Handling Class Imbalance via Counterfactual Generation in Medical Datasets

Asifa Mehmood Qureshi[1,*,†], Abhishek Kaushik[1,*,†], Gilbert Regan[1,†], Kevin McDaid[1,†] and Fergal McCaffery[1,†]

[1] *Regulated Software Research Centre, Dundalk Institute of Technology, Dundalk, Ireland*

Abstract

Real-world datasets often contain uneven class distributions, that if not handled properly result in biased Machine Learning (ML) models. Therefore, class balancing is important to avoid overfitting, improve model generalisation and ensure fairness. Most state-of-the-art techniques used to balance datasets do not take into account the majority class samples that contain greater distributional information of the dataset. Therefore, in this article, we propose a method that generates counterfactuals using majority-class samples. The method takes an imbalanced dataset as input, normalises the dataset, and trains a Support Vector Machine (SVM) classifier on it. Afterwards, the majority class samples that lie near the decision boundary are extracted and perturbed until they are classified as minority class samples. The method is evaluated on two benchmark datasets i.e., the Diagnostic Wisconsin Breast Cancer dataset and the Eye State Classification Electroencephalogram (EEG) dataset. The results show that our approach produces reasonable accuracy, Area Under Curve (AUC), and Geometric Mean (Gmean) scores. Also, the F1-score also improved for minority classes when oversampled using counterfactuals. Moreover, the model achieved promising results when compared with state-of-the-art techniques.

Keywords

Boundary enhancement, Over-sampling, SVM, decision boundary, classification, counterfactuals

1. Introduction

The class imbalance problem typically occurs when there are many more instances of one class called the majority class than others [1]. It is considered one of the significant challenges in relation to data quality [2]. Imbalanced datasets exist in numerous real-world fields such as text classification [3], object detection [4], network security [5], medical diagnosis [6] and many more. Machine Learning (ML) classifiers when trained on imbalanced datasets are skewed towards majority classes and frequently misclassify instances from minority classes resulting in biased outcomes [7]. These biases may result in discrimination in automated decision-making especially in critical sectors like healthcare [8]. For example, in a breast cancer dataset, if the number of data samples for the positive cancer diagnosis is smaller than healthy patient samples then the classifier trained on such a dataset may misclassify the patient as healthy which can lead to life-threatening consequences [9].

There are several methods to balance datasets including, algorithm-level methods, data-level methods, and hybrid methods [10]. Data-level methods are widely used because these methods directly address the shortcomings of data thus improving the data quality on which the model is being built. These methods tend to transform the original dataset to change the class distribution via re-sampling [7]. Re-sampling includes both under-sampling and over-sampling i.e., under-sampling involves the removal of the majority class samples from the dataset whereas over-sampling is the process of increasing minority class data samples by generating synthesised data [11]. Under-sampling may remove data points that contain important information, and it reduces the dataset size which may worsen the ML model performance [12]. Conversely, over-sampling adds essential information to the minority class without any information loss and prevents instances from being misclassified [13].

AICS'24: 32nd Irish Conference on Artificial Intelligence and Cognitive Science, December 09–10, 2024, Dublin, Ireland

[*] Corresponding author.

[†] These authors contributed equally.

D00273262@student.dkit.ie (A. M. Qureshi); abhishek.kaushik@dkit.ie (A. Kaushik)
0009-0002-4312-353X (A. M. Qureshi); 0000-0002-3329-1807 (A. Kaushik)

Several over-sampling methods use minority samples for new data generation. However, these methods ignore the majority class entirely in favour of focusing on minority class characteristics, which provide little distributional information. Consequently, they do not focus on the global properties of the dataset that are defined by majority class distribution and produce inaccurate synthetic training examples [14].

In this paper, an over-sampling approach is proposed that uses majority class data samples to generate minority class data. In this method, the majority class samples named actual samples are perturbed to generate counterfactuals that lie in the minority sample region. The method takes an imbalanced normalised binary class dataset as input. A Support Vector Machine (SVM) classifier is trained on the dataset. The samples of the majority class samples that are closest to the classifier decision boundary are extracted. These data samples are perturbed to a level so that they move to the minority class space. Two publicly available binary class medical datasets are used to validate our proposed model. The contributions of the paper are as follows:

- A method that uses majority class samples to generate minority data points. These newly generated data points can be termed counterfactuals.
- In order to lower the computation overhead and enhance the decision boundary, we trained an SVM classifier to extract data points closest to the decision boundary rather than selecting random samples from the majority class [15].
- The selection of data samples nearer to the decision boundary containing support vectors also ensures minimum deviation of the majority class samples to generate samples of the minority class rather than limiting the distance using a constant.
- The performance of the model is evaluated on two benchmark medical datasets using various evaluation metrics.

