Brazilian Journal of Biometrics, **42**, 225–244 (2024)
doi:10.28951/bjb.v42i3.684

BRAZILIAN JOURNAL OF
BIOMΣTRICS
ISSN:2764-5290

ARTICLE

Improving class probability estimates in asymmetric health data classification: An experimental comparison of novel calibration methods

Olushina Olawale Awe,*,[1] Babatunde Adebola Adedeji,[2] and Ronaldo Dias[1]

[1]Department of Statistics, University of Campinas, Brazil
[2]Obafemi Awolowo University, Ife, Nigeria
*Corresponding author. Email: oawe@unicamp.br

(Received: September 13, 2023; Revised: November 30, 2023; Accepted: February 21, 2024; Published: August 30, 2024)

Abstract

In the context of health data classification, imbalanced and asymmetric class distributions can significantly impact the performance of machine learning models. One critical aspect affected by these issues is the reliability of class probability estimates, which are crucial for informed decision-making in healthcare applications. Instead of predicting class values directly for a classification problem, it can be more convenient to predict the probability of an observation belonging to each possible class. This research aims to address the challenges posed by imbalanced and asymmetric responses in health data classification by evaluating the effectiveness of recent calibration methods in improving class probability estimates. We propose Beta calibration techniques and the Stratified Brier score and Jaccard's Score as novel calibration methods and evaluation metrics respectively. The experimental comparison involves implementing and assessing various calibration techniques to determine their impact on model performance and calibration accuracy of simulated and healthcare datasets with varying imbalance ratios. Our results show that the Beta calibration method consistently improved the classifiers' predictive ability. The findings of this study provide valuable insights into selecting the most suitable calibration method for enhancing class probability estimates in healthcare-related machine learning tasks.

Keywords: Imbalanced Data; Machine Learning; Jaccard Score; Calibration; Classification

1. Introduction

In the field of machine learning, model calibration is a crucial procedure in the creation of classification models because it improves their accuracy and reliability. Class probability estimates are intuitively interpretable and useful for estimating expected costs of classification decisions and in general offer more granular information than mere class membership predictions. They are especially useful when decisions have associated costs and experts would like to incorporate uncertainty into decisions made in such contexts. Class imbalance is the scenario in which the number of instances from each class is extremely unequal (Fu *et al.,* 2023). Imbalance is common in real-world learning tasks, like fraud detection and medical applications. In imbalanced scenarios, misclassification costs tend to be asymmetric (Ali *et al.,* 2013).

Healthcare data often exhibit imbalanced and asymmetric class distributions, where certain medical conditions or outcomes are rare or underrepresented. In such scenarios, traditional machine learning models may struggle to provide accurate class probability estimates, leading to sub-optimal decision-making and potentially compromising patient care. Calibration methods, designed to improve the reliability of class probability estimates, have shown promising results in addressing this challenge (Mukhiddinov *et al.,* 2022). However, there is a need for a comprehensive comparison of various calibration techniques to identify the most effective approach for health data classification tasks. This research aims to fill this gap by conducting an experimental evaluation of recent calibration methods in the context of imbalanced and asymmetric health data.

Although learning about well-calibrated models for health analysis is important, it has not been extensively researched in the literature (Dukhi *et al.,* 2021; More & Rana, 2021). In data science, modelers do not always care about the probability associated with each prediction, but there are cases in which we want to know the exact predictive probabilities, especially in medical diagnosis. Uncalibrated probability predictions can be biased for imbalanced data. Using machine learning models that predict probabilities is generally preferred when working on imbalanced (asymmetric) classification tasks (Zhou *et al.,* 2023). In imbalanced scenarios, it is important to have a model with good accuracy that is also calibrated. Despite the attention that classification under imbalance has received, only a few works have investigated the reliability of class membership probability estimates for imbalanced data in the literature (Alfhaid & Abdullah, 2021). Predicting probabilities allows some flexibility including deciding how to interpret the probabilities, presenting predictions with uncertainty, and providing more nuanced ways to evaluate the skill of the model. Predicted probabilities that match the expected distribution of probabilities for each class are often referred to as calibrated probabilities.

This present study focuses on calibrated probability estimators, which transform raw scores from classifiers into probability estimates. Many machine learning models can predict probability-like scores for asymmetric classes of data. However, the probabilities provide a required level of granularity for evaluating and comparing models, especially on imbalanced classification problems where the ROC-AUC metric is commonly used to compare model performance. Worse still, the severely skewed class distribution in imbalanced classification tasks may result in even more bias in the predicted probabilities as they over-favor the majority class (Wallace & Dahabreh, 2014). In this work, we demonstrate that class probability estimates obtained via supervised learning in imbalanced scenarios systematically underestimate the chances for minority class instances, despite ostensibly good overall calibrations. We show that calibrated probability estimators produce biased estimates in imbalanced scenarios, especially in medical scenarios (Ugarković & Oreški, 2022). We propose three novel metrics for this study, namely the stratified Brier Score (SBS), the Jaccard Score (JS), and Matthew's Correlation Coefficient (MCC), to capture class-specific calibrations that can assess the discriminative performance of classifiers in imbalanced classification problems.

2. Materials and Methods

2.1 Supervised Machine Learning Models

We focus on six supervised classification algorithms in this work. These models were selected because they have been shown to outperform other supervised learning algorithms with respect to class probability estimation in the literature. The models considered are briefly described in this section. More details about these models can be found in the cerebral work of Hastie *et al.,* 2009.

2.1.1 Logistic Regression (LR)

Logistic regression is a statistical method used in machine learning to predict the probability of a binary outcome. The LR is frequently used when attempting to predict whether an observation corresponds to one of two classes (as in the classification problems). The logistic function used in logistic regression models the connection between the predictor variables and the likelihood of belonging to a specific class. van den Goorbergh *et al.,* 2022 highlighted the fact that logistic regression can manage both continuous and categorical predictor variables, and it has become a popular technique for binary classification because it offers results that are easy to understand. Additionally, regularization methods can be easily incorporated into it, thereby enhancing their efficiency and avoiding overfitting. The logistic regression equation is given as

$$p = \frac{1}{1 + e^{-z}} \tag{1}$$

where p is the probability of the dependent variable taking on a certain value, z is the linear combination of the independent variables and their coefficients, and e is a constant known as Euler's number.

The variable z is given as

$$z = b_0 + b_1x_1 + b_2x_2 + ... + b_nx_n \tag{2}$$

where b_0 is the intercept term and $b_1, b_2, ..., b_n$ are the coefficients for each independent variable $x_1, x_2, ..., x_n$. The likelihood function for logistic regression coefficients is given as;

$$L(b) = \prod_{i=1}^{n} [p(i)^{y_i} * (1 - p(i))^{(1-y_i)}] \tag{3}$$

where yi is the observed value of the dependent variable for observation i and $p(i)$ is the predicted probability of yi taking on its observed value given the values of the independent variables. It has been popularly used for binary classification problems with imbalanced scenarios (Mahmudah *et al.,* 2021).

