An ensemble approach based classification of Binary imbalanced dataset Using SMOTE

Ms. Shivani sharma, and Dr. Nirupma tiwari
[1,2]Department of computer science and engineering, SRCEM, RGPV University, Bhopal

Abstract- **In different fields, such as machine learning & data mining, class imbalances have been one of the most complex issues for the past few decades. The unique condition of an imbalanced dataset that distributes each class of a particular dataset unevenly. The positive class is slightly smaller than the negative one. Many standard classification algorithms in this case do not classify instances related to the positive class. Typically the main goal of the classification task is a positive class. To deal with this problem, various approaches were proposed, for example sampling dependent over-sampling, under sampling, classification level enhancements, or a combination of two or more classifiers. The major problem however is that most solutions have negative class, computational cost, storage problem, or long training period. Data up sampling or down sampling may resolve a possible solution to the issue of skewness of data. In this work, a hybrid technique is presented, followed by a random forest algorithm (SMO-RF), to categorized binary imbalanced data using the Technique of Synthetic Minority Oversampling. We have tested our model with four standard imbalanced datasets & obtain higher G-mean, F-measure as well as ROC values for all data sets.**

Keywords— Imbalanced Data, Data Classification, Prediction, Binary Classification, Random Forest Classifier, SMOTE.

I. INTRODUCTION

patterns. Many application fields have been developed and successfully extended to several classifying learning algorithms, such as decision tree (DT), back propagation neural networks (BPNN), K-nearest neighbor (KNN), support vector machine (SVM), Bayesian network (BN), and modern associative classification. However, the imbalanced class distribution of a data set has made most classification learning algorithms with a sufficiently balanced distribution very challenging. The imbalanced data was distinguished by far more class instances than others. Because unusual cases happen rarely, classification rules forecasting small classes appear to be rare, uncovered, or ignored; thus, research samples from small classes are more commonly misclassified than those from the dominant classes. [1].

Imbalanced data is a general classification issue. This is an incredibly significant concept as it occurs in mainly real areas. It's especially pertinent to strongly imbalanced datasets (when the class ratio is high). Several approaches to resolving imbalanced problem training sets were built in supervised learning. This type of technology is categorized into 2 major categories: algorithm & data level [2]. Imbalanced data are applied where a single interest class is superfluous (discussed as a minority or a positive class), which means that the study is unevenly distributed among other interesting groups (called a major class or a negative class). Imbalanced data represent a typical issue with the credit rating that greatly raises the number of positive assessments relative to that of poor performance. This leads to a phenomenon in which the outcomes of the study are partially prejudiced by statistic damage to the governing party, while a false sample analysis leads to significant financial damage [3].

There may be two class labels for a classification predictive modeling problem. This is the most simple classification problem and is known as a two-class or binary classification [4]. Alternatively, there can be more than two classes in the problem, such as three, ten, or even centuries. These problems are referred to as problems of classification of multiple classes [5].

Our focus is on problems with data classification where only a binary data representation is available. Under different circumstances, these binary representations can occur. In certain cases, the compressive acquisition may naturally occur. For example, distributed systems can have bandwidth and energy restrictions that require extremely coarse measurement quantization. Binary data representation in hardware applications can also be especially fascinating because computer technology and promoting fast hardware equipment is relatively inexpensive. Such advantages have helped, for example, the success of 1-bit Sigma-Delta converters. Alternatively, in the interest of data compression and speed, binary, heavily quantized, or compressed representations may be part of a design classification algorithm [6].

A classifier of class-imbalanced is rule for forecasting new sample class members from an available data set where class sizes are significantly different. If class sizes are extremely different, most benchmark classification algos could benefit larger (majority) class producing inadequately in the proposition of minority class. Various methods were proposed in 2 groups, namely algorithm, and data, for the problem of imbalanced class. In data group, various sampling techniques, such as query-based learning sampling, are developed and clustered pre-processing methods, for example, SMOTE method [7].

