International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I7P215

Volume 70 Issue 7, 140-146, July 2022

Original Article

Support Vector Machine for Predicting BOD of Greywater

Samir Sadik Shaikh[1], Rekha Shahapurkar[2]

[1,2]*Computer Science & Engineering Department, Oriental University, Indore, Madhya Pradesh, India.*

[1]samirshaikh81@gmail.com

Received: 01 May 2022 Revised: 27 June 2022 Accepted: 11 July 2022 Published: 20 July 2022

Abstract *- Greywater reuse has been a time-honored tradition for decades and benefits most regions facing tremendous water scarcity. Greywater has a large-scale potential for reuse and a wide area of application. The water quality index (WQI) determines the system of reuse and area of application for greywater. Various models can be found in earlier literature to predict WQI, namely the values of TSS, TDS, TS, pH, COD, and BOD based on ANN, DNN, SVM, KNN, and other approaches. Recently, some researchers have also proposed Water quality monitoring techniques based on IoT. This study's objective is to establish an indirect way of estimating the major wastewater quality parameters constructed on machine learning techniques. SVR, a version of SVM, was used to implement the model based on the kernel trick in PYTHON. The mean squared error function is prepared to analyze the model's overall effectiveness, which was observed to be 0.514. It was also observed that the mean squared error resulted from the cost of error (C). As the value of C increases, the hyperplane becomes much smoother, thereby improving the model's performance.*

Keywords *- BOD, Error function, Greywater, Hyperplane, SVR.*

1. Introduction

Greywater can be considered as any wastewater emerging from household use except toilet wastewater (Casanova et al. 2001, Ledin et al. 2001, Ottoson 2003) [1][2][3]. Greywater may be seen as high quantity and low strength wastewater with a large-scale potential for reuse and wide application, including window and car washing, gardening, agriculture, and other non-potable uses. The physical and chemical properties of greywater are never fixed. It varies from region to region and is mostly affected by lifestyle and climatic conditions (Abedin 2013, do Couto et al. 2013, Katukiza et al. 2014) [4][5][6]. Greywater recycling has been around for decades and is proven useful in most places of the world experiencing severe water scarcity. e.g., Parts of Africa, Australia, India, etc. Some countries still do not have the practice of greywater reuse, or the laws for the same are so stringent that people avoid practicing them. If necessary attention is given to this practice, greywater reuse can help mankind to reduce their dependency on freshwater requirements.

Proper wastewater treatment can save the environment and reduce water pollution from untreated wastewater discharged into waterways. Greywater can be an additional water source in serious water shortage areas and regions with dry climates. Studies performed by Hernandez Leal et al. (2010) [7] claim that up to 75% of the water produced from households is greywater, and its volume can increase if dry toilets are utilized, up to 90%. Jamrah et al. (2011) [8] estimated that greywater contributes to nearly 69% of household water consumption. As most of the domestic wastewater originates from the bathroom or kitchen sink, it shows higher values of total suspended solids (TSS) due to washing clothes, showers, utensils, oils, etc. Wastewater may contain particles of sand, hairs, soap, detergents, oil, and other materials that increase the value of TSS. The pH value of greywater largely rests on the alkalinity/acidity based on the water supply. In addition, it is normally in the values between 5–9. The pH of greywater generally increases because of the alkaline material in soaps and detergents used in the laundry and utensils. Other important wastewater parameters are Chemical Oxygen Demand (COD) also Biochemical Oxygen Demand (BOD), where we usually witness a preponderance of COD over BOD. BOD/COD ratio determines the biodegradability of greywater. It indicates the degree to which microorganisms can decompose the organic stuff in greywater. The water quality index (WQI) of greywater determines its reuse and area of application.

Many pieces of literature exist where researchers have treated greywater using different conventional treatment techniques and recycled them for other non-potable uses. Researchers have also proposed machine learning and artificial intelligence algorithms to predict water quality indexes and classes. Various models can also be found in earlier literature to predict WQI, namely the values of TSS,

TDS, TS, pH, COD, and BOD based on the SVM, KNN, ANN, DNN, and others. Recently, some researchers have also proposed water quality monitoring solutions based on the IoT.

