Chapter – 4

A Cluster-based solution for Imbalanced Data

1.1 Introduction

There are few quality issues, that negatively influence the performances of the classifier, such as imbalance distribution of the data(between-class imbalance and within-class imbalance), high dimensional ,noisy and incomplete data. Learning from imbalanced data is one of the top 10 challenging problems in data mining.

Imbalance class distribution became noticeable with the application of data mining techniques in real-world applications. Japkowicz, et al (2000) had grabbed attention for the first time in the workshop on their work “Learning from imbalanced datasets”. One important issue drawing the attention of the data mining community for decades was learning from imbalanced data. This issue of imbalanced data handling is also identified as an open research problem by the data mining community.

Rapid technological inventions in the Data mining domain develop at an extraordinary pace. Its implementation of real-world problems in diverse areas - arises the problem of imbalanced nature of data. It has been considered as one of the TOP 10 challenging problems in data mining (Wu et. al., 2007). Many researchers have accepted the challenge and proposed their solutions. These solutions fall into two categories: Data level or Algorithmic level. At data level they proposed random oversampling with replacements, random oversampling without replacements, directed oversampling, directed under-sampling, oversampling with the informed generation of new samples, a combination of under and oversample and Feature Selection for imbalance datasets At algorithmic level they describe

cost adjusting function, adjusting decision threshold, reorganization based learning, a mixture of expert approaches.

1.2 Preliminaries and basic definitions

1.2.1 Imbalance Data

Imbalance data classification problem is a challenging problem as it affects the performance of standard classifiers so drastically due to the unequal distribution of data among classes. In imbalance data, some classes have a large number of instances and the other has a very less number of instances. A large number of examples belong to one class called Majority class and very few examples represent other classes called Minority Class. Most of the problems in real-world applications have such skewed nature of data.

Some real-world examples of Skewed data.

- Financial Fraud Detection: In this application, a very less number of transactions belong to the fraudulent transactions while the majority of truncations are legitimate.
- Network Intrusion Detection: In such applications, the dataset contains a large number of normal activities and very less number of malicious activities which are hidden in routine voluminous network connections.
- Medical Fraud Detection: Number of genuine claims are outnumbered compare to bogus cases.
- Real-Time Video Surveillance: In continuous video sequences suspicious activities are very insignificant.
- Oil Spillage: Finding Oil spill from continuous monitoring of satellite images is rare.

- Astronomy: In this application data is skewed because of the abnormal behavior is less frequent.
- Spam mail Detection: There is less number of spam mail compare to legitimate emails.
- Text Classification: In text classification data has skewed nature.
- Health Care: The people affected by rare diseases are negligible but can lead to severe consequences if classified incorrectly.

In such applications where data is skewed, we cannot deploy traditional classification models because of the following reasons.

(1) Traditional classifiers on imbalance data give weighted to the majority class only and discard minority class while building model even when the accuracy of the model is very high.
(2) Sometimes rare minority instances are treated as noise or noise can be treated as minority instances.
(3) Learning from imbalance data is not very difficult if the classes can be separated but when minority instances overlap with other regions the problem gets worse.
(4) Another challenge for learning from imbalance data is a small dataset with high dimensionality which makes it difficult for the learning model to detect rare patterns.

Fig. 4.1 depicts the imbalanced distribution of instances in the Majority and Minority class. Square symbols represent instances belonging to majority class and circles are instances belonging to the minority class. It is clearly noticeable that the majority class area is very dense compare to the minority class area.

While evaluating the performance of any classifier the impact of imbalance nature of real-world data cannot be ignored. Classifier's performance is always

biased towards the majority class and considers minority class instances as noise and do not give required weightage in the building of the model.

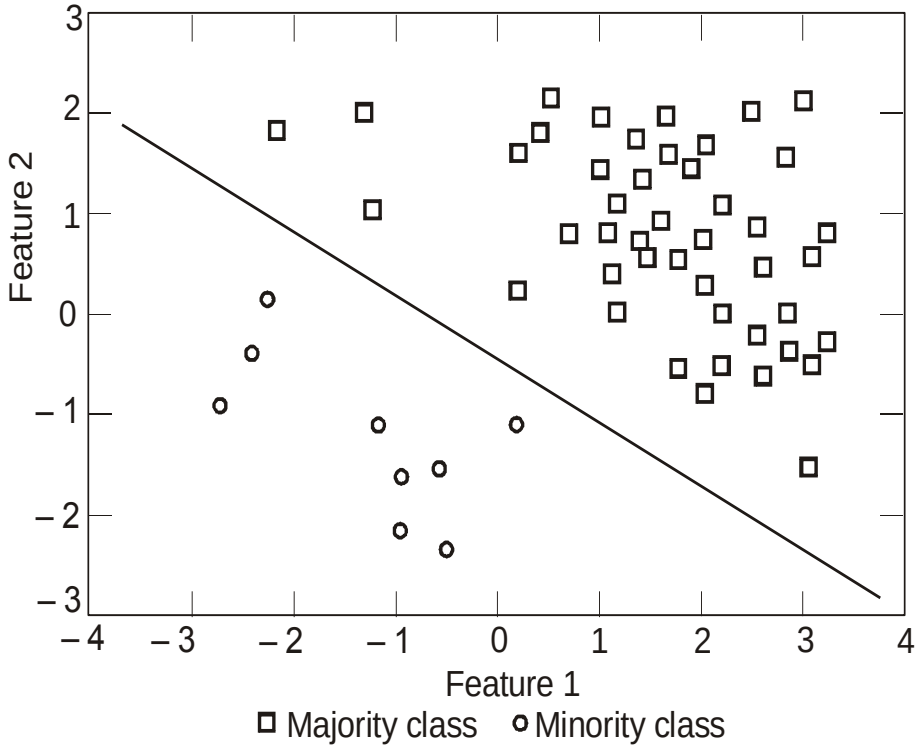

Figure 4.1 Majority and Minority instances Distribution in Imbalanced class

(Source: Chujai *et. al.*, [9]).

1.2.2 Between - class imbalance & within- class Imbalance

Between- class imbalance dataset where the number of instances representing the Majority of the class is extremely out-numbered the number of instances representing the Minority class;

Within - class imbalance dataset where a single class is composed of different sub-clusters and in these sub-clusters instances belonging to one sub-cluster are extremely outnumbered compare to other sub-clusters. Between-class imbalanced dataset problem exists in the two classes and within-class imbalanced dataset is present in a single class.

1.2.3 Imbalance Ratio

It is the proportion of minority instance over majority instance.

$$\text{Imbalance Ratio} = \frac{\text{No. of instance in Majority class}}{\text{No. of instances in Minority class}}$$

In this study IR = 0.5 and 1.0 are considered. When IR = 0.5, it means sampled majority instances will be half of the minority instance i.e. if there are 100 instances belongs to minority and 1000 from the majority, to make a final training set only 50 instances from the majority will be selected. If IR = 1 means sampled majority instances will be equal to the number of minority instance *i.e.* if there are 100 instances from minority and 1000 from majority only 100 instances from majority class will be selected to make the final training set.

1.2.4 Degree of imbalance distribution

The difference between the numbers of instances belongs to the majority class and the number of instances belongs to the minority class.