II. IMBALANCE DATA

A significant and controversial trend in our study is learning from imbalanced datasets. These forms of data generate partial results. Think of medical dataset with 50 true negative (majority class) & 20 true positives (minority

class) values. Whether half is chosen for preparation & remaining (25 are well & 10 are sick), we locate 90 percent accuracy. Outcome suggests that classification is fairly well done. However, where all the negative values (healthy individuals) have been appropriately defined and only 5 out of 10 positive values (sick persons) have been properly listed. The classifier is more sensitive in that case to identify patterns of a majority class, but less sensitive to identification of the patterns of minority class. That is due to an imbalance in training data. In further words, classifier concludes that 5 of 10 unwell people are well whenever it's not case. This sort of outcome eventually leads to more devastation if the data is disclosed in real-time environments like genetics, biomedicine, intrusion detection, radar signals, risk management & credit card scoring.

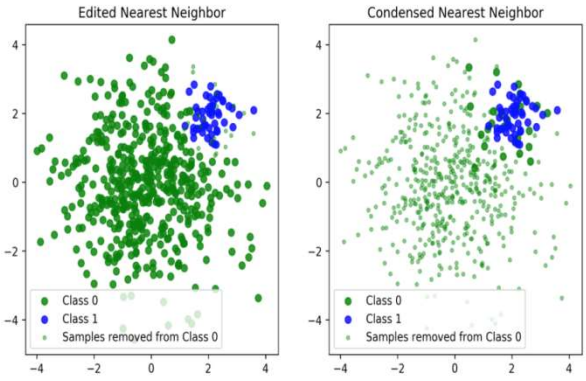

Figure 1: Imbalanced Data.

III. LITERATURE REVIEW

Liu, C.-L., & Hsieh, P.-Y. (2020) to cope with imbalance issues, to use modelling and sampling technologies for the production of synthetic data, suggest a new paradigm, known as model-based synthetic sampling (MBS). The core concept behind the approach is to use regression model to capture the relationship between characteristics and to take into consideration data diversity in the data generation process. We perform tests on 13 datasets and use 10 techniques to compare the proposed process. The findings of the experiments demonstrate that the proposed approach is both comparative and stable. We also carry out thorough analysis and visualizations of the proposed method to show empirically why good data samples can be produced [9].

Ma, X., & Shi, W. (2020) Offer a modern architecture for anomaly detection that facilitates intrusion detection in particular. Our proposed paradigm for anomaly detection combines reinforcement learning with strategies of class imbalance. To address the class imbalance issue, we implement an adapted SMOTE to reshape environmental agents' behaviour towards improved efficiency. The suggested AESMOTE model, in particular, exceeds in several cases the AE-RL. The test results show accuracy above 0.82 and F1 above 0.824 [10].

Chen, L., et al. (2019) In field of cancer diagnostics, blood sample centres & industrial equipment, imbalanced datasets are common. The training model uses conventional ML algorithms based on these imbalanced datasets assume that minority-level class samples would not work well and this could cause massive loss. Therefore, a research hotspot is how to boost the classification of imbalanced data sets. A new enhanced SMOTE algo is provided in this work because of this problem [11].

Nair, P., & Kashyap, I. (2019) Two approaches for data pre-processing were merged to create a hybrid pre-processing methodology in this proposed procedure. 2 pre-processing technologies are resembled technology & IQR (inter-quartile range) technology. Any Imbalanced data sets of outliers were taken for the analysis as parameters. The classification findings obtained were considered to be much superior to those produced without the pre-processing procedure [12].

Khadijah, S. N. Endah (2018) In this analysis, two separate approaches are tested, SMOTE (depends upon data method) & weighted ELM (depends upon algorithmic method), to control imbalanced data distribution. Two public, incomplete, multi-class data sets, GCM(Global Cancer Map), & Subtypes-leukemia, have been used to evaluate the performance of the proposed solution. Experiment outcome indicates that SMOTE & weighted ELM implementation on GCM datasets does not have a major impact on classification performance. In comparison to the Subtype Leukemia dataset, SMOTE and weighted ELM implementations have increased the accuracy of classifications compared to previous studies. In general, the findings indicate that weighted ELMs are marginally better than SMOTE to boost minority class accuracy [13].

