JOURNAL OF FUNDAMENTAL MATHEMATICS
AND APPLICATIONS (JFMA) VOL. 3 NO. 1 (JUN 2020)
Available online at www.jfma.math.fsm.undip.ac.id

EASY ENSEMBLE WITH RANDOM FOREST TO HANDLE IMBALANCED DATA IN CLASSIFICATION

S Abdullah[1*], G V Prasetyo[2]

[1,2]Department of Mathematics, FMIPA Universitas Indonesia, Depok 16424, Indonesia
[*]Corresponding author: sarini@sci.ui.ac.id

Abstract. Imbalanced data might cause some issues in problem definition level, algorithm level, and data level. Some of the methods have been developed to overcome this issue, one of state-of-the-art method is Easy Ensemble. Easy Ensemble was claimed can improve model performance to classify minority class, and overcome the deficiency of random under-sampling. In this paper we discussed the implementation of Easy Ensemble with Random Forest Classifiers to handle imbalance problem in credit scoring case. This combination method is implemented in two datasets which taken from data science competition website, finhacks.id and kaggle.com with class proportion within majority and minority is 70:30 and 94:6. The results showed that resampling with Easy Ensemble can improve Random Forest classifier performance upon minority class. Recall on minority class increased significantly after the resampling. Before resampling, the recall on minority class for the first dataset (finhacks.id) was 0.49, and increased to 0.82 after the resampling. Similar results were obtained for the second data set (kaggle.com), where the recall for the minority class was increased from just 0.14 to 0.73.

Keywords: Bagging, credit scoring, minority class, oversampling, undersampling, recall.

.

I. INTRODUCTION

In a real-world problem, cases with imbalanced data are common; for example, in medical case which classify breast cancer type [1], cervical cancer [2], and lung cancer [3]. In financial case, imbalanced data problems are also found, such as credit scoring classification [4] and fraud detection [5]. Imbalanced data may cause problem in building a model, output of the classification model tends to predict majority class.

The unfavorable effect of imbalanced data is also confirmed by our simulation on assessing the accuracy of classification models based on various class proportions for minority and majority classes, as shown in Figure 1. We generated hypothetical data with binary target variable (say, class 1 and class 2). We started off with a balance data, that is a 50:50 proportion of class 1 and class 2. Then, we changed the proportion by reducing the proportion of class 1 by 5 per cent and increasing the proportion of class 2 by the same amount, one at a time. The last data generated was that with the heavily imbalanced proportion between the two classes, that is at 5: 95.

It is clearly seen from the graph that the as the difference in proportion of minority and majority classes increases, the models' ability to identify minority class were dramatically reduced. Although the accuracy seemed not much affected by the classes imbalance, we cannot rely on the accuracy itself, as it results merely due to the contribution of high proportion of the majority class. The model itself failed to differentiate between the two classes, and in the

case of heavily imbalanced data, all data were classified in the majority class, as indicated by 0 recall on minority class and the accuracy is just equal to the precision on the majority class.

This numerical experiment showed the latent danger in having imbalanced data, that although the model's accuracy is high, we have to be cautious that the accuracy might be misleading and thus cannot be used for the inference purpose. Imbalance data problem should be handled prior to proceeding to the classification phase. Therefore, a method for handling imbalanced data is sought.

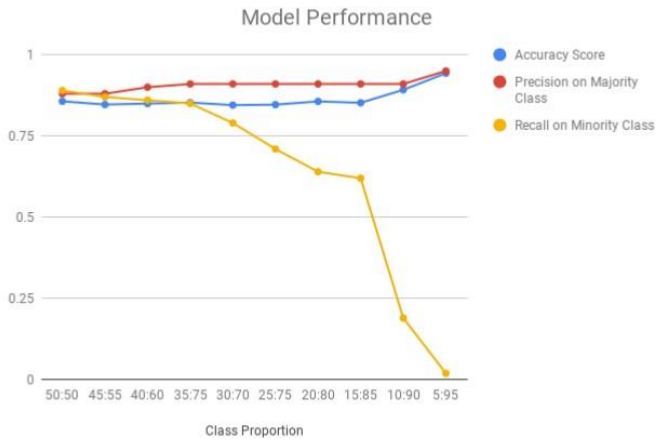

Figure 1. Performance of classification model for various proportions of majority and minority classes.

The imbalance data problem needs to be overcome in order to have a better classifier. In addition, the model can predict minority class much better when the imbalance problem is handled. Some of the methods have been developed to handle imbalanced problem, one of them is Easy Ensemble [6]. Easy Ensemble was claimed can improve model performance to classify minority class moreover can overcome the deficiency of other conventional imbalance learning such random under-sampling.