1.2.5 False-positive and false-negative

False-positive also referred to as TYPE-1 error and False Negative known as TYPE-2 errors are the undesirable outcomes of any classifier. False positives take place when the classifier is predicting it as positive which is a false case; actually, it should be predicted as a negative case. On the other hand, when the classifier is predicting it as Negative, which is a false case; actually it should be predicted as positive case, a false positive outcome will result in unnecessary treatments – e.g. while considering a medical case study - a false negative will give a false diagnosis. The false-positive outcome is very critical, where the disease is ignored can lead to the death of the patients because of no treatments.

1.2.6 Clustering

The division of data into meaningful groups is done by Clustering algorithms. The patterns in the same group are similar but dissimilar with other groups.In this experitment K-means clustering algorithm has been used.

1. K-Means

K-means method of data clustering is one of the oldest and yet very popular among the Data Miners community. One reason for its popularity is; it is data-driven, so less number of assumptions are required. It uses a greedy search strategy so it can divide large datasets into segments based on the number of clusters supplied as K. It means full convergence of clusters. It aims to minimize squared error given by

Sum of Squared Error=$\sum_{i=1}^{k} \sum_{j=1}^{ik} \mathrm{dist}(x_i, y_j)$

Distance will be calculated by Euclidean distance metric between x_i and y_j is given by

Euclidean distance=$\sqrt{\sum_{k=1}^{n} (x_{ik}, y_{ik})^2})$

1.2.7 SVM Classifier

Support Vector Machine is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data. In a high dimensional feature space, Support Vector machines use hypothesis space of a linear function. We try to achieve a plane that has the maximum margin.

1.3 Methods of handling Imbalanced Data

Fig. 4.2 displays two methods provided in the literature to tackle imbalanced class distribution.

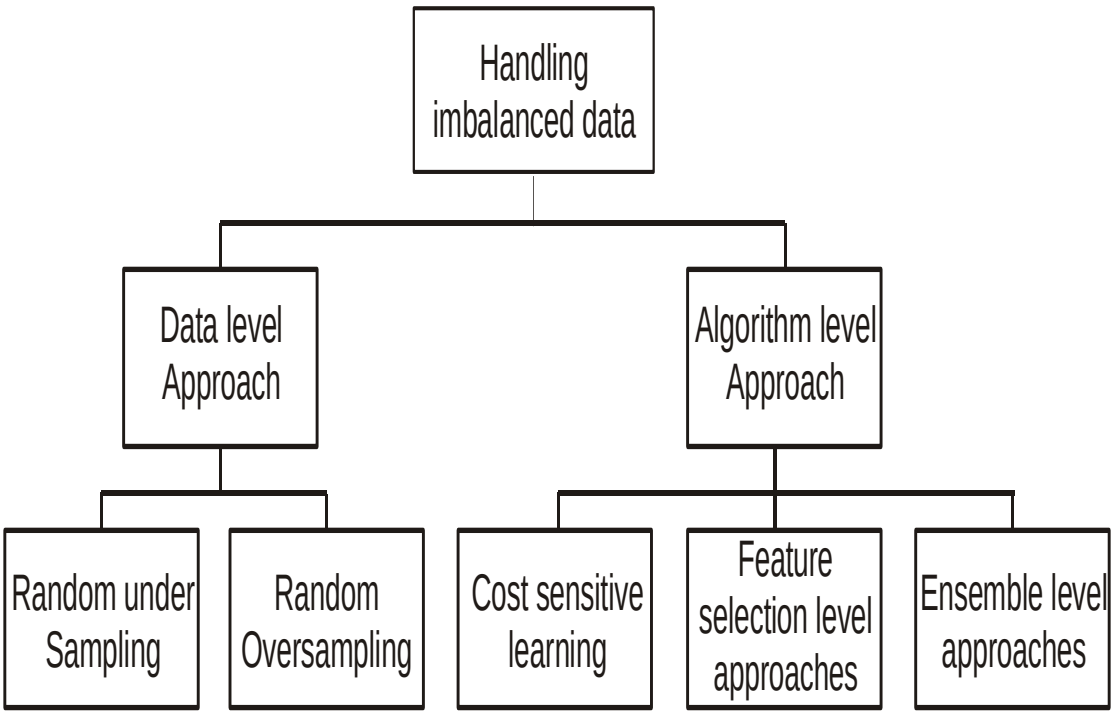

Figure 4.2 Methods to handle Imbalanced Data

(1) Data level approach: In this solution, data is modified to be applied to traditional classifiers.

(i) Random sampling Techniques

The most common sampling methods are random Oversampling and random under-sampling. Random oversampling increases the minority class instances, by randomly reproducing the minority class instances. While, Random under sampling reduces - the majority class by randomly removing some majority class instances.

Over-sampling increases training time and over-fitting .under-sampling works better compare to over-sampling in terms of both time and memory complexity. Fig. 4.3 depicts how instances are randomly selected to increase or decrease the sample size.

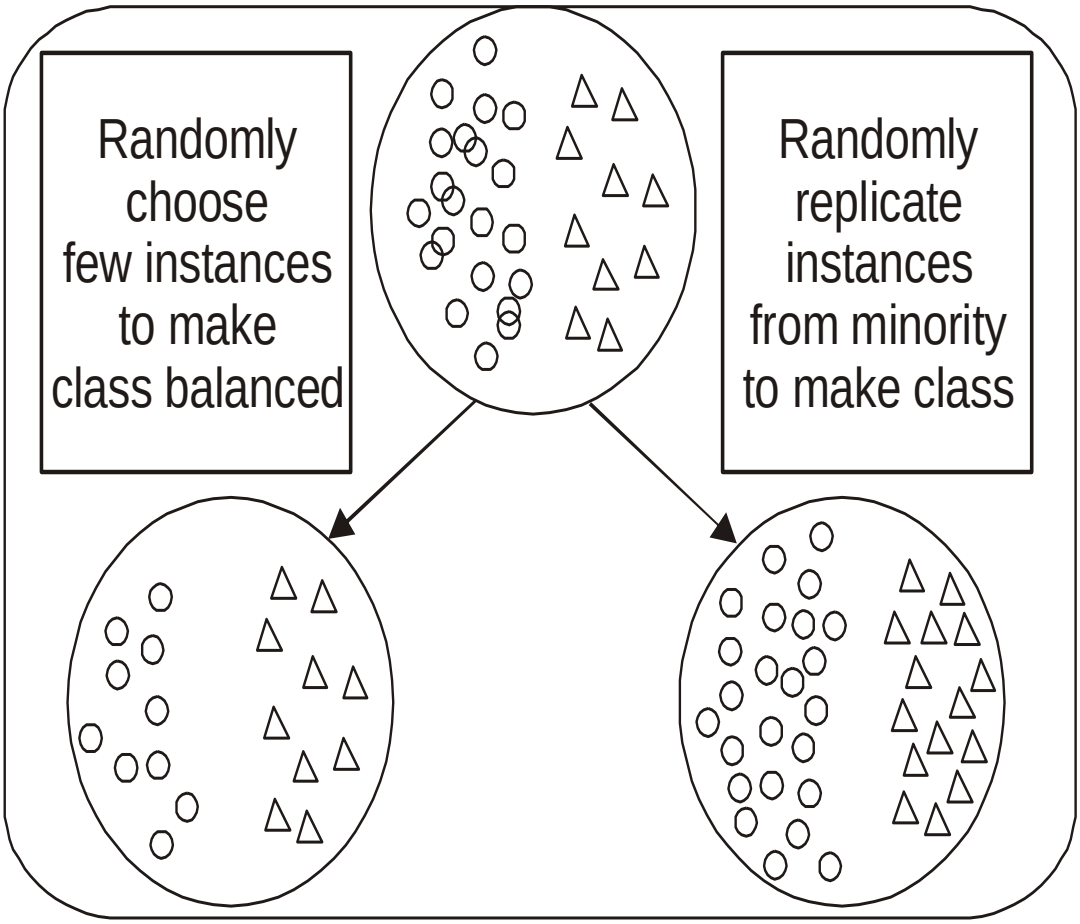

Figure 4.3 Random Under-Sampling and Random Oversampling

(ii) Synthetic Minority Oversampling Technique (SMOTE)

