Subset K-Means Approach for Handling Imbalanced-Distributed Data

Ch.N. Santhosh Kumar[1], K. Nageswara Rao[2], A. Govardhan[3], and N. Sandhya[4]

[1] Dept. of CSE, JNTU- Hyderabad, Telangana., India
santhosh_ph@yahoo.co.in
[2] PSCMR college of Engineering and Technology, Kothapet, Vijayawada, A.P., India
pricipal@pscmr.ac.in
[3] CSE, SIT, JNTU Hyderabad, Telangana, India
govardhan_cse@jntuh.ac.in
[4] CSE Department,VNR Vignana Jyothi Institite of Engineering & Technology, Hyderabad, India
sandhyanadela@gmail.com

Abstract. The effectiveness of clustering analysis relies not only on the assumption of cluster number but also on the class distribution of the data employed. This paper represents another step in overcoming a drawback of K-means, its lack of defense against imbalance data distribution. *K*-means is a partitional clustering technique that is well-known and widely used for its low computational cost. However, the performance of *k*-means algorithm tends to be affected by skewed data distributions, i.e., imbalanced data. They often produce clusters of relatively uniform sizes, even if input data have varied cluster size, which is called the "uniform effect." In this paper, we analyze the causes of this effect and illustrate that it probably occurs more in the *k*-means clustering process. As the minority class decreases in size, the "uniform effect" becomes evident. To prevent the effect of the "uniform effect", we revisit the well-known K-means algorithm and provide a general method to properly cluster imbalance distributed data.

The proposed algorithm consists of a novel under random subset generation technique implemented by defining number of subsets depending upon the unique properties of the dataset. We conduct experiments using ten UCI datasets from various application domains using five algorithms for comparison on eight evaluation metrics. Experiment results show that our proposed approach has several distinctive advantages over the original k-means and other clustering methods.

Keywords: data, *k*-means clustering algorithms, oversampling, K-Subset.

1 Introduction

Cluster analysis is a well-studied domain in data mining. In cluster analysis data is analyzed to find hidden relationships between each other to group a set of objects into clusters. One of the most popular methods in cluster analysis is k-means algorithm. The popularity and applicability of k-means algorithm in real time

S.C. Satapathy et al. (eds.), *Emerging ICT for Bridging the Future – Volume 2*,
Advances in Intelligent Systems and Computing 338, DOI: 10.1007/978-3-319-13731-5_54

applications is due to its simplicity and high computational capability. However, further investigation is the need of the hour to better understand the efficiency of k-means algorithm with respect to the data distribution used for analysis.

A good amount of research had done on the class balance data distribution for the performance analysis of k-means algorithm. For skewed-distributed data, the k-means algorithm tend to generate poor results as some instances of majority class are portioned into minority class, which makes clusters to have relatively uniform size instead of input data have varied cluster of non-uniform size. In [1] authors have defined this abnormal behavior of k-means clustering as the “uniform effect”. It is noteworthy that class imbalance is emerging as an important issue in cluster analysis especially for k-means type algorithms because many real-world problems, such as remote-sensing, pollution detection, risk management, fraud detection, and especially medical diagnosis are of class imbalance. Furthermore, the rare class with the lowest number of instances is usually the class of interest from the point of view of the cluster analysis.

Liu et al. [2], proposed a multiprototype clustering algorithm, which applies the k-means algorithm to discover clusters of arbitrary shapes and sizes. However, there are following problems in the real applications of these algorithms to cluster imbalanced data. 1) These algorithms depend on a set of parameters whose tuning is problematic in practical cases. 2) These algorithms make use of the randomly sampling technique to find cluster centers. However, when data are imbalanced, the selected samples more probably come from the majority classes than the minority classes. 3) The number of clusters k needs to be determined in advance as an input to these algorithms. In a real dataset, k is usually unknown. 4) The separation measures between sub-clusters that are defined by these algorithms cannot effectively identify the complex boundary between two sub-clusters. 5) The definition of clusters in these algorithms is different from that of k-means. Xiong et al. [1] provided a formal and organized study of the effect of skewed data distributions on the hard k-means clustering. However, the theoretic analysis is only based on the hard k-means algorithm. Their shortcomings are analyzed and a novel algorithm is proposed.

