Expert Systems with Applications 38 (2011) 12585–12591

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

K nearest neighbor reinforced expectation maximization method

Mehmet Aci *, Mutlu Avci

Department of Computer Engineering, University of Cukurova, Adana, Turkey

ARTICLE INFO

Keywords:
K nearest neighbor method
Bayesian method
Expectation maximization algorithm
Hybrid method
Classification
Clustering

ABSTRACT

K nearest neighbor and Bayesian methods are effective methods of machine learning. Expectation maximization is an effective Bayesian classifier. In this work a data elimination approach is proposed to improve data clustering. The proposed method is based on hybridization of *k* nearest neighbor and expectation maximization algorithms. The k nearest neighbor algorithm is considered as the preprocessor for expectation maximization algorithm to reduce the amount of training data making it difficult to learn. The suggested method is tested on well-known machine learning data sets iris, wine, breast cancer, glass and yeast. Simulations are done in MATLAB environment and performance results are concluded.

1. Introduction

Machine learning is a knowledge area starting at the point when data are explained or estimations are produced for the future. It generates functional approximation or classification models for the data. To convert the learning studies to machine learning some paradigms and approaches are used. Symbolic processing like decision trees and version spaces, connectionist systems, statistical pattern recognition, case based learning, evolutionary programming and genetic algorithms are some of them. At the machine learning the aim is to realize the human learning job by computers. Various methods and algorithms are used during this learning. Everyday new ones are added to these methods and algorithms or existings are developed. K nearest neighbor (KNN) and Bayesian methods are several of them. One of the goals of these methods is to find out the class of new data when the information about the classes of past data is given (Amasyali, 2006).

In KNN method a constant *k* value and a distance measurement criterion are chosen. In previous works different *k* values and distance measurement approaches are used. In some studies different methods like support vector machine and Bayesian methods are used with KNN method to obtain better results. Baoli et al. had selected varying *k* values for each class hence more sensitive measurements had done. More samples; nearest neighbors; had used for deciding whether a test document should be classified to a category, which has more samples in the training set. Experiments on Chinese text categorization has shown that their method was less sensitive to the parameter *k* than the traditional one, and it can properly classify documents belonging to smaller classes with a large *k* (Baoli, Shiwen, & Qin, 2003). Song et al. had used instructive data as criterion when determining *k* value at their study that was called informative KNN pattern classification. A new metric that measures the informativeness of objects to be classified had introduced. When applied as a query-based distance metric to measure the closeness between objects, Locally Informative-KNN (LI-KNN) and Globally Informative-KNN (GI-KNN) had proposed. By selecting a subset of most informative objects from neighborhoods, their methods had exhibited stability to the change of input parameters, number of neighbors (*K*) and informative points (*I*) (Song, Huang, Zhou, Zha, & Giles, 2007). Cucala et al. had proposed a reassessment of KNN procedure as a statistical technique derived from a proper probabilistic model. The assessment made in a previous analysis of this method had modified. A clear probabilistic basis for the KNN procedure and derived computational tools for conducting Bayesian inference on the parameters of the corresponding model had established. Their new model provides a sound setting for Bayesian inference (Cucala, Marin, Robert, & Titterington, 2009). Blanzieri and Melgani had developed a new method that inherits the attractive properties of both the KNN and the support vector machine classifiers. They had presented a new variant of the KNN classifier based on the maximal margin principle and exposed the advantages of new method. The proposed method had relied on classifying a given unlabeled sample by first finding its *k*-nearest training samples. A local partition of the input feature space was then carried out by means of local support vector machine (SVM) decision boundaries determined after training a multiclass SVM classifier on the considered *k* training samples. The labeling of the unknown sample had done by looking at the local decision region to which it belongs. The method had characterized by resulting global decision boundaries of the piecewise linear type (Blanzieri & Melgani, 2008). Weinberger and Saul had studied how to improve nearest neighbor classification by learning a Mahalanobis distance metric. The original framework for large margin nearest neighbor (LMNN) classification with three contributions had extended. First, a highly efficient solver for the partic-

* Corresponding author.
E-mail addresses: maci@cu.edu.tr (M. Aci), mavci@cu.edu.tr (M. Avci).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.04.046

ular instance of semidefinite programming that arises in LMNN classification had described. Second, how to reduce both training and testing times using metric ball trees; the speedups from ball trees are further magnified by learning low dimensional representations of the input space had shown. Third, how to learn different Mahalanobis distance metrics in different parts of the input space had shown (Weinberger & Saul, 2008).