Chawla *et al*., (2002) proposed a Synthetic Minority Oversampling Technique (SMOTE) - remarkable research in the area of oversampling for classification of Imbalanced data is used in many applications. The feature-based similarity is used to generate synthetic instances among minority instances. This method makes traditional classifiers to enhance the decision boundary close to minority instances.

(2) Algorithm level Approach

Traditional classifiers are modified to deal with imbalanced data.

1.4 Experimental investigations

1.4.1 Datasets

A study for binary class distribution on 12 data sets openly available with different degrees of imbalance nature is conducted. Table 4.1 contains the description of the data sets used for demonstrating the effectiveness of our proposed solution on various Parameters 12 datasets that were used from UCI or KEEL repository. Several instances, no. of attributes, and degree of imbalanced distribution (imbalance ratio) of the datasets are also given.

1.4.2 Experiment Setting

The results are evaluated over 12 datasets with WEKA 3.6.9 and Orange 3.20 Data mining tools.

(1) WEKA

The WEKA workbench is a machine learning and data preprocessing tool, under GNU General Public License. WEKA, acronym is Waikato Environment for Knowledge Analysis, was developed at the University of Waikato in New Zealand. It is written in Java and can run on Linux, Windows, and Macintosh operating systems. The current stable version, 3.8.0, is compatible with Java 1.7. WEKA provides the support for the whole data mining process, viz., and preparation of the input data by data transformation and preprocessing, analyzing the data using learning schemes, and visualizing the data.

Table 4.1: Imbalanced Datasets with different degree of Imbalanced distribution

S. No.	Data Set Name	Imbalanced Ratio	No. of Instances	No. of Attributes
1.	Abalone	129.44	4174	8
2.	Cleveland-0	12.62	177	13
3.	*E. coli*-3	8.6	336	7
4.	Glass-1	1.82	214	9
5.	Haberman	2.78	306	3
6.	New-Thyroid 1	5.14	215	5
7.	Page-Blocks 0	8.79	5472	10
8.	Pima	1.87	768	8
9.	Wine Quality White	58.28	1482	11
10.	Breast Cancer Wisconsin	1.86	683	9
11.	Yeast 1	2.46	1484	8
12.	Vowel	9.98	988	13

(2) Orange

Orange is a component-based data mining and machine learning software suite, it features visual programming for explorative data analysis that is existing in the front end and helps in visualization, libraries for scripting and Python bindings. Orange has widgets, supported on Mac OS, Windows, and Linux platforms.

1.5 Proposed Cluster-Based Under-sampling

In Random Under-Sampling some instances are removed randomly. So the valuable instances may get thrown away which may contain potential information it results as inaccurate outcomes and predictions. The solution to this problem -is the integration of unsupervised learning with supervised learning. Here we are using the clustering tool for sampling. The main purpose of this method is to selectively discard majority instances from the datasets. Clustering algorithms group the similar characteristic instances in one cluster so it can have representation from the overall population. On the other hand Random over- Sampling instances are randomly replicated increasing the dataset size results in longer training time. It can be visualized using Fig. 4. Under-sampling is a technique to reduce the number of samples in the majority class, where the size of the majority class sample is, reduced from the original datasets to balance the class distribution.

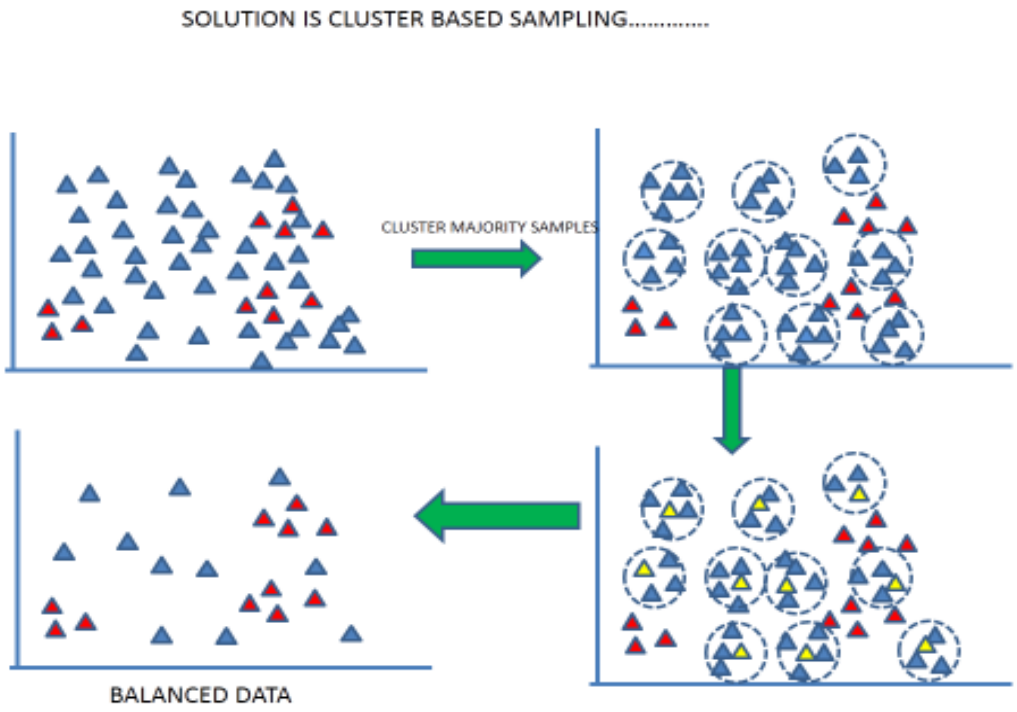

Figure 4.4 Framework for Cluster-Based sampling

1.6 Methodology

The overall process of transforming imbalanced data to balance can be divided into two phases: at first phase between–Class imbalance and Degrees of imbalanced distribution of data are resolved, by dividing the whole dataset into two classes and deal them separately throughout the process. The first class called Majority class contains instance belongs to the class containing a large number of instances and the second class called Minority class belongs to a group containing less number of instances.