Random forest could be considered as one of the best classifiers in machine learning, as implied by the results in recent studies [7]. It outperformed the Super Vector Machines, produced an accuracy of 93.4% in detecting the stroke lesions based on MRI of acute ischemic stroke [8] as well as in classification of alcohol use disorder [9]. In addition to its high classification accuracy, random forest is also considered as variable selection tool, which improves the performance of the predicting model [10,11,12]. This approach might be motivated due to the robustness of the result of random forest, where the selected important variables should come as a result of their consistency in the splitting rule when they were chosen in the random feature selection in generating a tree for each new bootstrap data. Considering these studies, we propose the use of random forest for classification in this study.

We also propose the Easy Ensemble method as an imbalance learning to handle imbalance problem in classification with Random Forest as a classifier. To show that Easy Ensemble with Random Forest can overcome imbalanced problem, we implemented the methods for churn classification using two credit scoring datasets. First dataset is taken from data science competition website *kaggle.com,* and the second is taken from another data science competition website *finhacks.id*

II. RESEARCH METHODOLOGY

II.1 Imbalance Learning as a Solution of Imbalance Problem

This section will explain some solutions to handle imbalance problem or can simply called as imbalance learning. The description in mainly refer to imbalance learning article [13].

Resampling Method

Resampling is currently mainstream method to handle imbalance problem in classification. This method is proposed to re-balance the class proportions between majority and minority, so that the model can have a better decision boundary.

Random Under-Sampling

Under-sampling is a basic technique in resampling method. This technique works by reducing the majority class until it has the same proportion with the minority class. Mathematically, it can be written with the following equation:

$$|\boldsymbol{S}'_{\boldsymbol{maj}}| \leftarrow |\boldsymbol{S}_{\boldsymbol{maj}}| - |\boldsymbol{E}| \quad (1)$$

The above formulation means that from the majority class $\boldsymbol{S}_{\boldsymbol{maj}}$, a subset data $\boldsymbol{E}$ is withdrawn randomly, and the remaining data become the new majority class, denoted as $\boldsymbol{S}'_{\boldsymbol{maj}}$. As the consequence, the size of the majority class is reduced. $|A|$ denotes the size, or cardinality, of A.

While the size of majority class is reduced, the minority class, $\boldsymbol{S}_{\boldsymbol{min}}$, is not changed. The new resulting dataset, $\boldsymbol{S}'$, is formed by the reduced majority class $\boldsymbol{S}'_{\boldsymbol{maj}}$ and the minority class $\boldsymbol{S}_{\boldsymbol{min}}$, written as

$$|\boldsymbol{S}'| \leftarrow |\boldsymbol{S}_{\boldsymbol{min}}| \cup |\boldsymbol{S}'_{\boldsymbol{maj}}| \quad (2)$$

Random Over-Sampling

Similar with under sampling, over-sampling is another basic technique in resampling method. This technique works by duplicating the minority class until it has the same proportion with the majority class. Mathematically, it can be written with the following equation:

$$|\boldsymbol{S}'_{\boldsymbol{min}}| \leftarrow |\boldsymbol{S}_{\boldsymbol{min}}| + |\boldsymbol{E}| \quad (3)$$

Equation (3) states that the new minority class, $\boldsymbol{S}'_{\boldsymbol{min}}$ is formed by adding a new dataset, that is $\boldsymbol{E}$, to the initial minority class $\boldsymbol{S}_{\boldsymbol{min}}$. This set $\boldsymbol{E}$ is obtained by random duplication of data in $\boldsymbol{S}_{(\boldsymbol{min})}$.

While the size of the minority class is increased, the majority class, $\boldsymbol{S}_{\boldsymbol{maj}}$, is not changed. Putting together the new minority class, $\boldsymbol{S}'_{\boldsymbol{min}}$, and the majority class, $\boldsymbol{S}_{\boldsymbol{maj}}$, results in the new and more balanced dataset $\boldsymbol{S}'$, written as

$$|\boldsymbol{S}'| \leftarrow |\boldsymbol{S}_{\boldsymbol{maj}}| \cup |\boldsymbol{S}'_{\mathrm{min}}| \quad (4)$$

II.2 Easy Ensemble

Let S be the dataset to be classified, partitioned into training set and testing set. For the training set, let P be the set with minority class and N be set with majority class. Using the undersampling method, choose $n_0 = |P|$ be the size of a subset N_0 of N such that $n_0 < |N|$. Elements of N_0 is randomly sampled from N.

