Imbalanced Datasets: Towards a better classification using boosting methods

DJAFRI Laouni[1] and BACHA Soufiane[2]

[1] Department of Computer Science, Ibn Khaldoun University, Tiaret, Algeria, EEDIS laboratory, Djillali Liabes University, Sidi Bel Abbes, Algeria
djaafri29tp@gmail.com,
[2] Department of Computer Science, Ibn Khaldoun University, Tiaret, Algeria, Bˆatiment 425, Tiaret, Algeria
gl.soufianebacha@gmail.com,

Abstract. Imbalanced datasets classification is inherently difficult. This situation becomes a challenge when amounts of data are processed to extract knowledge because traditional learning models fail to generate required results due to imbalanced nature of data. In this paper, we will address the problem of imbalanced datasets whether at the class level, or at the classifier level. In our work, we are interested in binary or multi-class classification. To do this, we present a set of techniques used to solve this problem in particular boosting methods and machine learning algorithms. Our goal is therefore to re-balance the dataset at the class level and to find an optimal classifier to handle these datasets after balancing. Through the results obtained, it was observed that the boosting methods are well suited to re-balance the data and thus give a very satisfactory classification result.

Keywords: Imbalanced Datasets; Supervised Classification; Boosting approach;Data Sampling Approach; SMOTEBoost Algorithm; RUSBoost Algorithm.

1 Introducction

Generally, we can describe the problem of imbalanced data set when the number of instances which represents one class is smaller(minority class) than the ones from other classes(majority classes). The minority classes (the positive class) are usually the most important concepts to be learnt.There are several approaches to learning methods used to overcome the problem of imbalanced data, two such techniques are data sampling(data-level)and boosting[1].

Data sampling [2][3] or called algorithm-level in which the training instances are modified in such a way as to produce a more balanced class distribution that allows the classifiers to perform in a similar manner to standard classification.This approach can be classified into three main groups [4][5]:The first, undersampling methods, the second is Oversampling methods and third, Hybrids methods. Actually, the most common sampling techniques in litterateur

called SMOTE[6] denote to "Synthetic Minority Oversampling TEchnique".The key behind SMOTE is to create synthetic examples. These new examples are created by interpolating between several positive instances that lie together.

Another technique that can be used to improve classification performance is boosting, in principle, combines a set of weak of classifiers that are trained in order to create a better classifier model that makes the boosting classifier more accurate than the original classifier in performing a classification.There are several boosting techniques in literature such as Adaboost, Gradient boost and Xgboost, perhaps the most popular of these is AdaBoost. The AdaBoost algorithm is proposed by Freund and Schapire [7][8] aims to increase the precision of a weak learning algorithm and reduce the bias associated with the prediction learner by encouraging new models to become experts for instances incorrectly handled by the previous model.

Actually, none of these algorithms directly deal the imbalance problem, while the data-level approach attempts to manipulate and focus their attention on difficult examples (mis-classification) without distinguishing which category they belong to. For this reason, we need to modify or combine several techniques to better handle the imbalanced data, to guide the learner not to ignore the minority class, and at the same time try to increase the precision by focusing on the instance mis-classification of the previous model.

Therefore, in this article, we provide a new approach that allows us to combine both SMOTEBoost with RUSBoost using Random Forest. This approach is flexible to handle the hybrid data sampling / amplification algorithms [9][10] and gives a good result. In particular,we would observe the behaviour of using a single model learning(RF), what is the benefit of using this data sampling/boosting and in another hand we try to answer the following questions:can the using of Data sampling /boosting methods suitable and achieve much better performance on unbalanced data sets?

2 Related works

Approach to dealing with the class imbalance is either by modifying the classification algorithm itself (algorithm level) by incorporating the costs of misclassifying different classes into the classification process, or by modifying the data (sampling data) used to train the classifier or using set methods.

Generally, data sampling is based on two keys. The first key is to add instances to the minority class (over-sampling). Whereas, the second key is to remove the majority class instance (under-sampling), or the combination of this two data sampling approach. The second technique also manages imbalanced data.

In fact, the focus of the approach and correcting for the missing classification of the prior learning model is enhanced by adjusting the weight of the samples, increasing the weight of the mis-classification instance and directly decreasing the weight of the correct cases. But combining them will give good results on unbalanced data sets. The greater integration of sampling and data amplification can be described in two of the most popular algorithms called SMOTEBoost and RUSBoost.