2.1.2 Naive Bayes [NB]

This is a simple yet powerful classification algorithm that can be used for both binary and multi-class classifications. Naive Bayes performs the classification task using the basic Bayes Theorem, that is, it classifies a new observation to a class that has the highest conditional probability for the particular observation amongst the target classes using prior information which can be gotten by using the Bayesian Network Classifiers below, subject to the assumption that the conditional probabilities exist:

$$P(y|\mathbf{x}) = \frac{P(y)P(\mathbf{x}|y)}{P(\mathbf{x})} \tag{4}$$

$$P(\mathbf{x}|y) = \Pi_{i=1}^{n} P(x_i|y) \tag{5}$$

hence,

$$P(\mathbf{x}) = \Pi_{i=1}^{k} P(c_i)P(\mathbf{x}|c_i) \tag{6}$$

where x_i is the value of the i^{th} attribute in $\boldsymbol{x}$, n is the number of attributes, k is the number of classes, and c_i is the i^{th} class. Naive Bayes has been reputed to have some interesting advantages such as it being computationally efficient because the time it takes for training is linearly related to both the training examples and number of attributes while the time it takes to classify new items is linearly related to the number of attributes and not determined by the training size. The variance of Naive Bayes is often low because it does not engage in searching, but it pays for the low variance with an increase in bias and it learns incrementally with continuous updating as new training data are supplied, moving from low order probabilities to higher ones. It can predict the posterior probability directly and it is also robust when there are missing values or noise since it uses all attributes in its prediction. The Naive Bayes classifier used in this study follows a Gaussian (normal) distribution which models each class's feature by estimating the mean and standard deviation.

2.1.3 Decision Tree (DT)

A decision tree is an approach used to break up a complex decision into multiple simpler decisions (Safavian & Landgrebe, 1991). This algorithm is one of the most successful data mining approaches. It has been widely employed in a variety of areas and it is robust even when they are missing values in the dataset (Hastie *et al.,* 2009). The algorithm collects observations of an object, represented by branches, and draws inferences about the problem goal value, represented by leaves. It may display the decisions in a tree-like structure. The tree is constructed by subdividing the target, which contains the tree's root node as well as its successive children. The basic concept of decision trees is the fundamental components of the model which are nodes and branches, and the most significant phases in developing the model are splitting, stopping, and pruning. The splitting of the trees is based on classification features splitting rules (Shalev-Shwartz & Ben-David, 2014). This algorithm is prone to high variance and overfitting of the imbalanced training data, which can be characterized by Entropy and Information Gain (Panigrahi *et al.,* 2021). Both Gini impurity and entropy are commonly used in data splitting for decision trees, and they often lead to similar trees. However, Entropy was used in this study as criteria for data splitting. Entropy is a measure of randomness in a dataset. It is defined as:

$$H(S) = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{7}$$

where S is the set of data, n is the number of classes, and p_i is the proportion of data belonging to class i. Entropy ranges from 0 (when all data belongs to one class) to 1 (when data is evenly distributed among all classes). Information gain tells us how much entropy is reduced after splitting the data based on an attribute. It is defined as:

$$IG(S, A) = H(S) - \sum_{v \in values(A)} \frac{|S_v|}{|S|} H(S_v) \tag{8}$$

where A is an attribute, $values(A)$ are its possible values, S_v is the subset of data where attribute A has value v, and $|S|$ and $|S_v|$ are the sizes of sets S and S_v, respectively.

2.1.4 Linear Discriminant Analysis (LDA)

Discriminant analysis (DA) is a multivariate technique used to separate two or more groups of observations (individuals) based on k variables measured on each experimental unit (sample) and find the contribution of each variable in separating the groups (Li *et al.,* 2023). The LDA and QDA perform as kernels that are used for the separation of classes by the hyperplane with the objective of an object within a class dependent on a discriminant rule with the highest probability.

The discriminant function is calculated as

$$D = a_1x_1 + a_2x_2 + \ldots + a_kx_k \tag{9}$$

where D is the discriminant score, $x_1, x_2, ..., x_k$ are the predictor variables, and $a_1, a_2, ..., a_k$ are the coefficients of the discriminant function. The coefficients are calculated as

$$a_i = \frac{(\mu_i \breve{\ } \mu)}{\sigma^2} \tag{10}$$

where μ is the overall mean of all predictor variables and σ^2 is the pooled variance across all groups (Awe *et al.,* 2023).

2.1.5 Support Vector Machine (SVM)

The SVM is a widely used algorithm in biological applications and one clear example is the automatic classification of microarray gene expression profiles. The algorithm is also used for image detection and recognization of fraudulent activities. SVM algorithm comprises the following basic concepts: a separating plane, a maximum-margin hyperplane, a soft margin, and finally the kernel functions (Noble, 2006). The basic idea behind SVM is to find a decision boundary that separates the data points into two classes with maximum margin. The margin is defined as the distance between the decision boundary and the closest data point from either class. The equation of a hyperplane in n-dimensional space can be written as:

$$w^T x + b = 0 \tag{11}$$

where x represents a vector of features that describes a point in the n-dimensional space, and b is the bias term.

In other to separate the data points into two classes with maximum margin, there is a need to solve an optimization problem:

$$minimize\ \frac{1}{2}||w||^2 \tag{12}$$

subject to

$$y_i(w^T x_i + b) \geq 1 \forall i \tag{13}$$

where y_i is the class label of data point i (+1 or -1), x_i is its feature vector, and $||w||^2$ is the squared Euclidean norm of w. After getting w and b, The predicted class label for a new data point x is given by:

$$y = sign(w^T x + b) \tag{14}$$

where $sign(w^T x + b)$ returns +1 or −1 depending on whether its argument is positive or negative.

In handling nonlinear separable problems, SVM uses kernels. A kernel function maps input vectors into a higher-dimensional feature space where they become linearly separable. The most used kernels are;

- Linear Kernel:

$$K(x_i, x_j) = x_i^T x_j \tag{15}$$

- Polynomial Kernel:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d \tag{16}$$

- Gaussian (RBF) Kernel:

$$K(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2) \tag{17}$$

2.1.6 K Nearest Neighbor (KNN)

KNN is a nonparametric model in machine learning that is used for classification where a k value is chosen for the classifications. The algorithm is a lazy learner technique that keeps all the training data for classification. In the process of choosing k's nearest neighbors, kNN computes the distance between all training samples and each test sample. Given a training set of N data points, each has d features as $X = x_1, x_2, ..., x_N$, where $x_i = (x_{i_1}, x_{i_2}, ..., x_{id})$ and corresponding labels given $Y = y_1, y_2, ..., y_N$, where $y_i \in C$ for classification or $y_i \in R$ for regression.

