



Hybrid approach redefinition with progressive boosting for class imbalance problem



Hartono ^{a,b,1,*}, Erianto Ongko ^{c,2}

^a Department of Computer Science, STMIK IBBI, Medan, Indonesia

^b Department of Computer Science, Universitas Potensi Utama, Medan, Indonesia

^c Department of Informatics, Akademi Teknologi Industri Immanuel, Medan, Indonesia

¹ hartonoibbi@gmail.com; ² eriantoongko@gmail.com

* corresponding author

ARTICLE INFO

Article history

Received June 13, 2019

Revised June 14, 2019

Accepted November 29, 2019

Keywords

Class Imbalance

Classification

Hybrid Approach Redefinition

Hybrid Ensembles

Progressive Boosting

ABSTRACT

Problems of Class Imbalance in data classification have received attention from many researchers. It is because the imbalance class will affect the accuracy of the classification results. The problem of the imbalance class itself will ignore the minority class, which has a smaller number of instances even though this is an exciting class to observe. In resolving imbalanced class problems, it is necessary to pay attention to diversity data, the number of classifiers, and also classification performance. Several methods have been proposed to overcome the imbalanced class problem, one of which is the Hybrid Approach Redefinition Method. This method is a good hybrid ensemble method in dealing with imbalance class problems, which can provide useful diversity data and also a smaller number of classifiers. This research will combine the Hybrid Approach Redefinition by replacing the use of SMOTE Boost by using Progressive Boosting to get better data diversity, a small number of classifiers, and better performance. This study will conduct testing in handling imbalance class problems using datasets sourced from the KEEL-Dataset Repository. This study concludes that the Hybrid Approach Redefinition with Progressive Boosting will provide better results in the number of classifiers, data diversity, and classification performance.

This is an open access article under the [CC-BY-SA](#) license.



1. Introduction

The imbalance class problem in the classification process gets increasingly widespread attention, not only because it can affect the accuracy but also because it ignores important information contained in the minority class [1]. The problem of imbalance class is often a problem of binary classification problem where the sample of one class, especially the class minority, will be compared with all samples of all existing classes [2], [3]. The class imbalance problem became a new research topic of machine learning with the theme "Learning from Imbalanced Data", which began to develop in the 2000s along with the first workshop on the imbalance class organized by the American Association for Artificial Intelligence Conference [4]. Several methods of solving class imbalance explained that there were 2 (two) things which are

the main issues, namely: regarding the number of classifiers and data diversity [5]. Then Luque et al. have proposed classification performance as one aspect to measure success in handling imbalance class problems [1]. Lachiche and Flach [6] and Yang et al. [7] use the Received Operating Characteristic (ROC) Curve to evaluate the classifier. Wang and Yao [5] and Sun et al. [8] using the F-Measure, G-Means, and Q-Statistic methods for determining diversity data. Galar et al. [2] suggested a taxonomy consisting of 4 (four) groups of approaches in the use of ensemble learning methods, namely: cost-sensitive boosting, boosting-based ensembles, bagging-based ensembles, and hybrid ensembles. Hybrid Ensembles is an ensemble learning method that combines Bagging and Boosting methods. Jian et al. [9] suggested a new ensemble learning method called Different Contribution Sampling (DCS), which can be said as a sampling-based and Boosting hybrid ensemble method. Ren et al. [10] suggested an ensemble learning method based on sampling by presenting the Ensemble Based Adaptive Over-Sampling method that modifies the Over-Sampling method by using Adaptive SMOTEBoost in overcoming the problem of imbalanced class. Hybrid Approach Redefinition (HAR) Method is one method of hybrid ensembles. In this method, the preprocessing stage will be done using the Random Under Sampling and SMOTEBoost (Random Balance Ensemble Method) method. The processing stages will be carried out using UnderBagging and Different Contribution Sampling [11]. Research conducted by Fernandez et al. showed that smote boost has a weakness if it is faced with small disjuncts, lack of data, and noise which indirectly can affect low diversity data because it allows misclassification only in one class [12]. According to Díez-Pastor et al. [13] is important to pay attention to the diversity of data in handling imbalance classes. It means that attempted misclassification produced by each classifier is as small as possible, and if there is misclassification, it is expected to occur on different objects or parts [14]. Progressive Boosting (PBoost) is a proposal that progressively groups samples that are not correlated to the Boost procedure. It is intended that information is not lost and can produce a collection of various classifications. Based on this, the PBoost method is expected to improve diversity data [15]. This research will combine the Hybrid Approach Redefinition by replacing the use of SMOTE Boost by using Progressive Boosting to get better data diversity, a small number of classifiers, and better performance.

