Empirical Oversampling Threshold Strategy for Machine Learning Performance Optimisation in Insurance Fraud Detection

Bouzgarne Itri[1], Youssfi Mohamed[2], Bouattane Omar[3], Qbadou Mohamed[4]
SSDIA Laboratory
ENSET Mohammedia University Hassan 2
Casablanca, Morocco

***Abstract*—Insurance fraud is one of the most practiced frauds in the sectors of the economy. Faced with increasingly imaginative underwriters to create fraud scenarios and the emergence of organized crime groups, the fraud detection process based on artificial intelligence remains one of the most effective approaches. Real world datasets are usually unbalanced and are mainly composed of "no-fraudulent" class with a very small percentage of "fraudulent" examples to train our model, thus prediction models see their performance severely degraded when the target class appears so poorly represented. Therefore, the present work aims to propose an approach that improves the relevance of the results of the best-known machine learning algorithms and deals with imbalanced classes in classification problems for prediction against insurance fraud. We use one of the most efficient approaches to re-balance training data: SMOTE. We adopted the supervised method applied to automobile claims dataset "carclaims.txt". We compare the results of the different measurements and question the results and relevance of the measurements in the field of study of unbalanced and labeled datasets. This work shows that the SMOTE Method with the KNN Algorithm can achieve better classifier performance in a True Positive Rate than the previous research. The goal of this work is to lead a study of algorithm selections and performance evaluation among different ML classification algorithms, as well as to propose a new approach TH-SMOTE for performance improvement using the SMOTE method by defining the optimum oversampling threshold according to the G-mean measure.**

Keywords*—Machine learning; oversampling; SMOTE; insurance fraud

I. Introduction

Insurance fraud costs several million dollars each year. In 2017, the Insurance Fraud Bureau of Australia (IFBA) detected $280 million in fraudulent claims [1]. In Morocco, according to the Economist newspaper [2], the FMSAR claims that traffic accident compensation fraud accounts for more than 21% of insurers' compensation. In France, according to the Agency for the Fight against Insurance Fraud (ALFA) 44814 acts were detected in 2013, for a recovered amount of 214 million Euros; In 2018, the amount is increased to 500 million Euro [3].

Fraudulent cases often have relatively similar characteristics to non-fraudulent cases, which also depends on the information entered by claim handlers on the system and its relevance. What makes fraud detection very difficult is that there is no particular variable or rule to characterize fraud cases in a simple and robust way. Thus, the use of automatic detection models based on machine learning algorithms becomes an operational necessity to fight fraud effectively. In our case study, we focus on the type of fraud in automobile claims. Our model is based on the supervised method to solve a classification problem. We build a statistical learning model to predict the affiliation of claims reported to one of the following classes (fraudulent claims, non-fraudulent claims).We try also to find the best classification model to estimate a high probability of belonging to the "fraudulent claim" class. A comparative analysis of ten best-known machine-learning algorithms are presented in this work. However, one of the main problems of these machine learning models, as in our case, is that they suffer from the problem of imbalanced classes in the data set. The class of fraudulent claims represents only 6% in our dataset. Indeed, when a binary classification problem has a lot less data in a “fraudulent” class than in a “non fraudulent” class, some machine learning algorithms will simply learn to ignore the minority class and classify all cases into the majority class, because this will trivially yield high classification accuracy, but the performance of the prediction models will be strongly degraded. Common methods to address this problem are called sampling techniques. Furthermore, the Synthetic Minority Oversampling Technique (SMOTE) [4] method is known as the pioneer in the development of oversampling techniques based on synthetic data. Based on the SMOTE method, we are inspired to develop and process our unbalanced dataset, notably to oversample the minority class in order to improve the performance, compare algorithms and present a new approach Threshold Synthetic Minority Oversampling Technique (TH-SMOTE) to determine an optimal oversampling threshold with a G-mean Score.

Our paper is organized as follows. Section 2 provides a brief overview of the research conducted in this study and the problem of data imbalance. In Section 3, we detail the methodology of our approach, including the different steps of data preparation, the oversampling method, and the different evaluation measures. Section 4 presents the result of our experience. We discuss the performance results between the different algorithms through the iterations of oversampling, comparing our result with previous study. Finally, we present our conclusions in Section 5.