To classify a new data point x_q with an unknown label y_q; the following steps can be followed:

1. Calculate the distance between x_q and all other data points in X using a distance metric such as the following:
   - Euclidean distance:

$$d(x_i, x_q) = \sqrt{\sum_{j=1}^{d} (x_{ij} - x_{qj})^2} \tag{18}$$

   or
   - Manhattan distance:

$$d(x_i, x_q) = \sum_{j=1}^{d} |x_{ij} - x_{qj}| \tag{19}$$

2. Select K nearest neighbors of x_q based on their distances.
3. Consider the following cases:
   - For classification: Assign the label y_q to the majority class among the K nearest neighbors. $y_q = argmax_{c \in C} \sum_{[y_i=c]} [d(x_i, x_q)$

$$\leq d(x_j, x_q), j = 1, ..., K] \tag{20}$$

   where $[W]$ is an indicator function that returns 1 if W is true and 0 otherwise.
   - For regression: Assign the value y_q to the average of the values of the K nearest neighbors.

$$y_q = \frac{1}{K} \sum_{[d(x_i,x_q) \leq d(x_j,x_q), j=1,...,K]} y_i \tag{21}$$

   where y_i is the value of the *ith* nearest neighbor.

2.2 Calibration Methods

There are various methods for obtaining calibrated scores in a binary classification setting of which Platt scaling (Sigmoid) and Isotonic Regression are the most prominent. In addition, we propose a novel calibration method named Beta calibration (Kull *et al.*, 2017) in this study for characterizing health data.

2.2.1 Sigmoid Method

Sigmoid calibration or Platt scaling fits a one-variable logistic regression model to predict the outcomes from the scores. This approach was motivated by the idea that the relationship between the scores given by SVMs and the actual probabilities often fits well by the sigmoid function. As one would expect from a strict parametric approach, it works well when the data actually fit the model but gives poor results when the desired calibration function is not well-approximated by a sigmoid function. Assume that there is a logistic relationship between the z scores and the true probability p.

$$log(\frac{p}{1-p}) = \alpha + \beta z \tag{22}$$

Hence,

$$p = \frac{1}{1 + exp(-(\alpha + \beta z))} \tag{23}$$

where α and β are the regression constants. However, this method has a very restrictive set of possible functions. Its main restriction is that the mapping function is monotonically increasing (Pan *et al.,* 2022).

2.2.2 Isotonic Regression

Isotonic Regression fits a piecewise constant, monotonically increasing function to map the scores to probabilities.

Given a machine learning model with f_i as the predicted value and y_i as the actual, then we make an assumption as follows:

$$y_i = m(f_i) + e_i, \tag{24}$$

where m is an Isotonic function.

Given the training set (f_i, y_i), we try to find the isotonic function m by minimizing the following equation:

$$\hat{m} = argmin_z \sum (y_i - z(f_i))^2 \tag{25}$$

The Pool Adjacent Violators Algorithm (PAVA) is used to solve this problem. This method works reasonably well on a wider variety of problems due to its non-parametric approach (Allikivi & Kull, 2019). However, piecewise-constant approximations yield room for improvement due to their coarseness. Another limitation of these approaches is that they do not extend directly to a multi-class setting. Rather, we propose an indirect approach of fitting multiple binary classifiers and then calibrating and combining them (Naeini & Cooper, 2016).

2.2.3 Beta Calibration

We propose an improvement on Sigmoid and Isotonic regression for binary classifiers (Kull *et al.,* 2017).

Beta calibration produces a 3-parameter family of curves given as:

$$p = (1 + \frac{1}{(exp(c)(\frac{z^a}{(1-z)^b})^{-1})}) \tag{26}$$

where a, b, and c are parameters and z is a z score. It does not give misleading results if the model is already calibrated. We consider Beta calibration as a post-processing technique used to calibrate the predicted probabilities of the machine learning models in this study. It becomes particularly useful when the predicted probabilities do not align well with the actual proportions of positive outcomes in the dataset. In particular, Beta calibration aims to improve the reliability of predicted probabilities by adjusting them to better match the true proportions of positive outcomes. The goal of beta calibration is to enhance the calibration of predicted probabilities, making them more meaningful and reliable for decision-making, particularly when the model's original probabilities are poorly calibrated.

Below is a simplified algorithm we used for beta calibration:

1. Input the predicted probabilities from the machine learning model (original probabilities).

2. Calculate the Brier score (BS) for the original probabilities. The methodology for calculating the BS is shown in the next section.
3. Calculate the observed proportion of positive outcomes in the dataset by dividing the number of positive outcomes by the total number of instances.
4. Estimate the calibration parameters (alpha and beta) that will transform the original probabilities into the calibrated probabilities. These parameters determine how much the probabilities need to be adjusted to align with the observed proportions.
5. For each original predicted probability p, apply the calibration transformation to adjust the probability using equation (22).

$$calibrated_{p}robability = (p * \alpha)/(p * \alpha + (1 - p) * \beta)$$

   . This transformation ensures that the calibrated probabilities are scaled based on α and β to better match the observed proportions.
6. Calculate the Brier score for the calibrated probabilities to assess the improvement in the reliability of predicted probabilities.
7. Obtain calibrated probabilities, which have been adjusted to better align with the observed proportions. These are the calibration parameters (alpha and beta) that were used in the transformation.

2.3 Model Evaluation Metrics

For a binary classification problem, the confusion matrix comprises four results from classification outputs that report the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) predicted by a model. In the case of imbalanced data classification, alternative metrics that emphasize good performance concerning both classes are now widely accepted as more suitable for imbalanced data (e.g., the F1-score, g -mean, or MCC). However, the corresponding problem for probability estimation—good overall calibration masking unreliable estimates for minority instances—has not been addressed. We propose a modification of the Brier score that stratifies the score by class in this study. This is considered alongside other popular metrics highlighted briefly as follows:

2.3.1 F1-Score

This is the weighted average of precision and recall (sensitivity). It was adopted in this study because it is often considered more useful than accuracy because it combines both precision and sensitivity. It can be obtained by the formula:

$$F1 - Score = 2 * \frac{Precision * Sensitivity}{Precision + Sensitivity} \tag{27}$$

To minimize both false positive and false negative outcomes at the same time, precision and sensitivity can be summarized by using the F1-score, where precision is defined as:

$$Precision = \frac{TP}{TP + FP}$$

. These metrics are highly useful for imbalanced data classification.

2.3.2 Matthew's Correlation Coefficient

The Matthews correlation coefficient (MCC) offers an equitable assessment, suitable even when classes possess varying sizes. A coefficient value of 1.0 signifies impeccable prediction; a value of

0.0 implies a prediction no more accurate than randomness; and a value of -1.0 indicates the most unfavorable prediction conceivable. It harmoniously merges prediction accuracy and coverage, culminating in a high score only if the classifier excels in all four cells of the confusion matrix. In the domains of machine learning and biostatistics, researchers widely adopt MCC as a performance metric for evaluating binary classifiers, regardless of dataset balance. Unlike balanced accuracy, which overlooks low precision, MCC adeptly captures the modest positive connection between reality and prediction. It can be mathematically represented as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{28}$$

2.3.3 Brier Score (BS)

We view this metric as more appropriate for assessing calibration in imbalanced scenarios. The value of the error score is always between 0 and 1 where a model with a score of 0 is perfect. This means that the closer the value of the Brier score to 0, the better the prediction, and the closer its value to 1, the less accurate the prediction is. The Brier score reflects both calibration and and discrimination. The Brier score measures the mean squared difference between the predicted probability y assigned to the possible outcomes for item i, and the actual outcome y. Therefore, the lower the Brier score is for a set of predictions, the better the predictions are calibrated. It is analogous to the mean square error (MSE). The Brier score measures the accuracy of predicted probabilities. It is calculated as the mean squared difference between the predicted probabilities and the actual binary labels. The most common formulation of the Brier score (Rufibach, 2010) in literature is given as

$$BS = \frac{1}{N} \sum_{t=1}^{N} (p_t - a_t)^2 \tag{29}$$

where p_t is the probability that was forecasted, and a_i is the actual outcome of the event at instance t.