In 2003, Chawla et al. [11] suggested the SMOTEBoost algorithm,which based on the integration of the SMOTE algorithm within the standard boosting (adaboost.M2) procedure. The author also mentioned that the proposed SMOTEBoost algorithm can result in better prediction of minority classes without hurting the prediction performance of the majority class. Originally, SMOTEBoost using SMOTE and numerical prediction enhancement methods for classification tasks is mostly applied to regression not straightforward. Nonetheless, SMOTEBoost for regression problem introduced by using four regression boosting methods: AdaBoost RT[12], AdaBoost+[13],AdaBoost.R2[14] and BEMBoost [14].

In 2009, Seiffert et.al [15] introduced RUSBoost algorithm which is performs similarly to SMOTEBoost, but it removes instances from the majority class by random undersampling the data-set in each iteration. The author in his paper mentioned the effective of this proposed methods. In summary, there has been a lot of recent research aimed at the improvement of imbalanced dataset resampling.

3 Realized Work

Standard AdaBoost Algorithm (Freund & Schapire)

Adaboost[15][16]Creates a collection of weak learners by maintaining a set of weights over training samples and adjusting these weights after each weak learning cycle adaptively.Initially, all examples are given the same weight.After each iteration The weights of misclassified instances are increased; on the contrary, the weights of correctly classified instances are decreased.

INPUT:

Training set $S = (x_i, y_i)$
T: number of iterations ,I: base classifier
$D_1(i) = \frac{1}{N}, \forall i \in \{1, ..., N\}$
For i=1 to T:

$$h_t = I(D_t, S) \text{ \# Train weak classifier with weights.} \quad (1)$$

$$E_t = \sum_{i, y \neq h_t(x_i)} D(i) \ \# \text{ Calculate the error.} \tag{2}$$

$$\alpha_t = \frac{1}{2} \ln(\frac{1 - E_t}{E_t}) \ \# \text{ Calculate the weight of the weak classifier.} \tag{3}$$

$$D_{t+1^{(i)}} = D(i). \exp^{(-\alpha . h_t(x_i) . y_i)} \forall i \in \{1, ..., N\} \ \# \text{ Update weights.} \tag{4}$$

$$D_{t+1^{(i)}} = \frac{D_{t+1}(i)}{\sum_{i=1}^{n} D_{i+1}(i)} \ \# \text{ Normalize the weights.} \tag{5}$$

Endfor $H_final = \sin(f(x)) = \sin(\sum_{t=1}^{T}(\alpha_t h_t(x)))$ # Final classifier composed of multiples weak classifiers.

OUTPUT:

H_final

From the equations mentioned above, we see that with each iteration the weights are updated based on the examples that failed with the previously trained classifier, with more emphasis on misclassified instances.

SMOTEBoost(Chawla)

SMOTEBoost,proposed by Chawla et al.[11], is a data-level method to deal with the imbalanced data problem.that combines the Synthetic Minority Oversampling Technique (SMOTE) and the standard boosting procedure.The pseudo code described bellow:

INPUT:

Training set $S = (x_i, y_i)$
T: number of iterations ,I: base classifier
$D_1(i) = \frac{1}{N}, \forall i \in \{1, ..., N\}$
For i=1 to T:

$$S_{syn} = SMOTE(S) \ \# \text{ Generate set of synthetic examples.} \tag{6}$$

$$S^{'} = S \cup S_{syn} \tag{7}$$

$$D_t^{'} = \left\{ d_i \in D_t : (x_i, y_i) \in S^{'} \right\} \ \# \text{ Weight distribution for selected examples.} \tag{8}$$

$$h_t = I(D_t^{'}, S^{'}) \ \# \text{ Train weak classifier with weights.} \tag{9}$$

$$E_t = \sum_{i, y \neq h_t(x_i)} D(i) \ \# \text{ Calculate the error.} \tag{10}$$

$$\alpha_t = \frac{1}{2}\ln(\frac{1-E_t}{E_t}) \text{ \# Calculate the weight of the weak classifier.} \quad (11)$$

$$D_{t+1^{(i)}} = D(i).\exp^{(-\alpha.h_t(x_i).y_i)}\ \forall i \in \{1,...,N\} \text{ \# Update weights.} \quad (12)$$

$$D_{t+1^{(i)}} = \frac{D_{t+1}(i)}{\sum_{i=1}^{n} D_{i+1}(i)} \text{ \# Normalize the weights.} \quad (13)$$

Endfor $H_final = \sin(f(x)) = \sin(\sum_{t=1}^{T}(\alpha_t h_t(x))$ # Final classifier composed of multiples weak classifiers.