Wu, X et al. (2018) Proposed methods so far have not been sufficiently strong to forecast kin among people only through their faces. Initial experiments using deep CNNs, primarily due to minimal training data, did not demonstrate their full potential. We suggest a new approach to kinship verification focused on color features & ELM (Extreme Learning Machines) to alleviate this challenge. ELM seeks to handle limited training sets, but it has shown that color features provide a major difference over gray-scale counterparts. We assess our approach to 3 kinship databases, KinFaceW-I, KinFaceW-II as well as TSKinFace that are publicly accessible and available. -The findings achieved are positively contrasted with other state-of-the-art approaches, comprising deep-learning methods [14].

Pristyanto, Y., et al. (2018) This study aims to explore the imbalanced class of multiclass EDM data set processing mechanisms using SMOTE and OSS combinations. The SMOTE and OSS approach offers the dataset distribution balance mechanism so that the effects of classification are increased in terms of classification accuracy. The findings show that SMOTE and OSS will increase the SVM's efficiency as the classification method used in this analysis. The combination of methods yields precision, sensitivity, specificity and g-mean value as high

as 88,637%, 92,292%, 95,554%, and 93,796%. Therefore, on EDM's multiclass data collection, SMOTE and OSS can be a suitable alternative for imbalanced class [15].

Koto, F. (2014) In the supervised learning process, the imbalanced data set also become an obstacle. An imbalance is a case where the example of data from one class of training is considerably higher than the examples from the other group. Our research discusses three changes for SMOTE to cover cases not already completed through SMOTE, SMOTE-Cosine, and the selected-SMOTE. Our studies have been done with 18 separate datasets to study the proposed procedure. The findings show that our SMOTE proposed to boost B-ACC and F1 score [16].

IV. RESEARCH METHODOLOGY

A. Problem Formulation

Due to the presence of missing values, data noise, and class imbalance, it is quite difficult to produce accurate results from data mining models. Most of the classification approaches proposed so far have ignored the missing values or chosen a specific missing value imputation method specific to a dataset. There is no single imputation method fit for all datasets. There is a need to have a comparative analysis of missing value imputation approaches that is independent of the problem and then accurately classify the unseen examples. Further, most of the missing value imputation techniques have not imputed missing values from datasets with noise. This raises a need to develop an imputation technique which can impute the missing value correctly in the presence of noise. Finally, most techniques handling imbalanced classes worked on binary classification. It necessitates having an approach to work on imbalanced datasets with multiple classes.

"To design approaches to handle missing values, attribute noise, and imbalanced classes to enhance the quality of data & prediction accuracy of classification."

B. Proposed Methodology

To overcome such problems we have proposed a random forest classifier for imbalanced data classification depends upon the SMOTE technique (SMO-RF). In the previous research work, an Extreme learning machine was hybridized with SMOTE on the same 4 datasets and was compared with many machine learning algorithms. However, ELM has some disadvantages such as over-fitting problems. Also, it is not more precise and accurate when compared to the usual MLP. Therefore, we need to find a better model for overcoming the limitations of ELM. Hence we introduced the SMO-RF model for the enhancement in the performance of the existing model thereafter improving the results.

V. VOTING CLASSIFIER

A Voting Classifier is a Machine Learning algo focused on the maximum likelihood of the chosen class as the output and trains on an ensemble of various examples.

This merely summarizes the results of the voting classifier and forecasts the performance category on basis of the largest majority. Instead of developing different unique models for each of them, we build a single model that trains and predicts performance based on the collective vote majority of the performance groups. Random forest and logistical regression (typically varying types) and basic statistics (such as the average) are introduced in our proposed projects.

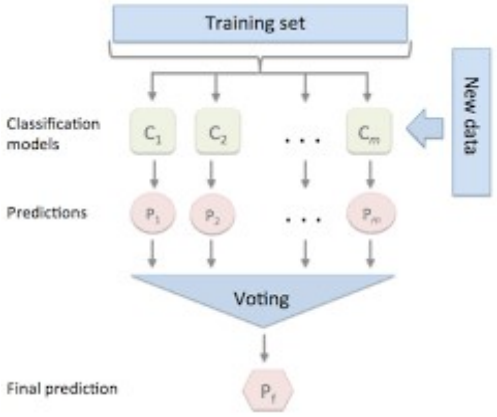

Figure 2: Voting Classifier Architecture.