Abyaneh et al. (2015) [9] estimated two significant WQI, i.e., BOD and COD using four other water quality constraints such as; a temperature of water (T), alkalinity of water (pH), total solids (TS) also total suspended solids (TSS). Parameters were gathered from a wastewater treatment plant. Researchers developed two different representations based on multivariate linear regression (MLR) also Artificial neural network (ANN) techniques. The performance of these dual models is assessed built on the value of Root Mean Square Error (RMSE), also Coefficient of Correlation (r). The ANN model outperformed the MLR model, according to the results.

Chou et al. (2018) [10] used the techniques of Artificial Intelligence to determine the Carlson Trophic State Index (CTSI), a statistic aimed at water purity. The researchers applied models based on ANN, SVM, LR, and CART methods. Four different techniques were applied for each model based on balloting, sacking, stacking, and tiering. In terms of MAE, MAPE, and RMSE, the ANN model based on tiering approach performed the best, with 3.941, 3.131, and 6.786 percent, respectively.

Mohammadpour et al. (2015) [11] used SVM along with FFBP (feedforward backpropagation) also RBF for predicting WQI in a constructed wetland. The researchers collected information based on 11 water quality factors over 14 months. Out of 11 parameters, the researchers used 6 important factors such as; pH, DO, COD, BOD, TS, and AN to put the models through their paces, approve them, and train them. The findings demonstrate that FFBP and SVM models can be useful in forecasting the quality of greywater in a built wetland.

Muharemi et al. (2018) [12] used a nine-model order calculation approach to estimate the water quality location model. The F1 score is adapted to equate the performance of various representations. The model established on the Logistic regression approach showed the highest approximation, about 0.58, is an F1 score, whereas SVM, in addition to NN models, failed in performance. Researchers also concluded that the number of false negatives besides false positive predictions is highly decreased in the case of the logistic regression approach.

Shafi et al. (2018) [13] proposed an IoT-based machine learning approach to gradually forecasting water quality. An adaptable application was used to examine water quality metrics in the suggested framework remotely. For estimating water quality, four Artificial Intelligence approaches were used: Support Vector Machine (SVM), Artificial Neural Network (ANN), k Nearest Neighbor (kNN), and Deep Neural Network (DNN). Performance of t. According to the results, the DNN model is better, with a precision of 93%.

Xiang et al. (2009) [14] employed an LSSVM-PSO representation to estimate water quality in the Liuxi River in Guangzhou. LSSVM (Least Square Support Vector Machine) may solve the MLP model's limitations, and PSO (Particle Swarm Optimization) can improve the LS-bounds. SVM's in terms of accuracy, the PSO-SVM representation outperformed both the BPNN and the ARIMA representation.

2. Theory of Support Vector Machine

2.1. The Error Function

Assume we have a collection of data for training (x_1, y_1) with an uncertain distribution of probabilities, P (x, y), besides the loss function $V(y, f(x))$That computes the error created after f(x) is "predicted" as a substitute for the dependent variable's real value "y" for a given value of independent data "x." The task's goal, in this case, is to discover an error function "f," which minimizes the predicted error in forecasting fresh data. Accordingly, the error function "f" may be as written:

$$f = \int V(y, f(x))\, P(x, y) dx dy$$

Since probability distribution, i.e., when $P\ (x, y)$ is unknown, apply some sort of induction principle which can generate the empirical error function from single available training examples with minimum expected error. Under such a condition, the principle of ERM, i.e., Empirical Risk Minimization, can be applied. Formally the empirical error function "f" may be written as:

$$f = \frac{1}{l} \sum_{i=1}^{l} V(x_i, f(x_i))$$

With "f" limited to the hypothesis space - H, for example. An essential query is how near the solution's empirical error is to the predicted error's minimal value. In 1971, Vapnik and Chervonenkis (Theorem 1) [15] proposed a theorem that defined the circumstances under which the two mistakes will be near each other, as well as the probability limitations on the distance between the two errors. The hypothesis space H is used to express these constraints: as H becomes more complicated, the probability gap between the two mistakes widens.