Bayesian method based works are also very popular. Some studies are based on EM algorithm and some of them are performed with combination of Bayesian method with some other algorithms and methods like decision tree learning algorithms, regression and support vector machine methods. Ozcanli et al. had modeled the relations statistically between image segments and word set at an explained database by using translation with computer method. The relations to label the given segments had used. EM algorithm had used during statistical modeling and Bayesian method had formed the base for it. The performance of the proposed method was dependent to used attributes and their importance according to each other had concluded (Ozcanli, Duygulu-Şahin, & Yarman-Vural, 2003). Kotsiantis et al. had improved the performance of the Naive Bayes MultiNomial Classifier. Naive Bayes MultiNomial with Logitboost had combined. Naive Bayes MultiNomial classifier had modified in order to run as a regression method. A large-scale comparison with other algorithms on 10 standard benchmark data sets had performed and better accuracy in most cases had taken (Kotsiantis, Athanasopoulou, & Pintelas, 2006). Zheng and Webb had proposed the application of lazy learning techniques to Bayesian tree induction and presented the resulting lazy Bayesian rule learning algorithm, called LBR. For each test example, LBR had built a most appropriate rule with a local naive Bayesian classifier as its consequent. The computational requirements of LBR are reasonable in a wide cross-section of natural domains. Experiments with these domains shows that, on average, new algorithm had obtained lower error rates significantly more often than the reverse in comparison to a naive Bayesian classifier, a Bayesian tree learning algorithm, a constructive Bayesian classifier that eliminates attributes and constructs new attributes using Cartesian products of existing nominal attributes, and a lazy decision tree learning algorithm (Zheng & Webb, 2000). Yu et al. had proposed a Bayesian approach to determine the separating hyper plane of a support vector machine (SVM). In the proposed model of b-SVM, all the parameters are estimated by the reversible jump Markov chain Monte Carlo (RJMCMC) strategies, and the location parameter of decision boundary is finally described by a posterior distribution. The method minimizes the Bayes error in some derived direction. Tested by many independent random experiments of twofold cross-validations, the experimental results on some high-throughput biodata sets had demonstrated the promising performance and robustness of their novel's classification method (Yu, Cheng, Xiong, Qu, & Chen, 2008).

However, many applications had done on KNN and Bayesian methods, a hybridization of these two methods like the suggested hybrid method is not taken place in literature. Suggested approach brings a new point of view. A hybrid method is formed by using KNN and Bayesian methods together and aimed to achieve successful results on classifying by eliminating data that make difficult to learn. In other words with the hybrid method a data elimination approach is proposed to improve data clustering. The hybrid algorithm is formed with modifications on Bayesian and KNN methods. Main structure of the proposed algorithm explained in two steps. In first step, the number of training data is reduced and most similar training data to the querying data are obtained with KNN method. In the second step, class of the querying data is guessed with expectation maximization (EM) algorithm of Bayesian method. With the suggested algorithm a better data classification is obtained with respect to traditional EM algorithm. Briefly; KNN method is considered as the preprocessor for Bayesian EM classifier.

Test processes are evaluated with five of well-known UCI (University of California, Irvine) machine learning data sets (Yildiz, Yildirim, & Altilar, 2008). Those are Iris, Breast Cancer, Glass, Yeast and Wine data sets. Test results are compared with the previous works, and the performance of the proposed method is considered.

2. K nearest neighbor and Bayesian methods

2.1. K nearest neighbor method

KNN method is one of the oldest and simplest methods for general, non-parametric classification and based on supervised learning (Bay, 1999). The aim is to find nearest k sample from the existing training data when a new sample appears and classify the appeared sample according to most similar class (Mitchell, 1997).

Given a point x' of the d-dimensional input feature space, an ordering function $f_{x'}:\Re^d\to\Re$ is defined. The typical ordering function is based on the Euclidean metrics: $f_{x'}(x)=\|x-x'\|$. By means of an ordering function, it is possible to order the entire set of training samples X with respect to x'. This corresponds to define a function $r_{x'}:\{1,\ldots,N\}\to\{1,\ldots,N\}$ that reorders the indexes of the N training points. Blanzieri and Melgani had defined this function recursively, as follows:

$$\begin{cases} r_{x'}(1)=\arg\min_i f_{x'}(x_i) & \text{with } i\in\{1,\ldots,N\} \\ r_{x'}(j)=\arg\min_i f_{x'}(x_i) & \text{with } i\in\{1,\ldots,N\} \text{ and} \\ i\neq r_{x'}(1),\ldots,i\neq r_{x'}(j-1) & \text{for } j=2,\ldots,N. \end{cases} \tag{1}$$

