Water Resour Manage
DOI 10.1007/s11269-014-0730-z

A Comparative Assessment of Support Vector Machines, Probabilistic Neural Networks, and K-Nearest Neighbor Algorithms for Water Quality Classification

Fereshteh Modaresi · Shahab Araghinejad

Received: 15 May 2013 / Accepted: 16 June 2014

Abstract Water quality is one of the major criteria for determining the planning and operation policies of water resources systems. In order to classify the quality of a water resource such as an aquifer, it is necessary that the quality of a large number of water samples be determined, which might be a very time consuming process. The goal of this paper is to classify the water quality using classification algorithms in order to reduce the computational time. The question is whether and to what extent the results of the classification algorithms are different. Another question is what method provides the most accurate results. In this regard, this paper investigates and compares the performance of three supervised methods of classification including support vector machine (SVM), probabilistic neural network (PNN), and k-nearest neighbor (KNN) for water quality classification. Using two performance evaluation statistics including error rate and error value, the efficiency of the algorithms is investigated. Furthermore, a 5-fold cross validation is performed to assess the effect of data value on the performance of the applied algorithms. Results demonstrate that the SVM algorithm presents the best performance with no errors in calibration and validation phases. The KNN algorithm, having the most total number and total value of errors, is the weakest one for classification of water quality data.

Keywords Classification · Water quality · Support vector machine · Probabilistic neural network · K-nearest neighbor · CCME

1 Introduction

Water quality is one of the major criteria in water resources planning, and plays a key role in determining the operation policies of water resources. In order to specify the quality of water, a variety of water quality indices have been used such as biotic and non biotic indices (Ogleni and

F. Modaresi
Water resources engineering, University of Tehran, Tehran, Iran
e-mail: fereshteh_modaresi@yahoo.com

S. Araghinejad (✉)
University of Tehran, Tehran, Iran
e-mail: araghinejad@ut.ac.ir

Published online: 25 June 2014

Topal, 2011). Horton (1965) presented one of the earlier water quality indices. He believed that the main challenge in water quality classification is to define a simple quality index so that the quality of water can be described using limited information. To calculate water quality indices such as NSF and ISQA, qualitative variables are weighted while defining the weight of each variable is complicated and depends on the comments of specialists. Another water quality index is the index of CCME which uses three factors, Scope, Frequency, and Amplitude, rather than weighted variables; however, in order to determine the quality of a water resource such as an aquifer or a river, calculating water quality classes through this method for a large number of water samples also takes a lot of time. Therefore, to get rid of the complicated calculations for each sample of water and reduce the computational time, this paper aims to use machine and statistical learning methods for classification. Consequently, when the qualitative variables are entered into the model, the class of the water sample is determined quickly.

A large number of machine and statistical learning methods for classification have been put forward such as: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), regularized discriminant analysis (RDA), bagging and boosting classification trees, soft independent modeling of class analogy (SIMCA), k-nearest neighbor classifier (KNN), neural networks (NN), support vector machines (SVM), etc. Nevertheless, the question is whether and to what extent the results of these methods are different; a further question is what method yields the most accurate results.

To compare these methods, a number of studies have been presented in various fields. For example, Werther et al. (1994) classified mass spectra in terms of 54 structural properties using four methods: KNN, DA, SIMCA, and NN. The results of this study showed that the neural network method had the best consequence, and that all these methods could classify data only according to a few characteristics.

In other research, Shaffer et al. (1999) compared seven methods including: Probabilistic neural networks (PNN), learning vector quantization (LVQ) neural networks, back-propagation artificial neural networks (BP-ANN), SIMCA, Bayesian LDA, Mahalanobis LDA, and nearest-neighbor pattern recognition algorithm for classifying chemical sensor array data. They categorized data based on five criteria, and their outcomes indicated that the algorithms PNN, LVQ, and BP-ANN provided the most accurate performance in classification. They suggested that the PNN was the best method of all because of its fast training and confidence measure.

Kiang (2003) performed a comparative assessment of five methods such as NN, Decision tree, Multivariate DA, Logistic models, and KNN. She used synthetic data to perform a controlled experiment in which the imperfections of these methods were determined by changing data characteristics. In this research, it was concluded that firstly, classification algorithms were considerably affected by the data characteristics; secondly, the neural network and logistic models provided the best performance under most of the designed scenarios. Moreover, it was recommended that a number of classification methods be evaluated in each study because one method cannot outperform all methods in all problems; then, either the method which has the most accurate results be selected or a number of different methods be combined to constitute a hybrid classifier in response to the presence of different biases in data.

Byvatov et al. (2003) compared the performance of a standard feed-forward neural network with a specified number of neurons in hidden layer with that of a SVM for drug/nondrug classification. The results of this paper revealed that the SVM method had a smaller standard error and more accurate answers than the ANN method. The authors maintained that the SVM method enjoys two particular advantages which cause it to excel at the ANN method. These two merits are: (1) the SVM method only depends on support vectors and the whole of data set do not affect the classifier function, (2) as the SVM method exploits kernel functions, this method can classify data based on a large number of features.

Lee and Park (2006) evaluated a variety of classification algorithms to identify true interacting protein pairs from noisy data. Their results showed that these algorithms were powerful implements to distinguish true interacting protein pairs and that the KNN and decision tree algorithms had the best performance among other methods.

In order to determine pesticide on spinach leaf, Tsuta et al. (2009) assessed two methods SVM and LDA. They categorized three types of fluorescence images, and concluded that Misclassification rates for LDA and SVM were 18.8 and 9.9 % respectively; therefore the SVM method excelled at LDA method.

