Handling Class Imbalance Problem using Oversampling Techniques: A Review

Anjana Gosain
Department of Information Technology
USICT, GGS Indraprastha University
New Delhi, India
anjana_gosain@hotmail.com

Saanchi Sardana
M. Tech Computer Science and Engineering
USICT, GGS Indraprastha University
New Delhi, India
saanchi.sardana@gmail.com

***Abstract*—The objective of classifier is to classify objects of a data set into one or more classes based on its characteristics. In real life applications, classifiers are applied on data sets which are unbalanced i.e. some classes having very less number of instances known as minority classes as compared to other classes known as majority classes. Classification algorithms are highly accurate for the majority classes but significantly less accurate for the minority classes. Unbalanced data sets have a negative effect on classification performance of traditional classification algorithms. Analyzing such problem is called class imbalance problem. To solve Class Imbalance Problem different techniques have been proposed at the Data level, Algorithm level and at the Hybrid level. Most commonly used data balancing techniques are over and under sampling for handling the class imbalance problem. In our paper we compare various oversampling techniques which are SMOTE (Synthetic minority oversampling approach), ADASYN, Borderline-SMOTE, Safe-Level SMOTE by applying different classifiers to the problem and observing various performance metrics.**

Keywords—Oversampling, SMOTE, ADASYN, BORDERLINE SMOTE, SAFE LEVEL SMOTE, class imbalance problem.

I. INTRODUCTION

Real world domains generates large amount of data with imbalanced distribution. Imbalanced distribution of classes in datasets appears when the proportion of one class has a higher ratio than the other class. Class having large number of instances is called majority class and the ones having less number of instances is called minority class [3]. The underrepresented classes i.e. the minority classes are apparently anticipated as rare events, or presumed as noise or outliers which lead to more misclassification of minority classes. In real life situations sometimes minority class is of more interest than the majority class for example- oil spill detection, credit card frauds, shuttle system failure, sentiment analysis, web spam detection, risk management and nuclear explosion, video mining, text mining, medical and fault diagnosis, anomaly detection and etc. Here, minority class is of more concern and importance than the prevalent class [1, 2, 5, 12, 14]. As traditional classification algorithms are not able to correctly classify the minority class such situation is called Class Imbalance Problem. Class imbalance problem significantly affects the performance and pose serious challenges for machine learning techniques [4].

The primary issue of conventional classification algorithms in learning with class imbalance problem is that they are based on the assumption of equal distribution of instances in all the classes. Performance of classification algorithms is mostly evaluated using predictive accuracy and their goal is to minimize overall error to which minority class contribution is little. Thus the predictions yielded by such algorithms are not precise and cause misinterpretation of data. So as to improve the accuracy of classification algorithms, number of solutions has been proposed by researchers to address the class imbalance problem. The techniques used to solve the imbalance data set problem are grouped under three categories – External (data level) approaches, Internal (algorithm level) approaches and their hybrid form [12-14].

In external approaches (data level) datasets are first balanced and the conventional classification algorithms are applied so that classifiers performance does not get biased towards the majority class. Rebalancing the data sets is done either by under sampling i.e. removing majority class instances or oversampling i.e. by adding new minority instances to the datasets [16, 17]. In data level approach classification algorithms remains unchanged i.e. independent from the classifier's logic, as rebalancing of datasets is done before classification, thus naming these techniques as pre-processing techniques [15].

In internal approaches (algorithm level) researchers developed new classification algorithms or improved the existing ones to deal with class imbalance problem, without any modification being done on the original dataset. Two sub-classes of these approaches are – Cost sensitive algorithms and ensemble methods [18]. Cost sensitive method assigns different weightage to each class i.e. different misclassification cost to reduce the overall cost for both internal and external level approaches [21]. Secondly, the Ensemble method is based on learning from multiple classifiers simultaneously, are also used to boost the performance of weak learners to strong learners.

Hybrid approaches combine the data level techniques and algorithm level techniques or data level techniques with ensemble methods into a single algorithm to produce a better solution to address Class Imbalance Problem.

In this research paper, we have reviewed different Oversampling strategies such as SMOTE, ADASYN, Borderline-SMOTE and Safe Level-SMOTE to handle the

978-1-5090-6367-3/17/$31.00 ©2017 IEEE

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.

imbalance dataset with different prediction models. Three different classification techniques are used which are Naïve Bayes, Support Vector machine and Nearest neighbor over six datasets to evaluate different performance measures.

The rest of the paper is organized as follows: Section 2 describes the Oversampling techniques used. Section 3 describes the evaluation metrics used to compare the performance of the different classifiers. Section 4 presents the Dataset analysis and Section 5 presents the results. Section 6, concludes the paper.

