Performance comparison of machine learning methods for prediction of estimating water production

A P Widowo, E A Sarwoko, and Suhartono
Department of Informatics, Faculty of Science and Mathematics,
Diponegoro University
Jl. Prof Soedarto, SH, Tembalang, Semarang 50275, Indonesia
E-mail: arispw@undip.ac.id

Abstract. The Regional Water Company (PDAM) of Grobogan Regency has endeavored to provide clean water services to the people of Grobogan Regency as much as possible. Although it has been running so far, PDAM still has several problems related to clean water production, among them is the increasing number of customers resulting in demand for clean water increases, not only that, the water loss factor is still high enough that the distribution of water is not going well. Therefore, a study is needed to consider the problem-solving in the form of clean water production prediction that can help PDAM determine policy regarding water production. This study performs a comparison of machine learning method to predict water production of PDAM Grobogan District. The methods used in the prediction are Neural Network, Deep Learning, and k-Nearest Neighbor. Processed data are influencing factors in producing clean water, including the number of customers, the amount of water sold, the amount of water loss and the amount of water source discharge. Data obtained from PDAM monthly reports for the last 5 years (January 2011 - December 2015).

1. Introduction
Water is an important compound for all life forms known on Earth. Water is the primary type of resource that commonly used by humans to consume or to carry out their daily activities. Water deprivation in the human body can cause dehydration. Moreover, without water, humans cannot carry out activities such as cooking, bathing or other activities. Over time, the increasing rate of population growth has resulted in increasing of people's need for clean water. This has an impact on the supply and service of clean water starting from aspects of quantity, quality, and continuity. Handling for meeting the needs of clean water can be done in various ways, in this case, the government gives authority to Regional Water Supply Companies (PDAMs) to develop and manage clean water consumed by the public.

Associated with the factors that affect the company in producing clean water, among others, the number of customers that always increases every month, this certainly makes the PDAM must try to maximize the amount of water production. The NRW (Non-Reserview Water) factor is related to the loss of water that has been produced by the PDAM, the loss of PDAM water related to the water that does not reach the customer, until now the PDAM has sold 4590276 m$^3$ of water. Water loss of 36.86% of the total production of water produced is 7270484.20 m$^3$, this number is still higher than the such as pipeline leaks, damage to water meters in raw water source dams, and inadequate human resources. The PDAM has attempted to overcome this problem by repairing the pipeline and also renewing the water meter, providing training to all its employees, but despite its best efforts, there are still other problems related to the loss of water produced by the PDAM, namely the recent happened in
the case of water theft carried out by irresponsible elements by cutting the pipe outside the meter then channeled it into the customer's home [1]. This is certainly a big problem for PDAMs because they have to suffer losses that are not small. When the PDAM is still experiencing water loss problems, the distribution of water to customers will be hampered. So that it is not only the PDAM who feels aggrieved but of course the customer will also feel a loss. With some problems faced, one of the study materials as input to find out PDAM water production in the following months, the PDAM as a government company providing clean water for Grobogan Regency requires a prediction of Grobogan community's clean water production. The prediction can be used as a benchmark in determining policies in the field of water production. This software can also be useful for customers to monitor PDAM performance regarding producing clean water.

Predictions can be calculated using a variety of machine learning methods, some predictive methods that are often used and developing today are backpropagation, deep learning, and k-Nearest Neighbor. The previous study about water production prediction had been successfully done using the neural network and resulted that the performance outperformed the performance of econometric model [2]. As for the methods, some researchers had also used the neural network, deep learning, and kNN to make some predictions. Zao [3] combined Backpropagation Neural Network with a Kalman filter to predict the quality of water and proved to be able to improve the accuracy value. Backpropagation Neural Network had also been applied to predict the toxicant levels of polluted soil and resulted in the average R² value of 0.9994 [4]. Another study used deep learning, conducted by Fan [5], to predict the building cooling load profiles and enhance the prediction performance. A study about the estimation of building energy consumption was done using deep learning algorithm [6]. Meanwhile, Wahid and Kim used K-Nearest Neighbor as prediction approach for analyzing the energy consumption in residential buildings [7].

Although they have some disadvantages such as training results that are not constant and unknown in detail how the results of the prediction are obtained because these methods cannot provide information about the most important weights among their input patterns, this method also has advantages. The advantages of this method can formulate experience and knowledge of forecasters, as well as being very flexible in changing forecast rules. Moreover, to the best of our knowledge, there hadn't a previous study that predicts the estimation of water production using these methods. So in this case, the prediction comparison will be made from the three methods to predict the water production estimation.