2.3.4 Jaccard Score

The Jaccard score, also known as the Jaccard index or Jaccard similarity coefficient, is a measurement in machine learning for evaluating how well a classification model performs, especially when dealing with imbalanced datasets. Mathematically, the Jaccard score (JS) is calculated by dividing the size of the intersection of the positive set and the actual positive set by the size of their union. It is represented as follows:

$$JS = \frac{TP}{TP + FP + FN} \tag{30}$$

Ranging from 0 to 1, the Jaccard score indicates how closely the predicted positive set aligns with the true positive set. A score of 1 indicates agreement, while a score of 0 means there is no overlap at all. This metric is particularly useful in situations where there is an imbalance between classes. It focuses on predicting positives while not considering true negatives. This proves advantageous when there is a difference in quantity between negative and positive classes, which often occurs with imbalanced datasets. Additionally, the Jaccard score remains reliable when faced with class imbalance.

2.3.5 Sensitivity (Recall)

This is the ability of a model to correctly identify patients with a disease. It is calculated as

$$Sensitivity = \frac{TP}{TP + FN} \tag{31}$$

where TP is true positive and FN is false negative.

2.3.6 Specificity

This is the metric used to judge if the classifier correctly predicts people without the disease in a health diagnosis study. It is calculated as

$$Specificity = \frac{TN}{FP + TN} \tag{32}$$

where TN is true negative, FP is false positive.

2.3.7 Accuracy

Accuracy measures how accurate the algorithm is. It is obtained as

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{33}$$

. It has been suggested not to be appropriate for imbalanced datasets.

2.3.8 Log Loss

Log loss is a common evaluation metric for binary classification models. It measures the performance of a model by quantifying the difference between predicted probabilities and actual values. It is indicative of how close the prediction probability is to the corresponding actual/true value (0 or 1 in the case of binary classification), penalizing inaccurate predictions with higher values. Lower log-loss indicates better model performance. Log Loss is the most important classification metric based on probabilities. Although it is difficult to interpret raw log-loss values, it is still a good metric for comparing calibrated models. For any given problem, a lower log-loss value means better predictions. Log Loss is the negative average of the log of corrected predicted probabilities for each instance. It provides a continuous and differentiable measure of the model's performance, making it suitable for optimization algorithms. The log loss penalizes confident and incorrect predictions more heavily, incentivizing calibrated probability estimates. Finally, log loss can be interpreted as the logarithmic measure of the likelihood of the predicted probabilities aligning with the true labels.

Mathematically,

$$Logloss = -\frac{1}{N}\sum_{N}^{i=1}(log(p_i)) \tag{34}$$

where p is the predicted probability and N is population size.

3. Results and Discussion

3.1 Experimental Setup

The main experimental setup of this research is to assess the effectiveness of six machine learning models – specifically K Nearest Neighbors (KNN) Naive Bayes (NB) Support Vector Machine (SVM), Decision Tree (DT) Linear Discriminant Analysis (LDA) and Logistic Regression (LR) – in predicting a binary target variable within a dataset that was oversampled. To maximize the predictive potential of these models, three calibration methods namely Isotonic Regression, Sigmoid, and Beta calibration were employed. Throughout this evaluation process, we extensively used various evaluation metrics to gain an understanding of how well these models performed. These metrics covered a range of aspects, including accuracy measurements and more nuanced indicators like

Stratified Brier Score, F1 Score, Precision, Recall, ROC-AUC, Log Loss, MCC, and Jaccard Score. By employing this set of metrics we were able to evaluate the models' predictive capabilities in detail in the presence of various calibration methods.

Performances of these classifiers on unsampled and oversampled data using the aforementioned metrics, validate the need for feature selection and resampling to avoid overfitting and gross inaccuracy. The Confusion matrices for the original dataset across all six classifiers showed gross inaccuracy in their predictions (Figure 2). The sum of the right diagonal of each matrix which represents the incorrect prediction is higher than the sum of the left diagonal which represents the correct prediction. This shows the poor predictive abilities of our model trained with unsampled data which begs the need for a structured sampling method. Therefore, the Synthetic Minority Oversampling Technique (SMOTE) was applied to the simulated and real health data to train the classifiers for correct predictions. Addressing the class imbalance in datasets is a critical challenge in machine learning, particularly when one class is significantly underrepresented compared to others, as it is in this study. SMOTE plays a pivotal role in methodologies dealing with such imbalanced datasets. This technique becomes necessary due to the inherent biases and suboptimal predictive performance that can result from models trained on imbalanced data.

The primary purpose of SMOTE is to rectify class imbalance by oversampling the minority class, usually the underrepresented class in the dataset (Fernández *et al.,* 2018) . It achieves this by generating synthetic samples that closely resemble the existing minority class instances, thereby balancing the class distribution. By creating synthetic samples through interpolation between existing minority class instances, SMOTE enriches the dataset without simply duplicating data points. This process effectively provides the model with a more balanced representation of both classes. One of the key advantages of using SMOTE is its ability to enhance model performance. Imbalanced datasets can cause machine learning algorithms to prioritize the majority class, leading to biased models that perform poorly in predicting the minority class. By synthesizing new samples for the minority class, SMOTE ensures that the model learns equally from both classes, reducing bias and improving its ability to generalize to new, unseen data. Moreover, SMOTE helps mitigate the risk of overfitting that might occur if the model were trained solely on duplicated minority class samples. It allows the model to learn from a more diverse set of instances, preserving the intrinsic characteristics and patterns present in the original data. Additionally, SMOTE's compatibility with various machine learning algorithms makes it a versatile and widely used technique in addressing class imbalance. It seamlessly integrates into different methodologies, ensuring that models trained on imbalanced datasets can benefit from a more representative and balanced distribution of classes.

We used two datasets with similar degrees of imbalance in this study. Specifically, one simulated dataset and a real-life health data on anemia. We split each of these into train, test, and validation sets of 60, 20, and 20 divides, respectively. We induced probability estimators via calibration over the train sets using six distinct classifiers: SVM, KNN, NB, LDA, LR, and DT which were selected due to their popularity and demonstrated performance in accurately estimating probabilities (Wallace & Dahabreh, 2014). To measure the probability estimation performance, we recorded overall Brier scores for both unsampled and oversampled datasets. We report the results achieved using standard Platt scaling versus those achieved by oversampling prior to scaling for both simulated and anemia datasets. We also show the results of calibrated over independently drawn balanced bootstrap samples of the training datasets. The test set is used to evaluate the performance of the classifiers in order to obtain sustainable and reliable results (Flach, 2019).