II. Related Work

Several authors find a modeling approach that sheds some light on the empirical investigation of fraud. They worked on data analytics and data mining approaches to improve model performance in the prediction, but some authors have addressed the same problem of our study, by dealing with the same dataset known as "carclaims.txt". Xu et al. [5] in 2011, proposed a neural network combined with a random rough subspace method to improve the consistency in the datasets. Sundarkumar et al. in 2015 [6], proposed an hybrid approach for rectifying the imbalance dataset problem by employing k Reverse Nearest Neighborhood and one class support vector machine (OCSVM). Nian et al. in 2016 [7], proposed an unsupervised spectral ranking method for detection anomaly (SRA) of forged instances in fraud detection problems, using auto insurance claim dataset. S. Subudhi and S. Panigrahi [8] in 2017, proposed a hybrid approach for detecting frauds in automobile insurance claims by applying Genetic Algorithm (GA) based Fuzzy C-Means (FCM) clustering and various supervised classifier models. Itri and Youssefi [9] in 2019, presented a new approach to improving the probability of fraud predictions by resampling methods with imbalanced dataset, as well as the methodology for evaluating performance of the ten best-known machine learning algorithms.

Furthermore, the problem of learning performance related to unbalanced datasets has received attention in different areas of research. Tora et al. (2019) [10] subdivided the resolution into three groups: Solution at data level, Solution at algorithm level and Hybrid solution. We focus like most of the articles on the techniques in the solution at the data level. Kubat and Matwin (1997) [10] applied the under sampling technique selectively on the majority class, while retaining the original population of the minority class. They applied the selection technique by subdividing the minority examples into four groups to eliminate overlapping noise data in the borderline region, as well as redundant samples. Chawla et al. (2002) [4] proposed the Synthetic Minority Oversampling Technique (SMOTE) method as a new approach to over-sampling the minority class, they are the founders of the SMOTE method, which has proven its worth in the problems of unbalanced datasets and oversampling techniques, in their approach. They have revolutionized the classical oversampling method, which inspired several authors [9][12][13][14] by proposing new methods derived from SMOTE to improve or remedy these weaknesses, such as neighbors, noise and wrong sample generation.

III. Proposed Methodology

In order to have a good choice between the classification algorithms, we have chosen to compare the performance of the ten best-known algorithms: KNN, C4.5, Naive Bayes, Random Forest, Multilayer Perceptron, Machine Vector Support, Logistic, Partial Decision Trees (PART), Decision Table, and Adaptive. Before training our model, we processed the data and subsequently applied the SMOTE oversampling method by increasing the percentage of the minority class, iteration by iteration, until the threshold where the performance becomes optimal. We chose a tenfold cross-validation method to train the model [15]. In the end, several classification measures are defined and discussed to evaluate these classifiers for each iteration. These steps are detailed in the following sections.

A. Data Collection and Data Pre-Processing

The data set for our study is represented by real data from an anonymous insurance company on automobile claims provided by Angoss Knowledge Seeker Software, known in the literature as "carclaims.txt". The dataset comprises 15420 claims reported between January 1994 and December 1996, with 32 predictor variables and one target variable representing the values 1 "Fraud" and 0 "No Fraud". Fraudulent claims constitute 6% of the data set, i.e. 923 samples. The data set is in CSV format. We have converted the attributes to nominal because, on the one hand, several classifiers only support nominal attribute types, on the other hand, when importing data, heuristics cannot always predict the exact type of the attribute.

B. Data Sampling

When working with real-world data such as the case of our study, in general cases, the datasets are highly composed of "normal" instances and only a small percentage represent target and abnormal instances. "Sampling" is a pre-processing procedure whose objective is to address the imbalance of a given data set by increasing or decreasing the training data before building a model, either by increasing examples from the minority class (over-sampling) or removing examples from the majority class (under-sampling). This one is only relevant if we have a large amount of data, which is not the case for our dataset. Therefore, we propose an over-sampling method SMOTE [4], whose approach generates new examples of the minority class by combining the data with those of their nearest neighbors, judged by the Euclidean Distance. But our new approach aims to define an optimal threshold of oversampling based on the evolution of the G-mean indicator and SMOTE method.

