Imbalanced K-Means: An algorithm to cluster imbalanced-distributed data

Ch.N.Santhosh Kumar, Dr. K.Nageswara Rao, Dr.A.Govardhan, DrK.Sudheer Reddy

***Abstract*— *K*-means is a partitional clustering technique that iswell-known and widely used for its low computational cost. However, the performance of *k*-means algorithm tends to beaffected by skewed data distributions, i.e., imbalanced data. Theyoften produce clusters of relatively uniform sizes, even if input datahave varied a cluster size, which is called the "uniform effect." Inthis paper, we analyze the causes of this effect and illustrate thatit probably occurs more in the *k*-means clustering process. As the minority class decreases in size, the "uniform effect" becomes evident. To prevent theeffect of the "uniform effect", we revisit the well-known K-means algorithmand provide a general method to properly cluster imbalance distributed data. We present Imbalanced K-Means (IKM), a multi-purpose partitional clustering procedure that minimizes the clustering sum of squared error criterion, while imposing a hard sequentiality constraint in theclustering step.**

The proposed algorithm consists of a novel oversampling technique implemented by removing noisy and weak instances from both majority and minority classes and then oversampling only novel minority instances. We conduct experiments using twelve UCI datasets from various application domains using fivealgorithms for comparison on eight evaluation metrics. Experimental results show the effectiveness of the proposed clustering algorithm in clustering balanced and imbalanced data.

***Index Terms*— Imbalanced data, *k*-meansclustering algorithms, oversampling, Imbalanced K-Means.**

I. 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 applications is due to its simplicity and high computational capability. Researchers have identified several factors [1] that may strongly affect the *k*-means clustering analysis including high dimensionality [2]–[4], sparseness of the data [5], noise and outliers in the data [6]–[8], scales of the data [9]–[12], types of attributes [13], [14], the fuzzy index *m* [15]–[18], initial cluster centers [19]–[24], and the number of clusters [25]–[27]. 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 [28] 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 [29], pollution detection [30], risk management [31], fraud detection [32], and especially medical diagnosis [33]–[36] 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.

Guhaet al.[37] early proposed to make use of multiple representative points to get the shape information of the "natural" clusters with nonspherical shapes [1] and achieve an improvement on noise robustness over the single-link algorithm. Liu *et al.* [38],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 subclusters that are defined by these algorithms cannot effectively identify the complex boundary between two subclusters. 5) The definition of clusters in these algorithms is different from that of *k*-means. Xiong*et al.* [33] 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.

This paper focuses on clustering of binary dataset problems. The rest of this paper is organized as follows: Section 2 presents the concept of class imbalance learning and the uniform effect in k-means algorithm. Section 3 presents the main related work about k-means clustering algorithm. Section 4 provides a detailed explanation of the Imbalanced

Manuscript received Feb. 17, 2014.
Ch.N.Santhosh Kumar, Research Scholar, Dept. of CSE, JNTU-Hyderabad, A.P., India.
Dr. K.Nageswara Rao, Principal, PSCMR college of Engineering and Technology, Kothapet, Vijayawada**,** A.P., India.
Dr.A.Govardhan, Professor in CSE & SIT, JNTU Hyderabad, A.P., India.
DrK.Sudheer Reddy, Researcher, Hyderabad. A.P., India.

K-Means algorithm. Section 5 presents the datasets used for experiments. Section 6 presents the algorithms used for comparison. Section 7 presents the experimental results. Section 8 draws the conclusions and points out future research.

II. 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) [2] is a more intelligent oversampling technique that creates new minority class examples, rather than duplicating existing ones. Wilson's editing (WE) [3] intelligently undersamples data by only removing examples that are thought to be noisy. In this study, we investigate the impact of intelligent oversampling technique on the performance of the clustering algorithms.While the impacts of noise and imbalance have been frequently investigated in isolation, their combined impacts have not received enough attention in research, particularly with respect to clustering algorithms. To alleviate this deficiency, we present a comprehensive empirical investigation of learning from noisy and imbalanced data using k-means clustering algorithm.

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, overfitting happens to the minority class [6], [36], [37], [38], [39]. Correspondingly, the majority class is often overgeneralized. 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.

Many algorithms and methods have been proposed to ameliorate the effect of class imbalance on the performance of learning algorithms. There are three main approaches to these methods.

• *Internal approaches acting on the algorithm*. These approaches modify the learning algorithm to deal with the imbalance problem. They can adapt the decision threshold to create a bias toward the minority class or introduce costs in the learning process to compensate the minority class.

• *External approaches acting on the data*. These algorithms act on the data instead of the learning method. They have the advantage of being independent from the classifier used. There are two basic approaches: oversampling the minority class and undersampling the majority class.

• *Combined approaches that are based on boosting accounting for the imbalance in the training set*. These methods modify the basic boosting method to account for minority class underrepresentation in the data set. There are two principal advantages of choosing sampling over cost-sensitive methods. First, sampling is more general as it does not depend on the possibility of adapting a certain algorithm to work with classification costs. Second, the learning algorithm is not modified, which can cause difficulties and add additional parameters to be tuned.

III. 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. Then, we introduce frequently used ensemble methods and evaluation criteria in class imbalance learning

In recent years, clustering techniques have received much attention in wide areas of applicability such as medicine, engineering, finance and biotechnology. The main intention of clustering is to group data together which are having similar characteristics. Kaufman and Rousseeuw (1990) referred to clustering as "the art of finding groups in data". It's not fair to declare one clustering method as the best clustering method since the success of clustering method will highly depend on the type of data and the way of investigation for a specific applicability. Although many researchers attempted to make clustering process as a pure statistical technique but still largely it is regarded as an exploration procedure for finding the similar group of data.