OUTPUT:

H_final

RUSBoost

RUSBoost[17],As its name suggests,it integrates within each iteration or boosting a random subsampling(remove instances) of the instances of the majority class.he pseudo code described bellow:

INPUT:

Training set $S = (x_i, y_i)$
T: number of iterations ,I: base classifier
$D_1(i) = \frac{1}{N}, \forall i \in \{1,...,N\}$
For i=1 to T:

$$S\_syn = RUS(S) \text{ \# Random undersampling of class instances majority.} \quad (14)$$

$$S^{'} = S \cup S_{syn} \quad (15)$$

$$D_t^{'} = \left\{ d_i \in D_t : (x_i, y_i) \in S^{'} \right\} \text{ \# Weight distribution for selected examples.} \quad (16)$$

$$h_t = I(D_t^{'}, S^{'}) \text{ \# Train weak classifier with weights.} \quad (17)$$

$$E_t = \sum_{i, y \neq h_t(x_i)} D(i) \text{ \#Calc error as the sum of the weights of the wrong examples classified (original dataset).} \quad (18)$$

$$\alpha_t = \frac{1}{2}\ln(\frac{1-E_t}{E_t}) \text{ \# Calculate the weight of the weak classifier.} \quad (19)$$

$$D_{t+1^{(i)}} = D(i).\exp^{(-\alpha.h_t(x_i).y_i)}\ \forall i \in \{1,...,N\} \text{ \# Update weights.} \quad (20)$$

$$D_{t+1^{(i)}} = \frac{D_{t+1}(i)}{\sum_{i=1}^{n} D_{i+1}(i)} \text{ \# Normalize the weights.} \quad (21)$$

Endfor $H_final = \sin(f(x)) = \sin(\sum_{t=1}^{T}(\alpha_t h_t(x))$ Final classifier composed of multiples weak classifiers.

OUTPUT:

H_final

3.1 Performance Evaluation Metrics

In learning extremely imbalanced data, the classification accuracy is often not an appropriate measure of performance. However, in the case of unbalanced class where the majority class dominates the total population, the results of the classification will achieve high accuracy as it sees only the majority class. The value of metrics, such as recall, precision and AUC [18] which refers to the error under the receiver operating characteristic (ROC) curve, A plot of true positive rate versus false positive rate was used to understand the performance of learning algorithms in minority classes. Based on Table 1, the performance metrics are defined as:

$$Recall := \frac{TP}{TP + FN}. \tag{22}$$

$$Precision := \frac{TP}{TP + FP}. \tag{23}$$

Table 1: Confusion matrix.

	Predicted Positive	Predicted Negative Class
Actual Positive class	TP (True Positive)	FN (False Negative)
Actual Negative class	FP (False Positive)	TN (True Negative)

4 Experimental results

This section presets the experimental results and the effectiveness of the proposed method. We investigate to find the impact of selected hybrid methods such as SMOTEBoost and RUSBoost . We compare also outcomes base classifiers with Data sampling/boosting and single classifiers. All data sets devised into training data and testing data by using stratified 5-fold cross validation in order to ensure that the training and testing data have same proportion of majority and minority class.

4.1 Data sets

The data sets taked from site web KEEL[3] witch have higher imblanced rate (IR)=11.39 and (IR)=6.02 are described briefly in Table 2.

Table 2: Summary Description of the Imbalanced Databases . Features is the Number of Attributes,Instance is the Total Number of Examples Class represent numbers of class

Data sets	Features	Instances	Classes	Majority calss(samples)	Minority calss(samples)	IR
glass2	9	214	binary	92.06%	7.94%	11.59
segment0	19	2308	binary	85.75%	14.25%	6.02