The first theorem (Vapnik and Chervonenkis, 1971) [15] states: According to the theory, if V is a hypothesis space's VC dimension and H is a hypothesis space, and likelihood is 1, then the least anticipated error (L) the minimal empirical error, note L_{emp}, must meet the following limitation:

$$L_{emp} - 4\sqrt{2}\sqrt{\frac{V\left(1+\log\left(\frac{2l}{V}\right)\right)-\log\left(\frac{\eta}{4}\right)}{l}} \leq L \leq L_{emp} + 4\sqrt{2}\sqrt{\frac{V\left(1+\log\left(\frac{2l}{V}\right)\right)-\log\left(\frac{\eta}{4}\right)}{l}}$$

The first theorem holds regardless of the data P's probability distribution (x, y).

When ERM is applied to a series of hierarchical hypothesis spaces, such as H_1, H_2, and H_3, …. Structural Risk Minimization (SRM) is the term for it (Vapnik, 1998) [16]. Evaluating the "complexity" of such a hierarchical hypothesis space to identify the best solution to such a learning issue is an important question that emerges from Statistical Learning Theory. Vayatis and Azencott [17] emphasized the study of complicated parameters connected to hypothesis space in 1999.

The distribution P (x, y) is considered as well in (Vayatis and Azencott) [17]. Also, the gap between empirical and predicted error is confined using a "complex" measure. The proposed constraints are tighter than theorem one, but they need an understanding of the distribution-dependent complex variable. To summarize, (Vayatis and Azencott, 1999) [17] discussed the elementary conceptual structure in which SVM and other learning machines are initially established also; their findings point to potential research avenues that might lead to advances in the theories, and hence advancements in SVM (i.e., the concept could just select SVM variables like the kernel, in addition, the regularization parameter C).

2.2. Formulation of SVM

The principles given in the previous section are used to formulate SVM. To understand SVM, two things now have to be understood: first, the hypothesis space that SVM works with, then second, the loss function. Support Vector Machine has the advantage of finding an "optimal" possible answer to the learning problem, hyperplane. The linear solution, in which the answer to the learning problem falls on input vector X's space, is the simplest formulation of SVM. Under such circumstances, as a subset of all hyperplanes, the hypothesis space is considered and may be written as:

$$f(x) = w.x + b$$

The universal explanation of SVM is that it seeks to locate a hyperplane that falls in the spotlight caused in the kernel K rather than the hyperplane that merely sits given the input vector X's space. (Wahba, 1990) [18]. The hypothesis space may alternatively be regarded by way of a collection of related actions during a Reproducing Kernel Hilbert Space (RKHS) expressed as means of K (Wahba, 1990) [18], thanks to the feature space caused by the kernel K. (Vapnik, 1998) [16]. Based on these observations, we may conclude that SVM's hypothesis space seems to be a subgroup of the hyperplanes' set sometimes described in space an RKHS, which is technically expressed by employing

$$\{f: ||f||_k^2 < \infty\}$$

Where in K is the kernel that determines the RKHS in addition $||f||_k^2$ is the function's RKHS norm (Wahba, 1990) While in the case of linear SVM, which has the form $f(x) = w.x + b$, such functions' RKHS norm is the norm of w, explicitly $||f||_k^2 = ||w||^2$. In the case of a linear relationship, SVM considers subsets with this space that are of the type

$$\{f: ||f||_k^2 < A^2\}$$

For a constant A,

The constant A remains utilized to determine the construction of the hypothesis space in a Statistical Linear Training problem; the bigger the value of A, the more complicated the hypothesis space. SVM's goal is to discover the answer for the "optimal" RKHS norm, i.e., the ideal value of A. Instead of searching a large number of hypothesis spaces in sequence using the ERM principle for the choice of A individually, SVM searches for the optimum value of A in a dissimilar way, this has received a lot of attention throughout the studies presented by (Bartlett and Shawe-Taylor, 1998) [19], (Burges, 1998) [20], and (Evgeniou et al., 1999) [21].

