# An Improved Adaptive Synthetic Sampling Technique and Machine Learning Model for Enhanced Imbalance Medical Data Classification

**Abdullahi Hafiz[1], Bashir Sulaimon Adebayo[2], Enesi Femi Aminu[3]**

*Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria[1,2,3].*

hafyzabdallah@.gmail.com[1], bashirsulaimon@futminna.edu.ng[2], enesifa@futminna.edu.ng[3],

***Abstract:***

*Medical data classification plays a pivotal role in healthcare decision-making. Addressing the challenges posed by imbalanced datasets is critical for accurate classification in this domain. This paper presents an innovative approach to enhancing the Adaptive Synthetic Sampling (ADASYN) algorithm, tailored specifically for medical data classification. The proposed Improved ADASYN algorithm integrates ADASYN with k-means clustering to address two key issues: generating synthetic minority samples and eliminating potential outliers introduced by ADASYN. By doing so, it aims to mitigate the adverse effects of reduced accuracy in the majority class, ultimately enhancing classification performance. The pre-processed medical data undergoes an estimation process to determine the requisite number of synthetic samples, which are subsequently generated using ADASYN. These synthesized samples are seamlessly merged with the original minority data. Subsequently, k-means clustering is employed to identify and filter out misclustered data, effectively removing outliers. If data imbalance persists, the algorithm iterates, recalculating the need for additional minority samples. This iterative process continues until a balanced dataset is achieved. The resulting balanced dataset is then primed for utilization by machine learning algorithms for classification purposes. Notably, the proposed algorithm was implemented using MATLAB version R2023a, ensuring reproducibility and applicability in practical medical data classification scenarios. This research presents a promising step towards improving the robustness and accuracy of medical data classification, thereby contributing to enhanced healthcare decision support systems.*

***Keywords:*** *Imbalance, Datasets, Adaptive, synthetic, Data mining, Machinelearning, Oversampling, Undersampling.*

## 1. Introduction

Class imbalance is a common problem in medical datasets, where the number of samples for one class is significantly smaller than the others. This can lead to poor performance of machine learning models, especially for the minority class. Various oversampling techniques have been developed to address this issue, among which Adaptive Synthetic Sampling (ADASYN) has become popular due to its ability to generate synthetic samples in proportion to the degree of imbalance. ADASYN Sampling is a novel oversampling technique for learning from imbalanced datasets. The essential idea of ADASYN is to use a weighted distribution for different minority class examples according to their level of difficulty in learning, where more synthetic data is generated for minority class examples that are harder to learn compared to those minority examples —.

When dealing with huge volume of data, data mining and machine learning procedure aid in the finding of solutions, with bulk of applications oriented towards healthcare sector. Because the rate of rise in the number of patients is directly related to the rate population growth and lifestyle changes, healthcare sector has a significant need for data processing service

**One of the procedures in medical data collection is shown in figure 2.1.**

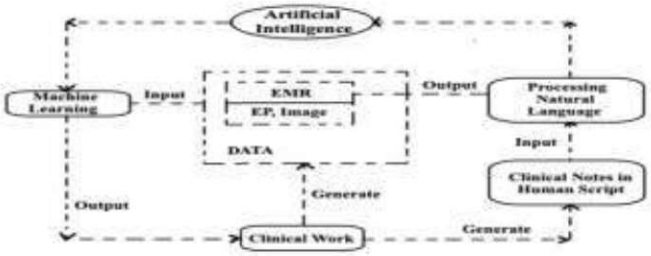


Figure 1.1: Roadmap for medical data collection (Patil et al., 2022)

In order to address the imbalanced data issue, different sampling methods have been proposed, which basically fall into two categories, i.e. under-sampling and over-sampling as shown in the figure 2.2(W. Zhang et al., 2020).