By taking only one subset of the majority class, there is a high possibility to ignore information from other data that are not selected. Therefore, in easy ensemble procedure, the above process is repeated, say T times. Thus, T sample of size n_0 are drawn independently from N, namely $N_1, N_2, \dots, N_T$. From each of this subset $N_j, j = 1, \dots, T$, pool it with the minority subset P to form training data, for which a classifier H_j is trained. The pseudo-code for Easy Ensemble is shown in Algorithm 1.

Algorithm 1. Easy Ensemble

Input: Data of size S, consisting a set of minority class P and a set of majority class N, $|P| < |N|$. t is the number of subsets of N, k_j is the number of iterations in training model ensemble random forest H_j.

Output: ensemble Random Forest model, $H_j(x) = \{h^b\}_1^B$

$j \leftarrow 0$

repeat

Form subset N_j of N, using random sampling such that $|N_j| = |P|$ and the union of those two classes is S_j, that is the subset of class S consisting of two sets of classes with a balanced-proportion.

Train H_j model with its subset N_j and P that already balanced in size.

H_j is the random forest model.

until $j = T$

Easy Ensemble produces an output as a single ensemble. However, since random forest can be considered as an ensemble method itself -i.e. relative to decision trees- then we do ensemble the random forests, this procedure seems as *'ensemble of ensembles'* [14].

II.3 Random Forest

Random forest is an ensemble method, formed by a number of trees. Instead of making decision with one tree, random forest's decision produced by the majority of votes (for classification cases) in trees. Since each of tree works independently, that is by bagging procedure in selecting dataset to build each tree, as well as the implementation of random feature selection for each split in each tree, it is reasonable to assume that the results from random forest are more robust and consistent than that from decision tree. This is also guaranteed by the convergence property of the generalization error [15]. Algorithm 2 explain the detail about Random Forest combining with Easy Ensemble.

Algorithm 2. Random Forest algorithm with Easy Ensemble

Input: Data of size S, consisting a set of minority class P and a set of majority class N, $|P| < |N|$. t is the number of subsets of N, k_j is the number of iterations in training model ensemble random forest H_j.

Output: ensemble Random Forest model,
1
$j \leftarrow 0$
for $b = 1,2,3, \dots, B$

- Do bootstrapping, by resampling with replacement, of size n from n observations on training data.
- Form decision tree h^b from each of the bootstrap sets, each node of the tree is randomly chosen.

- Choose p variables producing the highest homogeneity, based on
  $\hat{p} = A_j | Gini(S^b, A_j) = \min_{l \in 1,\dots,m} Gini(S^b, A_j)$ or
  $\hat{p} = A_j | Entropy(S^b, A_j) = \min_{l \in 1,\dots,m} Entropy(S^b, A_j)$
  if Gini index (or entropy, respectively) is used as the homogeneity criterion.
- Node splitting rule is as in decision tree algorithm.
- Form each tree without pruning.

End

III. EXPERIMENTS AND RESULTS

III.1 Simulation

For the simulation, we conducted the following procedures (depicted in Figure 2). First, we split data five times randomly into two parts, training set and testing set. Training set is the data that use build model/classifier, on the other hand, testing is a dataset that use to evaluate performance of the model. Proportion of the training and testing is 70:30. After splitting the data, second step is setting the hyperparameter value which will use to build the model. Every model has its own hyperparameter but in this experiment we will use some of them: number of trees, type of criterion, and number of possibility predictor variable.

III.2 Random Forest Performance Before Resampling

In this section we implement Random Forest classifier to both datasets (*kaggle.com* and *finhacks.id*). We run the model with the K-Fold Cross Validation (K = 5) to have an unbiased result. Table 1 show the detail result for both datasets. As we can see, in the first dataset, they only have 0.47 on recall minority. This means, Random Forest can only predict non-default class correctly about 47%, the rest of the class (53%) cannot properly classified by the model. From this result, we can say that imbalance problem can lead model to have huge misclassification in predict minority class.

In the second dataset, they only have 0.14 on recall minority. This means, Random Forest can only predict non-default class correctly about 14% and the rest of the class (86%) cannot properly classified by the model. This was obviously caused by the imbalance condition on the data and shown that imbalance problem lead model to misclassification.