2 Class Imbalance Learning

One of the most popular techniques for alleviating the problems associated with class imbalance is data sampling. Data sampling alters the distribution of the training data to achieve a more balanced training data set. This can be accomplished in one of two ways: under sampling or oversampling. Under sampling removes majority class examples from the training data, while oversampling adds examples to the minority class. Both techniques can be performed either randomly or intelligently.

The random sampling techniques either duplicate (oversampling) or remove (under sampling) random examples from the training data. Synthetic minority oversampling technique (SMOTE) [3] is a more intelligent oversampling technique that creates new minority class examples, rather than duplicating existing ones. Wilson’s editing (WE) intelligently under-samples data by only removing examples that are thought to be noisy.

Finding minority class examples effectively and accurately without losing overall performance is the objective of class imbalance learning. The fundamental issue to be resolved is that the clustering ability of most standard learning algorithms is significantly compromised by imbalanced class distributions. They often give high overall accuracy, but form very specific rules and exhibit poor generalization for the small class. In other words, over fitting happens to the minority class. Correspondingly, the majority class is often over generalized. Particular attention is necessary for each class. It is important to know if a performance improvement happens to both classes and just one class alone.

3 Related Work

In this section, we first review the major research about clustering in class imbalance learning and explain why we choose oversampling as our technique in this paper.

Tapas Kanungo et al., [4] have presented a simple and efficient implementation of Lloyd's k-means clustering algorithm, which stores the multidimensional data points in a kd-tree. A kd-tree is a binary tree, which represents a hierarchical subdivision of the point set's bounding box using axis aligned splitting hyperplanes. Renato Cordeiro de Amorim et al., [5] have proposed a variation of k-means for tackling against noisy features using feature weights in the criterion. Serkan Kiranyaz et al., [6] have proposed a framework using exhaustive k-means clustering technique for addressing the problem in a long term ECG signal, known as Holter register. The exhaustive K-means clustering is used in order to find out (near-) optimal number of key-beats as well as the master key-beats. The expert labels over the master key-beats are then back-propagated over the entire ECG record to obtain a patient-specific, long-term ECG classification.

Haitao xiang et al., [7] have proposed a local clustering ensemble learning method based on improved AdaBoost (LCEM) for rare class analysis. LCEM uses an improved weight updating mechanism where the weights of samples which are invariably correctly classified will be reduced while that of samples which are partially correctly classified will be increased. Amuthan Prabakar et al., [8] have proposed a supervised network anomaly detection algorithm by the combination of k-means and C4.5 decision tree exclusively used for portioning and model building of the intrusion data. The proposed method is used mitigating the Forced Assignment and Class Dominance problems of the k-Means method. Li Xuan et al., [9] have proposed two methods, in first method they applied random sampling of majority subset to form multiple balanced datasets for clustering and in second method they observed the clustering partitions of all the objects in the dataset under the condition of balance and imbalance at a different angle. Christos Bouras et al., [10] have proposed W-k means clustering algorithm for applicability on a corpus of news articles derived from major news portals. The proposed algorithm is an enhancement of standard k-means algorithm using the external knowledge for enriching the ‘‘bag of words’’ used prior to the clustering process and assisting the label generation procedure following it.

P.Y. Mok et al., [11] have proposed a new clustering analysis method that identifies the desired cluster number and produces, at the same time, reliable clustering solutions. It first obtains many clustering results from a specific algorithm, such as Fuzzy C-Means (FCM), and then integrates these different results as a judgment matrix. An iterative graph-partitioning process is implemented to identify the desired cluster number and the final result.Luis A. Leiva et al., [12] have proposed Warped K-Means, a multi-purpose partition clustering procedure that minimizes the sum of squared error criterion, while imposing a hard sequentiality constraint in the classification step on datasets embedded implicitly with sequential information. The proposed algorithm is also suitable for online learning data, since the change of number of centroids and easy updating of new instances for the final cluster is possible. M.F.Jiang et al., [13] have proposed variations of k-means algorithm to identify outliers by clustering the data the initial phase then using minimum spanning tree to identify outliers for their removal.