Haitaoxiang et al., [39]have proposed a localclustering ensemble learning method based on improved AdaBoost (LCEM) for rare class analysis. LCEM usesan improved weight updating mechanism where the weights of samples which are invariably correctlyclassified will be reduced while that of samples which are partially correctly classified will be increased. The proposed algorithm also perform clustering on normal class and produce sub-classes with relatively balanced sizes.AmuthanPrabakar et al., [40] 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., [41]have proposed two methods, in first method they applied random sampling of majority subset to form multiple balanced datasets for clustering and in second methodthey observed the clustering partitions of all the objects in the dataset under the condition of balance and imbalance at a different angle. Christos Bouraset al., [42]have

proposed W-k meansclustering 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 andassisting the label generation procedure following it.

P.Y. Mok et al., [43]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. Leivaet al., [44]have proposed Warped K-Means, a multi-purpose partition clustering procedure that minimizesthe sum of squared error criterion, while imposing a hard sequentiality constraint in theclassification 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.Jianget al., [45] have proposed variations of k-means algorithm to identify outliersby clustering the data the initial phase then using minimum spanning tree to identify outliers for their removal.

Jie Cao et al., [46] 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[47] has proposed anew 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.

IV. Framework of IKM 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 IKM 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 [48]. 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:

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

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 different components of our new proposed framework are elaborated in the following subsection.

In the initial stage of our frame work the dataset is applied to a base algorithm for identifying mostly misclassified instances in both majority and minority classes. The instances which are misclassified are mostly weak instances and removing those instances from the majority and minority classes will not harm the dataset. In fact it will be helpful for improving the quality of the dataset in two fold; one way by removing weak instances from majority class will help to reduce the problem of class imbalance to a minor extend. Another is the removal of weak instances from minority class for the purpose of finding good instances to recursively replicate and hybridized for oversampling is also the part of the goal of the framework. The mostly misclassified instances are identified by using a base algorithm in this case C4.5 [49] is used. C4.5 is one of the best performing algorithms in the area of supervised learning. Our approach is classifier independent;i.e there is no constraint that the same classifier (in this case C4.5) has to be implemented for identifying mostly misclassified instances. The framework is introduced, and more interested researchers are encouraged to vary the components of the framework for more exploration.

In the next phase, the datasets is partitioned into majority and minority subsets. As we are concentrating on over sampling, we will take minority data subset for further analysis to generate synthetic instances.

Minority subset can be further analyzed to find the missing or noisy instances so that we can eliminate those. For finding noisy, boarder line and missing value instances for generating pure minority set, one of the ways is to go through a preprocessing process.

The good instances remained in the minority subset are to be resampled i;e both replicated and hybridized instances are generated. The percentage of synthetic instances generated will range from 0 – 100 % depending upon the percentage of difference of majority and minority classes in the original dataset. The synthetic minority instances generated can have a percentage of instances which can be a replica of the pure instances and reaming percentage of instances are of the

hybrid type of synthetic instances generated by combing two or more instances from the pure minority subset.
The oversampled minority subset and the majority subset are combined to form an almost balanced dataset, which is applied to a clustering algorithm. In this case we have used k-means clustering algorithm. The improvements in the imbalance dataset can be made into balance or almost balance depending upon the pure majority subset generated. The maximum synthetic minority instances generated are limited to 100% of the pure minority set formed. Our method will be superior to other oversampling methods since our approach uses the only available pure instances in the existing minority set for generating synthetic instances.

Suppose that the whole training set is *T*, the minority classis *P* and the majority class is *N*, and

$P = \{p1, p2, ..., ppnum\}$, $N = \{n1, n2, ..., nnnum\}$

where*pnum*and *nnum*are the number of minority and majority examples. The detailed procedure of IKMis as follows.

_____***Algorithm: IKM***

_____**Input:** A set of minor class examples *P*, a set of major class examples *N*, *jPj<jNj*, and *Fj*,the feature set, j > 0.
Output: Average Measure { AUC, Precision, F-Measure, TP Rate, TN Rate }

External selection Phase

Step 1: For every pi (i= 1,2,..., pnum) in the minority class P, we calculate its m nearest neighbors from the whole training set T. The number of majority examples among the m nearest neighbors is denoted by m' ($0 \le m' \le m$) .

Step 2: If $m/2 \le m' < m$, namely the number of pi 's majority nearest neighbors is larger than the number of its minority ones, pi is considered to be easily misclassified and put into a set MISCLASS.

MISSCLASS = m'

Remove the instances m' from the minority set.

Step 3: For every ni(i= 1,2,..., nnum) in the majority class N, we calculate its m nearest neighbors from the whole training set T. The number of majority examples among the m nearest neighbors is denoted by m' ($0 \le m' \le m$) .

Step 4: If $m/2 \le m' < m$, namely the number of ni 's minority nearest neighbors is larger than the number of its majority ones, ni is considered to be easily misclassified and put into a set MISCLASS.

MISSCLASS = m'

Remove the instances m' from the majority set.

Step 5: For every pi' (i= 1,2,..., pnum') in the minority class P, we calculate its m nearest neighbors from the whole training set T. The number of majority examples among the m nearest neighbors is denoted by m' ($0 \le m' \le m$).

If m'= m, i.e. all the m nearest neighbors of pi are majority examples, pi' is considered to be noise or outliers or missing values and are to be removed.

Step 6: For every pi'' (i= 1,2,..., pnum'') in the minority class P, we calculate its m nearest neighbors from the whole training set T. The number of majority examples among the m nearest neighbors is denoted by m' ($0 \le m' \le m$).

If $0 \le m' < m/2$, pi is a prominent example and need to be kept in minority set for resampling.

Step 7: The examples in minority set are the prominent examples of the minority class P, and we can see that PR⊆P . We set

PR= {p'1 ,p'2 ,..., p'dnum}, $0 \le dnum \le pnum$

Step 8: In this step, we generate s × dnum synthetic positive examples from the pr examples in minority set, where s is an integer between 1 and k. One percentage of synthetic examples generated are replica of pr examples and other are the hybrid of pr examples.