3.2 Simulation Study

A simulated experiment was carried out using simulated data from 500 Bernoulli-distributed samples, where each sample contains 20 features. These results were recorded in Table 1. To intentionally create an imbalance, one class accounts for 90% of the samples while the other class makes up the remaining 10%. Out of all the features, there are 10 that provide information and help distinguish between classes. Additionally, there are five features introduced as noise. A class separator parameter that separates the classes from each other was introduced, and approximately 5% of the samples have labels that were randomly assigned. Each class is represented by n clusters per class, which emphasizes the distinction between classes.

Table 1. Results on Simulated Data

Classifier	Performance Metrics									
	Calibration Method	ROC AUC	F1 Score	Jaccard Score	Brier Score	Log Loss	ACC	MCC	PREC	REC
LR	Uncalibrated	0.6429	0.3333	0.2000	0.2500	8.6348	0.7500	0.2182	0.2500	0.5000
	Sigmoid	0.7429	0.3509	0.2128	0.1420	0.4459	0.7897	0.2421	0.2857	0.4545
	Isotonic	0.7486	0.3846	0.2380	0.1382	0.7839	0.8182	0.2856	0.3333	0.4545
	Beta	0.7651	0.3810	0.2353	0.1440	0.5573	0.7330	0.1723	0.2222	0.4545
NB	Uncalibrated	0.6558	0.3188	0.2273	0.1932	6.6724	0.8068	0.2673	0.3125	0.4545
	Sigmoid	0.7836	0.3704	0.2273	0.1346	0.4276	0.8068	0.2673	0.3125	0.4545
	Isotonic	0.7757	0.4390	0.2812	0.1130	0.3769	0.8693	0.3667	0.4737	0.4091
	Beta	0.7857	0.4651	0.3030	0.1125	0.3650	0.8693	0.3909	0.4762	0.4545
LDA	Uncalibrated	0.6331	0.3188	0.1897	0.2670	9.2236	0.7330	0.1990	0.2340	0.5000
	Sigmoid	0.7538	0.3548	0.2157	0.1446	0.4453	0.7727	0.2459	0.2750	0.5000
	Isotonic	0.7516	0.3793	0.2240	0.1490	0.8039	0.7954	0.2768	0.3056	0.5000
	Beta	0.7547	0.3667	0.2245	0.1469	0.4603	0.7841	0.2610	0.2895	0.5000
SVM	Uncalibrated	0.7792	0.5833	0.4118	0.1136	1.8179	0.8864	0.5206	0.5385	0.6364
	Sigmoid	0.8450	0.5000	0.3333	0.0867	0.2889	0.8864	0.4394	0.5556	0.4545
	Isotonic	0.8438	0.4615	0.3000	0.0845	0.7904	0.8807	0.3998	0.5294	0.4091
	Beta	0.8099	0.4444	0.2857	0.1001	2.5459	0.8864	0.3968	0.5714	0.3636
DT	Uncalibrated	0.6429	0.3446	0.2083	0.2159	7.4574	0.7841	0.2343	0.2778	0.4545
	Sigmoid	0.6886	0.3673	0.2250	0.1244	0.4091	0.8238	0.2681	0.3333	0.4091
	Isotonic	0.6874	0.3721	0.2286	0.1208	0.3978	0.8466	0.2848	0.3809	0.3636
	Beta	0.6494	0.3571	0.2174	0.2045	5.2132	0.7955	0.2502	0.2941	0.4545
KNN	Uncalibrated	0.7532	0.4688	0.3061	0.1932	6.6724	0.8068	0.3930	0.3571	0.6818
	Sigmoid	0.7975	0.4081	0.2564	0.1105	0.4014	0.8352	0.3158	0.3703	0.4545
	Isotonic	0.7841	0.4186	0.2647	0.1079	1.3994	0.8580	0.3379	0.4285	0.4091
	Beta	0.8099	0.4444	0.2857	0.1001	2.5459	0.8864	0.3968	0.5714	0.3636

The simulated imbalanced data from a Bernoulli distribution is defined by: $X \sim \text{Bernoulli}(p)$ where X represents the simulated data, and p is the probability of generating data from the minority class. The probability mass function (pmf) of the Bernoulli distribution is given as $f(x;p) = p^x(1-p)^{1-x}$ where x is a binary random variable that takes the value 1 with probability p and the value 0 with probability $1-p$. By adjusting the value of p, we generate imbalanced data with a smaller proportion of 1's (the minority class) and a larger proportion of 0's (the majority class). To generate imbalanced data, we set $p = 0.1$, then 10% of the data generated is from the minority class, while 90% is from the majority class. The selected aforementioned classifiers were employed to predict the instances in the oversampled data. This was done to understand how the models perform using evaluation metrics that describe the predictive capabilities of the classifiers. For LR, the Isotonic and Beta Calibration improved the ROC-AUC score from 0.6429 to 0.7486 and 0.7651 respectively.

Brier Score, Jaccard Score, and MCC were also improved by a significant percentage. The log loss improved from 8.6348 to 0.7839 and 0.5573 using Isotonic and Beta calibration methods respectively. Beta Calibration enhanced the NB classifier across the evaluation metrics employed. It greatly reduced the log loss from 6.6724 to 0.3650; the Brier score from 0.1932 to 0.1125; the ROC-AUC from 0.6558 to 0.7857. LDA was greatly improved using Isotonic and Beta calibration but Beta calibration had greater performance as the log loss was reduced from 9.2236 to 0.4603, ROC-AUC increased from 0.6331 to 0.7547, and Brier score was reduced from 0.2670 to 0.1469.

Also, SVM greatly improved across the three calibration methods namely Sigmoid, Isotonic, and Beta calibrations as can be seen in Table 1. However, Isotonic greatly enhanced all metric scores significantly, revealing a great predictive capability for an Isotonic calibrated model employed on oversampled data. It is also worthy of note that while the Beta calibration method increased the log loss from 1.8179 to 2.5459, it performed excellently across other evaluation metrics. Only Isotonic Regression could improve the DT classifier across all metrics. Even though, the ROC-AUC scores were just fairly improved from 0.6429 to 0.6874 while log-loss was reduced from 0.2159 to 0.1208. Beta greatly enhanced the metrics significantly. The Brier scores for both calibrations were reduced from 0.1105 to 0.1001. Even though, this method barely reduced the log-loss from 6.6724 to 2. 5459 and 2.6257 respectively. In summary, the calibration methods employed to enhance the model performances across all classifiers showed that the Beta calibration method has a consistently improved impact on the classifiers' predictive ability. This experiment showed that the Beta calibration, introduced in this study as an emerging calibration method performed greatly using key evaluation metrics across all classifiers. It performed better and in some cases alongside, Isotonic and Sigmoid Calibration methods. This result is in agreement with the work of Kull *et al.,* 2017. These results highlight the usefulness of using simulated data, for model assessments in situations where there is limited availability of real-world data.