Jie Cao et al., [14] have proposed a Summation-bAsed Incremental Learning (SAIL) algorithm for Information-theoretic K-means (Info-Kmeans) aims to cluster high-dimensional data, such as images featured by the bag-of-features (BOF) model, using K-means algorithm with KL-divergence as the distance. Since SAIL is a greedy scheme it first selects an instance from data and assigns it to the most suitable cluster. Then the objective-function value and other related variables are updated immediately after the assignment. The process will be repeated until some stopping criterion is met. One of the shortcomings is to select the appropriate cluster for an instance. Max Mignotte [15] has proposed a new and simple segmentation method based on the K-means clustering procedure for applicability on image segmentation. The proposed approach overcome the problem of local minima, feature space without considering spatial constraints and uniform effect.

4 Framework of k-Subset Algorithm

This section presents the proposed algorithm, whose main characteristics are depicted in the following sections. Initially, the main concepts and principles of k-means are presented. Then, the definition of our proposed K-subset is introduced in detail.

K-means is one of the simplest unsupervised learning algorithms, first proposed by Macqueen in 1967, which has been used by many researchers to solve some well-known clustering problems [16]. The technique classifies a given data set into a certain number of clusters (assume *k* clusters). The algorithm first randomly initializes the clusters center. The next step is to calculate the distance between an object and the centroid of each cluster. Next each point belonging to a given data set is associated with the nearest center. The cluster centers are then re-calculated. The process is repeated with the aim of minimizing an objective function knows as squared error function given by:

$$Jv = {}_{i=1}^{C}{}_{j=1}^{Ci}\left(\left\|x_i - v_j\right\|\right)^2 \tag{1}$$

Where, $\left(\left\|x_i - v_j\right\|\right)$ is the Euclidean distance between the data point x_i and cluster center v_j , ci is the number of data points in cluster and c is the number of i^{th} cluster centers.

The entire process is given in the following algorithm,

4.1 Dividing Majority and Minority Subset

An easy way to sample a dataset is by selecting instances randomly from all classes. However, sampling in this way can break the dataset in an unequal priority way and more number of instances of the same class may be chosen in sampling. To resolve this problem and maintain uniformity in sample, we propose a sampling strategy called weighted component sampling. Before creating multiple subsets, we will create the number of majority subsets depending upon the number of minority instances.

4.2 Identifying Number of Subsets of Majority Class

In the next phase of the approach, the ratio of majority and minority instances in the unbalanced dataset is used to decide the number of subset of majority instances (T) to be created.

T= no. of majority inst (N)./no. of minority inst (P).

4.3 Combing the Majority Subsets and Minority Subset

The so formed majority subsets are individual combined with the only minority subset to form multiple balanced sub datasets of every dataset. The number of balanced sub datasets formed depends upon the imbalance ratio and the unique properties of the dataset

4.4 Averaging the measures

The subsets of balanced datasets created are used to run multiple times and the resulted values are averaged to find the overall result. This newly formed multiple subsets are applied to a base algorithm; in this case k-means is used to obtain different measures such as AUC, Precision, F-measure, TP Rate and TN Rate.

Algorithm : K-Subset

```
{Input: A set of minor class examples $P$, a set Of major class examples $N$, $jPj <jN j$, and $T$, the number of subsets to be sampled from $N$.}
$i \leftarrow 0$, T=N/P. repeat
$i = i + 1$
Randomly sample a subset $Ni$ from $N$, $jNij = jPj$.
Combine P and Ni to form NPi
Apply filter on a NPi
```

```
7: Train and Learn on a base Algorithm (k-means) using
   NPi. Obtain the values of AUC,TP,FP,F-Measure
8: until $i = T$
9: Output: Average Measure;
```