1. Import dataset
2. import model selection
3. Import Bagging
4. Import Gradient Boosting
5. ensemble import Voting Classifier
6. Review=data set. load_ review()
7. x,y=review. data
8. clf1=Bagging(random state=1)
9. clf2=Gradient Boosting Classifier(random state=1)
10. eclf=Ensemble Vote Classifier (clfs=[clf1,clf2],weights=[1,1])
11. labels = ['Random Forest', 'Logistic Regression', 'Ensemble']
12. Scores = model selection. cross Val_ score (clf, x, y, cv=0, scoring='accuracy')
13. Stop

VI. RANDOM FOREST (RF) CLASSIFIER

Firstly, RF was introduced by Leo Breiman from California University in 2001. It is collected from several simple classifiers (decision tree) which are independent of each other. A sample will be included in the new classifier and class label of this sample depending on voting outcomes from every single classification [17].

The main steps for RF classifier are as follows:

1. Set the proper "M" value, which is the number of components of each sub-set of features.
2. Choose a new subset of feature hk from the entire feature set randomly based on the M value. hk is free from another subset in h1; ...; hk sequence.
3. Training the data set for each training category with the feature sub-set to construct a decision tree. Every single category can be represented as h(X, hk) (wherever X specifies inputs).
4. Select a new hk and repeat it until all the feature subsets are moving. An RF classifier has been achieved.
5. Input the test set. Decide based on voting outcomes for each classification of this sample.

VII. LOGISTIC REGRESSION

LR is a conventional & classical, commonly applied statistical process in academia and industry. In comparison to linear regression used to forecast numerical response, logistic regression solves a problem of classification.

For instance, when a person applies for a loan from the bank, the bank is interested in if this applicant will default in the future? (Default or not default). Another explanation, like LR, is to predict the probability that the applicant will default. Owing to probability nature, the prediction will fall in [0 or 1]. By rule of thumb, if the predicted probability is equal to or greater than 0.5, after that we can label this applicant as 'default'; if the predicted probability is smaller than 0.5, next we can label this applicant as 'not default'.

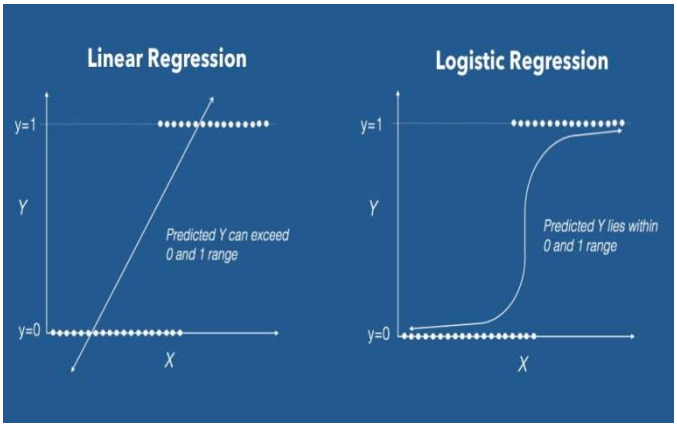

Figure 3: Graph showing the difference between linear regression and logistic regression.

VIII. SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE (SMOTE)

SMOTE is an over-sampling technique, generating synthetic class samples for minorities. It can be better than simple over-sampling also it is generally utilized [18].

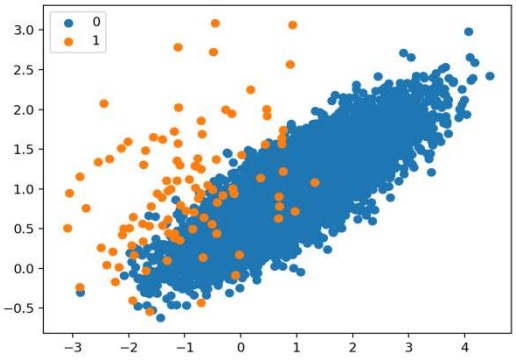

Figure 4: SMOTE for Imbalance Classification.