The remainder of the paper is structured as follows: Section 2 provides a literature review of relevant oversampling techniques. Section 3 explains the overall methodology. Section 4 defines the dataset and corresponding evaluation results. Finally, section 5 concludes the discussion and lists future work.

2. Related work

The problem of class imbalance has drawn a lot of attention from the scientific community. This section gives a summary of the techniques for over-sampling. For better understanding, we categorise the literature into two streams: Statistical and Machine Learning (ML) Methods and Deep Learning (DL) methods.

2.1. Statistical and Machine Learning (ML) methods

Several studies have been carried out to handle the issue of class imbalance within datasets. One of the most used techniques is the Synthetic Minority Over-sampling Technique (SMOTE) [16]. It generates new samples by utilising interpolation between decision minority samples nearest neighbours. Another SMOTE variant is the Borderline SMOTE which generates minority samples at the borderline to enhance the decision boundary of the classifier [17]. There are more than 81 variants of SMOTE proposed in the existing research work. The majority of these methods focus on utilising minority-class samples to produce new artificial samples that may lead to overfitting. In another study by Sharma et al. [14], the majority class samples were used to generate synthetic data. They utilised Mahalanbois distance to generate minority samples that are at an equal distance from the majority class samples. However, this technique does not consider boundary samples in their generation process. In another study [18], SVM-SMOTE is combined with ensemble learning to enhance the performance of the classifier. The primary goal is to find borderline cases in the minority class by using Kernel Density Estimation (KDE). After the identification of borderline instances, synthetic interpolating is used to generate new samples between the marginal instances and their current minority-class neighbours. Moreover, Wang et al.

[15] also presented a model that utilises majority-class samples to generate minority-class samples. The model produces reasonable results, but the random selection of the majority class sample increases the computational cost and results in multiple iterations to generate minority class samples that are at a minimum distance from the majority samples.

2.2. Deep Learning (DL) methods

Deep learning (DL) has also been used to generate synthetic data due to its advanced capabilities. For this purpose, Generative Adversarial Networks (GANs) are extensively used. In [19], the authors created synthetic electroencephalography (EEG) datasets using a GAN. Also, to balance the dataset used for automatic signal modulation classification, Patel et al. [20] employed a Conditional-GAN (CGAN) for data augmentation. However, the performance of the model was good but deep learning models are computationally complex when compared to conventional methods. Additionally, deep learning models lack explainability, thus providing minimal control over the parameters and the data-generating process [21, 22].

Therefore, we have presented a statistical over-sampling method that utilises the SVM classifier and majority class samples, unlike other techniques to balance the dataset.

3. Methodology

Figure 1 provides an overview of our proposed workflow diagram. Initially, the dataset is normalised and an SVM classifier is trained on the imbalanced dataset. Then, the majority class samples near the decision boundary are extracted using the Euclidean distance and their corresponding counterfactuals are generated. If the generated counterfactual after the perturbation is classified as a minority class sample by the SVM classifier, then it is added to the new dataset otherwise the sample is discarded. This process is repeated until a balanced dataset is obtained. Afterwards, different machine learning classifiers are trained on the newly generated balanced dataset and their performance is evaluated in terms of accuracy, F1-score, Area Under Curve (AUC), and Geometric Mean (Gmean).