5 Datasets

In the study, we have considered 10 binary data-sets which have been collected from the KEEL [18] and UCI [17] machine learning repository Web sites, and they are very varied in their degree of complexity, number of classes, number of attributes, number of instances, and imbalance ratio (the ratio of the size of the majority class to the size of the minority class). The number of classes' ranges up to 2, the number of attributes ranges from 8 to 60, the number of instances ranges from 57 to 3772, and the imbalance ratio is up to 15.32. This way, we have different Imbalance Ratios (IRs): from low imbalance to highly imbalanced data-sets. Table 1 summarizes the properties of the selected data-sets: for each data-set, S.no, Dataset name, the number of examples (#Ex.), number of attributes (#Atts.), class name of each class (minority and majority) and the IR. This table is ordered according to the name of the datasets in alphabetical order. We have obtained the AUC metric estimates by means of a 10-fold cross-validation. That is, the data-set was split into ten folds, each one containing 10% of the patterns of the dataset. For each fold, the algorithm is trained with the examples contained in the remaining folds and then tested with the current fold. The data partitions used in this paper can be found in UCI-dataset repository [17] so that any interested researcher can reproduce the experimental study. The algorithms used in the experimental study and their parameter settings, which are obtained from the KEEL [18] and WEKA [19] software tools.

Table 1. Summary of benchmark imbalanced datasets

S.no	Datasets	# Ex.	# Atts.	Class (_,+)	IR
1.	Breast_w	699	9	(benign; malignant)	1.90
2.	Colic	368	22	(yes; no)	1.71
3.	Diabetes	768	8	(tested-potv; tested-negtv)	1.87
4.	Ecolic	336	7	(cp; oml)	2.33
5.	Hepatitis	155	19	(die; live)	3.85
6.	Ionosphere	351	34	(b;g)	1.79
7.	Labor	57	17	(bad; good)	1.85
8.	Sick	3772	30	(negative; sick)	15.32
9.	Sonar	208	60	(rock ; mine)	1.15
10.	Vote	435	17	(democrat ; republican)	1.58

Several clustering methods have been selected and compared to determine whether the proposal is competitive in different domains with the other approaches. Algorithms are compared on equal terms and without specific settings for each data problem.

6 Experimental Results

Table 2-9 presents the performance of each clustering technique averaged across all learners and data sets. These tables give a general view of the performance of each technique using each of the eight performance metrics. Tables 2-9 provide both the numerical average performance (Mean) and the standard deviation (SD) results. If the proposed technique is better than the compared technique then '•' symbol appears in the column. If the proposed technique is not better than the compared technique then '○' symbol appears in the column. The mean performances were significantly different according to the T-test at the 95% confidence level.

We carry out the empirical comparison of our proposed algorithm with the benchmarks. Our aim is to answer several questions about the proposed learning algorithms in the scenario of two-class imbalanced problems.

1) In first place, we want to analyze which one of the approaches is able to better handle a large amount of imbalanced data-sets with different IR, i.e., to show which one is the most robust method.

2) We also want to investigate their improvement with respect to classic clustering methods and to look into the appropriateness of their use instead of applying a unique preprocessing step and training a single method. That is, whether the trade-off between complexity increment and performance enhancement is justified or not. Given the amount of methods in the comparison, we cannot afford it directly. On this account, we compared the proposed algorithm with each and every algorithm independently. This methodology allows us to obtain a better insight on the results by identifying the strengths and limitations of our proposed method on every compared algorithm. The clustering evaluations were conducted on ten widely used datasets. These real world multi-dimensional datasets are used to verify the proposed clustering method. Table 2, 3, 4, 5, 6, 7, 8 and 9 reports the results of Accuracy, AUC, Precision, Recall, F-measure, Specificity, FP Rate and FN Rate respectively for all the ten datasets from UCI.

A two-tailed corrected resampled paired t-test is used in this paper to determine whether the results of the cross-validation show that there is a difference between the two algorithms is significant or not. Difference in accuracy is considered significant when the p-value is less than 0.05 (confidence level is greater than 95%). The results in the tables show that K-Subset has given a good improvement on all the clustering measures. Two main reasons support the conclusion achieved above. The first one is the decrease of instances in majority subset, has also given its contribution for the better performance of our proposed K-Subset algorithms. The second reason, it is well-known that the resampling of synthetic instances in the minority subset is the only way in oversampling but conduction proper exploration – exploitation of prominent instances in minority subset is the key for the success of our algorithm. Another reason is the deletion of noisy instances by the interpolation mechanism of K-Subset.

