water

MDPI

Article

Estimating Design Floods at Ungauged Watersheds in South Korea Using Machine Learning Models

Jin-Young Lee [1], Changhyun Choi [2], Doosun Kang [3], Byung Sik Kim [4] and Tae-Woong Kim [5,*]

1 Department of Civil and Environmental Engineering, Hanyang University, Seoul 04763, Korea; hydrojy@hanyang.ac.kr
2 Risk Management Office, KB Claims Survey and Adjusting, Seoul 06212, Korea; karesma0cch@naver.com
3 Department of Civil Engineering, Kyung Hee University, Yongin 17104, Korea; doosunkang@khu.ac.kr
4 Graduate School of Disaster Prevention, Kangwon National University, Samcheok 25913, Korea; hydrokbs@kangwon.ac.kr
5 Department of Civil and Environmental Engineering, Hanyang University, Ansan 15588, Korea
* Correspondence: twkim72@hanyang.ac.kr; Tel.: +82-31-400-5184

Received: 12 September 2020; Accepted: 26 October 2020; Published: 28 October 2020

Abstract: With recent increases of heavy rainfall during the summer season, South Korea is hit by substantial flood damage every year. To reduce such flood damage and cope with flood disasters, it is necessary to reliably estimate design floods. Despite the ongoing efforts to develop practical design practice, it has been difficult to develop a standardized guideline due to the lack of hydrologic data, especially flood data. In fact, flood frequency analysis (FFA) is impractical for ungauged watersheds, and design rainfall–runoff analysis (DRRA) overestimates design floods. This study estimated the appropriate design floods at ungauged watersheds by combining the DRRA and watershed characteristics using machine learning methods, including decision tree, random forest, support vector machine, deep neural network, the Elman recurrent neural network, and the Jordan recurrent neural network. The proposed models were validated using K-fold cross-validation to reduce overfitting and were evaluated based on various error measures. Even though the DRRA overestimated the design floods by 160%, on average, for our study areas the proposed model using random forest reduced the errors and estimated design floods at 99% of the FFA, on average.

Keywords: design flood; machine learning; rainfall; ungauged watershed; random forest

1. Introduction

Recent climate change has brought about irregular rainfall patterns along with more frequent heavy rainfalls and guerilla rainstorms, resulting in colossal flood damage every year in South Korea. Design flood estimation is the most fundamental practice for predicting the flood magnitude to satisfy the design criteria of flood control structures and systems. There are two distinct methods for design flood estimation: flood frequency analysis (FFA) and design rainfall–runoff analysis (DRRA). FFA performs statistical frequency analysis of annual maximum flood data to calculate flood quantiles according to return periods. DRRA first performs statistical frequency analysis of annual maximum rainfall data to calculate rainfall quantiles according to return periods, and then transfers them to flood quantiles using a rainfall–runoff model [1].

In theory, FFA is the most appropriate method when abundant flow data are available. For example, the United States has issued and recommended an FFA-based guideline (Bulletin 17B) to estimate design floods for water resources design using abundant flow data [2]. However, due to the lack of flood data in many parts of South Korea, DRRA is practically always used in water resources design.

DRRA has the advantage that it can use relatively abundant rainfall data better than FFA. Therefore, many countries apply FFA for gauged watersheds and employ DRRA for ungauged watersheds [3,4]. Since the uncertainties and errors compound during estimating the time distribution of rainfall, the division of sub-watersheds, and the rainfall–runoff parameters, the flood quantiles calculated by DRRA also have great uncertainty and errors. To compensate for this, regional frequency analysis (RFA) is sometimes applied to supplement the frequency analysis method for ungauged watersheds, considering the watershed characteristics such as area, slope, and length [5–9]. However, to apply RFA for estimating design floods in South Korea, appropriate localization and flood estimation according to watershed characteristics are required. The application of RFA in South Korea is still under investigation to overcome the lack of a standard for regional classification and parameter estimation. Recently, as machine learning techniques (e.g., for image analysis and time series data processing) have been shown to improve predictive performance [10–13], many studies have also used them for predicting natural disasters [14–17].

Researchers, through introducing novel machine learning and hybridizing of the existing ones, have worked to discover more accurate and efficient prediction models [18]. From hydraulic and hydrological perspectives, machine learning techniques are used to predict storm rainfall, flood discharge, and river water level. In addition, since the predicted results are time series, results such as neural network are excellent in single machine learning [19,20], and prediction performance is often evaluated using hybrid machine learning [21,22].

In this study we developed a machine learning-based model for estimating design floods for ungauged watersheds, and we increased the applicability of the model through updating the DRRA's estimates. The predicted type was the flood by frequency, and the applicability of the machine learning technique was evaluated. As shown in Figure 1, in step 1 we compared design floods estimated by FFA and DRRA and identified that DRRA overestimated the design floods. In step 2 we employed machine learning techniques to adjust the DRRA's design floods to reduce the overestimates using specific watershed characteristics. Finally, to verify the applicability to ungauged watersheds, we used K-fold cross-validation to validate the models, and we found the best prediction model. More details are provided in the following sections.

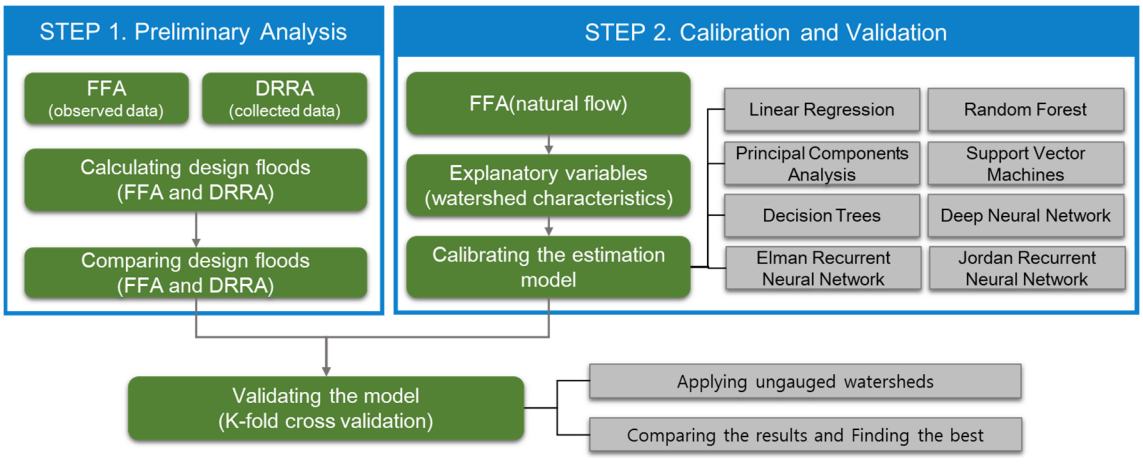

Figure 1. Flow chart of the study.

