k Nearest Neighbour for Classification Mining:Using weighted MkNN

Dr. Joydip Dhar, Ashaya Shukla, Mukul Kumar and Prashant Gupta

Information and technology department, Atal Bihari Vajpayee Indian Institute of Information Technology and Management, Gwalior (Madhya Pradesh), India 474010

Abstract

kNN is a very powerful and effective Instance based learning method, nevertheless it is easy to implement. However, performance of kNN greatly relies on data quality being used for mining purpose. The elimination of noise and pseudo neighbours is still a challenge. To enlighten this issue, in this paper we propose a new learning algorithm which is going to perform the task of anomaly detection and removal of pseudo neighbours. This algorithm also tries to minimize effect of those neighbours which are distant, by applying concept of distance-based voting instead of majority voting, used in case of kNN. A concept of certainty measure is also introduced in this paper that provides how much certain are the upcoming results. Consequently, the final results of proposed algorithm are also trustworthy. An extensive experimental analysis performed on UCI datasets provides empirical indication of the utility of the proposed algorithm.

Keywords: instance based learning, pseudo neighbours, mutual nearest neighbour, WMkNN, distance based voting.

1. Introduction

Data mining is the processing of data for different perspectives, in order to find out interesting patterns in large databases and compiling them into useful information. Recent years have attracted a significant amount of research in almost all the aspects of data mining, where pattern classification is one of the most fundamental and widely studied research topics in the field of data mining and classification. In the field of classification, a successful application of an algorithm relies very much on the quality of data. In practical situations, collection of data is often done from miscellaneous diverse data sources. As a result of the ignorance of quality of data, erroneous decisions may be predicted by the classification learning algorithms. Moreover, there would be increased complexity, involved in the construction of the classification models. Hence in order to implement classification algorithms correctly, development of an effective technique for removal of noise from the dataset has become essential.It is worthy to mention that noisy instances bring less impact to simple learning algorithms, such as NBC and kNN than sophisticated classifiers such as SVM or random forests [4]. Due to the simplicity and easiness in implementation of kNN, this off-the-shelf method has also been applied to get rid of the noisy instances in databases, and can achieve competitive results even compared to the most sophisticated learning algorithms [1, 2, 5, 6].In this work a non-parametric lazy learning algorithm kNN (k-Nearest Neighbours) and its five more variants with variations in algorithmic procedure, where the performance of each algorithm depends upon the various factors is discussed. We propose a new learning algorithm which is going to perform the task of anomaly detection and providing weightage to closer nearest neighbour in comparison with those who are distant. Rest of the paper is systematized as follows section-2 comprises problem definition, section 3 methodology, section 4 results and discussion followed by conclusion in section 5.

2. Problem Definition

Pattern classification provides intuition about any unseen data that what kind of behaviour this data is going to show on the basis of patterns seen in history i.e., on the basis of historical data (duda et al.) [8]. For a person it is generally easy to predict that which sound is the sound of a male and which one is of a female, differentiate between handwritten letters but when this comes to computer programs it is not the same situation. For a programmable computer it is difficult to solve these kinds of perceptual problems [9]. For this case, it is difficult because as each pattern comprises huge amount of information and recognition problems typically have a subtle, high dimensional structure [9]. Formally it can be argued that pattern classification is organization of patterns into groups of patterns sharing same kind of properties. It can be applied to many possible potentials but classification is most widely used and has attracted more attention of researchers.

* Corresponding author. Tel: +91-80-22082823; 9880148294 E-mail: arun@jncasr.ac.in

Classification is basically a procedure in which one is intended towards defining a model or precisely it can be said that hypothesis function that learns from given dataset about behaviour of data items on the basis of various features intended for able to predict the class label for upcoming unknown instances. Let x_i be an input instance represented as p-dimensional vector form, i.e., $x_i = (x_{i1},\ldots,x_{ip})$, $C = \{c_1,\ldots,c_m\}$ be a set of class labels. The class label of instance x_i fits to one of the categories c_i, i.e., there is a scalar function f, which allocates a class label, $c_i = f(x_i)$, to every instance. Given a dataset D consisting of n pairs of instance and label, i.e., $D = \{(x_1, c_1),\ldots,(x_n, c_n)\}$. The classification task is to determine the scalar function (i.e., hypothesis) *f*, on which the class labels of unclassified instances can be predicted specifically [10]. Starting from most basic and influential work of (Cover and Hart) [11] their proposed **1-NN** was perhaps the most influential step towards classification and straightforward learning algorithm. The fundamental idea behind this algorithm was to take in consideration only one nearest neighbour for prediction of class label for any upcoming unknown instances. Formally let's say x_i is an unknown instance a given a dataset *D.* In 1-NN algorithms the first job that is performed is to store of all the instances of training set in memory so that they can be used for further querying. Suppose the class label for the unknown instance x_i is to be predicted. It now searches for its nearest neighbourx_0 so that it can predict class of unknown instance x_i as *c* which is derived from $f(x_0)$.So class of the input instance x_i will be the same class that is of x_0. As 1-NN only considers the information regarding the closest nearest neighbour so it is more susceptible of noise and also decision boundary is over-fitted. So there are various variants which try to minimize these limitations in the upcoming sections we are going to discuss various variants which endeavour to solve this limitation of 1-NN by using their own methods.