In order to classify gasoline in terms of near infrared (NIR) spectroscopy data, Balabin et al. (2010) compared nine different multivariate classification methods including: LDA, QDA, RDA, SIMCA, partial least squares (PLS) classification, KNN, SVM, PNN, ANN-MLP. Their results indicated that KNN, PNN, and SVM methods provided the most effective performance than the other methods, among which the PNN method was the most efficient.

As presented in the literature, a number of various classification methods were evaluated and compared in different fields. However, in the field of water quality classification little research has been carried out with respect to assessing the efficiency of these methods. In this case, the only study which can be referred to is the study conducted by Chen et al. (2004). The aim of this study was to compare three supervised methods of classification including Maximum Likelihood (MLH), ANN, and SVM to categorize the spatial patterns of ocean color related to water quality. They used 88 real samples which 66 samples for training and 23 samples for testing were applied. Their results showed that ANN and SVM were more appropriate for a small set of samples compared to the MLH method and that the accuracy of the SVM method was a little better than two other methods.

In the light of what was said, it can be concluded that three methods including support vector machine (SVM), probabilistic neural network (PNN), and k-nearest neighbor (KNN) in most of the studies enjoyed the best performance and the most accurate results (e.g. Shaffer et al., 1999; Byvatov et al. 2003; Lee & Park, 2006; Tsuta et al. 2009; Balabin et al., 2010; Chen et al., 2004). With regard to this conclusion and because the SVM method is a new approach to classification using support vectors, and the PNN method is a specific kind of neural networks for classification, and the KNN is a simple and traditional method for classification, the aim of the current study is to classify water quality using these three methods, and compare their results to suggest the best one for water quality classification. The rest of this paper is as follows: in section 2 the case study is introduced and in section 3 the CCME water quality index used for classifying initial data is presented. Section 4 describes the classification methods used in this research, i.e. SVM, PNN, and KNN methods. The results are presented in section 5 and the conclusion is presented in section 6.

2 Case Study

In this research, water quality classification of the main aquifer of Tehran plain was investigated. Tehran plain is located in the north of Iran and in Tehran Province. This plain lies between latitudes 35° 28′ and 35° 49′ N, and longitudes 51° 15′ and 51° 36′ E. The main aquifer of this plain is surrounded by Abas Abad hills to the north, Lavizan hills to the east, Kan river to the west and Bibi Shahrbanoo mountains to the south. The area of this part of plain is more than 526 Km^2. The type of Tehran aquifer is unconfined, and this aquifer is directly polluted by wastewater absorbing wells. Two major pollutants founded in this aquifer are Nitrate and Chloride. As a result, water quality classification is this aquifer was performed based on these two contaminants. The data used in this work were obtained from 100 observed well across the aquifer during the period 2003–2004.

3 CCME Water Quality Index

The CCME Water Quality Index was introduced in 1995 by Water Quality Guidelines Task Group of the Canadian Council of Ministers of the Environment (CCME). To calculate the index, water quality variables are measured relative to a certain extent, and the amount exceeded that will be determined. This extent can be based on recommendations made in order to maintain the usability of water for intended purpose or any other standard proposed for different water consumptions. The advantage of applying this index is the possibility of using internal standards of each basin, city, or country and ability to classify on the basis of all measured available variables (CCME, 2001).

As noted above, to calculate the index, the quality standards and variables should be firstly defined. Then, three factors which constitute the index are calculated. These factors are as follows (CCME, 2001):

1) Factor 1: F_1 (Scope)

Scope represents the percentage of variables that do not meet their respective objectives at least once over the time period of interest ("failed variables"). It is calculated by:

$$F_1 = \left(\frac{Number\ of\ failed\ \text{variables}}{Total\ number\ of\ \text{variables}}\right) \times 100 \tag{1}$$

Factor 2: F_2 (Frequency)

Frequency represents the percentage of individual tests that do not meet their objectives ("failed tests"), calculated by:

$$F_2 = \left(\frac{Number\ of\ failed\ tests}{Total\ number\ of\ tests}\right) \times 100 \tag{2}$$

2) Factor 3: F_3 (Amplitude)

Amplitude represents the amount by which failed test values do not meet their respective objectives. This factor is calculated in the following three steps:

i) The number of times by which an individual concentration is greater than (or less than, when the objective is a minimum) the objective is termed an "excursion". When the test value must not exceed the objective, the excursion is expressed by:

$$excursion_i = \left(\frac{FailedTestValue_i}{Objective_j}\right) - 1 \tag{3a}$$

For the cases in which the test value must not fall below the objective, the excursion is expressed as follows:

$$excursion_i = \left(\frac{Objective_j}{FailedTestValue_i}\right) - 1 \tag{3b}$$

ii) The compliance of individual tests is determined by the collective amount calculated by summing the excursions of individual tests from their objectives and dividing by the total

number of tests (both those meeting and not meeting objectives). This variable which is referred to as the normalized sum of excursions, or *nse*, is calculated by:

$$nse = \frac{\sum_{i=1}^{n} excursion_i}{number\ of\ tests} \tag{4}$$

iii) Finally, F^3 is calculated as follows by an asymptotic function which scales the normalized sum of the excursions from objectives (nse) to yield a range between 0 and 100.

$$F_3 = \left(\frac{nse}{0.01\,nse + 0.01}\right) \tag{5}$$

When the factors have been obtained, the CCME Water Quality Index (CCME WQI) can be calculated by the following equation:

$$CCME = 100 - \left(\frac{\sqrt{F_1^2 + F_2^2 + F_3^2}}{1.732}\right) \tag{6}$$

Where, the divisor 1.732 scales the resultant values from 0 to 100 while 0 represents the worst water quality and 100 represents the best water quality.