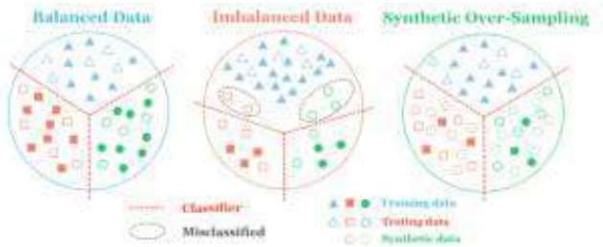


Figure 1.2: Data-driven fault diagnosis performances with balanced and imbalanced data(W. Zhang et al., 2020)

## 2. Review of Related Works

Boughorbel et al., (2017) investigated the use of an optimal classifier for biased data using the Matthews Correlation Coefficient (MCC) metric, which is a commonly used measure of performance in bioinformatics. They obtained an optimal classifier by applying an optimal Bayesian classifier based on the MCC metric using the Frechet derivative. The simulation data was used to check the accuracy of the optimality result and a large spatial search of all possible binaries was performed. The proposed classifier was then evaluated on 64 datasets with varying degrees of data imbalance. The performance of three classifiers were compared: the proposed algorithm (called the MCC classifier), the Bayesian classifier with a predefined threshold (MCC basis), and the unbalanced support vector machine (SVM-imba).

Duan et al., (2019) introduced Astraea, a self-balancing federated learning (FL) platform designed to eliminate imbalances by increasing data based on Z-Score and rescheduling more clients based on Mediator. In order to overcome the degradation in accuracy, it is necessary to rebalance the training data for each client. One way is to redistribute the local client data until it is evenly distributed.

Lin et al., (2017) introduced a cluster-based sampling technique to solve the problem of class bias in the data. They proposed two sampling strategies that use clustering in the data preprocessing phase. In the first strategy, the number of clusters in the majority class equals the number of data points in the minority class, and the cluster centers represent the majority class. The second strategy selects the nearest neighbors of the cluster centers to represent the majority class. To evaluate their approach, the researchers conducted two experimental studies. The first study included 44 small datasets with bias coefficients ranging from 1.8 to 129 and varying numbers of data samples collected ranging from 130 to 5500

Mathew et al., (2017) introduced an improved version of SMOTE, called Weighted Kernel Based SMOTE (WK-SMOTE), to address SMOTE's limitations for nonlinear problems. WK-SMOTE operates in the feature space of a Support Vector Machine (SVM) classifier and consists of three main phases. First, pairs of colors and neighbors are identified in the space characteristic of the minority class. Second, synthetic instances are generated along the line segment connecting the pair. Then, using kernel functions, the dot product of synthetic instances is calculated with the original training dataset. The extended Gram matrix consisting of training instances and generated synthetic minority instances is used to train the SVM classifier. The proposed algorithm shows better G-mean results than the base methods on unbalanced datasets from the KEEL data repository. It achieves the highest average G-Score of 0.371 and maintains a high overall accuracy of out of 88.1 %. Future work will focus on extending WK-SMOTE to develop new regression algorithms for probabilistic remaining life estimation of industrial machines.

Deng et al., (2020) presented a novel improved clustering algorithm that uses coefficients of variation or entropy to capture the local spatial distribution of data and perform majority-class hierarchical clustering. This algorithm features low complexity and the ability to improvedally adjusts clusters based on iteration of the AdaBoost algorithm, effectively adapting to changes caused by changes in sample weights. To measure the importance of each cluster, an index was developed, which in turn served as the basis for an improved sampling algorithm that selects samples with the greatest mass. Visual experiments confirmed the effectiveness of this sampling algorithm.


Figure 2.1: Illustration of distance between each type of minority sample (Chen *et al.*, 2020)

## 3. Materials and Methods

This section presents the methodology to be employed in order to accomplish the aim and objectives of this research work. Figure 3.1 illustrates the suggested approach for attaining each specific objective.