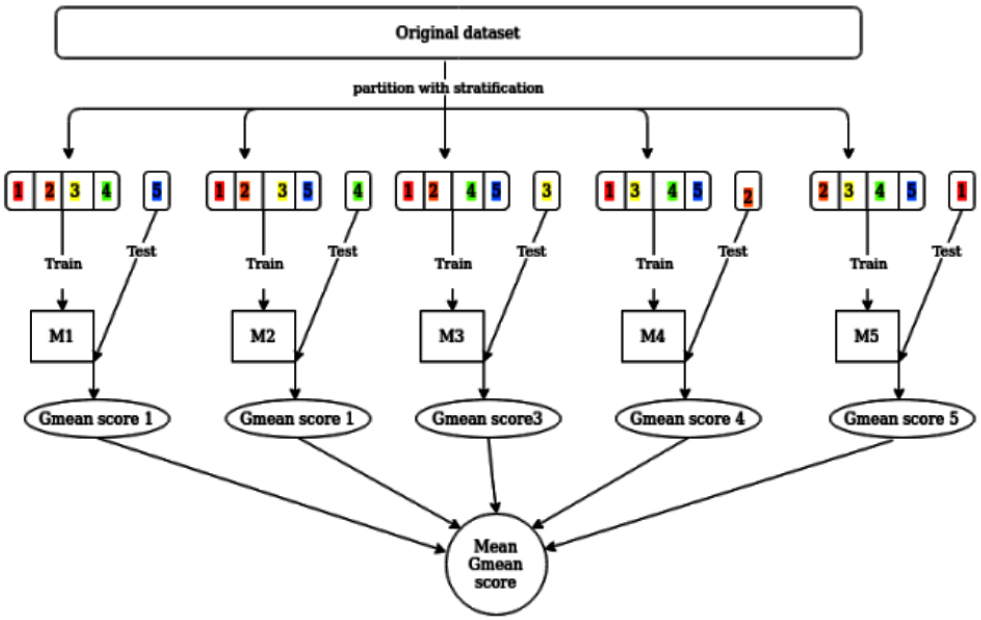

Fig. 1: Show 5-fold cross validation.

[3] https://sci2s.ugr.es/keel/imbalanced.php

4.2 Results and discussions

Experimental result with: IR=11.59 on glass2 datasets for: SMOTE-Boost, RUSBoost and Random-Forests models

Figure 2 shows the learning accuracy of the three models, based on this graph, we can say that RF algorithm archive higher accuracy than other algorithms, it gives very impressive results (0.92060%). After that SMOTEBoost comes in second and RUSBoost has lower accuracy.

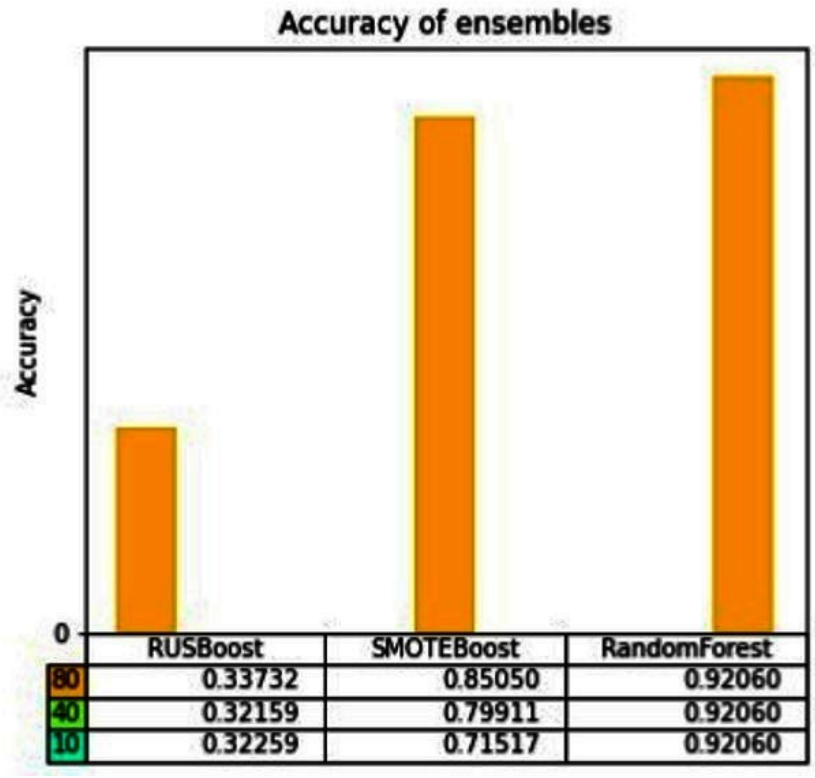

Fig. 2: Accuracy of Three algorithms: RF,SMOTEBoost and RUSBoost.

Figure 3 illustrates both precision and recall. From Figure 3 we can see that Random Forest gives poor precision and recall, it has very low recall (null) and very low precision (null) which means it literally works much worse than random Gaussian (less than $\frac{1}{2}$) and labeling almost 90 of positive classes(minority class) as negative (majority class)and the reason is because of imbalanced datasets. While the RUSBoost algorithm is the best, it gives very impressive results (100%).