2. Literature and Method Review

2.1. Backpropagation

Artificial Neural Network (ANNet) is an information processing system that has characteristics as same as to neural networks. ANNet is formed as an inheritance of the mathematical model of biological neural networks, assume that:

1) The process happening on several nodes called neurons
2) Between neurons, the data are sending through connectors
3) Connecting between neurons has a weight that will strengthen or weakness signal
4) To get the output, every neuron used the activation function. The amount of output is compared with a threshold.

The principle of Neural Network modeling is developed from the characteristics and worked as same as the human brain, wherein processed information the human brain contains a number of neurons that are interconnecting so that they can perform complex processing functions. Information process can only be done after going through the previous learning process. Information processing on artificial neural networks is adaptive, that means the relationship between neurons occurs dynamic, the strength of the relationship between neurons can updating from time to time, and always can learn new information [8].

The neural model is shown, its ability in emulation, analysis, prediction, and association. The capabilities possessed by ANN can be used to learn and produce rules or operations from many examples or input that is inputted and making predictions the possible output that will appear or store
the characteristics of the input stored in it [9]. The structure of neurons in artificial neural networks is described in Figure 1.

![Figure 1. ANN Neurons Structure [9]](image)

Some input signals are $x_1, x_2, \ldots, x_n$ multiplied by each corresponding weight, $w_1, w_2, \ldots, w_n$. The sum of all multiplication results ($\sum$) is performed to produce the output value ($V_k$). The $V_k$ value is activated by an activation function ($\varphi(.)$) to determine the output signal ($Y$). Artificial neural networks have several models, and one of them is the backpropagation model, which will be used in the final project as a method to predict the amount of PDAM clean water production.

2.2. Backpropagation Method
Backpropagation training algorithm or can be translated into back propagation, first formulated by Werbos and popularized by Rumelhart and McClelland for use in ANNet; then this algorithm is usually adopted as backpropagation named. This algorithm includes supervised training methods and is designed for operation on a many layer feed forward network.

Backpropagation is a neural network model with many layers that are often used in time series estimation. The backpropagation learning algorithm activates neurons using an activation function that can be differentiated to get an output error. Then this error output is used to change the weight value backward. Modification of changes in weight is done to minimized the errors that occur [10].

2.3. Backpropagation Network Architecture
Backpropagations has any units that it is single or than more hidden layers. Figure 2 is a backpropagations network architecture with $n$ input (plus a bias), a hidden layer contains $p$ units (plus a bias), and $m$ output units. $y_j$ have been on the line weight from input unit $x_i$ to hidden layer unit $z_j$ ($v_{j0}$ is the line weight. It connects the bias in the input unit to the $z_j$ hidden layer unit). $w_{kj}$ is the weight of the hidden layer unit $z_j$ to the output unit $y_k$ ($w_{k0}$ is the weight of the bias in hidden layer to $z_k$ output unit).
2.4. Activation Function

In backpropagation, the activated function used must several satisfied conditions, naming: continuous, different easily and it is a function that does not down. One function that fulfills these three conditions so that it is often used is a binary sigmoid function that has a range of values from 0 to 1. The binary sigmoid function equation can be seen in equation (1). Graphs of binary sigmoid functions are given on described Figure 3.

\[ f(x) = \frac{1}{1 + e^{-x}} \text{ with the derivative } f'(x) = f(x) (1-f(x)) \]  

(1)

![Figure 3. Graph Function of Binary Sigmoid Activation](image)

The other function was often used that it was a bipolar sigmoid function which functions is as same as to the binary sigmoid function, but with a range (-1, 1). The bipolar sigmoid function equation can be seen in equation (2). Bipolar Graphs sigmoid function are showing in Figure 4.

\[ f(x) = \frac{2}{1+e^{-x}} - 1 \text{ with the derivative } f'(x) = \frac{(1+f(x))(1-f(x))}{2} \]  

(2)

![Figure 4. Graph Function of Bipolar Sigmoid Activation](image)

The sigmoid function has been a maximum value = 1. Then, the patterns have value with a target > 1, the first, input and output patterns must be transformed. They have the same as range the sigmoid function used.