In this way, $x_{r_{x'}(j)}$ is the point of the set X in the jth position in terms of distance from x', namely the jth nearest neighbor, and $f_{x'}(x_{r_{x'}(j)})=\|x_{r_{x'}(j)}-x'\|$ is its distance from x'. Given the above definition, the decision rule of the KNN classifier for binary classification problems is defined by the following majority rule:

$$KNN(x)=sign\left(\sum_{i=1}^{k} y_{r_x(i)}\right) \tag{2}$$

where $y_{r_x(i)}\in\{-1,+1\}$ is the class label of the ith nearest training sample (Blanzieri & Melgani, 2008).

Generally closeness is defined with Euclidean distance. Mitchell (1997) had explained Euclidean distance precisely with a formula. An arbitrary instance x be described by the feature vector $\langle a_1(x),a_2(x),\ldots,a_n(x)\rangle$ where $a_r(x)$ denotes the value of rth attribute of instance x. Then the distance between two instances x_i and x_j is defined to be $d(x_i,x_j)$ as follows

$$d(x_i,x_j)\equiv\sqrt{\sum_{r=1}^{n}\left(a_r(x_i)-a_r(x_j)\right)^2} \tag{3}$$

Afterwards, unknown sample is assigned to most similar class from KNN. Also KNN method is used to guess a real value for an unknown sample (Yildiz et al., 2008).

Primarily choosing appropriate k value and distance measurement determines the performance of a KNN classifier (Song et al., 2007). When the data points are not uniformly distributed, determining the k value becomes difficult. Generally larger k values are chosen in the event of noised data sets to make the boundaries smooth among the classes (Song et al., 2007). A good k can be selected by various heuristic techniques like cross-validation. The special case where the class is predicted to be the class of the closest training sample is called the nearest neighbor algorithm. It is impossible to choose same k value for all different applications (Song et al., 2007).

Different attempts have done to propose new approaches to increase the performance of KNN method by using prior knowledge such as the distribution of the data and feature selection. Discriminant Adaptive NN (DANN), Adaptive Metric NN (ADAMENN), Weight Adjusted KNN (WAKNN), Large Margin NN (LMNN) are some of these approaches (Song et al., 2007).

In general the following steps are performed for KNN algorithm (Yildiz et al., 2008):

1. Chose of k value: k value is completely up to user. Generally after some trials a k value is chosen according to results.
2. Distance calculation: Any distance measurement can be used for this step. Generally most known distance measurements like Euclidean and Manhattan distances are preferred.
3. Distance sort in ascending order: Chosen k value is also important in this step. Found distances are sorted in ascending order and k of minimum distances are taken.
4. Classification of nearest neighbors: Classes of k nearest neighbor are identified.
5. Finding dominant class: In the last step, queried data is classified according to class of identified k nearest neighbor by utilizing maximum ratio. This ratio is calculated for each class of k nearest neighbor with the number of data owned by that class over k. Let $P = \{p_1, p_2, p_3, \ldots, p_c\}$ is the set of k nearest neighbor probabilities for each class where c is number of class. Maximum ratio is calculated as in Eq. (4)

$$P_{\max} = \max(P_i/k). \tag{4}$$

2.2. Bayesian method

Bayes theorem is effective and simple method that is why it is used frequently on classifying problems (Gungor, 2004; Kim, Rim, Yook, Lim, & Anseo-Dong, 2002). In machine learning determining the best hypothesis from some space H, given the observed training data D, is often interested in. Bayes theorem provides a way to calculate the posterior probability $P(h|D)$, from the prior probability $P(h)$, together with $P(D)$ and $P(D|h)$ (Mitchell, 1997).

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)} \tag{5}$$

where $P(D)$ and $P(D|h)$ denote the prior and the posterior probability of observed training data D, respectively.

In this study classifying process is optimized by using EM algorithm. EM algorithm is a method to guess units which has missing data and includes maximum similarity probabilities (Friedman, 1998). EM method is a repeated method with two stages. Expectation stage gives expectation for the data. Maximization stage gives expectation about mean, standard deviation or correlation when a missing data is appointed. This process continues until the change on expected values decreases to a negligible value (Ozcanli et al., 2003).

