공학석사 학위논문

Prediction of Water Quality Parameters at the Confluence of Nakdong and Kumho Rivers using Artificial Neural Network Model

인공신경망 모델을 이용한 낙동강과 금호강 합류부 수질인자예측

2018년 2월

서울대학교 대학원

건설환경공학부

윤 세 훈

Abstract

Prediction of Water Quality Parameters at the Confluence of Nakdong and Kumho Rivers using Artificial Neural Network Model

The pollutants in the tributaries with sufficient flow have a great influence on the water quality of the river after confluence. Stream confluences are elements of river networks that play a major role in the dynamics of fluvial systems. Substantial changes in flow hydrodynamics generally occur immediately downstream of confluence, and mixing of tributary flows may extend many kilometers downstream of confluence. Up-to-data investigations of the flow physics at river confluence rely primarily on physically-based numerical modeling. However, verification through field experiments is essential when a physics-based numerical model is applied to analyze the behavior of contaminants flowing through the tributaries downstream of the confluence. In addition, time and money should be invested in model construction and operation. However, in the case of data-based model, it is possible to make predictions with only accumulated data. Among the data-driven model, the ANN model is often used for application of water quality prediction. Many researchers used ANN technique to predict water quality parameters in river systems.

In this study, the ANN ensemble model with resilient propagation method was developed to predict the water quality parameters at the confluence of Nakdong and Kumho rivers. The data of EC tracing data conducted in 2015

were used to accurately understand the behavior of contaminants after confluence. The best fitted prediction results is shown when using water quality values of mainstream and tributary both as the input data (0.56 R^2 value of pH, 0.75 of DO, 0.80 of EC, 0.66 of Chl-a). The point where the best fitted prediction results shown is ARCWQ-2 and ARCWQ-3 (where the transverse mixing was completed). And the improvement rate was also the largest at the same case and point for pH, EC, and EC(22% of pH, 77% of DO, 26% of EC and 19% of Chl-a).

Keyword : Water quality prediction, Tributary confluence, ANN ensemble model, EC tracing test, Transverse mixing

Student number : 2015-22927

Table of Contents

List of Figures

List of Tables

List of Symbols

Latin Uppercase

E expected value(mean value)

E_r back-propagation error of the neurons in the output layer

K number of unknowns

N number of observed data set

N_H number of neurons in hidden layer

N_I number of neurons in input layer

N_0 number of neurons in output layer

$Q_{25}(y_i)$ 25th percentile values of the ANN ensemble model result for the ith data set.

$Q_{75}(y_i)$ 75th percentile values of the ANN ensemble model result for the ith data set

SS_{tss} the total sum of squared error between observations and mean observations

SS_{rss} the sum of squares of residuals between observations and outputs

Latin Lowercase

h number of the iteration, epoch

i index of neurons in input layer

j index of neurons in hidden layer

o_i ith observed value

t target output value

w weight vector

w^{I-H} weight value between input vectors and neurons in hidden layer

w^{H-0} weight of neurons between hidden layer and output layer

x_i ith observed value or target value.

y_i ith output; a result of an individual ANN model for the ith data set

$\overline{y_i}$ mean result of an ANN ensemble model for the ith data set

Greek Uppercase

Σ covariance matrix

Greek Lowercase

φ activation function

η learning rate

μ damping factor

θ output of a neuron

Chapter 1. Introduction

1.1 Background and Necessity

The pollutants in the tributaries with sufficient flow have a great influence on the water quality of the large river after confluence. However, verification through field experiments is essential when a physics-based numerical model is applied to analyze the behavior of contaminants flowing through the tributaries downstream of the confluence. In addition, time and money should be invested in model construction and operation.

However, in the case of data-based model, it is possible to make predictions with only accumulated data. Among the data-driven model, the ANN model is often used for application of water quality prediction. Many researchers used ANN technique to predict water quality parameters in river systems. Warren McCulloch and Walter Pitts (1943) created a computational model for neural networks based on mathematics and algorithms called threshold logic. ANNs have been widely used for solving a range of hydrological problems such as rainfall-runoff modeling, streamflow forecasting, water quality modeling, groundwater modeling, hydrological time series modeling, and reservoir operation [Task Committee, 2000].

Nevertheless, one of the reasons for avoiding the ANN model is that the ANN model is an opaque black box model. In fact, the methodological research for the development of the ANN model is less than application one. methods for developing ANN model are not yet well established. The methodological issues that have been assessed

traditionally are the proper selection and preprocessing of the inputs and outputs, and the choice of the architecture of the neural networks (Coulibaly et al., 1999; Dawson and wilby, 2001).

Another reason for avoiding the ANN model is that additional knowledge is needed for accurate prediction. As mentioned before, the advantage of a data-driven model is that it is easy and useful for people who do not have sufficient knowledge of basic physical processes. But additional statistical knowledge is essential to improve the accuracy of the model.

Previous studies used statistical analysis and factor analysis of the input factors to improve the accuracy of the ANN model. Kim & Seo (2015) used clustering method for water quality prediction in rivers. However, there are not enough studies to build an artificial neural network model using the input parameters based on the physical process and improve the accuracy.

1.2 Objectives

The purpose of this study is to predict the water quality at the tributary confluence by using a robust and accurate artificial neural network model. To achieve this goal,

Verify the behavior of water quality variables, analyzing the water qulity data after confluence.

(1) The characteristics of water quality data of mainstream and tributaries obtained from automatic water quality monitoring station.
(2) The spatial and temporal distribution of water quality data obtained from fixed quality monitoring sensors.
(3) The mixing patterns of the study site using the data of conductivity tracing experiment in natural streams.

Develop and verify the ANN model which predicts accurate and robust results.

1) development of ANN model using Resilient Backpropagation algorithm.
2) Enhencement of prediction accuracy using sampling method and Ensemble method
3) Comparing the results of developed ANN model with generally used ANN model and AR(2) model.

Predict the water quality at the tributary confluence using developed ANN model.

1) Adding the water quality data of tributary as a input data when predict water quality data of mainstream.
2) Analysis of water quality prediction results at the downstream of confluence.
3) comparing with the spatial distribution characteristics revealed in the field experiment.

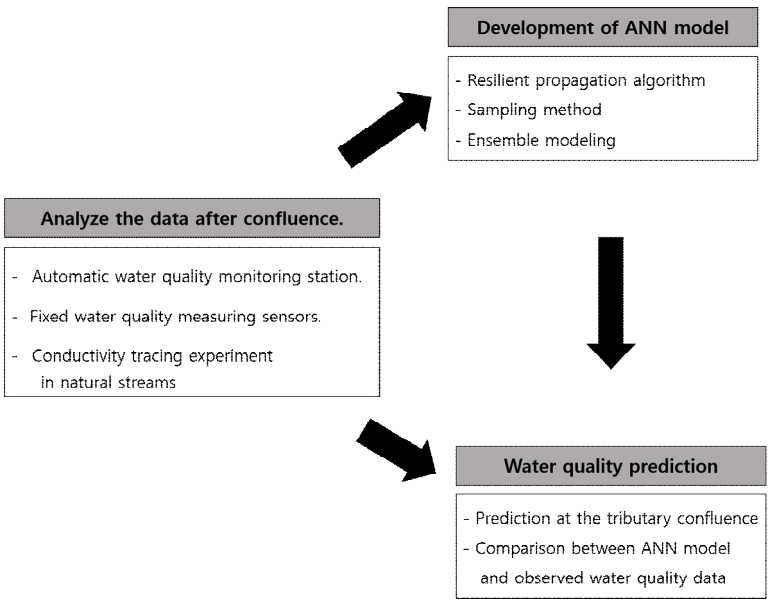


Figure 1. Thesis overview

Chapter 2. Theoretical Backgrounds

2.1 Data-driven Models

There are basically two approaches for hydrological modelling: the theory-driven (conceptual and physically-based) approach, and the data-driven (empirical and black-box) approach often associated by practitioners with statistical modelling. Conceptual models represent the general internal sub-processes and physical mechanisms without looking at the spatial variability and stochastic properties.

Physically-based models are based on the understanding of the underlying physical behaviour of the system. Typically, they involve the solution of a system of partial differential equations that represent our best understanding of the flow processes in the watershed.

In contrast to the two class of models above, data-driven (black-box) models, as the name suggests, are base on analysing the data about a system without explicit knowledge of the physical behaviour of the system. When a considerable amount of data describing this problem is available, data-driven models has been successfully applied to modeling a particular system or process. Traditionally, the data-driven models borrow the techniques developed in such (overlapping) areas as statistics, soft computing, computational intelligence, machine learning and data mining. As explosively growing, widely available, and gigantic body of data makes a large collection of data into knowledge. So it is very effective for researchers who deals with the nonlinear nature of the relationship and the complexity of the physically based models.

2.2 Artificial Neural Networks(ANNs)

Among many data-driven techniques, the artificial neural network (ANN) is the most widely used and become a new tool and an efficient model for the prediction and forecasting of variables in river systems, due to the inherent uncertainties of contaminant source and water quality data have. The concept of ANNs is inspired by the biological neural networks of the human brain. McCulloch and Pitts(1943) first created a computational model for neural networks based on mathematics and algorithms called threshold logic. Then a key advance that came later was the backpropagation algorithm which effectively solved the exclusive-or problem, and more generally the problem of quickly training multi-layer neural networks (Werbos 1975).

2.2.1 Architecture of ANNS

The basic structure of an ANN model is usually comprised of three distinctive layers. This configuration is also referred to as a multilayer perceptron(MLP) and it represents one of the most commonly used neural networks.

Data enter the network through the input layer, where computation of the weighted sum of the input is performed. These data are then fed forward through successive layers including the hidden layer or layers, where data are processed. The hidden layer is the essential component that allows the neural network to learn the relationships between input and output data. And the output layer, where the results of ANN are produced.

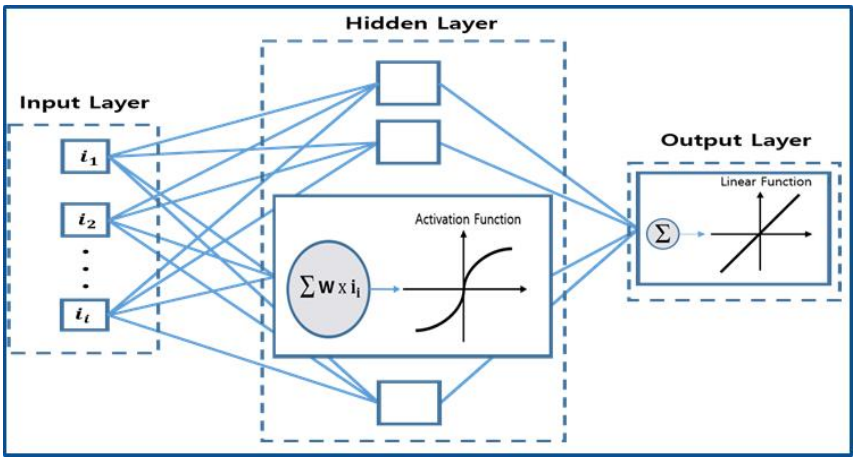


Figure 2. Schematic of a typical ANN models

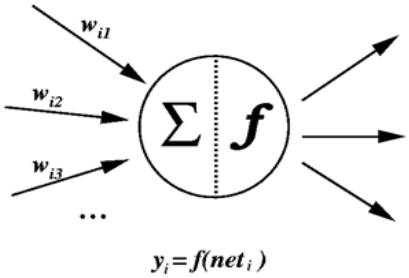
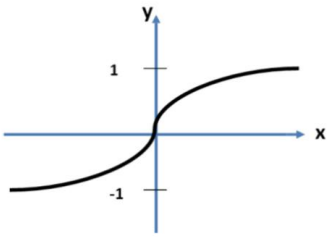


Figure 3. Schematic of Neuron



$$y = tansig(x) = \frac{2}{1 + e^{-2x}} - 1$$

Figure 4. Tangent Sigmoid Function

Each layer consists of one or more basic element(s) called a neuron or a node. Neurons are fundamental to the operation of a neural network. It is a highly interconnected processor and exchange messages between each other. Each neuron consists of a weight parameter and an activation function. When the input data is passed to the input layer, it is binding with the weight and was granted a non-linearity by activation function. The process of being transmitted to the next neuron is repeated until derive a final result. The inter-connection between neurons is accomplished by using known inputs and outputs, and presenting these to the ANN in some ordered manner. This process is called training. The strength of these interconnections is adjusted using an error convergence technique so that a desired output will be produced for a known pattern. The number of neuron(s) has been selected by a trial & error procedure.

As you can identify in figure 3., neuron has the three basic elements. The first element is synapse weight w. Synaptic weight refers to the strength or amplitude of a connection between two nodes, corresponding in biology to the amount of influence the firing of one neuron has on another. The synaptic weight is changed by using a learning rule, if a large signal from one of the input neurons results in a large signal from one of the output neurons, then the synaptic weight between those two neurons will increase. The rule is unstable, however, and is typically modified using the backpropagation algorithm. The second element is an adder. An adder sum the input signals and weight by the respective synapses of the neuron. The third element is an activation function. If there were no activation

functions, the whole neural network could be reduced to a group of linear function of the network input – one linear function for each output neuron. So, without activation functions, a neural network could not learn non-linear relationships. Multilayer networks typically use sigmoid transfer functions in the hidden layers. These functions are often called "squashing" functions, because they compress an infinite input range into a finite output range. Particularly tangent sigmoid activation function is commonly used in multilayer neural networks that are trained using the backpropagation algorithm, in part because this function if differentiable (Hagan, 1994).

There are two classes of ANN network architectures. The simplest kind of neural network is a single-layer perceptron network, which consists of a single layer of output nodes; the inputs are fed directly to the outputs via a series of weights. In this way it can be considered the simplest kind of feed-forward network. The term perceptron often refers to networks consisting of just one of these units. Perceptrons can be trained by a simple learning algorithm that is usually called the delta rule. It calculates the errors between calculated output and sample output data, and uses this to create an adjustment to the weights. Single-unit perceptrons are only capable of learning linearly separable problems. The second class of networks is multiple layer feedforward neural network. It consists of an input layer, one or more hidden layers, and an output layer. An example of a multilayer feedforward network is shown in figure 4. The input data pass through the input layer and are then weighted and fed simultaneously to a second layer of neuron, known as a hidden layer. The outputs of the hidden layer

neuron can be inputted to another hidden layer, and so on. The number of hidden layers is arbitrary, although in practice, usually only one is used. The weighted outputs of the last hidden layer are input to neuron making up the output layer, which emits the network's prediction for given tuples. It is a feed-forward network since none of the weights cycles back to an input unit or to a previous layer's output unit. It applies a nonlinear activation function to the weighted input. Multilayer feed- forward neural networks are able to model the class prediction as a nonlinear combination of the inputs.

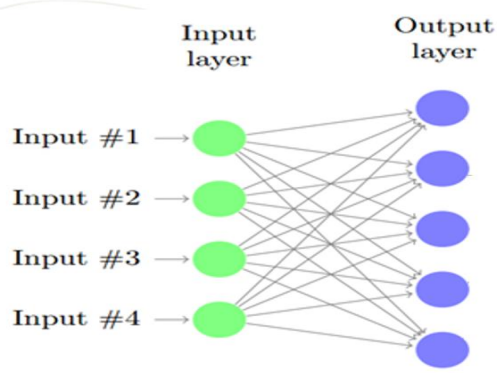


Figure 5. single layer feed forward networks

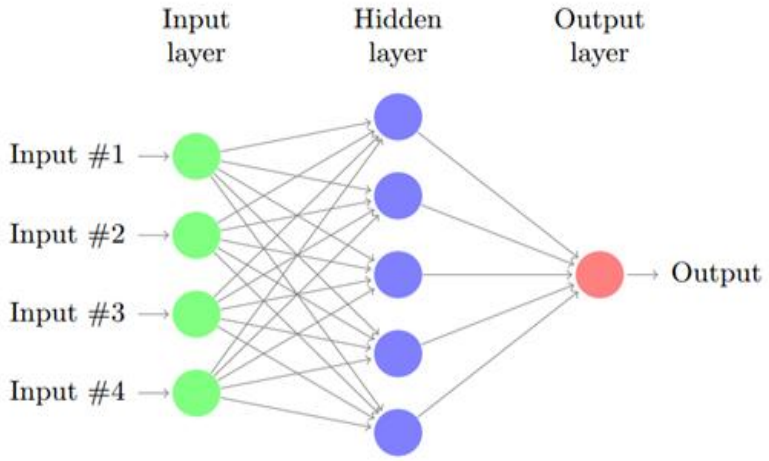


Figure 6. Multilayer feedforward neural network

2.2.2 Training Algorithms

The property that is of primary significance for a neural network is the ability of the network to learn from its environment, and to improve its performance through learning. Depending on the manner in which the neural networks are learning, the networks are divided into supervised learning network and unsupervised learning network, which are also referred to as learning with a teacher and learning without a teacher respectively. In conceptual, we may think of the teacher as having knowledge of the environment represented by a set of input-output examples. The environment is, however, unknown to the neural network of interest. Suppose that the teacher and the neural network are both exposed to a training vector (examples) drawn from the environment. By virtue of built-in knowledge, the teacher is able to provide the neural network with a desired response for that training vector. Indeed, the desired response represents the optimum action to be performed by the neural network. The network parameters are adjusted under the combined influence of the training vector and the error signal. The error signal is defined as the difference between the desired response and the actual response of the network. This adjustment is carried out iteratively in a step-by-step fashion with the aim of eventually making the neural network emulate the teacher, the emulation is presumed to be optimum in some statistical sense. In this way knowledge of the environment available to the teacher is transferred to the neural network through training as fully as possible. When this condition is reached, we may then dispense with the teacher and let the neural network deal with the environment completely by itself (Haykin,

2004).

In Supervised Mutilayer Feedforward Neural Network (SMFNN) model, input vectors and the corresponding target vectors are used to train the network until it can approximate a function, associate input vectors with specific output vectors, or classify input vectors in an appropriate way. The most common and standard algorithm for training the neural network is the backpropagation of error, of which the central idea is that the error for the neurons of the hidden layer are determined by back-propagating the error of the neurons of the output layer. Standard backpropagation algorithm has two processes which are feedforward operation and learning.

In feedforward operation input vectors or patterns are presented in input layer and each neuron in hidden layer is calculated into the activation of a single neuron net_j, which is a dot product of input vectors and weight value of neurons in hidden layer.

$$\mathrm{net}_j = \sum_{i=1}^{N_I} x_i w_{ij}^{I-H} + b_{1j} = \sum_{i=0}^{N_I} x_i w_{ij}^{I-H} \qquad (2.1)$$

where N_I is a dimension of input vector. i and j are the index of neurons in input layer and hidden layer respectively. w_{ij}^{I-H} are the weight value between input vectors and neurons in hidden layer. b_{1j} is the weight value of bias in hidden layer which is usually assumed as $b_{1j} = w_{0j}^{I-O}$, $x_0 = 1$.

Substituting net_j into activation function φ_1, θ_j is calculated.

$$\theta_j = \varphi_1(\mathrm{net}_j) \qquad (2.2)$$

Each neuron in output layer is calculated into the activation of a single neuron net_k, which is a dot product of θ_j and weight value of neurons in output layer.

$$\mathrm{net}_k = \sum_{j=1}^{N_H} \theta_j w_{jk}^{H-0} + b_{2k} = \sum_{j=0}^{N_H} \theta_j w_{jk}^{H-0} \tag{2.3}$$

where N_H is a number of neurons in hidden layer and k is the index of neurons in output layer. w_{jk}^{H-0} are the weight value of neurons between hidden layer and output layer.