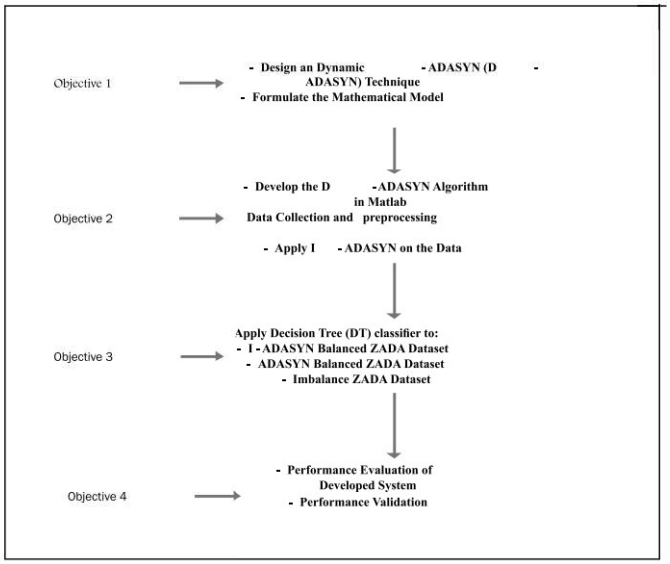


**Figure 3.1: Research Methodology**

### 3.1 Formulation of Mathematical Model

Let D denote the imbalanced dataset, where D = $\{x_i, y_i\}$ represent the input features vectors $x_i$ and their corresponding class labels $y_i$ The ADASYN algorithm involves the following steps:

a. Estimating the number of synthetic samples to generate:
Let N and M be the number of minority and majority class samples in D, respectively. the imbalance ratio R is defined as:

$$R = \frac{M}{N} \quad \text{..... (3.1)}$$

To determine the number of synthetic samples to generate for each minority sample we can use the following equation:

$$N_S = \frac{M-N}{N} \quad \text{..... (3.2)}$$

b. Selecting k nearest neighbors:

For each minority sample $x_i$, we select its k nearest neighbors from the same class. Let K($x_i$) denote the set of k nearest neighbors for $x_i$.

c. Computing the feature space distance:

To compute the feature space distance between a minority

sample $x_i$ and its nearest neighbors, we can use a distance metric such as Euclidean distance:

$$D(x_{i,j}) = \sqrt{(\sum(x_i - x_i^2))} \quad \text{..... (3.3)}$$

d. Normalizing the feature space distance:
Normalize the feature space distance to obtain a normalized distance value between 0 and 1:

$$Norm\left(D(x_{i,j})\right) = \frac{D(x_{i,j})}{Max\left(D(x_i,\ K(x_i))\right)} \quad \text{..... (3.4)}$$

a. Determining the synthetic sample distribution:
Based on the normalized feature space distance, we calculate the distribution of synthetic samples to be generated for each minority sample.

$$Distribution(x_i) = \frac{Norm\left(D(x_{i,j})\right)}{\Sigma\left(Norm\left(D(x_{i,j})\right)\right)} \quad \text{..... (3.5)}$$

f. Generating synthetic samples:
For each minority sample $x_i$ we generate synthetic samples by interpolating between $x_i$ and its randomly selected nearest neighbors based on the distribution calculated in the previous step.

g. K-Means Filtering
The k-means clustering algorithm aims to partition the balanced dataset BD' into k clusters, where k is determined as the number of classes in BD. The K-means algorithm is used for filtering by:

i. Randomly generate cluster centroids.
j. Compute distance of samples to cluster centroid using Euclidean distance.
ii. Assign samples to cluster centroids.
iii. Update cluster centroids until convergence.
iv. Set filter threshold.
v. Identify and remove outliers.