2. Method

In this research, there are three stages, namely preprocessing, processing, and evaluation. The datasets in this study are the KEEL-Dataset Repository by considering the imbalance ratio. The datasets of a low imbalance ratio will use the Pima dataset, while the datasets with a moderate imbalance ratio use the Abalone9vs18 dataset. Besides, the dataset of the high imbalance ratio uses the Yeast2vs8 dataset. All datasets consider the number of attributes and instances [16].

2.1. Preprocessing Stage

In the Hybrid Approach Redefinition with Progressive Boosting method, the preprocessing stage will be carried out using the Random Under Sampling method and also Progressive Boosting. The results of the preprocessing stage are in the form of a preprocessing dataset, which will then proceed to the processing stage. The preprocessing stages can be seen in Fig. 1.

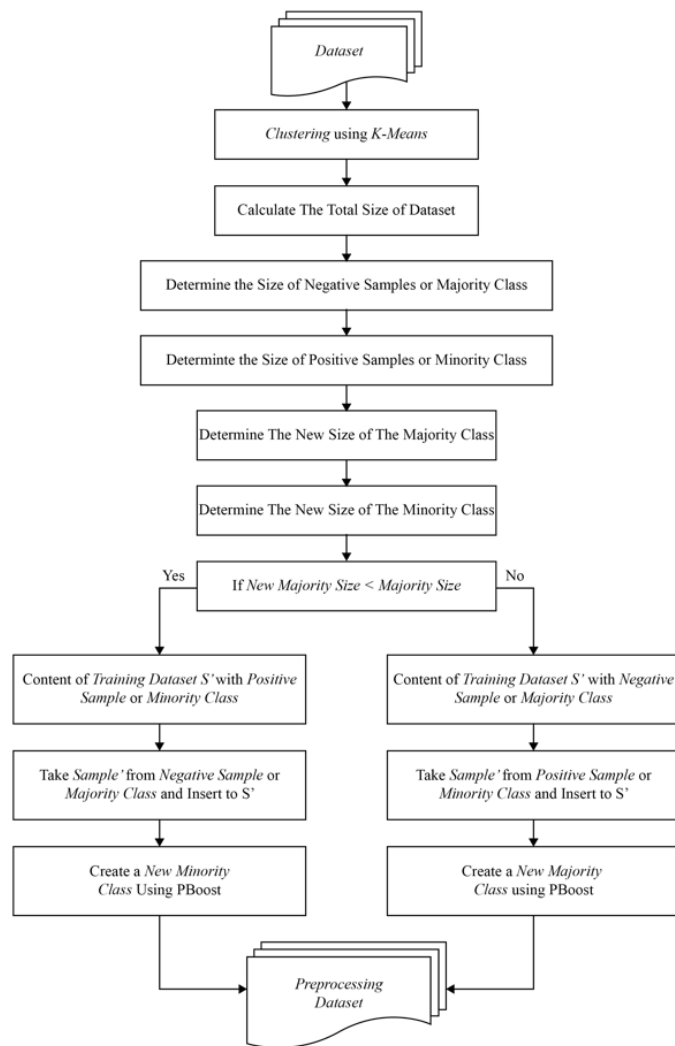


Fig. 1. Preprocessing Stage at Hybrid Approach Redefinition with Progressing Boosting

At the preprocessing stage, for example, clustering is done using K-Means. As a result, there are times when imbalance class problems occur. To overcome this, we need to handle the imbalance class, which starts from the preprocessing stage. At this stage, for the first time on the issue of the imbalance binary class, the majority and minority class will be determined. After that, a random number generation process is performed to determine the size of the new majority class. If the results show that the size of the new majority class is smaller than the old majority size, then this means that the new majority class still shows the difference in the number of large instances with minority class, so this needs to be addressed by moving several majority class instances to minority class through the PBoost process.

Conversely, if the new majority size is larger than the old majority size, then this indicates that there is a reverse where several instances in the minority class must be moved to majority classes through the PBoost process. This process is widespread because sometimes handling the imbalance class can cause the class that was originally a minority class to have a tremendous instance, so it needs to be considered about this. In Fig. 1, it can be seen that the main difference with the Hybrid Approach Redefinition classic is that the use of the SMOTEBoost method is replaced by using Progressive Boosting (PBoos).

2.2. Processing Stage

The processing stages will be carried out using UnderBagging and Different Contribution Sampling. The processing stages can be seen in Fig. 2.