II. Oversampling Techniques Used

This section presents a study of four oversampling techniques i.e. SMOTE, ADASYN, Borderline-SMOTE and Safe Level-SMOTE.

A. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE an over-sampling approach was designed by Chawla et al. in 2002 [5]. Samples are synthetically generated in the minority class rather than replacement of existing samples which leads to over-fitting problem. To overcome the over-fitting issue and to improve the accuracy SMOTE algorithm was proposed [6]. This technique is used to generate artificial minority examples along the line segments joining the minority samples and its 'k' minority class nearest neighbors. Based upon the rate of oversampling required, the neighbors from the 'k' nearest neighbors are randomly chosen. One of the shortcoming of SMOTE algorithm is the over generalization of the minority class space without considering majority class which may increase the overlapping between classes [7,8].

B. Adaptive Synthetic Sampling Approach (ADASYN)

Adaptively generating minority data samples according to its distributions is the basis of the approach known as ADASYN [9]. Harder to learn minority class samples are used for generating more synthetic data than the samples which are easier to learn; which in turn helps reduce the learning bias introduced originally due to imbalance data distribution. In SMOTE algorithm the numbers of synthetic samples generated for each minority class is the same whereas in ADASYN algorithm density distribution is used to automatically decide the number of synthetic samples that are needed to be generated for each minority class sample. The basic functioning of the algorithm is that it assigns weights to different minority class samples in order to generate different amounts of synthetic data for each sample.

C. Borderline SMOTE

The borderline and nearby examples are more important for classification as they are more prone to misclassification in comparison to the ones far away from the borderline. Instances far from the borderline usually don't contribute much in classification process, so variants of borderline-SMOTE are designed that only strengthens or oversample borderline minority instances [10]. This is achieved by the following process – First borderline minority instances are identified and then SMOTE algorithm is applied to generate synthetic samples to oversample the minority class. Two variants of borderline-SMOTE are borderline-SMOTE1 and borderline-SMOTE2.

D. Safe Level SMOTE

Safe-Level SMOTE was developed by Bunkhumpornpat, Sinapiromsaran, and Lursinsap in 2009, which assigns a safe level value to positive instances before generating synthetic samples [11]. It works according to the rule in which synthetic samples are generated closer to the largest safe level value i.e. only in safe regions. Safe level is defined as the number of minority instances in its k nearest neighbor, for each minority instance safe level is calculated before generating synthetic samples. If safe level value is 0 then the instance is considered as noise, if its value is close to k then the minority example is considered as safe i.e. belonging to safe region. This approach generates synthetic instances along the same line segment but locates them closer to a minority class than a majority class. Safe level ratio of a minority class is defined as the ratio of safe level of a positive instance to the safe level of a nearest neighbor.

III. Evaluation Metrices

Most commonly used metrics for accessing the performance of various classification algorithms are accuracy and error rate. Working with imbalanced datasets, it comes out that predictive accuracy is biased towards the majority class and is also seem to be highly sensitive to data distribution. Misclassifying the minority class has much higher error rate than misclassifying majority class and less likely to be predicted in comparison with the majority class instances. So, instead of accuracy and error rate other evaluation metrics such as sensitivity, specificity, precision, F-mean and g-mean are used in presence of imbalanced datasets [19].

In classification process, confusion matrix comprises of rows as actual class and columns as predicted class after applying classification algorithms, where TN refers to true negative the number of negative instances classified correctly as negative, FP refers to false positive the number of negative instances incorrectly classified as positive, FN refers to as false negative the number of positive instances incorrectly classified as negative and TP refers to as true positive the number of positive instances correctly classifies as positive.

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

Accuracy is defined as the ratio of correctly classified instances (true positive and true negative) to the total number of instances.

$$Accuracy = TP+TN/\ TP+FP+TN+FN \quad (1)$$

Sensitivity/Recall/TPrate is the number of positive instances correctly classified as positive, is a measure of correctness. It is the ability of a classifier in correctly classifying positive class as such.

$$Sensitivity = TP/\ TP+FN \quad (2)$$

Specificity/TNrate is the number of negative instances correctly classified as negative. It is the ability of a classifier in correctly classifying negative class as such.

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.

$$Specificity = TN/TN+FP \quad (3)$$

Precision is the measure of determining how many instances classified as positive are actually positive, it is a measure of exactness. It tells how well a classifier removes negative class being misclassified as positive class.

$$Precision = TP/TP+FP \quad (4)$$

These metrics are not able to completely evaluate the performance of classification algorithms so other metrics are designed including all these basic metrics.

