JURNAL INFOTEL
Informatics - Telecommunication - Electronics
Website: http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN: 2085-3688; e-ISSN: 2460-0997

Prediction model with artificial neural network for tidal flood events in the coastal area of Bandar Lampung City

Eka Suci Puspita Wulandari[1,*], Ramadhan Nurpambudi[2], RZ. Abdul Aziz[3]
[1,2,3]Darmajaya Lampung Institute of Informatics and Business
[1,2]Meteorological, Climatological, and Geophysical Agency of Lampung
[1,2,3]ZA. Pagar Alam Street, No. 93, Rajabasa District, Lampung 35141, Indonesia
[1,2]Radin Inten II Airport, Branti Raya Street, Natar District, Lampung 35362, Indonesia
[*]Corresponding email: ekasucipw.2121211011@mail.darmajaya.ac.id

Received 2 January 2023, Revised 30 January 2023, Accepted 30 January 2023

Abstract — The fastest sea level rise began in 2013 and reached its highest level in 2021. This condition is part of the ongoing global warming impact, where polar ice and glaciers also continue to melt, causing sea level rise. In the Bandar Lampung City area, several areas are threatened by tidal flooding, namely Karang City Village and Kangkung Village, Bumi Waras Village, and Sukaraja Village. Bandar Lampung itself is the city center in the coastal area where the majority of the population is in the Coastal area. So that rising sea levels cause the threat of tidal flooding. This research proposes to study the occurrence of tidal floods in the past. This research uses an Artificial Neural Network, which can study non-linear data, which is then carried out by training and testing until the best configuration model is obtained. Based on the conducted analysis and discussion, several significant points can be inferred. These include the ratios of 80:20 and 90:10, which were utilized. The effectiveness of these ratios is evident through the model's high accuracy in configuration and prediction of tidal flood events, accurately representing real-world conditions. The experiment model configuration can be set to produce the best training accuracy value reaching 100 %, while the best testing accuracy is 88 %. The average correlation value of training with the 50:50 dataset is 0.975, the 60:40 dataset is 0.975, the 70:30 dataset is 0.951, the 80:20 dataset is 0.935, and the 90:10 dataset is 0.929. For the average value of the correlation test with the 50:50 dataset of 0.514, the 60:40 dataset is 0.362, the 70:30 dataset is 0.488, the 80:20 dataset is 0.284, and the 90:10 dataset is 0.402. Whereas the average error value for the 50:50 dataset is 0.006, the 60:40 dataset is 0.006, the 70:30 dataset is 0.010, the 80:20 dataset is 0.007, and the 90:10 dataset is 0.007, the tidal flood prediction is made based on one configuration the best with a training accuracy rate of 98 % and a testing accuracy of 80 % with an error value of 0.004, namely configuration model 14, this model is the best configuration model out of 3 dataset divisions out of a total of 5. The tidal flood prediction uses sea level tides of 1.5 m. The prediction results for tidal floods are very good, especially when active astronomical phenomena occur. The results of this excellent prediction of tidal floods illustrate that Artificial Neural Network backpropagation can study datasets well and can be used by Meteorical, Climatological, and Geophysical Agency forecasters in making early warnings of tidal floods.

Keywords – artificial neural network, early warning, prediction, tidal flood

I. Introduction

Changes in climate conditions that are happening at this time bring many changes to human life. On the one hand, some areas experience excessive rainfall. On the other hand, some areas experience long droughts. In addition to shifting changes in seasonal patterns, climate change is also increasing air temperatures above the average, impacting sea level rise. The World Meteorological Organization (WMO) provides several indicators to detect climate change, including the occurrence of temperature rise, an increase in extreme rainfall, a significant shift in seasons either forward or backward, and changes in the amount of rainfall volume [1]. The fastest sea level rise began in 2013 and reached its highest level in 2021. This condition is part of the ongoing global warming impact, where polar ice and glaciers also continue to melt, causing sea level rise. The rise in sea level is closely related to tidal floods. As the sea level increases, the coastal areas near the coast will be increasingly inundated by sea water. Tidal flooding is a rise in sea level due to the phenomenon of tides which causes inundation on

the coast. According to [2] several areas are threatened by tidal flooding in the Bandar Lampung area, namely the Karang City Village and the Kangkung Village. These two locations have a greater risk of tidal flooding because they are in an illegal area where many areas that are supposed to be watersheds leading to the beach are used as residences.

Based on the research conducted by [3], there are several sub-districts in the coastal area of Bandar Lampung City. This area are prone to tidal flooding, namely Kangkung Sub-District, Bumi Waras Sub-District, and Sukaraja Sub-District. Bandar Lampung itself is the city center in the coastal area where the majority of the population is in the Coastal area, so the threat of tidal flooding caused by rising sea levels due to global warming will cause a lot of harm to the people who live in the area.

According to [4], many coastal residents in the Lampung region still do not meet the ideal residence requirements from the shoreline, where the ideal distance from the shoreline is 300 m. This distance is to anticipate when a disaster comes, be it a tsunami or tidal flood. Another problem is that the community still does not know where to find information about early warnings of tidal floods and other hydrometeorological disasters. Of course, it is still a challenge for meteorical, climatological, and geophysical agency (BMKG) in all provinces to continue educating and socializing the community so that all of their products can be used and accepted by the community.

Sea level rise in 2100 is predicted to increase as high as 43-84 cm from current conditions or around 0.4-1.5 cm per year [5]. This condition will certainly be very dangerous for people who live in coastal areas. Therefore, with the early warning of tidal floods, it is hoped that will help coastal communities to prepare themselves, especially securing important items or documents when tidal floods come.

Based on research that has been conducted by [6]–[12] all of them have the same drawbacks even though they already have good training accuracy results and test accuracy results but have not yet reached the process of making a tidal flood prediction table. Likewise, research conducted by [13]–[20] have carried out an analysis related to the predicted height of sea tides and tidal floods by conducting training and testing datasets in the past using various variables. Still, no one has yet carried out a tidal flood prediction simulation for future events with various existing variables.

Rob flood research conducted by [21] has made a prediction table for tidal floods in the coastal area of Bandar Lampung City. Still, it only uses two variables, namely predictions of sea level height and waves, while this study uses six variables to predict tidal floods. In addition, if flood early warnings are made in real-time, the weaknesses are the lack of time for analysis, the lack of mitigation and evacuation processes, and the lack of time to disseminate information to the public [22], [23]. Research [24] uses many variables for the model learning process but leaves out many important variables such as air humidity, air temperature, and soil moisture.

This research focuses on the early warning system for tidal floods by utilizing threshold values associated with past occurrences of tidal flood phenomena. The existing tidal flood prediction system at BMKG lacks accuracy as it relies solely on sea level predictions from Pushidrosal, applying average threshold values across all regions in Indonesia. However, each region has its unique characteristics, making a nationwide implementation impractical.

Due to the lack of research specific to each region, the current threshold values are used uniformly. By employing an artificial neural network (ANN), this study aims to collect data on past flash flood events, enabling the analysis of patterns and the development of more accurate predictions using comprehensive data beyond sea level height. The ANN's non-linear analysis capabilities facilitate the study of flood event patterns. Subsequently, a tidal flood prediction table will be created to prevent future casualties. This research aims to address the limitations of previous studies and provide greater benefits to the field of education and the community.

II. Research Method

In this section, we describe the research methodology employed to achieve the objectives of this study. The research method encompasses various aspects such as data collection, research time, research sites, dataset selection, data analysis techniques, research flow, and model flow chart. Each of these subsections plays a crucial role in ensuring the validity and reliability of the research findings.

A. Data Collection

The data collection was conducted using the following methods:

1) Tidal flood events were collected from online media sources and through direct reports from individuals residing in areas affected by tidal floods.
2) Tidal prediction data for the period 2020-2022 was obtained from Pushidrosal.
3) Atmospheric dynamics data, such as gradient wind maps or 3000ft layer winds, were acquired from the website http://www.bom.gov.au/australia/charts/archive/index.shtml.
4) Wave data, wind speed, and direction were obtained from the archives of the BMKG for the period 2020-2022.
5) Astronomical events were extracted from the astronomical calendar.

6) Additionally, these datasets can also be accessed through https://www.kaggle.com/datasets/ramadhannurpambudi/dataset-kejadian-banjir-rob-pesisir-bandar-lampung.

The total number of data points utilized in this research amounts to 364. To train the models effectively, the data was divided into several proportions for training and testing purposes, namely 50:50, 60:40, 70:30, 80:20, and 90:10.

B. Research Duration

The time specified for conducting this research is the period 2020-2022. Tidal flood events are collected based on this period. In addition, training and testing data variables such as predictive data on sea level, wind speed, atmospheric dynamics, waves, and astronomical events are also collected based on the period of tidal flood cases.

