Classification of Imbalanced Data Using a Modified Fuzzy-Neighbor Weighted Approach

Harshita Patel1* Ghanshyam Singh Thakur1

1 Department of Mathematics and Computer Applications, Maulana Azad National Institute of Technology, Bhopal-462003, India
* Corresponding author’s Email: hpatel.sati@gmail.com

Abstract: Classification of imbalanced datasets is one of the widely explored challenges of the decade. The imbalance occurs in many real world datasets due to uneven distribution of data into classes, i.e. one class has more instances while others have a few that results in the biased performances of traditional classifiers towards the majority class with large number of instances and ignorance of other classes with less data. Many solutions have been proposed to deal with this issue in various crisp and fuzzy methods. This paper proposes a new hybrid fuzzy weighted nearest neighbor approach to find better overall classification performance for both minority and majority classes of imbalanced data. Benefits of neighbor weighted K nearest neighbor approach i.e. assignment of large weights to small classes and small weights to large classes are merged with fuzzy logic. Fuzzy classification helps in classifying objects more adequately as it determines that how much an object belongs to a class. Experimental results exhibit the improvements in classification of imbalanced data of different imbalance ratios in comparison with other methods.

Keywords: Imbalanced data, K nearest neighbor, Fuzzy K nearest neighbor, Classification.

1. Introduction

In today’s fast pace scientific world, data generation is accelerated with exponential speed with availability of highly techno-friendly equipment from almost all areas of society. This brings various data mining opportunities with many challenges [1, 2]. Classification of imbalanced data has drawn significant attention of researchers from data mining and machine learning community. The issue got the interest due to existence of imbalance in real world datasets and inability of traditional classifiers to classify it correctly. We can understand the term with unequal or uneven distribution of data into classes. One class with more instances is known as majority class and other with fewer instances as minority class. Minority class with large number of instances dominate the classification results and minority classes are misclassified, even most of the time they are of most interest. The 10 challenging problems in data mining research includes the learning from imbalanced data [3] and this issue is coming under new trends of data mining [4]. Raeder mentioned it specifically again in [5]. Examples of real world imbalanced datasets are Medical Diagnosis [6, 7], Oil-spill Detection [8], Credit Card Fraud Detection, Network Intrusion, Text Categorization, Helicopter Gearbox Fault Monitoring [9] etc.

Four popular solutions are available to deal with this issue are (a) balance the data by re-sampling, (b) modification in traditional classification algorithms, (c) cost sensitive methods and (d) ensemble approaches [9]. In this paper we adopt second solution of modification of traditional classification algorithm and we study the nearest neighbor approach with its weighted and fuzzy variant to find proper classification results of imbalanced data. K nearest neighbor is a one of the top data mining algorithms [10] that is easily understandable, robust, and easily programmable. Error rate of K nearest neighbor is bounded above by twice the Bayes error.
rate [11, 12]. The proposed method gives improved version of neighbor weighted K nearest neighbor with combination of fuzzy logic. Weights from neighbor weighted K nearest neighbor algorithm become stronger with the use of fuzzy concepts by getting opportunity of classification on soft boundary values instead of crisp’s hard calculations. The membership function we used in this paper is a basic membership function given by Keller et. al. [26] designed for binary fuzzy classification. It is a benchmark membership function used and applicable in many nearest neighbor classification tasks. In results we can observe the better performance of proposed approach on different evaluation measures over the neighbor weighted and other algorithm.

The organization of the paper is as follows: Related work is shown in second section of paper which is followed by preliminaries in section three. Fourth section is dedicated for our proposed method. Section five contains experimental and results for the proposed approach and the last sixth section is providing concluding remarks with future scope.

2. Related Work

Conventional classification strategies assume the quantity of data into all classes is equal by default, so they are insensitive to the imbalanced distribution of the data. In results they lead to misclassification for the small classes having few elements in comparison with the large classes. In other proposed solutions re-sampling is very popular. Though it presents good classification of imbalanced data but in some manner it affects original distribution of data such as duplication of information in oversampling and data loss (information loss) in under-sampling techniques [9]. Cost sensitive approaches are also not applicable where cost is not given. Algorithm modification with almost all available classification algorithms is done by many researchers for imbalanced data. Modified K nearest neighbor approaches provide different solutions to deal with unequally distributed data.

