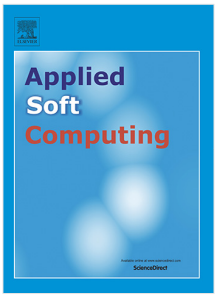
Journal Pre-proof



Handling imbalanced data for aircraft predictive maintenance using the BACHE algorithm

Maren David Dangut, Zakwan Skaf, Ian K. Jennions

PII: S1568-4946(22)00281-2
DOI: https://doi.org/10.1016/j.asoc.2022.108924
Reference: ASOC 108924

To appear in: *Applied Soft Computing*

Received date : 5 December 2020
Revised date : 9 February 2022
Accepted date : 19 April 2022

Please cite this article as: M.D. Dangut, Z. Skaf and I.K. Jennions, Handling imbalanced data for aircraft predictive maintenance using the BACHE algorithm, *Applied Soft Computing* (2022), doi: https://doi.org/10.1016/j.asoc.2022.108924.

Applied Soft Computing, Volume 123, July 2022, Article number 108924
DOI:10.1016/j.asoc.2022.108924

Handling Imbalanced Data for Aircraft Predictive Maintenance using the BACHE Algorithm

***Maren David Dangut,** Zakwan Skaf, ***Ian K. Jennions**

*,*** Integrated Vehicle Health Management (IVHM) Center, Cranfield University, Bedford, United

**Mechanical Engineering Department, Higher Colleges of Technology, United Arab Emirates

(*maren.dangut@cranfield.ac.uk, ** zskaf@hct.ac.ae, ***i.jennions@cranfield.ac.uk)

Abstract

Developing a prognostic model to predict an asset's health condition is a maintenance strategy that increases asset availability and reliability through better maintenance scheduling. Therefore, developing reliable vehicle health predictive models is vital in the aerospace industry, especially considering a safety-critical system such as aircraft. However, one of the significant challenges faced in building reliable data-driven prognostic models is the imbalance dataset. Training machine-learning models using an imbalanced dataset causes classifiers to be biased towards the class with majority samples, resulting in poor predictive accuracy in data-driven models. This problem can become more challenging if the imbalance ratio is extreme and classes overlap. In this paper, a novel approach called Balanced Calibrated Hybrid Ensemble Technique (BACHE) is developed to tackle the severe imbalanced classification problem. The proposed method involves the combination of hybrid data sampling and ensemble-based learning. It uses a cascading balanced approach to transfer a class imbalance problem into a sub-problem by decomposing the original problem into a set of subproblems, each characterized by a reduced imbalance ratio. Then uses a calibrated boosting with a cost-sensitive decision tree to enhance recognition of hard-to-learn patterns, which improves the prediction of the extreme minority class. BACHE is evaluated using a real-world aircraft dataset with rare component replacement instances. Also, a comparative experiment of the proposed approach with other similar existing methods is conducted. The performance metrics used are precision, recall, G-mean, and an area under the curve. The final results show that the proposed model outperforms other similar methods. Also, it can attain an excellent performance on large, extremely imbalanced datasets.

Keywords: Prognostic, Imbalanced learning, Ensemble learning, Predictive maintenance, Aerospace.

Abbreviations

Aircraft Communications Addressing and Reporting System	ACARS
Aircraft Condition Monitoring System	ACMS
Avionics Equipment Ventilation Computer	10HQ
Air traffic Service Unit	1TX1
A330 –Long-Range Aircraft	LR
A320 -Single-Aisle Aircraft	SA
Balanced Calibrated Hybrid Ensemble Technique	BACHE
Built-In Test Equipment	BITE
Central Maintenance System	CMS
Conditioned-Based Maintenance	CBM
Electronic Control Unit/ Electronic Engine Unit	4000KS
Flight Warning Computers	FWCs
Flight Deck Effect	FDE
Functional Item Number	FIN
High-Pressure Bleed Valve	4000HA
Imbalanced Ratio	IR
Line Replacement Unit	LRU
Overall Equipment Effectiveness	OEE
Pressure Regulating Valve	4001HA
Satellite Data Unit	5RV1
Synthetic Minority Oversampling Techniques	SMOTE
Trim Air Valve	438HC
The Air Transport Association	ATA

1 Introduction

The technological growth in the aerospace industry and the continued advancement in data analytics have made the generation and analysis of large quantities of aircraft data more affordable. Therefore, this has caused a transformation in maintenance strategies by shifting from preventive maintenance to predictive maintenance. Research into the development of data-driven prognostic models for condition-based maintenance is gaining more attention [1,2]. However, researchers' major problems are the low representation of faulty asset behaviour, which results in an imbalanced dataset. The imbalanced data problem arises when the distribution of classes present in the dataset is not uniform, such that the total number of instances in one class far outnumber that of the other class [3]. While training the traditional machine learning algorithms with the imbalance dataset, the resulting model will be biased, which degrades the performance of the data-driven model, causing imprecise prognostics. The rapid flow of data from the industrial process has increased research focus in big data analytics and its many applications in academics, industries, and government sectors [2,4,5]. Therefore, solving the imbalanced classification problem is necessary in order to build a high-performance predictive model. Research into this area is still an open issue [6–8], especially the data-driven approaches [9].

The imbalanced classification problem is prevalent in many application domains. For example, in building predictive maintenance for aircraft, the historical data is often imbalanced because the record about systems and processes is mostly healthy with fewer failure records [10]. Similarly, in financial fraud detection, in most cases, illegal transactions are often rare compared to the majority of legitimate ones. The fraudulent minority transactions are more critical to predict accurately to avoid the consequence of the successful occurrence of fraud [11]. The application of imbalanced learning is also seen in clinical science for rare disease detection. The majority of the population is healthy, and the minority is infected [12]. In this case, predicting the minority becomes critical. Likewise, imbalance learning can be seen in the oil spillage detecting problem. Large images of an ocean captured by satellite may show a few images representing the oil spillage portion, and most of the images represent the non-spillage areas [13]. In such cases, the target is to predict the minority spillage portion of the ocean. In a situation where the ratio between classes is not significantly large, and the existing machine learning methods can adequately handle such an imbalanced problem. However, in a situation where the ratio between classes in the dataset is extreme, say, 10000:1 [14], learning becomes more challenging because examples from the overwhelming class can be well-classified, whereas samples from the minority class can be misclassified. In the worst case, minority

examples are treated as outliers or noise of the majority class and ignored or dropped during learning. The learning algorithm ends up generating a trivial classifier that classifies every example as the majority class. Other factors that can impact the classification algorithm's performance apart from the imbalance ratio are; the class's small disjunct, the noise, and the class overlapping [15,16].

This study uses over eight years' worth of data recorded from 60 aircraft. The datasets are collected from two databases. The first database is the Aircraft Central Maintenance System log (ACMS) data, which comprises error messages from BIT (built-in test) equipment (that is, aircraft fault report records) and the flight deck effect (FDE). These messages are generated at different stages of flight phases (take off, cruise, and landing). The second database is the record of the aircraft maintenance activities (the full description of all aircraft maintenances recorded over time). These databases are associated with a fleet of civil aircraft. The data is grouped into two categories of aircraft families; the A330–long-range (LR) and the A320 -Single-aisle (SA) aircraft. So far, CMS data have only been used for troubleshooting, anomaly detection, Line Replacement Unit (LRU) removal assessment, and system failure analysis or test. However, no comprehensive study has shown the use of a hybrid ensemble approach using the aircraft CMS data to develop a predictive model for aircraft component replacement. Building a predictive model from CMS data (which is purely textual) in the total absence of digital sensor data measurements is quite a challenging task, and it becomes more challenging when the data distribution is extremely imbalanced

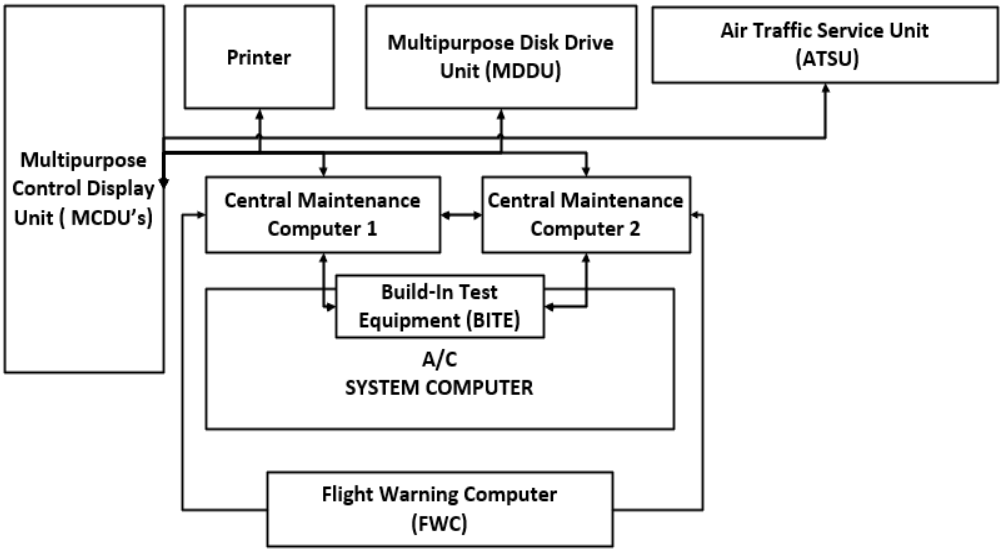


Figure 1 The Aircraft CMS architecture

As seen in Figure 1, CMS data is generated from an aircraft central maintenance computer (CMCs). The system is composed of LRU's (computers, sensors, actuators, and probes). The function of the CMC is to detect and memorize any failure occurring within the system, which is called the Built-in Test Equipment (BITE). During regular aircraft operation, the system is permanently monitoring and reporting the state of each failure. CMCs are responsible for centralizing and memorizing warnings generated by the flight warning computers (FWCs) and failure messages produced by the BITE function integrated into the aircraft computers. The CMC enables maintenance engineers to perform system operational tests, functional checks, and read-out BITE memory through the Multipurpose Control and Display Unit (MCDU). Reports can be printed on-board, saved on an external disk, or transmitted to the ground through the Air Traffic Service Unit (ATSU) as it is possible to perform an operational test from the cockpit. Most airlines hardly analyze this data further, especially for predictive maintenance modeling. The ACMS data is analysed in this work in order to construct a prediction model for aircraft component replacement. The problem of an imbalanced dataset in the context of aircraft predictive maintenance is examined in this article, which uses the ACMS dataset in particular.