C. Research Sites

The location of research focus was carried out in the coastal area of Bandar Lampung City. According to constitution of the Republic of Indonesia (UUD) Number 1 of 2014, the Coastal Area is a transition area between land and sea ecological systems that are affected by changes in land and sea. Furthermore, this area is the area where human life first existed from the coastline. Therefore, the research location is focused on the Panjang area, namely the Geospatial Information Agency (BIG) tidal station located at 05° 28 11.96 South Latitude and 105° 19 11.99 East Longitude, which has an MSL (Mean sea Level) of 2,318 m.

D. Dataset Description

The dataset used in this discussion consists of the following:

1) Tidal flood event data,
2) Sea level prediction data,
3) Wind speed data,
4) Atmospheric dynamics data,
5) Wave data, and
6) Astronomical event data.

Data on tidal flooding events serve as target data in training and testing. For prediction, data on sea level height, wind speed, atmospheric dynamics, waves, and astronomical events act as training data and testing data during the training process in Matlab. The data in the BMKG archive is wind speed and wave data. In addition, other data can be accessed via the internet.

E. Data Analysis

Analysis of making tidal flood prediction tables in the coastal area of Bandar Lampung City is carried out in several stages, namely:

1) Data collection stage,
2) Data training stage,
3) Data testing stage,
4) Formula selection stage,
5) The stage of making a tidal flood prediction table.

The data collection stage was carried out on Ms. Excel. Subsequently, the data training and testing were performed using the Matlab application with several configuration settings, including the number of hidden layers, number of neurons, and epoch. Configuration settings are carried out experimentally to get the best R-value or above 0.9 with minimal error values. The data that has been prepared in Excel will then be added to the Matlab database to go through the training and testing process.

The training dataset consisted of 150 data points, with 125 as training data and 25 as training targets. The test dataset comprised 48 data points, with 40 as test data and eight as test targets. The arrangement of hidden layers is from 2-5, the number of neurons is from 10-50, and the epoch is from 2,000-10,000. The backpropagation algorithm or feed forward backprop algorithm is used for Network Type. The Train Function uses traingd, and the Adaptation learning function uses learngd. For networks with more than two layers, the transfer function logsig was used for the 2nd layer and onwards, while the purelin function was employed for the last layer. In networks with only two layers, logsig was used for the first layer, and purelin for the second layer. The training parameter settings were adjusted solely based on the number of epochs, while the remaining parameters used the default values.

After configuring the data network, the next step is configuring the training parameters for epoch settings and max_fail or error rate during the training process. The epoch and max_fail values are equal in value so that during the training process, the epoch value can reach the specified value. If the max_fail value is made smaller, it is often the max_fail value that is reached before the epoch value reaches the specified target value. As for the other parameters, they are set in default mode. After going through this configuration setting process, the next step is to conduct data training until the desired R results are obtained. This study took the training value when R was more than 0.9. After training and testing up to 10 times, data was merged on Ms. Excel to do the accuracy test process.

Following the accuracy test process, the best formula was determined for creating tidal flood prediction tables using the provided schemes. The prepared tidal flood scheme data was then subjected to the prediction process using the best formula. The outcome of this process was a prediction table ready for use by BMKG forecasters to disseminate information to the community and local governments.

F. Model Flowchart

Fig. 1 outlines the training model process for predicting tidal floods. It starts with obtaining the dataset,

which is then divided into training and testing data. The ANN is utilized to train and test the model, with adjustable hyperparameters like hidden layers, neurons, and epochs. The model's accuracy is evaluated to assess its predictive capability, and the best-performing model is determined based on accuracy comparison. Finally, the trained model is used to predict tidal floods by inputting relevant data such as wave height, wind speed, and astronomical events, aiding in flood understanding and preparation.

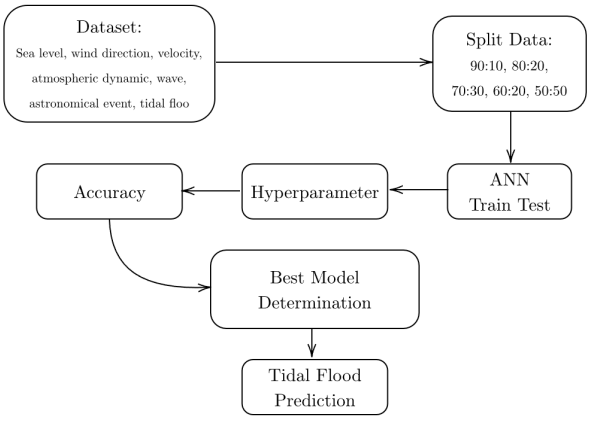

Fig. 1. Model flowchart.

III. RESULT

Training and testing of 364 datasets have been carried out using 20 experimental configuration models. The configurations carried out are related to the number of hidden layers used, the number of neurons, and the epoch or iteration, there is no guide in managing these three configurations, and the three are arranged experimentally to meet the training correlation target of 0.9. The training and testing dataset is divided into five parts, the first 50:50, the second 60:40, the third 70:30, the fourth 80:20, and the fifth 90:10. From a total of 364 datasets, it is then divided into five parts, 50:50, 60:40, 70:30, 80:20, and 90:10.

There are seven variables used in the training and testing process, namely predictions of sea level height, wind direction, wind speed, atmospheric dynamics, wave height, astronomical events, and tidal floods. The number of tidal flood events in the dataset is 26 incidents and the non-flooding events. The data between flood and non-flood events are balanced. The dataset is trained on Matlab. These variables will be processed to find training correlation values, test correlations, training accuracy, and test accuracy. The hyperparameter feature will also be used to find the one best model for making tidal flood prediction tables.

A. Summary of Backpropagation ANN Correlation

1) Training

The 50:50 dataset exhibits the highest training correlation value of 0.998, followed by the 60:40 dataset with a correlation value of 0.994, the 70:30 dataset with a correlation value of 0.990, the 80:20 dataset with a correlation value of 0.978, and the 90:10 dataset with a correlation value of 0.965. Among these divisions, the 50:50 dataset obtains the highest correlation value of 0.998, but its test correlation value reaches only 0.684.

The smallest error value is achieved by the 80:20 dataset, which is 0.0003. On average, there is a significant decrease in correlation values from the best training correlation to the testing correlation, amounting to 0.694. All datasets experience a notable decline in correlation values during the testing phase. Regarding the configuration parameters used, it is observed that the best correlation values are achieved with four or five layers, approximately 80 % of the neurons being utilized (neurons 50), and an epoch value of 10,000 for all datasets. However, it should be noted that having a large number of parameters does not guarantee equally good test correlation values, as observed in this study.

Table 1. Summary of Correlation of Backpropagation ANN Training Results

Train Test	Config	Layers	Neuron	Epoch	R Train	R Test	Diff R	Error
50:50	19	5	10	10,000	**0.998**	0.684	-0.313	0.001
60:40	13	4	50	10,000	**0.994**	0.26	-0.734	0.0013
70:30	9	5	50	10,000	**0.99**	0.289	-0.701	0.003
80:20	2	4	50	10,000	**0.978**	0.071	-0.907	0.0003
90:10	13	4	50	10,000	**0.965**	0.152	-0.813	0.005

2) Testing

The 50:50 dataset achieves the best test correlation value of 0.715, followed by the 60:40 dataset with a value of 0.708. The 70:30 dataset obtains a test correlation value of 0.678, while the 80:20 dataset shows the best value of 0.722. Lastly, the 90:10 dataset also achieves a test correlation value of 0.715. Thus, the highest test correlation value of 0.722 is obtained from the 80:20 dataset.

The average test correlation value for all five dataset divisions is 0.708, while the average training correlation value is 0.941. This indicates a decrease in the average correlation value of 0.233 during testing. In terms of error values, the best value obtained from the correlation test in Table 2 is 0.0007, and the average error across the five dataset divisions is 0.0057.

Regarding the configuration parameters, the five dataset divisions utilize 80 % of the layers, which corresponds to three layers. For neurons, the range varies from 20 to 50, excluding the use of 30. As for epoch, 80 % of the divisions employ an epoch value of 5,000. Based on the findings of this study, it is suggested to use a moderate number of layers and an epoch value of 5,000 to achieve high correlation test results.