**Table 3.1 summarizes the proposed I-ADASYN algorithm.**

**Table 3.1: Proposed I-ADASYN Algorithm**

| ***Algorithm Name: I-ADASYN*** | |
|---|---|
| ***Input: Input: Imbalanced medical dataset with majority class samples and minority class samples.*** | |
| ***Output: Balanced Dataset*** | |
| *Step* | *Description* |
| *1. Preprocessing:* | *Perform necessary preprocessing steps (e.g., cleaning, normalization, feature selection) on dataset D.* |
| *2. Initialization:* | *Set k = 0 (number of iterations) and BD as an empty set (balanced dataset).* |
| *3. While the dataset is still imbalanced:* | |
| | *a. Increment k by 1.*<br>*b. Apply ADASYN:*<br>*i. Estimate number of synthetic samples to generate for each minority sample based on imbalance ratio.*<br>*ii. Select k nearest neighbors from the same class for each minority sample.*<br>*iii. Compute feature space distance between each minority sample and its k nearest neighbors.*<br>*iv. Normalize the feature space distance.*<br>*v. Determine synthetic sample distribution by multiplying normalized distance by number of synthetic samples to generate.*<br>*vi. Generate synthetic samples for each minority sample by interpolating between the sample and its nearest neighbors based on the distribution.*<br>*c. Combine original minority samples and newly generated synthetic minority samples to form balanced dataset BD'.* |
| *5. Perform Filtering:* | |
| | *a. Set number of clusters as total number of class in BD'.*<br>*b. Assign each sample in BD' to its nearestcluster centroid.*<br>*c. Identify and remove outliers as misclustered samples outside a specified threshold from their assigned cluster centroid.* |
| *6. Check Dataset Balance:* | |
| | *a. If minority class is adequately represented (e.g., imbalance ratio below specified threshold or desired balance ratio achieved), proceed to step 6.*<br>*b. Otherwise, return to step 3 with updated balanced dataset BD'.* |

### 3.2. Dataset Collection

The dataset used in this analysis is derived from blood analysis of fasting sugar conducted at the Shaker laboratory in Zakho city, Kurdistan Region of Iraq (Hassan, 2020). Initially, the dataset contained various medical features for approximately 7,000 patients. After preprocessing steps such as cleaning, integration, and reduction, only the features that have a significant impact on diabetes were selected. Consequently, a new dataset called ZADA (Zakho Diabetes Analysis) was created, specifically focused on diabetes-related information.
The ZADA dataset comprises 909 records, with each record containing information on seven features. One of the essential features is the binary response variable "Class," which denotes the patient's diabetes status. It takes the value 0 for healthy individuals and 1 for diabetic individuals. Table 3.2 provides an overview of the general characteristics of the ZADA diabetes dataset. It summarizes the attributes, their descriptions, minimum and maximum values observed in the dataset, as well as the mean values.
Table 3.2: ZADA Dataset summary (Hassan, 2020)

| Attribute Name | Attribute Description | Min | Max | Mean |
|---|---|---|---|---|
| **Age** | Age of patients | 20 | 86 | 48.01 |
| **Cholesterol** | Test of Cholesterol | 110 | 340 | 200.56 |
| **L_HDL** | High-density Lipoprotein | 23 | 65 | 42.97 |
| **L_LDL** | Low-density Lipoprotein | 36.8 | 266.2 | 124.87 |
| **L_VLDL** | Very Low-Density Lipoprotein | 8.6 | 80 | 32.73 |
| **Uric Acid** | Test of Uric Acid | 2.22 | 10.2 | 5.72 |
| **Class** | 1– Positive, 0= Negative | - | - | - |

#### 3.1.1 Data Pre-processing

In order to pre-process the data and put it a form suitable for the machine learning models, missing values will be replaced with the average values of the particular feature to be replaced. The dataset will also be normalized using the min-max normalization approach. This is to prevent bias towards features with higher values.

### 3.3.Data Resampling Using I-ADASYN Algorithm

The pre-processed data will be passed through the developed k-smote algorithm for resampling the minority class. The balanced data will move to the machine learning models for classification.

### 3.4.Machine Learning Models Investigation

Three machine learning models will be investigated. They are; Decision Tree (DT) algorithm, Support Vector Machine (SVM), and Artificial neural network (ANN). The model that performs the best will be adopted. The unbalanced data will also be passed through the models in order to validate the performance of the proposed model.