Substituting net_k into activation function φ_2, network output y_k is calculated.

$$y_k = \varphi_2(\mathrm{net}_k) = \varphi_2(\sum_{j=0}^{N_H} \theta_j w_{jk}^{H-0}) = \varphi_2(\sum_{j=0}^{N_H} w_{jk}^{H-0}(\sum_{i=0}^{N_I} x_i w_{ij}^{I-H})) \tag{2.4}$$

In learning process a set of weight is updated to ensure that the network output y_k are close to the target output value t_k by back-propagating the error E_r of the neurons of the output layer. While there are a number of performance functions assessing the error, it is typical to use the squared error.

$$E_r = \sum_{k=1}^{N_0} (t_k - y_k)^2 \tag{2.5}$$

where N_0 is a number of neurons in output layer.

Standard backpropagation is based on gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the performance function. Weight adjustments are made by

multiplying the error gradient with respect to the weight by a learning rate η.

$$\triangle w = -\eta \frac{\partial E_r}{\partial w} \tag{2.6}$$

Weight value of neurons between hidden layer and output layer is updated, which is then used to update the weight value of neurons between input layer and hidden layer. Updating the weight value of neurons between hidden layer and output layer is defined as

$$\frac{\partial E_r}{\partial w_{jk}^{H-0}} = \frac{\partial E_r}{\partial net_k} \cdot \frac{\partial net_k}{\partial w_{jk}^{H-0}} = \frac{\partial E_r}{\partial y_k} \cdot \frac{\partial y_k}{\partial net_k} \cdot \frac{\partial net_k}{\partial w_{jk}^{H-0}} = -(t_k - y_k) \cdot \varphi^{'2}(net_k) \cdot \theta_j \tag{2.7}$$

$$\triangle w_{jk}^{H-0} = -\eta \frac{\partial E_r}{\partial w_{jk}^{H-0}} = \eta (t_k - y_k) \cdot \varphi^{'2}(net_k) \cdot \theta_j \tag{2.8}$$

Updating the weight value of neurons between input layer and hidden layer is defined as

$$\begin{aligned} \frac{\partial E_r}{\partial w_{jk}^{I-H}} &= \frac{\partial E_r}{\partial \theta_j} \cdot \frac{\partial \theta_j}{\partial net_k} \cdot \frac{\partial net_k}{\partial w_{ij}^{I-H}} \\ &= \frac{\partial}{\partial \theta_j} \left[\frac{1}{2} \sum_{k=1}^{N_0} (t_k - y_k)^2 \right] \cdot \frac{\partial \theta_j}{\partial net_k} \cdot \frac{\partial net_j}{\partial w_{ij}^{I-H}} \\ &= -\sum_{k=1}^{N_0} \left[(t_k - y_k) \cdot \frac{\partial y_k}{\partial \theta_j} \right] \cdot \frac{\partial \theta_j}{\partial net_j} \cdot \frac{\partial net_j}{\partial w_{ij}^{I-H}} \\ &= -\sum_{k=1}^{N_0} \left[(t_k - y_k) \cdot \frac{\partial y_k}{\partial net_k} \frac{\partial net_k}{\partial \theta_j} \right] \cdot \frac{\partial \theta_j}{\partial net_j} \cdot \frac{\partial net_j}{\partial w_{ij}^{I-H}} \end{aligned} \tag{2.9}$$

$$= - \sum_{k=1}^{N_0} \left[(t_k - y_k) \cdot \varphi_2^{'}(net_k) \cdot w_{jk}^{H-0} \right] \cdot \varphi_1^{'}(net_j) \cdot x_i$$

$$\triangle w_{ij}^{I-H} = - \eta \frac{\partial E_r}{\partial w_{ij}^{I-H}} = \eta \cdot \sum_{k=1}^{N_0} \left[(t_k - y_k) \cdot \varphi_2^{'}(net_k) \cdot w_{jk}^{H-0} \right] \cdot \varphi_1^{'}(net_j) \cdot x_i \qquad (2.10)$$

With backpropagation, the input data is repeatedly presented to the neural network. With each presentation the output of the neural network is compared to the desired output and an error is computed. This error is then backpropagated to the neural network and used to adjust the weights as

In Eq. (2.11) such that the error decreases with each iteration and the neural network gets closer and closer to producing the desired output.

$$w(h+1) = w(h) + \triangle w(h) \qquad (2.11)$$

where h is number of iteration, epoch.

There are a number of variations in backpropagation training algorithms that are based on other standard optimization techniques, such as conjugate gradient and Newton's method. The steepest descent algorithm is the simplest and often the slowest. The conjugate gradient algorithm and Newton's method generally provide faster convergence.

2.3 Confluence Mixing

Dispersion characteristics of pollutants must be well understood to deal with water pollution problems in streams and rivers. Physically, dispersion, which is the spreading out of pollutants across and along streams, arises due to the combined effect of the differences in velocity and turbulent diffusion.

In natural streams, stream width is much greater than depth. So transverse dispersion is more important than vertical mixing, especially when dealing with the mixing of pollutants in meandering streams.

Some of researchers have conducted a large number of experiments on transverse mixing, both in straight rectangular laboratory channels and in natural or irregular channels (Elder 1959; Miller and Richardson 1974; Holly 1975; Lau and Krishnappan 1977; Webel and Schatzmann 1984; Nokes and Wood 1988).

However, because natural streams are rarely straight and prismatic for considerable distances, the hydrodynamic effects induced by geometric non-uniformity and channel meandering should be taken into account in the analysis of pollutant mixing (Baek et al. 2006).

Natural channels differ from uniform rectangular ones in three important respects : the depth may vary irregularly, the channel is likely to curve, and there may be large sidewall irregularities such as groins or points of land. None of these factors are thought to have much influence on the rate of vertical mixing. The rate of transverse mixing is strongly affected by the channel irregularities because they are capable of generating a wide variety of transverse motions. The study of Holley et al. (1972) showed the effect of a cross-sectional

depth variation in a straight trapezoidal canal. It showed that the concentration distribution resulting from a side injection in a trapezoidal channel differs from that in a rectangular channel, the details depending on what cross-sectional variations of velocity and mixing coefficient are assumed. Bends and sidewall irregularities are common to many channels and have a major effect on transverse mixing. Holley and Abraham's (1973a) laboratory studies of groins gave some indication of the effect of sidewall irregularities.

If a pollutant is discharged at the side of a channel, the width over which mixing must take place is twice that for a centerline injection, but the boundary conditions are otherwise identical. The case of side injection (neglecting effects of momentum and buoyancy of the discharge) is similar to just one side of the centerline injection. For a given rate of injection of mass the concentration in the side injection is twice that of the centerline injection, but the rate of increase of the variance will be the same. This can be seen in the experimental results of a study in an irrigation canal near Albuquerque, New Mexico, by Fischer (1967b).

Study site of this study is in case of mixing of two streams which flow together at a smooth junction, and whose density is nearly the same so that the streams flow side by side until turbulence accomplishes the mixing.

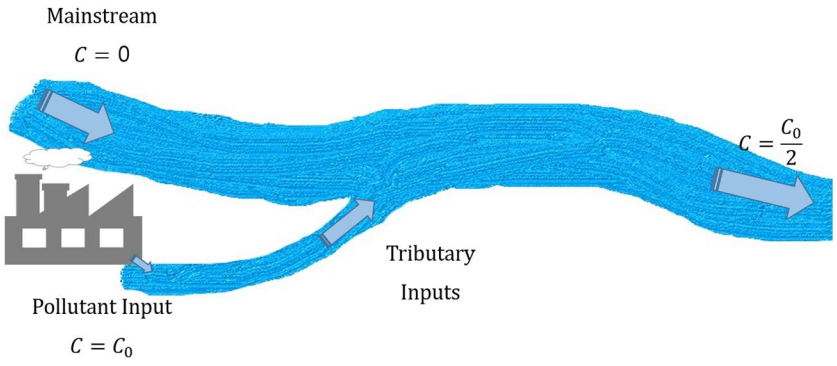


Figure 7. Confluence mixing

Chapter 3. Data Collection

3.1 Study Site and Available Data

3.1.1 Study site

For the application of this study, several conditions must be met. First, discharge of tributary should be sufficient. Second, the pollution constantly flows in. And last, daily water quality data should be available at confluence of the tributary stream.

In this study, the reach between upstream and downstream reaches of the Gangjeong-Goryeong Weir to Dalsung Weir of the Nakdong River was selected as the study site. In this area, Kumho River, the second largest tributary of Nakdong River, flows into the left bank of Nakdong River. Contaminants from the Daegu, one of the major cities in South Korea, are always introduced through the Kumho River. And there are 2 automatic water quality monitoring stations(Dasan, Gangchang) and 2 water discharge monitoring station(Seongju, Sungsu) before confluence and three fixed water quality measuring sensors and 1 automatic water quality monitoring stations(Goryeong) are located downstream of the confluence.

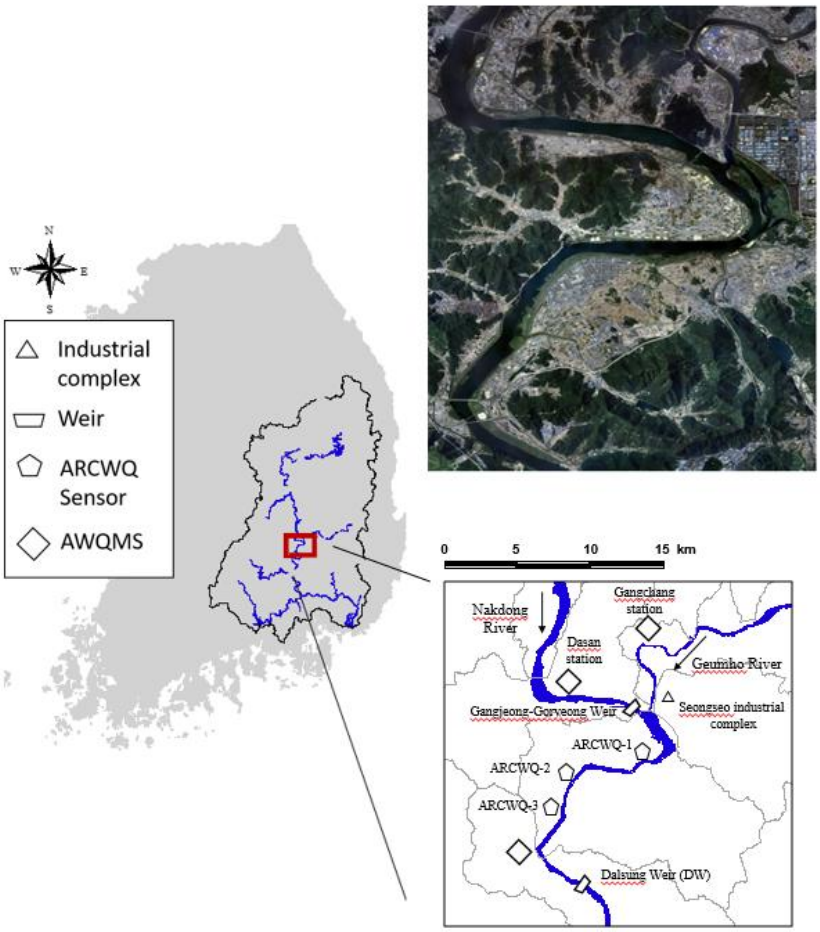


Figure 8. Study site (adoped from Park et al.,2017)

3.1.2 Monitoring Method

(1) Tracing test using EC

EC(Electric conductivity) is a measure of the ability of water to pass an electrical current. Conductivity in water is affected by the presence of inorganic dissolved solids such as chloride, nitrate, sulfate, and phosphate anions (ions that carry a negative charge) or sodium, magnesium, calcium, iron, and aluminum cations (ions that carry a positive charge). Organic compounds like oil, phenol, alcohol, and sugar do not conduct electrical current very well and therefore have a low conductivity when in water. Conductivity is also affected by temperature: the warmer the water, the higher the conductivity. For this reason, conductivity is reported as conductivity at 25 degrees Celsius (25 C). The basic unit of measurement of conductivity is the mho or siemens. Conductivity is measured in micromhos per centimeter (μmhos/cm) or microsiemens per centimeter (μs/cm). Distilled water has a conductivity in the range of 0.5 to 3 μmhos/cm. The conductivity of rivers in the United States generally ranges from 50 to 1500 μmhos/cm. Studies of inland fresh waters indicate that streams supporting good mixed fisheries have a range between 150 and 500 μhos/cm.

A YSI-6600OMS sensor was used to measure EC concentration. The sonde of the YSI-6600OMS utilize a cell with four pure nickel electrodes for the measurement of conductance. Two of the electrodes are current driven, and two are used to measure the voltage drop. The measured voltage drop is then converted into a conductance value in milli-Siemens per cm (mS/cm). The measurement range is 0~100000 μs/cm, the resolution is 1μs/cm, and the accuracy is 0.5%.

The EC tracer test was conducted at the confluence of the Nakdong and Kumho rivers located downstream of Gangjeong-Goryeong Weir in July 2015 on the study sites (National Institute of Environmental Research, 2015). As shown in Figure 1, the total length was 10km of mainstream. A YSI-6600OMS sensor was used to measure the transverse mixing of EC concentration flowing into the mainstream through the tributary. It was measured continuously at a total of 8 sections from Sec.-1 to Sec.7 including upstream and downstream of the confluence point (Sec.-1, Sec.1) in the transverse direction.

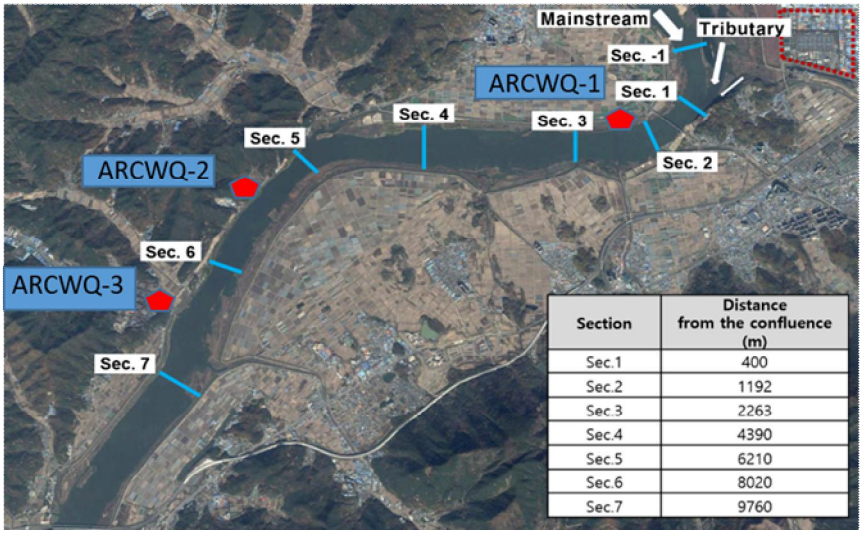


Figure 9. Site map of tracing test using EC

Table 1. Hydraulic data for Nakdong River

Case	Date	Q (m^3/s)	H (m)	U (m/s)	W (m)
ND-EC21	2015.7.21 ~ 2015.7.22	130.1	5.40	0.086	384.8

(2) Automatic water quality monitoring station

Over the four major rivers, a total of 70 automatic water quality monitoring stations are installed and operated by Ministry of Environment. Among them, 24 stations are located in the Nakdong River basin. At the automatic water quality monitoring station, the five factors (water temperature, pH, DO, electric conductivity, and total organic carbon) are essentially measured. And factors such as turbidity, TN, TP, Chl-a are selectively collected. The data obtained from the automatic water quality monitoring station are from January 2012 to June 2017.

The process of the automatic water quality monitoring station is shown in Figure 10. and detailed specifications of are shown in Table 3.

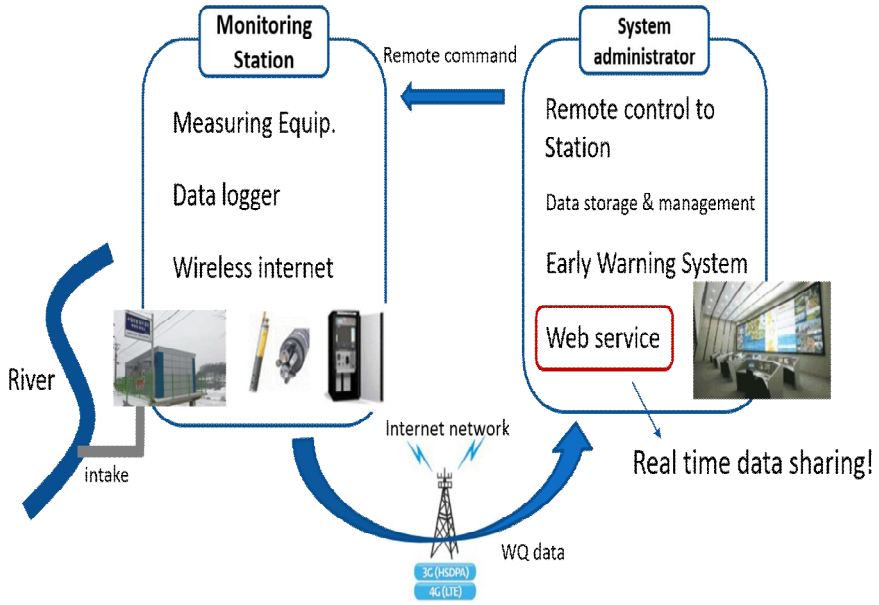


Figure 10. Process of automatic water quality monitoring station

a) Dasan automatic water quality monitoring station.

b) Gangchang automatic water quality monitoring station.

c) Goryeong automatic water quality monitoring station.

Figure 11. A view of automatic water quality monitoring station

Table 2. Discharge data of study site

Station	Location	Year	Discharge (m^3/s)			
			Drought Period	Low Period	Normal Period	Flood Period
Seongju (Nakdong River)	Seongju Bridge	2014	70.75	99.87	156.54	725.03
		2015	49.72	67.75	89.78	254.64
		2016	27.12	68.64	138.13	834.15
Seongseo (Gumho River)	Gang-chang Bridge	2014	15.71	21.98	33.97	200.21
		2015	13.74	18.64	27.17	74.24
		2016	14.76	21.37	35.96	182.1

Table 3. Water quality data of automatic water quality monitoring station

Station	Location	Period	Variables
Dasan	Nogok-ri 1467-1	2012.01 ~ 2017.06	WT, pH, EC, DO, Chl-a, Tubidity, TN, TP, TOC
Gangchang	Haryang Bridge	2012.01 ~ 2017.06	WT, pH, EC, DO, Chl-a, TOC
Goryeong	Goryeong Bridge	2012.01 ~ 2017.06	WT, pH, EC, DO, Chl-a, Tubidity, TN,, TP, TOC

(3) Fixed water quality measuring sensor

For the fixed water quality measuring sensors, YSI-6600 V2, YSI 6600EDS V2 are used. These devices are designed to be easy to use in the field and can measure the water temperature, electric conductivity basically and DO, pH, Chl-a, turbidity and BGA by installing additional probe sensor. The measured water quality variable data were saved every 5 minutes, and the sensor was calibrated and backed up once a every month.