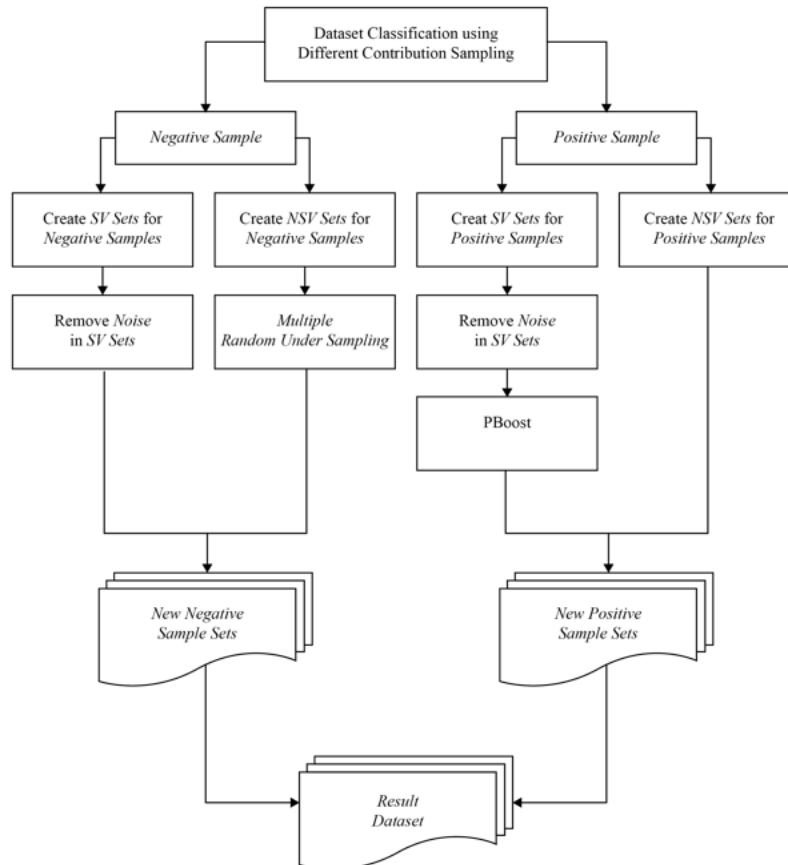


Fig. 2. Processing Stage at Hybrid Approach Redefinition with Progressing Boosting

At this stage, the dataset that has undergone the preprocessing stage will undergo a further process at the processing stage using Different Contribution Sampling, which begins by classifying both the Majority Class and Minority Class into SV Sets and NSV Sets. SV Sets and NSV Sets for majority classes and minority classes will experience different treatment processes. In Minority Class, SV Sets will experience a noise removal process and will then undergo PBoost stages while NSV Sets will be combined with SV Sets PBoost results to New Positive Sample. Whereas in the Majority Class, NSV Sets will undergo the sampling stage with the RUS method and will be combined with SV Sets, which has been eliminated by noise to New Negative Sample. In Fig. 2, it can be seen that the main difference with the Hybrid Approach Redefinition classic is that the use of the SMOTEBoost method is replaced by using Progressive Boosting (PBoost).

2.3. Evaluation Stage

The evaluation process will be conducted to determine the number of classifiers, data diversity, and also the determination of classification performance. A comparison will be made to compare the results obtained by HAR Method with Progressive Boosting. The Evaluation Stage can be seen in Fig. 3.

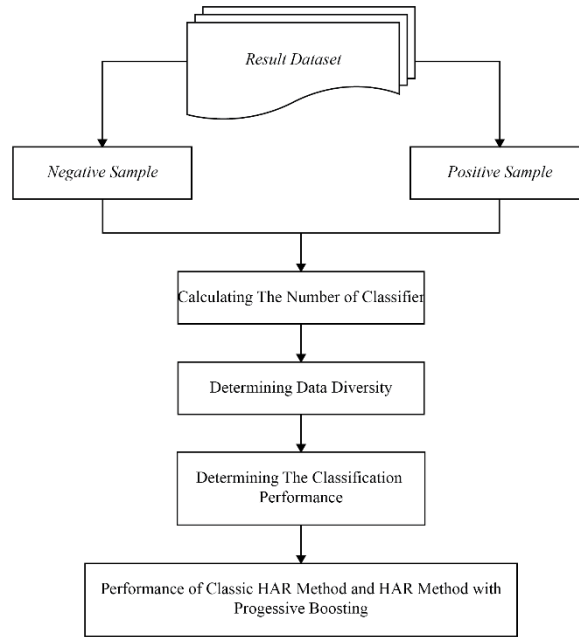


Fig. 3. Evaluation Stage at Hybrid Approach Redefinition with Progressing Boosting

After obtaining the result dataset, it is necessary to evaluate the number of classifiers, data diversity, and also classification performance. Comparisons will be made between Hybrid Approach Redefinition with Progressive Boosting with Hybrid Approach Redefinition classic.