In an imbalanced distribution, the majority class size is always very large compare to the minority. In the second phase, the imbalanced data is converted into a balanced form using under-sampling through the clustering method. The

proposed framework is implemented on Weka and Orange data mining tools. Initially the data is input in .Tab file format to Orange data mining tool. There whole dataset is divided into majority and minority classes. All the instances of minority and majority are compared to identify the similar instances belonging to both the classes. If similar instances are identified, the instances are removed from majority class, but not from a minority. After this outlier removal process, an updated majority class data set will be stored. Identify the most appropriate value of K using silhouette plot and build K clusters.In order to decide the imbalanced ratio for the resultant training set here in this experiment two ratio they are 0.50 and 1.0 are considered. To calculate how many instances should be selected from each cluster equation-4.1 is used. After getting the required number of instances to be selected, randomly select them from each cluster.

After this whole process cluster representative instances from every cluster will be merged with all instances of minority class instances to make an Imbalanced training set. This process will be repeated with IR = 1.

.

In order to prove the efficacy of the proposed algorithm for handling imbalanced data, performance measuring parameters are derived using Support vector machine classifier. The performance of these parameters is compared with the original dataset, IR = 0.50, and IR = 1 to identify the better performances.

The dry run of the algorithm with abalone dataset:

Dataset = Abalone
Total no. of instances = 4174
No. of Instances belonging to majority class=3813
No. of instances belonging to minority class=364
No of clusters optimized by silhouette plot K=2
No. of instances belonging to Cluster-1=2261
No. of instances belonging to Cluster-2=1552

No. of Instances to be sampled from cluster-1 is given by

SC_1^{inst} = IR×$\frac{\text{MIN}^{\text{size}}}{\text{MAJ}^{\text{size}}}$ × NC^1

For IR=.50

SC_1^{inst} =0.50×($\frac{364}{3813}$) × 2261

= 0.50 × (0.09) × 2261

= 0.05 × 215

= 107

Total number of instances selected from cluster one for IR= 0.50 is 107

SC_2^{inst} = 0.50 ×($\frac{364}{3813}$) × 1552

= 0.50 × (.09) × 1552

= 0.50×139

= 69.8

The total number of instances selected from cluster two for IR= 0.50 is 69.8.

For IR =1.0

SC_1^{inst} =1.0×($\frac{364}{3813}$) ×2261 = 215

SC_1^{inst} =1.0×($\frac{364}{3813}$) ×1552

= 139

The total number of instances selected from cluster-1 is 215.

The total number of instances selected from cluster-2 is 139.

These randomly selected instances from each cluster will be merged with minority class instances to make it balanced. majority class having less number of instances representing the overall population.

Algorithm for balancing data using Clustering as an under-sampling tool:

Step 1: Segregate whole data set into $\mathrm{MIN}^{\mathrm{inst}}$ and $\mathrm{MAJ}^{\mathrm{inst}}$

$\mathrm{MIN}^{\mathrm{size}}$ –No. of instance belongs to Minority class

$\mathrm{MAJ}^{\mathrm{size}}$ –No. of instances belongs to Majority instances.

In imbalanced datasets$\mathrm{MAJ}^{\mathrm{size}} > \mathrm{MIN}^{\mathrm{size}}$

Step 2: Removing Outliers

$\mathrm{MIN}_{\mathrm{i}}^{\mathrm{inst}}$ where i=1, 2, 3, 4……$\mathrm{MIN}^{\mathrm{size}}$

$\mathrm{MAJ}_{\mathrm{j}}^{\mathrm{inst}}$ where j=1, 2, 3, 4……..$\mathrm{MAJ}^{\mathrm{size}}$

If Distance ($\mathrm{MIN}_{\mathrm{i}}^{\mathrm{inst}}$,$\mathrm{MAJ}_{\mathrm{j}}^{\mathrm{inst}}$) =0

Remove $\mathrm{MAJ}_{\mathrm{j}}^{\mathrm{inst}}$ from $\mathrm{MAJ}^{\mathrm{size}}$ and update $\mathrm{MAJ}^{\mathrm{size}}$

Step 3: Decide the imbalanced ratio for each cluster by setting the ratio parameter from IR= {0.50, 1}

Step 4: Build clusters from majority instances using K mean algorithm. Draw a silhouette plot to find the most appropriate value

of K.

$\mathrm{MAJ}^{\mathrm{size}} = \sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{C}_{\mathrm{i}}^{\mathrm{inst}}$

Step 5: NC^{i} no. of instances in Ith cluster

$\mathrm{SC}_{\mathrm{i}}^{\mathrm{inst}}$ No. of instances to be sampled from each cluster is defined as:

$\mathrm{SC}_{\mathrm{i}}^{\mathrm{inst}} = \mathrm{IR} \times \frac{\mathrm{MIN}^{\mathrm{size}}}{\mathrm{MAJ}^{\mathrm{size}}} \times \mathrm{NC}^{\mathrm{i}}$

Step 6: Repeat the process to find no. of instances to be selected from each cluster no for i=1, 2, 3, 4……..….k

Step 7: Randomly select any instance from the given cluster

If Distance ($\mathrm{C}_1^{\mathrm{i}}$,$\mathrm{C}_{\mathrm{i}+1}^{\mathrm{i}}$) =0

Add $\mathrm{C}_1^{\mathrm{i}}$ in sampled training set and

Remove $\mathrm{C}_1^{\mathrm{i}}$ and duplicated $\mathrm{C}_{\mathrm{i}+1}^{\mathrm{i}}$ from cluster $\mathrm{C}_{\mathrm{i}}^{\mathrm{inst}}$

Repeat the process until we get the required instances from the cluster or instances of the cluster to get exhausted.

Step 8: K output clusters with selected no of quality instances. In order to get a balanced cluster output instances from each cluster will be merged with Minority instances to get a final Balanced training set.

Figure 4. 4 represents the flow of control of the proposed model. The classifier takes minority class instances as noise and does not consider them in the building model so the classifier gets biased towards the majority class. Xiong (2010) stated that the Class Imbalance, class overlap added with high dimension data makes classifying tasks complicated and challenging.

As stated above that this model is capable of giving 4 fold solutions. The first solution gives the capability of handling the different degrees of imbalanced nature of data. In our approach, we divide the majority and minority instances into separate datasets and then deals with majority class separately so in the first phase only we can handle the diverse. It balances imbalanced data using an under-sampling method. Here in our approach, we deployed a cluster-based selection to reduce the size of the majority class to make training set balanced to perform accurately on traditional classifiers. It is capable to handle between class imbalanced and within-class imbalanced nature of data. Between class imbalance distributions are solved in the first phase when we classify majority and minority instances and within-class imbalance problems can be solved by making clusters from majority class and selecting uniform cluster representatives. The outcomes of the experiments conducted on 12 datasets proved how the proposed algorithm reduces Type-1 (False positive) and Type-2 (False negative) errors which are a very serious concern while working on medical sophisticated datasets.