The data from ARCWQ-1 are from January 2014 to April 2017. The data from ARCWQ-2 are from June 2013 to April 2017. And the data from ARCWQ-3 are from October 2015 to April 2017.

The season where the missing data occurred most frequently due to the problem of operating equipment was summer and the season where data exceeding the measurement range of the sensor occurred most was summer and fall.

(a) YSI-6600 V2 (a) YSI-6600EDS V2

Figure 12. YSI-6600 V2 and YSI-6600EDS V2

a) ARCWQ-1

b) ARCWQ-2

c) ARCWQ-3

Figure 13. A view of fixed water quality measuring sensors

Table 4. Water quality data of fixed water quality measuring sensors

Station	Location	Period	Variables
ARCWQ-1	Samunjin Ferry	2013.06 ~ 2015.10	WT, pH, EC, DO, Chl-a, Tubidity
		2015.10 ~ 2017.05	WT, pH, EC, DO, BGA
ARCWQ-2	Wolseong Pumping station	2014.04 ~ 2015.10	WT, pH, EC, DO, Chl-a, Tubidity
		2015.10 ~ 2017.05	WT, pH, EC, DO, Chl-a, BGA, Tubidity
ARCWQ-3	Songgok Pumping station	2015.11 ~ 2017.05	WT, pH, EC, DO, Chl-a, BGA, Tubidity

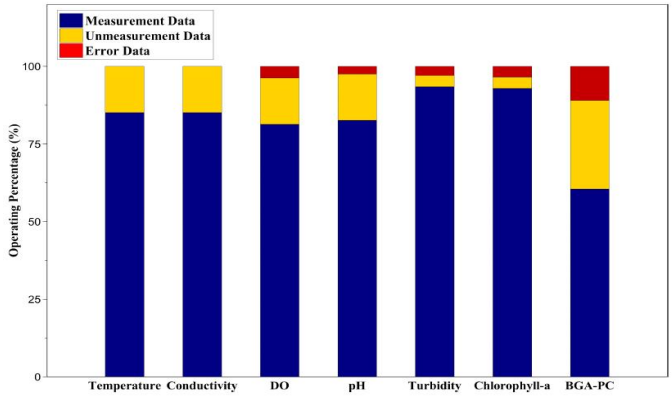


Figure 14. Operation efficiency of ARCWQ-1

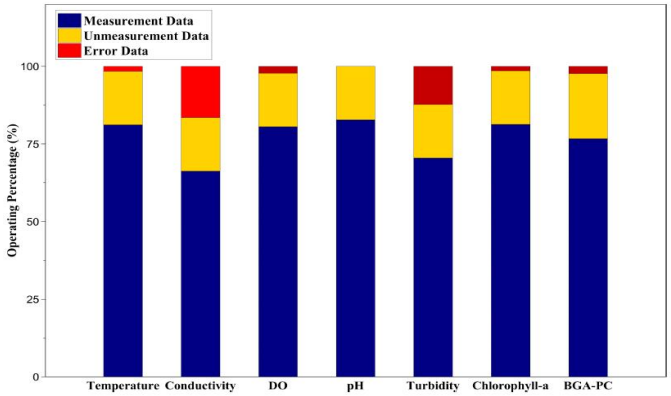


Figure 15. Operation efficiency of ARCWQ-2

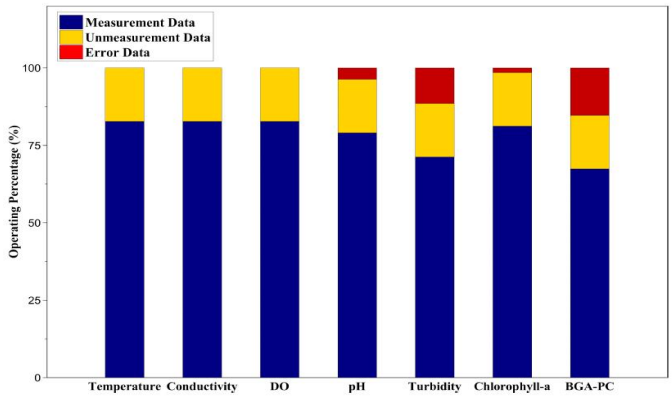


Figure 16. Operation efficiency of ARCWQ-3

3.2 Analysis of Data

In this study, 6 sets of water quality data were available. 3 sets are obtained from the automatic water quality monitoring stations (Dasan, Gangchang and Goryeong station) installed and operated by the Ministry of Environment. And others are obtained from the fixed water quality measuring sensors (ARCWQ-1, ARCWQ-2 and ARCWQ-3) by ARCROM (Advanced Research Center for River Operation and Management) research center. Among the obtained water quality data, 4 variables (pH, EC, DO, Chl-a) are selected for this study.

(1) Data from tracing test using EC

In the field EC tracer experiment conducted in July 2015, the measured EC value at the Sec.-1, upstream of the confluence, is 271 ~280 μS/cm. There was almost no difference in lateral distribution. The EC value of 495 flows through the Kumho River. At Sec.1, downstream of the confluence, the maximum concentration of EC in the left bank of the main stream increased to 430 μS/cm. Afterwards, lateral mixing was gradually increased from Sec.2 toward the flow direction. And it was found that mixing was almost completed in the lateral direction at Sec.6. At Sec.7, the value of EC at the left bank was higher due to the influx of another tributary.

As a result of measurement at Sec.1, the EC concentration was high on the left bank of the mainstream due to contaminants flowed through the tributary flow. At Sec.3, 2km downstream of the confluence, the transverse mixing increased gradually and after Sec.4,

4.5km downstream of the confluence, the mixing was completed and the EC concentration became the same in the left and right bank of mainstream.

Table 5. Results of the tracing test using EC

Section	EC_{max} (μS/cm)	EC_{min} (μS/cm)	△EC (μS/cm)
−1	280	271	9
1	430	256	174
2	318	281	37
3	366	302	64
4	351	340	11
5	364	355	9
6	354	350	4
7	352	330	22

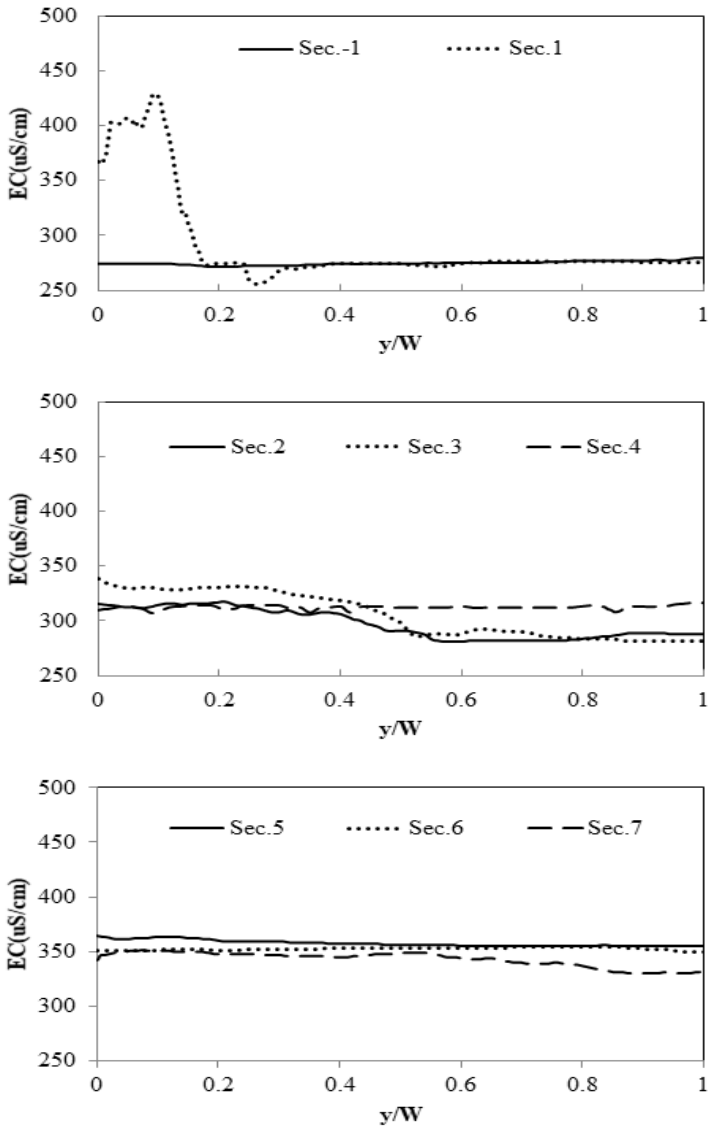


Figure 17. Concentration results of tracing test using EC

(2) Data from the automatic water quality monitoring stations

In case of pH, averaged value of mainstream (Dasan Stations) shows the 8.1 and higher than that of tributary (Gangchang Stations) 7.4. After confluence, pH value shows the 7.8 at Goryeong Stations.

Averaged EC value of mainstream shows the 289μs/Cm and almost 440μs/Cm lower than that of tributary 733μs/Cm. The EC value of mainstream after confluence, shows the 427μs/Cm. The averaged DO value before confluence, shows averaged 9.9mg/L at mainstream and 9.8mg/L at tributary. After confluence, DO value of mainstream shows the 9.2mg/L. The averaged Chl-a value of mainstream shows the 27.5μg/L and that of tributary shows 16.1μg/L before confluence. After confluence, value of mainstream shows 14.7μg/L.

At the tributaries, the averaged pH value are lower and the averaged EC value are higher than that of mainstream. However, as the pollutants flowed with the mainstream, the middle value was appeared at the Goryeong Stations located on about 13km downward from the confluence. In case of DO, the difference between mainstream and tributary was not large and showed similar values. But at the Goryeong Station, the averaged DO value showed the lowest value. The averaged Chl-a value shows 27.5, the high value at the mainstream. Before confluence, 16.1 at the tributary, and lowest value 14.7 at the mainstream after confluence.

Table 6. Statical analysis of data from the automatic water quality monitoring station

Dasan pH	Min	Max	Mean	Standard Deviation
	6.8	9.2	8.17	0.51
Dasan EC	Min	Max	Mean	Standard Deviation
	113	561	288.87	71.28
Dasan DO	Min	Max	Mean	Standard Deviation
	5.0	16.7	10.60	2.16
Dasan Chl-a	Min	Max	Mean	Standard Deviation
	0	310	22.43	22.62
Gangchang pH	Min	Max	Mean	Standard Deviation
	6.6	8.7	7.44	0.31
Gangchang EC	Min	Max	Mean	Standard Deviation
	135	1405	733.09	191.48
Gangchang DO	Min	Max	Mean	Standard Deviation
	1.8	16.3	9.04	2.88
Gangchang Chl-a	Min	Max	Mean	Standard Deviation
	0	139.6	20.80	18.13
Goryeong pH	Min	Max	Mean	Standard Deviation
	6.4	9.2	7.8	0.46
Goryeong EC	Min	Max	Mean	Standard Deviation
	138	802	427.85	119.79
Goryeong DO	Min	Max	Mean	Standard Deviation
	2.8	16.0	9.91	2.65
Goryeong Chl-a	Min	Max	Mean	Standard Deviation
	1.2	152.2	27.49	20.44

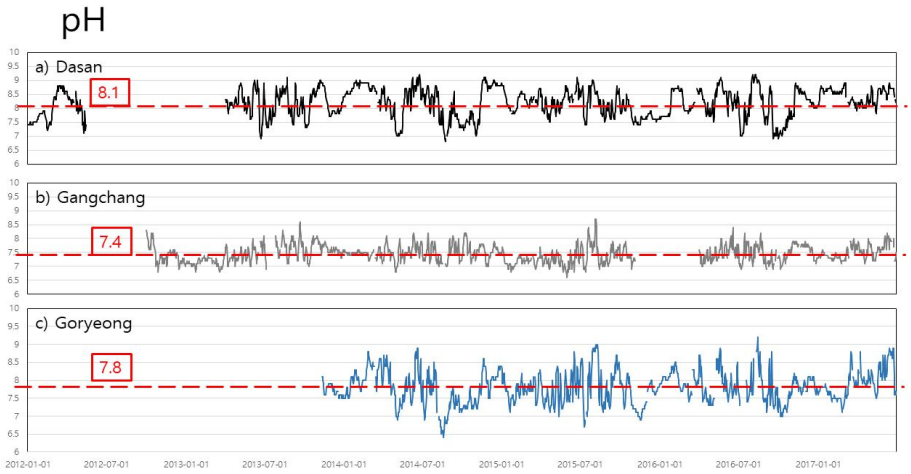


Figure 18. Data from the automatic water quality monitoring stations (pH)

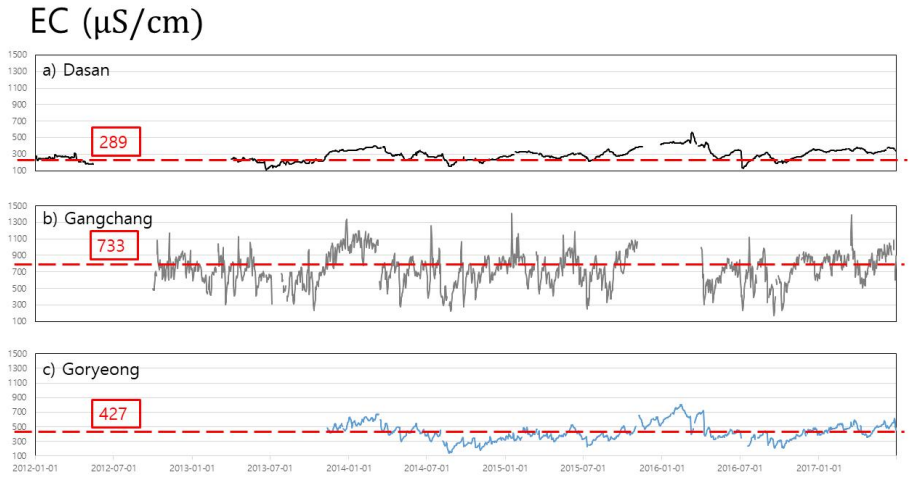


Figure 19. Data from the automatic water quality monitoring stations (EC)

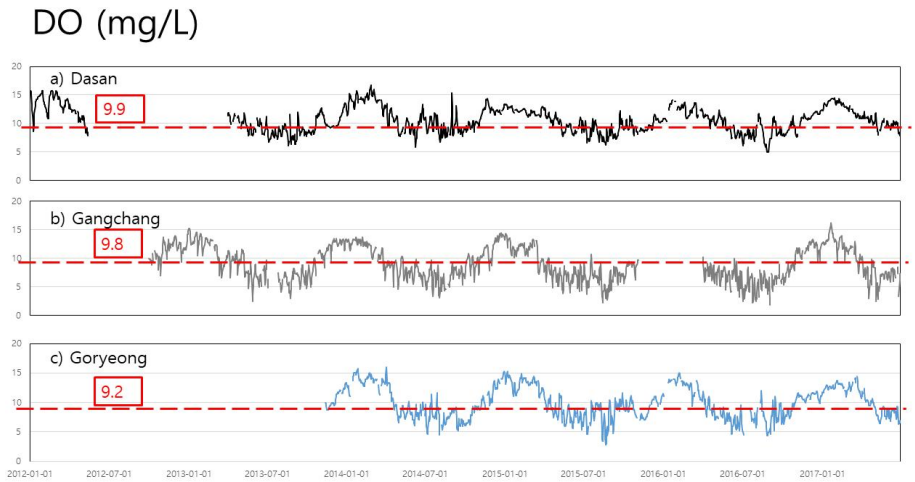


Figure 20. Data from the automatic water quality monitoring stations (DO)

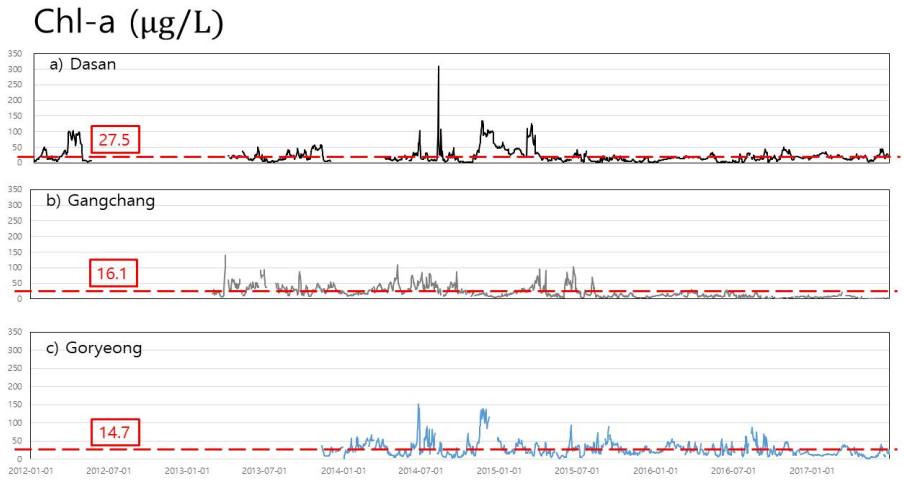


Figure 21. Data from the automatic water quality monitoring stations (Chl-a)

Table 7. Data distribution from the Dasan water quality monitoring station (pH)

	range	# of Data	ratio(%)
Interval 01	6.80 ~ 7.04	30	1.80
Interval 02	7.04 ~ 7.28	42	2.52
Interval 03	7.28 ~ 7.52	158	9.49
Interval 04	7.52 ~ 7.76	158	9.49
Interval 05	7.76 ~ 8.00	272	16.33
Interval 06	8.00 ~ 8.24	202	12.13
Interval 07	8.24 ~ 8.48	206	12.37
Interval 08	8.48 ~ 8.72	374	22.46
Interval 09	8.72 ~ 8.96	173	10.39
Interval 10	8.96 ~ 9.20	50	3.00
Total		1665	100

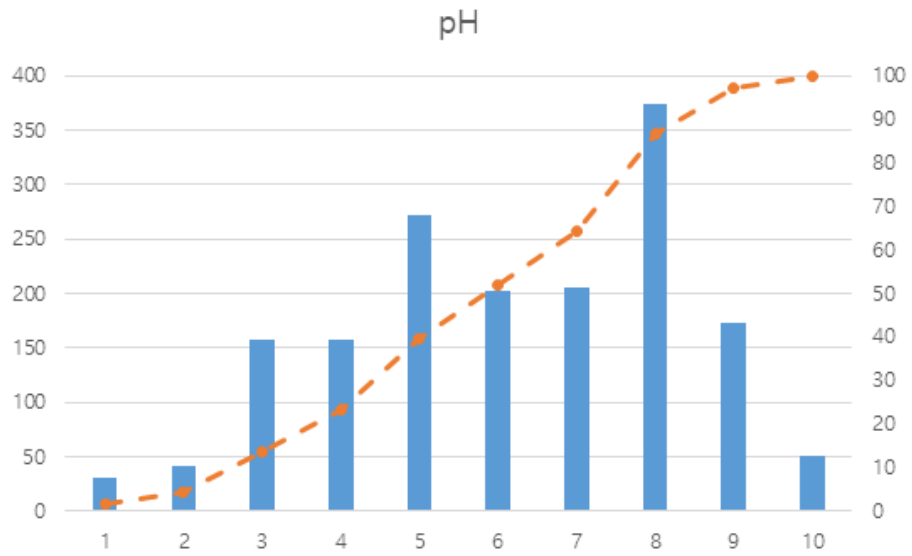


Figure 22. Data distribution from the Dasan water quality monitoring station (pH)