2. Materials and Methods

2.1. Study Area

To avoid the controversial issues associated with DRRA, such as the time distribution of design rainfall, the division of sub-watersheds, and the parameter estimation of the rainfall–runoff model, we collected pre-determined design floods for 64 mid-sized watersheds in South Korea, as shown in Figure 2: 18 in the Han River basin, 25 in the Nakdong River basin, 10 in the Gum River basin, 7 in the

Seomjin River basin, and 4 in the Youngsan River basin, which are represented by yellow polygons. The collected design floods had been calculated by DRRA for comprehensive flood mitigation planning in the large river basins in South Korea. To compare the design floods, we estimated the design floods by FFA after collecting the annual maximum flows at the outlets of the watersheds corresponding to the selected 64 watersheds, which are indicated by red dots in Figure 2. Ministry of Land, Infrastructure and Transport (MOLIT) estimated the natural flows for mid-sized basins during 1966–2003 to develop a national comprehensive water resources management plan [23]. The parameters of the tank model have been calibrated and verified through Nash-Sutcliffe Efficiency (NSE) and Ratio of Volume (ROV). The estimated natural flows have been evaluated and approved by the Korea Meteorological Administration (KMA). The natural flow provided is mid-sized.

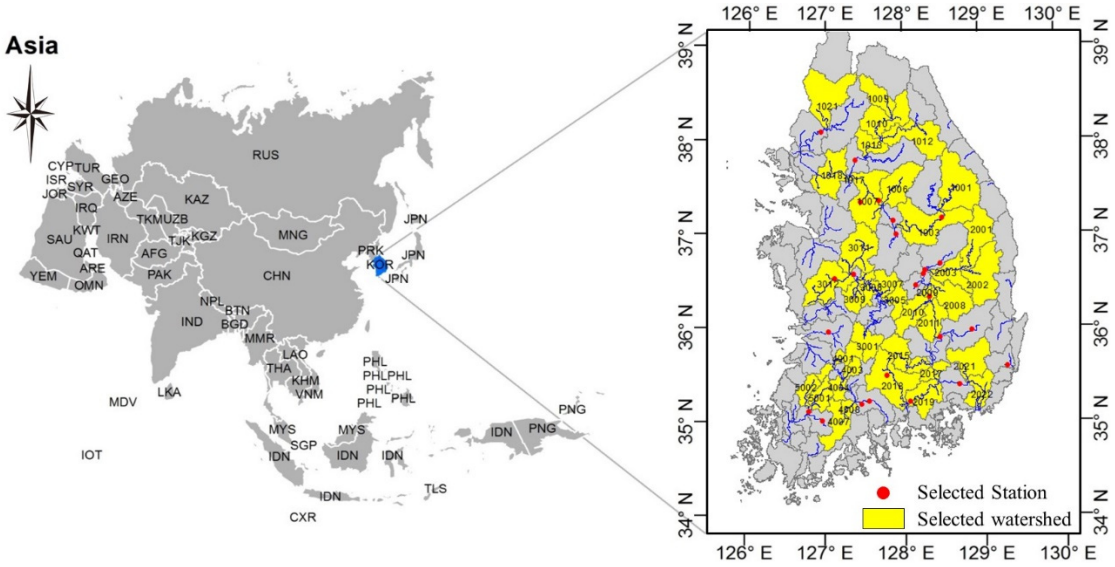

Figure 2. Study area.

2.2. Prediction Models

2.2.1. Linear Regression Model

The linear regression (LR) model assumes a linear relation between the dependent variable y and n independent variables, $x_1, x_2, \cdots, x_n$.

$$y_i = \alpha_0 + \alpha_1 x_{1i} + \alpha_2 x_{2i} + \cdots + \alpha_n x_{ni} + \varepsilon_i \quad (1)$$

where the subscript *i* means the i_{th} data. α_0 is for intercept, $\alpha_1, \alpha_2, \cdots, \alpha_n$ are regression coefficients for the independent variables, and ε is the error. When independent variables are highly correlated, multicollinearity problems can distort the estimation of the individual regression coefficients. For the trained regression model, k number of variance inflation factors (VIFs) can be obtained. VIFs are measures of the linear correlation between x_j (*j*-th variable) and x_{-j} (remainder) for $j = 1, 2, \cdots, i$. VIF values above 10 indicate multicollinearity, which is a strong linearity among independent variables.

2.2.2. Principal Component Analysis (PCA)

If the number of explanatory variables is too large or the correlation between the variables is too high, the predictive power may decline. In this case, a method of reducing the number of variables may be considered with PCA, which extracts only the significant principal components from high-dimensional data by calculating a covariance matrix or a correlation matrix. Since a small number of principal components derived through PCA include most of the variation of the explanatory variables, it is possible to use a small number of principal components that actually hold most of the information of the explanatory variables.

2.2.3. Decision Tree

The decision tree (DT) model expresses data with a tree-like graph based on the rules or conditional statements of the variables; subdivides them into similar data types by separation rules; and continues this classification until the final classification criteria are satisfied [24]. In a DT, through a binary recursive partitioning process, split variables and split points that minimize the mean squared error (MSE) are identified in each step, iterating the process of recursive partitioning and eventually forming the shape of an entire tree. In addition, "pruning" is performed to determine the size of the tree that minimizes the MSE using cross-validation to prevent overfitting [25]. In this study, we employed R-Studio's "rpart" library. Figure 3 shows a conceptual diagram of a DT.

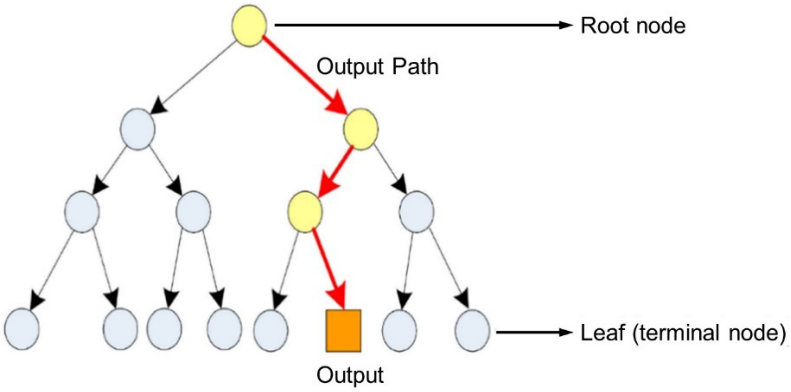

Figure 3. Conceptual diagram of a decision tree (DT).