The EM algorithm is a general approach to maximum likelihood in the presence of incomplete data. In EM algorithm for clustering, the "complete" data are considered to be $y_i = (x_i, z_i)$, where $z_i = (z_{i1}, \ldots, z_{iG})$ with

$$z_{ik} = \begin{cases} 1 & \text{if } x_i \text{ belongs to group } k \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

constitutes the "missing" data. The density of an observation x_i given z_i is given by $\prod_{k=1}^{G} f_k(x_i|\theta_k)^{z_{ik}}$. Each z_i is independent and identically distributed according to a multinomial distribution of one draw on G categories with probabilities $\tau_i, \ldots, \tau_G$. The resulting complete-data loglikelihood is

$$l(\theta_k, \tau_k, z_{ik}|x) = \sum_{i=1}^{n} \sum_{k=1}^{G} z_{ik}[\log \tau_k f_k(x_i|\theta_k)]. \tag{7}$$

The quantity $\hat{z}_{ik} = E[z_{ik}|x_i, \theta_1, \ldots, \theta_G]$ for complete-data loglikelihood is the conditional expectation of z_{ik} given the observation x_i and parameter values. The value z^*_{ik} of $\hat{z}_{ik}$ at a maximum of mixture likelihood approach is the conditional probability that observation i belongs to group k; the classification of an observation x_i is taken to be $\{j|z^*_{ij} = \max_k z^*_{ik}\}$ (Fraley & Raftery, 1998).

Bayesian methods are based on probability calculus. EM algorithm is one of them. It is utilized for unknown values with known probability distributions. Radial basis function is a popular function for explaining probability distributions (Mitchell, 1997; Neal, 2004). The expression of radial basis function is:

$$y = e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{8}$$

In Eq. (8), x denotes input data, μ denotes mean of training data and σ refers variance of training data.

One way to form an EM algorithm is to guess the average value of Gauss function. Let us have a sample data set which has k different classes. It means that data set is formed from a probability distribution which is a mixture of k different normal distribution. Each sample is formed with two step processes. At first step a random normal distribution is chosen from k normal distribution as seen in Fig. 1. At second step a sample data is formed according to this distribution. These steps are repeated for each datum in the set.

3. Proposed hybrid method

KNN and Bayesian methods are used together at this hybrid method. Unlike the previous given method by Cucala et al. (2009), KNN method considered as the preprocessor for EM algorithm of Bayesian method. Main idea is to reduce the number of data with KNN method and guess the class using most similar training data with EM algorithm.

On the beginning of the algorithm there are many train and test data. Generally some of them demonstrate large scale distribution. When Gaussian distribution function of them is calculated, this large scale distribution causes to a big variance. As marginal values are taken into consideration, the Gaussian distribution becomes more illusory. The same condition is also applied when calculating relative academic success by eliminating top and deep marks. Thus, selecting k nearest data closer to the queried data for Gaussian distribution function is more reliable. Also variance of this Gaussian distribution function is smaller than that of previous one.

Steps of this algorithm can be summarized as given below:

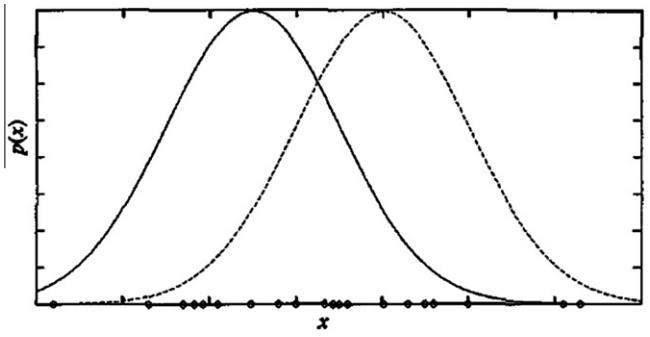

Fig. 1. The Normal distributions when $k = 2$ and sample data are on the x-axis (Mitchell, 1997).

Step 1: Train data of a set are read from a file. These data are used one by one for train and test purposes. Train mode is interactive and no batch mode exists.

Let $X = [x_1 \quad x_2 \quad x_3 \quad \dots \quad x_n]$ be the train set where each x_i ($i = 1$–n) is a vector and T is the test vector.

Step 2: Class number of the data set is identified and the data of the data set are divided into groups according to their classes. Then a distance measurement is chosen and distances between each train and test data are calculated by using selected distance measurement.