If using the binary sigmoid activation function, the data must be normalized to the interval [0, 1]. However, it would be better if normalized to smaller intervals, for example at intervals [0.1; 0.9]. Given that the sigmoid function is an asymptotic function that never touches the axis, so the value is never 0 or 1. To normalize the data to an interval [0.1; 0.9] uses equation (3) [11].

\[ y' = \frac{0.8(y-a)}{b-a} + 0.1 \]  

(3)

where:

- \( y' \) = data value after normalization
- \( y \) = data value before normalization
- \( a \) = minimum data value
- \( b \) = maximum data value
2.5. Training Process in Backpropagation Method

Backpropagation learning there are three steps. The first step is the feed forward. The input pattern is calculated forward begin input layer to the output layer used the specific activation function. Second, it is a backward step. Differences between network output and target goal are error finding. They are proposed backward, begin from the line directly relationship to units on the output layer. Third, it is modifying the weights to minimized errors was find [11].

2.5.1. Phase I: Forward Propagation

On the process forward propagation step, the input signal \((x_i)\) is proposed to the hidden layer using specific activation function. Output of each hidden unit \((z_j)\) was proposed forward again to the hidden layer. It used a specific activation function, and then until it produces network output \((y_k)\).

Network output \((y_k)\) is comparison to the goal that must be achieved \((t_k)\). If an error is smaller than specific limit tolerance, then the iteration is stopped. Soon, if the error is still large from the tolerance limit, then the weight of all line on the network will be modified to minimized errors.

2.5.2. Phase I: Backward Propagation

Based on the errors \(t_k - y_k\), calculated the factors \(\delta_k\) \((k = 1, 2, ..., m)\) used to disperse errors in the unit \(y_k\). \(\delta_k\) is also used to update the line weights that are directly related to the output unit.

Using the same method, calculated the factor \(\delta_j\) on every hidden layer unit as updating based on it of all lines derived from hidden units on the layer below. So, until all the \(\delta\) factors in the hidden unit, that they are directly related to input unit are computed.

2.6. Deep Learning

Generally, Deep Learning was called DL. DL is a machine learning branch that supports a set of algorithms to model high-level abstractions in knowledge by victimizing architectural models with complex structures or vice versa, contains several non-linear transformations. DL is a broader family element of machine learning that supports learning representation of information. Ordinary observation associates are diagrammed into differences ways such as value intensity vectors for all component, region of a particular form, and others. Some representations allow easier learning through previous, examples (e.g. face recognition, facial expression recognition). It is guaranteed from the use of DL is as a substitute for custom algorithm options in supervised learning or semi-supervised feature learning and multilevel feature extraction.

The Boltzmann machine is modeling with the input layer and a hidden layer which it usually contained of every binary unit. Hidden layer is processed as stochastic (deterministic), recurrent (feed-forward). Generating model can estimate the distribution of observations for traditional discriminative networks with labels. Energy on the network and the probability of a unit state (Scalar T is expressed as temperature) are described on the equation (4).

\[ E(s) = -\sum_i a_i s_i - \sum_{i<j} s_i w_{ij} s_j \]  \hspace{1cm} (4)

- A bipartite chart: There is no connection between the next step, feed-forward. RBM has no T factor, the rest as same as BM.
- The important features of the RBM are a visible unit and hidden contains an independent unit, which it was getting good results, as in equation (5), (6), and (7).

\[ P(s_j = 1) = \frac{1}{1+e^{-\sum_i w_{ij} s_i}} = \sigma(\sum_i w_{ij} s_i) \]  \hspace{1cm} (5)
\[ P(v|h) = \prod_{i=1}^m P(v_i|h) \] \hspace{1cm} (6)
\[ P(v|h) = \prod_{i=1}^m P(v_i|h) \] \hspace{1cm} (7)

Characters are used to define on a Restricted Boltzmann Machine
- Circumstances overall units: obtained through a probability distribution
• Weight on the network: obtained through training
• On earlier, RBM aims to estimate the distribution of input data. The target is entirely solved by weight and input
• Energy defined for RBM as in equation (8)
\[ E(v, h) = -\sum_i a_i v_i - \sum_j b_j h_j - \sum \sum h_j w_{ij} v_i \] (8)
• Visible layer distribution on RBM as in equation (9)
\[ p(v) = \frac{1}{Z} \sum_h e^{-E(v, h)} \] (9)
• Z is a partition function. It is defined as count every possible config (v, h)
• Training for RBM: Maximum Likelihood studied the probability of vector x with parameter W (weight) is describe on equation (10) and (11).
\[ p(x; W) = \frac{1}{Z(W)} e^{-E(x; W)} \] (10)
\[ Z(W) = \sum_x e^{-E(x; W)} \] (11)