2.2.4. Random Forest (RF)

Random forest (RF) is a model with a number of (decision) trees, and is a method of increasing the predictive power by generating a number of DT models. In a DT, the training data are generated once, are trained, and make a prediction with one DT model. However, the distinctive characteristic of RF is that multiple training data sets are generated from the data set, and multiple DTs are generated through multiple trainings; with the combination of the results, the predictive power is enhanced [25]. The DT generates and trains one set of training data from one data set and predicts it as a single DT model, but RF creates multiple sets of training data from one data set. It has improved predictive power as a result of creating multiple DTs through multiple learnings and then combining multiple DTs. In addition, in the case of continuous criterion variables, when the number of explanatory variables is m, a tree is constructed by selecting m/3 variables at random at each splitting [26]. In this study, we employed R-Studio's "randomForest" library. Figure 4 shows the conceptual diagram of RF.

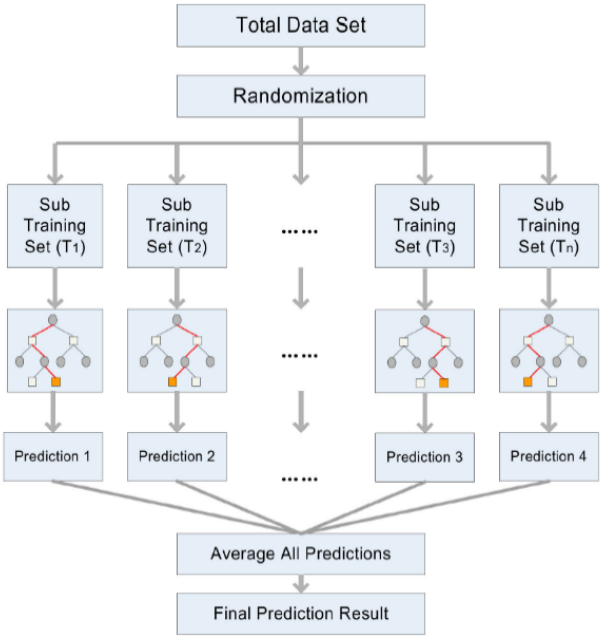

Figure 4. Conceptual diagram of the random forest (RF).

2.2.5. Support Vector Machine (SVM)

Support vector machine (SVM) can be applied to various types of data with very accurate classification results in various applications, such as document classification and customer classification [27]. SVM finds hyperplanes of support vectors that can perform classification with a maximized margin for the distance between the two vectors of linearly different classes. For data that cannot be classified linearly, they are first mapped onto a high-dimensional space in a plane using a kernel function. Representative kernel functions include polynomial, sigmoid, and radial basis function (RBF). SVM is mainly used for prediction of classification problems, and an extended method for use in regression analysis by introducing ε-insensitive loss function to SVM is called supported vector regression (SVR). In this study, we employed R-Studio's "e1071" library. Figure 5 shows the conceptual diagram of SVM.

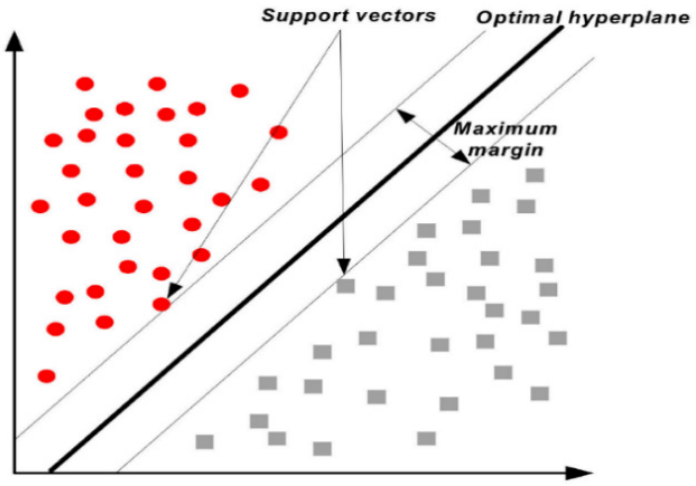

Figure 5. Conceptual diagram of the support vector machine (SVM).

2.2.6. Deep Neural Network (DNN)

A deep neural network (DNN) is an artificial neural network consisting of multiple hidden layers between an input layer and an output layer. In general, as the number of hidden layers increases and the deeper the neural network, various complex features and abstracted properties can be extracted and utilized. Thus, a DNN can model a complex non-linear relationship [28]. In this study, we employed R-Studio's "neuralnet" library. Figure 6 shows the conceptual diagram of a DNN.

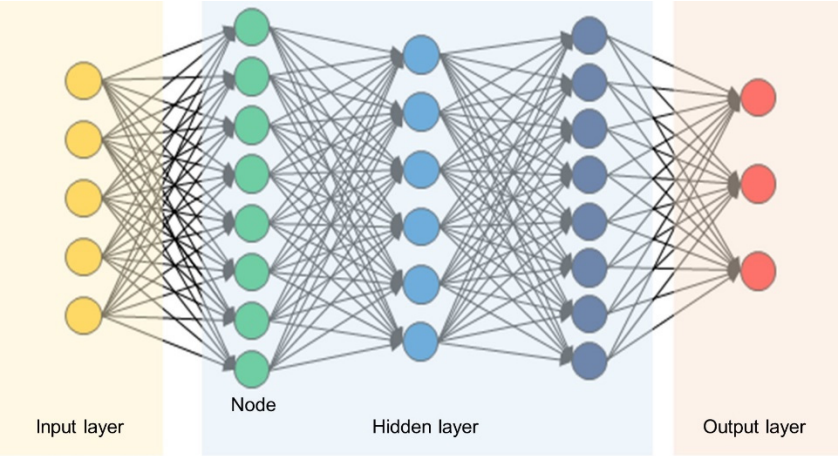

Figure 6. Conceptual diagram of a deep neural network (DNN). Circles represent nodes in each layer.

2.2.7. Recurrent Neural Network

A recurrent neural network (RNN) is an artificial neural network that has dynamic characteristics without using a buffer; this is accomplished by adding a feedback connection to the backpropagation model. The method of forming a feedback connection in a neural network may consider full connections between all of the neurons, but generally only some of the neurons form a feedback connection. Therefore, static neurons and dynamic neurons exist at the same time in an RNN. Static neurons do not have the memory of the activation history, but dynamic neurons do.

An Elman neural network consists of an input layer, a hidden layer, a context layer, and an output layer [29]. The context layer serves to store the past state of the hidden layer, and in the Elman recurrent neural network (ERNN), the value of the context layer is fed to the hidden layer with the input data of the next step, and with the feedback method of reinputting the past output, the network has the capability of memory. In this study, we employed R-Studio's "RSNNS".