### 3.5. Performance Evaluation Metrics

The proposed models' performance will be evaluated using accuracy, sensitivity, and specificity performance metrics. Mathematically, they are given as:

**i. Accuracy**

This is the ratio of all correctly classified instances over all instances given as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad \text{..... (3.6)}$$

**i. Sensitivity**

This is the ratio of all correctly classified malignant instances over sum of correctly classified malignant instances and wrongly classified benign given as:

$$Sensitivity = \frac{TP}{TP + FN} \times 100\% \quad \text{..... (3.7)}$$

**i. Specificity**

This is the ratio of all correctly classified benign instances over sum of correctly classified benign instances and wrongly classified malignant given as:

$$Specificity = \frac{TN}{TN + FP} \times 100\% \quad \text{..... (3.8)}$$

### 3.6. Performance Validation

To validate the performance of the Proposed Model, the following steps will be carried out:

i. Apply the proposed model, which is the Improved ADASYN algorithm, to address the issue of imbalanced data.
ii. Pass the balanced data through the Decision Tree (DT) ML model.
iii. Compare the performance of each model when using the balanced data with their performance when imbalance data is used..
iv. Evaluate and compare the metrics (accuracy, precision, recall, F1-score) of each model using the unbalanced data and the balanced data generated by the proposed model.
v. Determine if the proposed model improves the performance of the machine learning models on the unbalanced dataset.

## 4.Results and Discussion

Inthissection,theproposedmethodwasdiscussedindetailswitha pseudo-code,andadiagramof ourproposedmethod. Thedataset andcorrelationcoefficientbetweenattributeswere alsoexplained.

### 4.1 I-Adasyn ZADA Data Balancing Result

Table 4.1 illustrates the impact of balancing the dataset by increasing the number of samples in the minority class (One Class) from 186 to 704. A higher imbalance ratio means that the dataset is more imbalanced. In the unbalanced dataset, the minority class represents only 25.73% of the total samples with an imbalance ratio of 74.27%. While in the balanced dataset, it represents a much higher percentage, approximately 97.38% and a small imbalance ratio of 2.63%.

Balancing the dataset in this way was done to address issues related to class imbalance in machine learning, where one class has significantly fewer samples than the other. In the balanced dataset, the imbalance ratio is reduced, making it more suitable for training machine learning models that may be sensitive to class imbalances.

| Class | Number of samples | |
|---|---|---|
| | **Unbalanced** | **Balanced** |
| **Zero Class** | 723 | 723 |
| **One Class** | 186 | 704 |
| **Imbalance Ratio** | 74.27% | 2.63% |
| **Minority/Majority Class Ratio** | 25.73% | 97.38% |