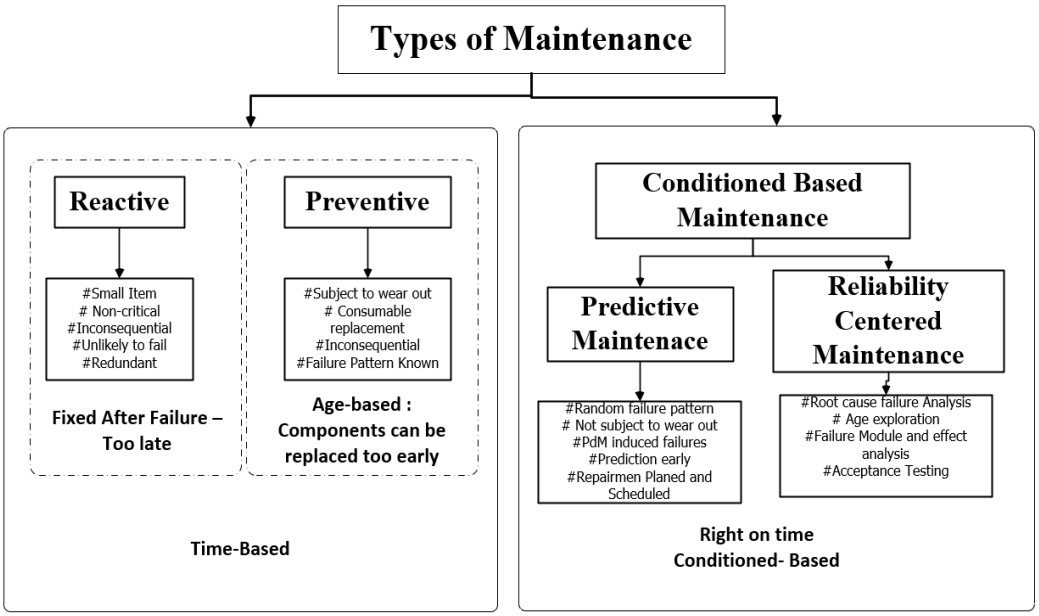


Figure 2 Types of maintenance strategies

Maintenance strategies have progressively developed over time, and the goal remains the same: to preserve equipment. Recently, industries are becoming more aware of the advantages of applying advanced machine learning methods to enhance quality, process performance, and system uptime by maintaining overall equipment effectiveness (OEE). This development has brought more research attention to predictive maintenance modelling for heavy equipment monitoring [5,9,17]. Maintenance can be categorized into a time-based and conditioned base, as seen in Figure 2. In time-based, we have reactive maintenance, which involves fixing after things have been broken down, and preventive maintenance involves keeping things from breaking. Reactive maintenance is quite expensive and time-consuming because no prior knowledge is available to plan effective maintenance.

On the other hand, preventive maintenance allowed the pre-emptive measure to be taken before equipment failure, for example, by conducting repairs at fixed intervals regardless of the equipment condition. Advancement in technology has allowed the second category of maintenance, known as conditioned-based maintenance (CBM). CBM optimizes preventive activities based on the actual conditions of the asset. Predictive maintenance is a form of CBM where a predictive model is developed to forecast future failure using past failure records.

The imbalanced dataset's problem in developing a data-driven prognostic model for predicting unplanned aircraft component failures is considered in this study. The study proposes a novel method that involves a unique fusion of two machine learning techniques (ensemble learning and cost-sensitive learning) to form a hybrid approach. In the proposed hybrid algorithm, we use a balance-cascading algorithm to cascade the majority class. Then the minority class is synthesized and boosted using boosting data expansion policy, which overcomes the extreme imbalance classification problems and reduces the computational cost-efficient for larger datasets compared to deep learning methods [18]. The ensemble process provides a unique classifier arrangement and cost sensitivity to each weak learner, which produces state-of-the-art performance.

The contribution of this paper is as follows:

One of the fundamental research questions that this study seeks to answer is can a class overlapping and small disjunct problem inherent in the extremely imbalanced ACMS dataset be overcome using a hybrid ensemble learning? A new algorithm known as Balanced Calibrated Hybrid Ensemble Technique (BACHE) is designed and implemented to answer the above question. The approach's novelty is found in the uniqueness of ensemble architecture that combines cost-sensitive weak classifiers to improve minority class sample prediction. The inclusion of cost-sensitive in the weaker learners, which is applied to all subsets, lowered the imbalance ratio, assisting in overcoming the challenge of class overlaps, reducing bias, and improving the prediction rate for both minority and majority classes. Another contribution is that the effectiveness of the proposed approach is validated using a real-world dataset (the aircraft central maintenance system-CMS dataset); this is a distinctive contribution because of the dataset's heterogeneous nature, which is challenging to mine for predictive modelling.

The remainder of this paper is organized as follows: Section 2 provides related work. Section 3 presents the methodology. Section 4 presents the experimentation. Section 5 discusses the results and model validation, and finally, section 6 presents the conclusion and future work.

2 Related Work

This section gives an overview of imbalanced classification problems. Several research approaches have been conducted to solve the imbalanced classification problem, and some comprehensive reviews can be found in [19–22]. The solution to the imbalanced classification problem can be categorized into three main groups [20]. The data level, the algorithm level, and the hybrid approach,

as seen in Figure 3. The data level approach involves resampling the dataset before presenting it as an input to the learning algorithm. The data level approach has gained a lot of research attention, especially the over-sampling techniques, which involve increasing the minority class samples to have a balanced class. Some of the methods are based on oversampling are the Synthetic Over-sampling Techniques (SMOTE) developed by Nitesh et al. [23]. Their technique creates new synthetic samples into the minority class to balance with the majority. Although the SMOTE approach has widely been used to address the imbalanced classification problem. However, SMOTE contains some drawbacks, such as class overlapping because it ignores adjacent samples when creating new synthetic points [24] and overgeneralization problem. Hence, many advanced versions of SMOTE have been developed, such as SMOTE-Boost [25], which introduces new dynamic weighted synthetic data points in the minority class at each round of boosting steps to eliminate the overgeneralization problem. SMOTE-boost tries to solve these drawbacks of the main SMOTE by adding synthetic data points in each weak classifier of the easy-ensemble method [26]. Other versions are the Easy-SMOTE Algorithm [27], Borderline-SMOTE [28], and many more. Also, Wing et al. [29] propose a neural-network training method to balanced an error yield by minority class via minimization of the cost-sensitive localized generalization error-based objective function. The approach shows better performance in terms of G-mean as compared to other similar algorithms.

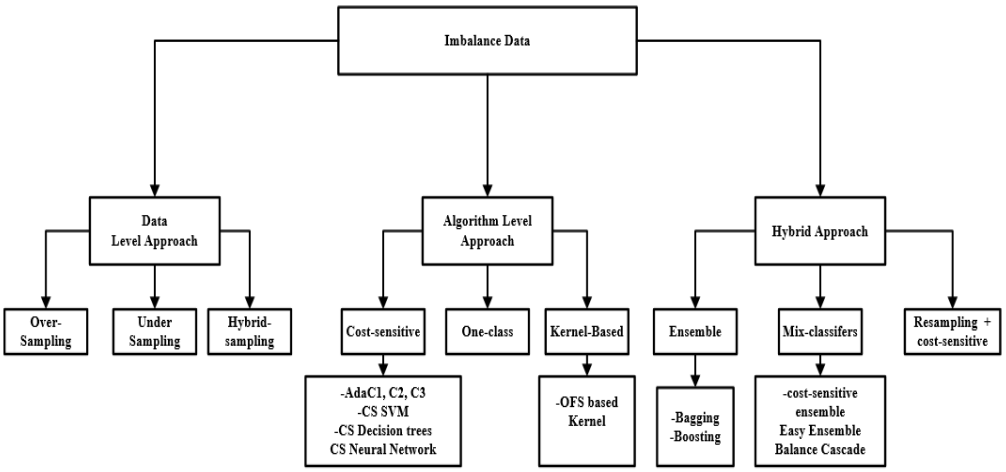


Figure 3 The three existing categories of the State-of-the-art approach of the handling imbalance problem

The algorithm level approach tackles the imbalanced learning problem by altering the learning algorithm to respond favourably to both classes during learning [20]. Cost-sensitive learning is an algorithm-level approach. The cost-sensitive method is explored by defining the cost of misclassification for each class. Determining the cost of misclassification is challenging in the traditional classification algorithms (such as support vector machines, decision trees, and more) because the algorithms presume that all classification errors carry the same cost. Hence, they focus on minimizing the error rate and the percentage of a class's incorrect prediction, ignoring the difference between the misclassification errors. Therefore, cost-sensitive learning takes into consideration the different costs that vary by type of classification (true-positive, true-negative, false-positive, false-negative) across all samples. The goal is to minimize the total cost, such as the G-mean score. From a business point of view, it is vital to determine the misclassification cost for each class. For instance, in aircraft maintenance, the cost of misclassifying the minority class (failure) has a higher impact on the business than the misclassifying majority class (healthy state). Hence, cost-sensitive learning is used to mitigate such problems [30]. Cost-Sensitive in Decision Tree algorithm (CS-DT) involves introducing cost into the decision tree algorithm for the algorithm to respond favourably to all classes during training [31]. Cost-sensitive learning is effective in classifying datasets with different imbalance distributions [30]. The changing misclassification costs are best understood using the idea of a cost matrix. As seen in Table 1, a Cost sensitives learning can be binary or multi-class; in either case, it associates different misclassification costs to every prediction.

Table 1 Cost or Confusion Matrix

	Actual Positives	Actual Negatives	TP: True Positive TN: True Negative FP: False Positives FN: False Negative
Predicted Positives	TP ($C_{1,1}$)	FP ($C_{1,-1}$)	
Predicted Negatives	FN ($C_{-1,1}$)	TN ($C_{-1,-1}$)	

Using the confusion matrix as shown in Table 1, the value ($C_{i,j}$) represents the cost of misclassifying a data point from its actual class (j) to a predicted class (i), 1 represents positive class, while -1 represents the negative class. Usually, the cost of correct prediction that is TP and TN should always be lower than the cost of misclassification error that is FN and FP, usually is set to zero. ($C_{i,i}$) is

regarded as a negated error since the data point is predicted correctly. Cost-sensitive learning has widely been applied in imbalanced learning [28]; the challenge is learning the cost matrix. In some domains, it might be obvious because the consequence of misclassification can just be based on monetary value. However, in areas such as predictive maintenance for aircraft, the consequence of the misclassification of faults can be grave.

The easiest way of defining the misclassification cost is to input it manually according to the domain expert advice or inversely calculate it based on class distribution [32–34]. The challenge of using a manual approach for calculating the cost of misclassification is that it is time-consuming and sometimes impractical. Another approach can be to fit the importance of features to adaptive equations [35], which involves incorporating second-order information to enhance the prediction of the minority class. However, because of the peculiarity of the dataset used in this study, neither method is suitable. Hence, we define the misclassification cost from cost-sensitive algorithms' evaluation functions, using weighted Platt calibration to measure the cost sensitivity of the classification algorithm.

The imbalanced learning hybrid approach involves combining more than one method, either from data levels or algorithm-level techniques, to enhance prediction [36]. An example of the hybrid approach is ensemble learning. Ensemble learning involves enhancing prediction by using a combination of weak learners to form a strong learner. The major course of error in machine learning is the presence of noise, variance, and bias in the dataset. Ensemble classifiers are built to minimize these factors, which improves the stability and learning performance of machine learning algorithms. A study by Zhou et al. [37] shows a broad overview of why and how ensemble learning improves prediction performance. Diverse ensemble learning strategies that focus on imbalanced learning have been proposed in the literature. For instance, Galar et al. [38] provide a broad overview of different combinations of multiple classifiers to improve predictive accuracy. The ensemble approach can either be constructed using boosting or bagging learning structures to optimize accuracy. The implementation of boosting learning can be found in AdaBoost [39], SMOTEBoost [40]. The bagging implementation that is bootstrap aggregating [41] can be seen in SMOTEBagging [42].