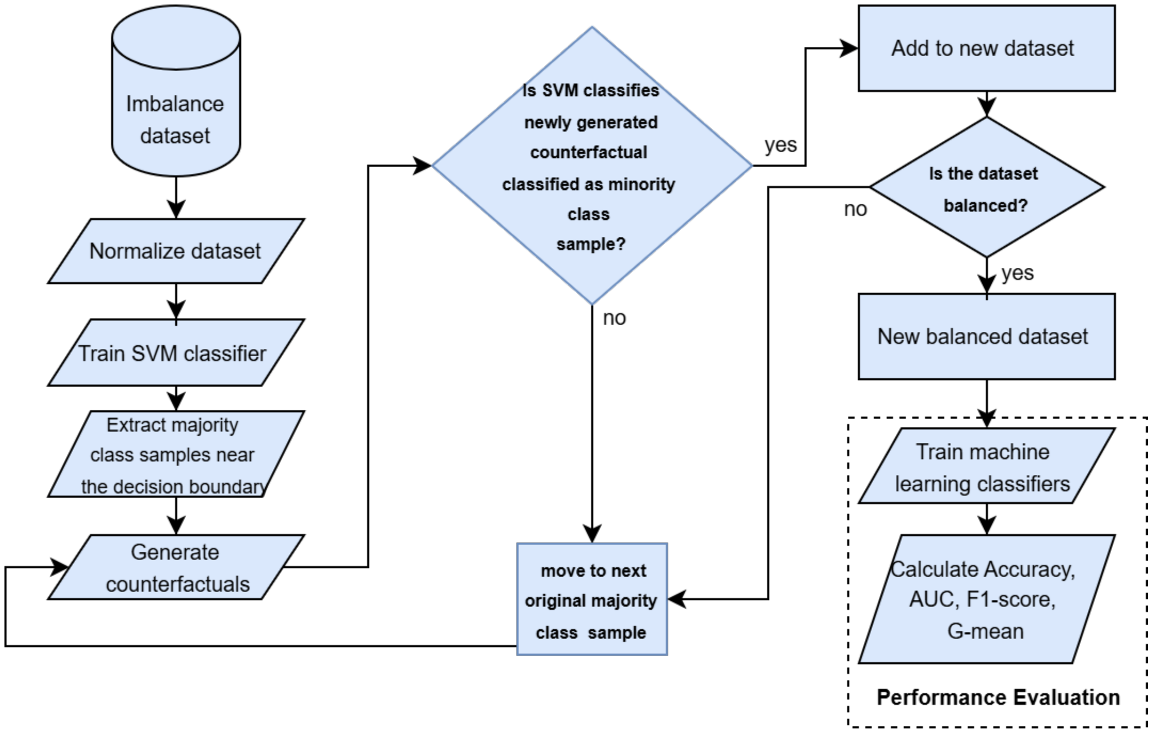


Figure 1: Overview of the proposed workflow diagram to generate counterfactuals

3.1. Data normalisation

Data normalisation includes the transformation of numerical features within a common range to prevent bigger numerical feature values from dominating over smaller numerical feature values [23]. It is an important preprocessing step to enhance the classification performance of the classifier. The dataset was normalised as follows:

$$k' = a + (b - a) \times \frac{k - k_{\text{min}}}{k_{\text{max}} - k_{\text{min}}} \tag{1}$$

Where k' is the normalised feature value, a and b are the desired minimum and maximum values for the normalised range.k presents the original feature value and k_{min} and k_{max} represent the minimum and maximum values of the original feature values. In our case, we kept the values of a and b to be 5 and 20 because normalising within a narrow range helps preserve the distribution shape and optimise the performance of the data generation algorithm.

3.2. Train SVM classifier

After normalisation, an SVM classifier is trained on the original dataset to learn the decision boundary that separates the minority and majority class instances. SVM is a supervised learning algorithm that analyses the dataset linearly and divides the hyperplane by the widest possible gap to classify the samples [23]. Then, the samples from the majority class that are nearest to the SVM classifier decision boundary are extracted based on Euclidean distance using the imbalance ratio to generate counterfactuals as shown in Figure 2.

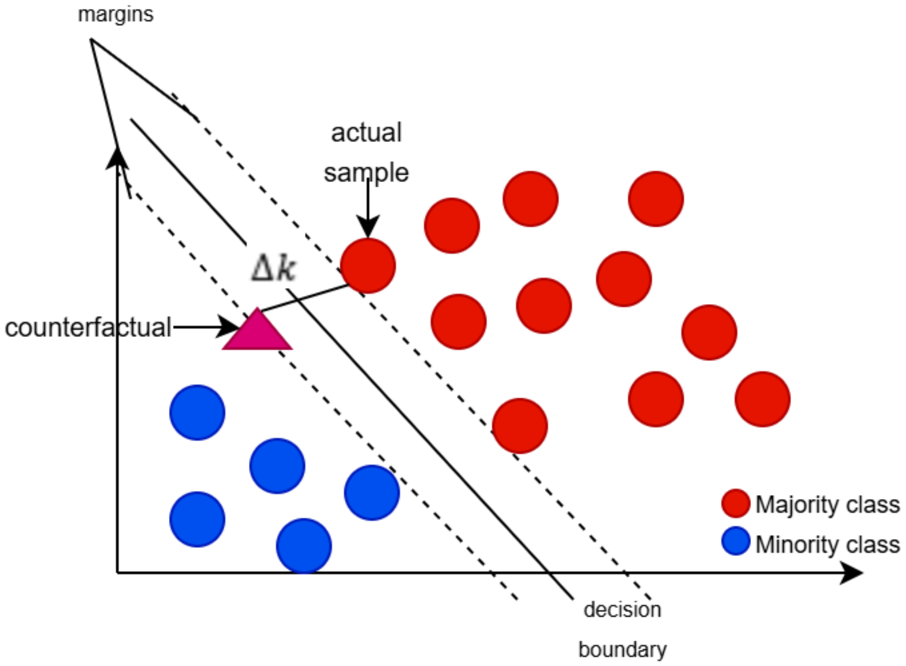


Figure 2: Visualisation of SVM boundary and counterfactual generation. Red colour shows the samples from the majority class, blue presents the samples from the minority class, Δk is the change calculated to generate counterfactuals