After SVM has determined the hypothesis space, the next decision is regarding the loss function. To do so, we must first distinguish between SVM regressor and SVM classifiers. The loss function arises owing to misclassification in classification issues, and this mistake must be reduced; hence, in classification problems, a loss function of the type $sign\,(-yf(x))$ must be utilized. In classification problems, y has binary values of 1; also, the sign of the function f(x) is used to classify things. Though, owing to expansion (Vapnik, 1998) [16], for SVM classification, the actual loss function is defined as |1-yf(x)|, commonly known as the "soft margin" loss function due to its traditional "margin" meaning.

Regarding SVM classification, the margin is a crucial geometric quantity.
The loss function utilized for regression problems is the epsilon insensitive loss function,

$|y - f(x)|\varepsilon = |y - f(x) - \varepsilon\ if\ |y - f(x)| > \varepsilon$, and 0 or else.

To summarise, SVMs are learning machines that evaluate the "complexity" space for hypotheses created via minimizing the RKHS norm of a result $||f||_k^2$. Along with reality, SVM minimizes the trade-off between experimental error with hypothesis space difficulty. It is accomplished informally through complex minimization problem execution:

SVM classifier:

$$\min \ ||f||_k^2 + C \sum_{i=1}^{l} |\, 1 - yi\, f(xi)|$$

SVM regressor:

$$\min \ ||f||_k^2 + C \sum_{i=1}^{l} |\, yi - f(xi)|$$

Here C means a "regularization parameter" that regulates the balance among logical errors in addition to the complication of the hypothesis space (Trafalis, 1999) [22], (Evgeniou, Theodoros, 2001) [23].

2.3. Implementation of SVM

Assume we have a set of training samples that may be classified into two groups, i.e., two Classes (circle as well as a square), as shown in Fig. 1. We need to develop a binary SVM classifier that remains accomplished to classify the test data samples into each of two categories to categorize the test dataset into two categories. To develop the classifier, we'll require a training data set wherein the samples are represented as p-dimensional vectors. We aim to discover a (p1) dimensional hyperplane that effectively divides the dataset into two categories. We can have a huge number of hyperplanes, as illustrated in Fig. 1, to divide the dataset into two groups. We can observe from H_1 that the hyperplane leads to misclassification. The distance between the hyperplane and the samples closest to it is insufficient for H_2. H_3, a suitable hyperplane, signifies the greatest separation or margin among the two groups. In the other sense, the average distance between each category's closest data point and the selected hyperplane is maximized. When SVM categorizes a dataset, it seeks the hyperplane with the highest margin. (Burges, 1998) [20].

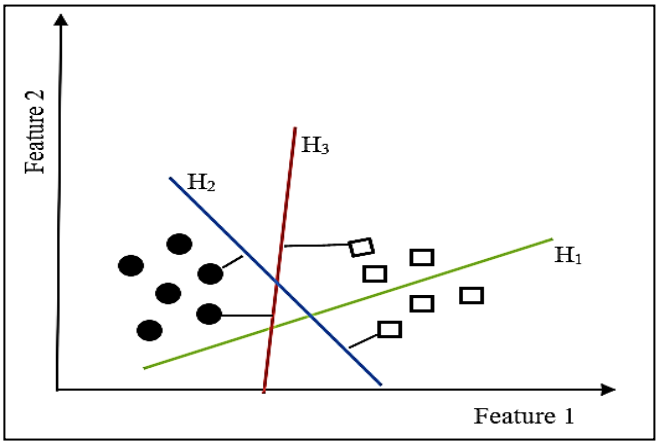

Fig. 1 Hyperplanes: All courses are not separated in H_1. H_2 has a little advantage, then only by a short margin. H_3 splits them by the widest possible margin.

A nonlinear problem remains defined as the categorization of data into two groups. Nevertheless, the linear SVM approach may also be used for a nonlinear variant, which effectively classifies nonlinear datasets using the kernel trick. The dataset's training samples are first recorded using the kernel function in a high dimensional space to model SVM. Then the Support Vector Machine chooses the best hyperplane to categorize these illustrations into distinct categories. We initially examine linear and nonlinear SVM before discussing multiclass classification using SVM.