Combining ensemble learning with data level approach (under-sampling or over-sampling) to solve the imbalanced classification problem has led to several proposals in the literature, with positive results [43]. Although ensemble learning is known to enhance machine learning model performance [44], the arrangement of classifiers alone cannot solve the class imbalance problem. Hence, the

ensemble approach needs to be explicitly designed for imbalanced learning to deal with imbalanced classification challenges. For example, The balance-cascading and easy-ensemble algorithms presented in the study by Liu et al. [14,45] uses the under-sampling technique with an ensemble approach to train the weak learners and then combine the result to form a robust classifier. These algorithms use the under-sampling method because of its advantage of less training time. They then focus on tackling its disadvantage, which is a reduction of informative samples. Easy-ensemble involves resampling the majority class into several subsets, then training each subset using weak learners (such as AdaBoost [46]) while keeping the minority class constant.

The result of each data subset will then be combined using majority voting. This approach has recoded positive results, which has led to more advances in this direction. An easy-synthetic minority over-sampling technique (easy-SMT) developed by Wu et al. [4]. Easy-SMT is an integrated ensemble-based method that uses a SMOTE-based over-sampling and under-sampling strategy to transfer imbalanced problems into an ensemble-based balance sub-problem. Using an easy-ensemble or balance-cascade algorithm to resampled the dataset involves exploring the data samples ignored by the random under-sampling technique. However, both methods keep the minority class constant while training the subsets, which creates computational cost if the data is large. Wankhade et al. [45] proposed a hybrid method to deal with an imbalance classification problem that addresses the above challenge. Their technique uses a combination of classification and clustering to enhance recognition of the rare class during learning. Likewise, Vluymnas et al. [47] proposed a hybrid method for solving the imbalanced problem, which combines a preprocessing and classification model. The results of both approaches show an improvement in predicting minority class. Another hybrid approach was developed by Le et al. [48] to predict bankruptcy. Their algorithm uses an over-sampling technique and cost-sensitive learning to handle imbalanced classification problems. The results show that the approach outperforms other existing methods in predicting bankruptcy, which is rare in the dataset used. Application of Imbalance learning has also been seen in rotating machinery; Yuyah et al. [7] show oversampling and future-leaning to handle imbalanced data in fault diagnosis. Different studies have also demonstrated how imbalanced data problems can be handled using deep learning [49,50].

As highlighted above, most of the methods are validated on diverse individual datasets, making them domain-specific. Thus, the peculiarities of our dataset make it challenging to apply off-shelf techniques. Among the different approaches, the hybrid methods show effectiveness and robustness in handling the imbalance problem compared to other single methods. Also, the open literature lacks an extensive study that uses ensemble learning to address extreme rarity, class overlapping, and class

disjunct especially using a system log dataset. Therefore, the arising research question is, how can an architecture of ensemble classifiers be constructed for tackling extremely imbalanced datasets taking into account class overlapping and small disjunct problems. Usually, the number of weak learners is selected arbitrarily, which can result in redundancy for similar classifies [51]. For example, relating the size of weak learners to the data complexities such as reducing bias and variance in the extremely imbalanced dataset.

Therefore, this study aims to advance the ensemble and hybrid approach by considering the challenge of extremely imbalanced classification problems combined with class overlapping. Also, our proposed method is inspired by two observations: first, the possibility of convergence of different boosting algorithms for an optimal solution heading to the direction of the gradient of the objective function, and the cost-insensitive predictor can then asymptotically minimize. Second, ensemble algorithms can perform shift decision threshold and calibration of probability estimation, which accounts for class imbalance [52].

3 Methodology

This section describes the methodology for this study which covers the proposed approach.

This study aims to enhance the learning algorithm's performance to reduce False Positive Rate (TPR) and False Negative Rate (FNR) while learning from the extremely imbalanced system log dataset. The reduction in false-negative and false-positive can translate to a reduction in the unplanned maintenance check, which can reduce the overall cost of maintenance.

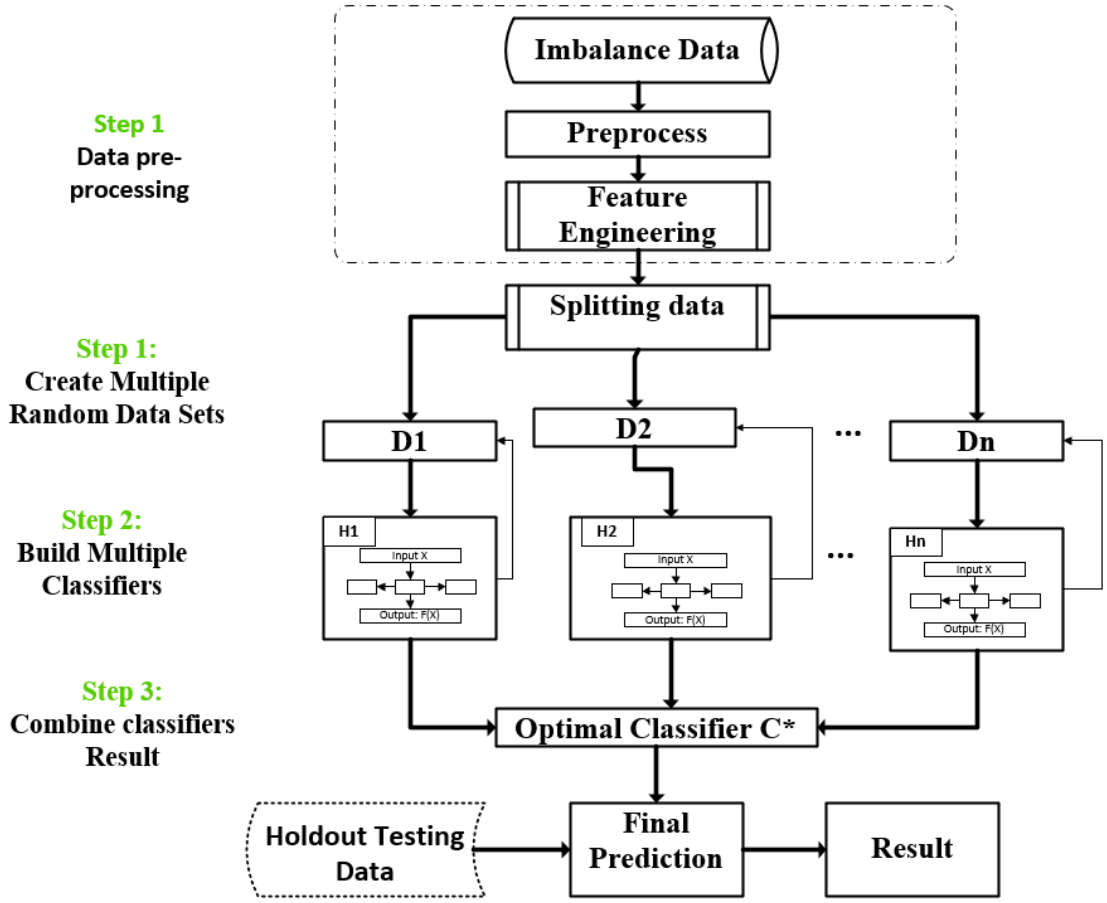


Figure 4 The Methodology of Imbalance learning using BACHE Algorithm

The three approaches that form the BACHE algorithm's key building blocks are the data level (under-sampling), ensemble learning, and cost-sensitive learning. In the under-sampling phase, a balance-cascading algorithm is used to reduce data from the majority class. The major course of error in machine learning algorithms is the presence of noise, variance, and bias in the dataset [37][53]. Ensemble classifiers are built to minimise these factors, which improves machine learning algorithms' stability and learning performance. A study by Zhou et al. [37] shows a broad overview of why and how ensemble learning improves prediction performance. The two most basic qualities expected of a model are a low bias and a low variance, which frequently fluctuate in opposite ways. Indeed, the model is required to have enough degrees of freedom to resolve the underlying Complexity of the data it is working with, but not too many degrees of freedom to avoid high variance and be more robust. This is the well-known tradeoff between bias and variance. Most of the time, in ensemble

learning, the weak learners do not perform well by themselves either because they have a high bias or high variance. Therefore, the idea of the ensemble is to try reducing the variance and bias at the same time by combining multiple weak classifies to create a strong one for enhancing performance. The ensemble learner approach is chosen because it combines multiple weak learners to produce a robust classifier, as seen in Figure 4. The choice of the balance-cascading approach for the BACHE is because of its low computation cost and its effectiveness in utilizing the majority class samples ignored by random under-sampling techniques. The calibrated cost-sensitive is used to define the misclassification cost in each weak learner's prediction, which tackles problems where the costs of different types of erroneous predictions are not equal. In the ensemble boosting phase, instead of using a standard decision tree, a cost of classification using a calibrated probability estimate is considered at each iteration by modifying the updating rule with regards to the modified loss function. Likewise, instead of finding the best classifier, the problem is directed to focus on finding the best learning rate γ [54–56].

Therefore, what makes a difference here is the tree structure and the model weight updating rule. The BACHE algorithm works as follows; first, data preprocessing and feature engineering is conducted. To improve the quality of the predictive model, new features from the existing variables are created using integer encoding and one-hot encoding methods. The choice of the method is based on the nature of the dataset because the data is heterogeneous with categorical features. Developing a machine learning model directly can not produce an optimal result for most of the traditional machine learning algorithms. In variables where ordinal relationships exist, an integer encoding was used, and where such a relationship does not exist, one-hot encoding was used; Therefore, creating new features was necessary for this project. After preprocessing the dataset and selecting the right features, the data is divided into two. 80% of the data is kept for model training and 20% for model testing. Then the dataset is divided into several subsets using a cascading balanced approach [57]. At every boosting integration step (selection with replacement), the samples of each subset are balanced to form Balanced Data ($D_{i's}$). After the dividing and balancing process, each subset is trained using weak learners. The process continues for the number of defined iterations. At each iteration, the subset learns using a weak learner (H_{i^s}) at the end of the ensemble process, the result of all the weak learners, is combined to get a hybrid ensemble classifier. The final model is then evaluated using new hold-out datasets.

As seen in Figure 4, the proposed BACHE methodology explores both the majority class ($\boldsymbol{N}$) and Minority class ($\boldsymbol{P}$) in a supervised learning manner. The weak learners H_{i^s} are trained in sequence

on a weighted version of the dataset using a cost-sensitive boosting algorithm. Considering N in the under-sampling process, if data point example say $\boldsymbol{x_i} \in \boldsymbol{N}$ is correctly classified to be in $\boldsymbol{N}$ it easier to infer that $\boldsymbol{x_i}$ is reasonably redundant in N, given that we already have the outcome as H_1 [58]. Therefore, x_i will be removed from N. (That shows N will be reduced after training each H_i). Every H_i deals with balanced sub-set $|N_i| = |P_i|$, after processing all the subsets of the cascaded dataset, the outcome of $H_i{}^s$ is combined using a weighted majority vote.

Elaborately, considering the majority class N and minority class P, the length of iteration is S_i and the length of each $n \in N$ subset is defined as $\boldsymbol{M}$ (we use an under-sampling technique to split N into random subsets $n_1, n_2, n_3 \ldots n_T \in N$). Then a subset of $p \in P$ is combined with each $n \in N$ to form a balanced sub-dataset (D_i). These $D_i{}^s$ are trained using weak classifiers, which are later combined using boosting approach to form an optimized classifier. In each weak classifier, a cost-sensitive calibrating boosting algorithm is used. Such as adaMEC [52], a score of the form $(x) \in [P, N]$ is generated. A cost marix for false negatives, false postive, true postives and true negatives is contructed as follows.