This section includes related literature with the proposed concept and providing knowledge of the field. A single CCNND algorithm is proposed by Kriminger et. al. [13], they applied local geometric structure in data. This approach is applicable on multi class imbalance as well as allows classification for different degrees of imbalance. Tomasev et. al. [14] have worked on K nearest neighbor’s hubness effect on high dimensional data which is responsible for high misclassification. This misclassification occurs due to minority classes while in low and medium dimensional datasets misclassification occur due to majority classes. For cross project defect prediction (CPDP) Ryu et. al. [15] proposed an instance hybrid selection using nearest neighbor (HISNN). In such cases class imbalance exists in source and target project distribution.

Many weighted KNN are performing well on imbalanced datasets. Some are being discussed here. Liu et. al. [16] proposed the concept of class confident weights to sort out the imbalance issue of data. They found posterior probabilities by converting attribute prior probabilities to weight prototypes. Class based weighted nearest neighbor algorithm is proposed by Dubey et. al. [17] for imbalanced data classification. The calculation of weights was based on the distribution of nearest neighbor of unknown examples. Patel et. al. [18] proposed a hybrid neighbor weighted approach to deal with large and small weight assignment to minority and majority classes with large and small K respectively. A mathematical model was proposed by Ando [19] with convex optimization technique by given to learn class wise weighted nearest neighbor approach that maximize nonlinear performance measure for training data.

Beside all these practices, fuzzy solutions too have a good scope to cope up with imbalance problem. A few contributions are being discussed here. Fernandez et. al. [20] analyzed fuzzy rule based system for imbalanced datasets. For improved classification of different imbalance ratios they applied adaptive parametric conjunction operators. Han et. al. [21] proposed a nearest neighbor approach with fuzzy and rough properties to minimize a biasness generated due to majority class. To solve imbalance problem in categorical data Liu et. al. [22] proposed a coupled fuzzy K nearest neighbor approach for unequally distributed data with strong bonds among attribute, classes and instances. Important functions of this approach are assignment of sized membership, similarity calculation and integration. Ramentol et. al. [23] proposed a fuzzy rough weighted nearest neighbor algorithm for imbalanced data, they proposed six weight vectors and some indiscernibility relations with these weight vectors. Our proposed method is inspired from these works and giving an improved easily computable neighbor weighted solution with fuzzy K nearest neighbor.

3. Preliminaries

Basic knowledge of K-nearest neighbor algorithm [12][24][25], fuzzy K-nearest neighbor
algorithm [26] and neighbor weighted approach [27] help to understand better the proposed method. Euclidian distance is considered as distance measure for all nearest neighbors discussed here in this paper by default. Mathematical models for these preliminaries are explained in following subsections.

3.1 K Nearest Neighbor

The K-nearest neighbor algorithm finds nearest neighbors of an unknown instance \( q_u \) from training dataset on the basis of distance between \( q_u \) and all training instances then class label is assigned to \( q_u \) for the class having maximum neighbors.

Mathematical formulation could be understood in following equation:

\[
C(q_u) = \arg \max_{C \in [C_j]} \sum_{x \in S(q_u, K)} T(x, C)
\]

Here \( C(q_u) \) = class label of \( q_u \), to be predicted,
\( m = \) Number of classes,
\( S(q_u, K) = \) Set of \( K \) nearest neighbors of \( q_u \) and
\( T(x, C) = \begin{cases} 1 & \text{if } x \in C \\ 0 & \text{otherwise} \end{cases} \)

3.2 Neighbor Weighted K nearest Neighbor

Neighbor Weighted K nearest Neighbor approach for imbalanced text datasets is proposed by Tan [27] for. Idea behind this method is to assign large weights to small classes and small weights to large classes to minimize the biasness of the classifier headed for majority class and avoidance of large weight by Tan [27] for. Idea behind this method is to assign large weights to small classes and small weights to large classes to minimize the biasness of the classifier headed for majority class and avoidance of large weight.

