-30	How many days of mental health problems in the past 30 days
PhysHlth	0-30	How many days of injuries/illness in the past 30 days
DiffWalk	0/1	The serious difficulty of walking or climbing the stairs
Sex	0/1	0 Female, 1 Male
Age	1-13	13- level age category 1 = 18-24 , 13 = 80 or older
Education	1-6	6-level Education level
Income	1-8	1 = less than \$10,000 5 = less than \$35,000 8 = \$75,000 or more

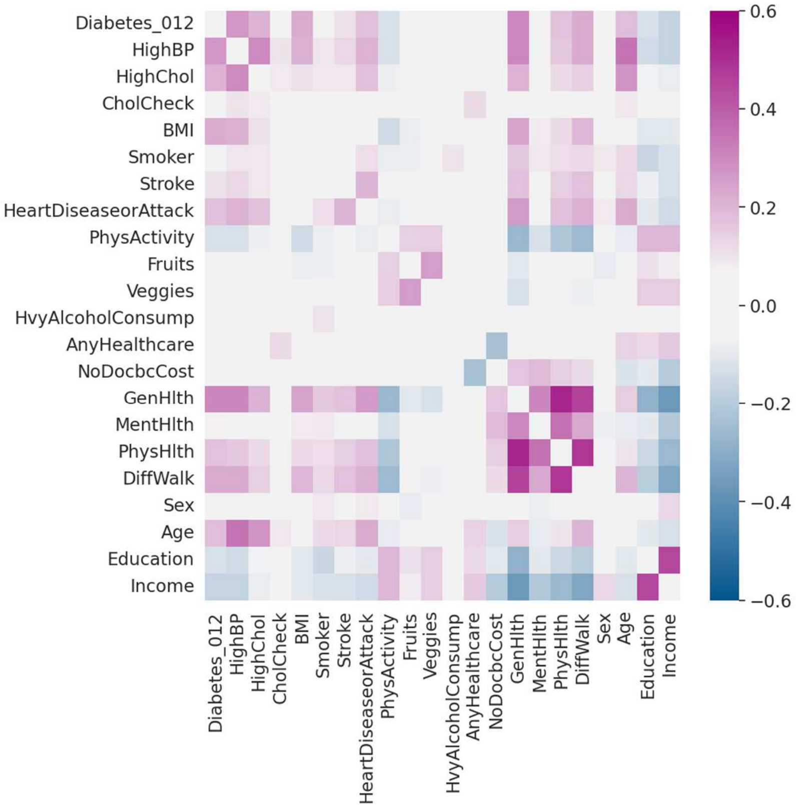

Figure 1: Correlation heatmap

C. Principle Components Analysis

Figure 2 shows the Principle Components Analysis (PCA) of the dataset. Based on the output of the PCA, all the attributes were kept, despite a high correlation between some attributes. PhysHlth and GenHlth are highly correlated, but dropping any of these led to information loss. Please note that StandardScaler was implemented on the attributes, but the StandardScaled attributes were not stored in place. This means that the StandardScaled attributes were only used by the PCA. This step is essential before using PCA to ensure that the attributes contribute equally to the PCA [11].

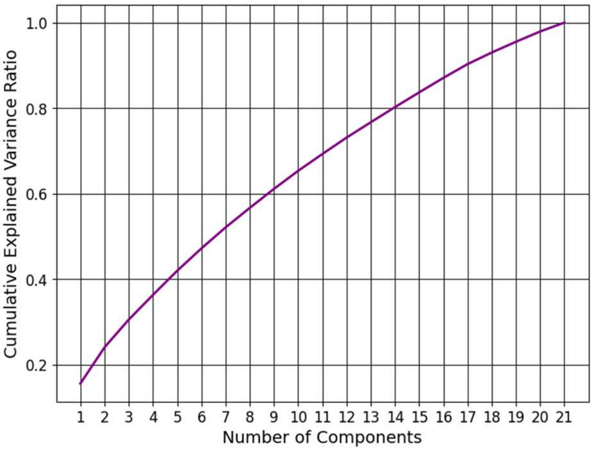

Figure 2: Principle Component Analysis

D. Data Balancing

The dataset is not balanced, with most instances labeled as '0'. Figure 3 shows the value counts of each label before balancing the dataset.