A Jordan recurrent neural network (JRNN) is particularly suitable when a series of continuous outputs is important [30,31]. The difference between the JRNN and the ERNN is that the JRNN is connected to the context layer. The JRNN has a limitation in that it can construct only one hidden layer in its architecture. In this study, we employed R-Studio's "JNN" library. Figure 7 shows the conceptual diagram of the ERNN and JRNN.

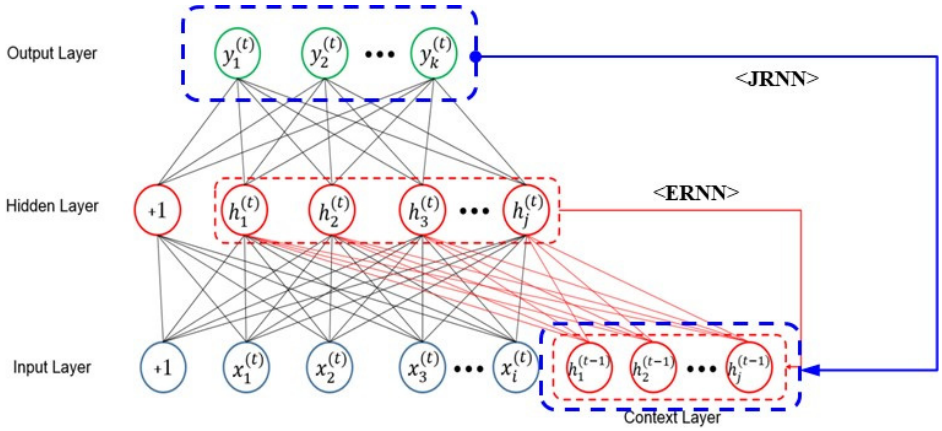

Figure 7. Conceptual diagram of the Elman recurrent neural network (ERNN) and Jordan recurrent neural network (JRNN).

2.3. Validation and Performance Assessment

The purpose of this study was to develop a design flood prediction model for ungauged watersheds. To do this, we selected 64 gauged watersheds and calculated the design floods using FFA, which were assumed to be a target variable, because FFA is the most appropriate statistical theory-based method when abundant flow data are available, as discussed in the Introduction. Thus, the prediction model needs to be developed to match the design floods calculated by FFA. Considering the lack of sufficient observed flood data for ungauged watersheds, in this study the prediction model was validated through K-fold cross-validation.

K-fold cross-validation is one way to evaluate a model in statistics, using a portion of the entire data (called the held-out validation) as a validation set to assess model performance. In general, if a data set is small in size, the reliability of the performance assessment for the test set will be compromised. If the performance depends on how the researcher defines the test set, the coincidental effect will bias the model assessment index. In this study, each model was developed by dividing the training set and test set into five, and the optimal model was selected after extracting the error values for the developed model, as shown in Figure 8.

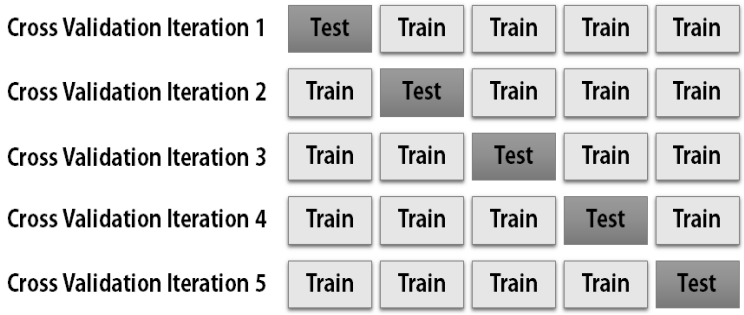

Figure 8. Concept of K-fold cross-validation.

As predictive performance evaluation indicators for the design flood estimation model, the normalized root mean squared error (NRMSE), the mean absolute error (MAE), the root mean squared log error (RMSLE), the mean absolute percentage error (MAPE) were used in this study. The NRMSE is a normalized version of the root mean squared error (RMSE) and the RMSLE is the log-applied RMSE. Because each indicator has a penalty for underestimation or overestimation, when we evaluated the predictive performance of all the models, the model with the best predictive performance was selected. Because all the performance indicators indicate that as their values approach 0 (%), the difference between the actual value and the predicted value is small, and these are represented in Equations (2)–(5):

$$\text{NRMSE}\ (\%) = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}}{\text{Max}\left(y_i\right) - \text{Min}\left(\hat{y}_i\right)} \times 100 \tag{2}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{3}$$

$$\text{RMSLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\log\left(y_i + 1\right) - \log\left(\hat{y}_i + 1\right)\right)^2} \tag{4}$$

$$\text{MAPE}\ (\%) = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{5}$$

where n is the number of data, y_i is the target value which is the design flood estimated by FFA, and $\hat{y}_i$ is the predicted value which is the design flood calculated by the machine learning-based model in this study.

3. Results and Discussion

3.1. Preliminary Analysis (Step 1)

This study developed machine learning-based design flood prediction models for ungauged watersheds using specific watershed characteristics. To apply the machine learning techniques, it is necessary to secure as much training data as possible. We therefore collected design flood data of 64 watersheds according to 30-, 50-, 80-, 100-, and 200-year of return periods, which were design floods calculated by DRRA (hereafter referred to as DF-DRRA), as explained in Section 2. We calculated the design floods using FFA after collecting the annual maximum flows at the outlets of the watersheds corresponding to the selected 64 watersheds (hereafter referred to as DF-FFA), which were the target values. We also calculated the design floods using FFA after collecting the natural flow data simulated by a tank model (hereafter referred to as NF-FFA), which were further used as complementary data at ungauged watersheds to improve the model performance.

Before a statistical frequency analysis is performed, the flow data should satisfy the stationarity and independence criteria. If there are autocorrelations or trends in the data, they should be removed and then the frequency analysis can be performed. We performed preliminary tests, including the run test, turning point test, Anderson exact test, Mann-Kendall test, Hostelling-Pabst test, and Wald-Wolfowitz test for (observed and simulated) annual maximum flow data, and showed that all the data sets had sufficient randomness and did not have any trends. We selected optimal distributions for individual watersheds based on the Kolmogorov-Smirnov goodness-of-fit test after estimating the distribution parameters using the probability-weighted moments method for normal, lognormal, generalized extreme value, Gumbel, Weibull, generalized Pareto, and gamma distributions.

Compared to the DF-DRRA, the DF-FFA of the Han River basin were estimated to be in the range between 25.2% (#1009) and 175.2% (#1006) (72.5% on average), while those of the Nakdong River basin were 21.3% (#2003) to 197.5% (#2022) (71.9% on average), the Gum River basin were 33.1% (#3012) to 53.8% (#3008) (41.4% on average), and the Seomjin–Youngsan River basin were 18.7% (#4008) to 81.2% (#4004) (46.6% on average). When only the averages for each region were compared, the DF-FFA were smaller than DF-DRRA in most basins, which was in accordance with previous studies.