Clustering Phase

Step 1: Select k random instances from the training data subset as the centroids of the clusters C1; C2; ...Ck.

Step 2: For each training instance X:
a. Compute the Euclidean distance *D(Ci,X),i = 1...k*
b. Find cluster Cq that is closest to X.
c. Assign X to Cq.
Update the centroid of Cq.
(The centroid of a cluster is the arithmeticmean of the instances in the cluster.)

Step 3: Repeat Step 2 until the centroids of clusters C1; C2; ...Ck stabilize in terms of mean-squared error criterion.

_____

The algorithm 1: IKM can be explained as follows,

The inputs to the algorithm are minority class "p" and majority class "n" with the number of features j. The output of the algorithm will be the average measures such as AUC, Precision, F-measure, TP rate and TN rate produced by the IKM method. The algorithm is mainly divided into two phases: External Selection Phase and Clustering Phase. In the External Selection phase, the imbalanced dataset is divided into majority, minority subclasses and noisy, outliers are detected and removed from both the subclasses. Then the consistent instances I the minority set are replicated by both synthetic and hybridation techniques. In the clustering phase the so formed datasets is applied to clustering algorithm K-means and evaluation metrics are measured. .

V. DATASETS

In the study, we have considered 12 binary data-sets which have been collected from the KEEL [50] and UCI [51] 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 155 to 3196, and the imbalance ratio is up to 3.85. This way, we have different 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.

Table 1 Summary of benchmark imbalanced datasets

S.no	Datasets	# Ex.	# Atts.	Class (_,+)	IR
1.	Breast	268	9	(recurrence; no-recurrence)	2.37
2.	Breast_w	699	9	(benign; malignant)	1.90
3.	Colic	368	22	(yes; no)	1.71
4.	Credit-g	1000	21	(good; bad)	2.33
5.	Diabetes	7688		(tested-potv; tested-negtv)	1.87
6.	Heart-c	30314		(<50,>50_1)	1.19
7.	Heart-h	294	14	(<50,>50_1)	1.77
8.	Heart-stat	270	14	(absent, present)	1.25
9.	Hepatitis	155	19	(die; live)	3.85
10.	Ionosphere	35134		(b;g)	1.79
11.	Kr-vs-kp	3196	37	(won; nowin)	1.09
12.	Sonar	208	60	(rock ; mine)	1.15

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 [52] so that any interested researcher can reproduce the experimental study.

VI. Comparison of Algorithms and Experimental Setup

This section describes the algorithms used in the experimental study and their parameter settings, which are obtained from the KEEL [50] and WEKA [51] software tools. 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. The parameters used for the experimental study in all clustering methods are the optimal values from the tenfold cross-validation, and they are now detailed.

Table 2Experimental Settings for standard clustering algorithms

Algorithm	Parameter	Value
K-Means	distance function	Euclidean
	max iterations	500
	Number of clusters	2
Density	cluster to wrap	k-means
	minstddev	1.0E-6
FF	number of clusters	2
EM	max iterations	100
	minstddev	1.0E-6
	Number of clusters	2
Hierarchical	distance function	Euclidean
	Number of clusters	2

K-Means: K-means clustering Density: Density based clustering
FF: Farthest First clustering EM: Expectation Maximization
Hier: Hierarchical clusteringclustering

VII. Experimental Results

In this section, 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.

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

Datasets	K-Means	Density	FF	EM	Hier	IKM
Breast	54.26±10.84•	53.66±10.74•	65.63±9.54○	49.29±8.10•	70.05±1.57○	55.78±11.87
Breast_w	95.82±2.26	96.22±2.19○	84.94±6.96•	93.75±2.79•	65.52±0.44•	95.36±2.19
Colic	60.57±11.89•	65.30±10.85•	58.67±9.91•	66.13±7.11•	63.05±1.13•	68.07±9.07
Credit-g	55.86±6.77•	56.37±6.72•	62.41±6.56○	60.60±5.33○	70.00±0.00○	57.08±5.99
Diabetes	65.42±5.87○	65.60±5.68○	65.16±3.42○	64.67±5.74○	65.11±0.34○	63.13±5.92
Heart-c	77.68±9.32	80.94±8.02○	68.77±10.88○	80.28±7.82○	54.15±2.06•	77.94±9.81
Heart-h	77.82±10.54•	80.04±8.68•	66.70±12.23•	81.01±6.51•	63.95±1.36•	83.86±6.99
Heart-stat	74.39±11.42•	75.35±11.60•	66.85±10.87•	81.78±6.55○	55.54±1.60•	77.38±11.49
Hepatitis	71.09±12.58•	73.15±12.16•	72.14±12.77•	73.83±10.53•	79.38±2.26○	74.46±11.04
Ionosphere	70.80±6.71•	73.06±6.35•	62.75±6.65•	73.08±6.47○	64.10±1.35•	71.01±8.06
Kr-vs-kp	54.72±4.77•	54.19±4.93•	53.37±3.80•	59.98±3.48○	52.13±0.65•	55.04±4.00
Sonar	52.43±10.28	50.12±10.40•	50.94±8.28•	49.59±9.55•	51.78±3.41•	52.73±11.13

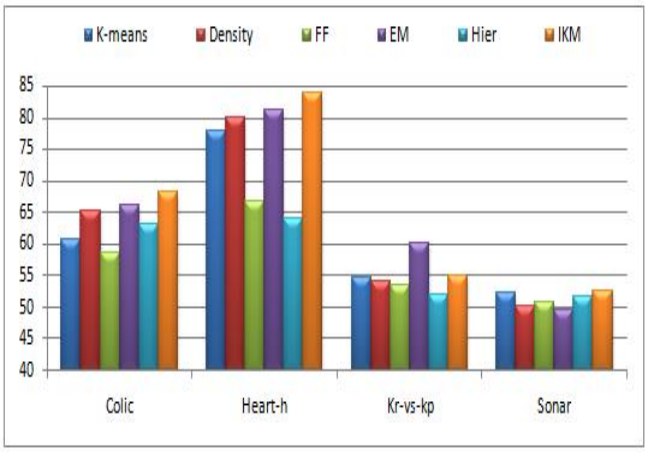

Fig. 1 Test results of Accuracy on K-means, Density, FF, EM, Hier and IKM for Colic, Heart-h, Kr-vs-kp and Sonar Datasets.