$D = [d_1 \quad d_2 \quad d_3 \quad \dots \quad d_n]$ where D is the distance matrix. Each d_i ($i = 1$–n) is the distance between test and each train data. Different distance measurements can be used for distance calculations. In this work distance calculations are done according to Eq. (9)

$$d_i = \sqrt[\lambda]{|x_i - T|^{\lambda}} \tag{9}$$

where d_i is the Minkowski distance. When λ is equal to 1, it is called as Manhattan distance, for λ is equal to 2, equation gives Euclidean distance. If λ is equal to 3 or more, the general name Minkowski is used for that distance. In this work, λ value is chosen in the order of 1, 2 and 3.

Step 3: For each class group, calculated distances are sorted in ascending order. At this point, closer distance means more similar data to the test data. Eq. (10) and procedure 1 show events of this step

$$X\_sorted = (sort_{asc}(D, X)) \tag{10}$$

Procedure 1:

procedure $sort(D, X)$
1. *for* $i = 1$ *to* n
2. *for* $j = 1$ *to* $n - 1$
3. *if* $d_{i+1} < d_i$ *then*
4. $temp\_d = d_i$
5. $d_i = d_{i+1}$
6. $d_{i+1} = temp\_d$
7. $temp\_x = x_i$
8. $x_i = x_{i+1}$
9. $x_{i+1} = temp\_x$
10. *end if*
11. *end for*
12. *end for*

Step 4: Optimum k value is determined heuristically considering the distances between queried and train data for the KNN algorithm. After that point old train set is updated with the new one as shown in equation (11).

$$X\_new = X\_sorted_h \tag{11}$$

where $h = 1$–k.

Step 5: In this step, KNN algorithm is completed and the EM algorithm is started. In general, the aim of EM is to form Gaussian distributions for each class and predict the class of queried data. EM algorithm considers maximum and minimum data of the new train set for each class. Also mean and variance are determined for each class to be used in the radial basis function. Eqs. (12)–(15) denote the calculation details

$$max\_x = \max(X\_new) \tag{12}$$

$$min\_x = \min(X\_new) \tag{13}$$

$$\mu = \text{mean}(X\_new) \tag{14}$$

$$\sigma = \text{var}(X\_new) \tag{15}$$

Step 6: A step value is calculated to fit the results in the same scale between maximum and minimum values for each class. Then radial basis functions are calculated for each class using step value, mean and variance of train data. In Eq. (16) step value calculation is shown

$$step_i = step_{i-1} + (max\_x - min\_x)/k \tag{16}$$

where $step_i$ is the step value, $step_{i-1}$ is the previous step value

$$G_i = e^{-\frac{(step_i - \mu)^2}{2\sigma^2}} \tag{17}$$

where $i = 1$–k and G is Gaussian distribution function.

Step 7: All data of new train set are located on formed Gaussian distributions for each class using radial basis function as in Eq. (18)

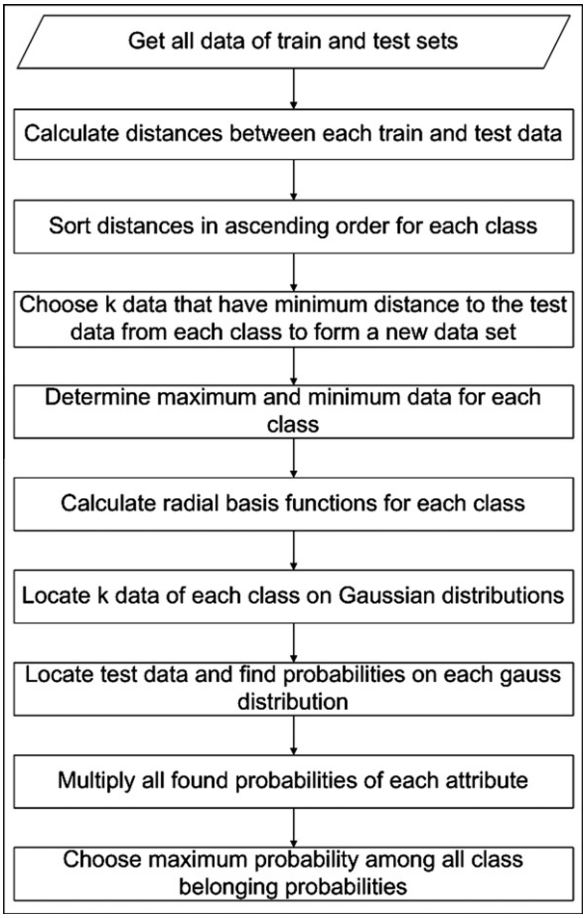

Fig. 2. Flowchart of the suggested hybrid method.

Table 1
Data sets and their statistical properties.