3.3 Results from Anemia Dataset

However, in order to validate our claims, we apply these methods using the evaluation metrics aforementioned on real health data. The results of this performance across all classifiers are recorded in Table 2. For LR, ROC-AUC improved from 0.6491 to 0.7915 and 0.7818, log loss reduced from 13.1769 to 0.5470 and 0.5678, Brier score was reduced from 0.3815 to 0.1835 and 0.1942 for Isotonic and Beta calibration respectively. Naive Bayes also improved across all metric scores for both Beta and Isotonic calibration. Its log loss reduced significantly for both Calibrations from 13.8158 to 0.5810 and 0.5794 respectively (Fig. 1, 2).

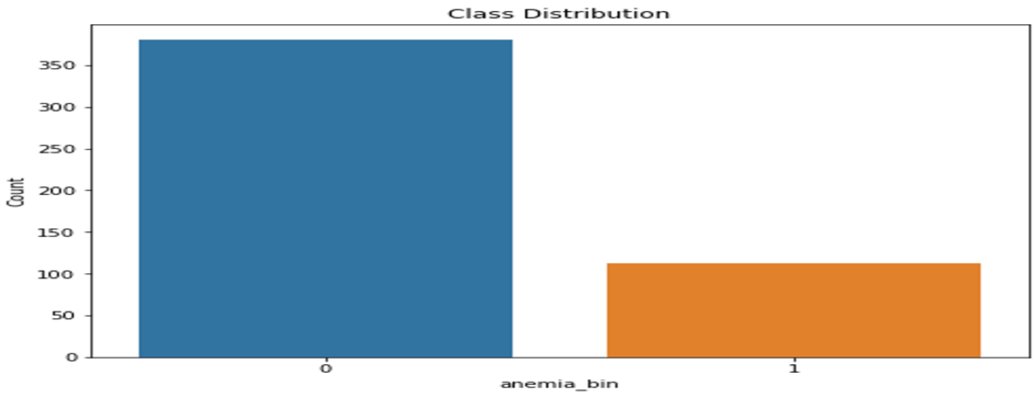

Figure 1. Target Variable.

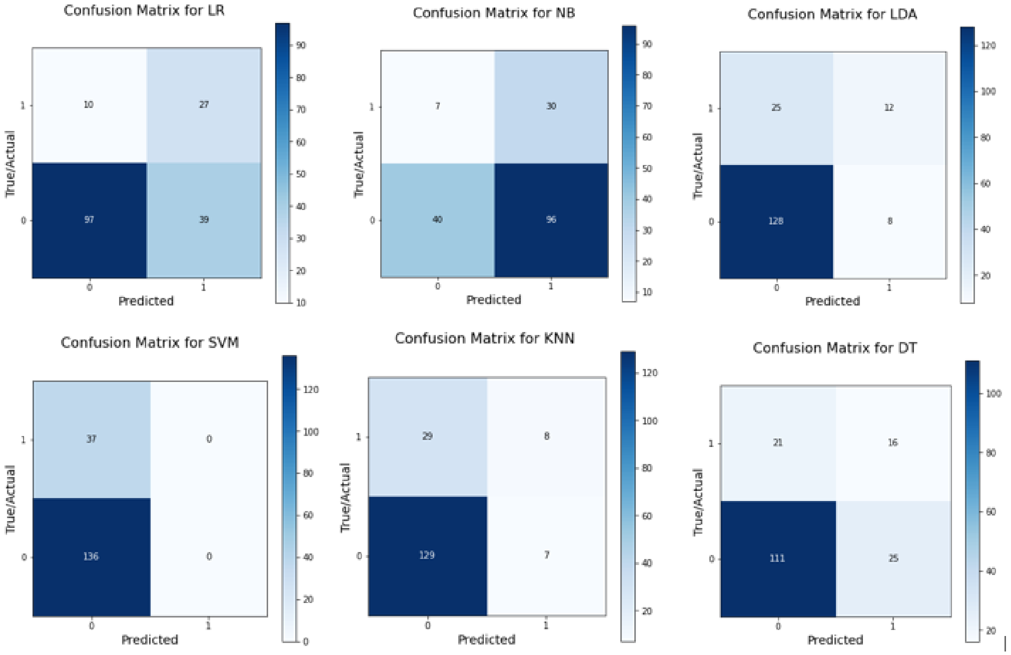

Figure 2. Anemia Data Confusion Matrices.

LDA was also improved across all evaluation metrics for both Beta and Isotonic calibration. Its predictive power was enhanced by both methods and validated by its increase on ROC-AUC from 0.6761 to 0.7985 and 0.8031 respectively. SVM, DT and KNN were also enhanced using Beta calibration yielding a significant improvement in the key performance metrics as recorded in Table 2.

Table 2. Results on Anemia Data

Classifier		Performance Metrics								
	Calibration Method	ROC AUC	F1 Score	Jaccard Score	Brier Score	Log Loss	ACC	MCC	PREC	REC
LR	Uncalibrated	0.6491	0.4407	0.2826	0.3815	13.1769	0.6185	0.2451	0.6923	0.7027
	Sigmoid	0.7833	0.4485	0.2892	0.1961	0.5718	0.6590	0.2593	0.3429	0.6486
	Isotonic	0.7915	0.5714	0.4000	0.1835	0.5470	0.8092	0.4495	0.5500	0.5946
	Beta	0.7818	0.4673	0.3049	0.1942	0.5678	0.6705	0.2880	0.3571	0.6757
NB	Uncalibrated	0.6848	0.4434	0.3093	0.3873	13.8158	0.6127	0.3034	0.3333	0.8108
	Sigmoid	0.7442	0.4786	0.3146	0.2158	0.6176	0.6473	0.3079	0.3500	0.7568
	Isotonic	0.7500	0.4776	0.3137	0.1897	0.5810	0.7977	0.3568	0.5333	0.4324
	Beta	0.7421	0.4742	0.3108	0.1958	0.5794	0.7052	0.3011	0.3833	0.6216
LDA	Uncalibrated	0.6761	0.4717	0.3086	0.3237	11.1804	0.6763	0.2949	0.3623	0.6757
	Sigmoid	0.7982	0.4630	0.3012	0.1911	0.5592	0.6647	0.2813	0.3521	0.6757
	Isotonic	0.8031	0.5121	0.3443	0.1777	0.5279	0.7687	0.3655	0.4667	0.5676
	Beta	0.7985	0.4706	0.3077	0.1871	0.5518	0.6879	0.2639	0.3692	0.6486
SVM	Uncalibrated	0.6747	0.4819	0.3175	0.2486	8.5849	0.7514	0.3242	0.4348	0.5405
	Sigmoid	0.7444	0.4883	0.3231	0.1720	0.5592	0.7457	0.3291	0.4286	0.5676
	Isotonic	0.7526	0.5208	0.3521	0.1827	0.5674	0.7341	0.3681	0.4237	0.6757
	Beta	0.7550	0.4026	0.2520	0.4412	1.8391	0.4682	0.1801	0.2650	0.8378
DT	Uncalibrated	0.6489	0.4444	0.2857	0.2890	9.9825	0.7110	0.2650	0.3774	0.5405
	Sigmoid	0.6597	0.3636	0.2222	0.1776	0.5361	0.7572	0.2188	0.4138	0.3432
	Isotonic	0.6397	0.2769	0.1607	0.1835	0.5486	0.7283	0.1153	0.3214	0.3636
	Beta	0.5801	0.3421	0.2063	0.2890	7.3661	0.7110	0.1572	0.3333	0.3512
KNN	Uncalibrated	0.6454	0.4381	0.2805	0.3410	11.7793	0.6590	0.2441	0.3382	0.6216
	Sigmoid	0.7048	0.4197	0.2656	0.1779	0.5409	0.7283	0.2457	0.3864	0.4595
	Isotonic	0.7033	0.3562	0.2167	0.1796	0.5581	0.7283	0.1841	0.3611	0.3514
	Beta	0.7198	0.4410	0.2586	0.1941	3.1326	0.7514	0.2535	0.4167	0.4054