2.3.1. Linear SVM

Assume we have a group with M training samples: $xi\ (i = 1, 2, \ldots, M)$. Consider the following scenario: the training dataset comprises two classes labelled as positive and negative for ease of separation. In this case, the positive class's anticipated output is $y_i = 1$, and the negative class's expected output is $yi = -1$. As a result, finding a hyperplane $f(x) = 0$ that correctly classifies the supplied dataset is our goal for error-free classification. The hyperplane in linear SVM can be written as follows:

$$f(x) = wTx + b = 0$$

$$Hence;\ f(x) = \sum_{j=1}^{M} w_j x_j + b = 0$$

w is an M-dimensional vector, and b is a scalar; they also define the hyperplane.

Suppose we can linearly divide these labelled data samples (positive and negative class data). In that case, we will receive two additional there will be no points between the samples since they will be separated by hyperplanes (Fig 2). Separating hyperplanes are extra is generated in this way also the region delimited because of them is termed "margin." SVM's goal is to increase this margin to enhance classification accuracy. The following equations can be used to characterize these two hyperplanes:

$$f(x_i) = 1\ if\ y_i = 1,$$

$$f(x_i) = -1\ if\ y_i = -1$$

Alternatively, the equation may be written as:

$$y_i f\ (x_i) = y_i\ (wTx_i + b) \geq 1\ for\ i = 1,2,\cdots M.$$

As a result, the best hyperplane is the one that generates the most margin.

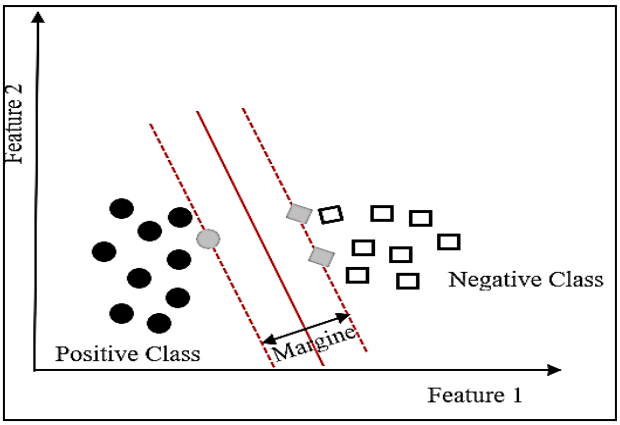

Fig. 2 SVM is being used to classify two classes.

Fig 2 depicts many samples with two distinct labels: circles for the positive class and squares for the negative class. SVM works by determining a hypothesis plane, maximizing the gap between the two hyperplanes between the two classes to separate them further (marked as a dotted line). Support vectors, shown by grey circles and squares, are samples that lie on or near these two dotted hyperplanes. The remaining samples aren't as crucial once the support vectors have been chosen because they include the information needed to define the classifier.

According to the previous discussion, the region enclosed by the dotted hyperplane is the margin equal to $||w||^{-2}$. As a result, by resolving the optimization problem below, the best separation hyperplane may be found:

$$\text{minimize } \frac{1}{2} ||w||^{-2}$$

$$\text{depending on } y_i (w^T x_i + b) \geq 1, \text{ for } i = 1, 2, \cdots M$$

The issue as mentioned above can be transformed into a Lagrangian double problem using Karush–Kuhn–Tucker (KKT) precondition. As a result, its double quadratic optimization problem may be written as follows:

$$maximize\ L(a) = \sum_{i=1}^{M} a_i - \frac{1}{2} \sum_{i,j=0}^{M} a_i a_j y_i y_j x_i T \cdot x_j$$

$$subject\ to\ ai\ \geq\ 0, for\ i = 1, 2, \cdots M.\ \sum_{i=1}^{M} a_i y_i = 0.$$

The coefficients of a_i are found by solving the preceding problem, giving the decision function as:

$$f(x) = sign\ (\sum_{i,j=1}^{M} a_i y_i x_i T \cdot x_j + b)$$

2.3.2. Soft margin

Consider an issue in which we can't discover a hyperplane that can readily divide the two classes owing to noisy data; in this case, the linear SVM will not be able to discover an optimum solution in the direction of the problem. The soft margin method is employed to fix the issue in this situation. The soft margin method works by presenting positive slack variables i resulting in the following original optimization problem:

$$minimize\ \frac{1}{2} ||w||^{-2} + C \sum_{i=1}^{M} \xi i$$

$$subject\ to: \begin{Bmatrix} y_i (wT x_i + b) \geq 1 - \xi_i, & \text{for } i = 1,2, \cdots M \\ \xi_i \geq 0, & \text{for } i = 1, 2, \cdots M \end{Bmatrix}$$