Table 2. Summary of Correlation of Backpropagation ANN Test Results

Train Test	Config	Layers	Neuron	Epoch	R Train	R Test	Diff R	Error
50:50	8	2	20	5,000	**0.951**	**0.715**	-0.235	0.0007
60:40	14	3	50	5,000	**0.955**	**0.708**	-0.247	0.01
70:30	1	3	40	5,000	**0.955**	**0.678**	-0.277	0.01
80:20	14	3	50	5,000	**0.929**	**0.722**	-0.208	0.004
90:10	4	3	20	10,000	**0.914**	**0.715**	-0.199	0.005

B. Backpropagation ANN Training and Test Results

1) 50:50

The results of backpropagation ANN training with a 50:50 dataset division yield the best correlation value of 1 in configuration models 13, 14, and 19. This is an excellent result where the value 1 is the best correlation value from the model's predictions to the target data. The lowest correlation value is 0.92, obtained in configuration model 3. The configuration model with the best results has various settings for the number of hidden layers, number of neurons, and epochs. The best value for the number of hidden layers comes from setting 3, 4, and 5 layers. Neurons, 10 and 50 were used.

Meanwhile, the epochs used in the model were 5,000 and 10,000. For the model configuration, the lowest value came from the configuration of layers 2, neurons 30, and epoch 3,000. No configuration setting can truly be stated that will produce a large correlation value. In configuration models 15 and 16, where the epoch used is also 3000, it produces a correlation value of 0.99, which is almost perfect. For the number of hidden layers 2 in other configuration models, good correlation values were also obtained, such as configuration model 6 (0.97), configuration model 8 (0.95), and configuration model 11 (0.98). Likewise, the number of neurons used does not guarantee better accuracy results. In configuration 4 model, which uses 20 neurons, produces a value of 0.97. So no standard can be used in setting the three parameters of this model to get a good correlation value.

The 50:50 ANN test, conducted using 20 configuration models, revealed a decrease in correlation values. The largest decrease was observed in configuration model 6, with a correlation drop from 0.968 during training to 0.092 during testing, resulting in a decrease of 0.876. On the other hand, the smallest decrease occurred in configuration model 3, with a decline of 0.216 from the training correlation value of 0.923 to 0.707 during testing.

Among the 20 configuration models, the average test correlation value was 0.514, while the average training correlation was 0.975. This indicates an average decrease in correlation value of 0.461 across the models. The highest test correlation value of 0.715 was obtained from configuration model 8. None of the correlations reached a value of 0.9, whereas during training, no configurations had a correlation value below 0.9.

2) 60:40

The results of backpropagation ANN training with a dataset division of 60:40 yield the best correlation value of 0.99 in configuration models 5, 9, 10, 13, 17, and 20. This exceptional outcome indicates a correlation value that is close to the ideal value of 1. On the other hand, the lowest correlation value of 0.95 is obtained in configuration models 3, 8, and 14. Notably, when the dataset is divided in a 50:50 distribution, one configuration achieves a correlation value of 1, while the highest correlation value in the 60:40 distribution is 0.99. However, the average correlation for all 20 configuration models is 0.97 for both the 50:50 and 60:40 datasets.

Similar to the 50:50 dataset, the 60:40 dataset does not provide a clear standard for setting parameters such as the number of hidden layers, number of neurons, and epochs. Merely using a large number of parameters does not guarantee a high correlation value. Consequently, this research was conducted experimentally to determine the optimal parameter settings. It should be noted that increasing the number of parameter configurations used will prolong the training process.

The testing results of the 60:40 dataset with backpropagation ANN show a general decrease in correlation values compared to the training results. Across the 20 configuration models used in the testing process, the average decrease in correlation value is 0.612. The average training correlation value is 0.975, whereas the average test correlation value is 0.362. The decrease in correlation during testing is quite substantial for this dataset.

Among the tested models, configuration model 14 achieves the highest test correlation value of 0.708, while the smallest value of 0.050 is obtained by configuration model 7. Model 7 experiences the most significant decrease in correlation value, dropping from 0.975 during training to 0.050 during testing. None of the test correlations for the 60:40 dataset can match or exceed the correlation values during training. This pattern is consistent with the testing of the 50:50 dataset, where all tests exhibit a decrease in correlation values compared to training.

The three model configurations with the lowest test correlation values have the following parameter settings: model 7 configuration (0.050) utilizes 3 layers, 30 neurons, and 5,000 epochs. Model 2 configuration (0.052) employs 4 layers, 50 neurons, and 10,000 epochs. Lastly, configuration model 10 (0.077) incorporates 5 layers, 20 neurons, and 10,000 epochs. The different layer options are 3, 4, and 5, while the different neuron options are 20, 30, and 50. The epoch parameter is set to 5,000 for most configurations, except for 2 configurations that use 10,000 epochs. These results indicate that larger parameter configurations do not necessarily yield higher test correlation values.

3) 70:30

Based on the training results of the 70:30 dataset using backpropagation ANN, the average training correlation of the 20 configuration models is 0.95. The highest correlation value, 0.99, is achieved by training configuration model 9. On the other hand, several model configurations (3, 8, and 13) have the lowest

correlation value of 0.92. Among all the parameters, the configuration model 9 stands out with the highest correlation value. It has the largest number of parameter configurations, with five layers, 50 neurons, and 10,000 epochs. Conversely, the configurations with the lowest correlation values tend to have smaller parameters. For instance, configurations 3 and 8 have 2 layers, 20 neurons (configuration 3), 30 neurons (configuration 8), and 3,000 epochs (configuration 3) or 5,000 epochs (configuration 8). Configuration model 13, however, exhibits larger layer size (4), 50 neurons, and 10,000 epochs.

Out of the model configurations utilizing two hidden layers (configurations 3, 6, 8, 11, and 18), the highest correlation value is 0.96, with an average of 0.94 for these five configurations. Interestingly, among these configurations, three of them (3, 8, and 11) have relatively low correlation values of 0.92 and 0.93. Even though configuration model 11 uses the largest number of neurons (50) and epochs (10,000), its correlation value is only 0.93. This suggests that in the 70:30 dataset, having a small number of hidden layers is sufficient to influence the resulting training correlation value.

Testing the 70:30 dataset with backpropagation ANN using 20 configuration models yields an average test correlation value of 0.488. In comparison, the average correlation value during training reaches 0.951, resulting in a decrease of 0.463 in correlation during testing. The highest test correlation value, 0.678, is achieved by model 1 configuration, while the lowest test correlation value, 0.289, is obtained from the model 9 configuration. Surprisingly, despite having the best training correlation value, the model 9 configuration performs poorly during testing.

The test correlation values for the 70:30 dataset are better than those of the 60:40 dataset, where the lowest value reaches 0.050, whereas in the 70:30 dataset, the lowest value is only 0.289. However, even when using the highest parameters, configuration model 9 fails to produce satisfactory results. This configuration actually contributes to the decrease in correlation during testing. Model configurations that employ three hidden layers (configurations 1, 4, 7, and 14) achieve an average test correlation of 0.572, surpassing the overall test average of 0.488. Moreover, the average decrease in test correlation for configurations using 3 layers, 0.378, is smaller than the total average decrease of 0.463. Among the configurations using three layers, the model 4 configuration exhibits the largest decrease in test correlation, with a decline of 0.591 from a training correlation value of 0.954 to a testing correlation value of 0.363.

4) 80:20

Based on the training results of the 80:20 dataset using backpropagation ANN, the average training value for the 20 configuration models is 0.94. This average value is the lowest compared to the 50:50, 60:40, and 70:30 datasets. The highest average values are still found in the 50:50 and 60:40 datasets, with an average correlation value of 0.97. The highest correlation value obtained in ANN training with the 80:20 dataset is 0.98, achieved by configuration model 2. On the other hand, the lowest correlation value of 0.91 is obtained from configuration models 10 and 13.

Out of the 20 configuration models, the ones using four hidden layers, namely models 2, 5, 13, and 20, have an average training correlation value of 0.95, slightly higher than the overall average. Most of the configuration models using 4 layers also utilize the largest epoch setting of 10,000, except for model 5, which uses 5,000 epochs. However, despite having correlation values greater than 0.9, the results are not significantly different. In fact, the configuration model with the smallest correlation value is one that uses 4 layers, specifically model 13. These findings indicate that using large parameter values, such as epochs (10,000) and neurons (50), does not guarantee significant results, especially in this study.

The results of the 80:20 dataset test indicate an average correlation value of 0.284 across the 20 model configurations. Among these configurations, the highest correlation value of 0.722 is achieved by configuration model 14, while the lowest correlation value of 0.034 is obtained by configuration model 3. Three configuration models have correlation values below 0.1, namely configuration model 3 (0.034), configuration model 2 (0.071), and configuration model 10 (0.074).

The average value of correlation decrease from training to testing is 0.651, with the highest decrease of 0.907 observed in configuration model 2. This decrease value is greater than the highest testing correlation value (0.722). Configuration model 2 is characterized by parameters such as four layers, 50 neurons, and 10,000 epochs. Despite using the largest values for neurons and epochs in this experiment, along with the second largest number of layers, the test result is only 0.071.