Table 2. Summary of tenfold cross validation performance for Accuracy on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	95.82±2.26○	96.22±2.19○	84.94±6.96●	93.75±2.79●	65.52±0.44●	94.64±3.15
Colic	60.57±11.89○	65.30±10.85○	58.67±9.91●	66.13±7.11○	63.05±1.13○	76.96±10.2
Diabetes	65.42±5.87○	65.60±5.68○	65.16±3.42○	64.67±5.74○	65.11±0.34○	63.07±6.22
Ecolic	55.86±6.77●	56.37±6.72●	62.41±6.56●	60.60±5.33●	70.00±0.00○	81.74±5.47
Hepatitis	71.09±12.58○	73.15±12.16○	72.14±12.77○	73.83±10.53○	79.38±2.26○	76.65±16.03
Ionosphere	70.80±6.71●	73.06±6.35○	62.75±6.65●	73.08±6.47○	64.10±1.35●	69.70±10.3
Labor	65.45±22.84●	69.12±21.69○	74.73±13.17○	55.67±21.78●	64.67±3.07●	64.37±25.06
Sick	73.75±7.86●	71.28±8.74●	87.29±6.06○	50.01±26.34●	93.88±0.08○	79.19±5.16
Sonar	52.43±10.28○	50.12±10.40	50.94±8.28	49.59±9.55●	51.78±3.41	70.10±14.44
Vote	85.73±5.30	87.22±4.64○	84.54±8.10●	60.59±7.74●	61.38±0.81●	88.19±8.84

Table 3. Summary of tenfold cross validation performance for AUC on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.950±0.027○	.966±0.021○	.785±0.098●	.951±0.022○	.500±0.000●	.947±0.030
Colic	.628±0.108○	.678±0.092○	.570±0.114●	.691±0.068○	.500±0.000●	.766±0.105
Diabetes	.608±0.067●	.617±0.068○	.520±0.044●	.670±0.070○	.502±0.006●	.634±0.062
Ecolic	.534±0.067●	.535±0.066●	.521±0.057●	.567±0.051●	.500±0.000●	.921±0.054
Hepatitis	.753±0.136○	.781±0.122○	.670±0.163●	.800±0.101○	.500±0.000●	.768±0.160
Ionosphere	.706±0.080●	.743±0.079○	.530±0.067●	.771±0.058○	.500±0.000●	.703±0.102
Labor	.631±0.237●	.668±0.233○	.657±0.169○	.586±0.196●	.500±0.000●	.640±0.254
Sick	.574±0.157○	.567±0.154○	.481±0.038●	.516±0.086●	.500±0.000●	.768±0.054
Sonar	.521±0.103○	.498±0.104	.513±0.082○	.497±0.096●	.500±0.000○	.719±0.148
Vote	.871±0.053○	.885±0.047○	.855±0.083●	.759±0.053●	.500±0.000●	.877±0.089

Table 4. Summary of tenfold cross validation performance for Precision on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.961±0.024●	.989±0.015○	.823±0.071●	.998±0.007○	.655±0.004●	.921±0.048
Colic	.784±0.107○	.821±0.083○	.719±0.120●	.838±0.069○	.630±0.011●	.765±0.130
Diabetes	.725±0.047○	.734±0.051○	.665±0.044●	.821±0.072○	.652±0.004●	.592±0.054
Ecolic	.727±0.052○	.727±0.053○	.712±0.035○	.746±0.036○	.700±0.000○	.810±0.093
Hepatitis	.426±0.150●	.453±0.156●	.405±0.233●	.457±0.136●	.000±0.000●	.787±0.220
Ionosphere	.557±0.147●	.573±0.145●	.309±0.369●	.585±0.068	.000±0.000●	.821±0.131
Labor	.474±0.389●	.523±0.388○	.532±0.484○	.364±0.437●	.000±0.000●	.631±0.327
Sick	.952±0.026●	.952±0.028●	.936±0.005●	.958±0.037○	.939±0.001●	.845±0.068
Sonar	.493±0.133●	.460±0.140●	.422±0.238●	.459±0.108●	.110±0.198●	.866±0.142
Vote	.952±0.048○	.960±0.042○	.929±0.077●	.996±0.017○	.614±0.008●	.906±0.131