3.3. Counterfactual generation

To generate counterfactuals, we employed regular perturbation on each of the selected samples from the majority class. In order to perturb a sample, we used the truncated normal distribution $F(\Delta(k_p))$ that presents the probability distribution obtained from normally distributed random variables by limiting the generated counterfactuals from both below and above [25] as shown in Figure 3.

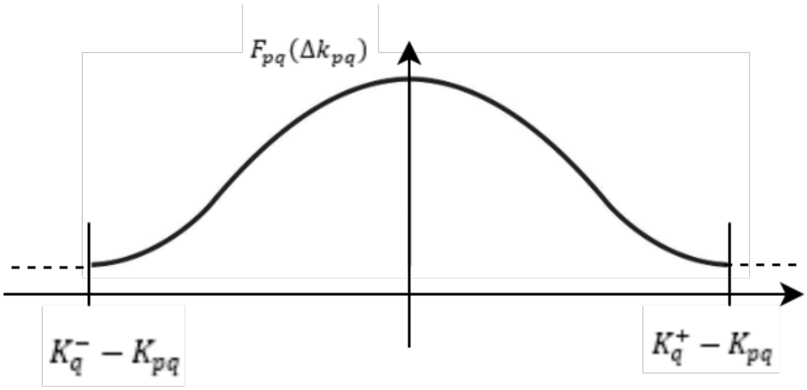


Figure 3: Truncated normal distribution of the actual dataset samples.

For any q^{th} feature of the actual sample k, we utilise the following conditional probabilities to estimate the distribution of the perturbation $\Delta(k_{pq})$ [15].

$$F_{pq}\left(\Delta k_{pq} \mid K_{pq}, K_q^-, K_q^+, \sigma\right) = \begin{cases} \dfrac{\frac{1}{\sigma}\psi\left(\frac{\Delta x_{nm}}{\sigma}\right)}{\Phi\left(\frac{K_q^+ - K_{pq}}{\sigma}\right) - \Phi\left(\frac{K_q^- - K_{pq}}{\sigma}\right)} & \text{if } K_q^- \leq K_{pq} + \Delta k_{pq} \leq K_q^+, \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Where K_q^- and K_q^+ present the minimum and maximum values of the q^{th} feature in the original dataset K, respectively. σ presents the standard deviation of the q^{th} feature. $\psi\left(\frac{\Delta x_{nm}}{\sigma}\right)$ indicates the standard normal distribution's probability density function given below:

$$\psi\left(\frac{\Delta x_{nm}}{\sigma}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\Delta x_{nm}}{\sigma}\right)^2} \tag{3}$$

Φ is the cumulative distribution function given below:

$$\Phi(g) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{g}{\sqrt{2}}\right)\right) = \int_{-\infty}^{g} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}\, dt \tag{4}$$

where

$$g = \frac{K_q^+ - K_{pq}}{\sigma} \quad \text{and} \quad g = \frac{K_q^- - K_{pq}}{\sigma} \tag{5}$$

where erf(.) presents the Gaussian error function. Using this method, any q^{th} feature will not exceed the corresponding range of the feature p.

Now, to generate Δk_{pq} that follows the distribution F_{pq}, we used the inverse transform method where the perturbation is given as follows:

$$\Delta k_{pq} = \Phi^{-1}\left(\Phi(\alpha) + R \cdot \left(\Phi(\beta) - \Phi(\alpha)\right)\right)\sigma + K_{pq} \tag{6}$$

where R is any random number between the range [0,1], and α and β are defined as below:

$$\alpha = \frac{K_q^+ - K_{pq}}{\sigma} \tag{7}$$

$$\beta = \frac{K_q^- - K_{pq}}{\sigma} \tag{8}$$

In the end, the perturbation on the actual data sample can be defined as:

$$S_{pq} = \left\{ \Delta k_p \mid \Delta k_{pq} \sim F_{nm},\ k'_p = k_p + \Delta k_p,\ k_p \in K_0,\ f(k_p) = n,\ f(k'_p) = m \right\} \tag{9}$$

where

$$k_p \in K_0,\ f(k_p) = n,\ f(k'_p) = m \tag{10}$$

where k_p and k'_p are the actual and counterfactual data samples respectively. $f(k_n)$ is the classifier function, and n and m are the class labels. After generating counterfactuals i.e., new data samples that are classified as minority class samples after perturbation by the SVM classifier, we obtained a new balanced dataset that is a combination of actual and synthetic data samples.