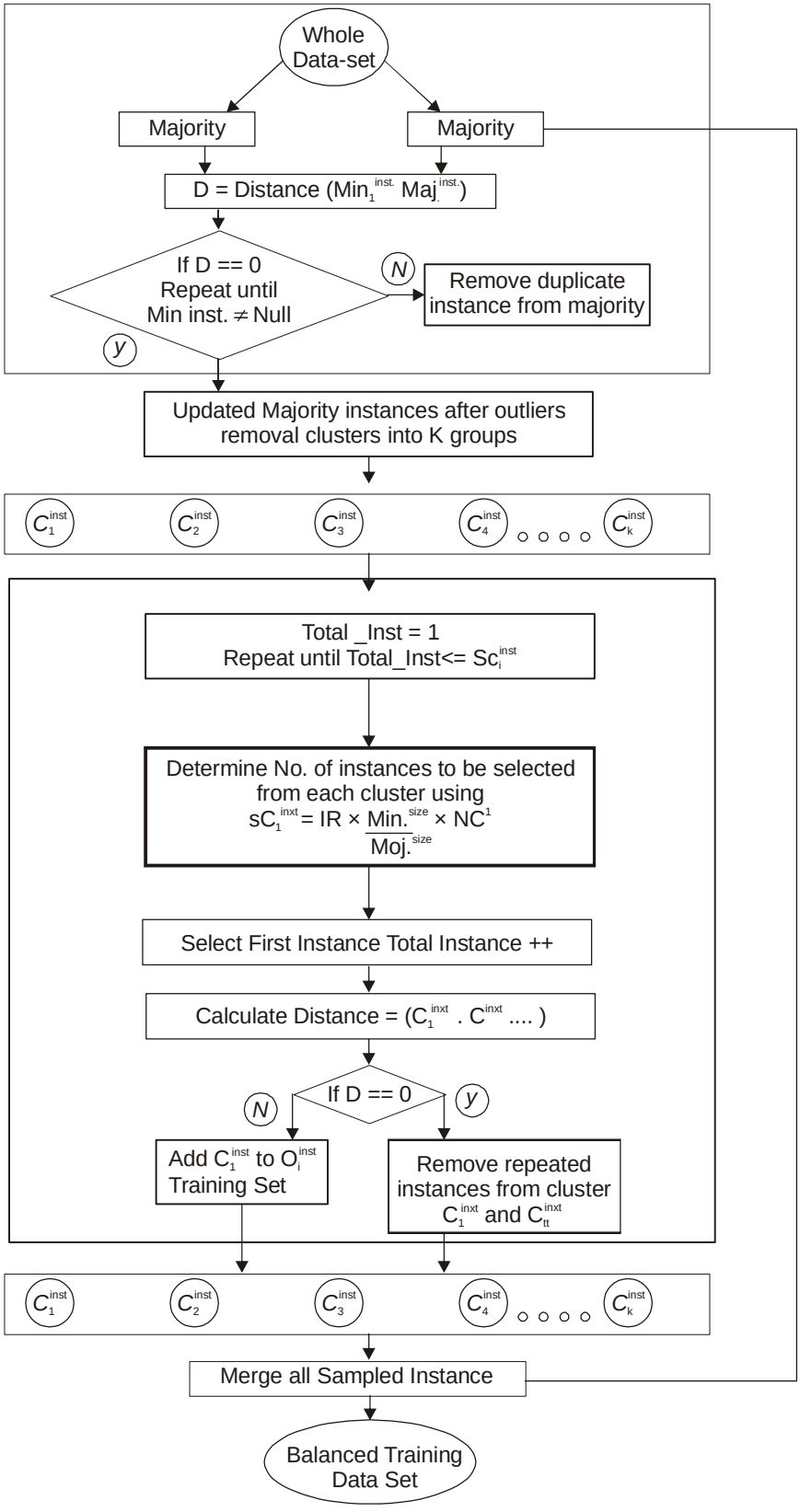

Figure 4.5 Flow Chart for the proposed Algorithm

Table 4.2 comparative analysis of SVM classifier on different 12 datasets for original imbalance data of different degrees

S. No.	Data set name	Original data set					Balanced with IR = 0.50					Balanced with IR =1				
		ACC	F-mea	FP	Pre.	ROC	ACC	F-mea	FP	Pre.	ROC	ACC	F-mea	FP	Pre.	ROC
1.	Abalone	98	0.89	0.16	0.90	0.90	97	0.96	0.07	0.96	0.98	97	0.93	0.58	0.94	0.97
2.	Cleveland-0	95.9	0.97	0.38	0.9	0.80	80	0.84	0.23	0.84	0.78	96.1	0.98	0.00	1.0	0.96
3.	*E coli*-3	89.5	0.94	1.0	0.89	0.50	92.3	0.94	0.17	0.91	0.89	81.1	0.83	0.32	0.75	0.81
4.	Glass-1	63.5	0.77	1.0	0.64	0.49	78	0.86	0.59	0.77	0.74	70	0.78	0.54	0.64	0.70
5.	Haberman	73	0.84	1.0	0.73	0.50	66.9	0.80	1.0	0.66	0.50	57	0.46	0.2	0.64	0.57
6.	New-thyroid 1	92	0.95	0.45	0.91	0.77	90.1	0.93	0.31	0.87	0.84	98	0.98	0.03	0.97	0.98
7.	Page-blocks 0	93	0.93	0.59	0.93	0.70	84	0.88	0.18	0.90	0.83	85	0.95	0.08	0.90	0.85
8.	Pima	77.3	0.62	0.10	0.74	0.72	77.3	0.84	0.49	0.78	0.70	71.7	0.70	0.04	0.73	0.75
9.	Wine quality white	98.3	0.99	1.0	0.98	0.50	62	0.76	1.0	0.65	0.46	69.3	0.70	0.33	0.69	0.69
10.	Breast cancer Wisconsin	96.9	0.97	0.03	0.98	0.96	98.3	0.99	0.02	0.99	0.78	97	0.97	0.03	0.97	0.77
11.	Yeast 1	74.3	0.84	0.81	0.74	0.57	80	0.89	0.23	0.83	0.80	76	0.85	0.31	0.75	0.68
12.	Vowel	95.9	0.73	0.50	0.91	0.80	90	0.84	0.11	0.92	0.93	96	0.70	0.23	0.89	0.74

Table-4. 2 presents a comparative analysis of the performance of SVM classifier on different 12 datasets for original imbalanced data of different degrees, Data with Imbalanced ration = 0.50, and data with Imbalanced ratio = 1.0. It gives collective information in order to identify which method's performance is better over others.

Table 4.3: Comparative Results for Accuracy measures with different imbalance Ratio.

S. No	Data set	Original data set	Balanced IR = 0.50	Balanced IR = 1.0
1.	Abalone	98	97	97
2.	Cleveland-0	95.9	80	96.1
3.	*E coli*-3	89.5	92.3	81.1
4.	Glass-1	63.5	78	70
5.	Haberman	73	66.9	57
6.	New-Thyroid 1	92	90.1	98
7.	Page-Blocks 0	93	84	85
8.	Pima	77.3	77.3	71.7
9.	Wine Quality White	98.3	62	69.3
10.	Wisconsin	96.9	98.3	97
11.	Yeast 1	74.3	80	76
12.	Vowel	95.9	90	96

Table-4.2 presents the accuracy of the SVM classifier for the original dataset, with an Imbalanced Ration of 0.50 and an Imbalanced ration of 1.0 on 12 different datasets. Fig. 4.6 is a pictorial representation of the accuracy of the SVM classifier for the original dataset, with an Imbalanced Ration of 0.50 and an Imbalanced

ration of 1.0 on 12 different datasets. As it is stated that accuracy is not the perfect measure for when classifying imbalanced data. In the figure-4. 6 high accuracy with few original imbalance data sets is observed but it does not prove the overall good performance of the classifier. The highest accuracy difference between original data, data with imbalance ration .50, and data with imbalance ratio 1.0 is reported with a wine quality white dataset.