The notable results that the DF-FFA were larger than the DF-DRRA were shown in three watersheds of the Han River basin (#1001, #1002, #1003) and five watersheds of the Nakdong River basin (#2005, #2011, #2014, #2017, #2022). These watersheds seemed to be affected by outflow of directly-upstream dams, such as the Peace Dam, the Imha Dam, the Hapcheon Dam, the Namgang Dam, and the Juam Dam. In addition, each watershed is located at the mouth of a river, such as the estuary bank of the

Nakdong River. It seems that the FFA over-estimated the design floods due to the lack of accurate observations due to the effects of the estuary bank and the tidal-water level. Figures 9–12 compare the 100-year floods for the Han River, the Nakdong River, the Gum River, the Seomjin River and the Youngsan River, respectively.

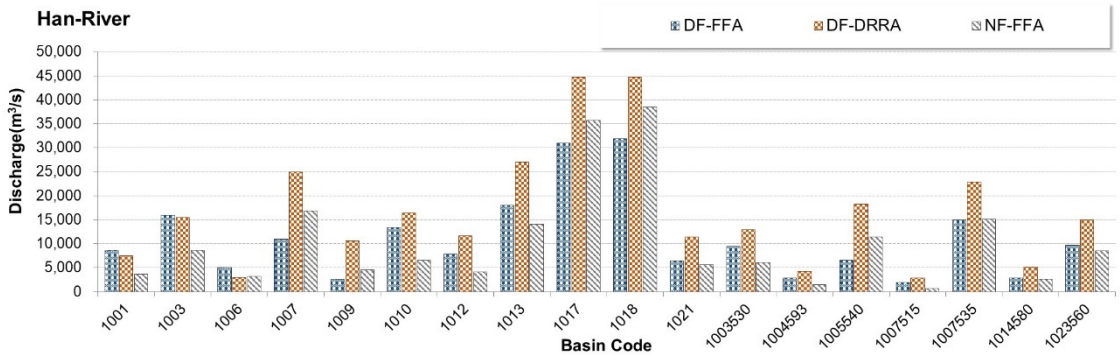

Figure 9. The 100-year design floods of Han River basin.

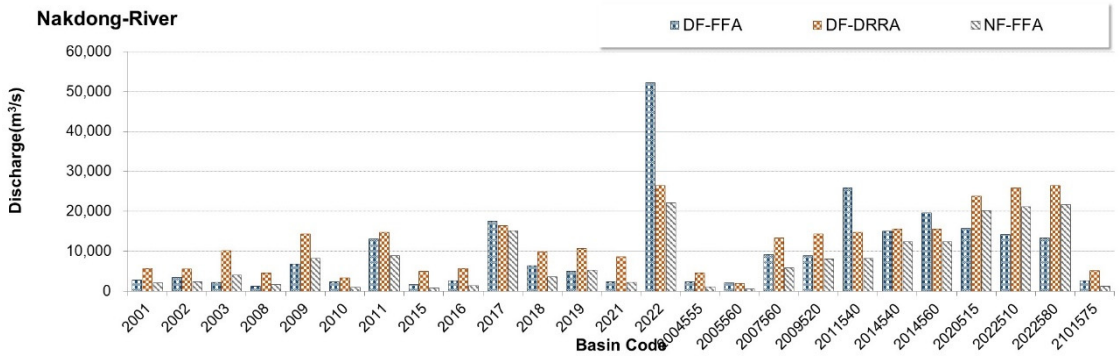

Figure 10. The 100-year design floods of Nakdong River basin.

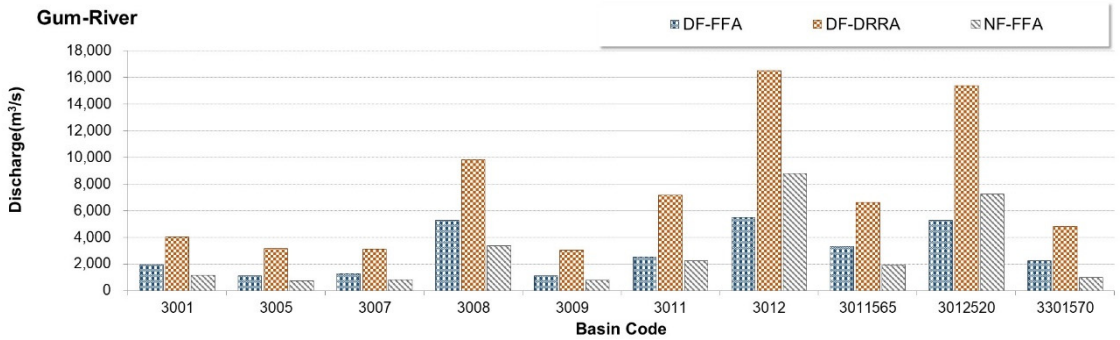

Figure 11. The 100-year design floods of Gum River basin.

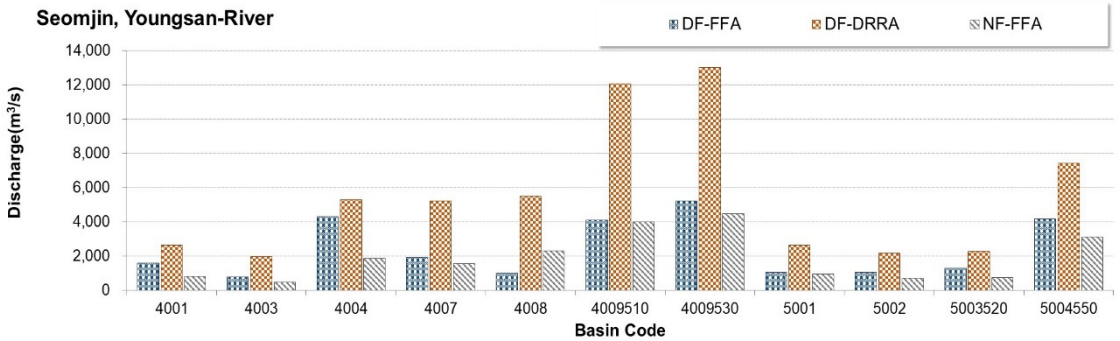

Figure 12. The 100-year design floods of Seomjin–Youngsan River basin.