After determining the CCME WQI value, water quality is classified based on this index according to the following categories:

1) **Poor**: (0–44)
2) **Marginal**: (45–64)
3) **Fair**: (65–79)
4) **Good**: (80–94)
5) **Excellent**: (95–100)

Using the CCME Water Quality Index, in the present paper the water quality of 100 observed wells in the study area was classified based on two pollutants Nitrate and Chloride. With respect to these contaminants, water quality of the wells was categorized into three classes Excellent, Marginal, and Poor and any well was not ranked in other classes. Figure 1 shows the classification of observed data based on CCME WQI according to two pollutants Nitrate and Chloride.

4 Methodology

As mentioned in the section of introduction, because the three supervised algorithms including support vector machine (SVM), probabilistic neural network (PNN), and k- nearest neighbor (KNN) enjoyed the best performance and the most accurate results in most of the researches which have been done on classification, in the current study these three algorithms were used for water quality classification. In the following, each of these algorithms will be described separately in details.

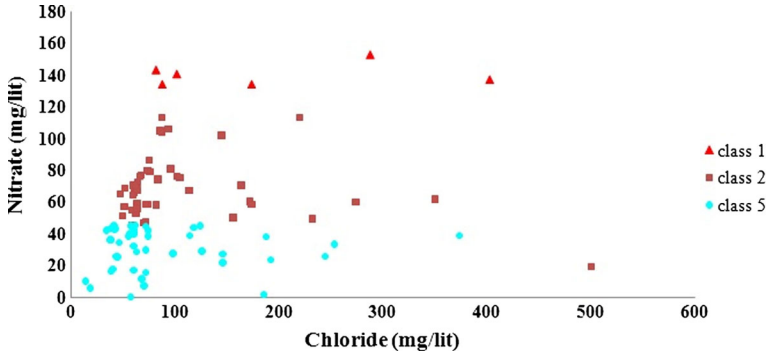

Fig. 1 The classification of observed data based on CCME WQI according to Nitrate and Chloride

4.1 Support Vector Machine Classification

Support Vector Machine was first introduced by Boser and Guyon in 1992 and its foundation was developed by Vapnik in 1995 (Vapnik, 1995). SVMs are a set of related supervised learning methods which have been used for classification and regression (Araghinejad, 2014, Aggarwal et al., 2012; Ghosh and Katkar, 2012; Hong and Pai, 2007). They belong to a family of generalized linear classifiers. The formulation of SVM uses the Structural Risk Minimization (SRM) principle, which has been shown to be superior to the traditional Empirical Risk Minimization (ERM) principle used by conventional neural networks (Burges, 1998). In what follows, the binary and multiclass SVM will be introduced.

4.1.1 Binary Classification with SVM

The aim of SVM is to define the optimal hyperplane according to the training data, which separates objects belonging to two classes, whereas it makes an attempt to maximize the margin between those classes.

The general formulation of SVM is defined over a training set of pairs $(x_i,y_i), i=1, 2,\ldots,n$, where x_i is the vector containing m features, and $y_i \in \{-1,1\}$ is the label related to x_i. In order to find the optimal hyperplane, the SVM algorithm solves the following optimization problem,

$$Minimize \frac{1}{2} w^2 + C\sum_{i=1}^{n} \xi_i$$

$$\begin{aligned} Subject\,to: y_i\left(w^T \varphi(x_i) + b\right) \geq 1-\xi_i, \forall i \in \{1,\ldots,n\} \\ \xi_i \geq 0, \forall i \in \{1,\ldots,n\} \end{aligned} \quad (7)$$

Here, the function $\varnothing$ maps the training vectors x_i into a higher dimensional space. The training errors ξ_i are considered by the objective function to achieve the

maximum margin hyperplane, while they are adjusted by parameter C which is chosen by the user. Dual formulation of this function is expressed as follows,

$$Maximize \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j . k(x_i, x_j) \; Subject\ to$$

$$: \alpha_i \geq 0, \forall i \in \{1, \ldots, n\} \qquad \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{8}$$

Where, $k(x_i,x_j)=(\varphi(x_i).\varphi(x_j))$ is called the kernel function.

After specifying the optimal parameters α, the decision function of classification for the *j-th* element becomes,

$$f(x_j) = sign\left(\sum_{i=1}^{n} \alpha_i y_i . k(x_i x_j) + b \right) \tag{9}$$

The kernel functions commonly used in SVM's formulations are:

a) Linear: $k(x_i,x_j)=x_i^T x_j$
b) Polynomial: $k(x_i,x_j)=(\gamma x_i^T x_j+r)^d, \gamma>0$.
c) Radial Basis Function (RBF): $k(x_i,x_j)=\exp(-\gamma\|x_i-x_j\|^2), \gamma>0$.
d) Sigmoid: $k(x_i,x_j)=\tanh(\gamma x_i^T x_j+r)$.

Here, γ, r and d are kernel parameters.

This original formulation of SVM can classify variables into two classes.

4.1.2 Multiclass Classification with SVMs

When one deals with more than two classes like water quality classification problems, an appropriate multiclass method is needed. A number of possible methods for this purpose are as follows (Chapelle et al., 1999).

- Modifying the design of the SVM to incorporate the multiclass learning directly in the quadratic solving algorithm.
- Combining several binary classifiers with two methods:

  a. "One against one" which applies pair comparisons between classes.
  b. "One against the others" which compares a given class with all the other classes.

  According to a comparison study (Weston & Watkins, 1998), the accuracy of these methods is almost the same. Therefore, in this study the method of "one against the others" was chosen, which has the lowest complexity.