2.4. Progressive Boosting

The main thing about Progressive Boosting is that there are weight matrices for positive and negative samples, which can be seen in (1) and (2). Weighted matrices for positive samples denoted by $W_e^{tmp,+}$ and Negative matrices for negative samples denoted by $W_e^{tmp,-}$. Weighted matrices for positive samples and negative samples can be calculated using (1) to (2) [15].

$$W_e^{tmp,+} = \{W_e^{tmp}(j), j = 1, \dots, (M^+ + \sum_{f=1}^e N_f) | y_j = 1\} \quad (1)$$

$$W_e^{tmp,-} = \{W_e^{tmp}(j), j = 1, \dots, (M^+ + \sum_{f=1}^e N_f) | y_j = -1\} \quad (2)$$

Weighted matrices for true positive, false positive, true negative, and false negative can be calculated using (3) to (6).

$$TP_e = \sum_{k:Y_k=1} W_e^{tmp,+}(k), k = 1, \dots, M^+ \quad (3)$$

$$FP_e = \sum_{k:Y_k=1} W_e^{tmp,-}(k), k = 1, \dots, \sum_{f=1}^e N_f \quad (4)$$

$$TN_e = \sum_{k:Y_k=-1} W_e^{tmp,-}(k), k = 1, \dots, \sum_{f=1}^e N_f \quad (5)$$

$$FN_e = \sum_{k:Y_k=-1} W_e^{tmp,+}(k), k = 1, \dots, M^+ \quad (6)$$

Error of classifier can be calculated using (7).

$$L_e = 1 - A_F = \frac{FP_e + \beta^2 FN_e}{(1 + \beta^2) TP_e + FP_e + \beta^2 FN_e} \quad (7)$$

Weight update parameter α_e can be seen in (8).

$$\alpha_e = \frac{L_e}{1+L_e} \quad (8)$$

The pseudocode from Progressive Boosting is as follows [15].

Input: Training Set: $S = \{(x_i, y_i); i = 1, \dots, M\}, y_i \in \{-1, 1\} M = M^- + M^+$
Output: Predicted score or label: $H(\cdot)$

- 1: Partition non-target samples from S into E clusters $\{P_e; e = 1, \dots, E\}$
- 2: Create a temporary training set and weight vector: $S_1^{temp} \leftarrow \{(x_i, y_i) \in S | y_i = 1\}$ and $w_1^{temp}(k) = 1, k = 1, \dots, M^+$
- 3: Initialize $w_1^{ini} = 1$
- 4: Set $l_b = \frac{M^-}{(1+\beta^2)M^+ + M^-}$
- 5: For $e = 1, \dots, E$ do
- 6: Initialize weight distribution of P_e as $W_e^p(k) = w_e^{ini}, k = 1, \dots, N_e$
- 7: $S_e^{tmp} \leftarrow S_e^{tmp} \cup P_e, W_e^{tmp} \leftarrow W_e^{tmp} \cup W_e^p$
- 8: Normalize W_e^{tmp} such that : $\sum w_e^{tmp} = 1$
- 9: Randomly select N_e non-target samples from S_e^{tmp} based on W_e^{tmp}
- 10: Train C_e on S_e' with W_e'
- 11: Train C_e on S_e^{tmp} and get back labels $Y_j, j = 1, \dots, (M^+ + \sum_{f=1}^e N_f)$
- 12: Calculated weighted matrices and error of classifier using (1) to (7)
- 13: if $L_e > l_b$ go to step 4
- 14: Calculate the weight update parameter using (8)
- 15: update $W_{e+1}^{tmp}(j) = w_e^{tmp}(j) \propto_e^{|y_j - Y_j|/2}$
- 16: Normalize W_{e+1}^{tmp} such that : $W_{e+1}^{tmp} = 1$
- 17: Set $W_{e+1}^{ini} = \max(W_e^{tmp}), y_j = -1$
- 18: End For
- 19: Output the final hypothesis: $H(\cdot) = \sum_{e=1}^E h_e(\cdot) \log \frac{1}{\alpha_e}$

In the pseudocode above, it can be seen that the determination of the weighted matrix for each majority class and minority class is the most basic thing from Progressive Boosting. The process begins with the determination of the weight distribution, which includes several training stages to obtain weighted matrices. Then the performance and errors in the classifier will be calculated. Based on the error classifier parameters can be determined for determining weight updates.

2.5. Classifier

Classifiers can generally be defined as *Decision Region* \mathfrak{R}^n that place an object into a set *class* Ω , where Ω consists of class ω_1, ω_2 , until ω_n . It can be seen in (9) [14].

$$D: \mathfrak{R}^n \rightarrow \Omega \quad (9)$$

Where D is the classifier and is the set of each point in the decision region \mathfrak{R}^i which is intended for *class* ω_i .