3.2. Calibration of Machine Learning-Based Models (Step 2)

3.2.1. Target and Explanatory Variables

The machine learning-based design flood prediction model was developed using the DF-DRRA, the NF-FFA, and watershed topographic factors as explanatory variables, and the DF-FFA as the target variable. Table 1 lists the explanatory variables. All variables were normalized to eliminate the effect of variable scale. Since the explanatory variables in the dataset for 64 stations were variables that can be estimated in ungauged watersheds, DF-FFA can be applied to ungauged watersheds. South Korea generally provides a digital topographic map for the whole country (http://www.nsdi.go.kr/). For example, when trying to apply the results of this study to an ungauged watershed, values from x3 to x8 can be calculated using a digital topographic map. In addition, DF-DRRA can be calculated just as the flood quantiles in the unmeasured watershed using rainfall data and rainfall–runoff models. Since NF is obtained from a tank model, NF-FFA can be also applied to the ungauged watershed. Therefore, the objective variable (DF-FFA) can be applied to the ungauged watershed.

Table 1. List of explanatory variables.

x1	DF-DRRA	x2	NF-FFA
x3	Watershed area	x4	River length
x5	Watershed circumference	x6	Mean watershed width
x7	Shape factor	x8	Maximum elevation difference
x9	Return period	-	-

The target set was composed of 320 DF-FFAs (5 return periods for 64 watersheds) and the corresponding explanatory data sets were composed of nine explanatory variables, as shown in Table 1. The 320 data sets were divided into two subsets: 256 training sets and 64 validating sets. This study employed machine learning techniques, as described in Section 2. The total number of available models was 14, depending on the PCA application (2 cases (PCA and no PCA) × 7 machine learning techniques (LR, DT, RF, SVM, DNN, ERNN, and JRNN)).

3.2.2. Design Flood Estimation with PCA and LR

To fully include the information of the explanatory variables, this study determined the number of principal components when the cumulative variance became no less than 90%. The two principal components accounted for more than 90% of the cumulative variance, so the final two principal components were selected.

The process of selecting explanatory variables is very important in developing an LR model with good forecasting performance, since numerous LR models are possible according to the combination of explanatory variables. The stepwise selection method was used in this study. Although the explanatory variables selected using the stepwise selection procedure were different according to cross-validation, it was confirmed that all of the LR models included the DF-DRRA (×1), watershed area (×3), river length (×4), and watershed circumference (×5).

If there is a high correlation between the explanatory variables, the regression coefficients cannot be accurately estimated, so the multicollinearity of the explanatory variables that form the developed linear regression model should be assessed once the variables are selected. In general, a variance inflation factor (VIF) of 10 or more indicates high multicollinearity [32]. Since LR models using explanatory variables had VIFs ≥ 10, they were excluded from the next analyses. In contrast, the VIFs of the LR model using two PCAs were 1.0, so there were no problems with multicollinearity and they were used for the next analyses.

3.2.3. Design Flood Estimation with Machine Learnings

DT repeatedly performs the process of bifurcating data by finding a branch point that minimizes the MSE. Because more bifurcation of data is likely to cause overfitting, cross-validation should be used to determine the size of the tree that minimizes errors. This study calculated the mean error by a complexity parameter (CP) expressing the size of the tree through 10-fold cross-validation. The pruning was carried out by finding the CP value with the lowest mean error, which determined the optimal DT model. When selecting the CP, the branch with the lowest error rate after the branching is appropriate, but if overfitting occurs it can be adjusted with the next value to estimate the CP.

The DT was constructed using the principal components derived from the PCA. The mean error was calculated for each CP. As with the general decision-making tree, the minimum CP was selected. However, if overfitting occurred it was adjusted with the next value to estimate the CP. In the DT, before pruning in the case of Iteration 1, prediction was made through the watershed circumference (×5), watershed area (×3), and NF-FFA (×2). In the DT after pruning, it was predicted using the basin circumference (×5) and watershed area (×3).

RF constructs trees by selecting m/3 exploratory variables randomly in each division if the criterion variables are continuous and the number of exploratory variables is m [29]. This study determined the optimal RF model by finding the number of trees that minimized the MSE. Watershed area (×3) was determined to be the most important variable in design flood estimation in all the iterations. PCA 1 was determined to be an important variable in estimating the design flood.

In this study, the SVM was converted into the SVR to predict the residuals, similar to a regression problem. For the kernel function we used the radial basis function (RBF) known to be relatively accurate for regression problems. The SVM selected the optimal parameter by adjusting the values of the cost and the value of ε of the ε-insensitive loss function, which determines generalization, and developed a model that reflects them. This study selected the optimal parameter with the adjustment of a cost range of 2^0 to 2^7 and a range of ε, 0 to 1 at 0.1 intervals based on the 10-fold cross-validation and referring to earlier researches [33]. For SVMs using the PCA, the optimal parameter conditions for minimizing errors were applied. The cost of Iteration 1 of the SVM was determined to be 128 and ε was 0.3, and the PCA results were found to be 4 and 0.3.

DNNs can extract a number of complex characteristics and abstracted properties as the number of hidden layers increases and the neural networks deepen. However, overly complex configurations tend to fall into local minimums or lead to increased computation processes. A DNN can determine the number of hidden layers and nodes. We constructed the optimal numbers by changing the number of hidden layers from two to five and the number of nodes from two to five. Compared with the observed data, the optimal results were obtained when there were four or five hidden layers and four nodes. Therefore, we decided to use four hidden layers and four nodes to assess the model in this study.

For weighted learning in the ERNN and JRNN, the weight was updated to minimize the loss function by using the gradient descent method applied with the backpropagation method, and 1000 iterations were performed to minimize the error. The relationship between the criterion variables and the three principal components was estimated by constructing the ERNN and the JRNN using the PCA.

3.3. Validation of Machine Learning-Based Models (Step 2)

To develop the machine learning-based design flood estimation model, this study used 80% of the total data set for the model development and 20% for the model validation. The total data set was used to assess each model using a 5-fold technique, and the remaining 20% of the test set was used to validate the model. In other words, 256 of the total 320 data items comprised the training sets and 64 comprised the test sets, and the applicability of the developed model was assessed randomly using K-fold cross-validation.

NRMSE, MAE, RMSLE, and MAPE were used as performance indicators. For all the indicators, when the result values approached 0 (%), the difference between the DF-FFA and the estimated values derived by the machine learning-based model were smaller.

Five iterations were developed for each machine learning using K-fold cross-validation for model performance assessment, and the RF showed the best performance compared with other machine learnings, as shown in Figure 13. Figure 13 shows a scatterplot of the normalized DF-FFA (target values) and the estimated values. The values above the red line indicate the overestimated values (the estimated values are larger than the target values), while the values below the red line indicate the underestimated values (the estimated values are smaller than the target values).