After finding K nearest neighbors for query instance \( q_u \) from traditional K nearest neighbor method we find weights from following equation:

\[
W_j = \frac{1}{(N(C_j) / \text{Min}(N(C_j) | j = 1, 2))^{1/p}}
\]

\( p \) is an exponent and \( p > 1 \),

Here \( W_j = \) Weight for \( j^{th} \) class, to be predicted,
\( N(C) = \) Number of nearest neighbors of \( q_u \) belongs to class \( j \)

These weights are combined with traditional classification algorithm to find class label of a query instance \( q_u \).

\[
C(q_u) = \arg \max_{C \in [C_j]} W_i ( \sum_{x \in S(q_u, K)} T(x, C))
\]

(3)

3.3 Fuzzy K-Nearest Neighbor Algorithm

The K-nearest neighbor algorithm in fuzzy form finds memberships of data examples into classes instead of finding their complete belongingness. This improves the accuracy of classification.

Keller et. al. (1985) [26] proposed the following model for fuzzy memberships of training data instances into classes.

\[
\mu_c(x) = \begin{cases} 0.51 + (n_c / K) \times 0.49 & \text{If } C = m \\ (n_c / K) \times 0.49 & \text{otherwise} \end{cases}
\]

(4)

Here \( n_c = \) nearest neighbors of \( x \) from class \( C \)

\( \mu_c(x) = \) Membership of \( x \) into class \( C \)

And for memberships of test instance \( q \)

\[
\mu_c(q_u) = \frac{\sum_{i=1}^{K} \mu_c(1 / \| q_u - q_i \|^{2/(p-1)})}{\sum_{i=1}^{K} (1 / \| q_u - q_i \|^{2/(p-1)})}
\]

(5)

Where \( p \) is an integer and \( p > 1 \)

And \( q_i \) is nearest neighbor of \( q_u \), \( (i = 1, ..., K) \)

4. Proposed Method

The proposed approach is a fuzzy extension of neighbor weighted nearest neighbor method [27]. Neighbor weighted approach (further called as NWKNN in this paper) is a good weighting strategy for imbalanced data classification that was proposed for text data and more accurately classify the imbalanced data than traditional nearest neighbor approach by assigning large weights to small classes and vice versa. With fuzzy logic, performance of neighbor weighted approach improved as membership says that how much an instance belongs to a class instead of assigning whole class label in hard manner. We first find memberships of
all training instances in step one. Then in step two find out the nearest neighbors for query instance using traditional K nearest neighbor approach. Weights will be calculated for all classes in step three. In step four we find the membership of query instance and class label is assigned in step five on the basis of membership of query instance in all classes.

Algorithm (Fuzzy-NWKNN)

Input: Training Dataset \( D \), Query instance \( x_t \), Set of class labels \( C \) and Parameter \( K \).

Output: Class label of \( x_t \).

Step 1. Find memberships of training data into each class using

\[
\mu_{ci}(x) = \begin{cases} 
0.51 + \frac{(nc_i/K) * 0.49}{(nc_i/K) * 0.49} & \text{if } i = m \\
0.51 + \frac{(nc_i/K) * 0.49}{(nc_i/K) * 0.49} & \text{otherwise}
\end{cases}
\]

While taking \( \sum \mu_{ci}(x) = 1 \)

Step 2. Find K nearest Neighbor for \( x_t \)

Step 3. Obtain weights with

\[
W_j = \frac{1}{(N(C_i) / \min\{N(C_i) \mid j = 1,2\})^{1/p}}
\]

Step 4. Find class memberships of test instance \( x_t \) using following formula

\[
\mu_{ci}(x_t) = \frac{\sum_{j=1}^{K} W_j * \mu_{cj}}{\sum_{j=1}^{K} W_j}
\]

While taking \( \sum \mu_{ci}(x_t) = 1 \)

Step 5. Assign class label to test instance \( x_t \) by

\[
C_w(x_t) = \begin{cases} 
C_w & \text{if } \mu_{ci}(x_t) \geq 0.51 \\
\text{Random Assignment} & \text{Otherwise}
\end{cases}
\]

5. Experiments and Results

Experiments for proposed method are done for binary imbalanced datasets. All features are taken for classification, keeping into the mind that all features of datasets playing important role in their classification no specific feature selection method is applied. Also it becomes very ambiguous for imbalanced data to decide which attribute is of more interest. Following subsections are providing description of datasets, about evaluation measure and comparative outcomes of proposed method with its predecessor.