Table 8. Data distribution from the Dasan water quality monitoring station (EC)

	range	# of Data	ratio(%)
Interval 01	113 ~ 158	37	2.29
Interval 02	158 ~ 203	125	7.73
Interval 03	203 ~ 247	343	21.21
Interval 04	247 ~ 292	383	23.69
Interval 05	292 ~ 337	314	19.42
Interval 06	337 ~ 382	263	16.26
Interval 07	382 ~ 427	69	4.27
Interval 08	427 ~ 471	7	4.51
Interval 09	471 ~ 516	4	0.24
Interval 10	516 ~ 561	6	0.37
Total		1617	100

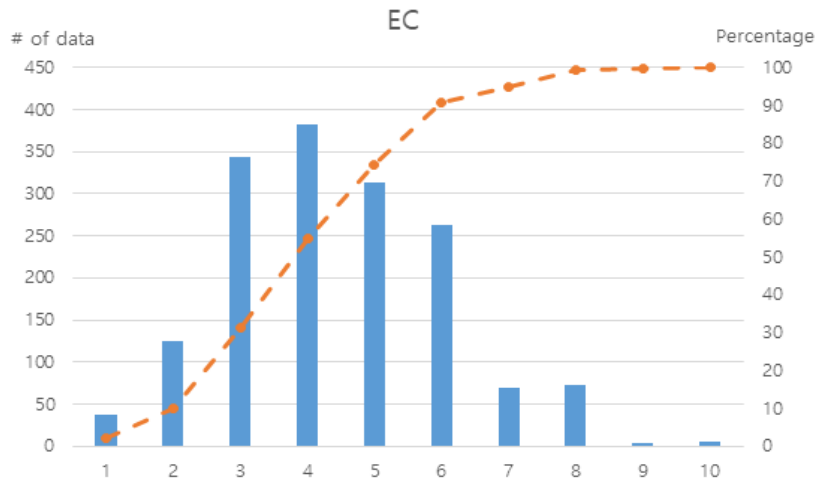


Figure 23. Data distribution from the Dasan water quality monitoring station (EC)

Table 9. Data distribution from the Dasan water quality monitoring station (DO)

	range	# of Data	ratio(%)
Interval 01	5.00 ~ 6.17	11	0.68
Interval 02	6.17 ~ 7.34	54	3.33
Interval 03	7.34 ~ 8.51	246	15.19
Interval 04	8.51 ~ 9.68	304	18.77
Interval 05	9.68 ~ 10.85	287	17.72
Interval 06	10.85 ~ 12.02	289	17.84
Interval 07	12.02 ~ 13.19	211	13.02
Interval 08	13.19 ~ 14.36	132	8.15
Interval 09	14.36 ~ 15.53	68	4.20
Interval 10	15.53 ~ 16.70	18	1.11
Total		1620	100

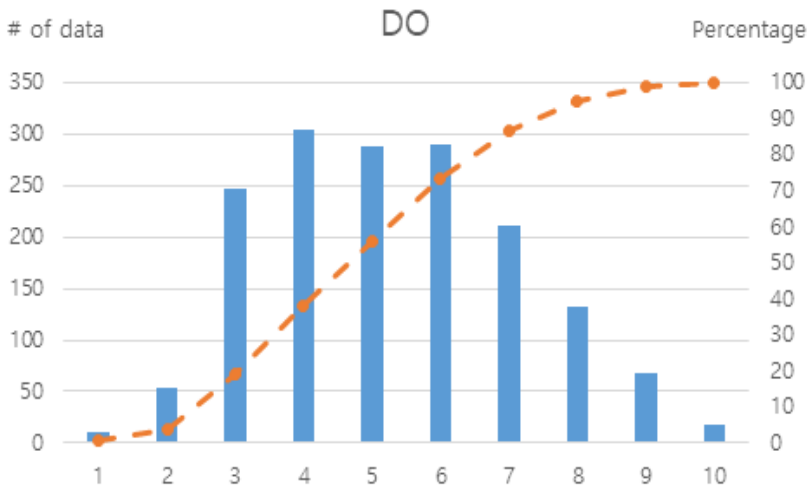


Figure 24. Data distribution from the Dasan water quality monitoring station (DO)

Table 10. Data distribution from the Dasan water quality monitoring station (Chl-a)

	range	# of Data	ratio(%)
Interval 01	0 ~ 31	1234	80.97
Interval 02	31 ~ 62	200	13.12
Interval 03	62 ~ 93	44	2.89
Interval 04	93 ~ 124	42	2.76
Interval 05	124 ~ 155		
Interval 06	155 ~ 186		
Interval 07	186 ~ 217	1	0.06
Interval 08	217 ~ 248	1	0.06
Interval 09	248 ~ 379	1	0.06
Interval 10	279 ~ 310	1	0.06
Total		1524	100

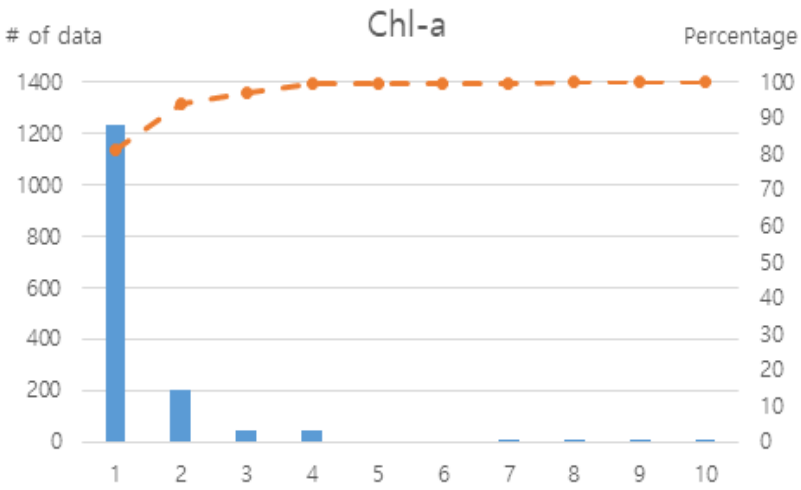


Figure 25. Data distribution from the Dasan water quality monitoring station (Chl-a)

Table 11. Data distribution from the Gangchang water quality monitoring station (pH)

	range	# of Data	ratio(%)
Interval 01	6.60 ~ 6.81	15	0.98
Interval 02	6.81 ~ 7.02	112	7.34
Interval 03	7.02 ~ 7.23	316	20.72
Interval 04	7.23 ~ 7.44	376	24.66
Interval 05	7.44 ~ 7.65	313	20.52
Interval 06	7.65 ~ 7.86	227	14.89
Interval 07	7.86 ~ 8.07	115	7.54
Interval 08	8.07 ~ 8.28	39	2.56
Interval 09	8.28 ~ 8.49	8	0.52
Interval 10	8.49 ~ 8.70	4	0.26
Total		1525	100

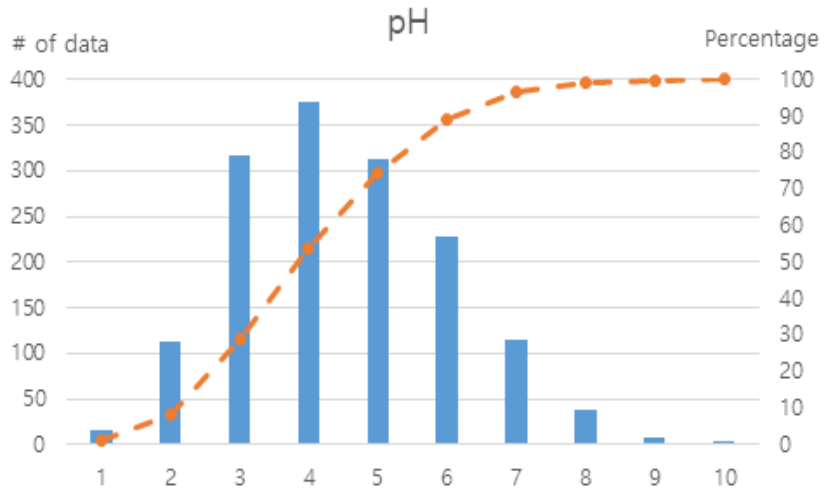


Figure 26. Data distribution from the Gangchang water quality monitoring station (pH)

Table 12. Data distribution from the Gangchang water quality monitoring station (EC)

	range	# of Data	ratio(%)
Interval 01	0 ~ 140.5	1	0.07
Interval 02	140.5 ~ 281.0	12	0.79
Interval 03	281.0 ~ 421.5	83	5.45
Interval 04	421.5 ~ 562.0	174	11.42
Interval 05	562.0 ~ 702.5	379	24.87
Interval 06	702.5 ~ 843.0	462	30.31
Interval 07	843.0 ~ 983.5	269	17.65
Interval 08	983.5 ~ 1124.0	118	7.74
Interval 09	1124.0 ~ 1264.5	20	1.31
Interval 10	1264.5 ~ 1405.0	6	0.39
Total		1524	100

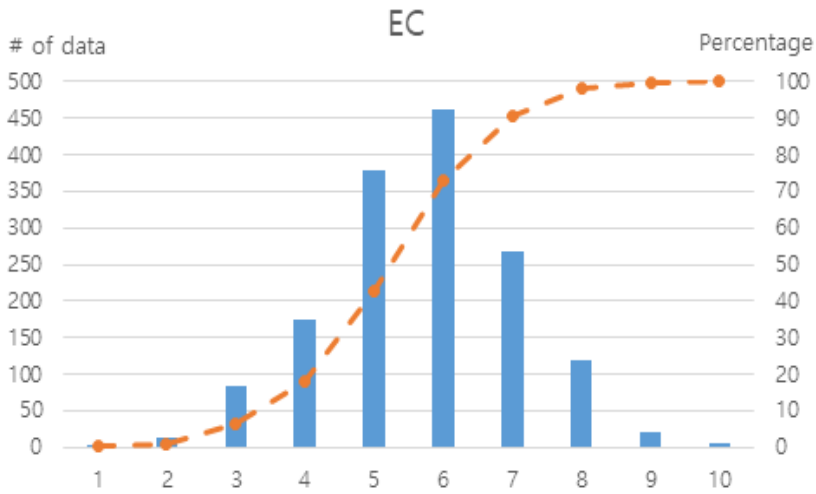


Figure 27. Data distribution from the Gangchang water quality monitoring station (EC)

Table 13. Data distribution from the Gangchang water quality monitoring station (DO)

	range	# of Data	ratio(%)
Interval 01	1.80 ~ 3.25	26	1.72
Interval 02	3.25 ~ 4.7	66	4.36
Interval 03	4.7 ~ 6.15	155	10.24
Interval 04	6.15 ~ 7.60	319	21.08
Interval 05	7.60 ~ 9.05	248	16.39
Interval 06	9.05 ~ 10.50	191	12.62
Interval 07	10.50 ~ 11.95	171	11.30
Interval 08	11.95 ~ 13.40	264	17.45
Interval 09	13.40 ~ 14.85	65	4.30
Interval 10	14.85 ~ 16.3	8	0.53
Total		1513	100

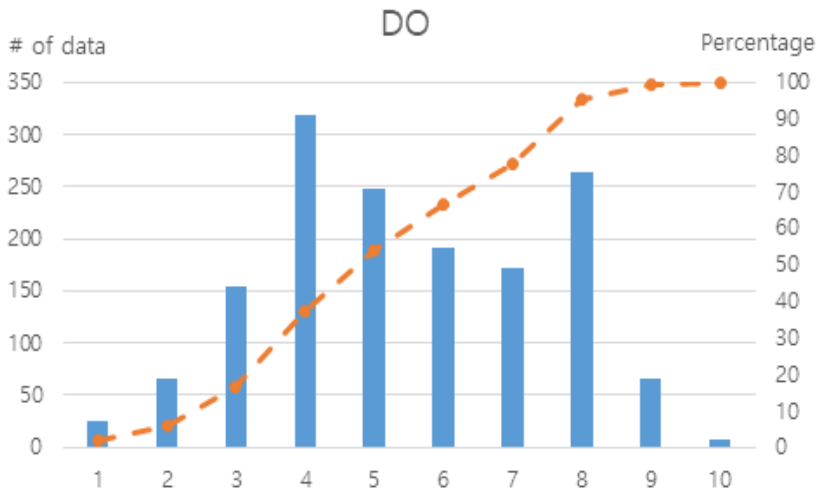


Figure 28. Data distribution from the Gangchang water quality monitoring station (DO)

Table 14. Data distribution from the Gangchang water quality monitoring station (Chl-a)

	range	# of Data	ratio(%)
Interval 01	0 ~ 13.96	710	47.84
Interval 02	13.96 ~ 27.92	390	26.28
Interval 03	27.92 ~ 41.88	201	13.54
Interval 04	41.88 ~ 55.84	110	7.41
Interval 05	55.84 ~ 69.80	35	2.36
Interval 06	69.80 ~ 83.76	21	1.42
Interval 07	83.76 ~ 97.72	12	0.81
Interval 08	97.72 ~ 111.68	4	0.27
Interval 09	111.68 ~ 125.64	0	0
Interval 10	125.64 ~ 139.6	1	0.07
Total		1484	100

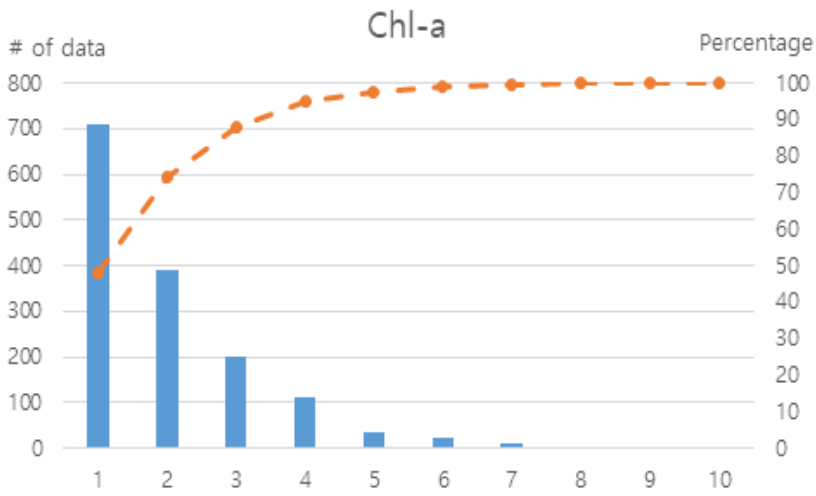


Figure 29. Data distribution from the Gangchang water quality monitoring station (Chl-a)

Table 15. Data distribution from the Goryeong water quality monitoring station (pH)

	range	# of Data	ratio(%)
Interval 01	6.4 ~ 6.7	7	0.55
Interval 02	6.7 ~ 7.0	19	1.48
Interval 03	7.0 ~ 7.2	121	9.43
Interval 04	7.2 ~ 7.5	224	17.46
Interval 05	7.5 ~ 7.8	359	27.98
Interval 06	7.8 ~ 8.1	206	16.06
Interval 07	8.1 ~ 8.4	189	14.73
Interval 08	8.4 ~ 8.6	97	7.56
Interval 09	8.6 ~ 8.9	55	4.29
Interval 10	8.9 ~ 9.2	6	0.47
Total		1283	100

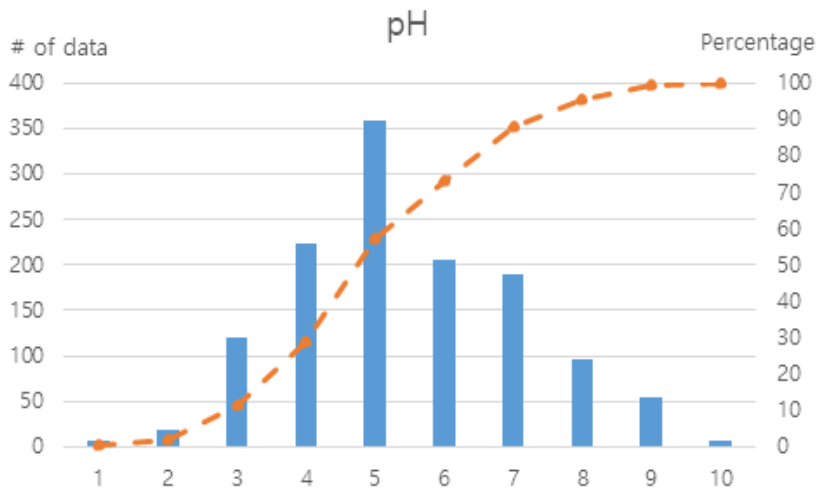


Figure 30. Data distribution from the Goryeong water quality monitoring station (pH)

Table 16. Data distribution from the Goryeong water quality monitoring station (EC)

	range	# of Data	ratio(%)
Interval 01	138 ~ 204	18	1.41
Interval 02	204 ~ 271	87	6.58
Interval 03	271 ~ 337	173	13.56
Interval 04	337 ~ 404	313	24.53
Interval 05	404 ~ 470	275	21.55
Interval 06	470 ~ 536	196	15.36
Interval 07	536 ~ 603	100	7.84
Interval 08	603 ~ 669	65	5.09
Interval 09	669 ~ 736	32	2.51
Interval 10	736 ~ 802	20	1.57
Total		1276	100

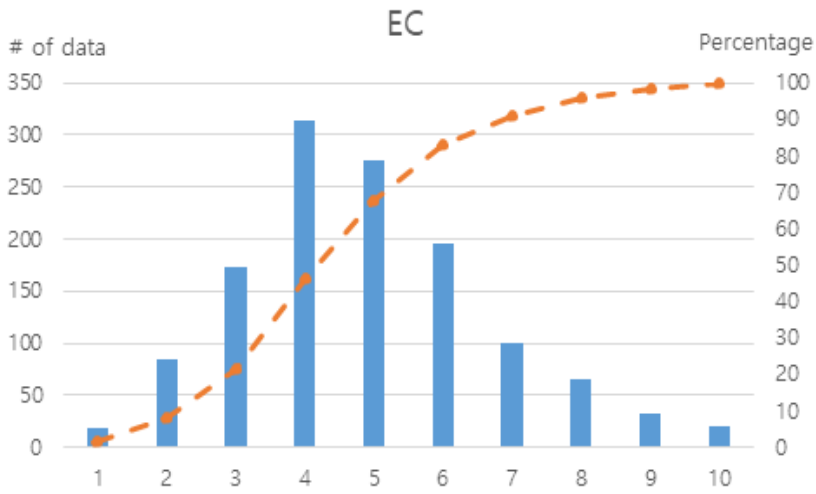


Figure 31. Data distribution from the Goryeong water quality monitoring station (EC)

Table 17. Data distribution from the Goryeong water quality monitoring station (DO)

	range	# of Data	ratio(%)
Interval 01	2.8 ~ 4.1	6	0.48
Interval 02	4.1 ~ 5.4	35	2.83
Interval 03	5.4 ~ 6.8	99	8.00
Interval 04	6.8 ~ 8.1	212	17.14
Interval 05	8.1 ~ 9.4	215	17.38
Interval 06	9.4 ~ 10.7	202	16.33
Interval 07	10.7 ~ 12.0	157	12.69
Interval 08	12.0 ~ 13.4	147	11.88
Interval 09	13.4 ~ 14.7	133	10.75
Interval 10	14.7 ~ 16.0	31	2.51
Total		1237	100