A probability of x belonging to a positive class P is given as $prb(y = (1|x))$, x will be assigned to a class with a minimized expected cost. In other words, a data point x_i will be assign to positive class P if and only if $prb\big(y = (1|x)\big)cv > prb\big(y = (-1|x)\big) \leftrightarrow prb\big(y = (1|x)\big) > \frac{1}{1+c}$. For example, using the imbalance learning cost matrix (see Table 1)

$$c = \begin{bmatrix} 0 & 1 \\ c & 0 \end{bmatrix}, \text{c } (y_i) = \begin{cases} c & if\ y_i = 1 \\ 0 & if\ y_i = -1 \end{cases} \quad (1)$$

Where $prb\big(y = (-1|x)\big) = 1 - prb\big(y = (1|x)\big)$.

Otherwise data point x_i is assigned to the negative class N. It is important to note that probability estimates are not always straightforward to obtain from a classifier's outputs [59]. Therefore, a generated score of the form (x) is calibrated using platt scaling. The classification of the extreme minority is accounted for in the calibration step as detailed in [59]. In the Platt calibration, it uses $\frac{P+1}{P+2}$ for positive class and $\frac{1}{N+2}$ for negative class, rather than 1 and 0 as the target probability estimation of the P and N. Therefore, in BACHE we aim to reduce the ensemble error rate by focusing on different positive class P, as we want to model P better to enhance detection of the

extreme minority and also avoiding accuracy degradation for the negative class N. The BACHE algorithm pseudocode is presented in algorithm 2.

In the proposed approach BACHE algorithm, the cost-sensitive decision tree algorithm is used as a weak classifier expressed as follows. In machine learning, classification involves predicting the class of a given data point, say y_i of a dataset Ds, given their k features $x_i \in R^k$. Classification in predictive modelling is about approximating a mapping function f (·) that minimizes the expected value of some specified loss function $L\left(y_{i,} F(x)\right)$, to makes a prediction c_i of the class of each example using its input variables x_i .

$$\hat{F} = \underset{\gamma}{argmax}\, E_{x,y}[L(y, f(x))] , \tag{2}$$

where γ is the learning rate

Similarly, as described by Hastie et al. [60], the gradient boosting methods uses a real value of $y_i \in R^y$ and then seek an approximation of $\hat{F}(x)$ that minimize the average value of loss function on the training dataset, this is achieved by starting with a constant function $F_0\,(x)$ and increment it greedily.

$$F_0\,(x) = \underset{F}{argmax} \sum_{i=1}^{n} (L(y_i, \gamma)) \tag{3}$$

$$F_m\,(x) = F(x)_{m-1}\,(x) + \underset{h_m \in H}{argmax} \left[\sum_{i=1}^{n} (L(y_i, F(x)_{m-1} + h_m\,(x_i)\,)\right] \tag{4}$$

$h_m \in H$ is the base learner function.

To further minimize the problem, the steepest descent approach is used to transform (eq. 2) as the gradient descent and taking the derivatives with respect to F_i for $i \in \{1, \ldots m\}$

$$\hat{F}_m\,(x) = \hat{F}(x)_{m-1}\,(x) + \gamma_m \left[\sum_{i=1}^{n} \nabla F(x)_{m-1}\,(L(y_i, F(x)_{m-1} + F(x)_{m-1}\,(x_i)\,)\right]$$

$$\gamma_m = \underset{\gamma}{argmax} \left[\sum_{i=1}^{n} (L(y_i, F(x)_{m-1} - \nabla F(x)_{m-1}\, L(y_i, F(x)_{m-1}\,)\right] \tag{5}$$

To improve the quality of fit of each base learner function, we use the Friedman approach [61],

considering m^{th} steps to fit a decision tree $h_m\,(x_i)$, and j_m are the leaves nodes, we get

$$F_m(x) = F(x)_{m-1} + \sum_{j=1}^{j_m} \gamma_{j_m} 1R_{j_m}(x), \ \gamma_{j_m} = \underset{\gamma}{argmax} \sum_{x_{i \in R_{j_m}}}^{n} L(y_i, F(x_i) + \gamma) \quad (6)$$

j , denotes the number of terminal leave notes in the tree.

Hence, the gradient boosting algorithm is expressed as

Input: the training set $\{(\boldsymbol{x_i}, \boldsymbol{y_i})\}_{\boldsymbol{i=1}}^{\boldsymbol{n}}$, a differentiable loss function $L(y_i, F(x_i))$ and number of iterations M.

1. Initialize the model with a constant value

$$F_0(x) = \underset{\gamma}{argmax} \sum_{i=1}^{n} (L(y_i, \gamma))$$

2. For $m \in \{1, \ldots M\}$:

a. compute $\gamma_m = -[\frac{\delta L(y_i, F(x_i))}{\partial F(x_i)}]$; for i= 1,…M

b. Fit a base learner h_m (using CS-DT) $\partial_m to (x_{i,\ \gamma_{mi}})$ for i =1,…n

c. compute multiplier γ_m using the following optimization function.

$$F_m(x) = F(x)_{m-1} + \sum_{j=1}^{j_m} \gamma_{j_m} 1R_{j_m}(x), \ \gamma_{j_m} = \underset{\gamma}{argmax} \sum_{x_{i \in R_{j_m}}}^{n} L(y_i, F(x_i) + \gamma)$$

d. update the model $F_m(x) = F(x)_{m-1} + \gamma_m \nabla_m(x)$

3. Output $F_M(x) = 0$

Gradient Boosting Tree Algorithm forms the core component of the BACHE algorithm. It is used as a weak classifier.

Algorithm 1: The Balanced Calibrated Hybrid Ensemble Technique (BACHE) Algorithm

INPUT:

Dataset: $\mathbf{D} = \{(\boldsymbol{x_i}, \boldsymbol{y_i})\}_{i=1}^{n}$ with minority class $\boldsymbol{P}$, majority class $\boldsymbol{N}$ and $\boldsymbol{P} < \boldsymbol{N}$.

The number iteration or the number of subsets to be sampled from N: $\boldsymbol{M}$, Whwre the length of each n ∈N subset is defined as $\mathbf{M}$

Imbalance Ratio $\boldsymbol{IR} = \frac{\boldsymbol{P}}{\boldsymbol{N}} * 100$

The number of iterations to train the calibrating ensemble $\boldsymbol{H_i}$: $\boldsymbol{s_i}$

K= is constant, it define the number of subset

$\boldsymbol{for\ i = 1\ to\ K\ Do}$:

$\boldsymbol{f} \leftarrow \sqrt[M-1]{\frac{n+}{n-}}$, $\boldsymbol{f}$ is the FP-rate that $\boldsymbol{H_i}$ should achieve.

Randomly sample a subset $\boldsymbol{N_i}$ of $\boldsymbol{n_+}$ with replacement.

$\boldsymbol{N' = (IR > 1.25\ (M_2 - 1),\ K);\ \ P = P + P'}$

$\boldsymbol{for\ i = 1\ to\ j\ Do}$:

Training Phase:

Split the data in training set D_t and calibration set D_c (for correcting distortion)

On $\boldsymbol{D_t}$:

Train $\boldsymbol{H_i}$ using $\boldsymbol{P'}$ U $\boldsymbol{N_i}$.

$\boldsymbol{H_i}$ is obtained using Algorithm 1 with $\boldsymbol{S_i}$ as weak classifier

$\boldsymbol{h_{i,j}}$ and corresponding weight $\boldsymbol{\alpha_{i,j}}$.

The ensemble shifted decision threshold is $\boldsymbol{\theta_i}$

On $\boldsymbol{D_c}$- calibrated boosting:

a. calculate score $s(x_i) = \frac{\sum_{\tau:} h_{i,j}(x_i)=1}{\sum_{\tau=1}^{S_i} a_{i,j}} \in [1,0] \forall x_i \in D_c$

b. calculate the number of P and N in D_c;

find A,B s.t $\sum_{i \in D_c} prb(y = (1|x_i) - y_i)^2$ is minimized.

Where $prb(y = (1|x) = \frac{1}{1+e^{As(x)+B}}$ and $y_i = \begin{cases} \frac{P+1}{P+2}\ if\ y_{i=positive} \\ \frac{1}{N+1}\ if\ y_{i=negative} \end{cases}$

Prediction Phase:

On new data-point (x):

Calculate prior weight score $s(x)$

Obtain prior weight probability estimate $prb_w\ (y = (1|x) = \frac{1}{1+e^{As(x)+B}})$

Predict class $H(x_j) = \underset{\gamma}{sign} \left[prb_w\ (y = (1|x) > \boldsymbol{\theta_j}\right]$

Adjust $\boldsymbol{\theta_j}$ such that $\boldsymbol{H_{ij}}^{\boldsymbol{s}}$ the false positive rate is $\boldsymbol{f}$

Remove from $\boldsymbol{N}$ all samples that are correctly classified by $\boldsymbol{H_j}$.

End

End

OUTPUT ENSEMBLE:

Return $\boldsymbol{H_i(x) = \underset{\gamma}{sign}\left(\sum_{i=1}^{M}\sum_{j=1}^{S_i}\alpha_{i,j}\, h_{i,j}(x) - \sum_{i=1}^{M}\theta_i\right)}$

4 Experiment

To validate the effectiveness of the proposed approach, we use the following datasets as input. The first data is the data generated from the central maintenance system (log-based CMS data), and the second data is the record of maintenance activities. The datasets are obtained from a fleet of long-range (A330) aircraft and A320 families. According to families, aircraft grouping is necessary because the data generated differ in properties and structure. The designation routes were different for each family; some were mainly used for long-distance routes, while some were primarily used for short distances. From the A330 aircraft family, the total number of failure/warning messages after preprocessing is 389902, and the A320 family has a total of 890120.

The main objective is to develop a predictive model to predict failure resulting in aircraft's unplanned repairs or components' replacement. Therefore, we choose target components identified by Functional Item Number (FIN). The representation of these components is extremely rare. The basic idea is to tackle class overlapping, which will enable us to correctly detect the extreme minority class samples and the majority class samples during model classification.

Apart from the high skewness and class overlapping problem in the dataset, the raw data has many challenges that require preprocessing, such as data incompleteness, lack of behaviours and trends, containing null values, lacking the features of interest, and containing noise. Therefore, the data knowledge discovery approach [62] is followed. The data is preprocessed and transformed into a suitable format for machine learning. After that, a Feature Engineering (FE) process is carried out. FE is the integral and critical step of the machine learning process because the model's performance output depends on the quality of data and the right features selected. After the preprocessing and feature engineering phase. The data is divided into two: For training and for testing the model. The

data was split into training and testing divided into 70/30 (from January 2011 to September 2016) and validation data from October 2016 to April 2018 (without known label).

The following requirements are considered in the design and develop the imbalance-learning framework.

1. Features obtained from the raw CMS dataset should adequately represent the component replaced.

2. The baseline learning algorithm and classifier should be suitable for large imbalanced datasets.

3. The model performance evaluation metrics should be suitable for an imbalance scenario.

4. Prognostic alert requirements: - Predictive model should flag up alerts for maintenance needs (component replacement), not more than ten and not less than two flight cycles before failure point. The window period is to avoid early replacement of a component, which will mean underutilizing resources and not too close to failure to give adequate room to prepare for maintenance.