2.6. Data Diversity

According to Díez-Pastor et al. from their research about class imbalances [13], the diversity of data is essential in handling imbalance classes. This means that attempted misclassification produced by each classifier is as small as possible, and if there is misclassification, it is expected to occur on different objects or parts [14].

Suppose that $Z = \{z_1, \dots, z_n\}$, which is a dataset that is in the decision region \mathfrak{R}^n , so that $z_j \in \mathfrak{R}^n$ it is an instance involved in the classification problem. Then the output of the classifier D_i as a classifier paired comparison matrix (relationship pairwise classifier) shown in Table 1.

Table 1. Relationship Pairwise Classifier Matrix

	D_k Correct (1)	D_k Wrong (0)
D_i Correct (1)	N^{11}	N^{10}
D_i Wrong (0)	N^{01}	N^{00}

Diversity data can be calculated using Q-Statistics (10) [17].

$$Q_{i,k} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}} \quad (10)$$

2.7. Confusion Matrix

Receiver Operating Characteristic (ROC) Curve is often used to describe performance from the results of a classification or diagnostic rule [18]. ROC Curve is one statistical method that is often used to determine the performance of a classifier. This curve is generated by plotting the true positive fraction of a positive sample in the Y axis with the false positive fraction of a negative sample (False Positive Rate) in the X axis [19]. True Positive and False Positive concepts in the Confusion Matrix are shown in Table 2.

Table 2. Confusion Matrix

	Classified as Positive	Classified as Negative
Positive Samples	True Positive (TP)	False Negative (FN)
Negative Samples	False Positive (FP)	True Negative (TN)

2.8. Classification Performance

In classification performance there are several terms that need to be known, namely: Sensitivity, Specificity, F-Measure, and G-Mean [8], [20].

Sensitivity relates to the ability of the classifier to classify a minority class (positive sample) correctly. The existing value range is in the range of 0 to 1. Sensitivity can be calculated using (11).

$$Sensitivity = \frac{TP}{TP + FN} \quad (11)$$

Specifity relates to the ability of the classifier to classify a negative sample or majority class correctly. The existing value range is in the range of 0 to 1. Specifity can be calculated using (12).

$$Specificity = \frac{TN}{TN + FP} \quad (12)$$

The F-Measure value is usually smaller than 2, the higher the value of F-Measure states that both recall and precision are quite high. G-Mean on the other hand states the balance between positive and negative samples (minority and majority class) [8]. F-Measure and G-Means calculations are shown in (13) to (14).

$$\Phi\text{-}Measure = \frac{2RP}{R+P} \quad (13)$$

$$G\text{-}Mean = \sqrt{TPRate \cdot TNrate} \quad (14)$$

2.9. Hybrid Approach Redefinition with Progressive Boosting

The Pseudocode of the Hybrid Approach Redefinition with Progressive Boosting is as follow.

Preprocessing

Using *Random Undersampling* + PBoost

Pseudocode

Input: Total Size $total_Size$, Number of Majority S_N , Number of Minority S_P

Process:

```

1:  $total\_Size \leftarrow |S|$ 
2:  $S_N \leftarrow \{(x_i, y_i) \in S | y_i = -1\}$ 
3:  $S_P \leftarrow \{(x_i, y_i) \in S | y_i = +1\}$ 
4:  $majority\_Size \leftarrow |S_N|$ 
5:  $minority\_Size \leftarrow |S_P|$ 
6:  $new\_MajoritySize \leftarrow$  Random integer between 2 and  $total\_Size-2$ 
7:  $new\_MinoritySize \leftarrow total\_Size - new\_MajoritySize$ 
8: if  $new\_MajoritySize < majority\_Size$  then
9:    $S' \leftarrow S_P$ 
10:  Take a random sample of size  $new\_MajoritySize$  from  $S_N$ , add the sample to  $S'$ 
11:  Create  $new\_MinoritySize - minority\_Size$  artificial using PBoost
12: else
13:    $S' \leftarrow S_N$ 
14:  Take a random sample of size  $new\_MinoritySize$  from  $S_P$ , add the sample to  $S'$ 
15:  create  $new\_MajoritySize - majority\_Size$  artificial using PBoost
16: end if
17: return  $S'$ 

```

In the above pseudocode, it can be seen that the preprocessing stage begins with a sampling process towards majority classes, to get the balance of the number of instances with minority classes. It is done by moving the majority class instance that has the closest proximity to the minority class into the minority class. It is done by determining a random number that will determine the number of new majority sizes, and this means determining how many instances are transferred to the minority class. In the process, it should also be noted that not too many instances are transferred to minority classes, causing minority classes that have too many instances. It is done by checking the number or size of the instance of the new majority class. If it is greater than the size of the old majority class, the minority class will experience PBoost to move several minority class instances to the majority class. Conversely, if the new majority class is still smaller in the number of instances, than the Majority class will experience PBoost.