Algorithm 1 summarises the steps of generating counterfactuals.

Algorithm 1 Oversampling via counterfactual generation

```
Input: imbalance binary label dataset $K = \{k_1, k_2, k_3, \ldots, k_n\}$
Output: $K_{\text{balanced}}$ //normalise the dataset
$K_{\text{norm}}$ = Normalise($K$)
$f(K_{\text{norm}})$ = Train SVM classifier on the dataset $K_{\text{norm}}$
$K_{\text{norm|near the decision boundary}}$=Extract data points near the decision boundary $f(K_{\text{norm}})$
$K_{\text{synthetic}} = \{\}$
For each $k_p \in K_{\text{norm|near the decision boundary}}$ do
    For $j = 1$ to $T$ do //perturb each sample for T times to control the number of perturbation
        $\Delta k_p = \Delta k_{pq} \sim F_{nm}$ //perturb features by sampling over $F_{nm}$
        $k'_p = k_p + \Delta k_p$
        If $f(k_p) = n$ and $f(k'_p) = m$ then //n is majority class sample and m is minority class sample
            $K_{\text{synthetic}} \leftarrow \{k'_p\}$ //insert the counterfactual into the synthetic dataset
        end if
    end for
end for
$K_{\text{balanced}} = K_{\text{norm}} \cup K_{\text{synthetic}}$ //final balanced dataset
return $K_{\text{balanced}}$
end
```

4. Performance evaluation

4.1. Datasets

To assess our model, we used two benchmark datasets i.e., Diagnostic Wisconsin Breast Cancer and the Eye State Classification EEG datasets as these medical datasets have binary imbalance classes with different imbalance ratios and only continuous features. Following is the description of both datasets:

Diagnostic Wisconsin Breast Cancer Dataset: The Diagnostic Wisconsin Breast Cancer [24] is a multivariate dataset consisting of 30 features and 569 samples. The binary output label classifies the tumour as malignant (0) and benign (1). The majority class for this dataset is 1 and the minority class is 0.

Eye State Classification Electroencephalogram (EEG) dataset: The Eye State Classification EEG [25] is a multivariate time series dataset comprising 14 features and 14980 samples. The output label classifies the eye state as 0 and 1 indicating the eye as open or close respectively. The majority class for this dataset is 0 and the minority class is 1.

Table 1 displays the imbalance ratio of both datasets as well as the number of samples to be generated per class.

4.2. Evaluation of our method

To evaluate the generated counterfactual samples, we trained commonly used ML classifiers on the dataset because they generalise well on diverse datasets. These classifiers include Random Forest (RF), Logistic Regression (LR), K-nearest neighbour and Decision Tree (DT). All these classifiers are trained

Table 1
Number of samples per class, Imbalance Ratio (IR), and number of samples generated per class for each dataset.

Dataset	**Number of samples per class**		**Imbalance Ratio (IR)**	**Number of synthetic samples to be generated**
	Class 0	Class 1		
Diagnostic Wisconsin Breast Cancer Dataset	212 (minority)	357	1.7	145
Eye State Classification EEG dataset	8257	6723 (minority)	1.2	1534

using default parameter settings. The datasets are split into train and test sets of 70:30 ratio. We used Accuracy, Area Under Curve (AUC), Geometric Mean (Gmean) and F1-score to evaluate the performance of our proposed model. These metrics are more comprehensive and largely used in the literature to assess the classifier performance for imbalanced datasets [17]. These parameters are calculated as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where True Positive (TP) represents the correctly classified positive cases, True Negative (TN) represents correctly classified negative cases, False Positive (FP) represents incorrectly classified positive cases and False Negative (FN) represents incorrectly classified negative cases.

$$\text{Gmean} = \frac{1}{n} \sum_{k=1}^{n} \sqrt{\text{TPR}_k \times \text{TNR}_k} \tag{12}$$

where True Positive Rate (TPR) is the ratio of true positives to actual number of positive cases and True Negative Rate (TNR) is the ratio of correctly classified negative cases to the actual number of negative cases of the kth class.

$$\text{F1-score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{13}$$

where the ratio of TP to all positive predictions, including FP and TP, is known as precision. Recall is the percentage of actual positive cases that are correctly predicted by the classifier.