5. Model should achieve more than 60% precision, recall more than 50%, or G-mean of greater than 50%.

From the dataset, a few aircraft components were selected for validation. The components are the Electronic control unit/ Electronic engine unit (4000KS), High-pressure bleed valve (4000HA), pressure regulating valve (4001HA), Satellite data unit (5RV1), Flow control valve (11HB), Avionics equipment ventilation computer(10HQ), Air traffic service unit (1TX1) and Flow control valve (8HB). The selection is based on descriptive analysis, which shows the percentage of each component replaced over the period under consideration. Components with the highest number of replacements are selected, containing enough patterns to train the machine learning model. The dataset (failure/ warning message) is clustered according to every specific component. Then, in each cluster, patterns that lead to component replacement are labelled as a positive class (representing the minority class-P), while patterns that did not lead to component replacement are are labelled as the negative class (representing the majority class-N). In each cluster, since the data is sequential in terms of date-time and flight circles, we group the data into windows using date-time and flight cycles; a window size of 30 aircraft flight cycles was used. The choice of window size is based on the domain of expert advice.

Our experiment compares the performance of existing ensemble boosting methods for imbalanced learning with the proposed approach. Therefore, the following experiment was set up to evaluate the proposed BACHE algorithm's performance on aircraft rare unplanned failure prediction problems.

Balance Bagging (BB): This is an ensemble learning method. It uses a bagging approach with an additional capability to balance the training dataset at the fitting time. During training, the parameter can be turned for the best results. Therefore, BB is considered our baseline method since our algorithm is based on the ensemble learning approach and focuses on tackling extremely rare failure problems in aircraft systems. The hyper-parameters are Base_estimator=None,n_estimators=10,max_samples=1.0,max_features=1.0,bootstrap=True,bootstrap_features=False,oob_score=False,warm_start=False,n_jobs=None,random_state=None,verbose=0.

SMOTE-Random Forest (SMT-RF): This method involves combining an imbalance learning with an ensemble algorithm. We first use the SMOTE algorithm to resample the minority class and then apply the ensemble-RF algorithm as the classifier. The Random Forest algorithm is implemented using the following hyperparameters: learning_rate = 0.1, Max_depth =10, Subsample = 50, Colsample = 0.3, n_estimator= 10

XGBoost (eXtreme Gradient Boosting): XGBoost is an ensembled learning based algorithm, ensembled are contructed from decision trees, trees are added using boosting approach (one at a time to the enseble as fit for classification) [63]. XGBoost Scikit_Learn API was used with the following hyperparameters**:** learning_rate = 0.1, Max_depth =10, Subsample = 50, Colsample = 0.3, n_estimator= 10.

Cost-Sensitive C4.5 Algorithm: This ensemble-based algorithm builds decision trees from a set of training data [64]; the trees are used for classification. C4.5 algorithm is implemented using the following hyperparameters: learning_rate = 0.1, Max_depth =10, Subsample = 50, Colsample = 0.3, n_estimator= 10.

Balance calibrated Hybrid Ensemble Technique (BACHE): The proposed approach.

Experiment Running Environments:
Operating system: The experiment was performed on MacBook pro (ios 14) with GPU.
Programing langege : Python
Machine learning Editor: Sublime and Jupyter notebook

Major Packages: Pandas, Scikit-learn, Keras, TensorFlow, sciPy and more.

Running the experiments with multiple seeds will ensure the approach is not sensitive to different start conditions. Some of the sensitivity to initial conditions could be that the failure distribution can substantially differ between the training and validation datasets, which will likely negatively affect model training. To mitigate that, stratified samples and random seed can be used so that the proportions of the dependent variable are similar in training, testing and validation dataset. In the implementation, each algorithm was run five times with the same hyperparameter for each target event using five random seeds then the average is obtained.

4.1 Evaluation Criteria

In machine classification, accuracy is the most significant performance metric usually used. However, the use of accuracy to evaluate performance under extremely imbalanced classification problems can be misleading because classifies will be biased towards the majority class to achieve high overall accuracy. Therefore, to evaluate the performance of the classifiers better, some alternative metrics are adapted: Precision, Recall, G-mean, Area Under the Curve (AUC) are used as evaluation metrics, defined as follow:

Precision: Measures how exact the model is predicting, i.e., percentage of predicted fault events that are correctly labelled or measure of classifier exactness.

$$\text{Precision} = \frac{TP}{TP+FP}, \tag{1}$$

where TP – is true positive, FP -is false positive

Therefore, low precision indicates a large number of False Positives.

Recall: Measure of how complete the model is predicting, i.e., the percentage of true fault events which are labelled or measure of classifier completeness

$$\text{Recall} = \frac{TP}{TP+FN}, \tag{2}$$

where TP – is true positive, FP -is false positive

Therefore, low recall indicates many False Negatives.

G-mean: is the mean average between the precision and the recall

$$G - mean = \sqrt{(pricision * recall)} \quad (3)$$

Receiver Operating Characteristic Curve (ROC) is a comprehensive index reflecting the continuous variable of sensitivity and specificity. It shows the ability of the classifier as the discriminant threshold is varied.

True Positive Rate - TPR: Also known as sensitivity.

$$TPR = \frac{TP}{TP+FN} = Recall, \quad (4)$$

where TP is true positive and FP is false positive

Specificity is defined as False Negative Rate (FNR)

$$FNR = \frac{(TP+TN)}{TP+FN+TN} = 1 - FPR \quad (5)$$

4.2 The time complexity of the BACHE algorithm

The algorithm use of computational resources is determined by time complexity computational analysis. In the worse cases, running time is expressed as a function of its input using a big Omicron (big-O) notation[65][66]. Big-O notation gives an upper bound on Complexity or the growth rate of a function, and hence it signifies the worst-case performance of the algorithm. The big-O notation is express in the order of growth from best to worst. O(1) constant runtime < O(ogn) logarithmic < O(n) Liner growth < O(nlogn) log-linear growth < O(n^2) quadratic growth < O(2^n) exponential growth < O(n!) factorial growth. The complexity analysis retains the dominant term while the scaling factors and constants are ignored since the concern is only about asymptotic. For instance, if an algorithm needs O (3n^3 + 10n +10) operations, its order is said to be O(n^3).

The Complexity of the BACHE algorithm is computed with respect to the data input size. The statements have an order of O(1); because of the nested loop, we have a runtime of O(k *j) because instead of j, we have to iterate on k, the Complexity becomes O(n^2). Since Algorithm 1 is invoked in the second loop, the Complexity of algorithm one is considered, which has a constant time loop of order O(1). putting it together

0(1) + (n^2)+O(1) = O(n^2 +2) = O(n^2)

Therefore BACHE has a time complexity of O(n^2) quadratic growth

5 Results and Discussion

This experiment investigates the proposed approach's performance against the existing ensemble learning algorithms (Balance Bagging as baseline) and hybrid imbalance learning algorithms (SMOTE + Random Forest). The choice of the baseline algorithms is to enable us to assess the proposed method's performance, which uses a cost-sensitive decision tree as a weak classifier and then employs an ensemble approach to get a hybrid algorithm (BACHE) as a solution to the extremely imbalanced classification problem.

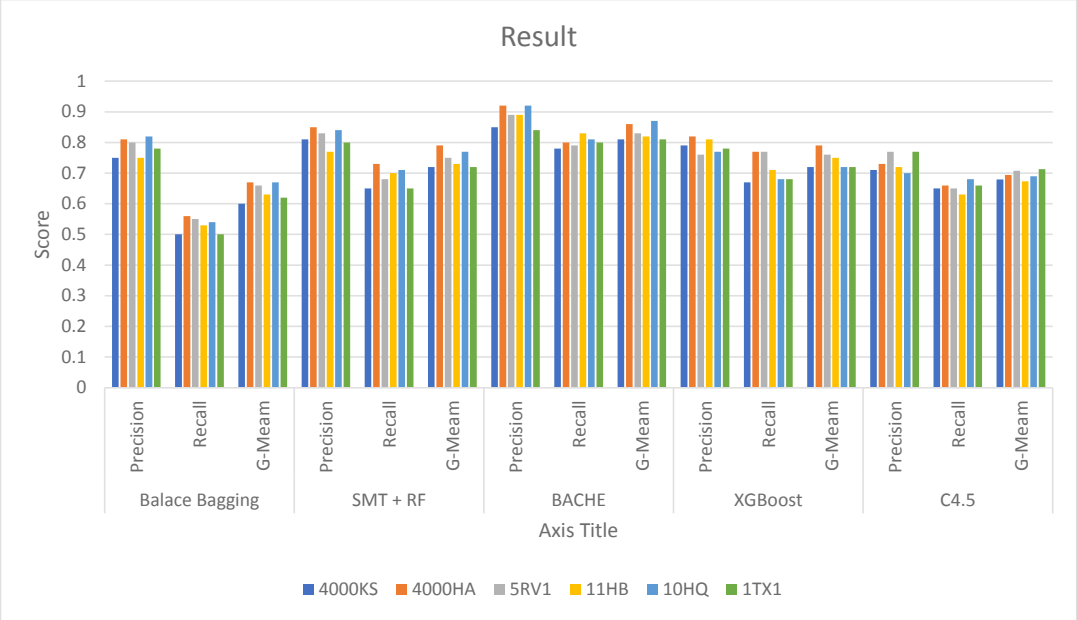


Figure 5 Comparison of performance of BACHE with other Algorithms

Tables 2 and Figure 5 presents the results of the experiment conducted. It can be observed that in all cases, the proposed BACHE algorithm outperforms the two algorithms in terms of recall and G-mean. The G-man's superior performance indicates the tradeoff between recognition in both classes, which is also a good classification effect for imbalanced datasets. Similarly, the high precision suggests that the false positive rate is low, and the high recall score indicates that the BACHE algorithm is sensitive to the minority class. Furthermore, Figure 5 shows how BACHE records a

significant percentage reduction in false positives compared to other methods. Although, the positive class (the minority class) is extremely rare. However, the BACHE algorithm is robust to skewed distribution by achieving a better result.