4.2 Classification by Probabilistic Neural Network

The Probabilistic Neural Network (PNN) is a method developed by Donald Specht in 1988. A supervised training set to develop distribution functions within a pattern layer is used in this method. In order to estimate the likelihood of an input feature vector being part of a learned class, or category, these functions are used. Indeed, a PNN is a specific form of Neural

Network used to perform Bayesian classification techniques incorporating Parzen univariate estimation. To classify two classes Bayesian classifiers which can be used are as follows (Wasserman, 1993):

$$d(X) = \begin{cases} C_1 \, if \, l_1 h_1 f_1(X) > l_2 h_2 f_2(X) \\ C_2 \, if \, l_1 h_1 f_1(X) < l_2 h_2 f_2(X) \end{cases} \tag{10}$$

where X is a p-dimensional random vector, $d(X)$ is an image of X in a set of classes, C_i is the *i-th* class, l_i is the loss associated with misclassifying a vector of the *i-th* class into other class, h_i is the prior probability of occurrence in the *i-th* class, and $f_i(x)$ is the probability distribution function *(pdf)* for *i-th* class.

The aim of Eq. (10) is to minimize the expected risk in classification (Kim et al., 2005). The product of h_i and $f_i(x)$ is a posterior probability from Bayesian theorem which permits the updating of available knowledge h_i with new information $f_i(x)$. The available knowledge h_i could be obtained from a previous sample or the opinion of an expert, and an established mathematical foundation determines $f_i(x)$ to estimate the univariate pdf of a population from its sample. This foundation takes an average sum of kernel (pdf) values which was suitably chosen for each observation in the sample (Parzen, 1962). The loss l_i can be calculated or subjectively estimated, but it is usually assigned the same value for all classes.

To estimate the multivariate density function, as discussed by (Cacoullos, 1966), one can firstly take the multivariate pdf of an observation as a product of its univariate kernel, then apply Parzen's average sum so that the multivariate pdf is estimated. The following is shown an example of using the Gauss kernel for each observation of a random variable to estimate its density function,

$$f_i(X) = \frac{1}{(2\pi)^{\frac{p}{2}}\sigma^p} \frac{1}{n_i} \sum_{k=1}^{n_i} e^{\frac{\left(X - X_{i,k}\right)^T \left(X - X_{i,k}\right)}{-2\sigma^2}} \tag{11}$$

In this equation, X is a p-dimensional random vector, $f_i(x)$ is the pdf of X for *i-th* class, n_i is the number of observations in the *i-th* class, $x_{i,k}$ is the *k-th* observation in the *i-th* class, and σ is the smoothing parameter. In the case of the Gauss kernel (this equation), the meaning of the smoothing parameter σ is that univariate Gauss is sharply peaked with σ smaller than one, and tends to flatten with increasing σ (Wasserman, 1993).

The foundation of probabilistic neural networks generally is composed of four layers, an input layer and three information processing layers ranging from pattern layer to summation layer, to decision layer. A configuration of the PNN with four layers is shown in Fig. 2.

In the input layer, the number of neurons is equal to the number of input factors. The neurons of this layer only transfer input data to all neurons of the second layer, and any processing is not done on the data in this layer.

In the pattern layer, the total number of neurons is equal to the sum of the numbers of neurons used to represent the patterns for each class. Each class can contain a large number of training patterns (training vectors) of which dimension is the same as the number of input factors, while it is taking a set of specific values of input factors. The training vectors are imported from sample data and therefore they necessarily do not always represent all existing patterns for that class. Nonetheless, this can be the advantage of PNN in that it can generalize to allow recognition of a new pattern of a class (Wasserman, 1993). The activation function in the pattern layer can be chosen from some kernel density functions (Scott, 1992), but the *Gaussian* kernel is more commonly used (Araghinejad, 2014).

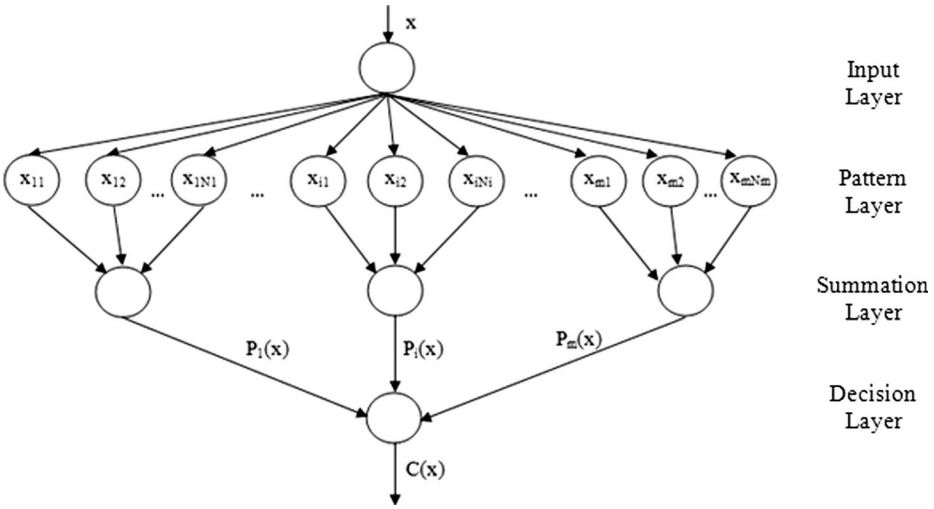

Fig. 2 A PNN schematic with four layers (Wasserman, 1993)

In the summation layer, the number of neurons is equal to the number of classes. Moreover, the activation function in this layer is a simple weighted sum function. The outgoing signals can be adjusted in accordance with loss and prior probability value.

Eventually, in the output layer, there is only one neuron to represent the max function, which outputs the class associated with the largest value between incoming signals (Kim et al., 2005).

As noted above, a PNN is a supervised algorithm. Thus, it needs a training stage before being used for classification. In the calibration stage, the smoothing parameter should be determined by applying the optimization methods to minimize the classification errors in the training vectors.