5.1 Datasets

Six datasets are taken from UCI [28] and KEEL [29] repositories. These datasets are of different natures like disease data, glass quality or radar signal’s data, all are imbalanced with uneven distribution of instances into classes. Brief description of data is given in table 1.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Source</th>
<th># Instances</th>
<th>Class(1/0)</th>
<th>#Attributes</th>
<th>Imbalance Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ionosphere</td>
<td>UCI</td>
<td>351</td>
<td>Bad/Good Radar Returns</td>
<td>34</td>
<td>1.79</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>KEEL</td>
<td>683</td>
<td>Positive/Negative</td>
<td>9</td>
<td>1.86</td>
</tr>
<tr>
<td>Phoneme</td>
<td>KEEL</td>
<td>5404</td>
<td>Oral/Nasal Sound</td>
<td>5</td>
<td>2.4</td>
</tr>
<tr>
<td>Vehicle0</td>
<td>KEEL</td>
<td>846</td>
<td>Positive/Negative</td>
<td>18</td>
<td>3.3</td>
</tr>
<tr>
<td>New-Thyroid</td>
<td>UCI</td>
<td>215</td>
<td>Positive/Negative</td>
<td>5</td>
<td>5.14</td>
</tr>
<tr>
<td>Glass2</td>
<td>KEEL</td>
<td>214</td>
<td>Positive/Negative</td>
<td>9</td>
<td>11.6</td>
</tr>
</tbody>
</table>

5.2 Evaluation Criteria

Accuracy is a common evaluation measure used to judge the performance of a classifier. But this seems inadequate while dealing with imbalanced datasets. F-measure, AUC and G-mean are some of the popular measures for evaluation of classifiers for such datasets. In this paper we are evaluating and comparing the performance of our proposed method with neighbor weighted approach on these evaluation criteria. To calculate the results of these measures confusion metric is needed, given as follows:
Table 2. Confusion Metric for Binary Classification

<table>
<thead>
<tr>
<th>Predictive/Actual Outcomes</th>
<th>Predicted Positive</th>
<th>Predicted Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual Positive</td>
<td>True Positive</td>
<td>False Positive</td>
</tr>
<tr>
<td>Actual Negative</td>
<td>False Negative</td>
<td>True Negative</td>
</tr>
</tbody>
</table>

True positive (TP): Actual positive instances correctly classified as positive,
False positive (FP): Actual positive instances incorrectly classified as negative,
True negatives (TN): Actual negative instances correctly classified as negative,
False negatives (FN): Actual negative instances incorrectly classified as positive.

Based on confusion metric evaluation measures could be understood in terms of following equations:

\[ \text{Precision} = \frac{TP}{TP + FP} \] (6)
\[ \text{Recall} = \frac{TP}{TP + FN} \] (7)
\[ TP_{Rate} = \frac{TP}{\text{Total } P} \] (8)
\[ FP_{Rate} = \frac{FP}{\text{Total } N} \] (9)
\[ F - \text{Measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \] (10)
\[ AUC = \frac{1 + TP_{Rate} - FP_{Rate}}{2} \] (11)
\[ G - \text{Mean} = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \] (12)

5.3 Empirical Results

Evaluation of our proposed method (Fuzzy-NWKNN) on all six datasets described in Table 1 is performed with F-measure, AUC and G-mean and its performance is compared with neighbor weighted algorithm (NWKNN) and Hybrid weighted nearest neighbor approach (Adpt-NWKNN) [18] for imbalanced data. Results for all mentioned evaluation measures are shown from Table 3 to Table 8 for individual datasets. We performed experiment for five values of K to find more generalized results.