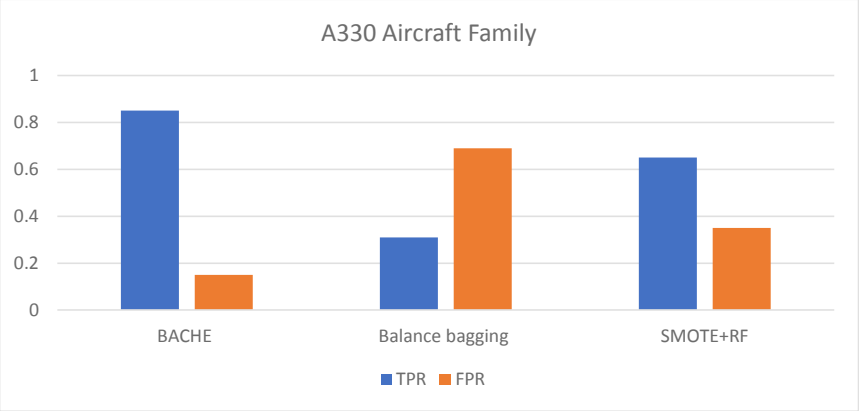
Table 2 Experimental Result using data from a fleet of A330 and A320 Aircraft families, the proposed BACHE algorithm is compared with baseline Balanced-bagging and other ensemble learning algorithms

Dataset (TFIN)	**IR%**	**Balance Bagging (baseline)**				**SMT +RF**				**BACHE**				**XGBoost (eXtreme Gradient Boosting)**				**C4.5**			
		Precision	Recall	GMean	Time (sec)	Precision	Recall	GMean	Time (sec)	**Precision**	**Recall**	**GMean**	Time (sec)	Precision	recall	GMean	Time (sec)	Precision	Recall	Gmean	Time (sec)
A330 (Long Range) Family																					
4000KS	0.0043	0.75	0.50	0.60	23	0.81	0.65	0.72	40	**0.85**	**0.78**	**0.81**	**51**	0.79	0.67	0.72	62	0.71	0.65	0.71	55
4000HA	0.0047	0.81	0.56	0.67	20	0.85	0.73	0.79	44	**0.92**	**0.80**	**0.86**	**55**	0.82	0.77	0.79	66	0.73	0.66	0.73	53
5RV1	0.0044	0.80	0.55	0.66	22	0.83	0.68	0.75	46	**0.89**	**0.79**	**0.83**	**53**	0.76	0.77	0.76	63	0.77	0.65	0.77	56
A320 (Short Aisle) Family																					
11HB	0.0028	0.75	0.53	0.63	26	0.77	0.70	0.73	42	**0.89**	**0.83**	**0.82**	**50**	0.81	0.71	0.75	61	0.72	0.63	0.72	55
10HQ	0.0031	0.82	0.54	0.67	28	0.84	0.71	0.77	40	**0.92**	**0.81**	**0.87**	**51**	0.77	0.68	0.72	64	0.70	0.68	0.7	54
1TX1	0.0021	0.78	0.50	0.62	25	0.80	0.65	0.72	41	**0.84**	**0.80**	**0.81**	**53**	0.78	0.68	0.72	66	0.77	0.66	0.77	55

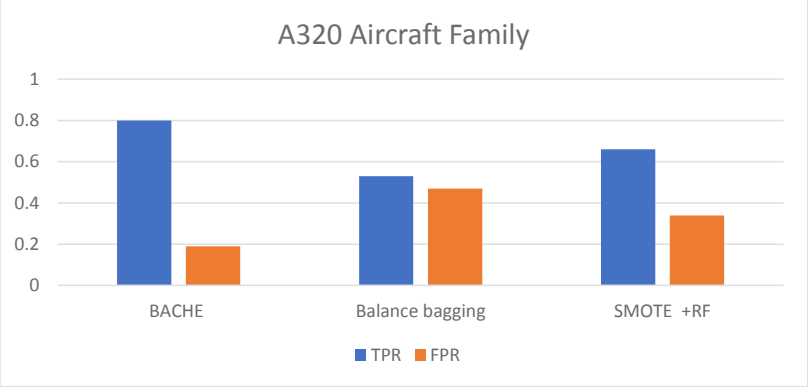
TFIN*:-Target Functional Item Number, ***IR*:- Imbalance Ratio, ***SMT***:- SMOTE, ***RF***:- Random Forest, BACHE:- Balanced Calibrated Hybrid Ensemble Technique, XGBoost:- eXtreme Gradient Boosting

It is also important to note that our goal is to achieve a G-mean score of greater than 50% as part of the target requirement for this study, which is the mean average of detecting extremely rare failure from the log-based dataset. The higher G-mean score for the BACHE algorithm shows that the model can distinguish the failure patterns leading to unexpected component replacement.

We also evaluate the proposed method's effectiveness in terms of false-positive and true-positive rates, considering the different aircraft families' datasets.



(a)

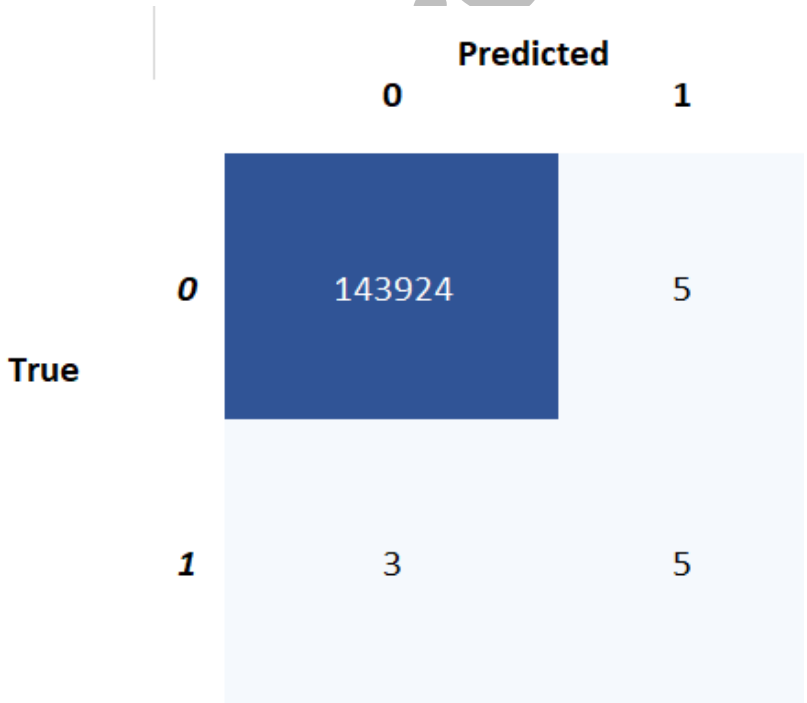


(b)

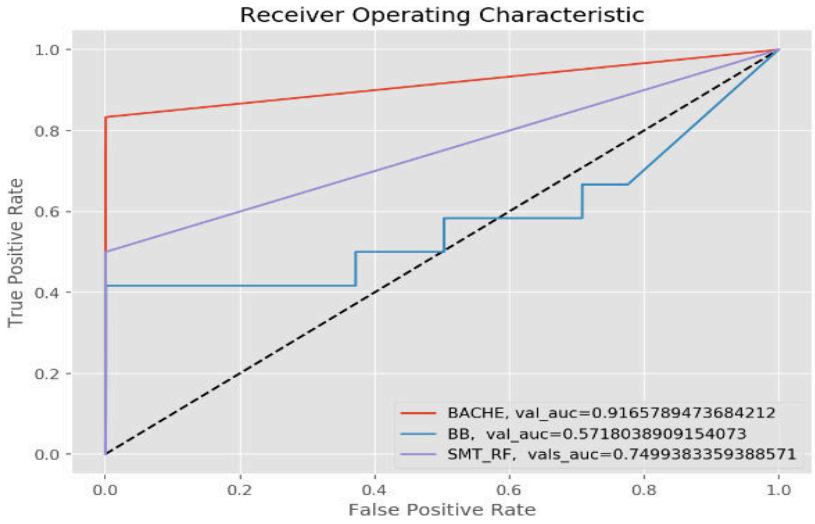
Figure 6 The average overall performance of each algorithm on the two aircraft families (a) A330 and (b) A320

Figure 6 shows the average FPR and TPR for each algorithm in both A330 and A320 aircraft families. BACHE averagely achieved a better (low) false-positive rate compared to the closest SMOTE+RF. In the A330 family, a balanced bagging algorithm has a predictive performance in terms of FPR of 69%, SMOTE+RF has 35%, while BACHE 15%. Comparing BACHE with closes SMOTE+RF, it is clear to see that there is a significant improvement of about 20%. Similarly, in the A320 family, the FPR for balance bagging is 47%, SMOTE+RF is 34%, and BACHE is 19%, showing an improvement of about 15%. The result validates the superior performance of BACHE in different aircraft families in the fleet.

Furthermore, another evaluation is the ROC curve reading, which shows that even though there is a significant percentage of false-positive rate (approximately 15%), the absolute probability is reasonably small.



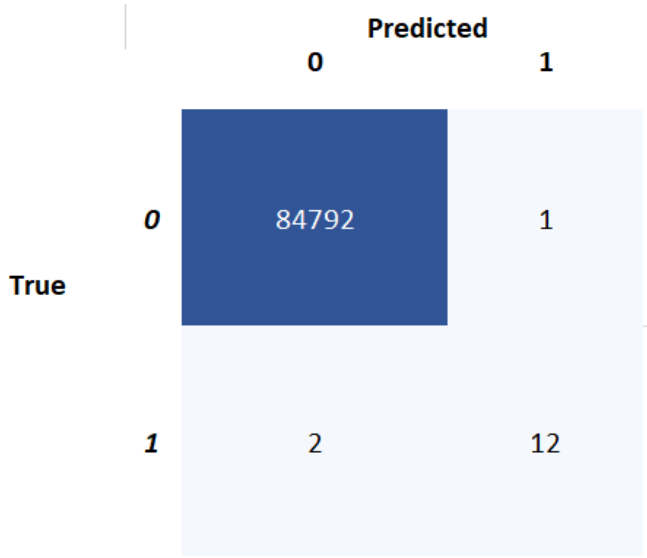
(a)



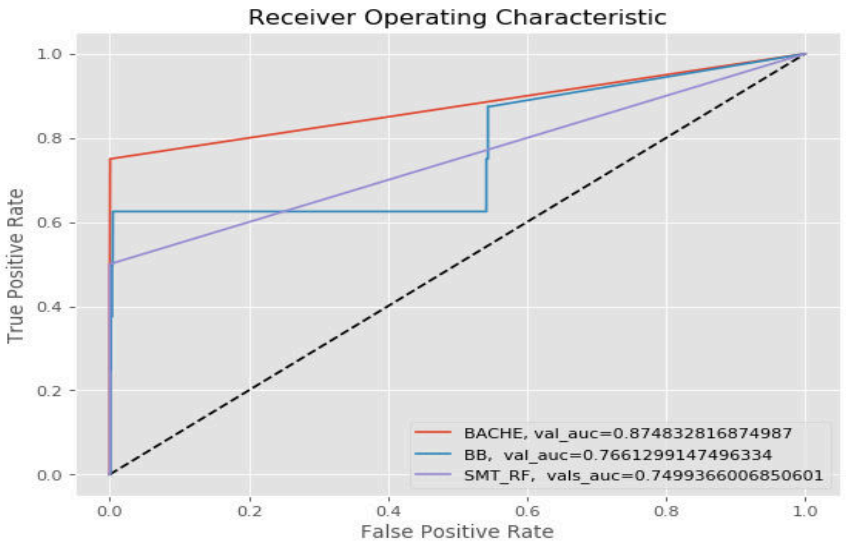
(b)

Figure 7 (a) is the Confusion Matric of BACHE prediction, and (b) the ROC-Curve showing the performance of the three algorithms considered in this study using data from the A330 aircraft family

As seen in Figure 7(a), the BACHE algorithm predicted 5 out of 8 unplanned failures, leading to the aircraft's pressure regulating valve replacement (FIN_4000HA). This prediction includes 10 flight cycles in advance. It can be observed that the model detected and predicts approximately 70% of extreme failure, which is a reasonable specificity, especially for aircraft maintenance. The area under the curve Figure 7(b) is 0.91. This shows that the BACHE algorithm can predict more than 90% of the probabilities of an observation belonging to each class in the A330 aircraft family.