Table 1. Model performance before resampling

Data	Accuracy	Precision Majority	Recall Minority
Data 1 (*finhacks.id*)	0.79	0.84	0.47
Data 2 (*kaggle.com*)	0.93	0.94	0.14

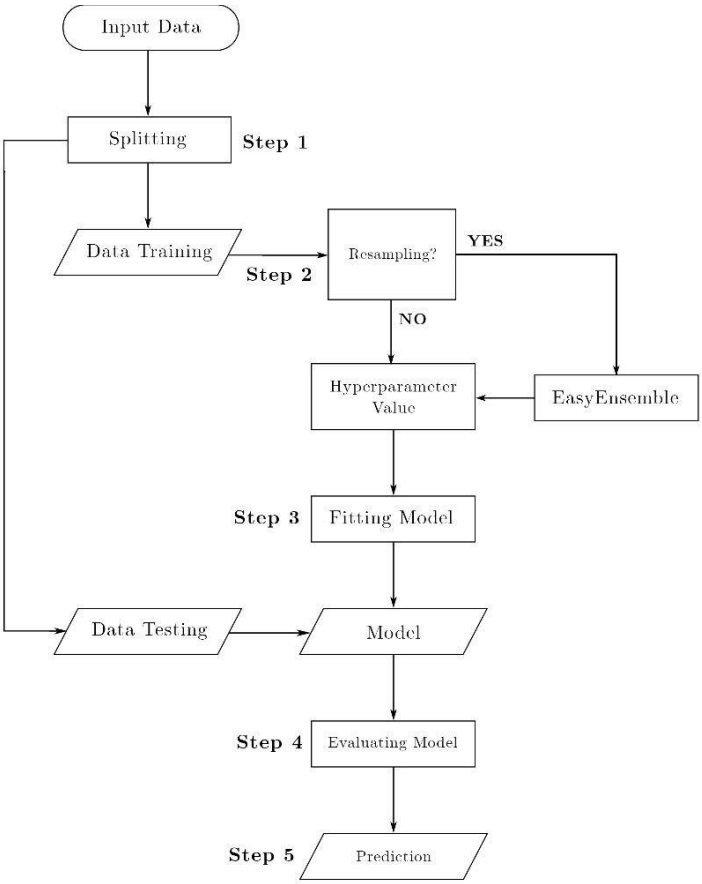

Figure 2. Simulation flow in credit scoring case

III.3 Random Forest Performance After Resampling

Similar with the simulation on the first and second data before resampling, we try to implement Random Forest on the data after resampling with Easy Ensemble. The following step is also same as before. We randomly split the data into training set and testing set five times, and then run the model with K-Fold Cross Validation, for K = 5. Table 2 shows the detail results of the model's performance after resampling.

For the first dataset, there are significant improvements in both majority precision and minority recall. The precision is improved by 8 per cent (for model with number of subsets of 10 and 15), while the recall is improved five times than that, that is increased by 35 per cent (for model with 15 subsets). It is also seen that the greater number of subsets results in a higher precision and recall. There is a reduction in overall accuracy, between 7 and 9 per cent. Yet, the accuracy is still high. Moreover, as explained in the background, precision and recall are more reliable and produce more insights than the overall accuracy, so the reduction in accuracy does not imply the reduction in model's performance. Instead, the model's performance is improved as shown by improvement in the precision and recall.

As for the second dataset, the model's improvement is much higher than the first dataset, particularly for the minority recall. The recall is drastically increased from 0.14 (for data before resampling) to 0.71 - 0.73, improving the recall by 57 percent (for data with 5 subsets) to 59 per cent (for data with 15 subsets). Similar with results in the first dataset, the accuracy is reduced, yet this reduction is just considered as a consequence of the change in size of minority

and majority classes, it does not necessarily imply that the reduction of the model's performance. On the contrary, the model's performance is significantly increased, as the model can differentiate the majority and minority classes far better than the model in the non-resampling scheme, indicated by the smaller gap between majority precision and minority recall.

Table 2. Model performance after resampling. The values in the brackets are the change (increment for positive sign, decrement for negative sign) in the corresponding quantity compared to the model performance before resampling.

Data	**Number of Subsets**	**Accuracy**	**Majority Precision**	**Minority Recall**
Data 1 (*finhacks.id*)	5	0.70 (-.09)	0.91 (+ 0. 07)	0.79 (+ .32)
	10	0.71 (- 0. 08)	0.92(+0.08)	0.81 (+0.34)
	15	0.72 (- 0. 07)	0.92 (+0. 08)	0.82 (+0.35)
Data 2 (*kaggle.com*	5	0.78 (- 0. 15)	0.97 (+ 0. 03)	0.71 (+ 0. 57)
	10	0.78 (- 0. 15)	0.98 (+ 0. 04)	0.72 (+ 0. 58)
	15	0. 78 (- 0. 15)	0.98 (+ 0. 04)	0.73 (+ 0. 59)

Our results confirm the result of previous study [16] that showcased the excel of easy ensemble to handle imbalanced data. Moreover, a study by Zhang et. al. [17] also showed the capability of easy ensemble method for binary and multi-class classification models. However, those studies focused on improving and analyzing the models' accuracy. While, as we showed in earlier in the background section that high accuracy might be misleading when there is a large gap between the recall and precision, and thus, we more focus on narrowing that gap.