F-measure, also known as F-score or F-measure is a typical metric for binary classification, which can be interpreted as a weighted average of the precision and recall, where greater σ gives higher weights on precision and recall. In the balanced case, σ is set to be one i.e. precision and recall are equally weighted. This yields to the harmonic mean between the precision and recall. σ is used to set the importance of precision and recall. Precision tells what percent of positive predictions were correct and Recall defines what percent of positive cases a classifier caught. σ=0.5 gives equal importance to both precision and recall.

$$F\text{-}measure = (1+\sigma^2) * Precision * Recall / \sigma^2 * Recall * Precision \quad (5)$$

G-mean or geometric mean takes the mean of sensitivity and specificity both i.e. assesses the degree of biasness in terms of ratio of positive class accuracy and negative class accuracy. Low G mean score implies that classifier is biased towards one class.

$$G\text{-}Mean = \sqrt{Sensitivity * Specificity} \quad (6)$$

ROC (Receiver Operating Characteristics) ROC curve is obtained by plotting the true positive rate on y-axis and false positive rate on x-axis. The objective of ROC curve is to select the best threshold value by varying the threshold to improve the performance of the classifier. A good classification model should yield points near the upper left coordinates. It represents a trade-off between benefits (true positives) and costs (false positives) of classification in regards to data distribution. Each point in the ROC space represents a prediction result from a confusion matrix. It can also be represented as sensitivity versus (1-specificity). AUC is the area under the ROC curve which shows the performance of the classifiers [20, 21]. AUC is the arithmetic mean of TPrate and TNrate. It is also similar to the probability that a classifier model will rank a randomly selected positive sample higher than a randomly selected negative sample.

$$AUC = (TPrate + TNrate) / 2 \quad (7)$$

IV. Dataset Analysis

In this paper we have considered six real world datasets which are described by the total number of instances in the dataset, number of majority instances, number of minority instances, Imbalance ratio (IR) and with the number of attributes used in the dataset in the following table:

Datasets	Total Instance	Attributes	Minority Class #	Majority Class #	IR
Diabetes (PID)	768	9	268	500	0.53
Wisconsin	699	11	241	458	0.52
German	1000	25	300	700	0.42
Heart	270	14	120	150	0.8
Ionosphere	351	35	126	225	0.56
Spam Base	4601	58	1813	2788	0.65

These Datasets are available on UCI Machine Repository.

Pima Indian Diabetes Dataset - Class attribute contains two variable represented as '0' [negative diabetes instances] or '1' [positive diabetes instances]. The dataset contains 268 minority class instances as positive diabetes cases and 500 instances as majority class.

Breast Cancer Wisconsin Dataset- Represents class variable '2' as benign i.e. non-cancerous sample or '4' as malignant i.e. cancerous sample. The dataset contains 241 minority class instances representing class benign and 458 majority class instances as malignant class.

German Credit Dataset - The class attribute is represented by two class variables which are '1' as good and '2' as bad class. The dataset contains 700 majority class instances as Bad and 300 minority class instances as Good.

Statlog Heart Dataset – The class attributes are represented as '1' [absence of heart disease] and '2' [presence of heart disease]. The dataset contains 120 minority class instances as presence and 150 majority class instances as absence.

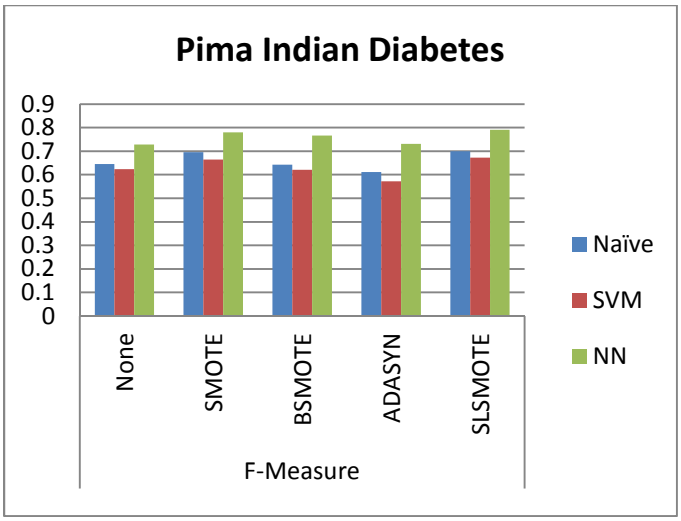
Ionosphere Dataset – The class attributes are represented as 'b' as bad radar and 'g' as good radar. The dataset contains 225 majority instances as good radar and 126 minority instances as bad radar.

Spam Base Dataset – The class attributes are represented by two class variables which are '1' as spam and '0' as solicited email. The dataset contains 2788 majority instances as solicited email and 1813 minority instances as spam.