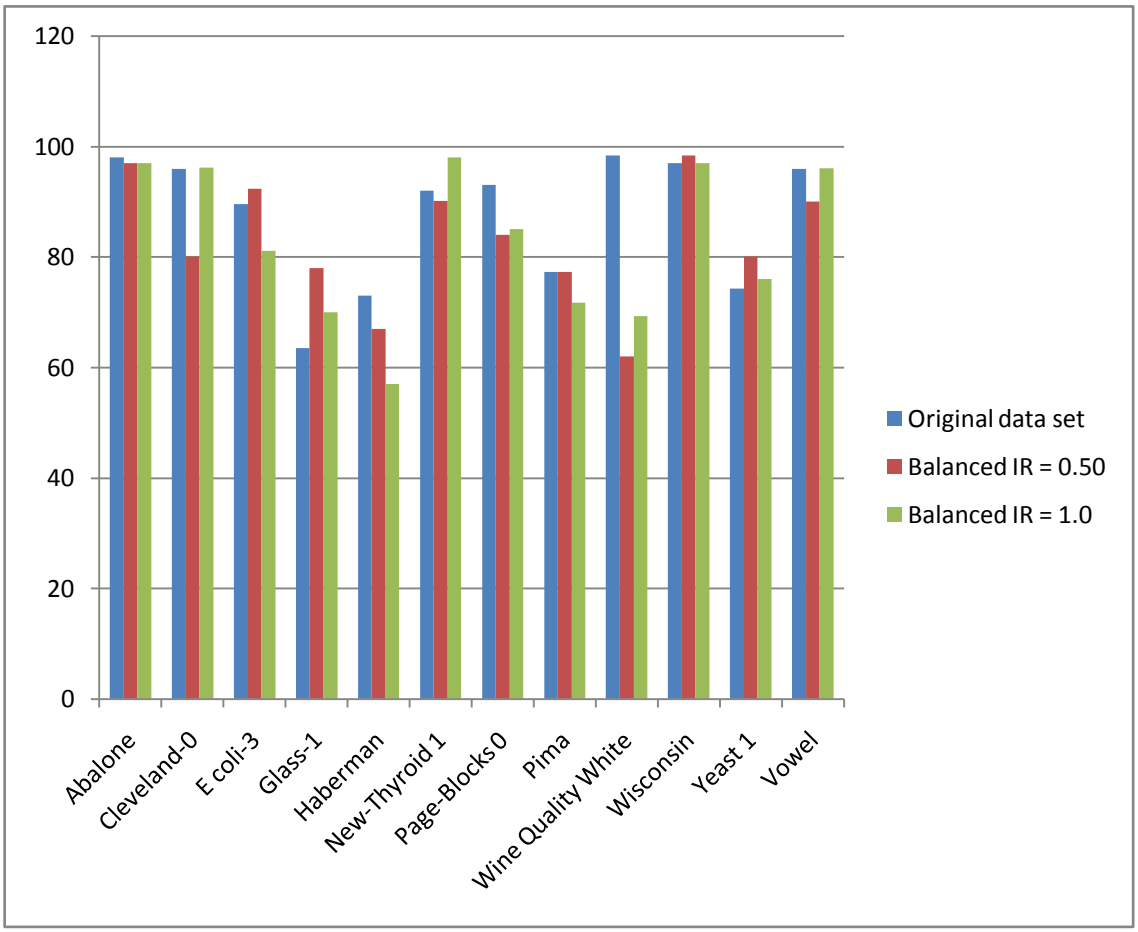

Figure 4.6 Chart for Accuracy

Table-4.4 presents the F-measure of SVM classifier for the original dataset, with an Imbalance ration of .50 and an Imbalance ration of 1.0 on 12 different datasets. The F-measure is a very important performance measuring parameter .it is a harmonic mean of precision and recall.

Table 4.4: Comparative Results for F-Measures with different imbalance ratio.

S.No	Data Set	Original data set	Balanced with IR = 0.50	Balanced with IR = 1.0
1.	Abalone	0.89	0.96	0.93
2.	Cleveland-0	0.97	0.84	0.98
3.	Ecoli-3	0.94	0.94	0.83
4.	Glass-1	0.77	0.86	0.78
5.	Haberman	0.84	0.80	0.46
6.	New-Thyroid 1	0.95	0.93	0.98
7.	Page-Blocks 0	0.93	0.88	0.95
8.	Pima	0.62	0.84	0.70
9.	Wine Quality White	0.99	0.76	0.70
10.	Wisconsin	0.97	0.99	0.97
11.	Yeast 1	0.84	0.89	0.85
12.	Vowel	0.73	0.84	0.70

Figure 4.7 is a pictorial representation of the F-measure of SVM classifier for the original dataset, with an Imbalance ration of 0.50 and an Imbalance ration of 1 on 12 datasets. It can be easily identified that improved value for F-measure with the proposed model is observed. The best F-measure for Abalone dataset is reported with .50 degree of imbalance ratio, for Cleveland-0 it is with 1.0, for Ecoli-3 it is with .50 degree, for Glass-1 dataset with .50, for Haberman with original dataset, for New-Thyroid 1 it is with 1.0 degree, for Page-Blocks 0 it is with .50, for Pima it is with .50, for Wine Quality White it is with original dataset, for Wisconsin, it is with .50, for Yeast 1 it is with .50 and for Vowel, the best F-measure is reported with IR=.50.