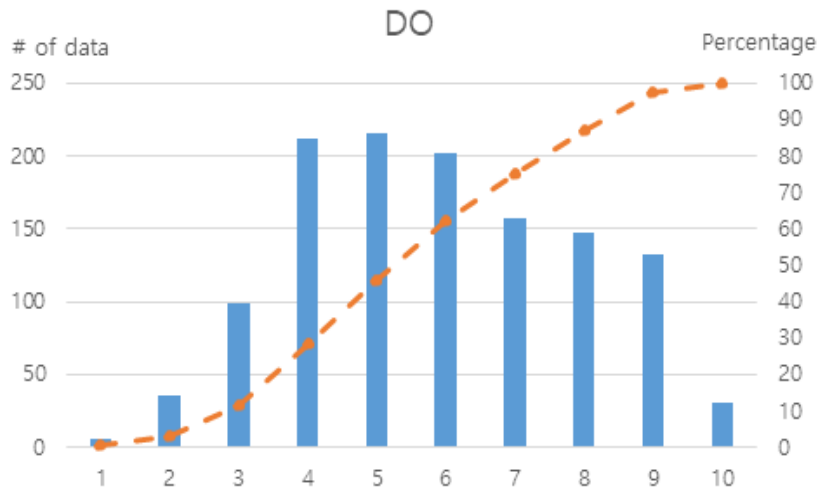


Figure 32. Data distribution from the Goryeong water quality monitoring station (DO)

Table 18. Data distribution from the Goryeong water quality monitoring station (Chl-a)

	range	# of Data	ratio(%)
Interval 01	0 ~ 15.2	374	30.31
Interval 02	15.2 ~ 30.4	4.7	35.41
Interval 03	30.4 ~ 45.7	266	21.56
Interval 04	45.7 ~ 60.9	97	7.86
Interval 05	60.9 ~ 76.1	23	1.86
Interval 06	76.1 ~ 91.3	11	0.89
Interval 07	91.3 ~ 106.5	7	0.57
Interval 08	106.5 ~ 121.8	10	0.81
Interval 09	121.8 ~ 137.0	5	0.41
Interval 10	137.0 ~ 152.2	4	0.32
Total		1234	100

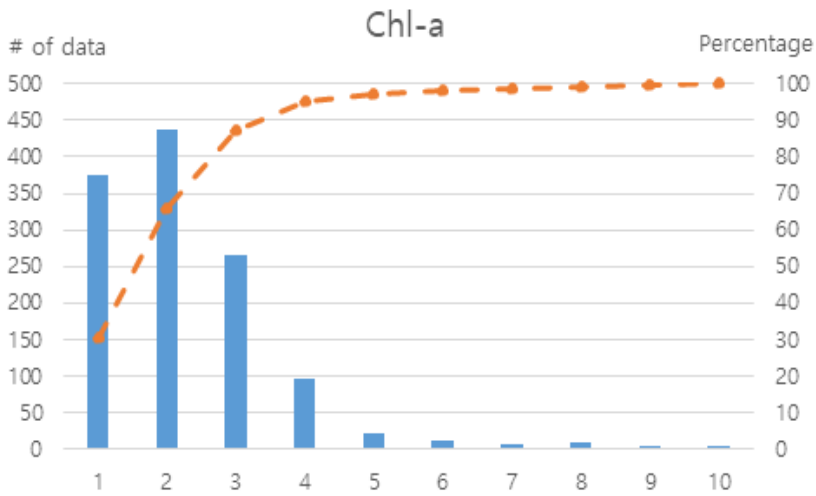


Figure 33. Data distribution from the Goryeong water quality monitoring station (Chl-a)

(3) Data from fixed water quality measuring sensors

The fixed water quality measuring sensors ARCWQ-1, ARCWQ-2, and ARCWQ-3 installed and operated by ARCROM are located downward from the confluence 2km, 6.5km, and 8.5km from the confluence respectively.

The data of water quality variables from the fixed water quality measuring sensors are correlated more with mainstream (Nakdong River) than tributary (Kumho River).

The data of pH, EC, DO shows similar mean value at ARCWQ-1 (pH 7.7, EC 428, DO 9.7) and ARCWQ-2 (pH 7.9, EC 430, DO 9.5). at the ARCWQ-3, mean value of pH are also similar with ARCWQ-1, and ARCWQ-2 but mean value of EC and DO show the higher mean value than one of ARCWQ-1 and ARCWQ-2. In case of Chl-a, the average value is decreased while flowing in the downstream direction (ARCWQ-1 16.1, ARCWQ-2 13.9, ARCWQ-3 10.8).

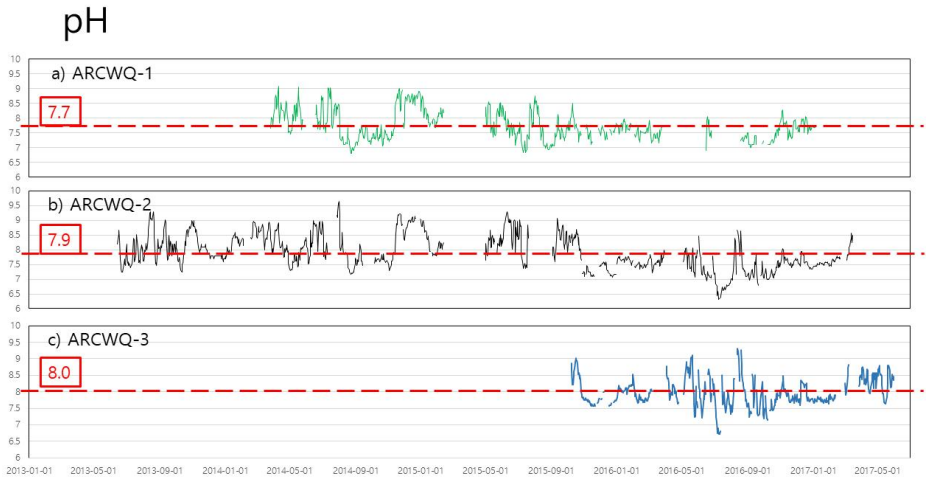


Figure 34. Data from the fixed water quality measuring sensor (pH)

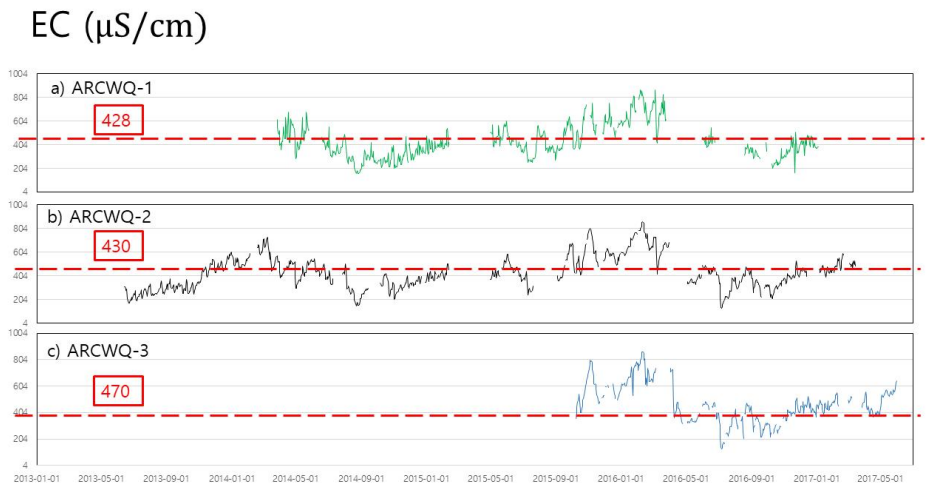


Figure 35. Data from the fixed water quality measuring sensor (EC)

Figure 36. Data from the fixed water quality measuring sensor (DO)

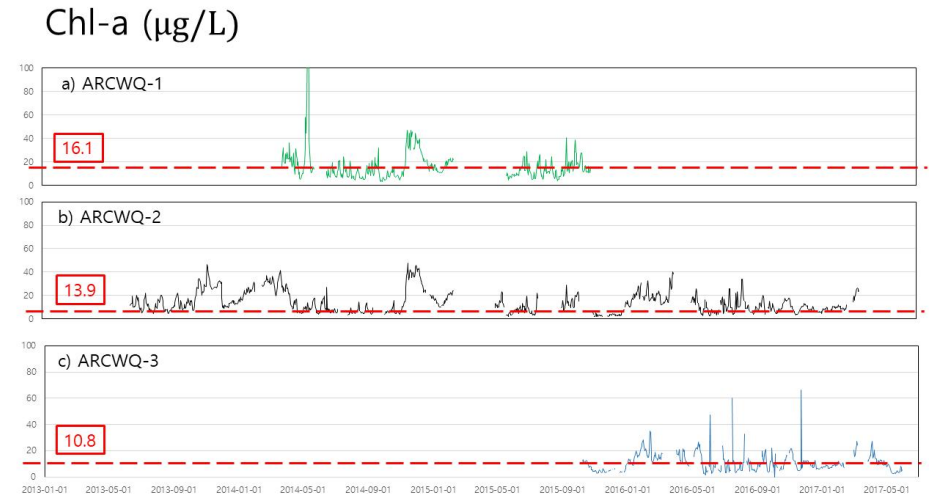


Figure 37. Data from the fixed water quality measuring sensor (Chl-a)

Table 19. Correlation factor with Dasan water quality variables

	Dasan pH	Dasan EC	Dasan DO	Dasan Chl-a
ARCWQ-1 pH	0.481			
ARCWQ-1 EC		0.753		
ARCWQ-1 DO			0.740	
ARCWQ-1 Chl-a				0.381
ARCWQ-2 pH	0.527			
ARCWQ-2 EC		0.854		
ARCWQ-2 DO			0.603	
ARCWQ-2 Chl-a				0.515
ARCWQ-3 pH	0.447			
ARCWQ-3 EC		0.837		
ARCWQ-3 DO			0.817	
ARCWQ-3 Chl-a				0.357
Goryeong pH	0.612			
Goryeong EC		0.826		
Goryeong DO			0.808	
Goryeong Chl-a				0.474

Table 20. Correlation factor with Gangchang water quality variables

	Gangchang pH	Gangchang EC	Gangchang DO	Gangchang Chl-a
ARCWQ-1 pH	0.222			
ARCWQ-1 EC		0.514		
ARCWQ-1 DO			0.769	
ARCWQ-1 Chl-a				-0.021
ARCWQ-2 pH	0.203			
ARCWQ-2 EC		0.682		
ARCWQ-2 DO			0.574	
ARCWQ-2 Chl-a				0.043
ARCWQ-3 pH	0.423			
ARCWQ-3 EC		0.609		
ARCWQ-3 DO			0.747	
ARCWQ-3 Chl-a				0.231
Goryeong pH	0.425			
Goryeong EC		0.559		
Goryeong DO			0.780	
Goryeong Chl-a				0.116

Chapter 4. Model development

4.1 Improvement of ANN model

4.1.1 Resilient Backpropagation Method

The neural network training is an unconstrained nonlinear minimization problem in which weights of a network are iteratively modified to minimize the overall mean or total squared error between the desired and actual output values for all output nodes over all input patterns. The existence of many different optimization methods (Fletcher, 1987) provides various choices for neural network training. There is no algorithm currently available to guarantee the global optimal solution for a general nonlinear optimization problem in a reasonable amount of time. As such, all optimization algorithms in practice inevitably suffer from the local optima problems and the most thing we can do is to use the available optimization method which can give the best local optima if the true global solution is not available.

The most popularly used training method is the backpropagation algorithm which is the most widely used algorithm for supervised learning with multi-layered feed-forward networks. The basic idea of the backpropagation learning algorithm is the repeated application of the chain rule to compute the influence of each weight in the network with respect to an arbitrary errorfunction E

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial s_i} \frac{\partial s_i}{\partial net_i} \frac{\partial net_i}{\partial w_{ij}}$$

(4.1)

where $w_{ij}$is the weight from neuron j to neuron i, s_i is the output, and net_i is the weighted sum of the inputs of neuron.

Once the partial derivative for each weight is known, the aim of minimizing the errorfunction is achieved by performing a simple gradient descent

$$w(h+1) = w(h) - \eta \frac{\partial E_r}{\partial w_{ij}}(h) \qquad (4.2)$$

Obviously, the choice of the learning rate η is crucial for backpropagation learning algorithm since it determines the magnitude of weight changes. It has an important effect on the time needed until convergence is reached. Furthermore it can be very sensitive to the choice of the learning rate. If it is set too small, too many steps are needed to reach an acceptable solution; on the contrary a large learning rate will possibly lead to oscillation, preventing the error to fall below a certain value.

An early way proposed to get rid of the above problem is to introduce a momentum-term :

$$\triangle w_{ij}(h) = -\eta \frac{\partial E_r}{\partial w_{ij}}(h) + \mu \triangle w_{ij}(h-1)$$

(4.3)

where the momentum parameter μ scales the influence of the previous step on the current.

The momentum-term is believed to render the learning procedure

more-stable and to accelerate convergence in shallow regions of the errorfunction.

However, the optimal value of the momentum parameter μ is equally problem dependent as the learning rate η, and no general improvement can be accomplished.

In that reason, many algorithms have been proposed so far to deal with the problem of appropriate weight-update by doing some sort of parameter adaptation during learning. They can roughly be separated into two categories : global and local strategies. Global adaptation techniques make use of the knowledge of the state of the entire network (eg. the direction of the previous weight-step) to modify global parameters. Whereas local strategies use only weight-specific information (eg. the partial derivative) to adapt weight specific parameters. Besides the fact, that local adaptation strategies are more closely related to the concept of neural learning and are better suited for parallel implementations. Their superiority over global learning algorithms has been impressively demonstrated in various published technical reports.

The majority of both global and local adaptive algorithms performs a modification of a (probably weight-specific) learning-rate according to the observed behavior of the errorfunction. The adapted learning rate is eventually used to calculate the weight-step.

What is often disregarded is that the size of the actually taken weight-step $\triangle w_{ij}$ is not only dependent on the (adapted) learning-rate, but also on the partial derivative $\frac{\partial E}{\partial w_{ij}}$. So the effect of the carefully adapted learning-rate can be drastically disturbed by the unforseeable behaviour of the derivative itself. In this study, to avoid the problem of 'blurred adaptivity', the

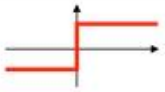
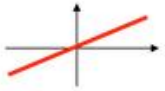
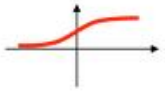
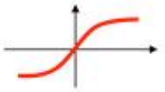
resilient propagation method was adopted. The resilient propagation method changes the size-of the weight-update $\triangle w_{ij}$ directly, without considering the size of the partial derivative.

Another reason for adopt resilient propagation method is "squasing" function. Multi-layer networks typically use sigmoid transfer functions in the hidden layers. These functions are often called "squashing" functions, since they compress an infinite input range into a finite output range. Sigmoid functions are characterized by the fact that their slope must approach zero as the input gets large. This causes a problem when using steepest descent to train a multi-layer network with sigmoid functions, since the gradient can have a very small magnitude; and therefore, cause small changes in the weights and biases, even though the weights and biases are far from their optimal values. In resilient propagation method simply 'clip' the logistic activation function in at a value, that could be reasonably distinguished from the asymptotic boundary value. This results in an always non-zero derivative, preventing the unit of getting stuck. Especially in more difficult problems, this worked far more stable than adding a small constant value to the derivation of the activation function.

In resilient back-propagation (RB) training algorithm, only the sign of the derivative is used to determine the direction of the weight update; the magnitude of the derivative has no effect on the weight update. The size of the weight change is determined by a separate update value. The update value for each weight and bias is increased by a factor whenever the derivative of the performance function with respect to that weight has the same sign for two successive iterations.

The update value is decreased by a factor whenever the derivative with respect to weight changes sign from the previous iteration. If the derivative is zero, then the update value remains the same. Whenever the weights are oscillating the weight change will be reduced. If the weight continues to change in the same direction for several iterations, then the magnitude of the weight change will be increased.

Table 21. Activation functions

Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer NN	

4.1.2 Ensemble Method

The supervised learning algorithms are commonly described as performing the task of searching through a hypothesis space to find a suitable hypothesis that will make good predictions with a particular problem. But it may be very difficult to find a good one. Ensembles combine multiple hypotheses to form a better hypothesis. The ensemble method is based on the idea that one might improve the performance of a single generic predictor, by combining the outputs of several individual predictors (Krogh and Vedelsby, 1995). Empirically, an ensemble techniques have been applied with considerable success in hydrology and environmental science, as an approach to enhance the skill of forecasts (Krogh and Vedelsby, 1995; Araghinejad et al., 2011). The application of an ensemble method is divided into two steps. The first step is the creation of individual ensemble members, and the second step is the combination of outputs from the ensemble members to produce the ensemble output (Shu and Burn, 2004, Araghinejad et al., 2011).

4.1.2.1 Creating Ensemble Members

Popular methods for creating ensemble members are Bagging

(Breiman, 1996) and Boosting (Schapire, 1990; Freund and Schapire, 1996, Shu and Burn, 2004). Bagging algorithm is based on the bootstrap statistical resampling technique (Efron and Tibshirani, 1993). In the bagging algorithm also known as 'bootstrap aggregation', the first step involves creating multiple models. These models are generated random sub-samples of the dataset which are drawn from the original dataset randomly. In bootstrap sampling, some original examples appear more than once and some original examples are not present in the sample. If you want to create a sub-dataset with m elements, you should select a random element from the original dataset m times. And if the goal is generating n dataset, you follow this step n times. At the end, we have n datasets where the number of elements in each dataset is m. The second step in bagging is aggregating the generated models. Each sub-samples can be generated independently from each other. So generation and training can be done in parallel.

The term "boosting" is used to describe a family of algorithms which are able to convert weak models to strong models. Boosting incrementally builds an ensemble by training each model with the same dataset but where the weights of instances are adjusted according to the error of the last prediction. In other words, boosting algorithm trains the first predictor with the original training set, and the training set of a new predictor is assembled based on the performance of the previous predictors. The learning patterns whose predicted values obtained from the previous predictor differ significantly from their observed values are adjusted with higher probability of being sampled. The main idea is forcing the models to focus on the instances which are hard.

4.1.2.2 Combining Outputs of Ensemble Members

Once a set of ensemble members has been created, an effective way of combining their outputs must be found. The voting and averaging are two of the easiest ensemble methods. They are both easy to understand and implement. Voting is used for classification and averaging is used for regression. In both methods, the first step is to create multiple classification/regression models using some training dataset. Each base model can be created using different splits of the same training dataset and same algorithm, or using the same dataset with different algorithms, or any other method.

Among the voting technique, majority voting is widely used technique. Every model makes a prediction (votes) for each test instance and the final output prediction is the one that receives more than half of the votes. If none of the predictions get more than half of the votes, we may say that the ensemble method could not make a stable prediction for this instance. Although this is a widely used technique, you may try the most voted prediction (even if that is less than half of the votes) as the final prediction. In some articles, this method being called "plurality voting". Unlike majority voting, in weighted voting, each model has the same rights, we can increase the

importance of one or more models.

In simple averaging method, for every instance of test dataset, the average predictions are calculated. This method often reduces overfit and creates a smoother regression model. Weighted averaging is a slightly modified version of simple averaging, where the prediction of each model is multiplied by the weight and then their average is calculated.