When examining configuration models that use five layers, namely models 9, 10, 12, 15, 16, 17, and 19, the average test correlation value is 0.271, which is lower than the overall average (0.284). These models show a significant decrease in correlation from training to testing, dropping from an average training correlation of 0.937 to 0.271 during testing, indicating a decrease of 0.666. The configuration model with the largest decrease in correlation is model 10, with a decrease of 0.840. This model employs 20 neurons and 10,000 epochs. The highest test correlation value among the configuration models with five layers is 0.525 (configuration model 12), while the lowest is 0.074 (configuration model 10).

Analyzing the performance of models with five

layers separately provides insights into the behavior of configurations using the largest number of layers in this study. The training results are considerably higher than the overall average (0.937), but the testing results are significantly lower, falling below the overall average (0.271).

5) 90:10

The training results for the last dataset division, the 90:10 distribution, yielded an average training correlation value of 0.93. Among the configuration models, models 10 and 13 achieved the highest training correlation value of 0.96, while model 6 had the lowest correlation value of 0.89. This indicates that there is a configuration model with a training correlation value below 0.9. Despite being trained extensively, this particular configuration model did not reach the desired correlation value. The inclusion of more training data (90 % of the dataset) was expected to enhance the learning of the ANN regarding the patterns of tidal flood events. The subsequent analysis will assess the test correlation value and accuracy of these configurations.

Model 6 had the following parameter settings: two layers, 30 neurons, and 5,000 epochs. To evaluate the performance of neurons, the configurations were segregated based on the presence of 20 neurons. The average training correlation obtained was 0.926. This result stemmed from models 4, 8, 10, 16, and 17. Among these models, model 10 achieved the highest training correlation value of 0.956, while model 8 attained the smallest value of 0.903.

The average test correlation for the 90:10 dataset is 0.402, while the average training correlation reaches 0.929, indicating a decrease of 0.527 during testing. The best test correlation value in the 90:10 dataset is 0.715, obtained from configuration model 4, whereas the worst value is 0.063, obtained from configuration model 2. Configuration model 2 experiences the most significant decrease in correlation value, dropping from 0.952 during training to 0.063 during testing, a decrease of 0.889.

To assess the performance of the configuration model using the number of neurons, an analysis is conducted on the dataset using 50 neurons. Seven configuration models (2, 9, 11, 13, 14, 15, and 18) utilize 50 neurons. Among these models, the average test correlation is 0.269, which is approximately 50 % lower than the overall average test correlation. However, the average training correlation for these models is higher than the overall average, with a value of 0.937 compared to 0.929. The best test correlation value of 0.559 is obtained from configuration model 11, while the worst value of 0.063 is obtained from configuration model 2. Surprisingly, the configuration model that employs 50 neurons exhibits the worst accuracy value.

C. Summary of Backpropagation ANN Accuracy

Table 3 displays the best accuracy values obtained from dataset distributions. The 50:50 dataset achieves a training accuracy of 100 % and a testing accuracy of 88 %. In the case of the 60:40 dataset, the best training accuracy is 100 %, while the best testing accuracy is 81 %. As for the 70:30 dataset, the training accuracy stands at 97 %, and the testing accuracy is 81 %. The 80:20 dataset records a training accuracy of 98 % and a testing accuracy of 80 %. Lastly, the 90:10 dataset demonstrates a training accuracy of 100 % and a testing accuracy of 80 %.

Across the five dataset divisions, there is an average decrease in accuracy from training to testing of 17 %. The average training accuracy is 99 %, while the testing accuracy is 82 %. The highest training accuracy is 100 %, and the highest testing accuracy is 88 %. Notably, 60 % of the best configuration models representing the dataset divisions stem from the same model, namely configuration model 14. This configuration employs three layers, 50 neurons, and 5,000 epochs. Given these outcomes, configuration model 14 emerges as the most reliable and stable model since it is the best configuration in three dataset divisions, attaining a training accuracy of 100 % and a testing accuracy of 81 %.

Table 3. Summary of ANN Backpropagation Training and Test Accuracy

Train & Test								
Data Split	**Config**	**Layers**	**Neuron**	**Epoch**	**R Train**	**R Test**	**Diff R**	**Error**
50:50	20	4	30	10,000	**100**	**88**	-12	Good
60:40	14	3	50	5,000	**100**	**81**	-19	Good
70:30	14	3	50	5,000	**97**	**81**	-16	Good
80:20	14	3	50	5,000	**98**	**80**	-18	Good
90:10	13	4	50	10,000	**100**	**80**	-20	Good

D. Accuracy of ANN Backpropagation

1) 50:50

In the 50:50 dataset distribution, the average training accuracy is 99 % with a testing accuracy of 71 %, indicating a 28 % decrease in accuracy. The largest decrease of 54 % occurs in configuration model 6, dropping from its initial 100 % training accuracy to 46 %. Only six configurations achieve an accuracy of 80 %, namely models 3, 4, 5, 8, 19, and 20. The highest test accuracy, 88 %, is achieved by model 20, while the highest training accuracy is 100 % with 80 % of the 20 configurations achieving that level of accuracy. The remaining 20 % of configurations have a training accuracy of 96 %. These six configurations with 80 % accuracy during testing have an average training accuracy of 99 % and testing accuracy of 83 %. Their training accuracy is 100 % while their testing accuracy is 81 %. These six models utilize a diverse range of hidden layers (2, 3, 4, and 5), neurons (10, 20, 30, and 40), and epochs (3,000, 5,000, and 10,000), with the exception of 50 neurons. This highlights the absence of a specific parameter configuration as a reference for achieving high accuracy values.

2) 60:40

The accuracy analysis of the 60:40 dataset shows an average training accuracy of 99.7 % and testing accuracy of 64 %. This indicates a 36 % decrease in accuracy, with the highest decrease of 52 % observed in model configuration 2 and the smallest decrease of 19 % in model 14. During training, 90 % of the data achieved 100 % accuracy, while the remaining 10 % had an accuracy of 97 %. The training accuracy results for the 60:40 dataset are generally excellent, with only one configuration model (model 14) achieving an 80 % testing accuracy. The other models ranged from 48 % to 76 % accuracy. During training, two models (models 8 and 16) surpassed 100 % accuracy with a value of 97 %, both utilizing 20 neurons. On average, these models outperformed the overall average, achieving 71 % accuracy compared to the total average of 64 %. The best testing accuracy of 81 % was obtained from model 14, while the worst testing accuracy of 48 % was observed in model 2, both using 50 neurons. Additionally, model 2 had a larger epoch value (10,000) compared to model 14 (5,000).

3) 70:30

Based on the 70:30 dataset accuracy results, the average training accuracy for the 20 configuration models is 97.5 %. Three model configurations (2, 9, and 10) achieved a perfect training accuracy of 100 %, while the lowest training accuracy of 94 % was observed in model configuration 19. Regarding testing accuracy, the average value is 69 %. The highest testing accuracy of 81 % was obtained in model configuration 14, while several configurations (9, 11, 12, 16, and 17) had the lowest testing accuracy of 63 %. On average, there was a 28 % decrease in accuracy from training to testing. Model configuration 9 exhibited the greatest decrease of 63 % from its perfect training accuracy, while model configuration 14 showed the smallest decrease of only 16 % from its training accuracy of 97 % to testing accuracy of 81 %. Model configuration 14, with three hidden layers, 50 neurons, and 5,000 epochs, yielded the best testing accuracy. Model configurations 2, 9, and 10, all with perfect training accuracy, shared the same 10,000 epochs but achieved an average testing accuracy of around 60 %, resulting in an average decrease of 33 %.

4) 80:20

In the analysis of the 80:20 dataset, the average training accuracy is 97 %, with the highest value of 100 % obtained from models 2 and 20, and the lowest accuracy value of 93 % from model 19. During testing, the average accuracy is 74 %, with the highest accuracy of 80 % and the lowest at 70 %. Compared to the previous dataset division, the 80:20 dataset has the highest average test accuracy, with no configuration model falling below 70 % accuracy. Several models, namely models 1, 6, 12, 14, 15, 17, and 19, achieve an accuracy of 80 %. The average decrease in accuracy from training to testing is 24 %, with the greatest decrease at 30 % from model 2 and the smallest at 13 % from model 19.

Further analysis is conducted on the seven configuration models with a test accuracy of 80 %. These models do not reach a perfect training accuracy of 100 %, with the highest at 98 %. The decrease in accuracy for these models is only 17 % below the average test accuracy of 24 %. Additionally, analysis is performed on the nine models using epoch 10,000. The average training accuracy is 97 %, and the average test accuracy is 72 %, slightly lower than the overall average accuracy of 74 %. Model 19, which employs epoch 10,000, has the lowest training accuracy of 93 %. Among the test accuracy results from epoch 10,000, 78 % have a value of 70 %, while only 22 % achieve an accuracy of 80 %. The average decrease in accuracy for these models is 25 %, higher than the overall average decrease of 17 %.