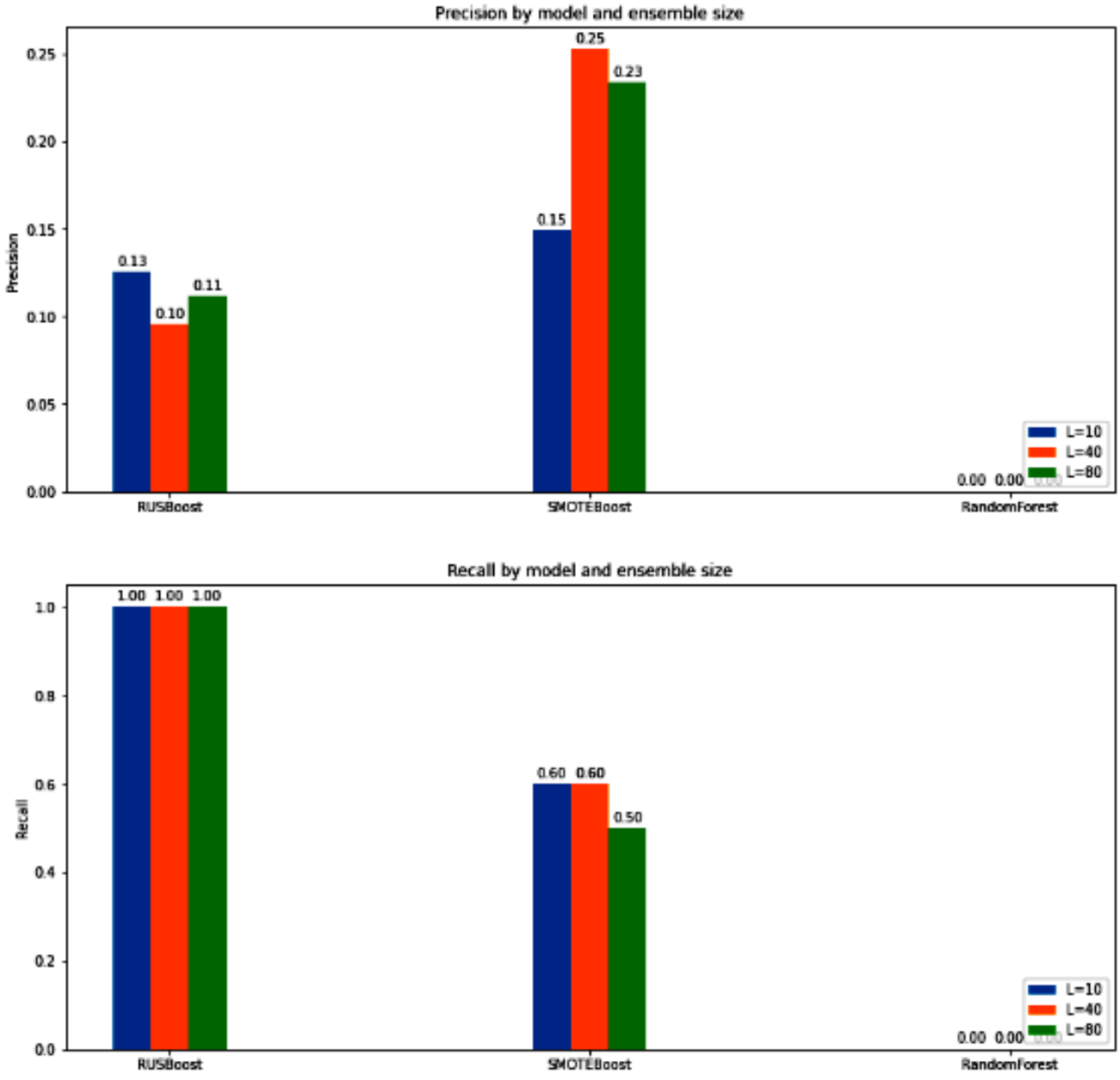

Fig. 3: Precision ans recall of Three algorithms: RF,SMOTEBoost and RUSBoost.