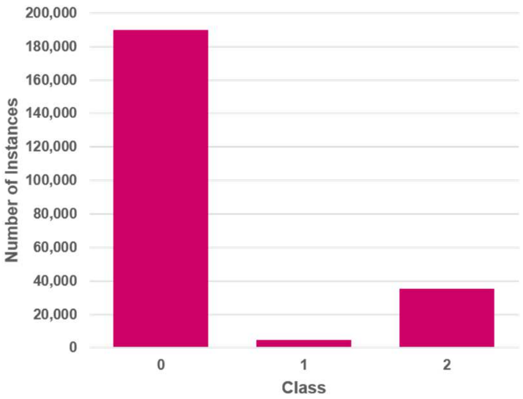

Figure 3: Distribution of instances across classes

To handle the class imbalance issue, Synthetic Minority Oversampling Technique (SMOTE) was used. SMOTE is a balancing technique proposed in a 2002 paper in the Journal of Artificial Intelligence Research [12]. Before splitting the training and testing set, this technique was used to ensure the whole dataset was balanced.

E. MinMaxScaler

Since the attributes have a diverse range of values, the MinMaxScaler approach was used to scale all the attributes to values between 0 and 1, which ensures that the contribution of each attribute to the label is the same.

III. Machine Learning Models

The following classifiers are selected as a diverse group of ML models commonly used in such problems. These classifiers differ in complexity, interpretability, and performance. According to the dataset's characteristics, these classifiers are potential candidates to yield a good model.

A. Decision Tree

A decision tree is a supervised classification algorithm. This algorithm splits data into smaller parts until all instances reach their classes [13].

B. K-Nearest Neighbour (KNN)

KNN is a supervised machine learning classification algorithm that compares the new instance with its neighbors and classifies it as a similar category [14].

C. Random Forest

Random Forest is a classifier that consists of several decision trees, each of which labels a particular instance, and the aggregate of their results is taken [15]. This algorithm is considered one of the most robust classification algorithms in Machine Learning [16].

D. Logistic Regression

Logistic Regression is considered a classification algorithm. This model is usually used in predicting when there are two outcomes (binomial) but can be adjusted to predict more than two outcomes (multinomial) [17]. It works by estimating the probability of an event based on the independent variables (attributes) of the dataset given [18].

E. Stochastic Gradient Descent (SGD) Classifier

SGD classifier is the one that implements a plain SGD learning routine. SGD is an optimization algorithm that finds optimal hyperparameters to minimize cost function [19].

IV. Results and Discussion

This section illustrates and briefly describes the results each model gave. Each model was built with certain parameter initialization and was validated using cross-validation with 10 Stratified k-folds. A cross-validation is an approach that helps test the model's performance on unseen data [20]. Each model was trained on 70% of the dataset and tested on the remaining 30%. Please note that the dataset was balanced before the split, and a stratified split was done. Each confusion matrix was plotted to compare the predicted value with the actual value of the training set. Please note that the dataset is shuffled within the split, and a fixed random-state value was used.

A. Decision Tree

A Decision Tree model was built with a max_depth of 30. On cross-validation, the model gave a mean of 83.7%, with the highest accuracy of approximately 84%. The model gave an accuracy of 84% on the test set. Figure 4 shows the confusion matrix of the model.

The model predicted most pre-diabetic patients scoring a recall of 97% for the label of 1. Around 3800 label 0 instances have been predicted as 1, while around 2700 instances with true label two are predicted as pre-diabetic, which leads to a precision value of 89% for label 1.

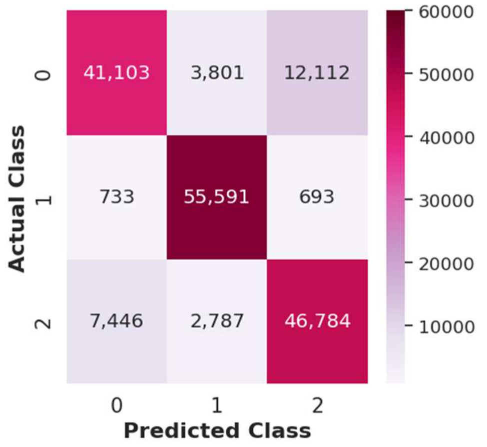

Figure 4: Confusion matrix of Decision Trees

For instances with labels 0, 77% of non-diabetic patients were correctly identified. Around 21% of non-diabetic instances have been predicted as diabetic, while the remaining 2% of the non-diabetic patients are predicted as pre-diabetic. Around 80% of diabetic instances have been correctly classified. Most diabetic patients were correctly classified, leaving around 20% labeled as either 0 or 1.