$$\text{AUC} = \sum_{j=1}^{n-1} \frac{(\text{FPR}_{j+1} - \text{FPR}_j) \times (\text{TPR}_{j+1} + \text{TPR}_j)}{2} \tag{14}$$

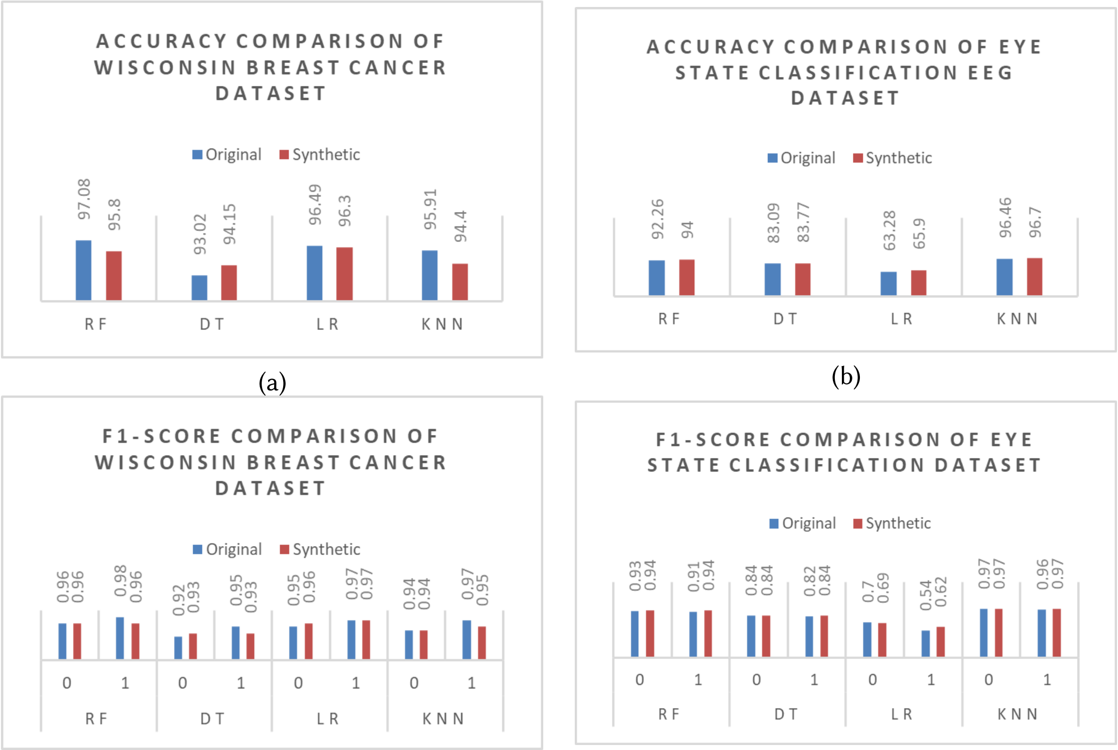
where False Positive Rate (FPR) are actual negative cases that are classified as positive by the classifier.

Figure 4 shows the comparison of accuracy and F1-score before and after applying our proposed method.

The original dataset was biased toward the majority class whereas the synthetic dataset generated using counterfactuals is balanced for each class label. Therefore although the accuracy for the Wisconsin dataset in Figure 4(a) is slightly lower than the original dataset we can say that overall our method maintains good accuracy scores for both datasets. Moreover, Figure 4(b) demonstrates that the F1-score particularly focusing on the minority class has improved for both datasets which represents a better generalisation of the model on each class label. For example, for the Wisconsin breast cancer dataset, the F1-score of the DT for class 0 (minority) has increased from 0.92 to 0.93. Similarly, for the eye state classification dataset, the F1-score of RF for class 1 (minority) has increased from 0.91 to 0.94.

4.3. Comparison with other State-of-the-Art techniques

Moreover, the performance is also compared with other conventional methods including SMOTE, Borderline, Safe-level, and ADASYN. Table 2 and Table 3 show the values for our evaluation parameters



(a) (b)

(c) (d)

Figure 4: Accuracy and F1-score comparison on original and synthetic dataset of Wisconsin breast cancer and eye state classification EEG datasets (a) Accuracy comparison (b) F1-score comparison where RF=Random Forest, DT=Decision Tree, LR= Logistic Regression, KNN=K-Nearest Neighbour and 0 and 1 are the output labels.

for the Wisconsin breast cancer and Eye state classification dataset respectively.