3. Methodology

As discussed in last section the most basic method used for classification is 1-NN. Irrespective of its simplicity and easiness in implementation it has a major drawback of its susceptible behaviour towards noise and over fitting of decision boundary a no of extensions of this algorithm came into existence.1st and the most general kind of extension is k-NN i.e. extending 1 nearest neighbour to k nearest neighbours. In case of 1-NN we were considering only 1 nearest neighbour here we are just taking in consideration k nearest neighbours (see figure 2).Let $N_k(x)$ be the set of k nearest neighbour of instance x. prediction of class of x will be based on only majority voting among all the k nearest neighbours of instance x [10]:

$$c = argmax \sum_{c_i \in C} \sum_{x_j \in N_k(x)} I(c_j = c_i). \quad (1)$$

In equation (1), c is the class to be predicted for instance x, c_j is the class label of instance x_j. I(.) is the indicative function which gives result 1 when $I(c_j=c_i)$ this means that class of instance x_j is same as that of class of instance x_i otherwise this indicative function I(.) will give result 0 [10]. Despite of its simplicity, kNN gives comparative results in comparison with other sophisticated algorithms of Machine learning. Impact of each neighbour in case of kNN is same weights can be applied to each neighbours for classification, this extension of kNN is named as weighted kNN [13].In [10](Huawen Liua,b, Shichao Zhangc) argued that concept of mutual neighbours can be applied to remove anomalies from datasets i.e, $M_k(x)=\{x_i \in D | x_i \in N_k(x) \land x \in N_k(x_i)\}$. Concept of certainty measure is used to describe how much certain are the upcoming results. It can be derived as:

$$CM_i = \frac{\textbf{Total votes}(i)}{\sum_{c=1}^{\#of\ classes}(\textbf{total votes}(c))}. \quad (2)$$

In equation (2), CM_i is certainty factor of any prediction 'i' and total votes (i) is frequency of the class i in k nearest neighbours. Flow diagram of WMkNN(weighted Mutual k-Nearest Neighbour) has been shown in figure 1. We have used, three benchmark datasets with different sizes for simulation process. All of them can be freely accessed at the UCI Machine Learning Repository [14]. Description of datasets used is summarized in table 1.

Table 1: Description of Datasets used.

Datasets	Instances	features	classes
Glass	214	10	7
Wine	178	14	3
ILPD	583	10	2

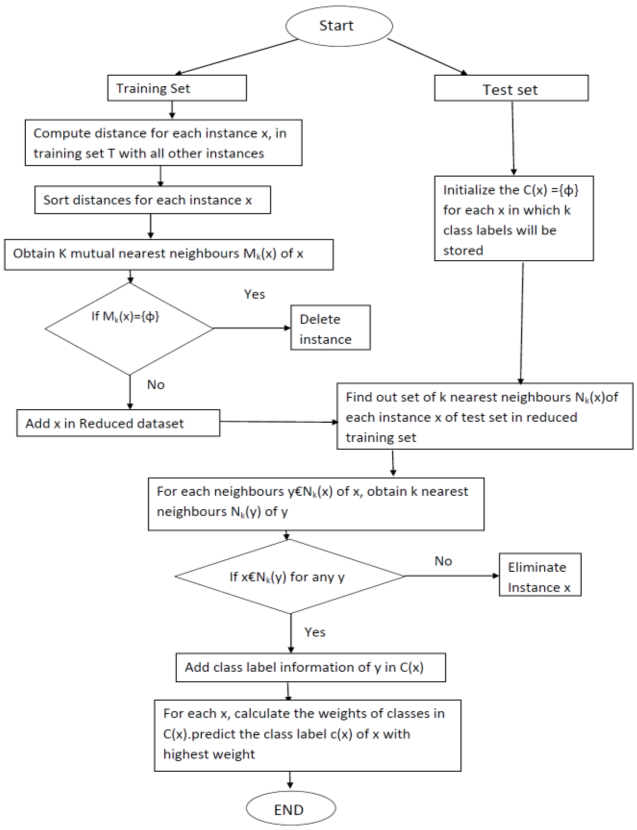

Figure 1: Flow Diagram of WMkNN.