Stacking, also known as stacked generalization(Wolpert, 1992), is an ensemble method where the models are combined using another machine learning algorithm. The basic idea is to train machine learning algorithms with training dataset and then generate a new dataset with these models. Then this new dataset is used as input for the combiner machine learning algorithm. The training dataset for combiner algorithm is generated using the outputs of the base algorithms. The base algorithm is generated using training dataset and then the same dataset is used again to make predictions. But in the real world, it does not use the same training dataset for prediction, so to overcome this problem some implementations of stacking where training dataset is splitted.

4.1.2.3 ANN Ensemble

The generalization ability of ANN can be improved by combining several ANN in redundant ensembles, where the member networks are redundant in that each of them provides a solution to the same task, or task component, even though this solution might be obtained by different methods (Sharkey, 1999). The ANN ensembles offer a number of advantages over a single ANN in that they have the potential for

improved generalization and increased stability (Sharkey, 1999). See and Abrahart (2001) used ANN data fusion strategies for continuous river level forecasting where data fusion is the amalgamation of information from multiple sensors and/or different data sources. Abrahart and See (2002) evaluated six data fusion strategies and found that ANN data fusion provided the best solution for a stable region.

The basic ANN ensemble with simple averaging method is very simple and several researchers (Agrafiotis et al., 2002; Opitz and Maclin, 1999) found that this simple method can produce results as accurate as the more complex bagging and boosting methods. In this study, the basic network ensemble with simple averaging method was used. To the estimate the result, the root-mean-square error (RMSE), interquartile range (IQR) and R^2 value were used. The RMSE value was used to measure the whole bias error between the ensemble means and the observed values, and The IQR value was used to measure the variance error of the ANN ensemble model itself. The R^2 value is a number that indicates how well data fit a statistical model. So it was used to estimate the performance of this models.

$$RMSE = \sqrt{\frac{SS_{rss}}{n}} \quad (4.4)$$

where, n is the number of the observed data.

$$IQR_i = Q_{75}(y_i) - Q_{25}(y_i) \quad (4.5)$$

where $Q_{75}(y_i)$ and $Q_{25}(y_i)$ are the 25th and 75th percentile values of the ANN ensemble model result for the ith data set.

$$R^2 = 1 - \frac{SS_{rss}}{SS_{tss}} \tag{4.6}$$

$$SS_{tss} = \sum_i (x_i - \bar{x})^2 \tag{4.7}$$

$$SS_{rss} = \sum_i (x_i - \bar{y_i})^2 \tag{4.8}$$

SS_{tss} is the total sum of squared, and SS_{rss} is the sum of squares of residuals. x_i is the ith observed value or target value. x_i is the mean value of x_i for all the observed data set. $\bar{y_i}$ is the ensemble mean of network for the ith data set.

4.2 Model Construction

4.2.1 Data sampling

As described above, ANN model is data-driven model. So most important thing is the quality of available data. But even if the quality of the data is guaranteed, The ANN model predict different results despite using the same training input data. So, the next important thing is the pre-treatment process of the data. The imbalance of the training data set is one of the fundamental problems in ANN modeling, and has recently drawn much attention (Zhou and Liu, 2006, Alejo et al., 2007, Yoon and Kwek, 2007 and Nguyen et al., 2008). The neural network has difficulty in learning from imbalanced data sets, since the network tends to ignore the minority class, and treats it as noise, due to the overwhelming training instances of the majority class (e.g. Murphey et al., 2004 and Nguyen et al., 2008). In this study, for this reason, the stratified sampling method was used to reduce the sampling error. The stratified sampling is the process of dividing members input data into homogeneous subgroups before sampling. The strata should be mutually exclusive and also be collectively exhaustive. This often improves the

representativeness of the sample by reducing sampling error. Also it can produce a weighted mean that has less variability than the arithmetic mean of a simple random sample of the population. To avoid the biased sampling in available data set and obtain the appropriate model, training, validation, test data set was sampled by a stratified sampling method according to the distribution ratio of parameters have.

In this study, the observed data of automatic water quality monitoring stations at Goryeong station was used for model developed. The data varies from November 2013 to June 2017.

The number of each variables are 1238 in pH. 1276 in EC, 1237 in DO, and 1234 in Chl-a respectively. The stratified sampling method are used to reduce the sampling error. The distribution of the data for each factor is shown in Table 22. to Table 25. The data classified as table are averaged through ensemble method. The ANN model constructed by stratified sampling shows a more robust and less varied accurate prediction accuracy.

Table 22. Stratified samplinig results of pH (Goryeong)

Interval	distribution of data		number of sampling data set		
	# of data	ratio (%)	Training	Validation	Test
01	7	0.55	22	2	2
02	19	1.48			
03	121	9.43	97	12	12
04	224	17.46	180	22	22
05	359	27.98	289	35	35
06	206	16.06	166	20	20
07	189	14.73	153	18	18
08	97	7.56	79	9	9
09	55	4.29	49	6	6
10	6	0.47			
Total	1283	100	1035	124	124

Table 23. Stratified samplinig results of EC (Goryeong)

Interval	distribution of data		number of sampling data set		
	# of data	ratio (%)	Training	Validation	Test
01	18	1.41	16	1	1
02	87	6.58	71	8	8
03	173	13.56	139	17	17
04	313	24.53	251	31	31
05	275	21.55	221	27	27
06	196	15.36	158	19	19
07	100	7.84	80	10	10
08	65	5.09	53	6	6
09	32	2.51	26	3	3
10	20	1.57	16	2	2
Total	1276	100	1028	124	124

Table 24. Stratified samplinig results of DO (Goryeong)

boundary Interval	distribution of data		number of sampling data set		
	# of data	ratio (%)	Training	Validation	Test
01	6	0.48	33	4	4
02	35	2.83			
03	99	8	79	10	10
04	212	17.14	170	21	21
05	215	17.38	173	21	21
06	202	16.33	162	20	20
07	157	12.69	127	15	15
08	147	11.88	119	14	14
09	133	10.75	107	13	13
10	31	2.51	25	3	3
Total	1237	100	995	121	121

Table 25. Stratified samplinig results of Chl-a (Goryeong)

boundary Interval	distribution of data		number of sampling data set		
	# of data	ratio (%)	Training	Validation	Test
01	374	30.31	300	37	37
02	4.7	35.41	39	4	4
03	266	21.56	214	26	26
04	97	7.86	79	9	9
05	23	1.86	19	2	2
06	11	0.89	9	1	1
07	7	0.57	15	1	1
08	10	0.81			
09	5	0.41	7	1	1
10	4	0.32			
Total	1234	100	682	81	81

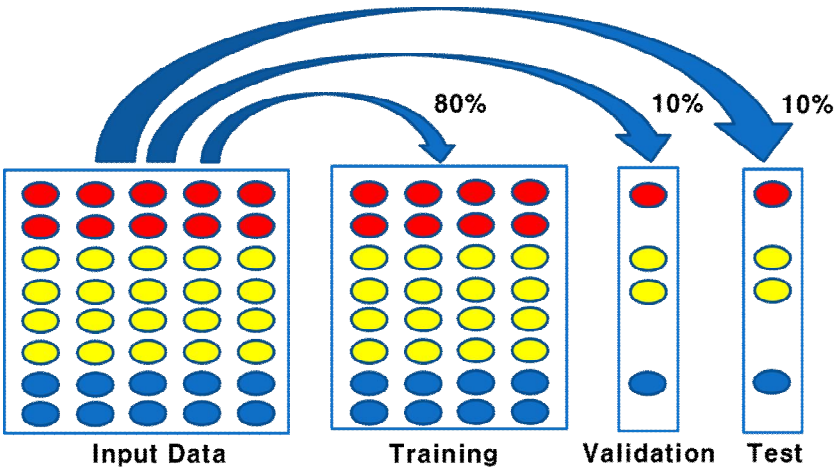


Figure 38. Stratified Sampling Method

4.2.2 ANN with RB method

The resilient propagation method performs a direct adaptation of the weight step based on local gradient information. The effort of adaptation is not blurred by gradient behaviour whatsoever.

To achieve this, each weight its individual update-value $\triangle_{ij}$, which solely determines the size of the weight-update. This adaptive update-value evolves during the learning process based on its local sight on the errorfunction E, according to the following learning-rule;

$$\triangle_{ij}^{(h+1)} = \left\{ \begin{array}{ll} \eta^{+} \bullet \triangle_{ij}^{(h)}, & \text{if } \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h)} \bullet \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h+1)} > 0 \\ \eta^{-} \bullet \triangle_{ij}^{(h)}, & \text{if } \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h)} \bullet \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h+1)} < 0 \\ \triangle_{ij}^{(h)}, & else \end{array} \right\} \qquad (4.9)$$

where $0 < \eta^{-} < 1 < \eta^{+}$

The adaptation-rule works as follows : Every time the partial derivative of the corresponding weight $\triangle w_{ij}$ changes its sign, which indicates that the last update was too big and the algorithm has jumped over a local minimum, the update-value $\triangle_{ij}$ is decreased by the factor η^{-}. If the derivative retains its sign, the update-value at is slightly increased in order to accelerate convergence in shallow regions.

Once the update-value for each weight is adapted, the

weight-update itself follows a very simple rule : if the derivative is positive (increasing error), the weight is decreased by its update-value, if the derivative is negative, the update-value is added :

$$\triangle w_{ij}^{(h+1)} = \begin{cases} -\triangle_{ij}^{(h+1)}, & \text{if } \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h+1)} > 0 \\ +\triangle_{ij}^{(h+1)}, & \text{if } \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h+1)} < 0 \\ 0 \quad , & else \end{cases} \qquad (4.10)$$

$$w_{ij}^{(h+1)} = w_{ij}^{(h)} + \triangle w_{ij}^{(h)}$$

However if the partial derivative changes sign, ie. the previous step was too large and the minimum was missed, the previous weight-update is reverted :

$$\triangle w_{ij}^{(h)} = -\triangle w_{ij}^{(h-1)}, \text{ if } \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h-1)} \cdot \left(\frac{\partial E}{\partial w_{ij}}\right)^{(h)} < 0 \qquad (4.11)$$

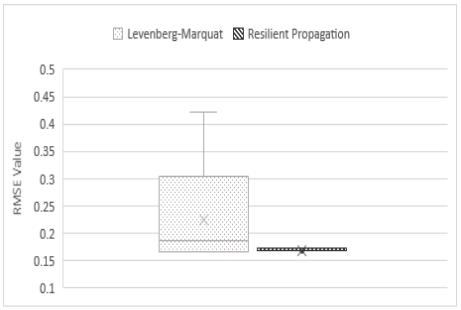
Due to that 'backtracking' weight-step, the derivative is supposed to change its sign once again in the following step. In order to avoid a double punishment of the update-value, there should be no adaptation of the update-value in the succeeding step. In practice this can be done by setting $\left(\frac{\partial E}{\partial w_{ij}}\right)^{(h-1)} = 0$ in the $\triangle_{ij}$ adaptation-rule above.

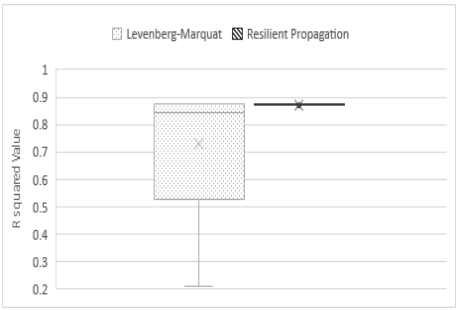
The update-values and the weights are changed every time the whole pattern set has been presented once to the n network (learning by epoch). In contrast to all other algorithms, only the sign of the partial derivative is used to perform both learning and adaptation. That

can be straight forward and very efficiently computed with respect to both time and storage consumption. Another often discussed aspect of common gradient descent is, that the weights far away from the output-layer are less modified and do learn much slower. Using resilient propagation method the size of the weight-step is only dependent on the sequence of signs, not on the magnitude of the derivative. For that reason, learning is spread equally all over the entire network; weights near the input layer have the equal chance to grow and learn as weights near the output layer.

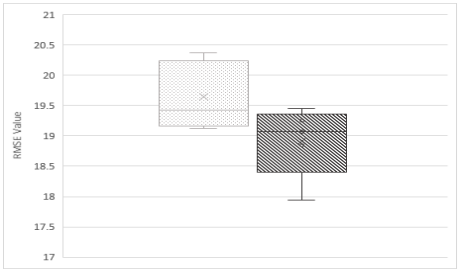
For the comparison, Levenberg-Marquat algorithm was used. The Levenberg-Marquat algorithm has been applied to the training of ANN to predict streamflow and water quality. (eg, Zamani et al, 2009).

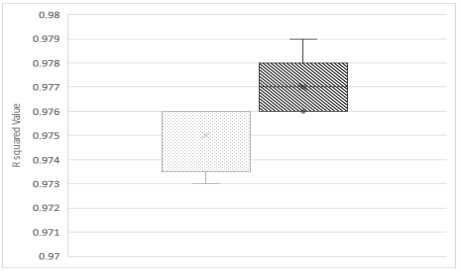
a) pH



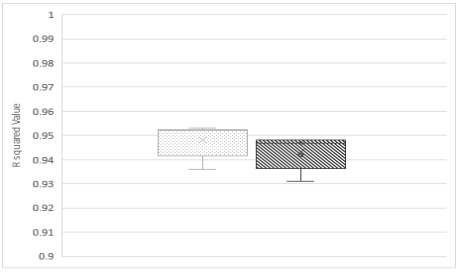


b) EC

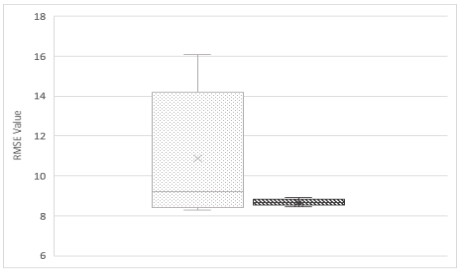


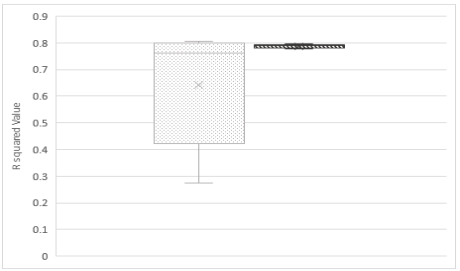


c) DO



Chl-a





< RMSE Value > < R squared Value >

Figure 39. Comparison of prediction results between LM and RB

4.3 Model Comparison

Each one-step ahead water quality prediction ANN ensemble model for pH, EC, DO and Chl-a was developed. And all available data sets of Goryeong water quality monitoring station was used for comparison. Single-output neural networks with one hidden layer were used, and the resilient propagation algorithm was used to train the network with 100 randomly generated weight parameter sets to reach the required training goal of 0.001 in 500 epochs. The tangent sigmoid activation function for the hidden layer and a linear activation function for the output layer were used.

For the comparison autoregressive models with antecedents of each variable, The AR(2) model in Eq. (4.12), was developed for training data set. The AR(2) models for each variable are shown as Table 26.

$$\text{AR(2)} : \ Y_{t+1} = \epsilon_{t+1} + \alpha Y_{t-1} + \beta Y_t \qquad (4.12)$$

where Y_{t+1} is a desired output, Y_{t-1} and Y_t are the antecedents of the desired output, and ϵ_{t+1} is a constant or noise.

Table 26. variables of AR(2) model

Y_{t+1}	ϵ_{t+1}	α	β
pH_{t+1}	0.961	−0.320	1.196
EC_{t+1}	7.396	−0.393	1.375
DO_{t+1}	0.480	−0.229	1.178
$Chl-a_{t+1}$	3.142	−0.103	0.996

Developed AR(2) models using training data were applied to test

and validation data set to compare the result with ANN ensemble models. Table 26. shows the results of each AR(2) model and the each ANN ensemble model for various numbers of hidden neurons.

In the result, ANN ensemble models show the slightly lower RMSE and higher R^2 value than the AR(2) models for all variables. In the training result of ANN ensemble models, the network with a higher number of hidden neurons was well trained, showing higher R^2 value. The optimum number of hidden neurons showing good performance in the test and validation result can be selected.

Best fitted results was show in the optimum number of hidden neurons. 8 hidden neurons for pH_{t+1}, 160 hidden neurons for EC_{t+1}, 6 hidden neurons for DO_{t+1}, and 1 hidden neuron for $Chl-a_{t+1}$.

However, the networks for $Chl-a_{t+1}$ were not well trained. Consequently, the test and validation result for $Chl-a_{t+1}$ show a higher RMSE and lower R^2 value, compared to the results of the other water quality ANN models. These higher errors were considered to be due to the imbalanced training data set of Chl-a. (Figure 33. shows the distribution ratio of the data set.) The ANN models' data set for pH, EC, DO is evenly distributed. But the data set of Chl-a is significantly imbalanced; almost 87% of the data set is included in the range between 0 and 45.7 NTU. This led to ill-training of the network, and a very high RMSE.

Table 27. Comparison of ANN model and AR(2) model

Output	# of neurons	Training RMSE	Training R^2	Validation RMSE	Validation R^2	Test RMSE	Test R^2
pH_{t+1}	1	0.191	0.834	0.173	0.842	0.172	0.879
	2	0.188	0.839	0.176	0.836	0.172	0.879
	6	0.185	0.844	0.177	0.834	0.168	0.885
	8	0.185	0.844	0.177	0.834	0.168	0.886
	12	0.184	0.847	0.179	0.830	0.168	0.885
	AR(2)	0.187	0.831	0.173	0.846	0.165	0.880
EC_{t+1}	150	19.081	0.978	18.787	0.979	19.074	0.977
	160	17.853	0.981	17.831	0.981	17.943	0.979
	170	16.784	0.983	17.470	0.982	19.258	0.976
	180	16.543	0.983	17.133	0.982	19.450	0.976
	190	16.354	0.984	16.898	0.983	18.849	0.977
	AR(2)	19.315	0.979	15.662	0.985	21.322	0.972
DO_{t+1}	1	0.768	0.922	0.754	0.921	0.742	0.931
	2	0.747	0.926	0.732	0.926	0.683	0.942
	6	0.737	0.928	0.737	0.925	0.648	0.948
	8	0.735	0.928	0.732	0.926	0.649	0.948
	12	0.729	0.929	0.740	0.924	0.652	0.947
	AR(2)	0.786	0.919	0.741	0.924	0.651	0.947
$Chl-a_{t+1}$	1	9.286	0.824	8.812	0.813	8.469	0.798
	2	9.060	0.833	8.956	0.807	8.644	0.790
	6	8.451	0.854	8.896	0.809	8.669	0.789
	8	8.283	0.860	8.983	0.805	8.752	0.785
	12	8.097	0.866	9.244	0.794	8.903	0.777
	AR(2)	7.665	0.825	8.460	0.841	8.301	0.804

Chapter 5. Model Application

5.1 Simulation Case

In this study, one-step ahead forcasting ANN models for water quality prediction were applied to the study site shown in Figure 8. In the application, the observed daily data from 2009 to 2012 was used to predict water quality in the following day.

The observed data of automatic water quality monitoring stations consists of 1524 data set of Dasan station with four water quality variables; pH, EC, DO, and Chl-a. 1484 data set of Gangchang station with four water quality variables; pH, EC, DO, and Chl-a. 1234 data set of Goryeong station with four water quality variables; pH, EC, DO, and Chl-a .

The observed data of fixed water quality monitoring sensors by ARCROM consists of 390 data set of ARCWQ-1 monitoring station with four water quality variables; pH, EC, DO, and Chl-a. 540 data set of ARCWQ-2 monitoring station with four water quality variables; pH, EC, DO, and Chl-a. 125 data set of ARCWQ-3 monitoring station with four water quality variables; pH, EC, DO, and Chl-a.