**Table 4.2 is a sample of the balanced dataset for both one and zero class.**

**Table 4.2: Sample balanced ZADA dataset**

| Age | L_Cholestrol | L_HDL | L_LDL | L_VLDL | Uric Acid | class |
|---|---|---|---|---|---|---|
| 65 | 218 | 39 | 157 | 22 | 4.25 | 0 |
| 51 | 160 | 44 | 102.2 | 13.8 | 6.1 | 0 |
| 51 | 160 | 44 | 102.2 | 13.8 | 6.1 | 0 |
| 65 | 218 | 39 | 157 | 22 | 4.25 | 0 |
| 38 | 174 | 44 | 106.2 | 23.8 | 8.92 | 0 |
| 33 | 170 | 45 | 101.4 | 23.6 | 5 | 0 |
| 38 | 226 | 43 | 167 | 16 | 6 | 0 |
| 50 | 201 | 49 | 126.8 | 25.2 | 4.5 | 0 |
| 65 | 254 | 49 | 153 | 52 | 6.8 | 0 |
| 45 | 196 | 44 | 104.8 | 47.2 | 5.5 | 0 |
| 38 | 185 | 48 | 113.4 | 23.6 | 6.75 | 0 |
| 44 | 252 | 47 | 153.6 | 51.4 | 6.8 | 0 |
| 36 | 154 | 36 | 72 | 46 | 7.1 | 0 |
| 64 | 204 | 34 | 133.2 | 36.8 | 8.1 | 0 |
| 51 | 200 | 45 | 137 | 18 | 5.4 | 0 |
| 45 | 193 | 42 | 125.2 | 25.8 | 4.7 | 0 |
| 27 | 250 | 40 | 178 | 32 | 6.9 | 0 |
| 65 | 231 | 53 | 147.4 | 30.6 | 6.85 | 0 |
| 58 | 149 | 35 | 100.2 | 13.8 | 5.8 | 0 |
| 60 | 211 | 42 | 132 | 37 | 6.1 | 1 |
| 44 | 212 | 58 | 120.4 | 33.6 | 6.1 | 1 |
| 55 | 203 | 30 | 137 | 36 | 4.5 | 1 |
| 63 | 295 | 56 | 201 | 38 | 5.08 | 1 |
| 62 | 226 | 50 | 111 | 65 | 8.1 | 1 |
| 40 | 222 | 50 | 144.4 | 27.6 | 7.5 | 1 |
| 52 | 220 | 39 | 125 | 56 | 5.2 | 1 |
| 62 | 300 | 41 | 204 | 55 | 5.2 | 1 |
| 65 | 226 | 28 | 171.6 | 26.4 | 4.3 | 1 |
| 62 | 215 | 54 | 135.8 | 25.2 | 5.8 | 1 |
| 49.98963 | 219.9862 | 55.00519 | 131.1813 | 33.79965 | 6.898617 | 1 |
| 45.63446 | 214.1793 | 57.18277 | 123.342 | 33.65448 | 6.317928 | 1 |
| 55.33787 | 214.0708 | 59.7466 | 116.1553 | 38.16894 | 5.071117 | 1 |
| 60.10883 | 219.1088 | 43.43533 | 149.9211 | 25.75237 | 4.20205 | 1 |
| 59.68063 | 215.1676 | 41.0838 | 147.5257 | 26.5581 | 4.172253 | 1 |
| 59.7786 | 219 | 43 | 150.4 | 25.6 | 4.3 | 1 |
| 74 | 313 | 41 | 229 | 43 | 6.46 | 1 |
| 67.03458 | 223.4381 | 38.11816 | 132.4795 | 52.84035 | 7.578818 | 1 |
| 62.54321 | 239 | 41 | 143.2 | 54.8 | 7.55 | 1 |
| 60.83706 | 225.3014 | 37.39511 | 139.7393 | 48.167 | 8.090734 | 1 |

### 4.2 I-ADASYN-DT Classification Results

Table 4.3 below shows the Decision Tree algorithm performance using kmeans-Adasyn over Accuracy, Recall, Precision, F1 and MSE respectively. The Accuracy of this algorithm which is one of the performance metrics for determining the performance of a model was found to be 92.31% having a recall of 89.74% and precision of 93.70% respectively.

**Table 4.3: I-Adasyn-DT Model Result**

| Algorithm | ACC | Recall | Precision | F1 | MSE |
|---|---|---|---|---|---|
| DT | 92.31 | 89.47 | 93.70 | 91.54 | 0.0769 |

Figure 4.1 shows the confusion matrix of the I-Adasyn-DT model, which was applied to the DT Balanced ZADA dataset. From the confusion matrix, 119 samples were correctly classified as Diabetic (True Positive) while 145 samples as healthy (True Negative) samples. 8 samples were wrongly classified as healthy (False Negative) while 14 samples were classified wrongly as diabetic (False positive).