5) 90:10

Lastly is an analysis of training and testing accuracy based on a 90:10 dataset division. The highest training accuracy is 100 % obtained from configuration model 13, while the smallest is 91 % obtained from configuration model 8. For accuracy testing, the highest value is 80 %, obtained from 55 % of the total configuration model. In comparison, smallest is 40 %, obtained from the 9 nine configuration model, and the configuration model, which has the greatest decrease in accuracy value from the previous 98 % during training, a decrease of 58 %. For the average training accuracy based on the 20 configuration models that have been run, a value of 97 % is obtained, and for testing, it is 70 % for an average decrease in accuracy from training to testing by 27 %. Based on the distribution of the previous data that has been analyzed, the distribution of the 90:10 dataset is the division with the highest accuracy value when testing reaches the standard, namely 80 %. 55 % of the configuration models can achieve an accuracy value of 80 %, and the remaining 45 % are below 80 %.

The analysis was carried out on the configuration model with an accuracy of 80 %, namely configuration models 1, 3, 5, 6, 8, 10, 11, 12, 13, 14, and 17 obtained an average value of training accuracy of 96 % and testing of 80 %. The average decrease in accuracy is 16 % lower than the decrease in the total dataset, which decreased by 27 %. The number of hidden layers used from the 11 configuration models varies from 2, 3, 4, and 5, or all hidden layers used in this study are included. The neurons used are 20, 30, 40, and 50. Only neuron ten is not included in it, as for the complete epochs used, starting from 3,000, 5,000, and 10,000. Again, the parameter settings do not have a benchmark that can be used as a reference for future research to get high correlation and accuracy results. Especially for the dataset used in this study.

E. Hyperparameters and Non-Hyperparameters

The hyperparameter feature, available in the latest edition of Matlab, is utilized to optimize accuracy when training or testing datasets. However, to assess its performance in this study, tests were conducted using various dataset divisions (50:50, 60:40, 70:30, 80:20, and 90:10) to ensure objective results. These tests were exclusively carried out on the training data, utilizing the configuration model established in the previous analysis. Nonetheless, due to limitations imposed by the hyperparameter feature, not all configuration models could be tested.

According to Table 4, the hyperparameter-based ANN model achieved the best values for root mean squared error (RMSE) at 0.229, mean squared error (MSE) at 0.052, mean absolute error (MAE) at 0.100, and r-squared at 0.810. In comparison, the non-hyperparameter-based ANN model yielded the best results with a response surface methodology (RSM) value of 0.102, MSE of 0.010, MAE of 0.077, and an r-squared value of 0.978. These outcomes indicate that the non-hyperparameter-based ANN model performs better across these four parameters.

Furthermore, in the subsequent analysis, the hyperparameter feature is tested for dataset accuracy in both training and testing phases. The ANN model employing hyperparameters exhibits higher error values compared to the model without this feature. Similarly, the r-squared parameter for the hyperparameter-based ANN model only reaches a maximum value of 0.810, indicating that it can explain 81 % of the variables related to tidal floods. On the other hand, the non-hyperparameter-based ANN model achieves an r-squared value of 0.978, suggesting that it can account for 97.8 % of the variables explaining tidal floods. Consequently, ANN models without the hyperparameter feature demonstrate better comprehension of the relationship patterns within the dataset.

Table 4. Summary of Comparison of the Best Value of Hyperparameter ANN and Non-Hyperparameter ANN Models

Hyperparameter								
Data Split	**Config**	**Layers**	**Neuron**	**Epoch**	**R Train**	**R Test**	**Diff R**	**Error**
50:50	6	2	30	5,000	**0.229**	**0.81**	0.052	0.100
60:40	3	2	30	3000	**0.279**	**0.74**	0.077948	0.14480
70:30	3	2	30	3000	**0.448**	**0.22**	0.200380	0.24386
80:20	18	2	50	5,000	**0.379**	**0.43**	0.143970	0.23393
90:10	11	2	50	10,000	**0.420**	**0.31**	0.176330	0.25548
Non-Hyperparameter								
Data Split	**Config**	**Layers**	**Neuron**	**Epoch**	**R Train**	**R Test**	**Diff R**	**Error**
50:50	7	3	30	5,000	**0.105**	**0.98**	0.011	0.078
60:40	11	2	50	10,000	**0.102**	**0.96**	0.010491	0.07697
70:30	1	3	40	5,000	**0.148**	**0.91**	0.021959	0.08837
80:20	1	3	40	5,000	**0.169**	**0.89**	0.028680	0.10737
90:10	14	3	50	5,000	**0.163**	**0.89**	0.026694	0.11047

For the 50:50 dataset, the best r-squared values were 0.81 and 1 for backpropagation ANNs with and without hyperparameters, respectively. The corresponding best MSE values were 0.052 and 0.002, and the best MAE values were 0.098 and 0.031. Similarly, the best RMSE values were 0.229 and 0.105. Out of the 20 configuration models used, only nine were compatible with the hyperparameter feature. However, the results indicated that backpropagation ANNs without hyperparameters performed better in terms of error values (RMSE, MSE, MAE). Only one configuration model achieved a perfect r-squared value of 1 using the hyperparameter feature.

Moving on to the 60:40 dataset, the average error values (RMSE, MSE, MAE) for backpropagation ANNs with hyperparameters were 0.329, 0.109, and 0.169, respectively, while the r-squared value was 0.63. Without hyperparameters, the corresponding average error values were lower (0.132, 0.018, 0.090), and the r-squared value was higher at 0.93. The best error values obtained were also better without using hyperparameters.

For the 70:30 dataset, backpropagation ANNs without hyperparameters consistently provided good accuracy results. The average error values (RMSE, MSE, MAE) for models with hyperparameters were 0.621, 0.403, and 0.332, respectively, while the r-squared value was 0.41. Without hyperparameters, the average error values were significantly lower (0.178, 0.032, 0.108), and the r-squared value was higher at 0.89.

Analyzing the 80:20 dataset, the average error values (RMSE, MSE, MAE) for models with hyperparameters were 0.540, 0.302, and 0.266, respectively, while the r-squared value was 0.25. Without hyperparameters, the average error values were significantly lower (0.188, 0.035, 0.128), and the r-squared value was higher at 0.86.

Comparing the best values across all datasets, models without hyperparameters consistently outperformed those with hyperparameters in terms of error values (RMSE, MSE, MAE) and r-squared values. However, the r-squared values for models without hyperparameters decreased as the training datasets increased.

Lastly, for the 90:10 dataset, the average error values (RMSE, MSE, MAE) for models with hyperparameters were 0.525, 0.284, and 0.292, respectively, while the r-squared value was 0.23. Without hyperparameters, the average error values were lower (0.194, 0.038, 0.129), and the r-squared value was higher at 0.84. The best error values were also achieved without using hyperparameters.

Based on these findings, the hyperparameter feature did not provide significant improvements in accuracy when applied to tidal flooding-related datasets. Therefore, it was not used further in the analysis process of this study.

In the analysis of the hyperparameter feature presented in Table 5, it is observed that the best average training accuracy value is 90 %, with the highest accuracy reaching 92 % and the lowest being 87 %. The average test accuracy value is 84 %, with the best accuracy reaching 88 % and the lowest accuracy value

at 80 %. Among the parameter settings that yield the best accuracy values for utilizing the hyperparameter feature, 60 % of the models have 2 layers and an epoch value of 5,000. However, there is no dominant parameter value for neurons.

On the other hand, for models that do not utilize the hyperparameter feature, the best average accuracy value is 98 %, with the highest accuracy reaching 100 % and the lowest being 96 %. The average testing accuracy is 81 %, with the best testing accuracy value at 85 % and the lowest accuracy at 80 %. Comparing these results, the average accuracy of models trained without hyperparameters yields better values.

Interestingly, the average accuracy of testing models with the hyperparameter feature is slightly better than models without it, with values of 84 % versus 81 %. However, it should be noted that the difference is not significant. The hyperparameter feature has been tested with nine different configurations across five different dataset divisions, but it has not demonstrated significantly better results compared to models trained and tested manually.