Among the observed data sets, the three data sets of fixed water quality monitoring sensor by ARCROM and one set of Goryeong water quality monitoring station were used for target data of prediction after confluence. The two data sets of water quality monitoring station, Dasan and Gangchang, were used for input data of prediction before confluence. All the variables of input data were discretized into time t (day) and t-1.

5.2 Prediction of Water Quality Variables

5.2.1 Prediction of Water Quality for Each site

ANN models with input variables for various number of hidden neurons were developed using resilience propagation algorithm. According to the validation result, the optimum number of hidden neurons for pH, EC and Chl-a was used. The test results also showed the best results when using the optimum number of hidden neurons.

(1) Prediction of water quality variables at ARCWQ-1

In case of pH, the case of best fitted prediction results is shown when using pH values of mainstream and tributary both as the input data. The increase rate of R^2 value is 34.3% when using pH data of mainstream and tributary both as the input data compared to using pH data of mainstream only. The increase rate of R^2 value is 30.0% when using pH data of both mainstream and tributary, also discharge data of tributary as the input data compared to using pH data of mainstream only. In case of pH, the addition of water quality data of tributary had more influence on the prediction accuracy than the addition of both water quality and discharge data of tributary as input data.

In case of EC, best fitted prediction results is shown when using EC values of mainstream and tributary both as the input data. The increase rate of R^2 value is 11.8% when using EC data of mainstream and tributary both as the input data compared to

using EC data of mainstream only. The increase rate of R^2 value is 10.6% when using EC data of both mainstream and tributary, also discharge data of tributary as the input data compared to using EC data of mainstream only. Similar to the case of pH, the addition of the water quality data of tributary had more influence on the prediction accuracy than the addition of water quality and discharge data of tributary as input data.

In case of DO, best fitted prediction results is shown when using DO data of both mainstream and tributary, also discharge value of tributary as the input data. the increase rate of R^2 value is 8.2% when using DO values of mainstream and tributary both as the input data compared to using DO data of mainstream only. The increase rate of R^2 value is 10.9% when using DO data of both mainstream and tributary, also discharge data of tributary as the input data compared to using DO data of mainstream only. In the case of DO, the addition of water quality value and discharge of tributary as input data had more influence on the prediction accuracy than the addition of water quality data of tributary as input data.

In case of Chl-a, best fitted prediction results is shown when using Chl-a data of mainstream and tributary both as the input data. The increase rate of R^2 value is 57.9% when using Chl-a values of mainstream and tributary both as the input data as compared with using Chl-a values of mainstream only. The increase rate of R^2 value is 54.1% when using Chl-a data of both mainstream and tributary, also discharge value of tributary as the input data as compared with using Chl-a data of

mainstream only. The addition of the water quality data of tributary had more influence on the prediction accuracy than the addition of water quality data and discharge of tributary as input data.

(2) Prediction of water quality variables at ARCWQ-2

In case of pH, the case of best fitted prediction results is shown when using pH values of mainstream and tributary both as the input data. The increase rate of R^2 value is 22.0% when using pH values of mainstream and tributary both as the input data compared to using pH data of mainstream only. However, prediction result was decrease when using pH values of both mainstream and tributary, also discharge value of tributary as the input data compared to using pH values of mainstream as the input data only. The addition of water quality variables of tributary had more influence on the prediction accuracy than the addition of water quality value and discharge of tributary as input data.

In case of EC, best fitted prediction results is shown when using EC values of mainstream and tributary both as the input data. The increase rate of R^2 value is 26.2% when using EC values of mainstream and tributary both as the input data compared to using EC data of mainstream only. The increase rate of R^2 value is 24.8% when using EC values of both mainstream and tributary, also discharge value of tributary as the input data compared to using EC data of mainstream only. Similar to the

case of ARCWQ-1, the addition of water quality variables of tributary had more influence on the prediction accuracy than the addition of water quality value and discharge of tributary as input data.

In case of DO, best fitted prediction results is shown when using DO values of both mainstream and tributary, also discharge value of tributary as the input data. The increase rate of R^2 value is 77.3% when using DO values of mainstream and tributary both as the input data compared to using DO data of mainstream only. The increase rate of R^2 value is 86.0% when using DO values of both mainstream and tributary, also discharge value of tributary as the input data compared to using DO data of mainstream only. In the case of DO, the addition of water quality value and discharge of tributary as input data had more influence on the prediction accuracy than the addition of water quality variables of tributary as input data.

In case of Chl-a, best fitted prediction results is shown when using Chl-a values of mainstream and tributary both as the input data. The increase rate of R^2 value is 19.5% when using Chl-a values of mainstream and tributary both as the input data as compared with using Chl-a data of mainstream only. The increase rate of R^2 value is 19.3% when using Chl-a values of mainstream and tributary both, also discharge value of tributary as the input data as compared with using pH data of mainstream only.

(3) Prediction of water quality variables at ARCWQ-3

In case of pH, the case of best fitted prediction results is shown when using pH values of mainstream and tributary both as the input data. The increase rate of R^2 value is 47.7% when using pH values of mainstream and tributary both as the input data compared to using pH data of mainstream only. The increase rate of R^2 value is 41.4% when using pH values of mainstream and tributary both, also discharge value of tributary as the input data compared to using pH data of mainstream only. Similar to the case of ARCWQ-1 the addition of water quality variables of tributary had more influence on the prediction accuracy than the addition of water quality value and discharge of tributary as input data.

In case of EC, best fitted prediction results is shown when using EC values of mainstream and tributary both as the input data. The increase rate of R^2 value is 11.4% when using EC values of mainstream and tributary both as the input data compared to using EC data of mainstream only. The increase rate of R^2 value is 9.6% when using EC values of mainstream and tributary both, also discharge value of tributary as the input data compared to using EC data of mainstream only. Similar to the case of pH, the addition of the water quality variables of tributary had more influence on the prediction accuracy than the addition of water quality variables and discharge of tributary as input data.

In case of DO, best fitted prediction results is shown when using DO values of mainstream and tributary both as the input data. The increase rate of R^2 value is 12.9% when using DO values of

mainstream and tributary both as the input data compared to using DO data of mainstream only. The increase rate of R^2 value is 12.3% when using DO values of mainstream and tributary both, also discharge value of tributary as the input data compared to using DO data of mainstream only.

In case of Chl-a, best fitted prediction results is shown when using Chl-a values of mainstream and tributary both as the input data. However, the prediction accuracy is similar when using Chl-a values of mainstream and tributary both as the input data and using Chl-a data of mainstream only. Prediction result was decrease when using Chl-a values of mainstream and tributary both, also discharge value of tributary as the input data compared to using Chl-a values of mainstream as the input data only.

(4) Prediction of water quality variables at Goryeong Station

In case of pH, the case of best fitted prediction results is shown when using pH values of mainstream and tributary both, also discharge value of tributary as the input data. The increase rate of R^2 value is 20.6% when using pH values of mainstream and tributary both as the input data compared to using pH data of mainstream only. The increase rate of R^2 value is 56.2% when using pH values of mainstream and tributary both, also discharge value of tributary as the input data compared to using pH data of mainstream only. In the case of pH, the addition of water quality value and discharge of tributary as input data had more influence on the prediction accuracy than the addition of

water quality variables of tributary as input data.

In case of EC, best fitted prediction results is shown when using EC values of mainstream and tributary both as the input data. The increase rate of R^2 value is 11.7% when using EC values of mainstream and tributary both as the input data compared to using EC data of mainstream only. The increase rate of R^2 value is 11.0% when using EC values of mainstream and tributary both, also discharge value of tributary as the input data compared to using EC data of mainstream only.

In case of DO, best fitted prediction results is shown when using DO values of mainstream and tributary both, also discharge value of tributary as the input data. The increase rate of R^2 value is 10.8% when using DO values of mainstream and tributary both as the input data compared to using DO data of mainstream only. The increase rate of R^2 value is 11.5% when using DO values of mainstream and tributary both, also discharge value of tributary as the input data compared to using DO data of mainstream only. In the case of DO, the addition of water quality value and discharge of tributary as input data had more influence on the result than the addition of water quality variables of tributary as input data.

In case of Chl-a, best fitted prediction results is shown when using Chl-a values of mainstream and tributary both, also discharge value of tributary as the input data. The increase rate of R^2 value is 46.6% when using Chl-a values of mainstream and tributary both as the input data as compared with using Chl-a data of mainstream only. The increase rate of R^2 value is 50.0% when

using Chl-a values of mainstream and tributary both, also discharge value of tributary as the input data as compared with using Chl-a data of mainstream only. In the case of DO, the addition of water quality value and discharge of tributary as input data had more influence on the result than the addition of water quality variables of tributary as input data.

5.2.2 Prediction of water quality through the reach

In all case of water quality variables prediction, the best fitted prediction results is shown when using water quality data of mainstream and tributary both as the input data within the 10km after the confluence.

Since the sensor is installed in the right bank of mainstream, the effect of water quality of tributary was able to be detected by the sensor after transverse mixing was completed. So all variables show best fitted prediction results at the point where the water quality of the tributary affects the water quality of the mainstream (0.56 of pH and 0.75 of DO at ARCWQ-3, 0.80 of EC and 0.66 of Chl-a at ARCWQ-2).

Especially for electric conductivity (EC), the accuracy of prediction was almost same when using EC values of mainstream and tributary both as the input data and using EC data of mainstream only at ARCWQ-1. While at ARCWQ-2, which is about 4.5km downstream from the ARCWQ-1, the prediction accuracy increased when using EC data of both main stream and tributary. This is because the water quality at ARCWQ-1 does not show the influence of water quality

introduced from the tributaries since the water quality inflowing from the tributaries are not mixed in the lateral direction immediately after the confluence. But, the mixing of the water quality at ARCWQ-2 was completed in the lateral direction. This result is quite similar to the result of analyzing the mixing behavior of EC through field experiments in the same target area (National Institute of Environmental Research, 2015).

Table 28. Prediction results of ANN model using water quality value of mainstream as the input data

Output	Station	Validation result		Test result	
		RMSE	R^2	RMSE	R^2
pH_{t+1}	ARCWQ1	0.581	0.211	0.559	0.236
	ARCWQ2	0.495	0.402	0.489	0.336
	ARCWQ3	0.389	0.563	0.459	0.377
	Goryeong	0.346	0.381	0.353	0.379
EC_{t+1}	ARCWQ1	73.481	0.518	74.476	0.415
	ARCWQ2	61.850	0.698	68.918	0.634
	ARCWQ3	51.583	0.888	64.471	0.665
	Goryeong	63.209	0.677	51.368	0.665
DO_{t+1}	ARCWQ1	2.082	0.556	2.060	0.534
	ARCWQ2	3.268	0.256	2.924	0.278
	ARCWQ3	0.978	0.731	1.048	0.661
	Goryeong	1.619	0.678	1.531	0.693
$Chl-a_{t+1}$	ARCWQ1	6.757	0.546	10.649	0.235
	ARCWQ2	5.258	0.605	5.695	0.554
	ARCWQ3	4.620	0.522	4.023	0.619
	Goryeong	15.877	0.493	16.175	0.431

Table 29. Prediction results of ANN model using water quality value of both mainstream and tributary as the input data

Output	Station	Validation result		Test result	
		RMSE	R^2	RMSE	R^2
pH_{t+1}	ARCWQ1	0.572	0.234	0.510	0.317
	ARCWQ2	0.514	0.356	0.487	0.410
	ARCWQ3	0.399	0.542	0.422	0.557
	Goryeong	0.329	0.441	0.293	0.457
EC_{t+1}	ARCWQ1	67.361	0.595	67.534	0.464
	ARCWQ2	52.730	0.781	50.988	0.800
	ARCWQ3	44.487	0.916	56.708	0.741
	Goryeong	47.750	0.815	44.985	0.743
DO_{t+1}	ARCWQ1	1.305	0.826	1.961	0.578
	ARCWQ2	3.093	0.333	2.45	0.493
	ARCWQ3	0.914	0.765	0.907	0.746
	Goryeong	1.347	0.778	1.329	0.768
$Chl-a_{t+1}$	ARCWQ1	6.027	0.639	9.655	0.371
	ARCWQ2	4.718	0.682	4.956	0.662
	ARCWQ3	4.677	0.510	3.996	0.624
	Goryeong	13.688	0.623	13.011	0.632

Table 30. Prediction results of ANN model using water quality value of both mainstream and tributary, also discharge value of tributary as an input data

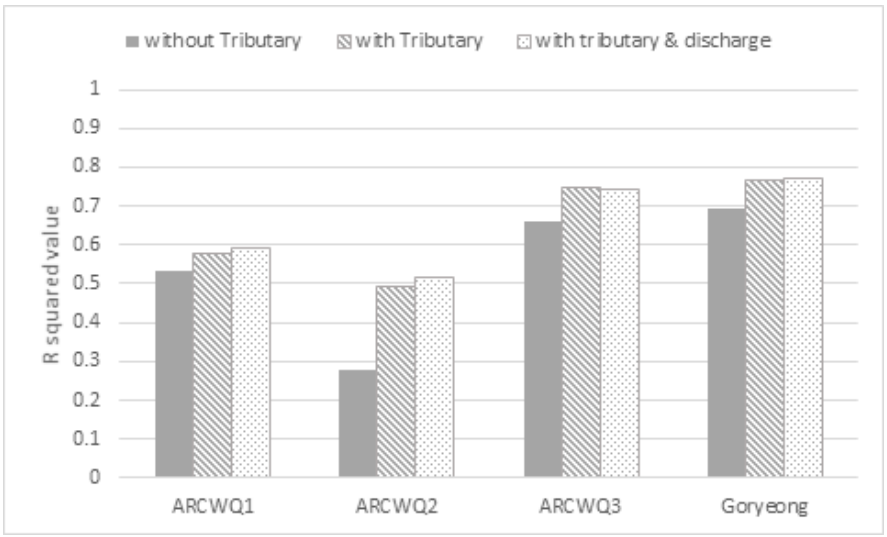
Output	Station	Validation result		Test result	
		RMSE	R^2	RMSE	R^2
pH_{t+1}	ARCWQ1	0.541	0.317	0.532	0.307
	ARCWQ2	0.492	0.410	0.494	0.322
	ARCWQ3	0.392	0.557	0.398	0.533
	Goryeong	0.324	0.457	0.285	0.592
EC_{t+1}	ARCWQ1	63.657	0.638	67.845	0.459
	ARCWQ2	49.958	0.803	52.12	0.791
	ARCWQ3	40.851	0.930	57.954	0.729
	Goryeong	48.932	0.806	45.368	0.738
DO_{t+1}	ARCWQ1	1.359	0.811	1.929	0.592
	ARCWQ2	3.005	0.371	2.391	0.517
	ARCWQ3	0.909	0.768	0.913	0.742
	Goryeong	1.333	0.782	1.361	0.773
$Chl-a_{t+1}$	ARCWQ1	6.432	0.589	9.728	0.362
	ARCWQ2	4.613	0.696	4.967	0.661
	ARCWQ3	4.497	0.547	4.220	0.581
	Goryeong	14.506	0.576	12.754	0.646

a) pH

b) EC

Figure 40. Improvement of prediction accuracy

c) DO



d) Chl-a

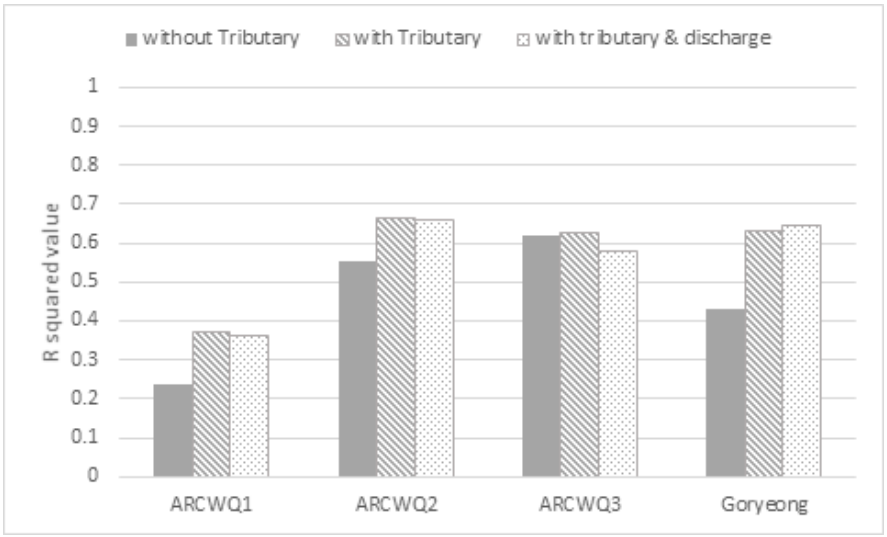
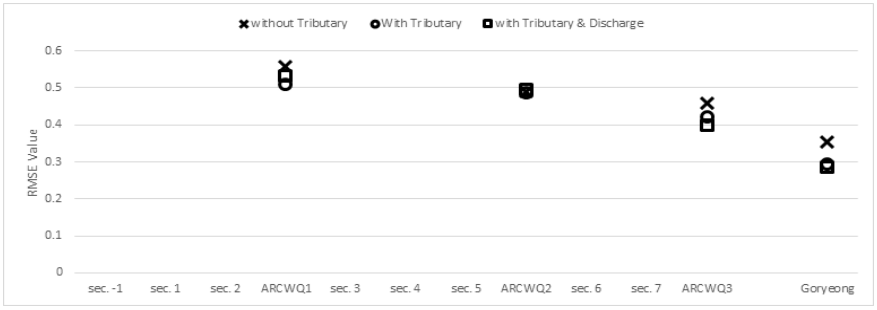
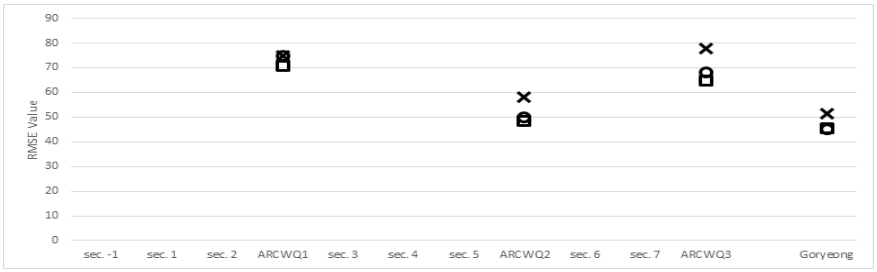


Figure 40. Improvement of prediction accuracy (cont.)