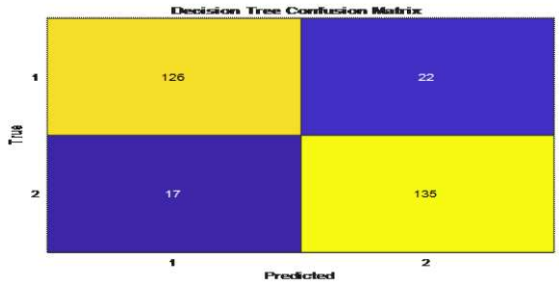


Fig 4.1: Adasyn model confusion matrix

## 5. Conclusions and Recommendation

In this study, we have successfully developed an improved Adaptive Synthetic Sampling Technique (I-Adasyn) in combination with a Decision Tree (DT) model for the enhanced classification of Diabetes using the ZADA Diabetes dataset. Our research has yielded significant and promising results, with the I-Adasyn-DT model achieving the highest accuracy rate, precision, and F1 values, all of which were critical metrics for evaluating the performance of our models. Specifically, the I-Adasyn-DT model achieved an accuracy rate of 92.31%, precision of 93.7%, and an F1 value of 91.54%, showcasing its robustness in predicting diabetes mellitus.It's noteworthy that while the DT model had the lowest accuracy at 85.71%, it displayed a higher recall of 93.84% compared to the I-Adasyn-DT model's 89.47%. This suggests that the DT model was better at identifying true positive cases but not as efficient at overall accuracy and precision compared to the I-Adasyn-DT model. On the other hand, the Adasyn-DT model exhibited the lowest precision, recall, and F1 values, indicating its limitations in correctly classifying diabetes cases.

Therefore, our research has clearly demonstrated that the improved I-Adasyn-DT model outperforms both the traditional DT model and the Adasyn-DT model in terms of accuracy, precision, recall, and F1 values. This signifies the I-Adasyn-DT model as the most suitable choice for predicting Diabetes Mellitus among the three models.

## References

Boughorbel, S., Jarray, F., & El-anbari, M. (2017). *Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric*. 1–17.

Chen, B., Xia, S., Chen, Z., Wang, B., & Wang, G. (2020). RSMOTE : A self-adaptive robust SMOTE for imbalanced problems with label noise. *Information Sciences*. https://doi.org/10.1016/j.ins.2020.10.013

Deng, X., Xu, Y., Chen, L., & Zhong, W. (2020). Dynamic clustering method for imbalanced learning based on AdaBoost. *The Journal of Supercomputing*, *0123456789*. https://doi.org/10.1007/s11227-020-03211-3

Duan, M., Liu, D., Chen, X., Liu, R., Tan, Y., & Liang, L. (n.d.). *Self-balancing Federated Learning with Global Imbalanced Data in Mobile Systems*. *X*(X), 1–13.

Hassan, M. M., & Amiri, N. N. (2019). *Classification of Imbalanced Data of Diabetes Disease Using Machine Learning Algorithms*. *October*.

Lin, W., Tsai, C., Hu, Y., & Jhang, J. (2017). *Clustering-based undersampling in class-imbalanced data*. *410*, 17–26. https://doi.org/10.1016/j.ins.2017.05.008

Mathew, J., Member, S., Pang, C. K., & Member, S. (2017). *Classification of Imbalanced Data by Oversampling in Kernel Space of Support Vector Machines*. 1–12.

Mehta, K., Ali, L., & Nuagah, S. J. (2023). *Retracted : Data Mining in Employee Healthcare Detection Using*. *2022*.

Patil, R. R., Ruby, U., Chaithanya, B. N., Swasthika Jain, T. J., & Geetha, K. (2022). Review of fundamentals of Artificial Intelligence and application with medical data in healthcare. *Journal of Integrated Science and Technology*, *10*(2), 126–133.

Zhang, C., Tan, K. C., Li, H., & Hong, G. S. (2018). *A Cost-Sensitive Deep Belief Network for Imbalanced Classification*. 1–14.

Zhang, W., Li, X., Jia, X. D., Ma, H., Luo, Z., & Li, X. (2020). Machinery fault diagnosis with imbalanced data using deep generative adversarial networks. *Measurement: Journal of the International Measurement Confederation*, *152*. https://doi.org/10.1016/j.measurement.2019.107377