Processing

Using Under Bagging and Different Contribution Sampling

Input: S : *Training_Set*, T : Number of Iterations, n : Bootstrap Size

Output: Bagged Classifier: $H(x) = \text{sign}(\sum_{t=1}^T h_t(x))$ where $h_t \in [-1, 1]$ are the induced classifiers

Process:

```

1: for  $t = 1$  to  $T$  do
2:    $S_t$  Preprocessed Data Test using Random Balance
3:   Ensemble Method ( $n, S$ )
4:   Classifying  $S_t$  Using B-SVM
5:   Identifying Negative Samples
6:   Identifying Positive Samples
7:   While ( $\neg \text{EndofNegativeSamples}$ ) do
8:     NewSVSets[] Deleting the Noise Samples in SV Sets
9:     NewNSVSets[] Multiple Random Under-Sampling in NSV Sets
10:  end while
11: For All NewSVSets and NewNSVSets do
12:   New NegativeSampleSets
13: End For
14: While ( $\neg \text{EndofPositiveSamples}$ ) do

```



```

15: PBoostSets[]Deleting the Noise Samples in SV Sets
16: end while
17: For All PBoostSets and NewNSVSets do
18:   New PositiveSampleSets
19: End For
20: For All NewNegativeSampleSets and NewPositiveSampleSets do
21:   ResultDataSet
22: End For
23: End For

```

The processing stage involves the use of Different Contribution Sampling. Based on the dataset that has undergone preprocessing, this stage will involve the division of majority classes and minority classes into SV Sets, and NSV Sets using the BSVM Method. SV Sets and NSV Sets results for each minority, and the majority class will undergo a further process. In Minority Class, SV Sets will experience a noise removal process and will then undergo PBoost stages while NSV Sets will be combined with SV Sets PBoost results to New Positive Sample. Whereas in the Majority Class, NSV Sets will undergo the sampling stage with the RUS method and will be combined with SV Sets, which has been eliminated by noise to New Negative Sample.

3. Results and Discussion

3.1. Dataset Description

The datasets used in this research are Pima, Abalone9vs18, and Yeast2vs8. The description of datasets can be seen in [Table 3](#).

Table 3. Dataset Description

Dataset	#Ex	#Atts	(%Min;%Max)	IR
Pima	768	8	(34.84,66.17)	1.9
Abalone9vs18	731	8	(5.65,94.25)	16.68
Yeast2vs8	482	8	(4.15,95.85)	23.1

3.2. Testing

Testing is done to measure the number of classifiers, data diversity, and classification performance. Testing is done 10 times for each method. The average value of classifier and diversity data can be seen in [Table 4](#).

Table 4. Testing Result for Number of Classifier and Data Diversity of Each Method

Dataset	Hybrid Approach Redefinition		Hybrid Approach Redefinition with Progressive Boosting	
	Number of Classifier	Data Diversity	Number of Classifier	Data Diversity
Pima	177.8	0.626	169.1	0.578
Abalone9vs18	150.8	0.483	148.9	0.477
Yeast2vs8	47.7	0.55	48.1	0.51

In [Table 4](#), it can be seen that for each dataset, Hybrid Approach Redefinition with Progressive Boosting gives better results for the data diversity category compared to the Hybrid Approach Redefinition classic. It is because both the positive samples and the negative samples will be weighted in the form of weighted matrices so that the existing misclassification can be spread to obtain better data diversity. Measurements based on the number of classifiers give results that Hybrid Approach Redefinition with Progressive Boosting will provide better results in the form of smaller classifiers for datasets with small and medium imbalance ratios, while for datasets with large imbalance ratios, Hybrid Approach Redefinition classic slightly better. The testing result for Sensitivity, Specificity, F-Measure, and G-Mean can be seen in [Table 5](#).

The average value of Sensitivity, Specificity, F-Measure, and G-Mean can be seen in [Table 5](#).

Table 5. Testing Result for Sensitivity, Specificity, F-Measure, and G-Mean of Each Method

Dataset	Hybrid Approach Redefinition				Hybrid Approach Definition with Progressive Boosting			
	Sensitivity	Specificity	F-Measure	G-Mean	Sensitivity	Specificity	F-Measure	G-Mean
Pima	0.736	0.606	0.667	0.668	0.741	0.612	0.712	0.673
Abalone9vs18	0.829	0.813	0.491	0.821	0.837	0.829	0.495	0.832
Yeast2vs8	0.79	0.957	0.766	0.869	0.809	0.961	0.781	0.882

Based on [Table 5](#), it can be seen that the Hybrid Approach Redefinition with Progressive Boosting provides better results for sensitivity, specificity, F-Measure, and G-Mean compared to Hybrid Approach Redefinition classes in the three datasets.