5 *K*- Nearest Neighbor Classification

The k-nearest neighbor is one of the common classification methods, which is on the basis of the use of distance measures. The *K*-NN technique assumes that the whole of sampling set includes both the data in the set and the desired class for each item. In order to classify a new item, its distance to each item which is in the sampling set must be computed. Then, the *K* closest items in the sampling set are selected to determine the class of the new item. The new item is then categorized to the class that contains the most items from this set of *K* closest items. The distance between two samples shows their similarity; therefore, ingredients of an instance determine features. Euclidean distance is the method usually used in the *K*-NN. For any two *n*-feature samples, say a vector of features for any sample like $X=(x_1, x_2, \ldots, x_n)$ and $Y=(y_1, y_2, \ldots, y_n)$, their Euclidean distance is computed based on the following equation:

$$dist(X, Y) = \sqrt{\sum_{i=1}^{n} c_i(x_i - y_i)^2} \tag{12}$$

where c_i is the weight of any feature (Yang Su, 2011, Araghinejad, 2014). In this algorithm, selecting the optimal value for *K* is important and depends on the type of the data. Generally, if *K* value is large, the precision of the results will be more because the larger *K* reduces the

overall noise but there is no guarantee. To determine a good value for *K,* another way is Cross-validation method that uses an iterative process so that the best value for parameter *K* is determined.

6 Results

As mentioned above, the aim of this study is to compare the performance of three methods SVM, PNN, and KNN for water quality classification. For this purpose, the water quality data of 100 observed wells were divided into 5 groups of 20 data. Then, 5 sets of data were produced in each of which 4 groups of 20 data were applied for training and one other group was applied for testing: for example, in set 1 four groups 1, 2, 3, and 4 were used for training and group 5 was used for testing, and in set 2, four groups 1, 2, 3, and 5 were used for training and group 4 was used for testing and so on. Actually, a 5- fold cross validation was performed to assess the effect of data value on the performance of these methods: for instance, the effect of presence or absence of maximum or minimum data in a data set on training the algorithm. To compare the efficiency of these three algorithms, two performance evaluation measures were defined:

1- Error Rate (ER): It is calculated by the following equation:

$$Error\,Rate(\%) = \frac{Number\,of\,misclassified\,data}{Total\,number\,of\,data} \times 100 \tag{13}$$

2- Error Value (EV): which is determined by the Eq. (14):

$$Error\,Value = (Observed\,class - Simulated\,class)^2 \tag{14}$$

Error Rate determines how many data have been misclassified, and Error Value shows the magnitude of classification error of the misclassified data. The zero value of them implies the absence of errors in classification, and the values greater than zero indicate the presence of errors such that the larger the value of them, the greater the error.

In what follows, the way of applying each method and its results will be separately presented, and finally, the results of the three methods will be compared.

6.1 Results of the SVM Method

As noted in the section of methodology, there are a variety of kernel functions for training the SVM algorithm. Moreover, parameter *C* in this algorithm must be defined by the user. With respect to the fact that the method of "one against the others" was chosen for training the algorithm in the present study, and that the quality of observed data was classified into three classes Excellent, Marginal, and Poor (i.e. 1, 2, and 5), this algorithm was trained three times, once for each class. In addition, in the training phase all of the kernel functions listed in methodology were investigated with different values of parameter *C*.

In the following, the simplest kernel function and the lowest value of parameter *C* which were enjoyed the lowest error were selected for each class. For example, if both the linear and quadratic kernel functions were appropriate for class 1, the linear kernel function would be selected, and if the values of 100 and 1,000 for parameter *C* had the same error, the value of

100 would be selected. The whole process was performed separately for each of the five data sets. Table 1 illustrates the optimum kernel function and value of parameter *C* for each class and each data set.

As can be seen from Table 1, the optimum kernel function type of each class is the same for all five data sets and it is linear for class 1 and quadratic for classes 2 and 5. Moreover, for classes 1 and 5, the optimum value of parameter *C* is also the same for all of the five data sets, and is equal to 10 and 1,000 respectively. However, for class 2, it is equal to 100 for data sets 2, 3, and 5 while its value is 10,000 and 100,000 for classes 1 and 4 respectively.

Regarding the point that the type of optimum kernel function of each class is the same for all data sets, it can be concluded that the SVM algorithm is almost insensitive to the value of data and that the general structure of data of each category plays an important role in determining the type of optimum kernel function. This is due to the fact that the SVM algorithm finds a hyperplain with maximum margin to classify the data.

Nevertheless, the amount of irregularity of scattering of data affects the value of parameter *C* such that the more irregular the data, the more value the parameter *C*. Therefore, the results of Table 1 indicate that the data of class 1 enjoy the lowest irregularity of scattering while the data scattering of classes 5 and 2 is more irregular; nonetheless, the data in class 2 have been irregularly scattered in five data sets. As a result, the value of parameter *C* is different for the five data sets.

Finally, based on the optimum kernel function and value of parameter *C* in Table 1, the SVM algorithm was calibrated (trained) and validated (tested) for each data set. The values of the Error Rate and Error Value of the calibration and validation phases have been presented in Table 2.

According to Table 2, the Error Rate and Error Value for the SVM algorithm in the calibration stage are equal to zero for the entire data sets. However, in the validation stage, the ER is equal to 0 % for data sets 1 and 4, 5 % (one misclassified data) for data sets 3 and 5, and 15 % (three misclassified data) for data set 2. Moreover, the EV is equal to 34, 4, and 9 for data sets 2, 3, and 5 respectively. It can be seen that while the ER is the same for data sets 3 and 5, the error of classification (EV) in data set 3 is less than data set 5, and that because there are three errors in the classification of data set 2, the EV of this data set is much larger than two other data sets which have one error in classification.