Table 5. Summary of the Accuracy of the Hyperparameter ANN Model and the Non-Hyperparameter ANN Model

Hyperparameter								
Data Split	**Config**	**Layers**	**Neuron**	**Epoch**	**Train**	**Test**	**Diff**	**Results**
50:50	11	2	50	10,000	**92**	**85**	-8	Good
60:40	1	3	40	5,000	**90**	**86**	-5	Good
70:30	3	2	30	3000	**89**	**88**	-1	Good
80:20	18	2	50	5,000	**91**	**80**	-11	Good
90:10	1	3	40	5,000	**87**	**80**	-7	Good
Non-Hyperparameter								
Data Split	**Config**	**Layers**	**Neuron**	**Epoch**	**Train**	**Test**	**Diff**	**Results**
50:50	3	2	30	3000	**96**	**85**	-11	Good
60:40	14	3	50	5,000	**100**	**81**	-19	Good
70:30	14	3	50	5,000	**97**	**81**	-16	Good
80:20	14	3	50	5,000	**98**	**80**	-18	Good
90:10	1	3	40	5,000	**98**	**80**	-18	Good

In the case of the 50:50 dataset, the average training accuracy of models using hyperparameters is 88 %, with the highest accuracy at 92 % and the lowest at 81 %. The average test accuracy is 83 %, with the best value at 89 % and the lowest at 73 %. Out of all the configuration models, 89 % achieve good test accuracy, while only 11 % fall below the standard (80 %), specifically at 73 %. On the other hand, models that do not utilize hyperparameters have an average training accuracy of 99 %, with the highest value at 100 % and the lowest at 96 %. However, the average test accuracy drops to 68 %, with the best value at 85 % and the lowest at 46 %. Among the nine configuration models, only three meet the standard value (80 %) in terms of test accuracy, while the remaining six fall short. The largest decrease in test accuracy from training is 54 %, observed in configuration model 6.

For the 60:40 dataset, the average training accuracy with hyperparameters is 86 %, ranging from 90 % as the highest value to 81 % as the lowest. The average test accuracy is 79 %, with the highest value at 86 % and the lowest at 62 %. Among the nine configuration models, only two fail to reach the standard value of 80 %, specifically models 4 and 11. The largest difference between training and testing accuracy is 28 %, as seen in the transition from 90 % during training to 62 % during testing. When hyperparameters are not utilized, the average training accuracy reaches 100 %, with the highest value at 100 % and the lowest at 97 %. However, the average test accuracy drops to 68 %, with the highest value at 81 % and the lowest at 57 %. Only one configuration model out of the nine achieves the standard value of 80 %, namely model 18 with a score of 81 %. The greatest difference in accuracy is 43 % in configuration models 1, 4, and 7, which all experience a decrease from 100 % during training to 57 % during testing.

For the 70:30 dataset, the analysis of the model using hyperparameters shows an average training accuracy of 87 %, with the highest value at 89 % and the lowest at 83 %. The average test accuracy is 78 %, with the highest value at 88 % and the lowest at 63 %. Only two configuration models fall below the standard test accuracy of 80 %. The largest decrease in accuracy, 26 %, is observed in configuration model 1, dropping from 89 % during training to 63 % during testing. Without using hyperparameters, the average training accuracy is 97 %, with the highest and lowest values also at 97 %. The average test accuracy is 72 %, with the maximum value at 81 % and the minimum at 63 %. The largest difference in accuracy from training to testing, 35 %, is seen in configuration model 11, decreasing from 97 % to 63 %. Only one configuration model achieves a test accuracy of 80 %, specifically configuration model 14.

For the 80:20 dataset, the average training accuracy of the hyperparameters model is 89 %, ranging from 91 % as the highest value to 83 % as the lowest. The average test accuracy is 79 %, with the best value at 80 % and the lowest at 70 %. Only one configuration model falls below the standard test accuracy, which is the seven configuration model with 70 % accuracy. Without using hyperparameters, the average training accuracy is 97 %, with the highest value at 98 % and the lowest at 95 %. The average test accuracy is 73 %, with the best value at 80 % and the lowest at 70 %. The greatest decrease in accuracy from training to testing is 28 %. Three configuration models achieve a standard test accuracy of 80 %, namely configuration models 1, 6, and 14.

Finally, for the 90:10 dataset, the average training accuracy for models using hyperparameters is 83 %, ranging from 87 % as the highest value to 77 % as the lowest. The average test accuracy is 71 %, with the best value at 80 % and the lowest at 60 %. Four out of nine configuration models fail to reach the minimum standard test accuracy, and the greatest decrease in accuracy is 27 %. Without using hyperparameters, the average training accuracy is 96 %, with the highest

value at 98 % and the lowest at 91 %. The average test accuracy is 71 %, with the best value at 80 % and the lowest at 60 %. Four configuration models achieve a standard test accuracy of 80 %, while the greatest decrease in accuracy, 38 %, is observed in configuration model 4. In terms of accuracy, the hyperparameter feature proves effective in providing good results. Additionally, for the accuracy test category across the five dataset divisions, the hyperparameter feature consistently yields better accuracy results compared to models trained without hyperparameters.

F. Best Model Determination

After analyzing different dataset divisions in terms of training, testing, and the use of the hyperparameter feature, the next step is to determine the most suitable model for creating prediction tables using the prepared dataset arrangement. The preferred model is one that excludes hyperparameters, as their usage did not yield satisfactory accuracy results, particularly in research related to tidal flooding datasets.

Four optimal configuration models were chosen based on their performance during training and testing. The 50:50 dataset was represented by 20 configuration models, the 60:40 dataset by 14 configuration models, the 70:30 dataset by 14 configuration models, the 80:20 dataset by 14 configuration models, and the 90:10 dataset by 13 configuration models.

Among these results, it can be observed that 60 % of the best models correspond to configuration model 14. This particular model utilizes a parameter arrangement of 4 hidden layers, 50 neurons, and 5,000 epochs. It demonstrates excellent performance across various dataset settings, exhibiting stability and above-average values for training correlation, testing correlation, training accuracy, and testing accuracy. The only unfavorable result is observed in the testing accuracy of the 50:50 dataset, which achieves a value of only 54 %. However, overall, the performance of the configuration model 14 is highly satisfactory. Therefore, this model will be utilized in creating the prediction table to determine the configuration model to be selected from the dataset. Notably, only a few of the 20 configuration models yielded a high correlation value when tested with the dataset.

For this reason, based on the highest test correlation value in the dataset, a test correlation value of 0.722 is obtained from configuration model 14 of the 80:20 dataset. In addition to having the highest test correlation value, this model also has the smallest error value (0.004) and the smallest decrease in correlation compared to the other best models (0.208). As for the value of the accuracy of the training has a value of 98, and the accuracy of the test is 80 %. Although not the highest, this configuration model has advantages in other sectors.

Table 6. Summary of Configuration Model Training and Testing Results

Data Split	*Config*	*Layer*	*Neuron*	*Epoch*	*Train Corr*	*Test Corr*	*error*	*Train Accu*	Test Accu	Results
50:50	20	4	30	10,000	0.990	0.703	0.013	100	88	Good
60:40	14	3	50	5,000	**0.955**	**0.708**	0.010	100	81	Good
70:30	14	3	50	5,000	**0.936**	**0.582**	0.006	97	81	Good
80:20	14	3	50	5,000	**0.929**	**0.722**	0.004	98	80	Good
90:10	13	4	50	10,000	**0.965**	**0.152**	0.005	100	80	Good

The dataset prepared for testing with the best model chosen is an arrangement of variables with various possibilities for future flood predictions. The dataset consists of high sea level tides ranging from 0.1-0.6 m, wind directions from various cardinal directions, wind speeds ranging from 2-45 knots, and the presence or absence of atmospheric dynamics, which is denoted by 0 if there are no atmospheric dynamics and one if there are atmospheric dynamics that occur around the Lampung area, wave heights ranging from 0-7 m. Then there is also an astronomical event that is happening whether it's a full moon or other astronomical phenomenon that affects the tides, is denoted by 0 if there is no astronomical phenomenon and one if there is an astronomical phenomenon that occurs.

G. Tidal Flood Prediction Table

Table 7 presents predictions for the dataset test, including a column showing the predicted tidal flood percentages. Overall, the results obtained are promising, as an increase in wave height and wind speed corresponds to a higher percentage of tidal flooding. In the table, a distinction is made based on the presence or absence of astronomical events. A value of 0 indicates no astronomical phenomenon affecting the tides, while a value of 1 signifies the occurrence of such events.

When there are no astronomical events (0), the predictions for tidal floods show slight inaccuracies in the percentage values when the wind direction changes in the dataset. This can be observed in table 7, where inaccuracies occur when the wind direction changes from 360 to 20 or from north to north-northeast. Specifically, when the wind speed is 0, there are no atmospheric dynamics, and the wave height is 0, the percentage of flood potential should ideally be minimal. However, in table 7, the percentage reaches 51 %, and it decreases to 24 % when the wind speed increases by 2 knots and the wave height is 0.5 m. This pattern continues until the wind direction changes to north-northwest. Nevertheless, when the wind speed reaches 4 knots and the wave height is 0.75 m, the percentage begins to improve and increases up to 40 knots and a wave height of 7 m.