We suggest an over-sampling technique wherever a class of minorities is over-sampled by producing 'synthetic' cases instead of over-sampling with replacement. It is based on a methodology that has proven effective in the identification of a handwritten character (Ha & Bunke, 1997). By conducting such real data procedures, they generated additional training data. Operations such as rotation and skew were in their case natural ways to disrupt training data. In a less application-specific way, we construct synthetic instances using "feature space" rather than "data Space." Minority class is over-sampled by taking an example of any minority class & by incorporating synthetic examples in segments that join each of KNN minority classes. Based on the amount of over-sampling required, neighbors by KNN are randomly selected. We currently have five nearest neighbors to execute our plan [19].

IX. RESULTS ILLUSTRATIONS

We experimented with the proposed method with four standard data sets (Pima, Ecoli, Segment, and Yeast) for assessment and validation purposes of our assertion. These datasets have been explicitly developed to research the imbalanced data classification problem [20].

a) Results for PIMA Dataset

```
Test Results :
ROC AUC    : 0.7269%
Accuracy   : 74.4589%
Precision  : 66.2791%
Recall     : 65.5172%
F1-measure: 65.8960%
Sensitivity : 79.8611%
Specificity : 65.5172%
Gmean : 72.3345%
```

```
plt.figure()
plt.matshow(matrix, cmap='Pastel1')

for x in range(0, 2):
    for y in range(0, 2):
        plt.text(x, y,matrix[x, y])
```

Figure 5: Accuracy measurement by ELM on Pima Diabetes dataset.

```
print('Test Results :')
print("ROC AUC   : {:.4f}%".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.7921%
Accuracy  : 78.3550%
Precision : 66.9811%
Recall    : 82.5581%
F1-measure: 73.9583%
Sensitivity : 75.8621%
Specificity : 82.5581%
Gmean : 79.1393%
```

Figure 6: Accuracy measurement by RF on Pima Diabetes dataset.

This dataset comes from the National Institute of Diabetes & Digestive and Kidney Diseases. The purpose of the dataset consists of estimating, based on some diagnostic measures used in the data set, whether a patient has diabetes or not. The extraction of these instances from a wide database has been subject to some limitations. All patients are females of Pima Indian heritage aged at least 21 years old.

b) Results for ECOLI Dataset

```
print('Test Results :')
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.9373
Accuracy  : 95.0495%
Precision : 87.5000%
Recall    : 91.3043%
F1-measure: 89.3617%
Sensitivity : 96.1538%
Specificity : 91.3043%
Gmean : 93.6977%
```

Figure 7: Accuracy measurement by ELM on Ecoli dataset.

```
recall=recall_score(y_test, predictions)
f1 = f1_score(y_test, predictions)
sensitivity = matrix[0,0]/(matrix[0,0]+matrix[0,1])
specificity = matrix[1,1]/(matrix[1,0]+matrix[1,1])
gmean=math.sqrt(sensitivity*specificity)
print('Test Results :')
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.9413
Accuracy  : 97.0297%
Precision : 94.4444%
Recall    : 89.4737%
F1-measure: 91.8919%
Sensitivity : 98.7805%
Specificity : 89.4737%
Gmean : 94.0120%
```

```
plt.figure()
```

Figure 8: Accuracy measurement by RF on Ecoli dataset.

c) Results for YEAST Dataset

```
recall=recall_score(y_test, predictions)
f1 = f1_score(y_test, predictions)
sensitivity = matrix[0,0]/(matrix[0,0]+matrix[0,1])
specificity = matrix[1,1]/(matrix[1,0]+matrix[1,1])
gmean=math.sqrt(sensitivity*specificity)
print('Test Results :')
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.8742
Accuracy  : 89.2377%
Precision : 49.3827%
Recall    : 85.1064%
F1-measure: 62.5000%
Sensitivity : 89.7243%
Specificity : 85.1064%
Gmean : 87.3848%
```

Figure 9: Accuracy measurement by ELM on Yeast dataset.

```
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.8253
Accuracy  : 92.1525%
Precision : 55.7692%
Recall    : 70.7317%
F1-measure: 62.3656%
Sensitivity : 94.3210%
Specificity : 70.7317%
Gmean : 81.6792%
```

```
plt.figure()
plt.matshow(matrix, cmap='Pastel1')
```