C. Classification Evaluation

This section presents the application of the methods to train our model and the evaluation and selection criteria to decide on the best performing classification algorithms given a case study.

Through the confusion matrix a whole bunch of performance criteria can be derived. The following Table I presents the confusion matrix for a binary classifier with four different combinations of predicted and actual values.

TABLE I. Confusion Matrix

Actual \ Prediction	Fraud	No Fraud
Fraud	TP - true positive	FN - false negative
No Fraud	FP - false positive	TN - true negative

The following are the various measures derived from the indicators obtained through the fusion matrix:

1) Accuracy: is the number of correct predictions (TP and TN) made by the model over all kinds predictions made. Accuracy is a great measure but only when the given dataset is symmetric and balanced. However, when the data is imbalanced as in the case of our study, accuracy doesn't really capture the effectiveness of a classifier, because our models look at the data and cleverly decide that the best thing to do is to always predict "NoFraud Class" and achieve high accuracy (accuracy paradox). The formula is given by

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

2) F-Measure: Several evaluation criteria can be used, indicating the better or worse performance of a prediction function. But it is difficult to compare two models with low precision and high recall or vice versa. To facilitate the interpretation of the algorithm performance, Van Rijsberjen, 1979 [16] created a synthetic measure F1-Measure or F-score, defined as the harmonic mean of the precision and recall of a binary decision rule. It is given by.

$$F - Measure = \frac{\left((1 + \beta^2) \times Precision \times Recall\right)}{\left((\beta^2 \times Precision) + Recall\right)}$$

Precision is a measure that tells us what proportion of claims that we predict as fraudulent (TP and FP), actually are fraudulent (TP). It is given by.

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

Recall or Sensitivity is a measure that tells us what proportion of claims that actually fraudulent (TP and FP), was predicted by the algorithm as fraudulent (TP). It is given by.

$$Recall = \frac{TP}{TP + FN}$$

The β parameter determines the weight of precision in the combined score. If we set parameter β to 1, it means that precision and recall have equal importance. β< 1 lends more weight to precision, while β > 1 favors recall.

3) AUC-ROC: To support and compare the results of our study, we introduced another measure, Area Under Receiver Operating Characteristic (AUC-ROC) curve. The ROC curve is the plot between Recall and (1- specificity), the greater its value, the more predictive the model is able to distinguish between fraudulent and non-fraudulent groups.

4) G-mean: Called Geometric Mean, was proposed by Kubat et al. (1997) [11]. This evaluation parameter shows the balance between sensitivity and specificity, calculated as:

$$G - mean = \sqrt{Recall \times Specifity}$$

Specificity is a measure that tells us what proportion of claims that actually not fraudulent, was predicted by the model as not fraudulent. It's the opposite of Recall, that's why we're going to settle for this one instead. It is given by.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

We used G-mean measure to decide on the oversampling threshold, by studying the evolution of the two indicators recall and specificity through the increase of the minority class percentage. Thus, the oversampling threshold will be calculated when the evolution curve becomes stagnant as the percentage of the minority class increases. In other words, the threshold is equal to the percentage of the minority class where the values of G-mean begins to converge towards a constant and its derivative is around zero.

IV. Experimental Analysis

A. Experimental Results Discussion

To improve the model's performance, we will study the evolution of the measures according to the minority class percentage after each SMOTE oversampling iteration. As a first step, we start with an overview of the performance for all contending algorithms. We calculate the overall average of each measure in each iteration for all algorithms whose results and evolution are shown in Fig. 1.

Before applying oversampling, we have a high value of Accuracy and specificity measurement, due to the low percentage of the minority class. Our model tends to predict that almost all claims are non-fraudulent, so we confront the accuracy paradox. Therefore, the accuracy measurement is not a relevant measure for evaluating the performance of an unbalanced dataset. On the other hand, the Specificity has an average of 99% while the average recall is less than 1%, but the Recall remains a more important metric to consider in the process of detecting fraudulent claims, because it is important to detect all possible frauds, even if it means that the insurer may have to tolerate some false positives. This involves that we will have to give more weight to the recall in the β parameter of the F-measure formula.