IV. CONCLUSION

Easy Ensemble with Random Forest works in a kind of way; separate data into subsets with same proportion within minority and majority by undersampling method. The more subsets are created, the better the classifier would be. These combined methods also have better results in handling imbalanced data problem. Based on experiments on two credit scoring data, we obtained that easy ensemble improves the models' performances significantly. The best improvement for data from *finhacks.id* is 8 and 35 per cents increment in the majority precision and minority recall, respectively. While, for data from *kaggle.com*, the best improvement is 4 per cent and 59 per cent increase in the majority precision and minority recall, respectively.

REFERENCES

[1] Z. H. Zhou and Y. Jiang, "*Medical diagnosis with C4.5 rule preceded by artificial neural network ensemble*", IEEE Transactions on Information Technology in Biomedicine,vol. 7, no. 1, pp. 37–42, 2003.

[2] Jeanne S. Mandelblatt, Karen Gold, Ann S. O'Malley, Kathryn Taylor, Kathleen Cagney, John S. Hopkins, Jon Kerner, "Breast and Cervix Cancer Screening among Multiethnic Women: Role of Age, Health, and Source of Care", Elsevier, (1999)

[3] Sakuma,Yuji et al., "A logistic regression predictive model and the outcome of patients with resected lung adeno carcinoma, Lung Cancer, Volume 65, Issue 1, 85 – 90, (2009)

[4] Bravo, C., Maldonado, S., Weber, R., Granting and managing loans for microentrepreneurs: new developments and practical experiences. Eur. J. Oper. Res. 227 (2), (2013)

[5] E.W.T. Ngai, Yong Hu, Y.H. Wong, Yijun Chen, Xin Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature", Decision Support System, Elsevier, (2011)

[6] Tian-Yu Liu, "EasyEnsemble and Feature Selection for Imbalance Data Sets", International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, pp. 517 – 518, (2009)

[7] Qiu, Xueheng, et al. "Oblique random forest ensemble via Least Square Estimation for time series forecasting." *Information Sciences* 420 (2017): 249-262.

[8] Subudhi, Asit, Manasa Dash, and Sukanta Sabut. "Automated segmentation and classification of brain stroke using expectation-maximization and random forest classifier." *Biocybernetics and Biomedical Engineering* 40.1 (2020): 277-289.

[9] Kamarajan, Chella, et al. "Random Forest Classification of Alcohol Use Disorder Using fMRI Functional Connectivity, Neuropsychological Functioning, and Impulsivity Measures." *Brain Sciences* 10.2 (2020): 115.

[10] Izquierdo-Verdiguier, Emma, and Raúl Zurita-Milla. "An evaluation of Guided Regularized Random Forest for classification and regression tasks in remote sensing." *International Journal of Applied Earth Observation and Geoinformation* 88 (2020): 102051.

[11] Dumitrescu, Elena, et al. "Machine Learning or Econometrics for Credit Scoring: Let's Get the Best of Both Worlds." (2020).

[12] Van Sang, Ha, Nguyen Ha Nam, and Nguyen Duc Nhan. "A novel credit scoring prediction model based on Feature Selection approach and parallel random forest." *Indian Journal of Science and Technology* 9.20 (2016).

[13] H. Heibe, A. G. Edwardo, "Learning from Imbalanced Data", Transactions on Knowledge and Data Engineering, IEEE, (2009)

[14] Y. Liu, A. An, and X. Huang, "*Boosting prediction accuracy on imbalanced datasets with SVM ensembles*", Lecture Notes in Artificial Intelligence, vol. 3918, pp. 107–118, (2006).

[15] L. Breiman, "Random Forest", Department of Statistics, UC Berkeley, Machine Learning, 45, 5–32, (2001).

[16] Galar, Mikel, et al. "Ordering-based pruning for improving the performance of ensembles of classifiers in the framework of imbalanced datasets." *Information Sciences* 354 (2016): 178-196.

[17] Zhang, Zhongliang, et al. "Empowering one-vs-one decomposition with ensemble learning for multi-class imbalanced data." *Knowledge-Based Systems* 106 (2016): 251-263.