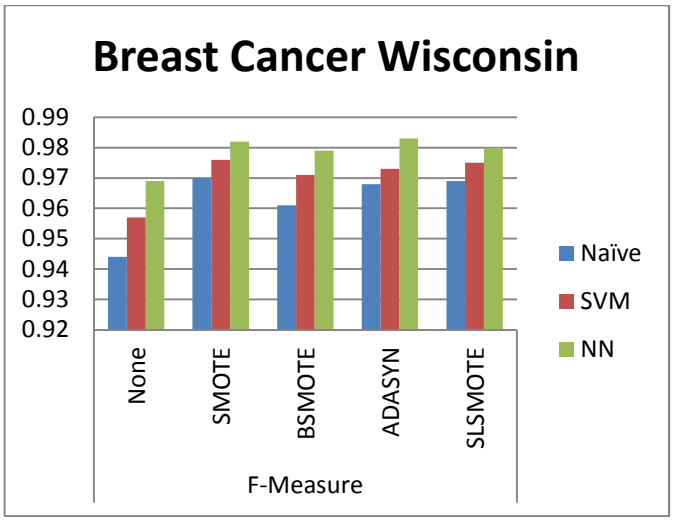
V. Experimental Results

This section presents the result of performance metrics of various oversampling methods on six different datasets. In this section we also provide Bar chart representing the F-measure value of minority class when oversampling techniques are applied on different datasets with Naïve Bayes, Support Vector Machine and Nearest Neighbor. Table I, II and III demonstrate the performance of SMOTE, ADASYN, Borderline SMOTE and Safe Level SMOTE. In addition to this we have also provided the performance of Naïve Bayes, SVM and Nearest Neighbor on the original imbalanced data sets. For all the oversampling techniques the number of nearest neighbor and oversampling rate is set to K=5 and N=100 respectively. Performance results for each method across different evaluation metrics is showed along with the winning times. The best results are highlighted.

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.



(a)



(b)

TABLE I

Naïve Bayes Classifier on different dataset

Naive Bayes								
Dataset	Methods	Overall Accuracy	TP Rate	Specificity/ TN Rate	Precision	F-Measure	G Mean	ROC Area
Pima India Diabetes	None	**0.763**	0.616	0.842	0.676	0.645	0.720	0.825
	SMOTE	0.723	0.612	0.842	**0.806**	0.696	0.718	0.823
	BSMOTE	0.727	0.573	0.842	0.729	0.642	0.695	0.817
	ADASYN	0.682	0.513	0.842	0.755	0.611	0.657	0.789
	SLSMOTE	0.726	**0.617**	0.842	0.805	**0.699**	**0.721**	**0.831**
Breast Cancer Wisconsin	None	0.960	0.971	0.954	0.918	0.944	0.962	0.987
	SMOTE	**0.969**	**0.983**	0.954	**0.958**	**0.970**	**0.968**	**0.988**
	BSMOTE	0.963	0.974	0.954	0.948	0.961	0.964	0.987
	ADASYN	0.967	0.981	0.954	0.956	0.968	0.967	0.987
	SLSMOTE	0.968	0.981	0.954	0.957	0.969	0.967	0.987
Statlog (Heart) Dataset	None	0.852	0.808	0.887	0.851	0.829	0.847	0.916
	SMOTE	0.831	0.796	0.887	0.918	0.853	0.840	0.908
	BSMOTE	0.854	0.824	0.887	0.885	0.853	0.855	0.916
	ADASYN	0.843	0.794	0.887	0.860	0.825	0.839	0.907
	SLSMOTE	**0.853**	**0.832**	0.887	**0.921**	**0.874**	**0.859**	**0.921**
Ionosphere	None	0.829	0.865	0.809	0.717	0.784	**0.836**	**0.940**
	SMOTE	0.820	0.829	0.809	0.829	0.829	0.819	0.938
	BSMOTE	0.818	0.831	0.809	0.756	0.792	0.820	0.924
	ADASYN	0.763	0.719	0.809	0.794	0.755	0.763	0.870
	SLSMOTE	**0.835**	**0.859**	0.809	**0.832**	**0.845**	0.833	0.931
Spam Base Dataset	None	0.795	0.956	0.691	0.668	0.786	0.813	0.941
	SMOTE	0.844	**0.962**	0.691	0.802	**0.875**	0.815	0.945
	BSMOTE	0.807	0.956	0.691	0.707	0.813	0.813	0.941
	ADASYN	0.820	0.955	0.691	0.747	0.838	0.812	0.938
	SLSMOTE	**0.843**	0.961	0.691	**0.800**	0.873	**0.815**	**0.946**
German	None	**0.766**	0.533	0.866	0.630	0.578	0.679	0.801
	SMOTE	0.714	0.537	0.866	0.774	0.634	0.682	0.748
	BSMOTE	0.727	0.518	0.866	0.719	0.603	0.670	0.800
	ADASYN	0.678	0.483	0.866	**0.777**	0.596	0.646	0.788
	SLSMOTE	0.720	**0.545**	0.866	0.773	**0.639**	**0.687**	**0.819**
Winning Times	None	2	0	-	0	0	1	1
	SMOTE	1	2	-	2	2	1	1
	BSMOTE	0	0	-	0	0	0	0
	ADASYN	0	0	-	1	0	0	0
	SLSMOTE	**3**	**4**	-	**3**	**4**	**4**	**4**

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.