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

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.519±0.109●	0.508±0.103●	0.574±0.099○	0.500±0.098●	0.499±0.007●	0.552±0.116
Breast_w	0.950±0.027●	0.966±0.021●	0.785±0.098●	0.951±0.022●	0.500±0.000●	0.953±0.022
Colic	0.628±0.108●	0.678±0.092●	0.570±0.114●	0.691±0.068○	0.500±0.000●	0.689±0.086
Credit-g	0.534±0.067●	0.535±0.066●	0.521±0.057●	0.567±0.051●	0.500±0.000●	0.569±0.059
Diabetes	0.608±0.067●	0.617±0.068●	0.520±0.044●	0.670±0.070○	0.502±0.006●	0.625±0.059
Heart-c	0.775±0.093●	0.808±0.081○	0.683±0.111●	0.804±0.078○	0.500±0.000●	0.780±0.098
Heart-h	0.775±0.101●	0.795±0.096●	0.614±0.138●	0.792±0.073●	0.500±0.000●	0.837±0.070
Heart-stat	0.746±0.114●	0.756±0.114●	0.662±0.105●	0.851±0.068○	0.504±0.013●	0.770±0.116
Hepatitis	0.753±0.136●	0.781±0.122○	0.670±0.163●	0.800±0.101○	0.500±0.000●	0.758±0.114
Ionosphere	0.706±0.080●	0.743±0.079○	0.530±0.067●	0.771±0.058○	0.500±0.000●	0.709±0.080
Kr-vs-kp	0.544±0.046●	0.540±0.046●	0.531±0.039●	0.588±0.032○	0.500±0.001●	0.550±0.040
Sonar	0.521±0.103●	0.499±0.104●	0.513±0.082●	0.497±0.096●	0.500±0.000●	0.527±0.111

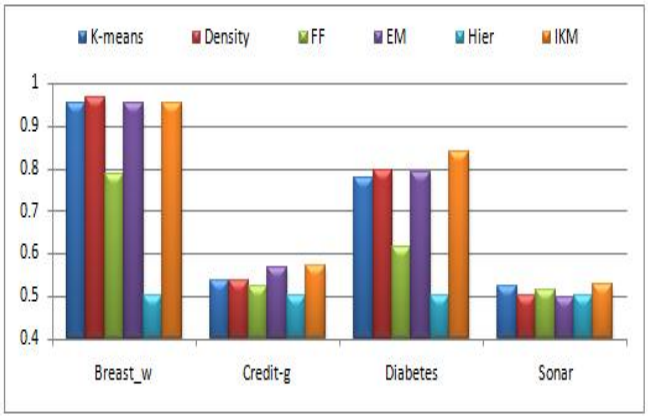

Fig. 1 Test results of Accuracy on K-means, Density, FF, EM, Hier and IKM for Breast_w, Credit-g, Diabetes, and Sonar Datasets.

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

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.718±0.090○	0.709±0.083○	0.747±0.061○	0.707±0.088○	0.702±0.014○	0.693±0.105
Breast_w	0.961±0.024○	0.989±0.015○	0.823±0.071●	0.998±0.007○	0.655±0.004●	0.936±0.032
Colic	0.784±0.107●	0.821±0.083○	0.719±0.120●	0.838±0.069○	0.630±0.011●	0.798±0.110
Credit-g	0.727±0.052○	0.727±0.053○	0.712±0.035○	0.746±0.036○	0.700±0.000○	0.667±0.058
Diabetes	0.725±0.047○	0.734±0.051○	0.665±0.044○	0.821±0.072○	0.652±0.004○	
	0.628±0.048	Heart-c	0.797±0.094○	0.831±0.086○	0.726±0.129●	0.843±0.082○
	0.526±0.101●	0.764±0.103				
Heart-h	0.867±0.084○	0.877±0.087○	0.735±0.119●	0.852±0.061○	0.640±0.014●	0.834±0.071
Heart-stat	0.806±0.116○	0.808±0.116○	0.706±0.113●	0.839±0.078○	0.549±0.068●	0.744±0.149
Hepatitis	0.426±0.150●	0.453±0.156○	0.405±0.233●	0.457±0.136●	0.000±0.000●	0.613±0.153
Ionosphere	0.557±0.147●	0.573±0.145●	0.309±0.369●	0.586±0.068●	0.000±0.000●	0.717±0.112
Kr-vs-kp	0.561±0.041○	0.559±0.042○	0.581±0.082○	0.579±0.022○	0.512±0.073●	0.546±0.040
Sonar	0.493±0.133●	0.460±0.140●	0.422±0.238●	0.459±0.108●	0.110±0.198●	0.530±0.128

Table 6 Summary of tenfold cross validation performance for Recallon all the datasets