Table 3. Results for F-Measure, AUC and G-Mean for Ionosphere dataset

<table>
<thead>
<tr>
<th>K</th>
<th>NWKNN</th>
<th>Adpt-NWKNN</th>
<th>Fuzzy-NWKNN</th>
<th>NWKNN</th>
<th>Adpt-NWKNN</th>
<th>Fuzzy-NWKNN</th>
<th>NWKNN</th>
<th>Adpt-NWKNN</th>
<th>Fuzzy-NWKNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.4252</td>
<td>0.4252</td>
<td>0.4733</td>
<td>0.3926</td>
<td>0.3926</td>
<td>0.4452</td>
<td>0.2303</td>
<td>0.2303</td>
<td>0.2467</td>
</tr>
<tr>
<td>10</td>
<td>0.4375</td>
<td>0.4733</td>
<td>0.4615</td>
<td>0.4057</td>
<td>0.4452</td>
<td>0.4321</td>
<td>0.2345</td>
<td>0.2467</td>
<td>0.2427</td>
</tr>
<tr>
<td>15</td>
<td>0.4561</td>
<td>0.4957</td>
<td>0.5085</td>
<td>0.469</td>
<td>0.5084</td>
<td>0.5216</td>
<td>0.4167</td>
<td>0.44</td>
<td>0.4476</td>
</tr>
<tr>
<td>20</td>
<td>0.52</td>
<td>0.549</td>
<td>0.5631</td>
<td>0.5734</td>
<td>0.5998</td>
<td>0.6129</td>
<td>0.5626</td>
<td>0.5839</td>
<td>0.5942</td>
</tr>
<tr>
<td>25</td>
<td>0.5</td>
<td>0.5769</td>
<td>0.5743</td>
<td>0.5621</td>
<td>0.6261</td>
<td>0.6278</td>
<td>0.5577</td>
<td>0.6044</td>
<td>0.6131</td>
</tr>
</tbody>
</table>

Table 4. Results for F-Measure, AUC and G-Mean for Wisconsin dataset
### Table 5. Results for F-Measure, AUC and G-Mean for Phoneme dataset

<table>
<thead>
<tr>
<th>K</th>
<th>F-Measure</th>
<th>AUC</th>
<th>G-Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NWKNN</td>
<td>NWKNN</td>
<td>NWKNN</td>
</tr>
<tr>
<td></td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
</tr>
<tr>
<td>5</td>
<td>0.5147</td>
<td>0.4974</td>
<td>0.8282</td>
</tr>
<tr>
<td>10</td>
<td>0.516</td>
<td>0.5334</td>
<td>0.5079</td>
</tr>
<tr>
<td>15</td>
<td>0.5162</td>
<td>0.5231</td>
<td>0.9748</td>
</tr>
<tr>
<td>20</td>
<td>0.5162</td>
<td>0.5231</td>
<td>0.9748</td>
</tr>
<tr>
<td>25</td>
<td>0.5092</td>
<td>0.5156</td>
<td>0.9748</td>
</tr>
</tbody>
</table>

### Table 6. Results for F-Measure, AUC and G-Mean for Vehicle0 dataset

<table>
<thead>
<tr>
<th>K</th>
<th>F-Measure</th>
<th>AUC</th>
<th>G-Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NWKNN</td>
<td>NWKNN</td>
<td>NWKNN</td>
</tr>
<tr>
<td></td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
</tr>
<tr>
<td>5</td>
<td>0.4236</td>
<td>0.4936</td>
<td>0.5278</td>
</tr>
<tr>
<td>10</td>
<td>0.4476</td>
<td>0.5417</td>
<td>0.5716</td>
</tr>
<tr>
<td>15</td>
<td>0.4527</td>
<td>0.5842</td>
<td>0.6083</td>
</tr>
<tr>
<td>20</td>
<td>0.4574</td>
<td>0.6024</td>
<td>0.6286</td>
</tr>
<tr>
<td>25</td>
<td>0.4587</td>
<td>0.6255</td>
<td>0.6465</td>
</tr>
</tbody>
</table>