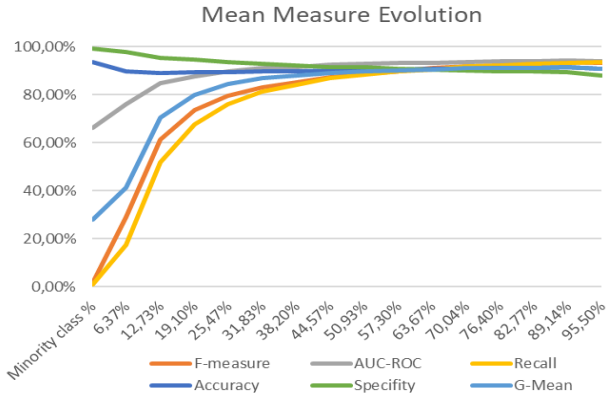

Fig. 1. Average Evolution of the Overall Algorithm Measures per Oversampling Iteration.

However, to compare and illustrate the importance of the parameter β for the evaluation according to the F-measure, we will keep the parameter β=1 in a first step as shown in Fig. 1. Then we scale this parameter in a second step when the results of the algorithms evolve and become closer. We applied fifteen iterations to oversample – with the SMOTE method - the minority class, increasing its percentage in the dataset to 12.73%, 19.10%, 25.47% ... up to 100% where both fraudulent and non-fraudulent classes will have the same proportion. After the first iteration (12.73%) as shown in Fig. 1, we see that the average measures results increase significantly after the first oversampling operations, even exponentially for the F-Measure and Recall, except for the Accuracy (accuracy paradox), which decreases in the first iterations. After the second iteration (25.47%), the average of the measurements, including accuracy, increases linearly to reach higher values close to 100%.

If we have a closer look at the measures by the five algorithms that inflate as presented in Table II, focusing on the top five algorithms with the highest F-measure and AUC-ROC values, the recall of KNN algorithm makes a considerable leap from a score of 8% to 79.60% after the first iteration (12.7%), far surpassing the other algorithms. As for the Area Under ROC values, KNN is ranked second with a value of 91.86% not far from the first place attributed to the Random Forest with 93.26%, but has a low Recall value with 22.48%. While the value of F-measure is increased by KNN with the highest value 70.82%, against Random Forest which holds the value 36.63% well far from the first place held by KNN.

Up to now, in this first iteration we can conclude that KNN takes the lead in the ranking of the algorithms, reacts faster on the Recall indicator. On the other hand, considering the importance of the recall measure, the Area Under Roc measure remains irrelevant at this stage in comparison with F-measure to evaluate and rank the algorithms' performances.

In the last iteration (100% SMOTE), as presented in Table III when the minority class reaches the same proportion as the majority class, Random Forest increases all measures including F-measure except the recall measure which is held by KNN with a value of 98.94%, followed by C4.5.

On the other hand, for the performance ranking according to the F-measure, we have considered until now that the recall and the precision have the same importance when we assigned the β parameter value to 1. However, we should give more attention to the Recall measure compared to the precision. In this case we need to give more weight to the recall (β>1). Thus, we assign the value 2 to β, meaning that recall is twice as important for us. According to the F-Measure results in Table III, we observe that the weight of the Recall indicator is taken into account, with an increase in the KNN algorithm not far from Random Forest. We conclude that KNN remains the most efficient algorithm to improve the model from the first iteration to the last. It is the most effective algorithm to increase the percentage of fraudulent instance we recalled from all fraudulent instance.

As some regions of minority and majority class groups are closely neighboured, SMOTE may overgeneralize the region of minority classes that is in the proximity of majority classes. Thus, new noisy instances may be generated [17], and may reduce the reliability of predictions for both minority and majority classes. If we repeat this generation several times, our model may deviate from reality because of the noisy instances, despite the improvement of the indicators over iterations. Therefore, according to the evolution of the performance of the KNN algorithm through the iterations of oversampling (see Fig. 2), if we look at the curve of the G-means indicator, the optimal is reached at the third iteration (25.47%), at this level the values of the measurements are very optimal to constitute our prediction model (Recall = 95.10%; Specifity = 90.62%; AUC-ROC = 95.37%).