TABLE II

SVM Classifier on different datasets

SVM								
Dataset	Methods	Overall Accuracy	TP Rate	Specificity/ TN Rate	Precision	F-Measure	G Mean	ROC Area
Pima India Diabetes	None	**0.775**	0.534	0.904	0.749	0.623	0.695	0.719
	SMOTE	0.692	0.541	0.904	0.858	0.664	0.699	0.723
	BSMOTE	0.735	0.508	0.904	0.797	0.621	0.678	0.706
	ADASYN	0.679	0.441	0.904	0.813	0.572	0.631	0.672
	SLSMOTE	0.726	**0.553**	0.904	**0.859**	**0.673**	**0.707**	**0.729**
Breast Cancer	None	0.970	0.963	0.974	0.951	0.957	0.968	0.968
	SMOTE	**0.975**	**0.977**	0.974	0.975	**0.976**	0.975	**0.975**
	BSMOTE	0.973	0.972	0.974	0.969	0.971	0.973	0.973
	ADASYN	0.973	0.972	0.974	0.974	0.973	0.973	0.973
	SLSMOTE	0.974	0.975	0.974	**0.975**	0.975	**0.975**	0.974
Statlog (Heart) Dataset	None	**0.852**	0.817	0.880	0.845	0.831	0.848	0.848
	SMOTE	0.846	0.804	0.880	0.915	0.856	0.841	0.842
	BSMOTE	0.848	0.818	0.880	0.878	0.847	0.848	0.849
	ADASYN	0.843	0.802	0.880	0.854	0.827	0.840	0.841
	SLSMOTE	0.850	**0.832**	0.880	**0.917**	**0.872**	**0.856**	**0.856**
Ionosphere	None	**0.915**	0.802	0.978	0.953	0.871	0.886	0.890
	SMOTE	0.885	0.802	0.978	0.976	0.880	0.886	0.890
	BSMOTE	0.901	0.794	0.978	0.962	0.870	0.881	0.886
	ADASYN	0.837	0.701	0.978	0.970	0.814	0.828	0.840
	SLSMOTE	0.898	**0.827**	0.978	**0.976**	**0.895**	**0.899**	**0.902**
Spam Base Dataset	None	**0.908**	0.838	0.953	0.920	0.877	0.894	0.896
	SMOTE	0.896	0.853	0.953	0.959	0.903	0.902	0.903
	BSMOTE	0.900	0.833	0.953	0.932	0.880	0.891	0.893
	ADASYN	0.885	0.815	0.953	0.943	0.874	0.881	0.884
	SLSMOTE	0.904	**0.866**	0.953	**0.959**	**0.911**	**0.908**	**0.910**
German	None	**0.788**	0.513	0.906	0.700	0.592	0.682	0.710
	SMOTE	0.718	0.498	0.906	0.819	0.620	0.672	0.702
	BSMOTE	0.743	0.499	0.906	0.779	0.608	0.672	0.702
	ADASYN	0.704	0.495	0.906	**0.835**	0.622	0.670	0.700
	SLSMOTE	0.735	**0.532**	0.906	0.825	**0.647**	**0.694**	**0.719**
Winning Times	None	**5**	0	-	0	0	0	0
	SMOTE	1	1	-	0	1	0	1
	BSMOTE	0	0	-	0	0	0	0
	ADASYN	0	0	-	1	0	0	0
	SLSMOTE	0	**5**	-	**5**	**5**	**6**	**5**