B. K-Nearest Neighbor

A K-Nearest Neighbor classifier model was built with three n_neighbors, producing a mean accuracy of 81.2% and the highest accuracy of 81.3%. Cross-validation was also implemented using this model, where the model had an accuracy of 82%. Figure 5 shows the confusion matrix of the model.

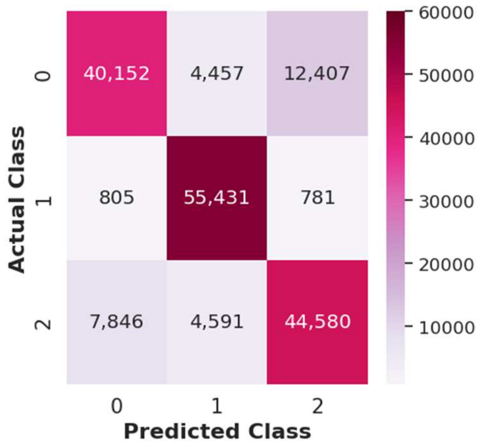

Figure 5: Confusion matrix of K-Nearest Neighbor

The model worked best on the label 1 instance by predicting 97% of them correctly, 78% of label 2 instances were predicted correctly, and around 70% of label 0 instances were classified with their true category.

C. Random Forest

A Random Forest classifier model was built with 500 estimators. This model gave a mean accuracy of 88.8% with cross-validation, giving approximately 89% as the highest score. When tested on the test set, the model scored 89%. Figure 6 shows the confusion matrix on the test set result.

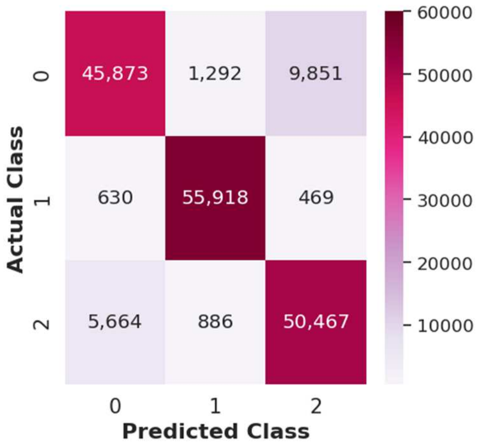

Figure 6: Confusion matrix of Random Forest

The model worked best on the label 1 instance. The model correctly classified 98% of label 1 instances, while the remaining were classified as 0 or 2. Around 88% of label 2 instances were correctly classified, while 81% of label 0 was correctly classified to its group.

D. Logistic Regression

A Logistic Regression model was built with a multinomial multi_class and a max iteration of 1000000. On cross-validation, the mean accuracy of the model is 58.5%, with the highest accuracy found to be 58.9%. The model had an accuracy of 58% on the test set. Figure 7 shows the confusion matrix of the model.

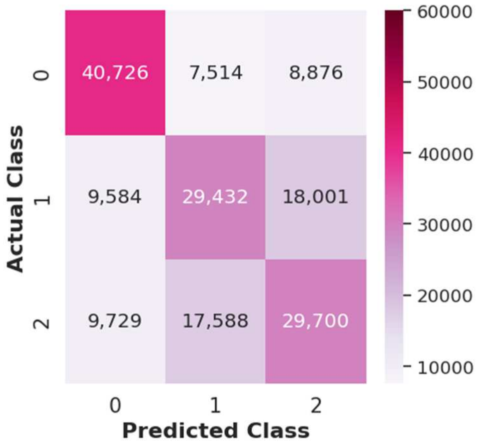

Figure 7: Confusion matrix of Logistic Regression

The model worked best on the label 0 instances by predicting 71% of them correctly, while 52% of label 1 and label 2 instances were each classified correctly.