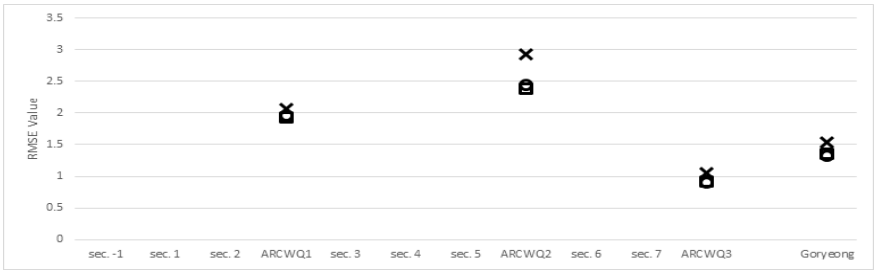
a) pH



b) EC



c) DO



d) Chl-a

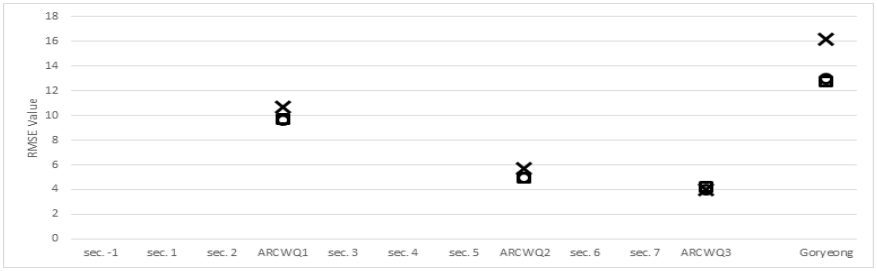
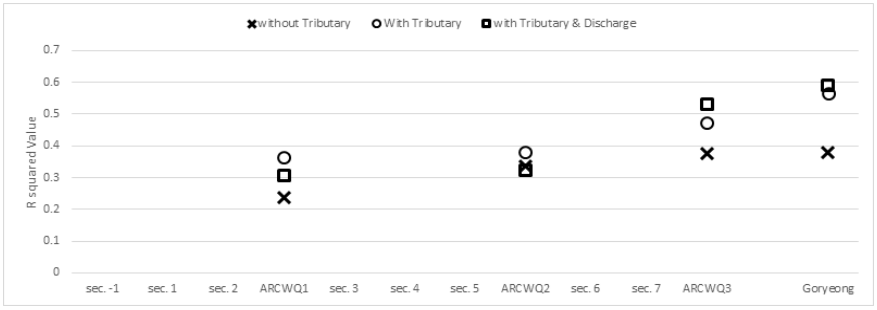
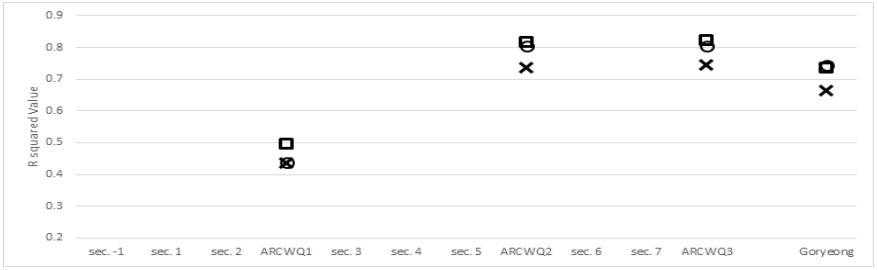


Figure 41. Comparison of Prediction results of ANN model (RMSE value)

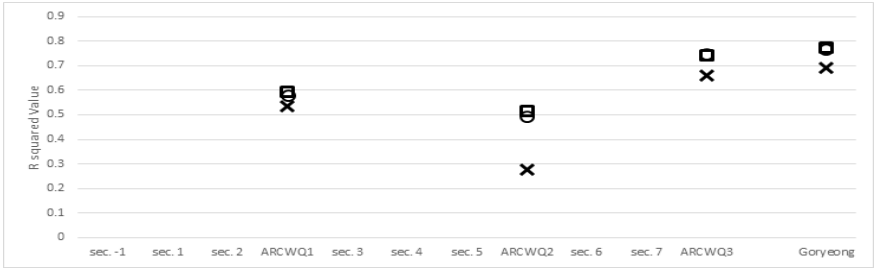
a) pH



b) EC



c) DO



d) Chl-a

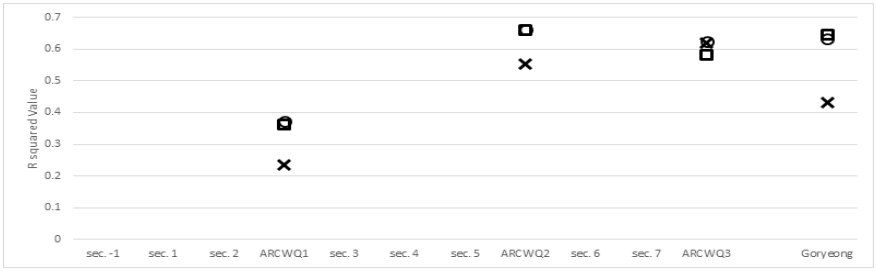
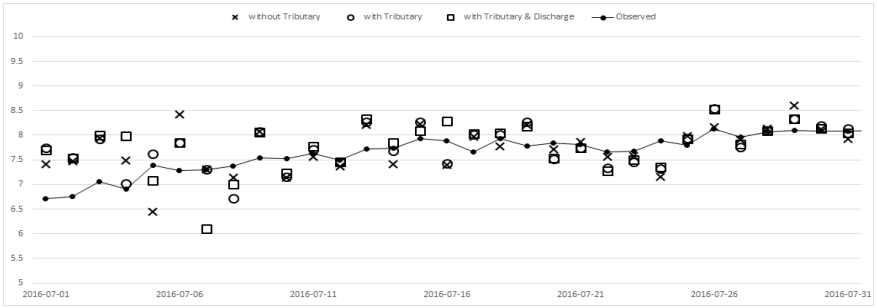
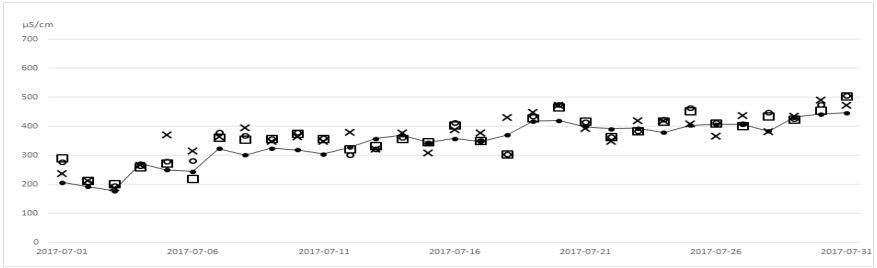


Figure 42. Comparison of Prediction results of ANN model (R Square value)

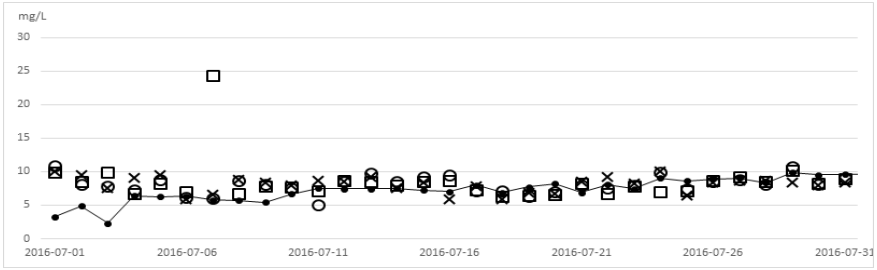
a) pH



b) EC



c) DO



d) Chl-a

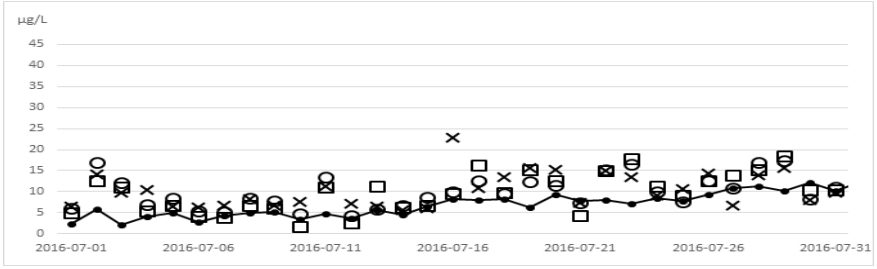


Figure 43. Prediction results of each variables

Chapter 6. Conclusions

In this study, prediction results of developed ANN ensemble model using resilient propagation algorithm and sampling method shows more accurate and robust than the AR(2) model and generally used ANN ensemble model with Levenberg-Marquardt algorithm.

Using this model, prediction of future water quality in unmeasured area was performed. In this study site, the water quality of the tributaries greatly influences the water quality of the mainstream, so the water quality value of the mainstream and tributary both are selected as the input data. As a result, the prediction accuracy was higher within 10 km after confluence when using the water quality value of the tributaries and mainstream both as an input data than the case of using the water quality value of mainstream as the input data only. However, the accuracy of prediction was not improved when using water quality values of mainstream and tributary and discharge value of tributary as the input data in the same section.

In the field EC tracer experiment conducted in 2015, it was confirmed that the transverse mixing is proceeding from 400m after the confluence. After 2 km after confluence, the transverse mixing increased gradually and after 4.5 km, the mixing was completed and the EC concentration became the same in the left and right bank of mainstream.

In the case of ARCWQ-1 where the mixing in the lateral direction was not completed, the accuracy of the model prediction was not improved when using the water quality value of the mainstream and tributary both as the input data. However, in case of ARCWQ-2 and ARCWQ-3 where the mixing in the lateral direction was completed, the

accuracy of the model prediction is greatly improved when the water quality value of tributary is added as input data. Especially in the case of EC and Chl-a, the accuracy improvement was larger when the water quality factor of the tributary was added after the lateral mixing was completed than before the lateral mixing. Also in case of EC prediction, the data model also reproduces the pollutant behavior measured by the field experiment.

As a result, in order to predict the water quality after the confluence, the influence of the water quality parameters from the tributary should be considered. But more important thing is to consider the effects of lateral mixing than simply adding the water quality parameters of the tributaries.

References

전태명(2005). 자연하천에서 오염물질의 횡분산 특성. 석사학위논문, 서울대학교

정성현(2015). 자연하천에서 전기전도도 추적을 통한 2차원 혼합에 관신 실험적, 수치적 연구. 석사학위논문, 서울대학교

Alejo, R., García, V., Sotoca, J. M., Mollineda, R. A., & Sánchez, J. S. (2007, June). Improving the performance of the RBF neural networks trained with imbalanced samples. In *International Work-Conference on Artificial Neural Networks (pp. 162-169).* Springer, Berlin, Heidelberg.

Alizadeh, M. J., Kavianpour, M. R., Tahershamsi, A., & Shahheydari, H. (2015). A wavelet-ANN approach toinvestigate the Effect of Seasonal Decomposition of Time Series in Daily River Flow Forecasting. In *Proceeding 10th International Congress on Civil Engineering*.

Araghinejad, S., Azmi, M., & Kholghi, M. (2011). Application of artificial neural network ensembles in probabilistic hydrological forecasting. *Journal of Hydrology*, 407(1-4), 94-104.

Khalil, B. M., Awadallah, A. G., Karaman, H., & El-Sayed, A. (2012). Application of artificial neural networks for the prediction of water quality variables in the nile delta. *Journal of Water Resource and Protection*, 4(06), 388.

Berardi, V. L., & Zhang, G. P. (1999). The effect of misclassification costs on neural network classifiers. *Decision Sciences*, 30(3), 659-682.

Boucher, M. A., Perreault, L., & Anctil, F. (2009). Tools for the assessment of hydrological ensemble forecasts obtained by neural networks. *Journal of Hydroinformatics*, 11(3-4), 297-307.

Chang, Y. T., Lin, J., Shieh, J. S., & Abbod, M. F. (2012). Optimization the initial weights of artificial neural networks via genetic algorithm applied to hip bone fracture prediction. *Advances in Fuzzy Systems*, 2012, 6.

Chiu, S. L. (1994). Fuzzy model identification based on cluster estimation. *Journal of Intelligent & fuzzy systems*, 2(3), 267-278.

Coulibaly, P., Anctil, F., & Bobee, B. (1999). Hydrological forecasting with artificial neural networks: the state of the art. *Canadian Journal of Civil Engineering*, 26(3), 293-304.

Dawson, C. W., & Wilby, R. L. (2001). Hydrological modelling using artificial neural networks. *Progress in physical Geography*, 25(1), 80-108.

Dogan, E., Sengorur, B., & Koklu, R. (2009). Modeling biological oxygen demand of the Melen River in Turkey using an artificial neural network technique. *Journal of Environmental Management*, 90(2), 1229-1235.

Elshorbagy, A., Corzo, G., Srinivasulu, S., & Solomatine, D. P. (2010). Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology-Part 1: Concepts and methodology. *Hydrology and Earth System Sciences,* 14(10), 1931-1941.

Elshorbagy, A., Corzo, G., Srinivasulu, S., & Solomatine, D. P. (2010). Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology-Part 2: Application. *Hydrology and Earth System Sciences,* 14(10), 1943-1961.

Faruk, D. Ö. (2010). A hybrid neural network and ARIMA model for water quality time series prediction. *Engineering Applications of Artificial Intelligence,* 23(4), 586-594.

Fischer, H. B., List, J. E., Koh, C. R., Imberger, J., and Brooks, N. H. (2013). Mixing in inland and coastal waters(p.109 - P.120). Academic press.

Gazzaz, N. M., Yusoff, M. K., Aris, A. Z., Juahir, H., & Ramli, M. F. (2012). Artificial neural network modeling of the water quality index for Kinta River (Malaysia) using water quality variables as predictors. *Marine pollution bulletin*, 64(11), 2409-2420.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. (2000). Artificial neural networks in hydrology. II: Hydrologic applications. *Journal of Hydrologic Engineering*, 5(2), 124-137.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks*, 5(6), 989-993.

Heydari, M., Olyaie, E., Mohebzadeh, H., & Kisi, Ö. (2013). Development of a neural network technique for prediction of water quality parameters in the Delaware River, Pennsylvania. *Middle-East Journal of Scientific Research*, 13(10), 1367-1376.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002). Cluster validity methods: part I. *ACM Sigmod Record*, 31(2), 40-45.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002). Clustering validity checking methods: part II. *ACM Sigmod Record*, 31(3), 19-27.

Hammouda, K., & Karray, F. (2000). A comparative study of data clustering techniques. *University of Waterloo, Ontario, Canada.*

Hansen, L. K., & Salamon, P. (1990). Neural network ensembles. *IEEE transactions on pattern analysis and machine intelligence*, 12(10), 993-1001.

Haykin, S., & Network, N. (2004). A comprehensive foundation. *Neural networks*, 2(2004), 41.

Kasiviswanathan, K. S., Cibin, R., Sudheer, K. P., & Chaubey, I. (2013). Constructing prediction interval for artificial neural network rainfall runoff models based on ensemble simulations. *Journal of hydrology*, 499, 275-288.

Khalil, B., Ouarda, T.B.M.J., St-Hilaire, A., 2011. Estimation of water quality characteristics at ungauged sites using artificial neural networks and canonical correlation analysis. J. Hydrol. 405, 277e287.

Khosravi, A., Nahavandi, S., Creighton, D., & Atiya, A. F. (2011). Comprehensive review of neural network-based prediction intervals and new advances. *IEEE Transactions on neural networks*, 22(9), 1341-1356.

Kişi, Ö. (2008). River flow forecasting and estimation using different artificial neural network techniques. *Hydrology Research*, 39(1), 27-40.

Kolen, J. F., & Pollack, J. B. (1991). Back propagation is sensitive to initial conditions. In *Advances in neural information processing systems* (pp. 860-867).

Krogh, A., & Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. In *Advances in neural information processing systems* (pp. 231-238).

Laucelli, D., Giustolisi, O., Babovic, V., & Keijzer, M. (2007). Ensemble modeling approach for rainfall/groundwater balancing. *Journal of Hydroinformatics,* 9(2), 95-106.

Lu, Y., Guo, H., & Feldkamp, L. (1998, May). Robust neural learning from unbalanced data samples. In *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on* (Vol. 3, pp. 1816-1821). IEEE.

Maier, H. R., Jain, A., Dandy, G. C., & Sudheer, K. P. (2010). Methods used for the development of neural networks for the prediction of water resource variables in river systems: current status and future directions. *Environmental Modelling & Software*, 25(8), 891-909.

Mulia, I. E., Tay, H., Roopsekhar, K., & Tkalich, P. (2013). Hybrid ANN-GA model for predicting turbidity and chlorophyll-a concentrations. *Journal of hydro-environment research*, 7(4), 279-299.

Murphey, Y. L., Guo, H., & Feldkamp, L. A. (2004). Neural learning from unbalanced data. *Applied Intelligence*, 21(2), 117-128.

Musavi-Jahromi, S. H., & Golabi, M. (2008). Application of artificial neural networks in the river water quality modeling: Karoon river, Iran.

Nguyen, G. H., Bouzerdoum, A., & Phung, S. L. (2008, December). A supervised learning approach for imbalanced data sets. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on* (pp. 1-4). IEEE.

Palani, S., Liong, S.-Y., Tkalich, P., Palanichamy, J., 2009. Development of a neural network model for dissolved oxygen in seawater. Indian J. Mar. Sci. 38, 151.

Palani, S., Liong, S. Y., Tkalich, P., & Palanichamy, J. (2009). Development of a neural network model for dissolved oxygen in seawater.

Ravansalar, M., Rajaee, T., & Ergil, M. A hybrid model of Wavelet with Artificial Neural Network for monthly prediction of river water quality.

Shrestha, R. R., & Nestmann, F. (2009). Physically based and data-driven models and propagation of input uncertainties in river flood prediction. *Journal of Hydrologic Engineering,* 14(12), 1309-1319.

Singh, K. P., Basant, A., Malik, A., & Jain, G. (2009). Artificial neural network modeling of the river water quality—a case study. *Ecological Modelling,* 220(6), 888-895.

Sudheer, K. P., Gosain, A. K., & Ramasastri, K. S. (2002). A data-driven algorithm for constructing artificial neural network rainfall-runoff models. *Hydrological processes*, 16(6), 1325-1330.

Thirumalaiah, K., & Deo, M. C. (2000). Hydrological forecasting using neural networks. *Journal of Hydrologic Engineering,* 5(2), 180-189.

Tryland, I., Myrmel, M., Østensvik, Ø., Wennberg, A. C., & Robertson, L. J. (2014). Impact of rainfall on the hygienic quality of blue mussels and water in urban areas in the Inner Oslofjord, Norway. *Marine pollution bulletin*, 85(1), 42-49.

Venkatesan, D., Kannan, K., & Saravanan, R. (2009). A genetic algorithm-based artificial neural network model for the optimization of machining processes. *Neural Computing and Applications*, 18(2), 135-140.

Welsh, W. D. (2008). Water balance modelling in Bowen, Queensland, and the ten iterative steps in model development and evaluation. *Environmental Modelling & Software,* 23(2), 195-205.

Yam, Y. F., & Chow, T. W. (1995). Determining initial weights of feedforward neural networks based on least squares method. *Neural Processing Letters*, 2(2), 13-17.

Yoon, K., & Kwek, S. (2007). A data reduction approach for resolving the imbalanced data issue in functional genomics. *Neural Computing and Applications*, 16(3), 295-306.

Zamani, A., Azimian, A., Heemink, A., & Solomatine, D. (2009). Wave height prediction at the Caspian Sea using a data-driven model and ensemble-based data assimilation methods. *Journal of Hydroinformatics*, 11(2), 154-164.

Zhang, G. P. (2007). A neural network ensemble method with jittered training data for time series forecasting. *Information Sciences*, 177(23), 5329-5346.

Zhou, Z. H., & Liu, X. Y. (2006). Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE *Transactions on Knowledge and Data Engineering*, 18(1), 63-77.

Zhou, Z. H., Wu, J., & Tang, W. (2002). Ensembling neural networks: many could be better than all. *Artificial intelligence,* 137(1-2), 239-263.