TABLE II. PERFORMANCE ANALYSIS AFTER FIRST OVERSAMPLING ITERATION

Model	**Precision**	**Recall**	**Accuracy**	**Specifity**	**AUC-ROC**	**F-measure**
KNN	63,73%	**79,69%**	92,58%	94,23%	**91,86%**	**70,82%**
PART	55,33%	39,06%	89,56%	95,99%	83,09%	45,79%
Naive Bayes	42,60%	45,83%	86,91%	92,14%	83,83%	44,15%
M.Perceptron	56,92%	29,20%	89,51%	97,19%	81,33%	38,60%
RandomForest	98,81%	22,48%	91,21%	99,97%	**93,26%**	**36,63%**

TABLE III. PERFORMANCE ANALYSIS IN THE LAST OVERSAMPLING ITERATION (100% SMOTE)

Model	**Precision**	**Recall**	**Accuracy**	**Specifity**	**F-measure**	**F-measure**
					β = 1	**β = 2**
KNN	88,47%	**98,94%**	93,02%	87,10%	93,41%	**57,99%**
RandomForest	99,42%	95,29%	97,37%	99,45%	**97,31%**	57,65%
M.Perceptron	90,07%	94,49%	92,03%	89,58%	92,22%	56,14%
C4.5	92,14%	**96,26%**	94,03%	91,79%	94,16%	57,24%
PART	94,60%	95,56%	95,05%	94,55%	95,08%	57,22%

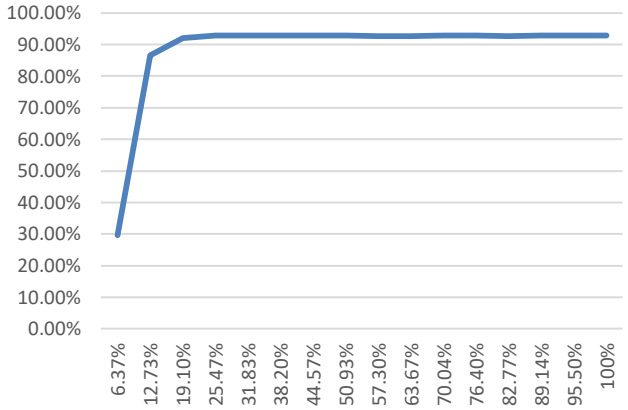

Fig. 2. G-Mean Evolution for KNN Algorithm through Iterations of SMOTE Percentage.

B. Comparison with Previous Literature in the Same Scope

Several research works have addressed the same dataset. We present in Table IV, a comparison of the expected results by referring to the same measures (accuracy, recall, Specificity) used in the same literature.

TABLE IV. COMPARATIVE PERFORMANCE ANALYSIS USING CARCLAIMS.TXT

Research Articles	Accuracy	Recall	Specificity
Xu et al. (2010)	88,7	-	-
Sundarkumar et al. (2015)	58,92	95,52	56,58
Sundarkumar and Ravi (2015)	60,31	90.79	58,69
Nian et al. (2016)	-	91	52.00
Sharmila and Panigrahi. (2017)	87,02	83,21	88,45
Bouzgarne and Youssefi (2020)	91,52	95,1	90,62

We obtained the best performance compared to previous studies. We can still improve the results by pushing the oversampling percentage to 100% when the dataset becomes balanced and symmetrical, however this will generate a large amount of non-real data that will penalize the quality of the training data, impacting the model veracity with the risks of oversampling. This is why in our approach, we oversampled the minority class only up to the threshold (25%), determined by the mean G curve, thus achieving a better result.

V. CONCLUSION

In this article, an approach TH-SMOTE to improving the performance of the prediction model in auto insurance fraud was proposed. A comparative study of machine learning classification algorithms applied on a labeled and unbalanced dataset was carried out. Over-sampling based on the SMOTE method was applied, by increasing the minority class by iteration to the threshold where the measurements became more significant for the model. Particular attention was given to the performance evaluation methods of the classification model. The relevance of the results of the most known measure to classify the algorithms were discussed, by giving more weight to the Recall indicator, one of the most important indicators in detecting fraud.