### Table 7. Results for F-Measure, AUC and G-Mean for New-Thyroid dataset

<table>
<thead>
<tr>
<th>K</th>
<th>F-Measure</th>
<th>AUC</th>
<th>G-Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NWKNN</td>
<td>NWKNN</td>
<td>NWKNN</td>
</tr>
<tr>
<td></td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
</tr>
<tr>
<td>5</td>
<td>0.3491</td>
<td>0.4696</td>
<td>0.5099</td>
</tr>
<tr>
<td>10</td>
<td>0.3698</td>
<td>0.5063</td>
<td>0.5422</td>
</tr>
<tr>
<td>15</td>
<td>0.3465</td>
<td>0.4801</td>
<td>0.5525</td>
</tr>
<tr>
<td>20</td>
<td>0.3629</td>
<td>0.5064</td>
<td>0.5757</td>
</tr>
<tr>
<td>25</td>
<td>0.339</td>
<td>0.4828</td>
<td>0.5886</td>
</tr>
</tbody>
</table>

### Table 8. Results for F-Measure, AUC and G-Mean for Glass2 dataset

<table>
<thead>
<tr>
<th>K</th>
<th>F-Measure</th>
<th>AUC</th>
<th>G-Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NWKNN</td>
<td>NWKNN</td>
<td>NWKNN</td>
</tr>
<tr>
<td></td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
<td>Adpt-NWKNN Fuzzy-NWKNN</td>
</tr>
<tr>
<td>5</td>
<td>0.2388</td>
<td>0.4463</td>
<td>0.4963</td>
</tr>
<tr>
<td>10</td>
<td>0.2373</td>
<td>0.4611</td>
<td>0.5148</td>
</tr>
<tr>
<td>15</td>
<td>0.2456</td>
<td>0.4796</td>
<td>0.5056</td>
</tr>
<tr>
<td>20</td>
<td>0.25</td>
<td>0.4889</td>
<td>0.5426</td>
</tr>
<tr>
<td>25</td>
<td>0.2545</td>
<td>0.4981</td>
<td>0.5426</td>
</tr>
</tbody>
</table>

Comparison of performances could also be represented in form of bar graphs. Figure 1 to figure 3 are showing the combined result for all datasets for F-measure, AUC and G-mean for average values of K.

<table>
<thead>
<tr>
<th>K</th>
<th>F-Measure</th>
<th>AUC</th>
<th>G-Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NWKNN</td>
<td>Adpt-NWKNN</td>
<td>Fuzzy-NWKNN</td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
<td>0.0938</td>
<td>0.1311</td>
</tr>
<tr>
<td>10</td>
<td>0.0408</td>
<td>0.1449</td>
<td>0.1154</td>
</tr>
<tr>
<td>15</td>
<td>0.0465</td>
<td>0.127</td>
<td>0.1277</td>
</tr>
<tr>
<td>20</td>
<td>0.0541</td>
<td>0.1667</td>
<td>0.1463</td>
</tr>
<tr>
<td>25</td>
<td>0.0606</td>
<td>0.1587</td>
<td>0.1622</td>
</tr>
</tbody>
</table>

![Figure 1. F-measure Performances for NWKNN, Adpt-NWKNN and Fuzzy-NWKNN](image1)

![Figure 2. AUC Performances for NWKNN, Adpt-NWKNN and Fuzzy-NWKNN](image2)
6. Conclusion and Future Scope

Proposed fuzzy neighbor weighted algorithm is a fruitful extension of neighbor weighted approach. Neighbor weighted approach was performing well with imbalanced text data. In this paper we extract results of neighbor weighted method for numerical imbalanced data and proposed its fuzzy variant. In comparative study our proposed method (Fuzzy-KNN) is extracting better results than NWKNN and Adpt-NWKNN for all three evaluation measures on datasets from table 1. We consider all features necessary for classification. In further studies feature selection could also be considered. The experiments were performed for binary class datasets, this study could be extended for multiclass datasets in future.

References