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.577±0.153○	0.579±0.159○	0.776±0.137○	0.482±0.094●	1.000±0.004○	0.573±0.169
Breast_w	0.976±0.022	0.953±0.029●	0.992±0.033○	0.907±0.042●	1.000±0.000○	0.976±0.021
Colic	0.541±0.231●	0.582±0.195○	0.635±0.229○	0.576±0.100○	1.000±0.000○	0.561±0.221
Credit-g	0.595±0.108○	0.606±0.112○	0.778±0.121○	0.664±0.081○	1.000±0.000○	
	0.579±0.118	Diabetes	0.760±0.091○	0.747±0.089○	0.957±0.106○	0.594±0.075●
	1.000±0.000○	0.742±0.122				
Heart-c	0.798±0.124●	0.826±0.104○	0.733±0.187●	0.791±0.113●	0.956±0.184○	0.811±0.121
Heart-h	0.786±0.171●	0.815±0.126●	0.810±0.208●	0.856±0.073●	1.000±0.000○	0.872±0.106
Heart-stat	0.724±0.157●	0.737±0.169●	0.723±0.230●	0.842±0.086○	0.985±0.122○	0.740±0.174
Hepatitis	0.824±0.225○	0.865±0.190○	0.583±0.319●	0.906±0.153○	0.000±0.000●	0.799±0.194
Ionosphere	0.702±0.187●	0.787±0.195○	0.185±0.284●	0.912±0.074○	0.000±0.000●	0.732±0.160
Kr-vs-kp	0.624±0.197○	0.582±0.199●	0.600±0.312●	0.847±0.100○	0.980±0.141○	0.620±0.196
Sonar	0.471±0.201●	0.471±0.215●	0.524±0.392○	0.525±0.194○	0.235±0.425●	0.519±0.180

Table 7 Summary of tenfold cross validation performance for F-measure on all the datasets

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.630±0.112○	0.627±0.112○	0.755±0.087○	0.569±0.082●	0.825±0.010○	0.617±0.129
Breast_w	0.968±0.017○	0.971±0.017○	0.898±0.045●	0.950±0.024●	0.792±0.003●	0.955±0.021
Colic	0.608±0.162●	0.662±0.137○	0.638±0.141○	0.678±0.082○	0.773±0.008○	0.629±0.146
Credit-g	0.649±0.074○	0.655±0.074○	0.739±0.066○	0.700±0.052○	0.824±0.000○	0.613±0.076
Diabetes	0.739±0.053○	0.737±0.050○	0.779±0.049○	0.685±0.055○	0.789±0.003○	
	0.676±0.078	Heart-c	0.793±0.098○	0.824±0.081○	0.711±0.116●	0.811±0.081○
	0.680±0.130●	0.783±0.100				
Heart-h	0.810±0.110●	0.836±0.079●	0.745±0.129●	0.852±0.052○	0.780±0.010●	0.848±0.080
Heart-stat	0.752±0.125○	0.761±0.134○	0.691±0.144●	0.837±0.059○	0.705±0.087●	0.732±0.148
Hepatitis	0.549±0.157●	0.582±0.153●	0.451±0.226●	0.598±0.131●	0.000±0.000●	0.679±0.141
Ionosphere	0.617±0.155●	0.660±0.157●	0.173±0.222●	0.711±0.059	0.000±0.000●	0.710±0.130
Kr-vs-kp	0.573±0.111○	0.553±0.111●	0.524±0.183●	0.686±0.052○	0.672±0.097○	0.563±0.109
Sonar	0.462±0.149●	0.447±0.159●	0.414±0.257●	0.480±0.133●	0.149±0.270●	0.512±0.136

Table 8 Summary of tenfold cross validation performance for Specificity on all the datasets

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.630±0.112○	0.437±0.194●	0.373±0.190●	0.518±0.182●	0.000±0.000●	0.531±0.185
Breast_w	0.968±0.017○	0.979±0.029○	0.578±0.196●	0.996±0.012○	0.000±0.000●	0.931±0.036
Colic	0.608±0.162●	0.773±0.157●	0.506±0.336●	0.807±0.093●	0.000±0.000●	0.816±0.162
Credit-g	0.649±0.074○	0.464±0.149●	0.264±0.145●	0.470±0.100●	0.000±0.000●	0.559±0.132
Diabetes	0.739±0.053○	0.487±0.155●	0.082±0.163●	0.746±0.145○	0.000±0.000●	
	0.507±0.116	Heart-c	0.793±0.098○	0.789±0.122○	0.633±0.225●	0.817±0.107○
	0.035±0.184●	0.748±0.119				
Heart-h	0.810±0.110○	0.775±0.193●	0.418±0.322●	0.728±0.127●	0.000±0.000●	
	0.801±0.105	Heart-stat	0.752±0.125●	0.775±0.155●	0.601±0.220●	0.787±0.120●
	0.019±0.123●	0.799±0.149				
Hepatitis	0.549±0.157○	0.696±0.143●	0.757±0.154○	0.695±0.125●	1.000±0.000○	0.716±0.148
Ionosphere	0.617±0.155●	0.699±0.110○	0.875±0.215○	0.629±0.098●	1.000±0.000○	0.687±0.153
Kr-vs-kp	0.573±0.111○	0.497±0.167○	0.461±0.330○	0.330±0.057●	0.020±0.141●	0.480±0.174
Sonar	0.462±0.149●	0.528±0.198●	0.502±0.357●	0.470±0.171●	0.768±0.420○	0.535±0.186

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

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.539±0.199○	0.563±0.194○	0.627±0.190○	0.482±0.182○	1.000±0.000○	0.469±0.185
Breast_w	0.076±0.049○	0.021±0.029●	0.422±0.196○	0.004±0.012●	1.000±0.000○	0.069±0.036
Colic	0.285±0.240●	0.227±0.157●	0.494±0.336●	0.193±0.093●	1.000±0.000○	0.814±0.162
Credit-g	0.526±0.143○	0.536±0.149○	0.736±0.145○	0.530±0.100○	1.000±0.000○	0.441±0.132
Diabetes	0.544±0.145○	0.513±0.155○	0.918±0.163○	0.254±0.145●	1.000±0.000○	0.493±0.116
Heart-c	0.249±0.148●	0.211±0.122●	0.367±0.225○	0.183±0.107●	0.965±0.184○	0.252±0.119
Heart-h	0.236±0.190○	0.225±0.193○	0.582±0.322○	0.272±0.127○	1.000±0.000○	0.199±0.105
Heart-stat	0.231±0.168○	0.225±0.155○	0.399±0.220○	0.213±0.120○	0.981±0.123○	0.201±0.149
Hepatitis	0.318±0.146○	0.304±0.143○	0.243±0.154●	0.305±0.125○	0.000±0.000●	0.284±0.148
Ionosphere	0.289±0.110●	0.301±0.110●	0.125±0.215●	0.371±0.098○	0.000±0.000●	0.313±0.153
Kr-vs-kp	0.536±0.181○	0.503±0.167●	0.539±0.330○	0.670±0.057○	0.980±0.141○	0.520±0.174
Sonar	0.429±0.195●	0.472±0.198○	0.498±0.357○	0.530±0.171○	0.232±0.420●	0.465±0.186