The results show that the TH-SMOTE approach can significantly improve the performance of the classifiers for the whole set of classification algorithms. In particular, the KNN algorithm reacts faster to the first over-sampling percentages to offer better model performance, notably through the Recall indicator.

Furthermore, we have shown that the F-Measure allows a better comparison of the performance of the classification algorithms for unbalanced datasets cases in comparison with the AUC-ROC measurement, provided that more weight is given to the True Positive Rate (recall) via its β parameter. In contrast, the G-mean measure allowed us to measure and define the threshold of the percentage of oversampling, along with the minority class to avoid too much over-sampling at the expense of the model quality. Specifically, it is to prevent overfitting and to reduce generating noisy examples into the dataset without skewing the performance results.

Finally, this study's results were compared with previous research on the same scope, our approach found a higher result.

Although the experimental results of this study have proven that this approach allows a comparison and choose the best algorithm for the case of an unbalanced dataset using the TH-SMOTE method combined with G-mean, there are also other research works in the same area of study which offer variants of the SMOTE method. Hence, the extension of our approach to these variants may be possible in future work.

REFERENCES

[1] Insurance Fraud Bureau of Australia Homepage, https://ifba.org.au, last accessed 30/05/2020.

[2] Assurance auto: La fraude explose.Edition N°:5302, 27/06/2018. https://www.leconomiste.com/article/1030288-assurance-auto-la-fraude-explose, last accessed 30/05/2020.

[3] French Agency for the Fight against Insurance Fraud – ALFA. https://www.alfa.asso.fr/.

[4] Chawla et al. 2002. "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research16 : 321–357.

[5] Xu, W., Wang, S., Zhang, D., Yang, B., 2011. Random rough subspace based neural network ensemble for insurance fraud detection. In: Computational Sciences and Optimization (CSO), 2011 Fourth International Joint Conference on. IEEE, pp. 1276–1280.

[6] Sundarkumar, G.G., Ravi, V., 2015. A novel hybrid undersampling method for mining unbalanced datasets in banking and insurance. Eng. Appl. Artif. Intell. 37, 368–377.

[7] Nian, K., Zhang, H., Tayal, A., Coleman, T., Li, Y., 2016. Auto insurance fraud detection using unsupervised spectral ranking for anomaly. J. Finance Data Sci. 2 (1), 58–75.

[8] Sharmila Subudhi, Suvasini Panigrahi, 2017. Use of optimized Fuzzy C-Means clustering and supervised classifiers for automobile insurance fraud detection; Department of Computer Science and Engineering & IT, Veer Surendra Sai University of Technology, Burla, Odisha 768018, India.

[9] Bouzgarne Itri and Youssefi Mohammed, et al.2019:"Performance comparative study of machine learning algorithms for automobile insurance fraud detection".2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS), 28-30 Oct. 2019.

[10] Tora Fahrudin, et al. 2019. Enhancing the performance of smote algorithm by using attribute weighting scheme and new selective sampling method for imbalanced data set. ICIC International, 2019 ISSN 1349-4198, pp. 423-444.

[11] Miroslav Kubat and Stan Matwin: "Addressing the Curse of Imbalanced Training Sets: One-Sided Selection". Proceedings of the 14th International Conference on Machine Learning, (1997)179-186.

[12] E.Ramentol, Y.Caballero, R.Bello, F.Herrera, 2012. Smote-rsb: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using smote and rough sets theory. Knowl. inf. Syst. 33 (2012), 245–265.

[13] S. Barua et al., Mwmote–majority weighted minority oversampling technique for imbalanced data set learning, IEEE Trans. Knowl. Data Eng. 26 (2014), 405–425.

[14] Z. Zheng et al., Oversampling method for imbalanced classification, Comput. Informat. 34 (2016), 1017–1037.

[15] Refaeilzadeh et al. 2009. Cross-validation. In: Encyclopedia of Database Systems. Springer, pp. 532–538.

[16] C. J. V. Rijsbergen, 1979. Information Retrieval. Butterworth-Heinemann, Newton, MA, USA, 2nd edition.

[17] Haibo He and Edwardo A Garcia. 2009. Learning from imbalanced data. Knowledge and Data Engineering, IEEE Transactions on, 21(9):1263-1284.