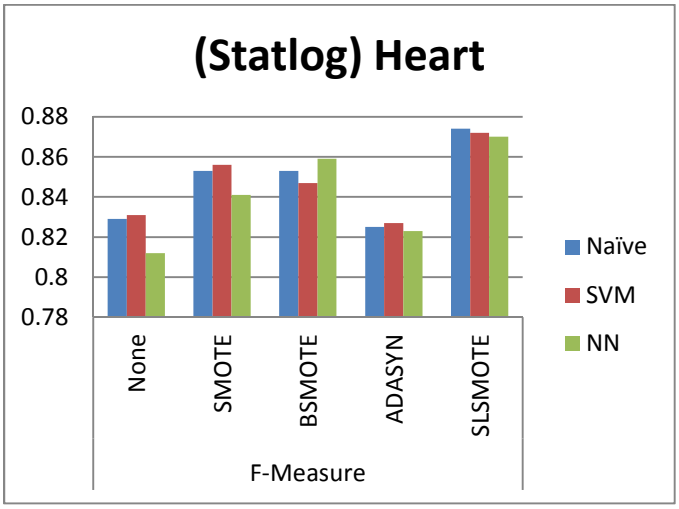
TABLE III

Nearest Neighbor Classifier on different datasets

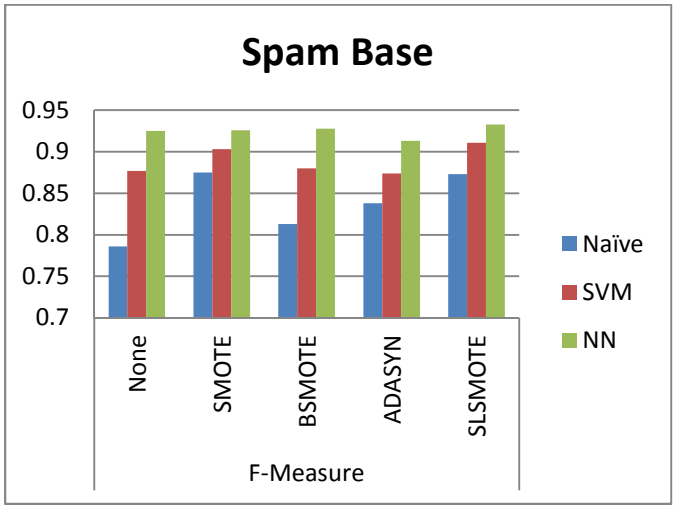
Nearest Neighbor								
Dataset	Methods	Overall Accuracy	TP Rate	Specificity/ TN Rate	Precision	F-Measure	G Mean	ROC Area
Pima India Diabetes	None	**0.823**	0.679	0.900	0.784	0.728	0.782	0.896
	SMOTE	0.796	0.700	0.900	0.882	0.780	0.794	0.902
	BSMOTE	0.789	0.682	0.900	0.875	0.766	0.783	0.894
	ADASYN	0.772	0.637	0.900	0.858	0.731	0.757	0.885
	SLSMOTE	0.805	**0.716**	0.900	**0.883**	**0.791**	**0.803**	**0.909**
Breast Cancer Wisconsin	None	0.978	0.979	0.978	0.959	0.969	0.978	0.998
	SMOTE	0.982	0.985	0.978	0.979	0.982	0.982	0.997
	BSMOTE	0.979	0.981	0.978	0.978	0.979	0.980	0.997
	ADASYN	**0.983**	**0.987**	0.978	**0.979**	**0.983**	**0.983**	**0.998**
	SLSMOTE	0.980	0.981	0.978	0.979	0.980	0.980	0.997
Statlog (Heart) Dataset	None	0.833	0.808	0.853	0.815	0.812	0.830	0.931
	SMOTE	0.815	0.792	0.853	0.896	0.841	0.822	0.929
	BSMOTE	0.836	0.825	0.853	0.896	0.859	0.839	0.936
	ADASYN	0.836	0.817	0.853	0.829	0.823	0.835	0.930
	SLSMOTE	**0.845**	**0.840**	0.853	**0.901**	**0.870**	**0.846**	**0.940**

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.