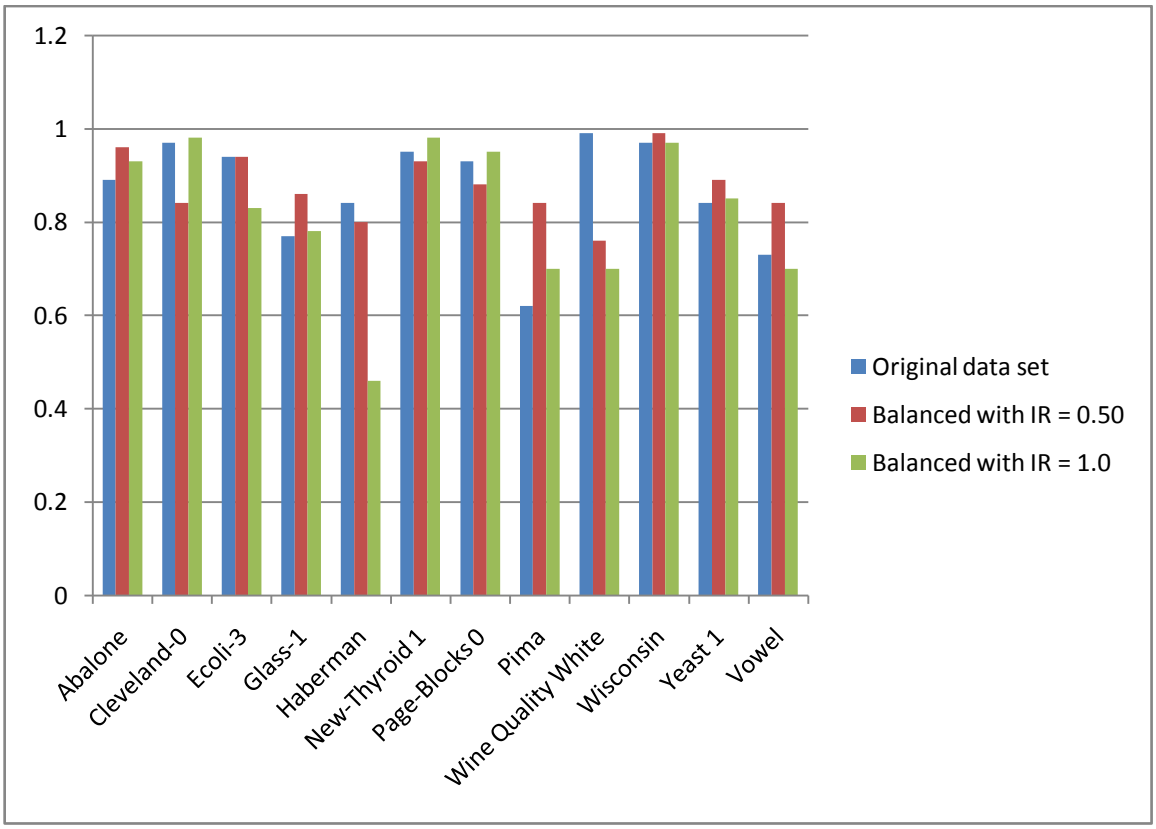

Figure 4.7 Chart for F-Measure

Table-4.5 presents the FP (False Positive) rate of SVM classifier for the original dataset, with an Imbalance ration of 0.50 and an Imbalance ration of 1.0 on 12 datasets.

Table 4.5: Comparative Results for F-P rate with different imbalance ratio.

S. No	Data set	Original data set	Balanced With IR = 0.50	Balanced with IR = 1.0
1.	Abalone	0.16	0.07	0.058
2.	Cleveland-0	0.38	0.23	0.00
3.	Ecoli-3	1	0.17	0.32
4.	Glass-1	1.0	0.59	0.54
5.	Haberman	1.0	1.0	0.2
6.	New-Thyroid 1	0.45	0.31	0.03
7.	Page-Blocks 0	0.59	0.18	0.081

8.	Pima	0.10	0.49	0.04
9.	Wine Quality White	1	1.0	0.33
10.	Wisconsin	0.15	0.02	0.03
11.	Yeast 1	0.81	0.23	0.31
12.	Vowel	0.50	0.11	0.23

Fig-4. 8 is a graph where the FP rate is on X-axis and it can be easily identified that improved value for the FP rate with the proposed model is observed.

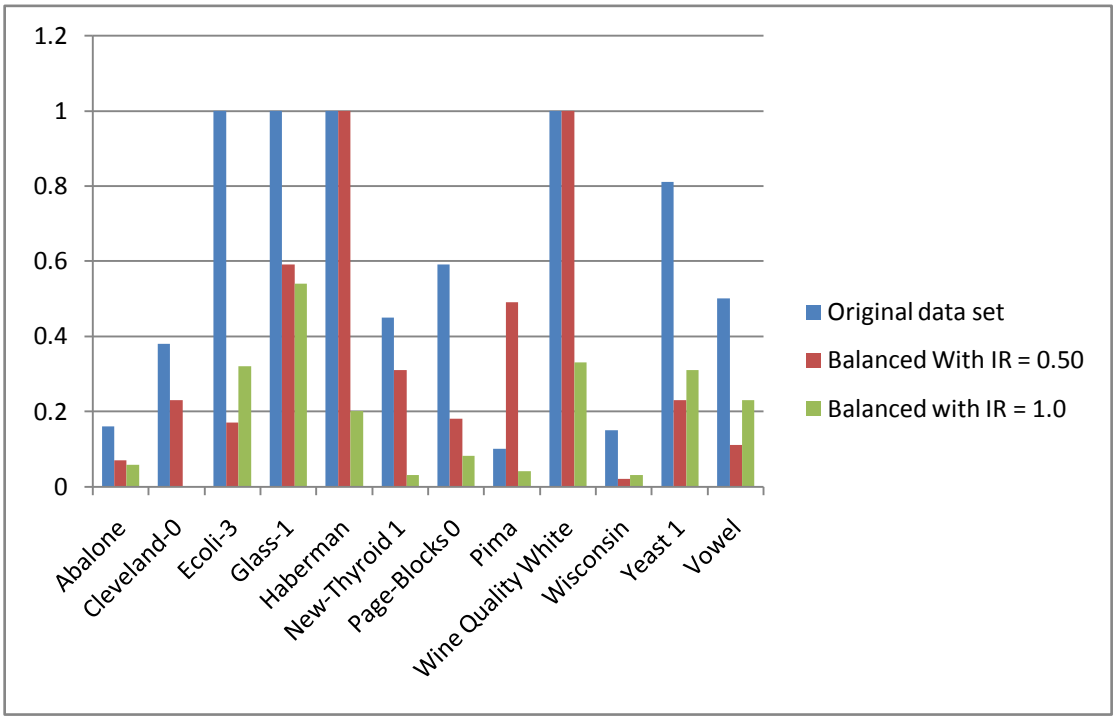

Figure 4. 8 Chart for F-P rate.

Fig. 4.8 is a pictorial representation of the F-P rate of SVM classifier for the original dataset, with Imbalance ration of 0.50 and Imbalance ration of 1 on 12 datasets.

Table 4.6 presents the precision of the SVM classifier for the original dataset, with Imbalance ration of 0.50 and Imbalance ration of 1 on 12 datasets. It can be easily identified that we are getting improved values for precision with the proposed model.

Table 4.6: Comparative Results for Precision rate with different imbalance ratio.

S. No	Data set	Original dataset	Balanced with IR = 0.50	Balanced with IR = 1.0
1.	Abalone	0.90	0.96	0.94
2.	Cleveland-0	0.9	0.84	1.0
3.	*E coli*-3	0.89	0.91	0.75
4.	Glass-1	0.64	0.77	0.64
5.	Haberman	0.73	0.66	0.64
6.	New-Thyroid 1	0.91	0.87	0.97
7.	Page-Blocks 0	0.93	0.90	0.90
8.	Pima	0.74	0.78	0.73
9.	Wine Quality White	0.98	0.65	0.69
10	Wisconsin	0.98	0.99	0.97
11	Yeast 1	0.74	0.83	0.75
12	Vowel	0.91	0.92	0.89