2.7. K-Nearest Neighbor

Based on learning by analogy algorithm, the k-Nearest Neighbor algorithm compares a given test example with training examples that are similar to it. The training samples contain n attributes. Every sample represents a point on n-dimension space, and all of the training samples are placed in an n-dimension space. When given an unknown data, k-Nearest neighbor algorithm finds the pattern space for k training samples that it is closing to it. Closeness can be defined as a distance metric, computed using Euclidean distance.

K-Nearest Neighbor algorithm is one of simple machine learning algorithms. The basic k-Nearest Neighbor algorithm is joined of two steps: Find the k (small positive integer) training samples that are closing to the unknown data. Commonly, to get an occurring classification for these k samples (or, in the case of regression, take average values of this k label).

The neighbors are taken from a set of data samples for which the correct classification (or, in the case of regression, the value of the label) is knowing. Called as a training set for the algorithm, through no specific training step has been need. If k = 2, then the unknown data merely is assignment to the classes of its two nearest neighbors. Regression can use as same as the method, that it is by assigning the label value for the unknown data to be the average of the values of its k-nearest neighbors. It can be used to show the role weights of the neighbors so that the nearer neighbors role than more the distant ones.

3. Result and Analysis

The prediction results of each method are presented in graphical form, the results of the prediction of Deep Learning, Neural Network, and k-Nearest Neighbor methods are shown respectively in Figure 5, Figure 6, and Figure 7. Based on the prediction results in Figure 5, Figure 6, and Figure 7, it can be seen that the k-Nearest Neighbor produced the best prediction. The prediction results from the Deep Learning and Neural Network method had been able to follow the actual target path, but it was still not optimum, there were still some major errors occurred in the predictions. Deep Learning method required much more data to produce a better prediction, and the use of Neural Network method in this study did not use any optimization or parameter tuning (using default setting in RapidMiner) so that the results of the prediction had not produced high accuracy value. K-Nearest Neighbor was not model-based, it calculated the distance and compared every training sample to give the prediction, the data used in this study was simple and did not need much observation, so the predictions produced by k-Nearest Neighbor was good.
4. Conclusion

In this paper, we performed a comparison of machine learning method to predict water production of PDAM Grobogan District. The methods used in the prediction were Neural Network, Deep Learning, and k-Nearest Neighbor. Based on the prediction results, it can be concluded that k-Nearest Neighbor produced the best predictions. The prediction results from the Deep Learning and Neural Network method had been able to follow the actual target path, but it was still not optimum. K-Nearest Neighbor was not model-based, it calculated the distance and compared every training sample to give the prediction, the data used in this study was simple and did not need much observation, so the
predictions produced by k-Nearest Neighbor was good. The prediction of clean water production using k-Nearest Neighbor can help PDAM determine the policy regarding water production.

Acknowledgment
This work was supported by Research Funding Diponegoro University 2017, Faculty of Sciences and Mathematics.

References

[1] Agus D and Hazami A 2015 “PDAM Grobogan Tak Berdaya Tangani Pencurian Air,” Koran Grobogan, Mei.
[2] Ajbar A and Ali E 2013 Journal of King Saud University - Engineering Sciences 27 83.
[3] Zhao Y, Zou Z and Wang S 2015 Back Propagation Neural Network Model Based on Kalman Filter for Water Quality Prediction in 11th International Conference on Natural Computation.
[4] Olayowin R 2016 “Chemosphere 161 145.
[5] Fan C, Xiao F and Zhao Y 2017 Applied Energy 195 222.
[6] Mocanu E, Nguyen P, Gibescu M and Kling W 2016 Sustainable Energy, Grids and Networks 6 91
[7] Wahid F and Kim D 2016 International Journal of Smart Home 10 97
[8] Szoplik J 2015 Energy 85 208
[9] Yadav A and Sahu K, International Journal of Research in Science & Engineering, 3 148
[10] Lillicrap T, Cownden D, Tweed D and Akerman C 2016 Nature Communication 7 13276.
[11] Traore S, Yufeng L, Guy F, 2016 Agricultural Water Management 163 363.