Figure 10: Accuracy measurement by RF on Yeast dataset.

d) Results for SEGMENT Dataset

```
gmean=math.sqrt(sensitivity*specificity)
print('Test Results :')
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.9800
Accuracy  : 98.8456%
Precision : 94.8454%
Recall    : 96.8421%
F1-measure: 95.8333%
Sensitivity : 99.1639%
Specificity : 96.8421%
Gmean : 97.9961%
```

Figure 11: Accuracy measurement by RF on Segment dataset.

```
gmean=math.sqrt(sensitivity*specificity)
print('Test Results :')
print("ROC AUC   : {:.4f}".format(auc))
print("Accuracy  : {:.4f}%".format(accuracy*100))
print("Precision : {:.4f}%".format(precision*100))
print("Recall    : {:.4f}%".format(recall*100))
print("F1-measure: {:.4f}%".format(f1*100))
print("Sensitivity : {:.4f}%".format(sensitivity*100))
print("Specificity : {:.4f}%".format(specificity*100))
print("Gmean : {:.4f}%".format(gmean*100))
```

```
Test Results :
ROC AUC   : 0.9953
Accuracy  : 99.8557%
Precision : 100.0000%
Recall    : 99.0654%
F1-measure: 99.5305%
Sensitivity : 100.0000%
Specificity : 99.0654%
Gmean : 99.5316%
```

Figure 12: Accuracy measurement by ELM on Segment dataset.

Table 5.1 is the tabular representation of the results obtained by the Random Forest algorithm on all the datasets collected. We utilized G-mean (Geometric Mean), F-measure & ROC curve as calculation tools for evaluating purposes since these methods are mainly used for the assessment of data classification algorithms.

Table 1: Testy Result

Parameters	Accuracy	Precision	Recall	F1-measure	Sensitivity	Specificity	Gmean
Pima Dataset	78.355%	66.98%	82.55%	73.95%	75..862%	82.55%	79.13%
Ecoli Dataset	97.029%	94.444%	89.47%	91.89%	98.78%	89.78%	94.012%
Yeast Dataset	92.15 %	55.76%	70.73%	62.36%	94.321%	70.73%	81.67%
Segment Dataset	99.85%	100.0000%	99.06%	100.0000%	100.0000%	99.06%	99.53%

Proposed model has shown to perform best b/w 2 models for all 4 datasets, each of them having a different imbalance ratio. Figure 13 is the graphical representation of the ROC curves of all the datasets using the RF algorithm.

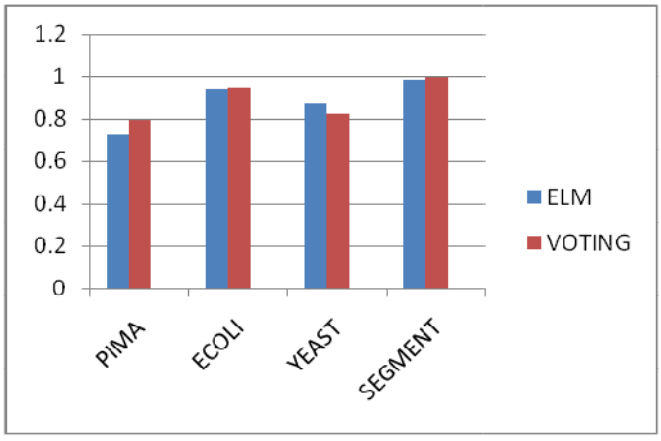

Figure 13: ROC comparison of base and proposed methodology.

X. CONCLUSION

This work tackled the issues of imbalanced datasets. The classical classifiers usually concentrated on dominant classes and ignore minor classes exclude essential details related to minor classes. Subsequent approaches for over-sampling & undersampling are also used to resolve imbalanced data collection in the skewed repository. Nevertheless, either the lack of valuable data or the addition of irrelevant classification data will impact the predictive precision in the imbalanced dataset of minority instances. This work suggests a new hybrid classifier SMO-VOT (which uses entropy and information gain as a fitness function), as a solution to the imbalanced classification task. The findings show that the proposed SMO-VOT methodology performs more effectively and efficiently on assessment processes, including precision, GM, ROC, F-measure, and Precision and recall than the previous method proposed.