3.3. Results

In datasets with several instances that are not too large, the number of classifiers may not affect the computational process, whereas, in datasets with a large number of instances, the number of classifiers needs to be considered. In general, both Hybrid Approach Redefinition Classic and Hybrid Approach Redefinition with Progressive Boosting both provide several classifiers that are not too large. The fewer classifiers in the Hybrid Approach Redefinition are supported by the existence of preprocessing stages using the Random Under Sampling method and also the SMOTEBoost, which can effectively reduce the classifier. It can be seen that Progressive Boosting with weighted matrices on both positive samples and negative samples can make the number of classifiers more efficient so they can provide the number of classifiers in the Hybrid Approach Redefinition that uses SMOTEBoost. However, there is a tendency that for datasets with a larger imbalance ratio, Hybrid Approach Redefinition classic tends to give results that are not too far apart and even in datasets with large imbalance ratios, Hybrid Approach Redefinition classes can produce slightly better results compared to Hybrid Approach Redefinition with Progressive Boosting.

Better data diversity, on the other hand, shows that the ensemble process in the form of classifier merger has been done well, in the Hybrid Approach Redefinition with Progressive Boosting, the results given are better because the weighting on positive samples and negative samples on PBoost is more effective than the SMOTEBoost method. The discussion about sensitivity and specificity is to describe the performance of the classifier in classifying an instance in the majority and minority class. The higher sensitivity means, the more appropriate a classifier is in placing an instance of the minority class correctly. At the same time, the specificity states the accuracy of the classifier in placing an instance in the majority class. The sensitivity and specificity produced by Hybrid Approach Redefinition with Progressive Boosting are better than Hybrid Approach Redefinition classic for both datasets with low, medium, or large imbalance ratios.

The value of F-Measure produced by Hybrid Approach Redefinition classic and Hybrid Approach Redefinition with Progressive Boosting is excellent, which shows that both methods successfully classify instances in minority classes correctly or instances that should be minority classes but placed in majority classes. So that based on the results of handling the imbalance class, an instance of minority class can be obtained, so that an interesting pattern of minority classes can still be obtained. The G-Means measurement itself states the balance of the accuracy of the classification results between minority and majority class. Hybrid Approach with Progressive Boosting Redefinition still gives better results for F-Measure and G-Mean.

4. Conclusion

Based on the tests that have been done, the results show that Hybrid Approach Redefinition with Progressive Boosting gives better results in handling the imbalance class when compared to Hybrid Approach Redefinition classic. A number of indicators such as the number of

classifiers, diversity data, and also classification performance provided by Hybrid Approach Redefinition with Progressive Boosting are better compared to Hybrid Approach Redefinition classic.

Through the results of this study, an excellent multi class imbalance treatment was obtained through Hybrid Approach Redefinition with Progressive Boosting which is a hybrid ensemble approach that can obtain good data diversity, small number of classifiers, and good classification performance. However, what needs to be considered is the tendency of a large number of classifiers for datasets with a large imbalance ratio. This may be refined in future research.