The percentage results or predictions significantly improve when there are astronomical events or phenomena, such as new moons, full moons, lunar eclipses, and others that influence tidal activity. When the wind direction changes from north to north-northeast, the percentage results are very accurate. However, when the wind speed is 2 knots, there are no atmospheric dynamics, and the wave height is

0.5 m, the percentage drops to only 2 %. In such conditions, the chances of flooding are nearly non-existent. The percentage results remain stable until the wind direction changes to north-northwest, as long as there are active astronomical phenomena. The probabilities consistently start with the smallest numbers and gradually increase for each prepared parameter. In contrast, without astronomical events, the percentage immediately jumps to a higher value when the wind direction changes, then decreases, and eventually increases again. The percentage remains stable when the wind speed is 4 knots and the wave height is 0.75 m. However, below these values, the percentage results are not very reliable.

Table 7. Tidal Flood Prediction Table with 1.5 m Sea Level Tide (Part 1)

Sea Level	Wind direction	Wind velocity	Atmos Dynamic	Wave	Astronomical Events	Predict
1.5	360	0	0	0	0	7
1.5	360	2	0	0.5	0	23
1.5	360	4	0	0.75	0	21
1.5	360	6	0	1	0	27
1.5	360	8	0	1.25	0	42
1.5	360	10	0	1.5	0	68
1.5	360	15	0	2	0	97
1.5	360	20	0	2.5	0	81
1.5	360	25	0	3.5	0	100
1.5	360	30	0	4	0	100
1.5	360	35	0	5	0	100
1.5	360	40	0	6	0	100
1.5	360	45	0	7	0	100
1.5	360	0	0	0	1	30
1.5	360	2	0	0.5	1	57
1.5	360	4	0	0.75	1	64
1.5	360	6	0	1	1	67
1.5	360	8	0	1.25	1	59
1.5	360	10	0	1.5	1	40
1.5	360	15	0	2	1	56
1.5	360	20	0	2.5	1	93
1.5	360	25	0	3.5	1	100
1.5	360	30	0	4	1	100
1.5	360	35	0	5	1	94
1.5	360	40	0	6	1	88
1.5	360	45	0	7	1	87
1.5	90	0	0	0	0	46
1.5	90	2	0	0.5	0	24
1.5	90	4	0	0.75	0	18
1.5	90	6	0	1	0	8
1.5	90	8	0	1.25	0	0
1.5	90	10	0	1.5	0	0
1.5	90	15	0	2	0	28
1.5	90	20	0	2.5	0	70
1.5	90	25	0	3.5	0	100
1.5	90	30	0	4	0	100
1.5	90	35	0	5	0	100
1.5	90	40	0	6	0	99
1.5	90	45	0	7	0	95
1.5	90	0	0	0	1	19
1.5	90	2	0	0.5	1	14
1.5	90	4	0	0.75	1	26
1.5	90	6	0	1	1	39
1.5	90	8	0	1.25	1	39
1.5	90	10	0	1.5	1	33
1.5	90	15	0	2	1	64
1.5	90	20	0	2.5	1	90
1.5	90	25	0	3.5	1	50
1.5	90	30	0	4	1	69
1.5	90	35	0	5	1	83
1.5	90	40	0	6	1	93
1.5	90	45	0	7	1	96

IV. Discussion

The dataset analysis process utilizing backpropagation ANN has been conducted, resulting in the identification of the best configuration model for generating prediction tables. The process involved dividing the dataset proportions into 50:50, 60:40, 70:30, 80:20, and 90:10 to determine the most effective distribution for correlation values and accuracy. Each dataset underwent training and testing using 20 different configuration models with varying parameter settings, determined experimentally without specific guidance.

Table 8. Tidal Flood Prediction Table with 1.5 m Sea Level Tide (Part 2)

Sea Level	Wind direction	Wind velocity	Atmos Dynamic	Wave	Astronomical Events	Predict
1.5	180	0	1	0	0	2
1.5	180	2	1	0.5	0	21
1.5	180	4	1	0.75	0	37
1.5	180	6	1	1	0	46
1.5	180	8	1	1.25	0	53
1.5	180	10	1	1.5	0	64
1.5	180	15	1	2	0	81
1.5	180	20	1	2.5	0	47
1.5	180	25	1	3.5	0	48
1.5	180	30	1	4	0	61
1.5	180	35	1	5	0	67
1.5	180	40	1	6	0	69
1.5	180	45	1	7	0	74
1.5	180	0	1	0	1	6
1.5	180	2	1	0.5	1	21
1.5	180	4	1	0.75	1	31
1.5	180	6	1	1	1	38
1.5	180	8	1	1.25	1	47
1.5	180	10	1	1.5	1	64
1.5	180	15	1	2	1	79
1.5	180	20	1	2.5	1	100
1.5	180	25	1	3.5	1	97
1.5	180	30	1	4	1	96
1.5	180	35	1	5	1	97
1.5	180	40	1	6	1	97
1.5	180	45	1	7	1	95
1.5	270	0	1	0	0	0
1.5	270	2	1	0.5	0	1
1.5	270	4	1	0.75	0	9
1.5	270	6	1	1	0	15
1.5	270	8	1	1.25	0	28
1.5	270	10	1	1.5	0	55
1.5	270	15	1	2	0	100
1.5	270	20	1	2.5	0	91
1.5	270	25	1	3.5	0	75
1.5	270	30	1	4	0	91
1.5	270	35	1	5	0	92
1.5	270	40	1	6	0	94
1.5	270	45	1	7	0	100
1.5	270	0	1	0	1	21
1.5	270	2	1	0.5	1	35
1.5	270	4	1	0.75	1	50
1.5	270	6	1	1	1	67
1.5	270	8	1	1.25	1	84
1.5	270	10	1	1.5	1	86
1.5	270	15	1	2	1	80
1.5	270	20	1	2.5	1	100
1.5	270	25	1	3.5	1	100
1.5	270	30	1	4	1	100
1.5	270	35	1	5	1	100
1.5	270	40	1	6	1	100
1.5	270	45	1	7	1	100

In the 50:50 dataset, the best training correlation value achieved was 0.998, with a corresponding testing correlation of 0.715. The highest training accuracy reached 100 %, while the highest testing accuracy was 88 %, resulting in a decrease in accuracy from training to testing of 11 %. The smallest error value obtained was 0.0001. Among the six configuration models achieving testing accuracy above 80 %, the best configuration model was model 20.

Moving on to the 60:40 dataset analysis, the best training correlation value obtained was 0.994, with a testing correlation of 0.708. The decrease in correlation value was 0.247, and the best training accuracy reached 100 %, with the best testing accuracy at 81 % and a decrease in accuracy from training to testing of 19 %. The smallest error value was 0.0004, and only one configuration model achieved a testing accuracy of 80

%. The best configuration model for the 60:40 dataset was model 14.

Similarly, for the 70:30 dataset, the best training correlation value obtained was 0.990, with a testing correlation of 0.678. The decrease in correlation value was 0.277. The best training accuracy reached 100 %, with the best testing accuracy at 81 % and a decrease in accuracy from training to testing of 16 %. The smallest error value was 0.001, and only one configuration model achieved a testing accuracy of 80 %. The best configuration model for the 70:30 dataset was also model 14, making it the top performer twice so far.

Proceeding to the 80:20 dataset analysis, the best training correlation value obtained was 0.978, with a testing correlation of 0.722. The decrease in correlation value was 0.208. The best training accuracy reached 100 %, with the best testing accuracy at 80 % and a decrease in accuracy from training to testing of 13 %. The smallest error value was 0.0003, and seven configuration models achieved an accuracy value of 80 %. The best configuration model for the 80:20 dataset was, again, model 14, marking its third consecutive success.

Lastly, in the analysis of the 90:10 dataset, the best training correlation value obtained was 0.965, with a testing correlation of 0.715. The decrease in correlation value was 0.199. The best training accuracy reached 100 %, with the best testing accuracy at 80 % and a decrease in accuracy from training to testing of 11 %. The smallest error value was 0.001, and seven configuration models achieved an accuracy value of 80 %. The best configuration model for the 90:10 dataset was model 13.

The summary results of the training and testing process for all dataset divisions using 20 configuration models yielded the best correlation value of 0.998 for the test correlation of 0.722. The correlation value decreased by at least 0.199 from training to testing. The lowest error value obtained during the analysis process was 0.0001. The highest accuracy achieved was 100 % during training and 88 % during testing, resulting in an 11 % decrease in accuracy from training to testing.

Subsequently, an experiment was conducted using the hyperparameter feature, but only nine models could be tested due to the feature's limitation of accommodating a maximum of three layers, while this study employed up to five layers. The analysis with hyperparameters yielded the best RMSE value of 0.229, MSE value of 0.052, MAE value of 0.098, and r-squared value of 0.95. These results indicated that the variables used could explain 95 % of the occurrence of tidal flooding.