4. Discussion

The calibration results highlighted in Tables 1 and 2 represent those with high performances across all the model evaluation metrics, indicating improvements in the uncalibrated model performances. The Beta calibration method introduced in this study as an emerging calibration method performed notably well across all classifiers both in simulated and real health data. It is worthy of note that calibration methods that performed well on each classifier in simulated data remained consistent on real health data. It also reveals the potency of the Beta calibration method as an efficient way of improving model performance. Naive Bayes scores improved after calibration and produced better results with Isotonic Regression than Platt/Sigmoid Scaling. SVM demonstrated some stability in terms of probabilities and higher reliability. We can also observe that the F1 Score of SVC + Sigmoid is better than that of SVC + Isotonic. Platt Scaling is most effective when the distortion in the predicted probabilities is sigmoid-shaped while Isotonic Regression is a more powerful calibration method that can correct any monotonic distortion although it is prone to over-fitting. In summary, SVM, LDA, and DT classifiers were the best classifiers due to their accuracy, ROC-AUC values, and well-calibrated probabilities. While accuracy is a popular metric for evaluation, it is crucial to consider other metrics like precision, recall, F1 score, and MCC to fully understand the implications of positives and false negatives in the context of imbalanced data (Figures 3 and 4).

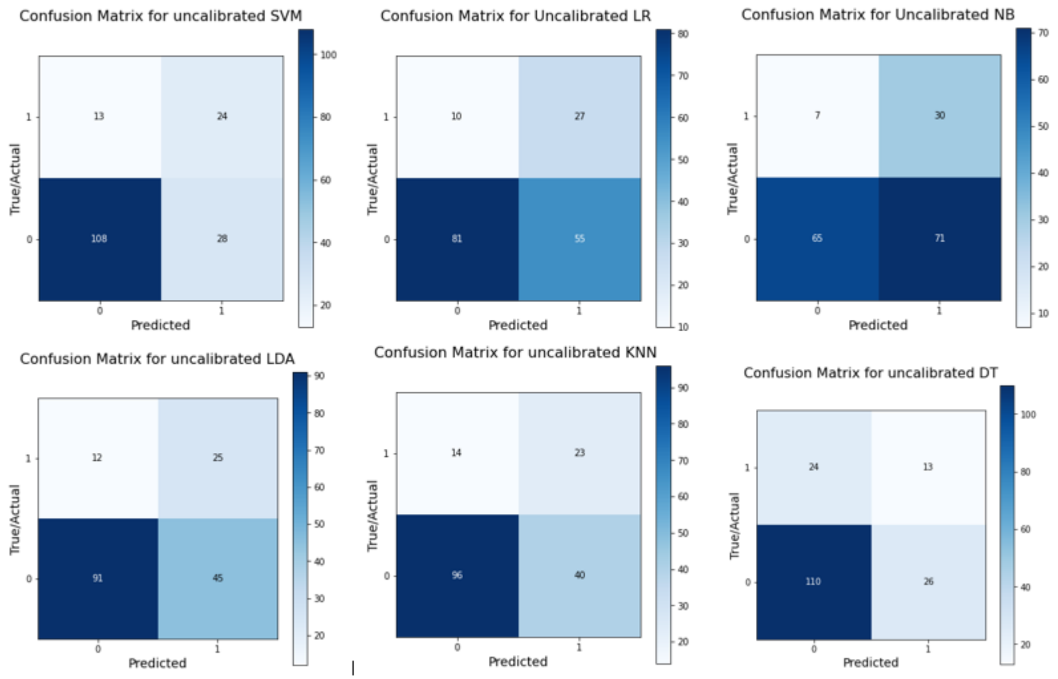

Figure 3. Uncalibrated Confusion Matrices.

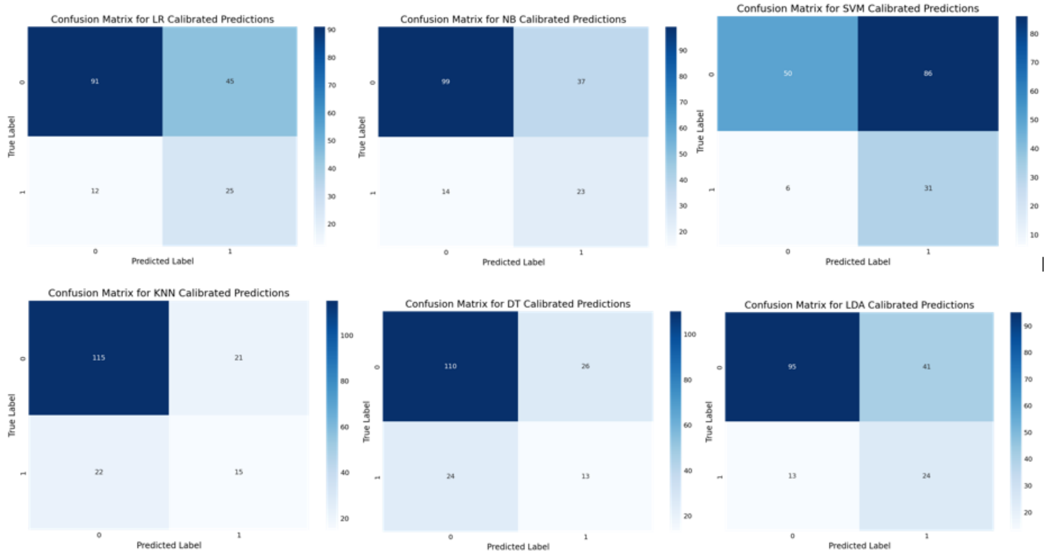

Figure 4. Calibrated Confusion Matrices.

The comparison of how the classifiers performed on both simulated and real-life data emphasizes that the ROC-AUC values align closely. Overall, when comparing the performances of models trained on real-life data, we observe alignment in the classifier's performances on both real data and simulated data using ROC-AUC values as the referenced performance metrics. The findings indicate that the behavior of the data closely resembles real-life scenarios, which provides an argument for using it in initial model assessments or when actual data is scarce.