The results show that in general, the SVM algorithm has been well calibrated and validated for all data sets especially for data sets 1 and 4. Nevertheless, the number of misclassified data and the magnitude of error in other data sets are also low. Consequently, because of the low error rate of the SVM algorithm for all data sets in both calibration and validation phases, it can be concluded that the SVM algorithm has a great compatibility with different types of division of the data while it can be well calibrated and validated.

Table 1 The optimum kernel function and value of parameter *C* for each class and each data set

Class 5		Class 2		Class 1		Number of data set
Parameter *C*	Type of kernel function	Parameter *C*	Type of kernel function	Parameter *C*	Type of kernel function	
1,000	Quadratic	10,000	Quadratic	10	Linear	1
1,000	Quadratic	100	Quadratic	10	Linear	2
1,000	Quadratic	100	Quadratic	10	Linear	3
1,000	Quadratic	100,000	Quadratic	10	Linear	4
1,000	Quadratic	100	Quadratic	10	Linear	5

Table 2 The error indices of the SVM algorithm in calibration and validation phases

Error value		Error rate (%)		Number of data set
Validation	Calibration	Validation	Calibration	
0	0	0	0	1
34	0	15	0	2
4	0	5	0	3
0	0	0	0	4
9	0	5	0	5

6.2 Results of PNN Method

In the phase of training the PNN algorithm, there is a parameter named *SPREAD* which is the representative of parameter σ defined in the section of methodology. Therefore, the optimum value of *SPREAD* plays an important role in the results of this algorithm. Hence, to determine this parameter, the cross validation method was used in the present study. On the basis of this method, the error resulted from training the algorithm was calculated for a variety of values of parameter *SPREAD*. Then, the value of parameter *SPREAD* for the lowest error was selected, and by the use of this value, the algorithm was trained and tested. It is noted that this process was performed separately for each of the five data sets. The optimum value of parameter *SPREAD* and the Error Rate and Error Value of the calibration and validation phases for each data set have been presented in Table 3.

As can be seen from Table 3, the optimum value of parameter *SPREAD* is equal to 1 for data sets 1, 2, and 5, 4 for data set 3, and 22 for data set 4. The values of this parameter indicate that the spread of kernel function of the algorithm is normal for data sets 1, 2, and 5, while for data sets 3 and 4 the kernel function tends to flatten, and its spread for data set 4 is much more than that for data set 3.

Moreover, in the calibration phase, the Error Rate and Error value are equal to zero for data sets 1, 2, and 5. The ER for data set 3 is equal to 2.5 % (two misclassified data) and 1.25 % (one misclassified data) for data set 4. Accordingly, the EV for data set 3 is twice as much as that for data set 4. It means that the classification errors of all misclassified data in data set 3 and 4 are the same.

Table 3 The optimum value of parameter *SPREAD* and error indices of the PNN algorithm in calibration and validation phases

Error value		Error rate (%)		Optimum value of *SPREAD*	Number of data set
Validation	Calibration	Validation	Calibration		
10	0	10	0	1	1
25	0	10	0	1	2
2	18	10	2.5	4	3
18	9	10	1.25	22	4
2	0	10	0	1	5

Furthermore, in the validation phase, the ER is equal to 10 % (two misclassified data) for all of the data sets; however, the EV is not the same for them such that it is the least for data sets 3 and 5 and the most for data set 2.

All in all, as the ER is the same for all data sets in validation phase and its value is low for all data sets in calibration phase, it can be concluded that the performance of the PNN algorithm is not sensitive to the way of dividing the data, But, it should be considered that the EV is related to that.

6.3 Results of the KNN Method

As mentioned in the methodology section, in the KNN algorithm the value of parameter *K* affects the results. Therefore, the optimum value of the parameter *K* was determined in this study using the cross validation method. Because two pollutants, Nitrate and Chloride, had the same effect on the value of CCME WQI, their weights were considered the same in this algorithm. After determining the optimum value of parameter *K*, the algorithm was trained and tested with this value. This process was performed separately for all of the five data sets, and the optimum value of parameter *K* and the Error Rate and Error Value of the calibration and validation phases for each data set have been presented in Table 4.

According to Table 4, the optimum value of parameter *K* is equal to 2 for data sets 1 and 2, 10 for data set 3, 22 for data set 4, and 3 for data set 5. The large value of *K* for data sets 3 and 4 indicates the irregularity of scattering of data in the training data sets, which causes the difficult acceptability of the pattern by the algorithm.

Besides, in the calibration step, the Error Rate and Error Value are equal to zero for data sets 1 and 2. For data set 3, the EV is equal to 8.75 % (seven misclassified data), and it is equal to 10 % (eight misclassified data) and 6.25 % (five misclassified data) for data sets 4 and 5 respectively. Comparing the EV for data sets 3 and 4, it can be realized that although the number of misclassified data for data set 3 is fewer than that for data set 4, the magnitude of the classification errors for data set 3 is higher than that for data set 4.

Furthermore, in the validation step, the Error Rate for data set 1 is equal to 5 % (one misclassified data), and it is equal to 15 % (three misclassified data) for data sets 2 and 5, and 25 % (five misclassified data) for data sets 3 and 4. As can be seen, like the calibration step, in the phase of validation the highest value of the ER belongs to data sets 3 and 4. But, in this phase data set 4 has the highest value of the EV. Besides, although data set 5 has the fewer number of misclassified data than data set 3, its magnitude of the classification errors is higher than that of data set 3. Moreover, despite the equality of the number of misclassified data for data sets 2 and 5, the EV for data set 5 is higher than that for data set 2.