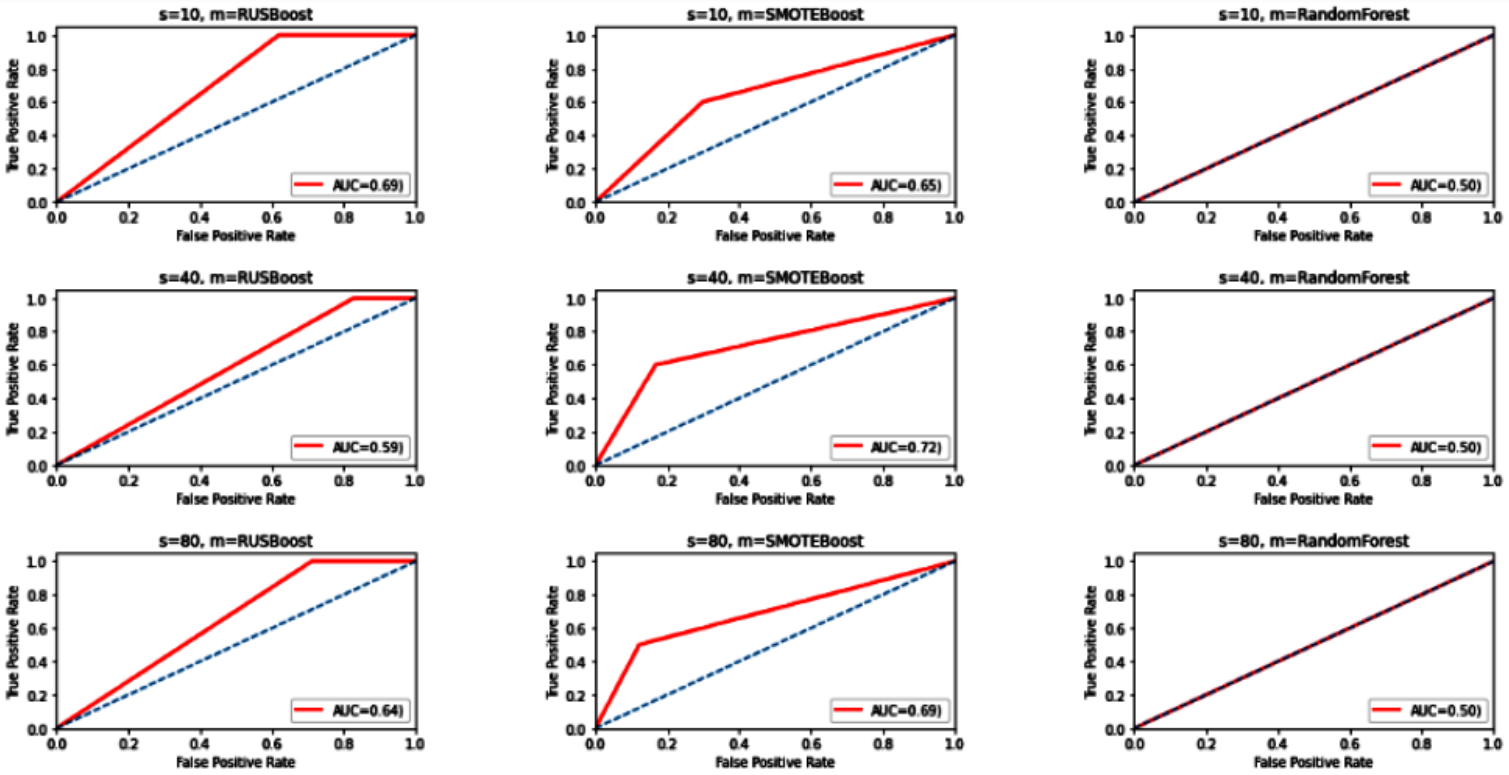

Fig. 4: ROC curve of Three algorithms: RF,SMOTEBoost and RUSBoost.

Figure 4:describe Roc/Auc curve,the higher AUC means model can distinguish between classes better.As we can see in many cases increasing ensemble size decreases AUC.Increasing ensemble size decreases recall so AUC will decrease which means that ROC curve will tend towards y=x.RandomForest has almost same curve as random model y=x.In other hand,we can see clearly the SMOTEBoost and RUSBoost outperformed better than random Gaussian,we think that achieved because in each iteration of boosting techniques there is synthic sample generated(created) ,so that incease the positive class(minority class),thus forced the weak learners to take the minority instanced in in their consideration and not ignore them suck as single random forest done.

Experimental result with: IR=6.02 on segment0 datasets for: SMOTEBoost, RUSBoost and Random-Forests models

In this experimental, we use the number of estimators **T** and the number of synthetic samples of SMOTE algorithm **N**.