4. RESULTS AND DISCUSSIONS

4.1 Experimental Comparison among All the Classifiers

4.1.1 Comparison among kNN, kNN* & MkNN

The experimental results on datasets are given in table 2. It can be noticed that the accuracy of MkNN is not always better than kNN and kNN*. For example, for k=3 and k=4 accuracy of kNN and kNN* is better than MkNN for the given dataset. It has been observed that the accuracy of MkNN is greater than kNN and kNN* for all other values of k, Since some of the instances of test data is eliminated while predicting the class labels of test data set using MkNN. But the accuracy of kNN* is lesser than kNN in almost all the cases.

Table 2: Accuracy values for ILPD Dataset

	k=3	k=4	k=5	k=6	k=7
kNN	60.09	61.66	61.30	62.47	63.24
WkNN	62.86	60.86	62.09	61.69	64.07
kNN*	58.92	60.46	60.90	62.47	63.24
WkNN*	61.29	60.44	62.09	61.69	64.07
MkNN	58.16	60.05	62.23	63.54	65.34
WMkNN	58.63	59.16	62.21	63.59	65.37

Table 3: Certainty values for ILPD Dataset

	k=3	k=4	k=5	k=6	k=7
kNN	79.39	76.74	73.95	73.55	72.56
WkNN	80.38	76.75	74.49	73.63	72.86
kNN*	78.52	75.76	73.47	73.31	72.57
WkNN*	79.64	75.83	74.12	73.38	72.87
MkNN	85.92	81.30	78.39	77.92	77.40
WMkNN	85.92	81.32	78.42	77.93	77.45

4.1.2 Comparison between Weighted and Non-weighted (frequency based) Algorithms

With reference to the data given in table 2, it can be seen that WkNN does not perform better than kNN always for all values of k. In comparision with kNN* and WkNN*, WkNN* comes up with better results than kNN*. While comparing MkNN and WMkNN, generally MkNN performs better than WMkNN for some values of k. However, WMkNN performs better than MkNN for some other values of k (see in figure 2).

4.1.3. Comparison for Different Values of k

Here, The appropriate k value after observing average accuracy values for different k is to decided. Observations shows that for different k values, algorithms are performing differently. For example at k=3 KNN is providing best results, secondly KNN* and lastly MKNN. But overall if average accuracies of all the algorithms for different values of k is computed, then it is noticeable that for k=7, they are coming up with best results.

4.2 Experimental Comparison among All the Classifiers

4.2.1 Comparison among kNN, kNN* & MkNN

In case of Certainty measure, in table 3, it is notable that MkNN is providing better results than kNN and kNN* in all the cases, because in MkNN, prediction of the class label of those instances, which are suspicious are not being done. While on an average kNN is providing more certain results than kNN*

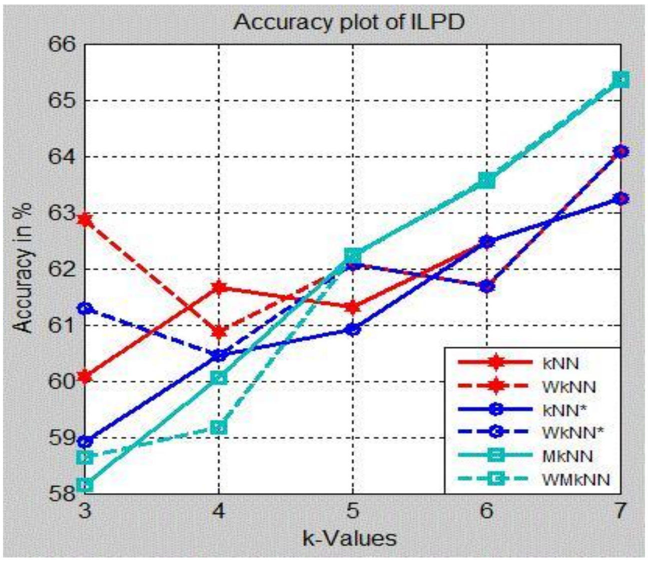

Figure 2: Accuracy values for ILPD Dataset

4.2.2 Comparison between Weighted and Non-weighted (frequency based) Algorithms

Here, it can be seen in figure 3 that weighted algorithms are providing more certain results than the corresponding non-weighted algorithms.