2.3.3. Nonlinear classification

When data collection is not linearly separable, SVM may be used to handle nonlinear classification issues. Nonlinear samples are transferred by employing a high dimensional feature space, the kernel trick, permitting linear classification. A kernel function k (x, y) is the inner product of the samples, where k (x, y) = (ø(x), ø(y)). Kernel functions that are valid and essential meet Mercer's requirements, which state that k (y, x) is essentially equal to k (x, y).

The kernel trick may be used to create a nonlinear version of SVM, then the dual form of the optimization issue can be written as:

$$maximize\ L(a) = \sum_{i=1}^{M} a_i - \frac{1}{2} \sum_{i,j=0}^{M} a_i a_j y_i y_j k (x_i, x_j)$$

$$subject\ to: \begin{Bmatrix} 0 \leq a_i \leq C, & \text{for } i = 1,2, \cdots M \\ \sum_{i=1}^{M} a_i y_i = 0 & \end{Bmatrix}$$

The nonlinear SVM's decision function may be stated as follows:

$$f(x) = sign(\sum_{i,j=1}^{M} a_i y_i k(x_i, x_j) + b)$$

To conclude, samples whose coefficients a_i are not equal to zero are those that lie on the separating hyperplane or are very near to the separate hyperplanes. Support vectors are the names given to these samples. The information required to generate the ideal hyperplane is included in support vectors, but other samples do not influence the development of the optimal hyperplane. This is why SVM may be utilized even when data samples are small compared to other classification algorithms.

For nonlinear situations, SVM may use various kernel functions, including linear, polynomial, and Gaussian RBF. Since the kernel specifies the high-dimensional space where the samples will be categorized, selecting the right kernel function is crucial to SVM performance. In intelligent fault diagnostics cases, Gaussian RBF is among the utmost usually employed kernel functions (Yaguo Lei 2017) [24].

3. Implementing The SVM Model

Several academics have previously sought to solve diverse water engineering difficulties with Regression Trees (RT) and Support Vector Regression (SVR). Support Vector Regression (SVR) is a two-layer machine-learning algorithm Support Vector Machine (SVM) differs in this way. This approach comprises two layers: kernel outputs weighted sum and kernel function weighting on the input data sequence. The wastewater quality metrics Chemical Oxygen Demand (COD), Biochemical Oxygen Demand (BOD), Total Suspended Solids (TSS), Total Dissolved Solids (TDS), and alkalinity (pH) reflect the principal pollutants present in greywater.

Algorithm:

1. Import SVM. (SVR)
2. Input Sample data (X_{ij}, Y_i) ([[X_{11}, X_{21}…X_{n1}], [X_{12}, X_{22}…. X_{n2}] … [X_{1n}, X_{2n}…X_{nn}]]), ([[Y_1], [Y_2] … [Y_n]])

3. Separate the samples interested in training and testing data. (X_{train}, X_{test}, Y_{train}, Y_{test})
4. Create the model (fit.model())
5. Train the model (Kernel trick)
6. Test the model. (Y_{test})
7. Determine model accuracy. (predict model_accuracy)

In the case of nonlinear SVM, the optimization goal should be to discover the smoothest function in the kernel's feature space rather than the input data space. Mercer's condition must be met in this case, which states that k (x, y) should be equivalent to k (y, x) in some feature space, which is a dot product. This investigation employs a radial basis function (RBF) as the kernel. The RBF is in the following format:

$$k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$$

In the PYTHON environment, a Support Vector Regression technique was built. A test and error iteration method is often used to detect the model's greatest structure and constraints. The epsilon (ε) parameter was used to manage the difference between the predicted and goal values. Finally, the value of ε is set near 0.01, with the cost of a C error, which influences the function's flatness, set near 1500. The model's performance was assessed using the mean squared error (MSE), which was determined to be 0.5147, as shown in Fig. 3. Calculated MSE using target output and predicted output represented in Table 1.

```
Results:
Input_samples [[ 19.8   7.3 462.  118. ]
 [ 25.9   8.4 833.  447. ]
 [ 21.1   7.5 523.  160. ]
 [ 24.6   8.1 721.  327. ]
 [ 21.7   7.6 554.  181. ]
 [ 23.7   7.9 646.  246. ]]
Predicted_output [124.86272076 411.22046781 168.08428697 318.77858364 190.39192241
 256.98108032]
Target_output [[124]
 [411]
 [169]
 [319]
 [191]
 [258]]
MSE 0.5147340645679228
```

Fig. 3 Calculation of MSE

Table 1. Calculation of MSE using target output and predicted output

Target Output	Predicted Output	Squared Error
124	124.86272076	0.86272076
411	411.22046781	0.22046781
169	168.08428697	0.91571303
319	318.77858364	0.22141636
191	190.39192241	0.60807759
258	256.98108032	1.01891968
		MSE = 0.51473406

Fig. 4 compares targeted output and predicted output of BOD values. The results show a minimum error between the target and the predicted value of BOD.

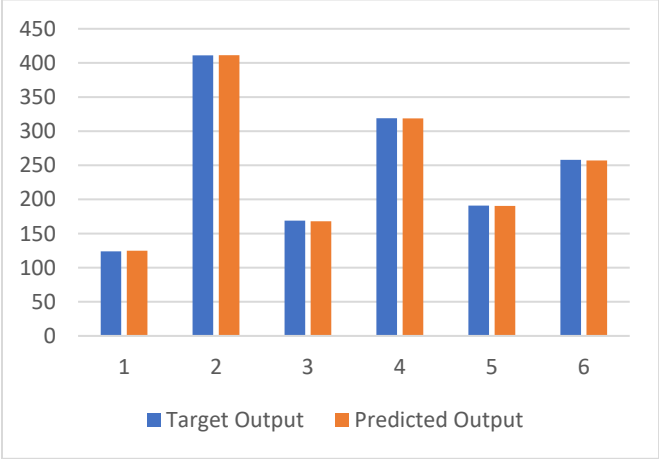

Fig. 4 Target Output vs. Predicted Output

4. Conclusion

Support vector machines are machine learning algorithms based on kernel tricks. SVR, a version of SVM, was used to implement the model based on the kernel trick. During trial and error, linear and polynomial kernels were used to implement the model, but the results were not as promising. Lastly, the Radial basis function (RBF) is also employed by the kernel, and using the trial and error method, the value of C (cost of error) gradually increased from 1.0 to 1500. The results show a minimum error between the target and the predicted value of BOD. The mean squared error function is used for analyzing the model's performance, which was observed to be 0.5147. It was also observed that the mean squared error is a function of the cost of error (C). As the value of C increases, the hyperplane becomes much smoother, thereby improving the model's performance in terms of predictability.

References

[1] Casanova LM, Gerba CP, Karpiscak M, " Chemical and Microbial Characterization of Household Greywater," *Journal of Environmental Science and Health, Part A. Toxic/Hazardous Substances and Environmental Engineering*, Vol.34, Pp. 395–401, 2001.

[2] Ledin A, Eriksson E, Henze M, "Aspects of Groundwater Recharge Using Grey Wastewater, " In: Lens P, Zeeman G, Lettinga G (Eds) Decentralised Sanitation and Reuse: *Concepts, Systems and Implementation IWA Publishing*, London, Pp. 354–370, 2001.

[3] Ottoson J, Stenstrom TA, "Fecal Contamination of Greywater and Associated Microbial Risks, "*Water Research*, Vol.37, Pp. 645–655, 2003.

[4] Abedin SB, Rakib ZB, "Generation and Quality Analysis of Greywater At Dhaka City*," Environmental Research, Engineering, and Management*, Vol. 64, Pp. 29–41, 2013.

[5] De Aguir Do Couto E, Calijuri ML, Assemany PP, Da Fonseca Santiago A, De Castro Carvalho I. "Greywater Production In Airports: Qualitative and Quantitative Assessment Resources," *Conservation and Recycling*, Vol.77, Pp. 44–51, 2013.