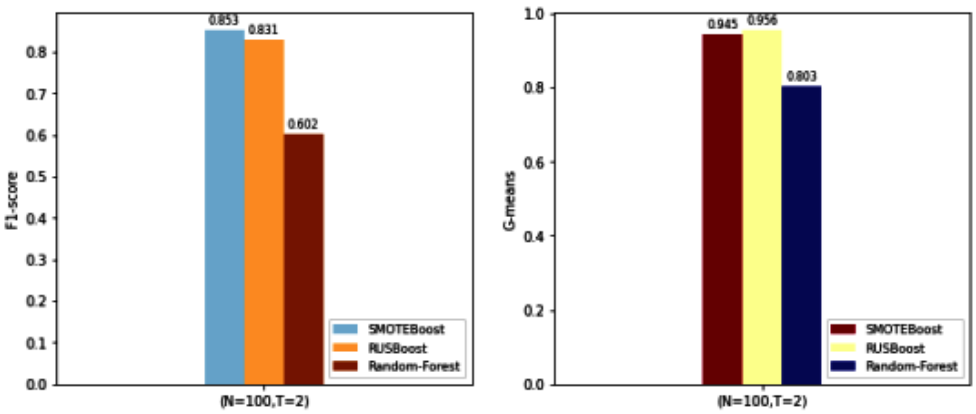

Fig. 5: F1-score and G-means on the segment0 dataset.

Figure 5 shows the classification performance in terms of F-score and G-means that was obtained using random forest with ensemble boosting methods techniques. As indicated by the results, the proposed SMOTEBoost and RUSBoost methods with Random Forest algorithm have shown very satisfactory results. So that, RF-SMOTEBoost has F1-score (0.8534) is better than random Forest alone (without SMOTEBoost, F1-score=0.602), also RF-RUSBoost has better F1-score(0.8306) is better than random Forest alone (without SMOTEBoost, F1-score=0.602). The same thing using G-means where we get: RF-SMOTEBoost (G-means = 0.945) is better than Forest random alone (without SMOTEBoost, G-means = 0.803), RF-RUSBoost a is also better (G-means = 0.956) than Random Forest alone (without SMOTEBoost, G-means = 0.803).

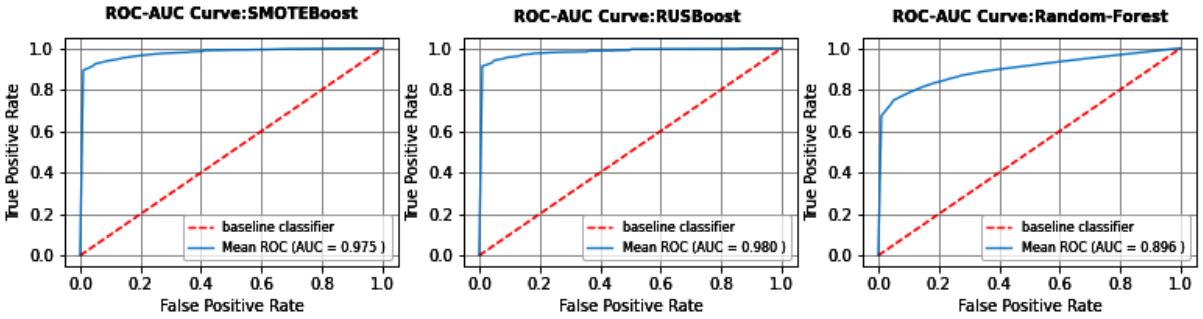

Fig. 6: ROC-AUC Curve on the segment0 dataset.

The Roc/Auc graph above represents the models' ability to differentiate between classes of data sets (between the majority class and the minrity class), the high value of Auc (Area Under the ROC Curve) indicates that models can differ strongly between classes. Our curve clearly shows that the auc value of RF-SMOTEBoost and RF-RUSBoost is higher that the Random Forest alone (effectiveness of boosting methods in imbalanced datasets classification).