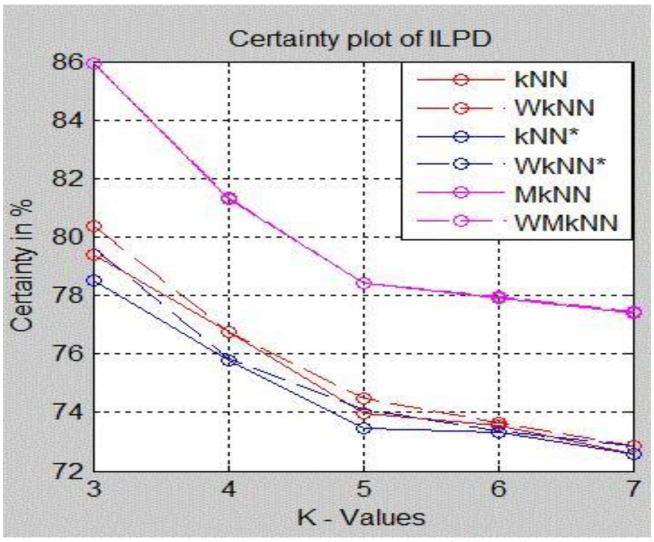

Figure 3: Certainty values for ILPD Dataset

For example, for k=3, WMKNN performs exceptionally well certain results than other algorithms.

4.2.3 Comparison for Different Values of k

With increment in the value of k, certainty measure is decreasing, because when k is increasing the overall weight is also increasing, which is the denominator value of our certainty measure. The probability of increment in weight of numerator is less than the probability of increment of weight in denominator (see in figure 3).

5. CONCLUSION

In this paper, we proposed new learning algorithms that does work of noise removal and apply weighted concept at the same time called WMkNNC. The motivation behind this algorithm was to eliminate noisy instances and provide weightage to closer neighbours in the job of prediction of class labels for unknown instances. Precisely it removed noisy instances by the virtue of concept of mutual neighbour and provide more weightage to closer neighbour by virtue of distance-based voting. Besides, a variation of proposed algorithm was also been performed that used notion of mutual neighbour only for training set. An extensive analysis of some benchmark datasets of UCI machine learning database was done. Results of analysis showed that the new proposed algorithm were came up with better results than conventional kNN. It provided result that WMkNNC is performing better than MkNNC and kNNC for value of k=3in almost all the datasets we used for our experimentation procedure. Proposed certainty measure came up with conclusion that it is associated with the value of k. With increment in k certainty measure decreases. It also concluded that certainty measure for MkNNC is better than kNNC. The accuracy results provided shows that classification performance by WMkNNC is better than conventional kNNC.

6. REFERENCES

[1] 1.Chandola, V, Banerjee, A., Kumar, V., 2009. Anomaly detection: a survey. ACM Computing Surveys 41 (3), 1–58.

[2] 2.Brighton, H., Mellish, C., 2002. Advances in instance selection for instance based learning algorithms. Data Mining and Knowledge Discovery 6, 153–172.

[3] 3.Yu, H., 2011. Selective sampling techniques for feedback-based data retrieval. Data Mining and Knowledge Discovery 22 (1), 1–30.

[4] Zhang, Y., Wu, X., 2010.Integrating induction and deduction for noisy data mining. Information Sciences 180 (14), 2663– 2673

[5] Chapman, A.D., (2005).Principles and Methods of Data Cleaning-Primary Species and Species-Occurrence Data, version 1.0. Report for the Global Biodiversity Information Facility, Copenhagen

[6] Pyle, D., 1999.Data Preparation for Data Mining. Morgan Kaufmann Publishers, San Francisco, CA, USA

[7] 7.http://en.wikipedia.org/wiki/Lazy_learning

[8] 8.Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification, 2nd ed.Wiley, NewYork

[9] http://www.byclb.com/TR/Tutorials/neural_networks/ch1_1.ht m

[10] Noisy data elimination using mutual k-nearest neighbor for classification mining (Huawen Liua, Shichao Zhangc).

[11] Cover, T.M., Hart, P.E., 1967. Nearest neighbour pattern classification. IEEE Transactions on Information Theory 13, 21–27.

[12] Minkowski distance http://en.wikipedia.org/wiki/Minkowski_distance

[13] Chapter-5(nearest neighbor) by Tan and kumar

[14] http://archive.ics.uci.edu/ml/datasets.htm.

Dr. Joydip Dhar is Associate Professor at Indian Institute of Information Technology and Management Gwalior.He received his PhD degree from IIT-Kanpur. His area of interest includes Industrial Mathematics: Mathematical Modelling and Simulation in Environmental, EMS, Management systems: Soft Computing and its applications in image processing.

Ashaya Shukla is doing Integrated Post Graduation (M.Tech in Information Technology) from Indian Institute of Information Technology and Management Gwalior.

Mukul Kumar is doing Integrated Post Graduation (M.Tech in Information Technology) from Indian Institute of Information Technology and Management Gwalior.

Prashant Gupta is doing Integrated Post Graduation (M.Tech in Information Technology) from Indian Institute of Information Technology and Management Gwalior.