Table 10 Summary of tenfold cross validation performance for FN Rate on all the datasets

Datasets	K-Means	Density	FF	EM	Hire	IKM
Breast	0.423±0.153●	0.421±0.159●	0.224±0.137●	0.518±0.094○	0.000±0.004●	0.427±0.169
Breast_w	0.024±0.022	0.047±0.029○	0.008±0.033●	0.093±0.042○	0.000±0.000●	0.024±0.021
Colic	0.459±0.231○	0.418±0.195●	0.365±0.229●	0.424±0.100●	0.000±0.000●	0.439±0.221
Credit-g	0.405±0.108●	0.394±0.112●	0.222±0.121●	0.336±0.081●	0.000±0.000●	0.421±0.118
Diabetes	0.240±0.091●	0.254±0.089●	0.043±0.106●	0.406±0.075○	0.000±0.000●	
	0.258±0.122	Heart-c	0.202±0.124○	0.174±0.104●	0.267±0.187○	0.209±0.113○
	0.035±0.184●	0.189±0.121				
Heart-h	0.214±0.171○	0.185±0.126○	0.190±0.208○	0.144±0.073○	0.000±0.000●	
	0.128±0.106	Heart-stat	0.276±0.157○	0.263±0.169○	0.277±0.230○	0.158±0.086●
	0.015±0.122●	0.260±0.174	Hepatitis	0.176±0.225●	0.135±0.190●	0.417±0.319○
	0.094±0.153●	1.000±0.000○	0.201±0.194			
Ionosphere	0.298±0.187○	0.213±0.195●	0.815±0.284○	0.088±0.074●	1.000±0.000○	0.268±0.160
Kr-vs-kp	0.376±0.197●	0.418±0.199○	0.400±0.312○	0.153±0.100●	0.020±0.141●	0.380±0.196
Sonar	0.529±0.201○	0.529±0.215○	0.476±0.392●	0.475±0.194●	0.765±0.425○	0.481±0.180

The clustering valuations were conducted on twelve wide lyused datasets. Theserea lworld multi-dimensional datasetsareusedtoverifytheproposedclustering method.
Table 3, 4, 5, 6, 7, 8, 9 and 10 reports the results of Accuracy, AUC, Precision, Recall, F-measure, Specificity, FP Rateand FN Rate respectively for all the twelve datasets from UCI. A two-tailed corrected resampled paired t-test [46] 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 IKM 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 IKM 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 IKM.

Finally, we can make a global analysis of results combining theresults offered by Tables from 3–10:

- Our proposals, IKMis 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;eK-means, Density, FF, EM and Hier. However, they are not able to outperform IKM. In this sense, the competitive edge of IKM can be seen.

Considering that IKM behaves similarly or not effective than K-means shows the unique properties of the datasets where there is scope of improvement in majority subset and not in minority subset. Our IKM can mainly focus on improvements in minority subset which is not effective for some unique property datasets.

Table 11 Summary of experimental results for Improved K-means

Results	Systems	Wins	Ties	Losses
Accuracy	IKM versus K-means	7	4	1
	IKM versus Density	9	0	3
	IKM versus FF	8	0	4
	IKM versus EM	6	0	6
	IKM versus Hier	8	0	4
AUC	IKM versus K-means	12	0	0
	IKM versus Density	9	0	3
	IKM versus FF	11	0	1
	IKM versus EM	6	0	6
	IKM versus Hier	12	0	0
Precision	IKM versus K-means	4	0	8
	IKM versus Density	2	0	10
	IKM versus FF	8	0	4
	IKM versus EM	3	0	9
	IKM versus Hier	9	0	3
Recall	IKM versus K-means	6	0	6
	IKM versus Density	5	0	7
	IKM versus FF	6	0	6
	IKM versus EM	5	0	7
	IKM versus Hier	3	0	9
F-measure	IKM versus K-means	5	0	7
	IKM versus Density	5	0	7
	IKM versus FF	8	0	4
	IKM versus EM	4	0	8
	IKM versus Hier	7	0	5
TN Rate	IKM versus K-means	4	0	8
	IKM versus Density	8	0	4
	IKM versus FF	9	0	3
	IKM versus EM	9	0	3
	IKM versus Hier	9	0	3
FP Rate	IKM versus K-means	4	0	8
	IKM versus Density	5	0	7
	IKM versus FF	3	0	9
	IKM versus EM	4	0	8
	IKM versus Hier	3	0	9
FN Rate	IKM versus K-means	5	1	6
	IKM versus Density	7	0	5
	IKM versus FF	6	0	6
	IKM versus EM	7	0	5
	IKM versus Hier	9	0	3

The summary of experimental results of IKM on all the compared clustering algorithms is shown in Table 11. The results show that proposed IKM clustering algorithm is at least as effective as and at times more effective than K-means, Density, FF, EM and Hierarchical clustering algorithms. IKM compared with accuracy onK-means wins on 7 dataset and ties on 4 datasets and loses ononly 1 dataset. The performance of IKM compared withDensity based clustering wins on 9 datasetsand losses ononly 3 datasets.The performance of IKM comparedwith FF wins on 8 datasets andlosses on 4 datasets. The validation of IKM on EM wins on 6 datasets and losses on 6 datasets. However, performance of IKM on Hierarchical clustering wins on 8 datasets and losses on 4 datasets. The AUC, Precision, Recall, F-measure, TN Rate, FP Rate and FN Rate measure have shown to perform well with respect to IKM.