Table 4 The optimum value of parameter *K* and error indices of the KNN algorithm in calibration and validation phases

Error value		Error rate (%)		Optimum value of *K*	Number of data set
Validation	Calibration	Validation	Calibration		
9	0	5	0	2	1
19	0	15	0	2	2
21	39	25	8.75	10	3
37	32	25	10	22	4
27	29	15	6.25	3	5

Finally, it can be concluded that this algorithm is relatively sensitive to the way of dividing the data such that the best and the worst results belong to data sets 1 and 4 respectively.

6.4 Comparison of the results

As noted above, each of these three algorithms has a parameter (*C* for SVM algorithm, *SPREAD* for PNN algorithm, and *K* for KNN algorithm), the value of which affects the performance of the algorithm. Therefore, in the current study the optimum value of these parameters was determined. Table 5 illustrates the optimum value of these parameters.

Comparing the optimum value of the algorithms' parameters in Table 5 shows that:

1- For the SVM algorithm, it is necessary that an optimum value for parameter *C* be determined for each class while each of the PNN and KNN algorithms has one parameter which should be optimized only once for all of the classes. In this respect, the PNN and KNN algorithms are superior to the SVM algorithm.
2- The SVM algorithm enjoyed the lowest optimum value of parameter *C* for data sets 2, 3, and 5. However, the lowest optimum value of the PNN and KNN algorithms' parameters belongs to data sets 1, 2, and 5. As a result, all of the three algorithms jointly own the lowest optimum value of their parameters for data sets 2 and 5, which indicates the more regularity of scattering of the data in these two data sets.
3- The largest optimum value of the parameters of the three algorithms belongs to data set 4, which indicates the irregularity of scattering of data in this data set.
4- The optimum values of the parameters *SPREAD* and *K* are almost the same for all of the five data sets.

Based on the optimum value of the parameters, these three algorithms were performed, and their Error Rate and Error Value of the calibration and validation phases separately were compared in Tables 6 and 7.

Comparing the results of these three algorithms in the Tables 6 and 7 shows that:

1- In general, the SVM algorithm is a powerful algorithm compared to the two others for classifying the water quality data, having no errors in the calibration phase and the lowest total number and total value of errors in the validation phase. On the contrary, the KNN algorithm which has the most total number and total magnitude of errors in the both calibration and validation phases has the least performance of all.
2- Although the SVM algorithm has the best performance, the number and magnitude of errors of the PNN algorithm are also low in both calibration and validation phases in addition to the fact that training the PNN algorithm is much easier than the SVM

Table 5 The optimum value of the algorithms' parameters

K (KNN)	*SPREAD* (PNN)	*C* (SVM)			Number of data set
		Class 5	Class 2	Class 1	
2	1	1,000	10,000	10	1
2	1	1,000	100	10	2
10	4	1,000	100	10	3
22	22	1,000	100,000	10	4
3	1	1,000	100	10	5

Table 6 The comparison of Error Rate index (%) of the algorithms in calibration and validation phases

Validation			Calibration			Number of data set
KNN	PNN	SVM	KNN	PNN	SVM	
5	10	0	0	0	0	1
15	10	15	0	0	0	2
25	10	5	8.75	2.5	0	3
25	10	0	10	1.25	0	4
15	10	5	6.25	0	0	5

algorithm. Therefore, in situation of the absence of the SVM algorithm, the PNN algorithm can be an appropriate alternative for classification.

3- The SVM algorithm has the best results for both indices for data sets 1 and 4 while the PNN algorithm has those for data sets 1, 2, and 5, and the KNN algorithm has also those for data sets 1 and 2. Consequently, all of the three algorithms enjoy the best results for data set 1. But, the PNN and KNN algorithms have the worst results for data sets 3 and 4 while the SVM algorithm has those for data set 2.

4- The SVM algorithm has the best results for both indices for data sets 1 and 4 while the PNN algorithm has those for data sets 1, 2, and 5, and the KNN algorithm has also those for data sets 1 and 2. Consequently, all of the three algorithms enjoy the best results for data set 1. But, the PNN and KNN algorithms have the worst results for data sets 3 and 4 while the SVM algorithm has those for data set 2.

5- The differences between the results of the KNN algorithm for the five data sets are more than those for the PNN and SVM algorithms. It implies that the KNN algorithm is more sensitive to the way of dividing the data than two other algorithms.

7 Conclusion

The aim of this study is to apply the classification algorithms for water quality classification in order to reduce the computation time. In this regard, the performance of the three supervised methods of classification including SVM, probabilistic neural network (PNN), and k-nearest neighbor (KNN) has been assessed and compared. For this purpose, water quality data obtained from 100 observed wells have been used while they have been classified based on

Table 7 The comparison of Error Value index of the algorithms in calibration and validation phases

Validation			Calibration			Number of data set
KNN	PNN	SVM	KNN	PNN	SVM	
9	10	0	0	0	0	1
19	25	34	0	0	0	2
21	2	4	39	18	0	3
37	18	0	32	9	0	4
27	2	9	29	0	0	5

two pollutants Nitrate and Chloride by the use of the CCME Water Quality Index. With respect to these contaminants, water quality of the wells has been categorized into three classes Excellent, Marginal, and Poor and any well was not ranked in other classes.

To carry out this research, first the 100 water quality data were divided into 5 groups of 20 data. Then, 5 sets of data were produced in each of which 4 groups of 20 data were applied for training and one other group was applied for testing. Indeed, a 5- fold cross validation was performed to assess the effect of data value on the performance of these methods.

To assess the efficiency of the algorithms, two performance evaluation measures including Error Rate (ER) and Error Value (EV) have been defined such that the former indicates the number of misclassified data and the latter implies the magnitude of classification error of the misclassified data.

The results show that the SVM algorithm enjoys the best performance of all although the PNN algorithm also has the low number and magnitude of errors.