4.1 Friedman's Tests

The performance evaluation of algorithms, across the datasets, involves the use of the Friedman test, which is a statistical non-parametric test that does not depend on any specific assumptions about the data. This test was employed to assess metrics such as ROC-AUC, F1 Score, Jaccard Score, Brier Score, Log Loss, ACC, MCC, PREC and REC. It is particularly valuable when the data does not meet the requirements of tests like having a normal distribution. To analyze the results of the Friedman test we compared the p-value of each metric with a significant level of 0.05. The result of our Friedman test is shown in Table 3.

Table 3. Test Statistics for Friedman Test

	Metrics	Test Statistic	p-value
1	Roc Auc	11.3571	0.0779
2	F1 Score	10.0714	0.1217
3	Jaccard Score	11.0357	0.0872
4	Brier Score	12.6188	0.0495
5	Log Loss	11.8654	0.0650
6	Acc	4.3857	0.6246
7	Mcc	11.0357	0.0872
8	Prec	7.7143	0.2598
9	Rec	9.9821	0.1254

Based on our findings from this analysis, we observe a difference in performance for the Brier Score metric. Its p-value was calculated to be 0.049504. On the other hand, all other metrics had p-values greater than 0.05 implying that there were no significant differences in their performance across all models tested. Specifically looking at ROC-AUC, F1 Score, Jaccard Score and MCC metrics showed p-values of 0.05 indicating some variations in model performance without reaching the statistical significance threshold. Furthermore; Log Loss, ACC, PREC and REC metrics had p values indicating even less evidence supporting significant differences, in their performance. To sum up, based on the results of the Friedman test it was found that only the Brier Score metric displayed a variation, among all the models. The absence of disparities in the metrics could be attributed to similar performance across all models.

5. Conclusions

In this work, we have presented Beta calibration, a fairly novel approach to calibration that improves on previous methods in several respects. It transforms the data to provide appropriate scaling for "over-confident" models and it was identified as the most effective calibration method for improving class probability estimates in health data classification with imbalanced and asymmetric responses. Our work highlights the potential implications and applications of novel calibration techniques in real-world healthcare scenarios. There are future research directions to further enhance class probability estimation in health data classification. For instance, it is unknown how to choose the right amount of data for calibration. Finally, there may be more efficient and clever ways to obtain a calibration like the brute-force cross-validation approach, which would greatly improve the efficiency of the calibration methods. Spline-based calibration methods would also be examined in our future studies. The findings from this study provide valuable insights into selecting the most suitable calibration methods for enhancing class probability estimates in healthcare-related machine learning tasks.

Acknowledgments

The authors gratefully acknowledge and thank FAPESP for the grant provided to conduct this study (Grant number 21/05136-4-Analise de dados-BV FAPESP).

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Alfhaid, M. A. & Abdullah, M. Classification of imbalanced data stream: Techniques and challenges. *Artificial Intelligence* **9,** 36–52 (2021).
2. Ali, A., Shamsuddin, S. M. & Ralescu, A. L. Classification with class imbalance problem. *Int. J. Advance Soft Compu. Appl* **5,** 176–204 (2013).
3. Allikivi, M.-L. & Kull, M. *Non-parametric Bayesian isotonic calibration: Fighting over-confidence in binary classification* in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (2019), 103–120.
4. Awe, O. O., Dukhi, N. & Dias, R. Shrinkage heteroscedastic discriminant algorithms for classifying multi-class high-dimensional data: Insights from a national health survey. *Machine Learning with Applications* **12,** 100459 (2023).

5. Dukhi, N., Sewpaul, R., Sekgala, M. D. & Awe, O. O. Artificial intelligence approach for analyzing anemia prevalence in children and adolescents in BRICS countries: a review. *Current Research in Nutrition and Food Science Journal* **9,** 01–10 (2021).

6. Fernández, A., Garcia, S., Herrera, F. & Chawla, N. V. SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research* **61,** 863–905 (2018).

7. Flach, P. *Performance evaluation in machine learning: the good, the bad, the ugly, and the way forward* in *Proceedings of the AAAI conference on artificial intelligence* **33** (2019), 9808–9814.

8. Fu, S., Su, D., Li, S., Sun, S. & Tian, Y. Linear-exponential loss incorporated deep learning for imbalanced classification. *ISA Transactions* (2023).

9. Hastie, T., Tibshirani, R., Friedman, J. H. & Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction* (Springer, 2009).

10. Kull, M., Silva Filho, T. & Flach, P. *Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers* in *Artificial Intelligence and Statistics* (2017), 623–631.

11. Li, S., Zhang, H., Ma, R., Zhou, J., Wen, J. & Zhang, B. Linear discriminant analysis with generalized kernel constraint for robust image classification. *Pattern Recognition* **136,** 109196 (2023).

12. Mahmudah, K. R., Indriani, F., Takemori-Sakai, Y., Iwata, Y., Wada, T. & Satou, K. Classification of Imbalanced Data Represented as Binary Features. *Applied Sciences* **11,** 7825 (2021).

13. More, A. S. & Rana, D. P. in *Data Preprocessing, Active Learning, and Cost Perceptive Approaches for Resolving Data Imbalance* 1–22 (IGI Global, 2021).

14. Mukhiddinov, M., Muminov, A. & Cho, J. Improved classification approach for fruits and vegetables freshness based on deep learning. *Sensors* **22,** 8192 (2022).

15. Naeini, M. P. & Cooper, G. F. *Binary classifier calibration using an ensemble of near isotonic regression models* in *2016 IEEE 16th International Conference on Data Mining (ICDM)* (2016), 360–369.

16. Noble, W. S. What is a support vector machine? *Nature biotechnology* **24,** 1565–1567 (2006).

17. Pan, Z., Gu, Z., Jiang, X., Zhu, G. & Ma, D. A modular approximation methodology for efficient fixed-point hardware implementation of the sigmoid function. *IEEE Transactions on Industrial Electronics* **69,** 10694–10703 (2022).

18. Panigrahi, R., Borah, S., Bhoi, A. K., Ijaz, M. F., Pramanik, M., Kumar, Y. & Jhaveri, R. H. A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets. *Mathematics* **9,** 751 (2021).

19. Rufibach, K. Use of Brier score to assess binary predictions. *Journal of clinical epidemiology* **63,** 938–939 (2010).

20. Safavian, S. R. & Landgrebe, D. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics* **21,** 660–674 (1991).

21. Shalev-Shwartz, S. & Ben-David, S. *Understanding machine learning: From theory to algorithms* (Cambridge university press, 2014).

22. Ugarković, A. & Oreški, D. *Supervised and Unsupervised Machine Learning Approaches on Class Imbalanced Data* in *2022 International Conference on Smart Systems and Technologies (SST)* (2022), 159–162.

23. Van den Goorbergh, R., van Smeden, M., Timmerman, D. & Van Calster, B. The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression. *Journal of the American Medical Informatics Association* **29,** 1525–1534 (2022).

24. Wallace, B. C. & Dahabreh, I. J. Improving class probability estimates for imbalanced data. *Knowledge and information systems* **41,** 33–52 (2014).

25. Zhou, Q., Qi, Y., Tang, H. & Wu, P. Machine learning-based processing of unbalanced data sets for computer algorithms. *Open Computer Science* **13,** 20220273 (2023).