The strengths of our model are that IKM only over-sample the most prominent examples recursively thereby strengthens the minority class. One more point to consider is our method tries to remove the most misclassified instances from both majority and minority set. Firstly, the removal of some weak instances from majority set will not harm the dataset; in fact it will reduce the root cause of our problem of class imbalance as a whole by reducing majority samples in a small proportion. Second, the removal of weak instances from the minority set will again help in better generation of synthetic examples of both same and hybrid type.

Finally, we can say that IKM are one of the best alternatives to handle class imbalance problems effectively.This experimental study supports the conclusion that the a prominent recursive oversampling approach can improve the CIL behavior when dealing with imbalanced data-sets, as it has helped the IKM methods to be the best performingalgorithms when compared with four classical and well-known algorithms: K-means, Density, FF, EM and a well-established Hierarchical algorithm.

VIII. Conclusion

In this paper, a novel clustering algorithm for imbalanced distributed data hasbeen 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 IKM considerably reduces the uniform effect while retaining or improving the clustering measure when compared with benchmark methods. In fact, the proposedmethod may also be useful as a frame work for data sources for better clustering measures.

REFERENCES:

[1] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*.Reading, MA: Addison-Wesley, 2005.

[2] Z. X. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weightingin *k*-means type clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*,vol. 27, no. 5, pp. 657–668, May 2005.

[3] E. Y. Chan, W. K. Ching, M. K. Ng, and Z. X. Huang, "An optimizationalgorithm for clustering using weighted dissimilarity measures," *PatternRecognit.*, vol. 37, no. 5, pp. 943–952, 2004.

[4] Y. H. Qian, J. Y. Liang, W. Pedrycz, and C. Y. Dang, "Positive approximation:An accelerator for attribute reduction in rough set theory," *Artif.Intell.*, vol. 174, no. 5–6, pp. 597–618, 2010.

[5] L. P. Jing, M. K. Ng, and Z. X. Huang, "An entropy weighting *k*-meansalgorithm for subspace clustering of high-dimensional sparse data," *IEEETrans. Knowl. Data Eng.*, vol. 19, no. 8, pp. 1026–1041, Aug. 2007.

[6] J. S. Zhang and Y. W. Leung, "Robust clustering by pruning outliers,"*IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 33, no. 6, pp. 983–998,Dec. 2003.

[7] A. Zhou, F. Cao, Y. Yan, C. Sha, and X. He, "Distributed data streamclustering: A fast EM-based approach," in *Proc. 23rd Int. Conf. DataEng.*, 2007, pp. 736–745.

[8] M. Breunig, H. P. Kriegel, R. Ng, and J. Sander, "LOF: Identifying densitybased local outliers," in *Proc. Int. Conf. ACM Special Interest GroupManag. Data*, 2000, pp. 427–438.

[9] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithmfor discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. ACM Special Interest Group Knowl. Discovery Data Mining*,1996, pp. 226–231.

[10] P. Bradley, U. Fayyad, and C. Reina, "Scaling clustering algorithms tolarge databases," in *Proc. 4th Int. Conf. ACM Special Interest GroupKnowl. Discovery Data Mining*, 1998, pp. 9–15.

[11] F. Murtagh, "Clustering massive data sets," in *Handbook of Massive DataSets*. Norwell, MA: Kluwer, 2000.

[12] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient dataclustering method for very large databases," in *Proc. ACMSpecial InterestGroup Manag. Data*, 1996, pp. 103–114.

[13] Z. X. Huang, "Extensions to the *k*-means algorithm for clustering largedata sets with categorical values," *Data Mining Knowl. Discov.*, vol. 2,no. 3, pp. 283–304, 1998.

[14] F. Y. Cao, J. Y. Liang, L. Bai, X. Zhao, and C. Dang, "A frameworkfor clustering categorical time-evolving data," *IEEE Trans. Fuzzy Syst.*,vol. 18, no. 5, pp. 872–882, Oct. 2010.

[15] J. C. Bezdek, "A physical interpretation of Fuzzy ISODATA," *IEEETrans. Syst., Man, Cybern.*, vol. SMC-6, no. 5, pp. 387–390, May 1976.

[16] L. O. Hall, A. M. Bensaid, and L. P. Clarke, "A comparison of neural networkand fuzzy clustering techniques in segmenting magnetic resonanceimages of the brain," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 672–682,Sep. 1992.

[17] R. L. Cannon, J. V. Dave, and J. C. Bezdek, "Efficient implementationof the fuzzy c-means clustering algorithms," *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. PAMI-8, no. 2, pp. 248–255, Mar. 1986.

[18] J. Yu and M. S. Yang, "Optimality test for generalized FCM and itsapplication to parameter selection," *IEEE Trans. Fuzzy Systems*, vol. 13,no. 1, pp. 164–176, Feb. 2005.

[19] F. Y. Cao, J. Y. Liang, and G. Jiang, "An initialization method for the*k*-means algorithm using neighborhood model," *Comput. Math. Appl.*,vol. 58, no. 3, pp. 474–483, 2009.

[20] M. Laszlo and S. Mukherjee, "A genetic algorithm using hyper-quadtreesfor low-dimensional k-means clustering," *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. 28, no. 4, pp. 533–543, Apr. 2006.

[21] D. Arthur and S. Vassilvitskii, "K-means++: the advantages of carefulseeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algo.*, 2007,pp. 1027–1035.

[22] A. Likas, M. Vlassis, and J. Verbeek, "The global *k*-means clusteringalgorithm," *Pattern Recognit.*, vol. 35, no. 2, pp. 451–461, 2003.

[23] A. M. Bagirov, "Modified global *k*-means algorithm for minimum sum-ofsquaresclustering problems," *Pattern Recognit.*, vol. 41, no. 10, pp. 3192–3199, 2008.

[24] Z. C. Lai and T. J. Huang, "Fast global *k*-means clustering using clustermembership and inequality," *Pattern Recognit.*, vol. 43, no. 5, pp. 1954–1963, 2010.