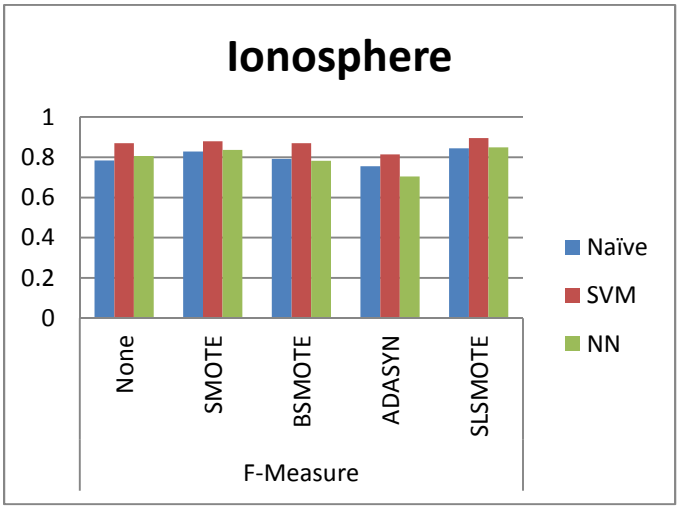
Ionosphere	None	0.880	0.690	0.987	0.967	0.806	0.825	0.989
	SMOTE	0.849	0.726	0.987	0.984	0.836	0.846	0.988
	BSMOTE	0.814	0.650	0.987	0.981	0.782	0.801	0.976
	ADASYN	0.765	0.550	0.987	0.977	0.704	0.737	0.983
	SLSMOTE	**0.861**	**0.746**	0.987	**0.984**	**0.849**	**0.858**	**0.992**
Spam Base Dataset	None	0.931	0.900	0.952	0.924	0.911	0.925	0.984
	SMOTE	0.923	0.902	0.952	0.960	0.930	0.926	0.983
	BSMOTE	0.925	0.905	0.952	0.960	0.931	0.928	0.983
	ADASYN	0.915	0.876	0.952	0.945	0.910	0.913	0.978
	SLSMOTE	**0.931**	**0.915**	0.952	**0.961**	**0.937**	**0.933**	**0.986**
German	None	**0.807**	0.577	0.906	0.724	0.642	0.723	0.874
	SMOTE	0.762	0.595	0.906	0.844	0.698	0.734	0.877
	BSMOTE	0.765	0.589	0.906	0.833	0.690	0.730	0.875
	ADASYN	0.750	0.589	0.906	**0.858**	0.699	0.730	0.879
	SLSMOTE	0.771	**0.610**	0.906	0.844	**0.708**	**0.743**	**0.879**
Winning Times	None	2	0	-	0	0	0	0
	SMOTE	0	0	-	0	0	0	0
	BSMOTE	0	0	-	0	0	0	0
	ADASYN	1	1	-	2	1	1	1
	SLSMOTE	**3**	**5**	-	**4**	**5**	**5**	**5**



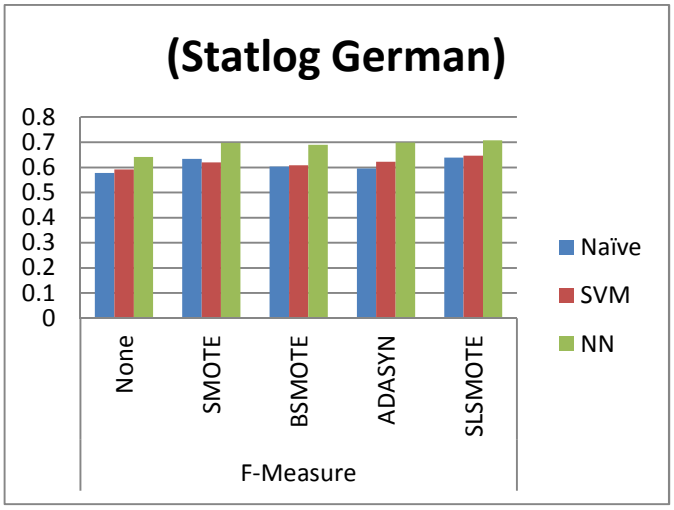
(c)



(e)



(d)



(f)

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.

Fig (a), (b), (c), (d), (e) and (f) represent the F-value for minority class when oversampling techniques are applied on the Datasets Pima Indian Diabetes, Brest Cancer Wisconsin, (Statlog) Heart, Ionosphere, Spam Base and (Statlog) German respectively with Naïve Bayes, SVM and Nearest Neighbor classifier. None represents the original dataset, SMOTE denotes Synthetic Minority Oversampling, BSMOTE denotes Borderline-SMOTE, ADASYN denotes Adaptive Synthetic Minority Oversampling and SLSMOTE denotes Safe Level SMOTE.

Safe Level SMOTE technique outperforms the other methods according to the winning times. Safe Level SMOTE also provides with best performance result in terms of F-measure and G-mean. This signifies that the algorithm does not produce biased results towards one class. As Safe Level SMOTE generates more minority instances near safe level region, it achieves better performance results than other algorithms.

VI. CONCLUSION

In this paper we have addressed the Class Imbalance Problem (CIP) and has focused on implementing four oversampling techniques in Matlab – Synthetic Minority Oversampling Technique (SMOTE), Adaptive Synthetic Sampling approach (ADASYN), Borderline-SMOTE, Safe-Level SMOTE. The performances of the oversampling techniques used to deal with the CIP are then compared with four publicly available datasets on UCI machine learning repository using four popular classification models – Naïve Bayes, Support Vector Machine (SVM) and Nearest Neighbor (NN). Different evaluation metrics used for analyzing the performance are Overall Accuracy, Sensitivity, Specificity, Precision, F-measure, gmean and ROC area i.e. Area under the curve (AUC) value. Based on the results Safe Level SMOTE outperforms the other methods, it provides best performance in f-measure and g-mean in most of the datasets. As Safe Level SMOTE generates minority instances more around larger safe level, it achieves a better accuracy performance than SMOTE, ADASYN and Borderline SMOTE.