E. Stochastic Gradient Descent Classifier

A Stochastic Gradient Descent (SGD) classifier model was built with 0.01 alpha and “optimal” learning_rate. Cross-validation gave a mean accuracy of 53.9%, with the highest accuracy being 54.2%. The model scored an accuracy of 54% on the test set.

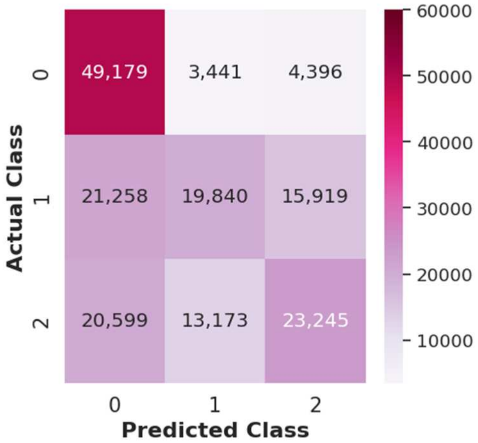

Figure 8: Confusion matrix of Stochastic Gradient Descent

Figure 8 shows the confusion matrix of the model. The model could predict 86% of the label 0 instances correctly, while it predicted 41% of label 2 instances and only 35% of label 1 instances correctly.

Figure 9 and Table 2 summarize the scores achieved by different models. Each model has given the same value for precision and recall based on the results given by the classification report. Since the dataset was balanced before using it on the models, the marco average and weighted average of recall and precision for each model were the same. The f1-score of each model showed the same value as the precision and recall, except for the SGD classifier. This could be due to the classification report's way of calculating the scores [21].

Table 2: Summary of the classifiers' performance indicators

Model	Accuracy w/ 10-fold CV	Accuracy	Precision	Recall	F1-score
DT	83.7%	84%	84%	84%	84%
KNN	81.2%	82%	82%	82%	82%
RF	88.8%	89%	89%	89%	89%
LR	58.5%	58%	58%	58%	58%
SGD	53.9%	54%	54%	54%	52%

V. Conclusion

Random forest showed the best results in accuracy, precision, recall, and f1-score, scoring 89% accuracy in each. Both SGD and logistic Regression showed poor results. Despite performing well in predicting the instances labeled 0, both classifiers showed catastrophic results when dealing with instances labeled 1 and 2, mispredicting many of these instances. KNN and Decision Tree classifiers gave satisfactory results, with 82% and 84% accuracy, respectively. Notice that on cross-validation, the accuracy results were close to that on the test set. One of the possible reasons is the balancing of the dataset.

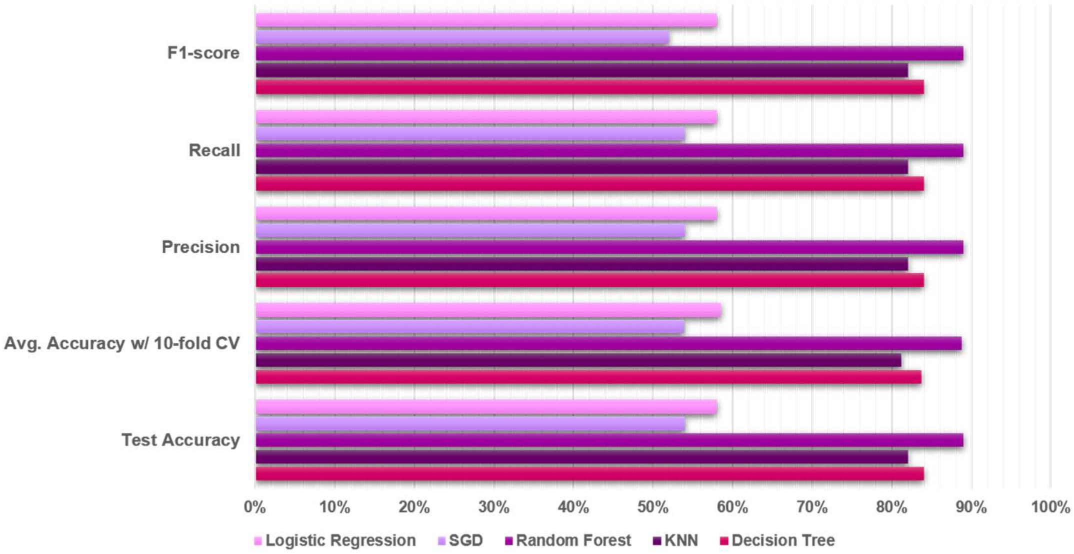

Figure 9: Performance metrics of all classifiers

VI. References

[1] "Diabetes," MayoClinic, [Online]. Available: https://www.mayoclinic.org/diseases-conditions/diabetes/symptoms-causes/syc-20371444.