(a)



(b)

Figure 8 (a) is the Confusion Matric of BACHE prediction and (b) ROC-Curve showing the performance of the three algorithms considered in this study using data from the A320 aircraft family

Figure 8 shows the predictive performance of BACHE on the A320 aircraft family. The result indicates that the BACHE algorithm predicted 12 out of 14 unplanned failures, as seen in Figure 8(a), leading to the aircraft flow control valve (FIN_11HB). The area under the curve is 0.87, as seen in 8(b). The BACHE algorithm can predict more than 85% probabilities of an observation belonging to each class in the A320 aircraft family.

We presented a confusion matrix and ROC for target functional items 4000HA and 11HB because the prediction performance is at the same range for other components in each aircraft family. We considered the remaining components from the A330 family, the electronic control unit/ electronic engine unit (4000KS), the satellite data unit (5RV1). The A320 family are the avionics equipment ventilation computer (10HQ) and the air traffic service unit (1TX1).

Also, it can be observed that the imbalanced ratio has an impact on performance. For instance, looking at Table 2 in cases where the IR is low, we obtain a lower G-mean compared to the ones with higher IR. For instance, in the A320 family, 1TX1 has the lowest IR of 0.21% and a G-mean score of 0.81, Compared to 10HQ with the highest IR of 0.31% and G-mean score of 0.87. Similar performance can be seen in the A330 family, where 4000KS has the lowest IR of 0.43% and the G-mean score is 0.81 compared to 400HA with the highest IR of 0.47% and G-mean score of 0.86. Despite the extremely imbalanced ratio in all the cases considered, our proposed algorithm still achieved better performance compared to other similar algorithms.

Another data factor that can impact the algorithm is the class small disjunct. Small disjunct arises when data in the same class is represented with different clusters (within class imbalance). The less represented small sub-clusters can further worsen classification performance degradation in an extreme imbalance dataset. We handled the challenge of class small disjunct problems intrinsically in the BACHE algorithm by clustering each class independently to identify clusters in each class. We subsequently oversampled sub-clusters in each class so that clusters in each class are balanced before the classification step.

One of the objectives of this study is the performance optimization of an imbalance learning algorithm. Evolution of the proposed BACHE against other similar algorithm was performed, the result displayed in Figure 9. Running each algorithm for classification of individual component failure. The result indicates that balance bagging has the fastest training time (averagely 20 seconds), with the XGBoost algorithm having the worst training time (averagely 60 seconds). In contrast to the

Proposed BACHE algorithm, which has an average training time of 50 seconds. Although Balance bagging and Random Forest (RF) show less computation time than BACHE, as observed, the difference is less than 20 seconds for balanced bagging and less than 10 seconds for the random forest. On the other hand, BACHE performed better in precision, recall, and G-mean (see table 2). Mis-classifying an example from the majority class as an example from the minority class is called a false-positive. False-positive is often not desired but less critical than classifying an instance from the minority class as belonging to the majority class, known as a false-negative. In the context of this study, false-negative means misclassifying fault as healthy, very critical as it can lead to equipment damage. In this study, false-positive means misclassifying a healthy component as a faulty component. This can result in the extra cost of maintenance checking. BACHE high precision indicates a less number of False Positives, and high recall means fewer False Negatives.

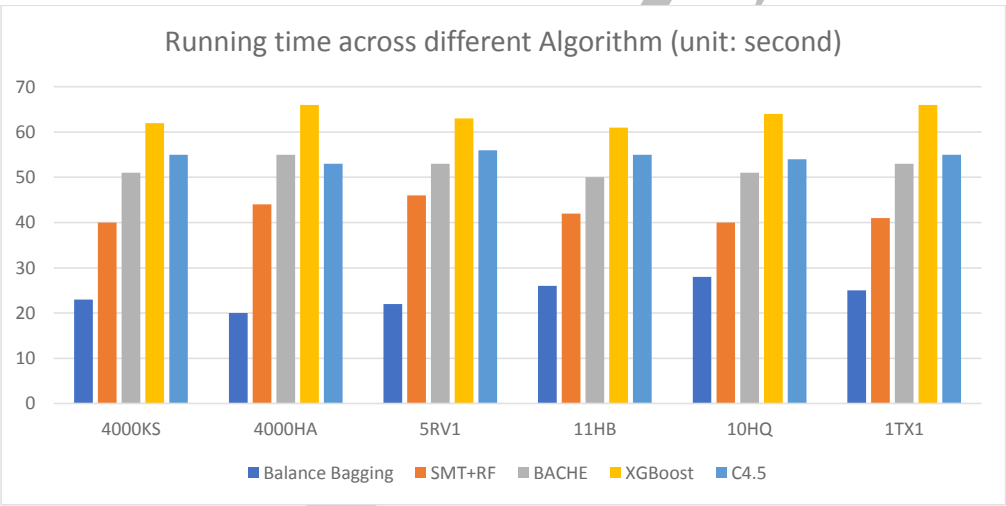


Figure 9 Comparing BACHE algorithm with other ensemble-based methods

G-mean is a metric that measures the balance between classification performances on both the majority and minority classes. G-mean measures the root of the product of class-wise sensitivity; it attempts to maximise each class's accuracy and keeps the accuracy balanced. It is a performance metric that correlates both. A low G-Mean indicates poor performance in the classification of the positive cases even if the negative cases are correctly classified as such. This measure is important in the avoidance of overfitting the negative class and underfitting the positive class. The algorithm can classify samples from both minority and majority classes which is shown in higher G-mean for BACHE compared to others.

6 Conclusion

This paper proposes and develops a novel imbalance-learning algorithm known as the Balance-Calibrated Hybrid Ensemble Technique (BACHE). The new algorithm is designed to handle extremely imbalanced classification problems in predictive modelling. Also, BACHE is trained using real-world test cases from the log-based central maintenance system data to produce a model for predicting aircraft component replacement. The novel approach significantly reduced false-positive and false-negative rates compared to similar approaches. The results showed that the model could predict aircraft component replacement within the target defined range; this contribution can enhance predictive maintenance in fleet reliability analysis. The model, when validated, can be used for predictive aircraft maintenance to improve the efficiency of the component replacement prognostic model. Though having a larger dataset from a different domain would offer further insight, this work focus on rare event prediction in aircraft predictive maintenance. In the future, the application of BACHE will be explored in domains other than aircraft. Also, the work can be developed further by studying the impact of class overlapping in the process of over-sampling the minority class.

7 Acknowledgement

Appreciation goes to the Integrated Vehicle Health Management Center (IVHM), Cranfield University, for allowing me to carry out this study and PTDF Nigeria for sponsoring the study

8 References

[1] Eickmeyer J, Li P, Givehchi O, Pethig F, Niggemann O. Data Driven Modeling for System-Level Condition Monitoring on Wind Power Plants. Int Work Princ Diagnosis 2015;1507:43–50.

[2] Sahal R, Breslin JG, Ali MI. Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case. J Manuf Syst 2020;54:138–51. https://doi.org/10.1016/j.jmsy.2019.11.004.

[3] Dangut MD, Skaf Z, Jennions IK. An integrated machine learning model for aircraft components rare failure prognostics with log-based dataset. ISA Trans 2021;113:127–39. https://doi.org/10.1016/j.isatra.2020.05.001.

[4] Wu Z, Lin W, Ji Y. An Integrated Ensemble Learning Model for Imbalanced Fault Diagnostics and Prognostics. IEEE Access 2018;6:8394–402. https://doi.org/10.1109/ACCESS.2018.2807121.

[5] Wang J, Ma Y, Zhang L, Gao RX, Wu D. Deep learning for smart manufacturing: Methods and applications. J Manuf Syst 2018:1–13. https://doi.org/10.1016/j.jmsy.2018.01.003.

[6] He H. Imbalanced Learning. New Jersey: John Wiley & Sons, Inc.,Hoboken, New Jersey.; 2011. https://doi.org/10.1002/9781118025604.ch3.

[7] Zhang Y, Li X, Gao L, Wang L, Wen L. Imbalanced data fault diagnosis of rotating machinery using synthetic oversampling and feature learning. J Manuf Syst 2018;48:34–50. https://doi.org/10.1016/j.jmsy.2018.04.005.

[8] Lee DH, Yang JK, Lee CH, Kim KJ. A data-driven approach to selection of critical process steps in the semiconductor manufacturing process considering missing and imbalanced data. J Manuf Syst 2019;52:146–56. https://doi.org/10.1016/j.jmsy.2019.07.001.

[9] Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. J Manuf Syst 2018;48:157–69. https://doi.org/10.1016/j.jmsy.2018.01.006.

[10] Branco P, Torgo L, Ribeiro RP. A Survey of Predictive Modeling on Imbalanced Domains. ACM Comput Surv 2016;49:1–50. https://doi.org/10.1145/2907070.

[11] Nghiem LT, Thu TT, Nghiem TT. MASI: Moving to adaptive samples in imbalanced credit card dataset for classification. 2018 IEEE Int. Conf. Innov. Res. Dev. ICIRD 2018, 2018, p. 1–5. https://doi.org/10.1109/ICIRD.2018.8376315.

[12] Sajana T, Narasingarao MR. A comparative study on imbalanced malaria disease diagnosis using machine learning techniques. J Adv Res Dyn Control Syst 2018;10:552–61. https://doi.org/https://www.jardcs.org/backissues/abstract.php?archiveid=2962&action=fulltext&uri=/backissues/abstract.php?archiveid=2962.

[13] Jiao Z, Jia G, Cai Y. A new approach to oil spill detection that combines deep learning with unmanned aerial vehicles. Comput Ind Eng 2018:1–12. https://doi.org/10.1016/j.cie.2018.11.008.

[14] Liu XY, Wu J, Zhou ZH. Exploratory under-sampling for class-imbalance learning. Proc. - IEEE Int. Conf. Data Mining, ICDM, IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS; 2006, p. 965–9. https://doi.org/10.1109/ICDM.2006.68.

[15] Lu Y, Cheung Y-M, Tang YY. Bayes Imbalance Impact Index: A Measure of Class Imbalanced Data Set for Classification Problem. IEEE Trans Neural Networks Learn Syst 2019;1:1–15. https://doi.org/10.1109/tnnls.2019.2944962.

[16] Ali A, Shamsuddin SM, Ralescu AL. Classification with class imbalance problem: A review. Int J Adv Soft Comput Its Appl 2015;7:176–204.

[17] Chang F, Zhou G, Zhang C, Xiao Z, Wang C. A service-oriented dynamic multi-level maintenance grouping strategy based on prediction information of multi-component systems. J Manuf Syst 2019;53:49–61. https://doi.org/10.1016/j.jmsy.2019.09.005.

[18] Ning F, Shi Y, Cai M, Xu W, Zhang X. Manufacturing cost estimation based on a deep-learning method. J Manuf Syst 2020;54:186–95. https://doi.org/10.1016/j.jmsy.2019.12.005.

[19] Fernández Alberto, Garcia Salvador, Galar Mikel, Prati Ronaldo, Krawczyk Bartosz HF. Learning From Imbalanced Data Sets. 2018. https://doi.org/https://link.springer.com/content/pdf/10.1007%2F978-3-319-98074-4.pdf.