Table 5. Summary of tenfold cross validation performance for Recall on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.961±0.024●	.989±0.015○	.823±0.071●	.998±0.007○	.655±0.004●	.921±0.048
Breast_w	.976±0.022○	.953±0.029●	.992±0.033○	.907±0.042●	1.000±0.000○	.977±0.030
Colic	.541±0.231○	.582±0.195○	.635±0.229○	.576±0.100○	1.000±0.000○	.735±0.183
Diabetes	.760±0.091○	.747±0.089○	.957±0.106○	.594±0.075●	1.000±0.000○	.772±0.083
Ecolic	.595±0.108●	.606±0.112●	.778±0.121●	.664±0.081●	1.000±0.000○	.982±0.080
Hepatitis	.824±0.225○	.865±0.190○	.583±0.319●	.906±0.153○	.000±0.000●	.781±0.251
Ionosphere	.702±0.187	.787±0.195○	.185±0.284●	.912±0.074○	.000±0.000●	.594±0.178
Labor	.555±0.413●	.588±0.412○	.345±0.338●	.330±0.403●	.000±0.000●	.710±0.363
Sick	.779±0.122●	.755±0.139●	.957±0.078○	.719±0.169●	1.000±0.000○	.772±0.087
Sonar	.471±0.201●	.471±0.215●	.524±0.392●	.525±0.194●	.235±0.425●	.654±0.190
Vote	.810±0.069○	.829±0.064○	.814±0.098●	.931±0.062○	1.000±0.000○	.837±0.133

Finally, we can make a global analysis of results combining the results offered by Tables from 2–9:

Our proposals, K-Subset is the best performing one when the data sets are of imbalance category. We have considered a complete competitive set of methods and an improvement of results is expected in the benchmark algorithms i;e K-means, Density, FF, EM and Hier. However, they are not able to outperform K-Subset. In this sense, the competitive edge of K-Subset can be seen.

Table 6. Summary of tenfold cross validation performance for F-measure on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.968±0.017●	.971±0.017●	.898±0.045●	.950±0.024●	.792±0.003●	.947±0.030
Colic	.608±0.162●	.662±0.137○	.638±0.141●	.678±0.082○	.773±0.008○	.729±0.143
Diabetes	.739±0.053●	.737±0.050●	.779±0.049●	.685±0.055●	.789±0.003○	.668±0.056
Ecolic	.649±0.074○	.655±0.074○	.739±0.066○	.700±0.052○	.824±0.000○	.876±0.079
Hepatitis	.549±0.157●	.582±0.153●	.451±0.226●	.598±0.131	.000±0.000●	.755±0.197
Ionosphere	.617±0.155●	.660±0.157○	.173±0.222●	.711±0.059○	.000±0.000●	.661±0.145
Labor	.481±0.358○	.522±0.357○	.404±0.370●	.328±0.387○	.000±0.000●	.640±0.303
Sick	.851±0.066○	.834±0.077○	.945±0.044○	.807±0.109○	.968±0.000○	.804±0.066
Sonar	.462±0.149●	.447±0.159●	.414±0.257●	.480±0.133●	.149±0.270●	.724±0.151
Vote	.873±0.048○	.888±0.042○	.864±0.075●	.961±0.037○	.761±0.006●	.862±0.102