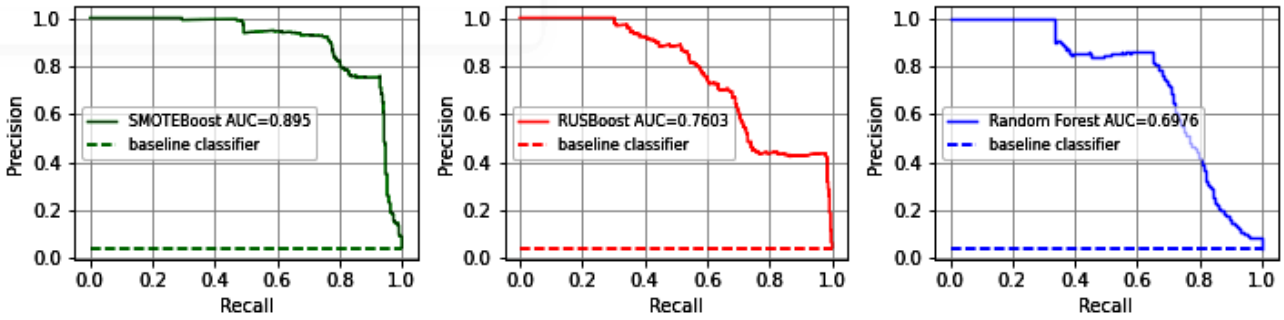

Fig. 7: Precision-Recall Curve on the segment0 dataset.

From Figure 7, we notice that the performance of SMOTEBoost and RUSboost gives better classification results (Precision-Recall) than the Random Forest classifier alone. Based on experimental results, we have found that proposed methods performed favourably.

5 conclusions

In this paper, we have carried out an overview on the topic of imbalanced classification in the scenario boosting-based ensembles methods which is hybrid data sampling with boosting methods.These combination methods has been a very successful technique for solvings classification problem.We analysis the performance of some of learning algorithms decision tree stumps,random forest with these hybrid methods based on imbalanced data classification problem and introduce Random Forest,SMOTEBoost and RUSBoost algorithm. from the results of this study.the main conclusions can be derived as follows:

- SMOTEBoost is combination of two different approach data sampling(oversampling) and boosting algorithm .The experimental proved the successful of this methods in imblanced data.

- RUSBoost is hybrid methods witch combine data sampling (under-sampling) and standard boosting methods(mor specific boost.M2).

References

1. Weiss, G. M. (2004). Mining with rarity: a unifying framework. ACM Sigkdd Explorations Newsletter, 6(1), 7-19.
2. He, H., Garcia, E. A. (2009). Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21(9), 1263-1284.
3. Sun, Y., Wong, A. K., Kamel, M. S. (2009). Classification of imbalanced data: A review. International journal of pattern recognition and artificial intelligence, 23(04), 687-719.
4. Lopez, V., Fernandez, A., Garcıa, S., Palade, V., Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information sciences, 250, 113-141.
5. Batista, E., Prati, R. C., Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD explorations newsletter, 6(1), 20-29.
6. Chawla, N. V., Bowyer, K. W., Hall, L. O., Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research,16, 321-357.
7. Freund, Y., Schapire, R. E. (1996). Experiments with a new boosting algorithm. In icml (Vol. 96, pp. 148-156).
8. Freund, Y., Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1), 119-139.
9. Amami, Rimah, Dorra Ben Ayed, and Noureddine Ellouze (2013). "The challenges of SVM optimization using Adaboost on a phoneme recognition problem." 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom). IEEE, 2013.
10. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

11. Chawla, N. V., Lazarevic, A., Hall, L. O., Bowyer, K. W. (2003). SMOTEBoost: Improving prediction of the minority class in boosting. In European conference on principles of data mining and knowledge discovery (pp. 107-119). Springer, Berlin, Heidelberg.
12. Solomatine, D. P., Shrestha, D. L. (2004). AdaBoost. RT: a boosting algorithm for regression problems. In 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541) (Vol. 2, pp. 1163-1168). IEEE.
13. Kankanala, P., Das, S., Pahwa, A. (2013). AdaBoost +: An Ensemble Learning Approach for Estimating Weather-Related Outages in Distribution Systems. IEEE Transactions on Power Systems, 29(1), 359-367.
14. Drucker, H. (1997). Improving regressors using boosting techniques. In ICML (Vol. 97, pp. 107-115).
15. Wang, Xiaodan, (2006). "Improved algorithm for adaboost with SVM base classifiers." 2006 5th IEEE International Conference on Cognitive Informatics. Vol. 2. IEEE, 2006.
16. Mishina, Y., Murata, R., Yamauchi, Y., Yamashita, T., Fujiyoshi, H. (2015). Boosted random forest. IEICE TRANSACTIONS on Information and Systems, 98(9), 1630-1636.
17. Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., Napolitano, A. (2009). RUSBoost: A hybrid approach to alleviating class imbalance. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 40(1), 185-197.
18. Jones, Robbie, and David Lin (2016). "Stack overflow query outcome prediction.