Properties	Iris	Wine	Breast cancer	Glass	Yeast
Class number	3	3	2	6	10
Sample number	150	178	699	214	1484
Distribution	50–50–50	59–71–48	458–241	70–17–76–13–9–29	463–429–244–163–51–44–37–30–20–5

Table 2
Distribution of wrong classified data after classifying with hybrid method (H) and k nearest neighbor method (K) at iris, wine and breast cancer data sets.

	Iris						Wine						Breast cancer					
	H			K			H			K			H			K		
k	$\lambda=1$	$\lambda=2$	$\lambda=3$	$\lambda=1$	$\lambda=2$	$\lambda=3$	$\lambda=1$	$\lambda=2$	$\lambda=3$	$\lambda=1$	$\lambda=2$	$\lambda=3$	$\lambda=1$	$\lambda=2$	$\lambda=3$	$\lambda=1$	$\lambda=2$	$\lambda=3$
1	1	1	1	3	3	3	13	12	11	15	19	19	3	3	4	10	14	17
2	3	3	3	6	6	5	24	25	24	18	25	26	8	5	5	17	16	15
3	3	5	5	5	5	4	25	23	25	25	30	33	12	11	9	18	15	15
4	2	5	5	5	5	5	36	36	37	25	35	35	16	12	12	16	19	21
5	2	4	3	4	4	5	35	33	32	27	35	36	14	15	15	18	17	20
6	4	3	3	4	4	3	29	32	32	26	42	41	16	15	16	19	17	16
7	3	3	3	5	5	3	34	34	34	29	39	39	17	16	18	21	15	18
8	3	2	2	5	3	3	36	34	33	27	38	42	17	17	18	24	20	20
9	3	3	2	4	4	3	38	34	34	34	39	41	16	17	16	22	19	21
10	2	3	2	4	4	2	38	38	38	34	39	40	18	18	15	24	21	21
11	3	3	3	5	3	2	42	37	37	33	42	42	18	18	15	23	19	20
12	2	3	3	5	3	2	41	39	39	39	46	47	18	18	16	25	21	22
13	2	3	3	4	4	3	42	36	37	35	50	50	17	18	16	23	19	20
14	3	3	3	4	2	2	40	33	32	41	48	47	17	19	19	25	22	22
15	4	3	3	4	4	3	41	36	34	42	48	48	18	22	20	25	20	22
16	4	3	4	5	3	3	38	36	35	44	48	47	19	21	21	25	23	22
17	4	4	3	5	3	3	41	39	41	43	48	48	19	22	21	25	22	22
18	4	4	3	4	3	3	41	37	38	45	49	48	19	21	22	25	22	23
19	3	3	3	4	3	3	40	37	36	43	48	48	20	21	22	25	22	21
20	3	3	3	4	3	3	42	42	38	45	48	49	19	22	23	26	25	21
21	3	3	3	3	3	3	43	39	39	45	46	48	20	22	23	26	23	21
22	4	3	4	4	3	3	44	43	43	46	49	49	20	22	23	29	25	22
23	4	3	4	3	4	3	42	42	43	44	47	48	20	22	24	29	23	21
24	4	4	4	4	3	3	42	41	42	46	48	49	20	20	24	30	24	23
25	4	4	4	3	4	3	43	41	42	44	48	49	20	20	24	29	24	22
26	5	5	4	4	4	5	42	42	40	49	49	49	20	20	24	28	24	22
27	6	6	4	6	3	3	43	42	42	47	49	49	18	20	23	27	24	23
28	6	6	6	5	4	5	45	44	43	47	49	49	18	20	24	28	25	24
29	6	6	6	5	4	4	43	44	43	47	48	48	18	19	24	27	25	24
30	6	6	6	6	6	6	42	42	41	47	50	49	19	18	24	28	25	25
31	7	7	7	5	7	6	42	42	41	45	49	49	19	18	23	28	25	25
32	7	7	7	6	6	6	41	43	42	48	51	51	19	19	23	29	25	26
33	7	7	7	5	7	7	43	43	42	45	48	48	18	19	23	28	25	26
34	7	7	7	6	6	7	43	41	41	46	49	49	18	20	22	29	25	26
35	7	7	7	5	6	6	43	40	40	46	49	49	18	20	22	29	25	26
36	7	7	7	4	6	7	44	41	41	45	45	48	18	20	22	30	25	26
37	7	7	7	5	5	6	46	40	40	43	45	48	17	20	22	30	25	26
38	7	7	7	7	7	8	45	40	40	45	45	48	17	20	22	31	26	26
39	7	7	7	6	6	7	45	44	44	45	49	49	16	20	22	31	26	26

$$V_i = e^{-\frac{(X\_new_i-\mu)^2}{2\sigma^2}} \tag{18}$$

where $i = 1$–k and V is probability values of X_new train data on Gaussian distribution.