Table 7. Summary of tenfold cross validation performance for Specificity on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.968±0.017○	.979±0.029○	.578±0.196●	.996±0.012○	.000±0.000●	.911±0.055
Colic	.608±0.162●	.773±0.157●	.506±0.336●	.807±0.093○	.000±0.000●	.799±0.148
Diabetes	.739±0.053○	.487±0.155●	.082±0.163●	.746±0.145○	.000±0.000●	.500±0.094
Ecolic	.649±0.074○	.464±0.149○	.264±0.145●	.470±0.100○	.000±0.000●	.861±0.074
Hepatitis	.549±0.157○	.696±0.143○	.757±0.154○	.695±0.125○	1.000±0.000○	.755±0.263
Ionosphere	.617±0.155●	.699±0.110●	.875±0.215○	.629±0.098●	1.000±0.000○	.812±0.162
Labor	.481±0.358●	.749±0.281○	.968±0.117○	.854±0.257○	1.000±0.000○	.570±0.380
Sick	.851±0.066○	.401±0.379○	.033±0.095●	.500±0.434○	.000±0.000●	.852±0.063
Sonar	.462±0.149○	.528±0.198○	.502±0.357○	.470±0.171○	.768±0.420○	.784±0.251
Vote	.873±0.048●	.941±0.075	.895±0.122●	.996±0.017○	.000±0.000●	.918±0.116

Table 8. Summary of tenfold cross validation performance for FP Rate on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.076±0.049	.021±0.029●	.422±0.196○	.004±0.012●	1.000±0.000○	.083±0.054
Colic	.285±0.240○	.227±0.157●	.494±0.336○	.193±0.093●	1.000±0.000○	.201±0.148
Diabetes	.544±0.145○	.513±0.155●	.918±0.163○	.254±0.145●	1.000±0.000○	.500±0.094
Ecolic	.526±0.143○	.536±0.149○	.736±0.145○	.530±0.100○	1.000±0.000○	.138±0.074
Hepatitis	.318±0.146●	.304±0.143●	.243±0.154●	.305±0.125●	.000±0.000●	.245±0.263
Ionosphere	.289±0.110○	.301±0.110○	.125±0.215●	.371±0.098○	.000±0.000●	.188±0.162
Labor	.292±0.313○	.251±0.281●	.032±0.117●	.146±0.257●	.000±0.000●	.430±0.380
Sick	.614±0.366●	.599±0.379●	.967±0.095○	.500±0.434●	.000±0.000●	.148±0.062
Sonar	.429±0.195●	.472±0.198●	.498±0.357●	.530±0.171●	.232±0.420●	.216±0.251
Vote	.068±0.078○	.059±0.075	.105±0.122○	.004±0.017●	1.000±0.000○	.083±0.116

Table 9. Summary of tenfold cross validation performance for FN Rate on all the datasets

Datasets	K-Means	Density	FF	EM	Hier	K-Subset
Breast_w	.024±0.022●	.047±0.029○	.008±0.033●	.093±0.042○	.000±0.000●	.023±0.030
Colic	.459±0.231●	.418±0.195●	.365±0.229●	.424±0.100●	.000±0.000●	.265±0.183
Diabetes	.240±0.091●	.254±0.089●	.043±0.106●	.406±0.075○	.000±0.000●	.229±0.083
Ecolc	.405±0.108○	.394±0.112○	.222±0.121○	.336±0.081○	.000±0.000●	.019±0.080
Hepatitis	.176±0.225●	.135±0.190 ●	.417±0.319○	.094±0.153●	1.000±0.000○	.219±0.251
Ionosphere	.298±0.187●	.213±0.195●	.815±0.284○	.088±0.074●	1.000±0.000○	.407±0.178
Labor	.445±0.413●	.413±0.412●	.650±0.340○	.540±0.436○	1.000±0.000○	.290±0.363
Sick	.221±0.122○	.245±0.139○	.043±0.078●	.281±0.169○	.000±0.000●	.278±0.132
Sonar	.529±0.201○	.529±0.215○	.476±0.392○	.475±0.194○	.765±0.425○	.346±0.190
Vote	.190±0.069●	.171±0.064●	.186±0.098●	.069±0.062●	.000±0.000●	.163±0.133

Considering that K-Subset behaves similarly or not effective than K-means shows the unique properties of the datasets where there is scope of improvement in minority subset and not in majority subset. Our K-Subset can mainly focus on improvements in majority subset which is not effective for some unique property datasets. The Accuracy, AUC, Recall, F-measure, TN Rate, FP Rate and FN Rate measure have shown to perform well with respect to K-Subset. The strengths of our model are that K-Subset only increases the number of majority subsets thereby strengthens the minority class. One more point to consider is our method tries to remove the noisy instances from both majority and minority set if any applicable. Finally, we can say that K-Subset is one of the best alternatives to handle class imbalance problems effectively. This experimental study supports the conclusion that the a prominent recursive oversampling approach of majority subsets can improve the class imbalance behavior when dealing with imbalanced data-sets, as it has helped the K-Subset methods to be the best performing algorithm when compared with four classical and well-known algorithms: K-means, Density, FF, EM and a well-established Hierarchical algorithm.