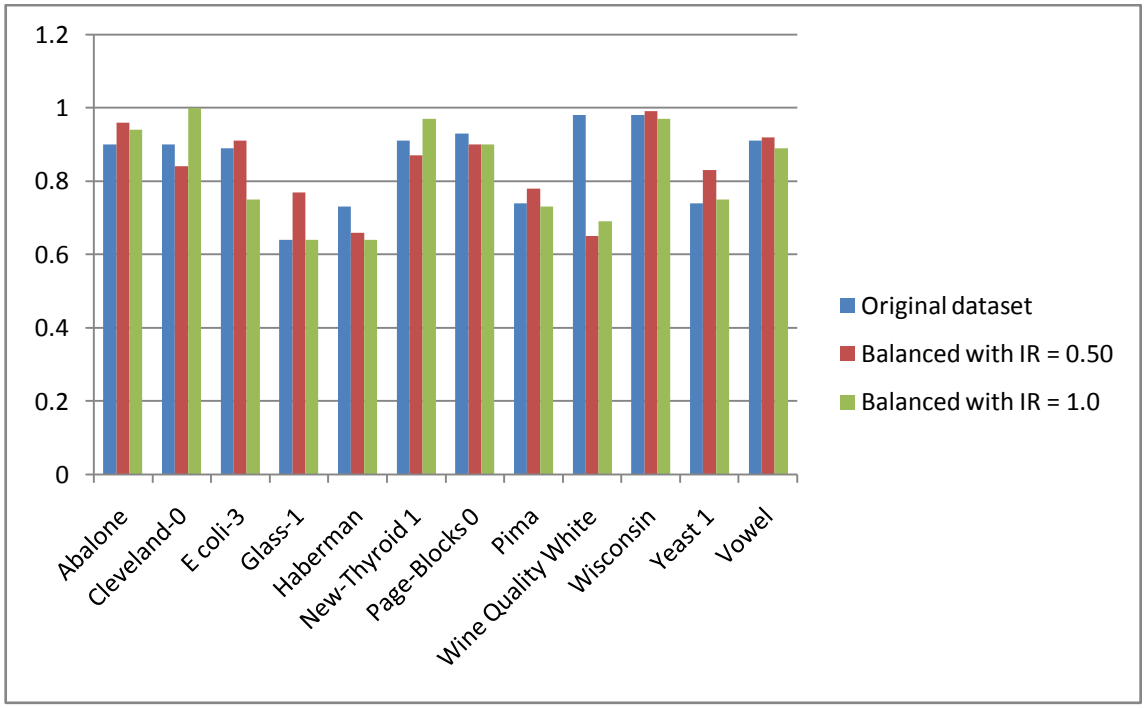

Fig.4.9 Chart for precision rate

Fig. 4.9 is a pictorial representation of the Precision of the SVM classifier for the original dataset, an Imbalanced ration of 0.50, and an Imbalance ration of 1 on 12 datasets.

Table 4. 7: Comparative Results for ROC rate with different imbalance ratio.

S.No	Data set	Original data set	Balanced with IR = 0.50	Balanced with IR = 1.0
1.	Abalone	0.90	0.98	0.97
2.	Cleveland-0	0.80	0.78	0.96
3.	Ecoli-3	0.50	0.89	0.81
4.	Glass-1	0.49	0.74	0.70
5.	Haberman	0.50	0.50	0.57
6.	New-Thyroid 1	0.77	0.84	0.98
7.	Page-Blocks 0	0.70	0.83	0.85
8.	Pima	0.72	0.70	0.75
9.	Wine Quality White	0.50	0.46	0.69
10.	Wisconsin	0.96	0.78	0.77
11.	Yeast 1	0.57	0.80	0.68
12.	Vowel	0.80	0.93	0.74

Table 4. 7 presents the ROC of SVM classifier for the original dataset, with Imbalance ration of .50 and Imbalance ration of 1 on 12 datasets. It can be easily identified that we are getting improved values for ROC with the proposed model.

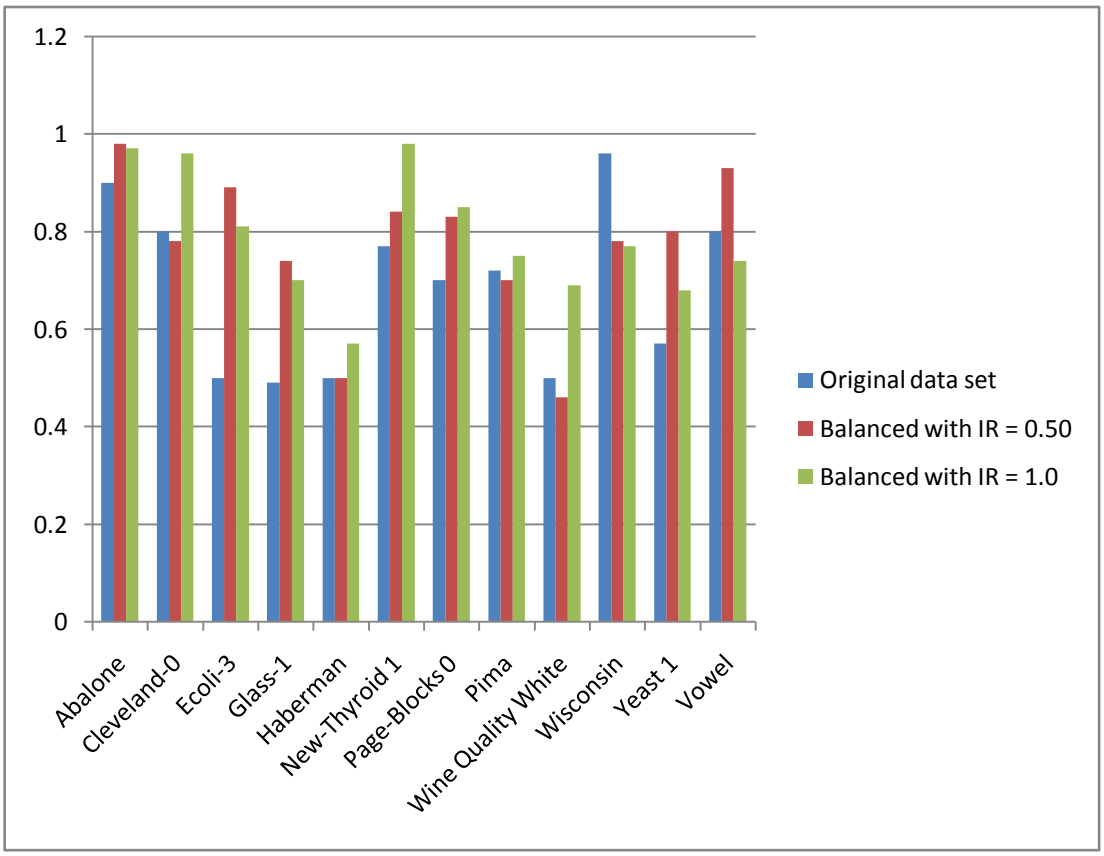

Fig. 4.10 Chart for ROC

Fig. 4.10 is a pictorial representation of ROC of SVM classifier for the original dataset, with an Imbalance ration of 0.50 and Imbalance ration of 1 on 12 datasets.

1.7 Summary

The main purpose of this approach is to selectively discard majority instances from the dataset to make the distribution balanced, which can be applied to any traditional classifier. Secondly, it is capable to handle between-class Imbalance distribution and within-class distribution. Thirdly it can handle the different degrees of Imbalanced distribution. The proposed model is simple yet effective in order to classify the Imbalanced distribution of Data.