Moreover, the KNN algorithm, having the most total number and total value of errors, is the weakest one for classification data.

One of the important factors which affects the results of the KNN algorithm is the weight of any feature. In the current study, as two pollutants, Nitrate and Chloride, have the same effect on the value of CCME WQI, their weights have been considered the same. Nevertheless, changing the weight of these features may be able to get better results for this algorithm.

It is worth noting that the training process of the SVM algorithm is more difficult than the PNN and KNN algorithms especially because the parameter C of the SVM algorithm should be optimized for each class while the parameters of the two other algorithms should be optimized only once for all of the classes.

Comparing the results of the algorithms and the optimum value of their parameters for all of the five data sets indicates that:

Firstly, all of the three algorithms have the best results for data set 1; however, the PNN and KNN algorithms have the worst results for data sets 3 and 4 while the SVM algorithm has those for data set 2.

Secondly, the irregular scattering of data in the data set 4 is high as a result of which the largest optimum value for parameters C, *SPREAD*, and K has been obtained.

Thirdly, despite the largest optimum values of parameter C for the SVM algorithm belongs to data sets 1 and 4, which indicates the irregularity of scattering of data in the two data sets, this algorithm enjoys the best results for these data sets. On the contrary, the PNN algorithm has the best results and the lowest optimum value of parameter *SPREAD* for data sets 1, 2, and 5. Like PNN, the KNN algorithm enjoys the best results and the lowest optimum value of parameter K for data sets 1 and 2.

Finally, the KNN algorithm is more sensitive to the way of division of the data than two other algorithms.

References

Aggarwal SK, Goel A, Singh VP (2012) Stage and discharge forecasting by SVM and ANN techniques. Water Resour Manag 26:3705–3724

Araghinejad S (2014) "Data-driven modeling: using MATLAB in water resources and environmental engineering", springer, water science and technology library, volume (67)

Balabin RM, Safieva RZ, Lomakina EI (2010) Gasoline classification using near infrared (NIR) spectroscopy data: comparison of multivariate techniques. J Anal Chimica Acta 671:27–35

Burges CJC (1998) A tutorial on support vector machines for pattern recognition, vol 2. Kluwer Academic Publishers, Boston, pp 1–43

Byvatov E, Fechner U, Sadowski J, Schneider G (2003) Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. J Chem Inf Comput Sci 43:1882–1889

Cacoullos T (1966) Estimation of a multivariate density. Ann Inst Stat Maths 18:179–189

Chapelle O, Haffner P, Vapnik VN (1999) Support vector machines for histogram-based classification. IEEE Trans Neural Netw 10(5):1055–1064

Chen X, Li YS, Liu Z, Yin K, Li Z, Wai OWH, King B (2004) Integration of multi-source data for water quality classification in the Pearl River estuary and its adjacent coastal waters of Hong Kong. J Cont Shelf Res 24:1827–1843

Canadian Council of Ministers of the Environment (CCME) (2001) Canadian water quality guide-lines for the protection of aquatic life: CCME water quality index 1.0.", technical report. Canadian Council of Ministers of the environment winnipeg, Canada

Ghosh S, Katkar S (2012) Modeling uncertainty resulting from multiple downscaling methods in assessing hydrological impacts of climate change. Water Resour Manag 26:3559–3579

Hong WC, Pai PF (2007) Potential assessment of the support vector regression technique in rainfall forecasting. Water Resour Manag 21:495–513

Horton RK (1965) An index number system for rating water quality. J Wat Poll Cont Fed 37(3):300–306

Kiang MY (2003) A comparative assessment of classification methods. J Decis Support Syst 35:441–454

Kim DK, Lee JJ, Lee JH, Chang SK (2005) Application of probabilistic neural networks for prediction of concrete strength. ASCE J Mat Civ Engrg 17(3):353–362

Lee MS, Park SS (2006) Comparative analysis of classification methods for protein interaction verification system. Lect Notes Comp Sci Adv Inf Syst 4243:227–236

Ogleni N, Topal B (2011) Water quality assessment of the Mudurnu River, Turkey, using biotic indices. Water Resour Manag 25:2487–2508

Parzen E (1962) On estimation of a probability density function and mode. Ann Math Stat 33:1065–1076

Scott DW (1992) Multivariate density estimation. Wiley, New York

Shaffer RE, Rose-Pehrsson SL, McGill RA (1999) A comparison study of chemical sensor array pattern recognition algorithms. J Anal Chimica Acta 384(3):305–317

Specht, D. (1988), "Probabilistic neural networks for classification, mapping, or associative memory." *IEEE International Conference on Neural Networks*., 525–532

Tsuta, M., Masry, G. E., Sugiyama, T., Fujita, K., and Sugiyama, J. (2009). "Comparison between linear discrimination analysis and support vector machine for detecting pesticide on spinach leaf by hyper spectral imaging with excitation-emission matrix." *proceedings, European Symposium on Artificial Neural Networks-Advances in Computational Intelligence and Learning*. Bruges (Belgium), 337–342

Vapnik VN (1995) The nature of statistical learning theory. Springer, New York

Wasserman PD (1993) Advanced methods in neural computing. Van Nostrand Reinhold, New York

Werther W, Lohninger H, Stancl F, Varmuza K (1994) Classification of mass spectra: a comparison of yes/no classification methods for the recognition of simple structural properties. J Chemom Intel Lab Syst 22(1):63–76

Weston, J., and Watkins, C (1998) "Multiclass support vector machines." *Tech. Rep. CSD-TR-98-04*., Department of Computer Science, Royal Holloway, University of London

Yang Su M (2011) Real-time anomaly detection systems for denial-of-service attacks by weighted k-nearest-neighbor classifiers. J Exp Sys Appl 38:3492–3498