7 Conclusion

In this paper, a novel clustering algorithm for imbalanced distributed data has been proposed. This method uses unique oversampling technique to almost balance dataset such that to minimize the "uniform effect" in the clustering process. Empirical results have shown that K-Subset considerably reduces the uniform effect while retaining or improving the clustering measure when compared with benchmark methods. In fact, the proposed method may also be useful as a frame work for data sources for better clustering measures.

References

1. Xiong, H., Wu, J.J., Chen, J.: K-means clustering versus validation measures: A data-distribution perspective. IEEE Trans. Syst., Man, Cybern. B, Cybern. 39(2), 318–331 (2009)
2. Liu, M.H., Jiang, X.D., Kot, A.C.: A multi-prototype clustering algorithm. Pattern Recognit. 42, 689–698 (2009)
3. Chawla, N., Bowyer, K., Kegelmeyer, P.: SMOTE: Synthetic minority over-sampling technique. J. Artif. Intell. Res. 16, 321–357 (2002)
4. Kanungo, T., Mount, D.M., Netanyahu, N.S., Piatko, C.D., Silverman, R., Wu, A.Y.: An Efficient k-Means Clustering Algorithm: Analysis and Implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7) (July 2002)
5. de Amorim, R.C., Mirkin, B.: Minkowski metric, feature weighting and anomalous cluster initializing in K-Means clustering. Pattern Recognition 45, 1061–1075 (2012)
6. Kiranyaz, S., Ince, T., Pulkkinen, J., Gabbouj, M.: Personalized long-term ECG classification: A systematic approach. Expert Systems with Applications 38, 3220–3226 (2011)
7. Xiang, H., Yang, Y., Zhao, S.: Local Clustering Ensemble Learning Method Based on Improved AdaBoost for Rare Class Analysis. Journal of Computational Information Systems 8(4), 1783–1790 (2012)
8. Muniyandi, A.P., Rajeswari, R., Rajaram, R.: Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm. In: International Conference on Communication Technology and System Design 2011. Procedia Engineering, vol. 30, pp. 174–182 (2012)
9. Li, X., Chen, Z., Yang, F.: Exploring of clustering algorithm on class-imbalanced data
10. Bouras, C., Tsogkas, V.: A clustering technique for news articles using WordNet. Knowl. Based Syst. (2012), http://dx.doi.org/10.1016/j.knosys.2012.06.015
11. Mok, P.Y., Huang, H.Q., Kwok, Y.L., Au, J.S.: A robust adaptive clustering analysis method for automatic identification of clusters. Pattern Recognition 45, 3017–3033 (2012)
12. Leiva, L.A., Vidal, E.: Warped K-Means: An algorithm to cluster sequentially-distributed data. Information Sciences 237, 196–210 (2013)
13. Jaing, M.F., Tseng, S.S., Su, C.M.: Two Phase Clustering Process for Outlier Detection. Pattern Recognition Letters 22, 691–700 (2001)
14. Cao, J., Wu, Z., Wu, J., Liu, W.: Towards information-theoretic K-means clustering for image indexing. Signal Processing 93, 2026–2037 (2013)
15. Mignotte, M.: A de-texturing and spatially constrained K-means approach for image segmentation. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec, 09.016

16. Maimon, O., Rokach, L.: Data mining and knowledge discovery handbook. Springer, Berlin (2010)
17. Blake, C., Merz, C.J.: UCI repository of machine learning databases. Machine-readable data repository. Department of Information and Computer Science, University of California at Irvine, Irvine (2000), http://www.ics.uci.edu/mlearn/MLRepository.html
18. http://www.keel.com
19. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)