[6] Katukiza AY, Ronteltap M, Niwagaba CB, Kansiime F, Lens PNL, "Grey Water Characterization and Pollutant Loads In An Urban Slum," *International Journal of Environmental Science and Technology*, Vol.12, Pp. 423–436, 2014.

[7] Hernandez Leal L, Temmink H, Zeeman G, Buisman C, " Comparison of Three Systems For Biological Greywater Treatment, Water," Vol.2, Pp.155–169, 2010.

[8] Jamrah A, Al Omari A, Al Qasem L, Abdel Ghani N, " Assessment of Availability and Characteristics of Greywater In Amman," *Water Science and Technology*, Vol.50, Pp.157–164, 2011.

[9] H.Z. Abyaneh, "Evaluation of Multivariate Linear Regression and Artificial Neural Networks In Prediction of Water Quality Parameters*," J. Environ. Health Sci. Eng*, Pp.12–40, 2015.

[10] J.S. Chou, C.C. Ho, H.S. Hoang, "Determining Quality of Water In Reservoir Using Machine Learning," *Ecol. Inform*, Vol.44, Pp. 57–75, 2018.

[11] R. Mohammadpour, S. Shaharuddin, C.K. Chang, N.A. Zakaria, A. Ab Ghani, N.W. Chan, "Prediction of Water Quality Index In Constructed Wetlands Using Support Vector Machine," *Environ. Sci. Pollut. Res*, Vol.22, No.8, Pp.6208–6219, 2015.

[12] Muharemi, D. Logof˘atu, C. Andersson, F. Leon, "Approaches To Building A Detection Model For Water Quality: A Case Study, In Modern Approaches For Intelligent Information and Database Systems," Springer, Pp. 173–183, 2018.

[13] U. Shafi, R. Mumtaz, H. Anwar, A.M. Qamar, H. Khurshid, "Surface Water Pollution Detection Using Internet of Things," *In Proceedings of 15th International Conference on Smart Cities: Improving Quality of Life Using ICT & Iot*, Pakistan, Vol.8, No.10, Pp. 92–96, 2018.

[14] Y. Xiang, L. Jiang, "Water Quality Prediction Using LSSVM and Particle Swarm Optimization, Second International Workshop, Knowledge Discovery and Data Mining WKDD," Pp. 900–904, 2009.

[15] Vapnik V. and Chervonenkis A., "on the Uniform Convergence of Relative Frequencies of Events To Their Probabilities," *In Th. Prob. and Its Applications*, Vol.17, No.2, Pp.264—280, 1971.

[16] Vapnik V., "Statistical Learning Theory," Wiley, New York, 1998.

[17] Vayatis N. and Azencott R., "How To Estimate the Vapnik-Chervonenkis Dimension of Support Vector Machines Through Simulations," ACAI99, 1999.

[18] Wahba G., "Splines Models For Observational Data," *Series In Applied Mathematics*, SIAM, Vol.59.

[19] Bartlett P. and Shawe-Taylor J., "Generalization Performance of Support Vector Machine and Other Pattern Classifiers," In C.~Burges B.~Scholkopf, Editor, "*Advances In Kernel Methods--Support Vector Learning*." MIT Press, 1998.

[20] Burges C., "A Tutorial on Support Vector Machines For Pattern Recognition," In "Data Mining and Knowledge Discovery," *Kluwer Academic Publishers*, Boston, Vol.2 ,1998.

[21] Evgeniou T., Pontil M., and Poggio T., "A Unified Framework For Regularization Networks and Support Vector Machines" *A.I. Memo* No. 1654, *Artificial Intelligence Laboratory*, MIT, 1999.

[22] Trafalis T., "Primal-Dual Optimization Methods In Neural Networks and Support Vector Machines Training," ACAI99, 1999.

[23] Evgeniou T, Pontil M., "Support Vector Machines: Theory and Applications," *Conference: Machine Learning and Its Applications, Advanced Lectures*, 2001.

[24] Yaguo Lei, "Chapter 3 Individual Intelligent Method-Based Fault Diagnosis, In Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery," *Science Direct*, Pp. 67-174, 2017.