REFERENCES

[1] G. Weiss, “Mining with Rarity: A Unified Framework”, SIGKDD Explorations, Vol. 6, No. 1, pp. 7-19, 2004.

[2] A. Akkartal, F. Sunar, “THE USAGE OF RADAR IMAGES IN OIL SPILL DETECTION”, ISPRS, Vol. XXXVII. Part B8. Beijing 2008

[3] T. Jo and N. Japkowicz, ‘‘Class imbalances versus small disjuncts’’, ACM SIGKDD Explorations Newslett., vol. 6, no. 1, pp. 40–49, 2004.

[4] S.Maheshwari, J.Agrawal and S.Sharma, ‘‘New approach for classification of highly imbalanced datasets using evolutionary algorithms’’, Int. J. Sci. Eng. Res., vol. 2, no. 7, pp. 1–5, 2011.

[5] N. V. Chawla et al., “SMOTE:Synthetic Minority Over Sampling Technique”, Journal of Artificial Intelligence Research, Vol. 16, pp 321-357, 2002.

[6] N. V. Chawla, ‘‘Data mining for imbalanced datasets: An overview,’’ Data Mining and Knowledge Discovery Handbook. Springer, pp. 853–867, 2005.

[7] J. A. Saez et al., “Managing Borderline and Noisy examples in Imbalanced Classification by combining SMOTE with Ensemble Filtering”, IDEAL2014, Springer LNCS, Vol. 8669, pp. 61-68, 2014.

[8] B.X. Wang and N. Japkowicz, “Imbalanced Data Set Learning with Synthetic Samples”, Proc. IRIS Machine Learning Workshop, 2004.

[9] H. He, Y. Bai, E. A. Garcia, and S. Li, ‘‘ADASYN: Adaptive synthetic sampling approach for imbalanced learning,’’ in Proc. IEEE Int. Joint Conf. Neural Netw., IEEE World Congr. Comput. Intell., pp. 1322–1328, Jun. 2008.

[10] H. Han, W. Wang, and B. Mao, “Borderline-SMOTE: A new oversampling method in Imbalanced Data-sets Learning”, In. ICIC 2005 LNCS, Springer, Heidelberg, Vol. 3644, pp. 878-887, 2005.

[11] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, “Safe-Level-SMOTE: Safe Level- Synthetic MI Over-Sampling Technique for handling the Class Imbalance Problem”, PADD2009, Springer LNAI, Vol. 5476, pp. 475-482, 2009.

[12] Guo, X., et al. “On the class imbalance problem”, Natural Computation, ICNC'08. IEEE, Fourth International Conference, 2008.

[13] Vaishali, G., “An Overview Of Classification Algorithms For Imbalanced Datasets”, Int Journal of Emerging technology and Advanced Engineering, 2(4), 2012.

[14] K. P. N. V. Satyashree, and J. V. R. Murthy, “An Exhaustive Literature Review on Class Imbalance Problem”, Int. Journal of Emerging Trends and Technology in Computer Science Vol. 2, No.3, pp 109-118, 2013.

[15] G. E. A. P. A Batista et al., “A study of the behaviour of several methods for balancing machine learning training data”, SIGKDD Expl. Newl., Vol 6, No. 1, pp 20-29, 2004.

[16] G. E. Batista, R. C. Prati, and M. C. Monard, ‘‘A study of the behavior of several methods for balancing machine learning training data’’, ACM SIGKDD Explorations Newslett., vol. 6, no. 1, pp. 20–29, 2004.

[17] Nitesh V.Chawla, Nathalie Japkowicz and Aleksander Kolcz, “Editorial: Special Issue on Learning from Imbalanced Data Sets”, SIGKDD Explorations 6 (1), pp 1-6, (2004)

[18] J. Huang and C.X. Ling, “Using AUC and Accuracy in Evaluating Learning Algorithms”, IEEE Transactions on Knowledge and Data Engineering, Vol 17, No. 3, March 2005

[19] J. A. Hanley, and B. J. McNeil, “A method of comparing the areas under receiver operating characteristic curves derived from the same cases”, Radiology, 148(3), pp 839-843, 1983.

[20] A. Amin, S. Anwar, “Comparing Oversampling Technique to Handle the Class Imbalance Problem: A Customer Churn Prediction Case Study”, IEEE Access.

[21] A. Ali, S.M. Shamsuddin and A. L. Ralescu, “ Classification with class imbalance problem: A Review”, Int. J. Advance Soft Compu. Appl, Vol. 7, No. 3, November 2015.

Authorized licensed use limited to: University of Melbourne. Downloaded on August 31,2021 at 04:23:51 UTC from IEEE Xplore. Restrictions apply.