Table 2
Accuracy, AUC and Gmean scores for the Wisconsin Breast Cancer detection dataset

Classifier	Method	Accuracy	AUC	Gmean
Random Forest	SMOTE	0.963	0.995	0.963
	Borderline	**0.967**	**0.999**	**0.968**
	Safe-level	0.963	0.995	0.964
	ADASYN	0.963	0.994	0.963
	Our Method	0.958	0.992	0.959
Logistic Regression	SMOTE	0.949	0.997	0.946
	Borderline	**0.986**	**0.997**	**0.986**
	Safe-level	0.949	0.997	0.946
	ADASYN	0.958	0.991	0.958
	Our Method	0.963	0.997	0.962
K-Nearest Neighbour	SMOTE	0.926	0.956	0.925
	Borderline	0.898	0.956	0.899
	Safe-level	0.926	0.956	0.925
	ADASYN	0.902	0.953	0.902
	Our Method	**0.944**	**0.968**	**0.943**
Decision Tree	SMOTE	0.926	0.796	0.927
	Borderline	**0.949**	**0.949**	**0.949**
	Safe-level	0.926	0.927	0.927
	ADASYN	0.939	0.939	0.939
	Our Method	0.930	0.929	0.929

The results indicate that the performance of our algorithm is comparable to the existing conventional

Table 3
Accuracy, AUC and Gmean scores for the Eye State Classification EEG dataset

Classifier	Method	Accuracy	AUC	Gmean
Random Forest	SMOTE	0.940	0.986	0.940
	Borderline	**0.946**	**0.989**	**0.946**
	Safe-level	0.940	0.986	0.940
	ADASYN	0.939	0.985	0.938
	Our Method	0.940	0.987	0.939
Logistic Regression	SMOTE	0.623	0.679	0.623
	Borderline	0.612	0.664	0.612
	Safe-level	0.623	0.679	0.623
	ADASYN	0.617	0.665	0.615
	Our Method	**0.659**	**0.729**	**0.653**
K-Nearest Neighbour	SMOTE	**0.967**	**0.994**	**0.967**
	Borderline	0.965	**0.994**	0.965
	Safe-level	**0.967**	**0.994**	**0.967**
	ADASYN	0.962	0.991	0.961
	Our Method	**0.967**	**0.994**	**0.967**
Decision Tree	SMOTE	0.835	0.835	0.835
	Borderline	0.825	0.825	0.825
	Safe-level	0.835	0.835	0.835
	ADASYN	0.833	0.833	0.833
	Our Method	**0.838**	**0.838**	**0.838**

synthetic data generation models. Our approach yields comparative results to the Borderline approach for all three metrics i.e., Accuracy, AUC, and Gmean. For the classifier performance, LR and KNN performed well for the Wisconsin Breast Cancer and Eye Movement datasets respectively. Additionally, we have statistically compared our method with the borderline, as it has better performance in comparison to other approaches, using a paired t-test. The test is performed using AUC scores as it assesses the classifier performance better in case of class imbalance. The obtained p-values of 0.91 and 0.31 on the Wisconsin Breast Cancer and Eye Movement datasets respectively indicate that there is no significant statistical difference between the performance of the two. Notably, our approach has the potential to generate counterfactuals with minimum inversion that enhances the boundary of the classifier.

5. Conclusion and future work

In this article, we presented a new counterfactual generation method that generates samples of minority class using the majority class samples in order to balance the dataset. The method makes use of the rich distributional information that lies in the majority class with minimal inversions. The proposed method is assessed on two benchmark datasets: the Diagnostic Wisconsin Breast Cancer dataset and the Eye State Classification EEG dataset. The findings indicate that the F1-score for the minority class have improved which represents better model generalisation. Furthermore, our method yields promising AUC and Gmean values in comparison to existing approaches. In future, we will extend our model to remove any outliers or noisy samples before generating counterfactuals. Also, we will evaluate our model on more diverse medical datasets including different data types and multiclass labels to increase its applicability to diversified real-world datasets. Also, we will extend our experiment by using other classifiers to analyse and improve the shortcomings of SVM.

Acknowledgments

This publication has emanated from research conducted with the financial support of Research Ireland (RI) under Grant number 21/FFP-A/9255.

References

[1] V. Kumar, G. S. Lalotra, P. Sasikala, D. S. Rajput, R. Kaluri, K. Lakshmanna, M. Shorfuzzaman, A. Alsufyani, M. Uddin, Addressing binary classification over class imbalanced clinical datasets using computationally intelligent techniques, in: Healthcare, volume 10, MDPI, 2022, p. 1293.

[2] Y. F. Zhao, J. Xie, L. Sun, On the data quality and imbalance in machine learning-based design and manufacturing—a systematic review, Engineering (2024).

[3] C. Padurariu, M. E. Breaban, Dealing with data imbalance in text classification, Procedia Computer Science 159 (2019) 736–745.