References

- [1] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, "The impact of class imbalance in classification performance metrics based on the binary confusion matrix," *Pattern Recognit.*, vol. 91, pp. 216–231, Jul. 2019, doi: [10.1016/j.patcog.2019.02.023](https://doi.org/10.1016/j.patcog.2019.02.023).
- [2] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.)*, vol. 42, no. 4, pp. 463–484, 2012, doi: [10.1109/TSMCC.2011.2161285](https://doi.org/10.1109/TSMCC.2011.2161285).
- [3] A. Fernández, V. López, M. Galar, M. J. del Jesus, and F. Herrera, "Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches," *Knowledge-Based Syst.*, vol. 42, pp. 97–110, Apr. 2013, doi: [10.1016/j.knosys.2013.01.018](https://doi.org/10.1016/j.knosys.2013.01.018).
- [4] N. Japkowicz and R. Holte, "AAAI 2000 Workshop Reports on learning from imbalanced data-sets," *AI Mag.*, vol. 22, no. 1, pp. 127–136, Mar. 2001, doi: [10.1609/aimag.v22i1.1552](https://doi.org/10.1609/aimag.v22i1.1552).
- [5] S. Wang and X. Yao, "Diversity analysis on imbalanced data sets by using ensemble models," in *2009 IEEE Symposium on Computational Intelligence and Data Mining*, Mar. 2009, pp. 324–331, doi: [10.1109/CIDM.2009.4938667](https://doi.org/10.1109/CIDM.2009.4938667).
- [6] N. Lachiche and P. Flach, "Improving Accuracy and Cost of Two-Class and Multi-Class Probabilistic Classifiers Using ROC Curves," in *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, 2003, pp. 416–423, available at: <https://www.aaai.org/Papers/ICML/2003/ICML03-056.pdf>.
- [7] Z. Yang, T. Zhang, J. Lu, D. Zhang, and D. Kalui, "Optimizing area under the ROC curve via extreme learning machines," *Knowledge-Based Syst.*, vol. 130, pp. 74–89, Aug. 2017, doi: [10.1016/j.knosys.2017.05.013](https://doi.org/10.1016/j.knosys.2017.05.013).
- [8] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognit.*, vol. 40, no. 12, pp. 3358–3378, Dec. 2007, doi: [10.1016/j.patcog.2007.04.009](https://doi.org/10.1016/j.patcog.2007.04.009).
- [9] C. Jian, J. Gao, and Y. Ao, "A new sampling method for classifying imbalanced data based on support vector machine ensemble," *Neurocomputing*, vol. 193, pp. 115–122, Jun. 2016, doi: [10.1016/j.neucom.2016.02.006](https://doi.org/10.1016/j.neucom.2016.02.006).
- [10] F. Ren, P. Cao, W. Li, D. Zhao, and O. Zaiane, "Ensemble based adaptive over-sampling method for imbalanced data learning in computer aided detection of microaneurysm," *Comput. Med. Imaging Graph.*, vol. 55, pp. 54–67, Jan. 2017, doi: [10.1016/j.compmedimag.2016.07.011](https://doi.org/10.1016/j.compmedimag.2016.07.011).
- [11] Hartono, E. Ongko, O. S. Sitompul, Tulus, E. B. Nababan, and D. Abdullah, "Hybrid Approach Redefinition (HAR) Method with Loss Factors in Handling Class Imbalance Problem," in *Proceeding - 2018 International Symposium on Advanced Intelligent Informatics: Revolutionize Intelligent Informatics Spectrum for Humanity, SAIN 2018*, 2019, doi: [10.1109/SAIN.2018.8673370](https://doi.org/10.1109/SAIN.2018.8673370).
- [12] A. Fernández, S. García, F. Herrera, and N. V. Chawla, "SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary," *Journal of Artificial Intelligence Research*. 2018, doi: [10.1613/jair.1.11192](https://doi.org/10.1613/jair.1.11192).
- [13] J. F. Díez-Pastor, J. J. Rodríguez, C. I. García-Osorio, and L. I. Kuncheva, "Diversity techniques improve the performance of the best imbalance learning ensembles," *Inf. Sci. (Ny)*, vol. 325, pp. 98–117, Dec.

2015, doi: [10.1016/j.ins.2015.07.025](https://doi.org/10.1016/j.ins.2015.07.025).

- [14] L. I. Kuncheva, *Combining Pattern Classifiers*. Wiley, 2004, available at: [Google Scholar](#).
- [15] R. Soleymani, E. Granger, and G. Fumera, "Progressive boosting for class imbalance and its application to face re-identification," *Expert Syst. Appl.*, vol. 101, pp. 271–291, Jul. 2018, doi: [10.1016/j.eswa.2018.01.023](https://doi.org/10.1016/j.eswa.2018.01.023).
- [16] J. Alcalá-Fdez *et al.*, "KEEL: a software tool to assess evolutionary algorithms for data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009, doi: [10.1007/s00500-008-0323-y](https://doi.org/10.1007/s00500-008-0323-y).
- [17] G. U. Yule, "VII. On the association of attributes in statistics: with illustrations from the material of the childhood society, &c," *Philos. Trans. R. Soc. London. Ser. A, Contain. Pap. a Math. or Phys. Character*, vol. 194, no. 252–261, pp. 257–319, Jan. 1900, doi: [10.1098/rsta.1900.0019](https://doi.org/10.1098/rsta.1900.0019).
- [18] L. B. Lusted, "Signal Detectability and Medical Decision-Making," *Science (80-.)*, vol. 171, no. 3977, pp. 1217–1219, Mar. 1971, doi: [10.1126/science.171.3977.1217](https://doi.org/10.1126/science.171.3977.1217).
- [19] C. Gigliarano, S. Figini, and P. Muliere, "Making classifier performance comparisons when ROC curves intersect," *Comput. Stat. Data Anal.*, vol. 77, pp. 300–312, Sep. 2014, doi: [10.1016/j.csda.2014.03.008](https://doi.org/10.1016/j.csda.2014.03.008).
- [20] A. Ali, S. M. Shamsuddin, and A. L. Ralescu, "Classification with class imbalance problem: A review," *Int. J. Adv. Soft Comput. its Appl.*, vol. 7, no. 3, pp. 176–204, 2015, available at: [Google Scholar](#).