Step 8: There exist equal number of Gaussian distributions with the number of classes. Test data are also located on each Gaussian distribution. Then belonging probabilities of the test data are determined on each Gauss distribution for each class.

$P = [p_1 \quad p_2 \quad p_3 \quad \dots \quad p_a]$ is the probability vector of the test data according to a attributes.

Step 9: Previous two steps are done for only one attribute of both train and test data. In Step 9, steps 7 and 8 are repeated for each attribute of the train and test data. Then all found belonging probabilities are multiplied for each class of the test set. In Eq. (19) this belonging probability calculation of the test data is shown

$$P\_test = \prod_{s=1}^{a} p_s \tag{19}$$

Step 10: In last step the classification is completed according to maximum belonging probability of the classes. The maximum belonging probability gives the class of the test data. Through the method all data are used for both train and test purpose.

$P_{class} = \{p_{test_1}, p_{test_2}, p_{test_3}, \dots, p_{test_m}\}$ is the set of belonging probabilities of the test data for all classes

$$P\_belonging = \max(P\_class) \tag{20}$$

gives the belonging probability of test data.

Flowchart of the suggested hybrid method is given in Fig. 2.

There is only one limitation for the k value. k value must not exceed the data number of the class with minimum data. In this work k values are swept from 1 to 39, and the k value giving the best clustering performance is determined.

4. Test results

Suggested method is tested on well-known data sets iris, wine, breast cancer, glass and yeast (Yildiz et al., 2008). The main properties of data sets are given in Table 1.

After implementing the steps of the proposed method, classification results given in Tables 2–4 are obtained. According to the results in Tables 2–4; hybrid method results better than Bayesian algorithm for all k and λ values. For small k values the hybrid method gives better results with respect to k nearest neighbor algorithm. Different λ values cause the best clustering performance. For breast cancer data set Manhattan ($\lambda = 1$) distance determines better results than that of other λ values. Generally, Euclidean

Table 3
Distribution of wrong classified data after classifying with hybrid method (H) and k nearest neighbor method (K) at glass and yeast data sets.

	Glass						Yeast					
	H			K			H			K		
λ	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$	$k=1$	$k=2$	$k=3$
1	15	13	13	25	24	24	197	179	177	303	316	318
2	15	20	21	32	35	37	316	305	282	379	383	383
3	30	28	25	37	46	42	387	357	385	415	423	404
4	33	34	34	40	47	51	419	406	412	457	457	456
5	39	37	40	43	51	54						
6	41	45	44	50	55	55						
7	42	44	44	55	58	60						
8	46	43	46	54	58	62						

($\lambda = 2$) and Minkowski ($\lambda \geqslant 3$) distances gives similar poor results. Minkowski distance obtains the best performance on wine data set. According to the average clustering results Euclidean distance is more applicable for the measurements. It obtains acceptable results for all data sets.

According to the obtained results, determining the optimum k and λ values are the most important points. These values vary for each data set according to data distribution of each attribute. It is obvious that the smaller k values, the better clustering results. Since more similar data are chosen for the clustering. So the probability of the wrong clustering decreases. On the other hand, data that cover wider data distribution interval are expected as train data for increasing reliability. As a result k values should be small enough for the successful clustering and big enough for the reliability. For the hybrid method optimum k and λ values are listed in Table 5. According to the table for iris data set optimum k values are between 1 and 14 when λ is equal to 1. For wine and breast cancer data sets optimum k values are between 1 and 20, and 1 and 13 respectively when λ is equal to 3. There is no chance to choose big k values for glass and yeast data sets. Both of them gives better results when λ is equal to 2. Optimum k values are between 1 and 5 for glass data set, and 1 and 4 for yeast data set.

For iris data set maximum clustering errors are 7 at hybrid method for all distances. With k nearest neighbor algorithm maximum clustering errors are 7 for Manhattan and Euclidean distances and 9 with Bayesian algorithm. For wine data set there are maximum 81 wrong classified data with Bayesian algorithm, 49 with k nearest neighbor algorithm when λ is equal to 1 and 44 with hybrid method when λ is equal to both 1 and 2. Although Bayesian algorithm does not perform a good classifying performance at breast cancer data set, hybrid method classifies only 20 data as wrong with Manhattan distance. Both hybrid method and k nearest neighbor algorithm obtain good results at glass and yeast data sets. Hybrid method classifies 45 and 406 data as wrong at glass and yeast data sets respectively with Euclidean distance. Although, k nearest neighbor algorithm's best results are 55 and 456 at these data sets.