[4] L. Zhang, C. Zhang, S. Quan, H. Xiao, G. Kuang, L. Liu, A class imbalance loss for imbalanced object recognition, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 13 (2020) 2778–2792.

[5] T. Hasanin, T. M. Khoshgoftaar, J. L. Leevy, A comparison of performance metrics with severely imbalanced network security big data, in: 2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI), IEEE, 2019, pp. 83–88.

[6] N. Liu, X. Li, E. Qi, M. Xu, L. Li, B. Gao, A novel ensemble learning paradigm for medical diagnosis with imbalanced data, IEEE Access 8 (2020) 171263–171280.

[7] K. Napierala, J. Stefanowski, Types of minority class examples and their influence on learning classifiers from imbalanced data, Journal of Intelligent Information Systems 46 (2016) 563–597.

[8] J. Gesi, X. Shen, Y. Geng, Q. Chen, I. Ahmed, Leveraging feature bias for scalable misprediction explanation of machine learning models, in: 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE), IEEE, 2023, pp. 1559–1570.

[9] S. Adinarayana, E. Ilavarasan, An efficient decision tree for imbalance data learning using confiscate and substitute technique, Materials Today: Proceedings 5 (2018) 680–687.

[10] M. Khushi, K. Shaukat, T. M. Alam, I. A. Hameed, S. Uddin, S. Luo, X. Yang, M. C. Reyes, A comparative performance analysis of data resampling methods on imbalance medical data, IEEE Access 9 (2021) 109960–109975.

[11] R. Mohammed, J. Rawashdeh, M. Abdullah, Machine learning with oversampling and undersampling techniques: overview study and experimental results, in: 2020 11th international conference on information and communication systems (ICICS), IEEE, 2020, pp. 243–248.

[12] G. Douzas, F. Bacao, Self-organizing map oversampling (somo) for imbalanced data set learning, Expert systems with Applications 82 (2017) 40–52.

[13] M. S. Shelke, P. R. Deshmukh, V. K. Shandilya, A review on imbalanced data handling using undersampling and oversampling technique, Int. J. Recent Trends Eng. Res 3 (2017) 444–449.

[14] S. Sharma, C. Bellinger, B. Krawczyk, O. Zaiane, N. Japkowicz, Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance, in: 2018 IEEE international conference on data mining (ICDM), IEEE, 2018, pp. 447–456.

[15] S. Wang, H. Luo, S. Huang, Q. Li, L. Liu, G. Su, M. Liu, Counterfactual-based minority oversampling for imbalanced classification, Engineering Applications of Artificial Intelligence 122 (2023) 106024.

[16] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority over-sampling technique, Journal of artificial intelligence research 16 (2002) 321–357.

[17] H. Han, W.-Y. Wang, B.-H. Mao, Borderline-smote: a new over-sampling method in imbalanced data sets learning, in: International conference on intelligent computing, Springer, 2005, pp. 878–887.

[18] R. Nithya, T. Kokilavani, T. L. A. Beena, Balancing cerebrovascular disease data with integrated ensemble learning and svm-smote, Network Modeling Analysis in Health Informatics and Bioinformatics 13 (2024) 12.

[19] F. Fahimi, Z. Zhang, W. B. Goh, K. K. Ang, C. Guan, Towards eeg generation using gans for bci applications, in: 2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), IEEE, 2019, pp. 1–4.

[20] M. Patel, X. Wang, S. Mao, Data augmentation with conditional gan for automatic modulation classification, in: Proceedings of the 2nd ACM Workshop on wireless security and machine

learning, 2020, pp. 31–36.
[21] W. J. Von Eschenbach, Transparency and the black box problem: Why we do not trust ai, Philosophy & Technology 34 (2021) 1607–1622.
[22] S. F. Ahmed, M. S. B. Alam, M. Hassan, M. R. Rozbu, T. Ishtiak, N. Rafa, M. Mofijur, A. Shawkat Ali, A. H. Gandomi, Deep learning modelling techniques: current progress, applications, advantages, and challenges, Artificial Intelligence Review 56 (2023) 13521–13617.
[23] N. G. Ramadhan, Comparative analysis of adasyn-svm and smote-svm methods on the detection of type 2 diabetes mellitus, Scientific Journal of Informatics 8 (2021) 276–282.
[24] UCI, Breast cancer wisconsin (diagnostic), 2024. URL: https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic, accessed: 2024-08-15.
[25] UCI, Eeg eye state, 2024. URL: https://archive.ics.uci.edu/dataset/264/eeg+eye+state, accessed: 2024-08-15.