[20] Haixiang G, Yijing L, Shang J, Mingyun G, Yuanyue H, Bing G. Learning from class-imbalanced data: Review of methods and applications. Expert Syst Appl 2017;73:220–39. https://doi.org/10.1016/j.eswa.2016.12.035.

[21] Abd Elrahman SM, Abraham A. A Review of Class Imbalance Problem. vol. 1. 2013. https://doi.org/www.mirlabs.net/jnic/index.html.

[22] Qiu M, Peng L, Pang Y, Yang B, Li P. Similarity-evaluation-based evolving of flexible neural trees for imbalanced classification. Appl Soft Comput 2021;111:107852. https://doi.org/10.1016/j.asoc.2021.107852.

[23] Chawla N V, Lazarevic A, Hall LO, Bowyer KW. SMOTEBoost : Improving Prediction. Lavrač N., Gamberger D., Todorovski L., Blockeel H. Knowl. Discov. Databases PKDD 2003. LNCS, vol. 2838, 2003, p. 107–19.

[24] Wu Z, Lin W, Ji Y. An Integrated Ensemble Learning Model for Imbalanced Fault Diagnostics and Prognostics. IEEE Access 2018;6:8394–402. https://doi.org/10.1109/ACCESS.2018.2807121.

[25] Chawla N V., Lazarevic A, Hall LO, Bowyer KW. SMOTEBoost: Improving Prediction of the Minority Class in Boosting 2003:107–19. https://doi.org/10.1007/978-3-540-39804-2_12.

[26] Chawla N V, Lazarevic A, Hall LO, Bowyer KW. SMOTEBoost: Improving Prediction of the Minority Class in Boosting 2003:107–19. https://doi.org/10.1007/978-3-540-39804-2_12.

[27] Sun M, Qian H, Zhu K, Guan D, Wang R. Ensemble learning and SMOTE based fault diagnosis system in self-organizing cellular networks. 2017 IEEE Glob. Commun. Conf. GLOBECOM 2017 - Proc., vol. 2018- Janua, 2018, p. 1–6. https://doi.org/10.1109/GLOCOM.2017.8254569.

[28] Han H, Wang WY, Mao BH. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. Lect. Notes Comput. Sci., vol. 3644, 2005, p. 878–87. https://doi.org/10.1007/11538059_91.

[29] Ng WWY, Liu Z, Zhang J, Pedrycz W. Maximizing minority accuracy for imbalanced pattern classification problems using cost-sensitive Localized Generalization Error Model. Appl Soft Comput 2021;104:107178. https://doi.org/10.1016/j.asoc.2021.107178.

[30] Domingos P, Ling CX, Sheng VS. MetaCost-AGeneralMethodforMakingClassifiersCostSensitivity. Encycl Mach Learn 2008:231–5. https://doi.org/10.1.1.15.7095.

[31] Bahnsen AC, Aouada D, Ottersten B. Example-dependent cost-sensitive decision trees. Expert Syst Appl 2015;42:6609–19. https://doi.org/10.1016/j.eswa.2015.04.042.

[32] Lu H, Xu Y, Ye M, Yan K, Gao Z, Jin Q. Learning misclassification costs for imbalanced classification on gene expression data. BMC Bioinformatics 2019;20:1–10. https://doi.org/10.1186/s12859-019-3255-x.

[33] Maheshwari S, Jain RC, Jadon RS. An insight into rare class problem: Analysis and potential solutions. J Comput Sci 2018;14:777–92. https://doi.org/10.3844/jcssp.2018.777.792.

[34] Liu XY, Zhou ZH. The influence of class imbalance on cost-sensitive learning: An empirical study. Proc - IEEE Int Conf Data Mining, ICDM 2006:970–4. https://doi.org/10.1109/ICDM.2006.158.

[35] Zhao P, Zhang Y, Wu M, Hoi SCH, Tan M, Huang J. Adaptive Cost-Sensitive Online Classification. IEEE Trans Knowl Data Eng 2019;31:214–28. https://doi.org/10.1109/TKDE.2018.2826011.

[36] Krawczyk B. Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 2016;5:221–32. https://doi.org/10.1007/s13748-016-0094-0.

[37] Zhou ZH. Ensemble methods: Foundations and algorithms. 2012. https://doi.org/10.1201/b12207.

[38] Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F. A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches. IEEE Trans Syst Man Cybern Part C Appl Rev 2012;42:463–84. https://doi.org/10.1109/TSMCC.2011.2161285.

[39] Lu W, Li Z, Chu J. Adaptive Ensemble Undersampling-Boost: A novel learning framework for imbalanced data. J Syst Softw 2017;132:272–82. https://doi.org/10.1016/j.jss.2017.07.006.

[40] Yuan X, Xie L, Abouelenien M. A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data. Pattern Recognit 2018;77:160–72. https://doi.org/10.1016/j.patcog.2017.12.017.

[41] Sun J, Lang J, Fujita H, Li H. Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates. Inf Sci (Ny) 2018;425:76–91. https://doi.org/10.1016/j.ins.2017.10.017.

[42] Feng W, Huang W, Ren J. Class imbalance ensemble learning based on the margin theory. Appl Sci 2018;8. https://doi.org/10.3390/app8050815.

[43] Feng W, Huang W, Ren J. Class Imbalance Ensemble Learning Based on the Margin Theory. Appl Sci 2018;8:815. https://doi.org/10.3390/app8050815.

[44] Zhou ZH. Ensemble methods: Foundations and algorithms. 2012.

https://doi.org/10.1201/b12207.

[45] Liu XY, Wu J, Zhou ZH. Exploratory under-sampling for class-imbalance learning. Proc. - IEEE Int. Conf. Data Mining, ICDM, IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS; 2006, p. 965–9. https://doi.org/10.1109/ICDM.2006.68.

[46] Schapire RE. A brief introduction to boosting. IJCAI Int. Jt. Conf. Artif. Intell., vol. 2, 1999, p. 1401–6. https://doi.org/citeulike-article-id:765005.

[47] Vluymans S, Triguero I, Cornelis C, Saeys Y. EPRENNID: An evolutionary prototype reduction based ensemble for nearest neighbor classification of imbalanced data. Neurocomputing 2016;216:596–610. https://doi.org/10.1016/j.neucom.2016.08.026.

[48] Le T, Vo MT, Vo B, Lee MY, Baik SW. A Hybrid Approach Using Oversampling Technique and Cost-Sensitive Learning for Bankruptcy Prediction. Complexity 2019;2019:1–12. https://doi.org/10.1155/2019/8460934.

[49] David Dangut M, Skaf Z, Jennions I. Rescaled-LSTM for Predicting Aircraft Component Replacement Under Imbalanced Dataset Constraint. 2020 Adv. Sci. Eng. Technol. Int. Conf., IEEE; 2020, p. 1–9. https://doi.org/10.1109/ASET48392.2020.9118253.

[50] Lee J, Lee YC, Kim JT. Fault detection based on one-class deep learning for manufacturing applications limited to an imbalanced database. J Manuf Syst 2020;57:357–66. https://doi.org/10.1016/j.jmsy.2020.10.013.

[51] Krawczyk B. Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 2016;5:221–32. https://doi.org/10.1007/s13748-016-0094-0.

[52] Masnadi-Shirazi H, Vasconcelos N. Cost-Sensitive Boosting. {IEEE} Trans Pattern Anal Mach Intell 2016;33:294–309.

[53] Krawczyk B, Woźniak M, Schaefer G. Cost-sensitive decision tree ensembles for effective imbalanced classification. Appl Soft Comput J 2014;14:554–62. https://doi.org/10.1016/j.asoc.2013.08.014.

[54] Kull M, Silva Filho TM, Flach P. Beyond Sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. Electron J Stat 2017;11:5052–80.

https://doi.org/10.1214/17-EJS1338SI.

[55] Zadrozny B, Elkan C. Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. Icml 2001:1–8.

[56] Dal Pozzolo A, Caelen O, Bontempi G, Johnson RA. Calibrating Probability with Undersampling for Unbalanced Classification Fraud detection View project Volatility forecasting View project Calibrating Probability with Undersampling for Unbalanced Classification 2015. https://doi.org/10.1109/SSCI.2015.33.

[57] Guo H. Learning from Imbalanced Data Sets with Boosting and Data Generation : The DataBoost-IM Approach n.d.;6:30–9.

[58] Liu X-Y, Wu J, Zhou Z-H. Exploratory Undersampling for Class Imbalance Learning. IEEE Trans Syst Man Cybern 2009;39:539–50. https://doi.org/10.1109/TSMCB.2008.2007853.

[59] Masnadi-Shirazi H, Vasconcelos N. Cost-Sensitive Boosting. {IEEE} Trans Pattern Anal Mach Intell 2011;33:294–309.

[60] Hastie T, Tibsshirani R, Friedman J. The Elements os Statistical Learning. 2nd ed. New York, New York, USA: Springer US; 2009. https://doi.org/10.1007/b94608.

[61] Friedman J, Hastie T, Tibshirani R. Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors). Ann Stat 2000;28:337–407. https://doi.org/10.1214/aos/1016218223.

[62] Lee DH, Yang JK, Lee CH, Kim KJ. A data-driven approach to selection of critical process steps in the semiconductor manufacturing process considering missing and imbalanced data. J Manuf Syst 2019;52:146–56. https://doi.org/10.1016/j.jmsy.2019.07.001.

[63] Chen T, Guestrin C. XGBoost: A scalable tree boosting system. Proc ACM SIGKDD Int Conf Knowl Discov Data Min 2016;13-17-Augu:785–94. https://doi.org/10.1145/2939672.2939785.

[64] Quinlan JR. Improved Use of Continuous Attributes in C4.5 2006:77–90.

[65] Knuth DE. Big Omicron and Big Omega and Big Theta (1976). Ideas That Creat Futur

2021:441–6. https://doi.org/10.7551/mitpress/12274.003.0045.

[66] Atamazhori S, Hassanvand A, Omidvar S. On Asymptotic Notation : an Introduction to Analyses of Algorithms On Asymptotic Notation : an Introduction to Analyses of Algorithms 2021:0–6.

Highlights for Review

Advanced techniques for the analysis of an aircraft central maintenance system -CMS dataset to develop reliable vehicle health predictive models is required.

Imbalance dataset is still a challenge faced in building reliable data-driven prognostic models.

BACHE algorithm for handling extreme imbalance problems is proposed.

Important log messages that hold direct links to the causes of aircraft component failure leading to replacement are shown.

The result shows the effective handling of data imbalanced problem produces a high-performance model for aircraft predictive maintenance.

CRediT author statement

Maren David Dangut: Conceptualization, Methodology, Software, Validation, Data curation, Writing-Original draft preparation.
Maren David Dangut and Zakwan Skaf: Visualization, Investigation.
Ian K Jennions: Supervision.
Ian K Jennions and Zakwan Skaf : Reviewing and Editing.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships, which may be considered as potential competing interests:

Maren David Dangut
Ian K. Jennions
Zakwan Skaf

2022-05-14

Handling imbalanced data for aircraft predictive maintenance using the BACHE algorithm

Dangut, Maren David

Elsevier

Dangut MD, Skaf Z, Jennions IK. (2022) Handling imbalanced data for aircraft predictive maintenance using the BACHE algorithm, Applied Soft Computing, Volume 123, July 2022, Article number 108924

https://doi.org/10.1016/j.asoc.2022.108924