REFERENCES

[1] SUN, Y., WONG, A. K. C., & KAMEL, M. S. (2009). *CLASSIFICATION OF IMBALANCED DATA: A REVIEW. International Journal of Pattern Recognition and Artificial Intelligence, 23(04), 687–719.* doi:10.1142/s0218001409007326.

[2] Enislay Ramentol, Yailé Caballero, Rafael Bello, Francisco Herrera," 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 DOI 10.1007/s10115-011-0465-6

[3] Feng Shen," A novel ensemble classification model based on neural networks and a classifier optimization technique for imbalanced credit risk evaluation",Physica A 526 (2019) 121073.

[4] https://www.hindawi.com/journals/mpe/2015/269856/

[5] https://machinelearningmastery.com/what-is-imbalanced-classification/

[6] https://jmlr.org/papers/volume19/17-383/17-383.pdf
[7] Purwar, A., & Singh, S. K. (2014), "*Issues in data mining: A comprehensive survey", 2014 IEEE International Conference on Computational Intelligence and Computing Research,* doi:10.1109/iccic.2014.7238447.
[8] https://link.springer.com/chapter/10.1007/978-3-642-24958-7_85
[9] Liu, C.-L., & Hsieh, P.-Y. (2020). Model-Based Synthetic Sampling for Imbalanced Data. IEEE Transactions on Knowledge and Data Engineering, 1–1. doi:10.1109/tkde.2019.2905559.
[10] Ma, X., & Shi, W. (2020). AESMOTE: Adversarial Reinforcement Learning with SMOTE for Anomaly Detection. IEEE Transactions on Network Science and Engineering, 1–1. doi:10.1109/tnse.2020.3004312.
[11] Chen, L., Dong, P., Su, W., & Zhang, Y. (2019). Improving Classification of Imbalanced Datasets Based on KM++ SMOTE Algorithm. 2019 2nd International Conference on Safety Produce Informatization (IICSPI). doi:10.1109/iicspi48186.2019.9096022
[12] Nair, P., & Kashyap, I. (2019). Hybrid Pre-processing Technique for Handling Imbalanced Data and Detecting Outliers for KNN Classifier. 2019 International Conference on Machine Learning, Big Data, Cloud, and Parallel Computing (COMITCon). doi:10.1109/comitcon.2019.8862250
[13] Khadijah, S. N. Endah, R. Kusumaningrum, and Rismiyati, "The Study of Synthetic Minority Over-sampling Technique (SMOTE) and Weighted Extreme Learning Machine for Handling Imbalance Problem on Multiclass Microarray classification," 2018 2nd International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 2018, pp. 1-6, doi: 10.1109/ICICOS.2018.8621657
[14] Wu, X., Feng, X., Boutellaa, E., & Hadid, A. (2018). Kinship Verification using Color Features and Extreme Learning Machine. 2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP). doi:10.1109/siprocess.2018.8600423
[15] Sanabila, H. R., & Jatmiko, W. (2018). Ensemble Learning on Large Scale Financial Imbalanced Data. 2018 International Workshop on Big Data and Information Security (IWBIS). doi:10.1109/iwbis.2018.8471702
[16] Koto, F. (2014). SMOTE-Out, SMOTE-Cosine, and Selected-SMOTE: An enhancement strategy to handle an imbalance in data level. 2014 International Conference on Advanced Computer Science and Information System. doi:10.1109/icacsis.2014.7065849
[17] Parmar, A., Katariya, R., & Patel, V. (2018). *A Review on Random Forest: An Ensemble Classifier. Lecture Notes on Data Engineering and Communications Technologies, 758–763.* doi:10.1007/978-3-030-03146-6_86
[18] https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-106
[19] Nitesh V. Chawla et al., "SMOTE: Synthetic Minority Over-sampling Technique", Journal of Artificial Intelligence Research 16 (2002) 321–357.
[20] F. Herrera, M. J. del Jesus, S. Garc´ıa, and A. Fernandez, "A study of ´ the behavior of linguistics fuzzy rule-based classification system in the framework of imbalanced data sets," Fuzzy Sets and Systems, 2008.