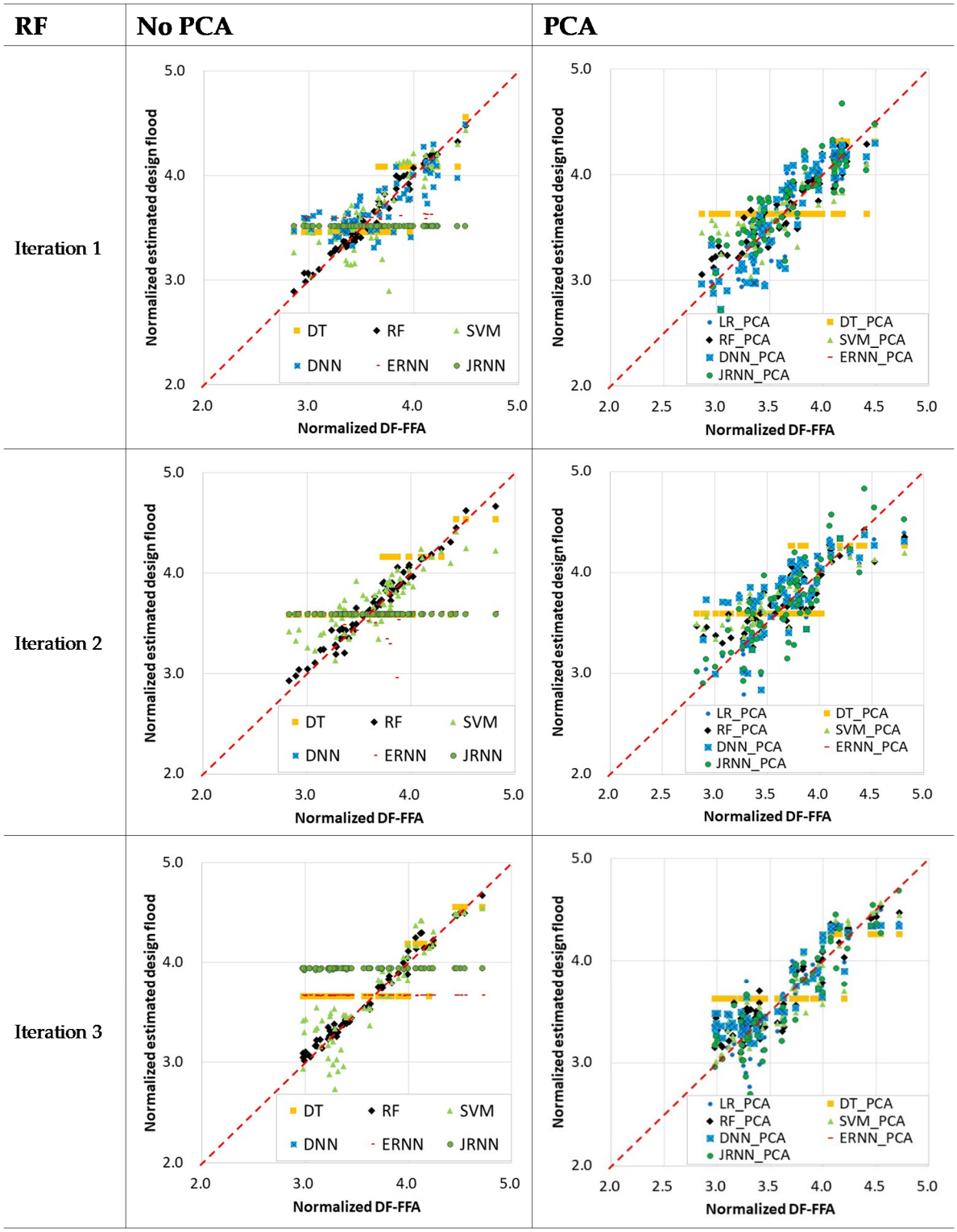

Figure 13. Performance assessment of design flood estimation models.

The performance assessment of RF is shown in Table 2. Iteration 3 showed the best performance according to NRMSE, Iteration 4 showed the best forecasting performance according to MAE, and Iteration 1 showed the best performance according to RMSLE and MAPE. In the case of Iteration 3, the performance was analyzed as the third best according to MAE and the second best according to RMSLE and MAPE. Iteration 4 is the third best according to NRMSE and the fourth out of five according to RMSLE and MAPE, whereas Iteration 1 was analyzed as having the second best performance according to RMSLE and MAE. Therefore, we selected Iteration 1 as the design flood estimation model in this study. The accuracy of DF-DRRA was 4.12 of NRMSE and 0.79 of MAPE. Therefore, we determined that the RF model developed in this study was superior.

Table 2. Performance assessment of the RF model.

RF	NRMSE	MAE	RMSLE	MAPE
Iteration 1	3.63	702.77	0.06	0.11
Iteration 2	12.31	1243.03	0.08	0.17
Iteration 3	3.40	841.26	0.06	0.11
Iteration 4	3.95	642.61	0.08	0.14
Iteration 5	4.81	1249.81	0.07	0.12

4. Conclusions

Despite many previous attempts to develop design flood estimation methods for South Korea watersheds, it has been difficult to develop standardized methods that can be applied in practice because of the lack of reliable flood data. This study aimed to develop a machine learning-based design flood estimation model considering the watershed characteristics to supplement the DRRA method for ungauged watersheds in South Korea.

As with the prior studies, we found that DRRA over-estimates; the design floods estimated by FFA were smaller than those by DRRA: an average of 27.5% smaller in the Han River basin, 28.1% smaller in the Nakdong River basin, 58.6% smaller in the Gum River basin, and 53.4% smaller in the Seomjin, Youngsan River basin. Kim et al. [1] found even lower FFA values: the Han River basin was estimated to be as small as 27.71%, the Nakdong River basin by 15.1%, and the Gum River basin by 33.1%, because there was a large difference in the number of basins applied in their study and this one.

Our technique in developing the design flood prediction model basically applied LR and machine learning techniques (DT, RF, SVM, DNN, ERNN, and JRNN) depending on the application of the PCA. Performance assessment using NRMSE, MAE, RMSLE, and MAPE showed that RF had the best prediction performance among all the models, and its Iteration 1 was the best model when compared to the others. Because DRRA is used as an explanatory variable, its applicability in practice is excellent. The comparatively better generalization ability of machine learning makes it attractive to predict design floods of ungauged watersheds for future flood risk assessment. Robust and accurate prediction contribute highly to comprehensive analyses, water recourse management strategies, and policy suggestions.