[25] G. Hamerly and C. Elkan, "Learning the *k* in *k*-means," in *Proc. 17thAnn. Conf. Neural Inf. Process. Syst.*, Dec. 2003, pp. 1–8.

[26] J. J. Li, M. K. Ng, Y. M. Cheng, and Z. H. Huang, "Agglomerative fuzzy*k*-means clustering algorithm with selection of number of clusters," *IEEETrans. Knowl. Data Eng.*, vol. 20, no. 11, pp. 1519–1534, Nov. 2008.

[27] M. Halkidi and M. Vazirgiannis, "A density-based cluster validity approachusing multi-representatives," *Pattern Recognit. Lett.*, vol. 29,pp. 773–786, 2008.

[28] H. Xiong, J. J. Wu, and J. Chen, "K-means clustering versus validationmeasures: A data-distribution perspective," *IEEE Trans. Syst., Man, Cybern.B, Cybern.*, vol. 39, no. 2, pp. 318–331, Apr. 2009.

[29] W.-Z. Lu and D.Wang, "Ground-level ozone prediction by support vector machine approach with a cost-sensitive classification scheme," *Sci. Total. Enviro.*, vol. 395, no. 2-3, pp. 109–116, 2008.

[30] Y.-M. Huang, C.-M. Hung, and H. C. Jiau, "Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem," *Nonlinear Anal. R. World Appl.*, vol. 7, no. 4, pp. 720–747, 2006.

[31] D. Cieslak, N. Chawla, and A. Striegel, "Combating imbalance in network intrusion datasets," in *IEEE Int. Conf. Granular Comput.*, 2006, pp. 732–737.

[32] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural Netw.*, vol. 21, no. 2–3, pp. 427–436, 2008.

[33] A. Freitas, A. Costa-Pereira, and P. Brazdil, "Cost-sensitive decision trees applied to medical data," in *Data Warehousing Knowl. Discov. (Lecture Notes Series in Computer Science)*, I. Song, J. Eder, and T. Nguyen, Eds.,

[34] K.Kilic¸,O¨ zgeUncu and I. B. Tu¨rksen, "Comparison of different strategies of utilizing fuzzy clustering in structure identification," *Inf. Sci.*, vol. 177, no. 23, pp. 5153–5162, 2007.

[35] M. E. Celebi, H. A. Kingravi, B. Uddin, H. Iyatomi, Y. A. Aslandogan, W. V. Stoecker, and R. H. Moss, "A methodological approach to the classification of dermoscopy images," *Comput.Med. Imag. Grap.*, vol. 31, no. 6, pp. 362–373, 2007.

[36] X. Peng and I. King, "Robust BMPM training based on second-order cone programming and its application in medical diagnosis," *Neural Netw.*, vol. 21, no. 2–3, pp. 450–457, 2008.Berlin/Heidelberg, Germany: Springer, 2007, vol. 4654, pp. 303–312.

[37] S. Guha, R. Rastogi, and K. Shim, "Cure: An efficient clustering algorithm for large databases," in *Proc. Int. Conf. ACM Special Interest Group Manag. Data*, 1998, pp. 73–84.

[38] M. H. Liu, X. D. Jiang, and A. C. Kot, "A multi-prototype clustering algorithm," *Pattern Recognit.*, vol. 42, pp. 689–698, 2009.

[39] Haitaoxiang , Yi yang, Shouxiangzhao. "Local Clustering Ensemble Learning Method Based on Improved AdaBoost for Rare Class Analysis", Journal of Computational Information Systems 8: 4 (2012) 1783{1790, pp,no:1783 – 1790.

[40] AmuthanPrabakarMuniyandi, R. Rajeswari, R. Rajaram. Network Anomaly Detection by Cascading K-Means Clustering and C4.5 Decision Tree algorithm, International Conference on Communication Technology and System Design 2011, Procedia Engineering 30 (2012) 174 – 182.

[41] Li Xuan, Chen Zhigang, Yang Fan. "Exploring of clustering algorithm on class-imbalanced data".

[42] C. Bouras, V. Tsogkas, A clustering technique for news articles using WordNet, Knowl. Based Syst. (2012), http://dx.doi.org/10.1016/j.knosys.2012.06.015.

[43] P.Y. Mok, H.Q.Huang,Y.L.Kwok,J.S.Au. "A robust adaptive clustering analysis method for automatic identification of clusters", Pattern Recognition 45 (2012) 3017–3033.
[44] Luis A. Leiva, Enrique Vidal." Warped K-Means: An algorithm to cluster sequentially-distributed data", Information Sciences 237 (2013) 196–210.
[45] M.F.Jaing, S.S.Tseng and C.M. Su, "Two Phase Clustering Process for Outlier Detection", pattern recognition letters 22 (2001) pp no: 691-700.
[46] Jie Cao, ZhiangWu, JunjieWu and WenjieLiu, "Towards information-theoretic K-means clustering for image indexing", Signal Processing 93 (2013) 2026–2037.
[47] Mignotte, M. A de-texturing and spatially constrained K-means approach for image segmentation. Pattern Recognition Lett. (2010), doi:10.1016/j.patrec.2010.09.016
[48] O. Maimon, and L. Rokach, *Data mining and knowledge discovery handbook*, Berlin: Springer, 2010.
[49] J. R. Quinlan, *C4.5: Programs for Machine Learning*, 1st ed. San Mateo, CA: Morgan Kaufmann Publishers, 1993.
[50] Keel
[51] Witten, I.H. and Frank, E. (2005) Data Mining:Practical machine learning tools and techniques.2nd edition Morgan Kaufmann, San Francisco.
[52] Blake C, Merz CJ (2000) UCI repository of machine learning databases. Machine-readable data repository. Department of Information and Computer Science, University of California at Irvine, Irvine. http://www.ics.uci.edu/mlearn/MLRepository.html