[2] "Type 1 diabetes," MayoClinic, [Online]. Available: https://www.mayoclinic.org/diseases-conditions/type-1-diabetes/symptoms-causes/syc-20353011.

[3] "What is type 2 diabetes?," NHS, [Online]. Available: https://www.nhs.uk/conditions/type-2-diabetes/.

[4] "Prediabetes – Your Chance to Prevent Type 2 Diabetes," Centers for Disease Control and Prevention, [Online]. Available: https://www.cdc.gov/diabetes/basics/prediabetes.html#:~:text=Prediabetes%20is%20a%20serious%20health,1%20in%203%E2%80%94have%20prediabetes..

[5] "Diabetes Health Indicators Dataset," Kaggle, [Online]. Available: https://www.kaggle.com/code/ratchaphonp/diabetes-health-indicators-dataset/data?select=diabetes_012_health_indicators_BRFSS2015.csv.

[6] "Behavioral Risk Factor Surveillance System," Kaggle, [Online]. Available: https://www.kaggle.com/datasets/cdc/behavioral-risk-factor-surveillance-system.

[7] R. Atallah and A. Al-Mousa, "Heart Disease Detection Using Machine Learning Majority Voting Ensemble Method," in *2019 2nd International Conference on new Trends in Computing Sciences (ICTCS)*, Amman, 2019.

[8] H. Al-Zubaidi, M. Dweik and A. Al-Mousa, "Stroke Prediction Using Machine Learning Classification Methods," in *International Arab Conference on Information Technology (ACIT)*, Abu Dhabi, UAE, 2022.

[9] N. Abdulhadi and A. Al-Mousa, "Diabetes Detection Using Machine Learning Classification Methods," in *2021 International Conference on Information Technology (ICIT)*, Amman, 2021.

[10] D. Buxton, "Application of Machine Learning for Classification of Diabetes," ResearchGate, 2022.

[11] Z. Jaadi, "A Step-by-Step Explanation of Principal Component Analysis (PCA)," builtin, [Online]. Available: https://builtin.com/data-science/step-step-explanation-principal-component-analysis.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research,* vol. 16, pp. 321-357, 1 June 2002.

[13] Partha Pratim Sarangi, Madhumita Panda, Subhashree Mishra, Bhabani Shankar Prasad Mishra, Banshidhar Majhi, Machine Learning for Biometrics - Concepts, Algorithms and Applications, ScienceDirect, 2022.

[14] "K-Nearest Neighbor(KNN) Algorithm for Machine Learning," JavaTpoint, [Online]. Available: https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning.

[15] T. Yiu, "Understanding Random Forest," Towards Data Science, 12 June 2019. [Online]. Available: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

[16] A. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'REILLY, 2017.

[17] "Machine Learning - Logistic Regression," w3schools, [Online]. Available: https://www.w3schools.com/python/python_ml_logistic_regression.asp.

[18] "What is logistic regression?," IBM, [Online]. Available: https://www.ibm.com/topics/logistic-regression#:~:text=Similar%20to%20linear%20regression%2C%20logistic,variable%20versus%20a%20continuous%20one..

[19] "Scikit Learn - Stochastic Gradient Descent," tutorialspoint, [Online]. Available: https://www.tutorialspoint.com/scikit_learn/scikit_learn_stochastic_gradient_descent.htm.

[20] "Cross-Validation in Machine Learning," javaTpoint, [Online]. Available: https://www.javatpoint.com/cross-validation-in-machine-learning.

[21] K. Leung, "Micro, Macro & Weighted Averages of F1 Score, Clearly Explained," Towards Data Science, 4 January 2022. [Online]. Available: https://towardsdatascience.com/micro-macro-weighted-averages-of-f1-score-clearly-explained-b603420b292f.