Author Contributions: Conceptualization, T.-W.K.; Methodology, C.C.; Validation, D.K. and B.S.K.; Formal Analysis, J.-Y.L.; Investigation, D.K. and B.S.K.; Resources, C.C.; Data Curation, J.-Y.L.; Writing—Original Draft Preparation, J.-Y.L.; Writing—Review and Editing, T.-W.K.; Visualization, C.C.; Supervision, T.-W.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a grant (2020-MOIS33-006) from the Lower-level and Core Disaster-Safety Technology Development Program funded by the Ministry of Interior and Safety (MOIS, Korea).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, N.W.; Lee, J.-Y.; Park, D.H.; Kim, T.-W. Evaluation of future flood risk according to RCP scenarios using a regional flood frequency analysis for ungauged watersheds. *Water* **2019**, *11*, 992. [CrossRef]
2. Flynn, K.M.; Kirby, W.H.; Hummel, P.R. *User's Manual for Program. PeakFQ, Annual Flood-Frequency Analysis Using Bulletin 17B Guidelines*; U.S. Geological Survey: Reston, VA, USA, 2006.
3. Institute of Hydrology. *Flood Estimation Handbook*; Institute of Hydrology: Wallingford, UK, 1999.
4. Centre for Ecology & Hydrology. *European Procedures for Flood Frequency Estimation*; European Cooperation in Science and Technology: Lancaster, UK, 2012.
5. Burnham, M.W. *Adoption of Flood Flow Frequency Estimates at Ungaged Location, Training Document 11*; US Army Corps of Engineers: Institute for Water Resources: Davis, CA, USA, 1980.
6. Cunnane, C. *Statistical Distributions for Flood Frequency Analysis*; World Meteorological Organization Operational: Geneva, Switzerland, 1989; Volume 33, p. 718.
7. Hosking, J.R.M.; Wallis, J.R. Some statistics useful in regional frequency analysis. *Water Resour. Res.* **1993**, *29*, 271–281. [CrossRef]
8. Potter, K.W.; Lattenmaier, D.P. A comparison of regional flood frequency estimation methods using are sampling method. *Water Resour. Res.* **1990**, *26*, 415–424. [CrossRef]
9. Stedinger, J.R.; Tasker, G.D. Regional hydrologic analysis 1. Ordinary, weighted and generalized least squares compared. *Water Resour. Res.* **1985**, *21*, 1421–1432. [CrossRef]
10. Tong, S.; Chang, E. Support vector machine active learning for image retrieval. In Proceedings of the Ninth ACM International Conference on Multimedia, Ottawa, ON, Canada, 1 October 2001; pp. 107–118.
11. Ahmed, N.K.; Atiya, A.F.; Gayar, N.E.; El-Shishiny, H. An empirical comparison of machine learning models for time series forecasting. *Econom. Rev.* **2010**, *29*, 594–621. [CrossRef]
12. Ak, R.; Fink, O.; Zio, E. Two machine learning approaches for short-term wind speed time-series prediction. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *27*, 1734–1747. [CrossRef] [PubMed]
13. Qu, Y.; Qian, X.; Song, H.; Xing, Y.; Li, Z.; Tan, J. Soil moisture investigation utilizing machine learning approach based experimental data and Landsat5-TM images: A case study in the Mega City Beijing. *Water* **2018**, *10*, 423. [CrossRef]
14. Randall, M.; Fensholt, R.; Zhang, Y.; Bergen Jensen, M. Geographic object based image analysis of WorldView-3 Imagery for Urban Hydrologic Modelling at the catchment scale. *Water* **2019**, *11*, 1133. [CrossRef]
15. Marjanović, M.; Kovačević, M.; Bajat, B.; Voženílek, V. Landslide susceptibility assessment using SVM machine learning algorithm. *Eng. Geol.* **2011**, *123*, 225–234. [CrossRef]
16. Goetz, J.N.; Brenning, A.; Petschko, H.; Leopold, P. Evaluating machine learning and statistical prediction techniques for landslide susceptibility modeling. *Comput. Geosci.* **2015**, *81*, 1–11. [CrossRef]
17. Choubin, B.; Borji, M.; Mosavi, A.; Sajedi-Hosseini, F.; Singh, V.P.; Shamshirband, S. Snow avalanche hazard prediction using machine learning methods. *J. Hydrol.* **2019**, *577*, 123929. [CrossRef]
18. Mosavi, A.; Ozturk, P.; Chau, K.W. Flood prediction using machine learning models: Literature review. *Water* **2018**, *10*, 1536. [CrossRef]
19. Chang, F.-J.; Chen, P.-A.; Lu, Y.-R.; Huang, E.; Chang, K.-Y. Real-time multi-step-ahead water level forecasting by recurrent neural networks for urban flood control. *J. Hydrol.* **2014**, *517*, 836–846. [CrossRef]
20. Yu, P.-S.; Yang, T.-C.; Chen, S.-Y.; Kuo, C.-M.; Tseng, H.-W. Comparison of random forests and support vector machine for real-time radar-derived rainfall forecasting. *J. Hydrol.* **2017**, *552*, 92–104. [CrossRef]
21. Zhou, Y.; Guo, S.; Chang, F.J. Explore an evolutionary recurrent ANFIS for modelling multi-step-ahead flood forecasts. *J. Hydrol.* **2019**, *570*, 343–355. [CrossRef]
22. Kao, I.F.; Zhou, Y.; Chang, L.C.; Chang, F.-J. Exploring a long short-term memory based encoder-decoder framework for multi-step-ahead flood forecasting. *J. Hydrol.* **2020**, *583*, 124631. [CrossRef]
23. Ministry of Land, Infrastructure and Transport (MOLIT). *Long-Term Comprehensive Plan for Water Resources*; MOLIT: Seoul, Korea, 2006.
24. Breiman, L.; Friedman, J.; Stone, C.J.; Olshen, R.A. *Classification and Regression Trees*; CRC Press: Boca Raton, FL, USA, 1984.
25. Choi, C.; Kim, J.; Kim, J.; Kim, H.S. Development of combined heavy rain damage prediction models with machine learning. *Water* **2019**, *11*, 2516. [CrossRef]
26. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]

27. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1995.
28. Wiley, J.F.R. *Deep Learning Essentials*; Packt Publishing Ltd.: Birmingham, UK, 2016.
29. Elman, J.L. Finding Structure in Time. *Cogn. Sci.* **1990**, *14*, 179–211. [CrossRef]
30. Jordan, M.I. *A Parallel Distributed Processing Approach*; Tech. Rep. No. 8604; University of California, Institute for Cognitive Science: San Diego, CA, USA, 1986.
31. Jordan, M.I.; Rosenbaum, D.A. *Action Technology*; Rep. No. 8826; University of Massachusetts, Department of Computer Science: Amherst, MA, USA, 1988.
32. Marquardt, D.W. Generalized inverses, ridge regression, biased linear estimation, and nonlinear estimation. *Technometrics* **1973**, *12*, 591–612. [CrossRef]
33. Tay, F.E.; Cao, L. Application of support vector machines in financial time series forecasting. *Omega* **2001**, *29*, 309–317. [CrossRef]