Table 4
Number of wrong classified data after classifying with Bayesian method.

Iris	Wine	Breast cancer	Glass	Yeast
9	81	458	135	1235

Table 5
Optimum k and λ values for the hybrid method.

Data set	k	λ
Iris	1–14	1
Wine	1–20	3
Breast cancer	1–13	3
Glass	1–5	2
Yeast	1–4	2

5. Conclusions

In this work a hybrid classification method based on combination of k nearest neighbor and Bayesian method is introduced. k nearest neighbor algorithm considered as the preprocessor for Bayesian classifier to reduce the number of training data that make difficult to learn. This hybrid method is compared with KNN and EM algorithms on iris, wine, breast cancer, glass and yeast data sets. Better classification performance is obtained with respect to the methods.

The hybrid method is applicable to low cost hardware based clustering solutions and noisy data set classifying applications. According to test results the proposed method shows better classifying performance than k nearest neighbor and expectation maximization classifiers.

References

Amasyali, M. F. (2006). *Introduction to machine learning.* <http://www.ce.yildiz.edu.tr/mygetfile.php?id=868>.

Baoli, L., Shiwen, Y., & Qin, L. (2003). An improved k-nearest neighbor algorithm for text categorization. In *20th international conference on computer processing of oriental languages*.

Bay, S. D. (1999). Nearest neighbor classification from multiple feature subsets. *Intelligent Data Analysis, 3*(3), 191.

Blanzieri, E., & Melgani, F. (2008). Nearest neighbor classification of remote sensing images with the maximal margin principle. *IEEE Transactions on Geoscience and Remote Sensing, 46*(6), 1804–1811.

Cucala, L., Marin, J. M., Robert, C. P., & Titterington, D. M. (2009). A Bayesian reassessment of nearest-neighbour classification. *Journal of the American Statistical Association, 104*(485), 263–273.

Fraley, C., & Raftery, A. E. (1998). How many clusters? Which clustering method? Answers via model-based cluster analysis. *The Computer Journal, 41*(8), 578–588.

Friedman, N. (1998). The Bayesian structural EM algorithm. In *14th conference on uncertainty in artificial ıntelligence* (*UAI'98*) (pp. 129–138).

Gungor, T. (2004). *Developing dynamic and adaptive methods for Turkish spam messages filtering*. Technical Report 04A101. Bogazici University Research Fund.

Kim, S. B., Rim, H. C., Yook, D. S., Lim, H. S., & Anseo-Dong, C. (2002). Effective methods for improving Naive Bayes text classifiers. In *PRICAI 2002: Trends in artificial intelligence: 7th Pacific. LNAI* (Vol. 2417, pp. 414–423).

Kotsiantis, S., Athanasopoulou, E., & Pintelas, P. (2006). Logitboost of multinomial Bayesian classifier for text classification. *International Review on Computers and Software (IRECOS), 1*(3), 243–250.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw-Hill.

Neal, R. M. (2004). Bayesian methods for machine learning. *NIPS Tutorial*.

Ozcanli, Ö. C., Duygulu-Şahin, P., & Yarman-Vural, F. T. (2003). Açıklamalı Görüntü Veritabanları Kullanarak Nesne Tanıma Ve Erişimi. *11. Sinyal İşleme ve İletişim Uygulamaları Kurultayı.*, 18–20.

Song, Y., Huang, J., Zhou, D., Zha, H., & Giles, C. L. (2007). IKNN: Informative K-nearest neighbor pattern classification. In *PKDD* (pp. 248–264).

Weinberger, K. Q., & Saul, L. K. (2008).Fast solvers and efficient implementations for distance metric learning. In *ACM international conference proceeding series* (Vol. 307, pp. 1160–1167).

Yildiz, T., Yildirim, S., & Altilar, D. T. (2008). Spam filtering with paralellized KNN algorithm. In *Akademik Bilişim 2008*.

Yu, J., Cheng, F., Xiong, H., Qu, W., & Chen, X. (2008). A Bayesian approach to support vector machines for the binary classification. *Neurocomputing, 72*, 177–185.

Zheng, Z., & Webb, G. I. (2000). Lazy learning of Bayesian rules. *Machine Learning, 41*, 53–87.