In contrast, the dataset analysis without using hyperparameters resulted in the best RMSE value of 0.102, MSE value of 0.002, MAE value of 0.031, and r-squared value of 0.99. These findings demonstrated that 99 % of the variables could accurately describe the presence or absence of tidal flood events. As the analysis without hyperparameters yielded better test results for all parameters, this study opted to continue the analysis without utilizing the hyperparameter feature.

In addition to analyzing the error value, the accuracy of models using the hyperparameter feature and those not using it was also examined. Testing accuracy consistently showed better results for models employing the hyperparameter feature across the five dataset divisions. However, the training accuracy for models with hyperparameters remained lower compared to models without this feature. Although the hyperparameter feature aims to increase accuracy, its performance in terms of error values falls short of models without this feature. Nevertheless, considering the context of this study, the accuracy and error values of the hyperparameter feature did not yield significantly better results than the manual method, particularly in the analysis of the tidal flood dataset. Nine models with different configurations and dataset distributions were tested, demonstrating good performance and representing the dataset being analyzed.

Moving forward, the analysis aimed to determine the best model for the provided dataset. Based on the training and testing results, the 14 configuration model was selected as the best among the best models, taking into account various considerations. This model consistently emerged as the best configuration model in three out of five dataset divisions. It exhibited the best test correlation value of all the models representing each dataset, with a value of 0.722 and the smallest error value of 0.004. Additionally, it demonstrated the smallest decrease in correlation value from training to the goal at 0.208.

The 14 configuration model then underwent a testing process on the dataset using variable settings, including a sea level tide of 1.5 m. Table 4.7 presents the results, showing that the model encounters slight difficulties in providing percentage results when there are changes in wind direction and variable settings. In actual conditions and considering the tidal flood dataset, higher wind speeds and wave heights increase the likelihood of tidal flooding. However, in Table 7, the results tend to be larger, with the percentage starting from a large value and progressing to smaller values. Since the dataset is structured to account for parameter values from smallest to largest, the percentage follows the dataset arrangement.

These unfavorable results persist until an average wind speed of 2 knots and a wave height of 0.5 m. Starting from a wind speed of 4 knots and a wave height of 0.75 m, the results show an increasing trend with more visible percentage composition. This pattern repeats with changes in wind direction. Conversely,

when an astronomical phenomenon affects tide height, the results are highly accurate, with the percentage values arranged from smallest to largest as expected. However, whenever there is a change in wind direction with other parameter conditions, the percentage decreases as anticipated. The tidal flood prediction table was created using the selected best model, the 14 configuration model. After undergoing an extensive selection process, the obtained results can be utilized by BMKG forecasters to issue early warnings to the public and stakeholders regarding the percentage of tidal floods based on observed parameters. This initiative aims to reduce casualties and minimize losses caused by tidal floods in the future.

V. Conclusion

Based on the analysis and discussion, important findings have emerged regarding the training and testing results. The model configuration experimentally achieved high accuracy, with a training accuracy of 100 % and the best testing accuracy of 88 %. The average correlation values for training were 0.975 for the 50:50 dataset, 0.975 for the 60:40 dataset, 0.951 for the 70:30 dataset, 0.935 for the 80:20 dataset, and 0.929 for the 90:10 dataset. The average correlation values for testing were 0.514 for the 50:50 dataset, 0.362 for the 60:40 dataset, 0.488 for the 70:30 dataset, 0.284 for the 80:20 dataset, and 0.402 for the 90:10 dataset. Additionally, the average error values were 0.006 for the 50:50 dataset, 0.006 for the 60:40 dataset, 0.010 for the 70:30 dataset, 0.007 for the 80:20 dataset, and 0.007 for the 90:10 dataset.

The flood prediction table, based on the best configuration model (model 14), achieved a training accuracy rate of 98 %, testing accuracy of 80 %, and an error value of 0.004. This model outperformed the other configurations in three out of the five dataset divisions. The prediction table accurately estimated tidal flood percentages, particularly during active astronomical phenomena. These results demonstrate that the backpropagation ANN effectively learns from the datasets and can be utilized by BMKG forecasters to issue early warnings and prevent future fatalities caused by tidal floods.

Acknowledgment

We would like to express our sincere gratitude to the Meteorology, Climatology, and Geophysics Agency (BMKG) for granting us permission and providing invaluable support throughout the completion of this research. We would also like to extend our appreciation to all the related parties involved in this study, whose contributions cannot be individually named but have been instrumental in its accomplishment.

References

[1] World Meteorological Organization, WMO Atlas of Mortality and Economic Losses From Weather, Climate, and Water Extremes, no. 1267. 2019. [Online]. Available https://library.wmo.int/doc_num.php?explnum_id=10902.

[2] W. Z. Ilmi, A. M. Asbi, and T. Syam, "Identifikasi karakteristik kawasan informal pesisir Kota Bandar Lampung dan kerentanan terhadap dampak perubahan iklim (Studi kasus: Kelurahan Kota Karang dan Kangkung)," *J. Pembang. Wil. Kota*, vol. 17, no. 2, pp. 149–167, 2021, doi: 10.14710/pwk.v17i2.33130.

[3] W. Wibisono, "Strategi penataan ruang berbasis mitigasi bencana berdasarkan tingkat kerentanan bencana Kota Bandar Lampung," *PWK Inst. Teknol. Sumatera*, no. 24, pp. 1–16, 2019.

[4] M. T. Sinaga, "Kesiapsiagaan masyarakat desa tangguh bencana di Desa Sukaraja Kecamatan Rajabasa Kabupaten Lampung Selatan," *J. Penelit. Geogr.*, vol. 7, no. 6, 2019.

[5] M. A. Marfai, D. Mardiatno, A. Cahyadi, F. Nucifera, and H. Prihatno, "Pemodelan spasial bahaya banjir rob berdasarkan skenario perubahan iklim dan dampaknya di pesisir Pekalongan," *Bumi Lestari*, vol. 13, no. 2, pp. 244–256, 2017.

[6] M. S. Elbisy, A. H. Aljahdali, A. H. Natto, A. A. Bakhsh, A. F. Almaliki, M. A. Alharthi, and A. O. Hassan, "Prediction of daily tidal levels along the central coast of Eastern Red Sea using artificial neural networks," *Int. J. GEOMATE*, vol. 19, no. 76, pp. 54–61, 2020, doi: 10.21660/2020.76.79417.

[7] X. Chen, H. Li, and Y. Yang, "A genetic neural network for tide forecasting," in *2021 IEEE 4th Int. Electr. Energy Conf.*, 2021, doi: 10.1109/CIEEC50170.2021.9510502.

[8] N. Raj and Z. Gharineiat, "Evaluation of multivariate adaptive regression splines and artificial neural network for prediction of mean sea level trend around northern australian coastlines," *Mathematics*, vol. 9, no. 21, Nov. 2021, doi: 10.3390/MATH9212696.

[9] B. V. Primo de Siqueira and A. de M. Paiva, "Using neural network to improve sea level prediction along the southeastern Brazilian coast," *Ocean Model.*, vol. 168, pp. 101898, Dec. 2021, doi: 10.1016/J.OCEMOD.2021.101898.

[10] M. R. Khaledian, M. Isazadeh, S. M. Biazar, and Q. B. Pham, "Simulating Caspian Sea surface water level by artificial neural network and support vector machine models," *Res. Artic. - Hydrol.*, 2020.

[11] M. P. Quang and K. Tallam, "Predicting Flood Hazards in the Vietnam Central Region: An Artificial Neural Network Approach," *Sustain.*, vol. 14, no. 19, 2022, doi: 10.3390/su141911861.

[12] R. Tabbussum and A. Q. Dar, "Performance evaluation of artificial intelligence paradigms—artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference system for flood prediction," *Environ. Sci. Pollut. Res.*, 2021, doi: 10.1007/s11356-021-12410-1.

[13] A. Ilyas, P. Wellyantama, S. Soekirno, M. Putra, D. P. Djenal, and A. M. Hidayat, "The implementation of artificial neural network (ANN) in the prediction of tides level data in Indonesia," in *2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)*, Bali, Indonesia, 24-26 November 2022.

[14] M. Bagheri, Z. Z. Ibrahim, L. A. Manaf, M. F. Akhir, and W. I. A. W. Talaat, "Simulation and analysis of sea-level change from tide gauge station by using artificial neural network models," *Sains Malaysiana*, vol. 51, no. 7, pp. 2003–2012, 2022, doi: 10.17576/jsm-2022-5107-05.

[15] H. A. Saputri and D. D. Santika, "Flood prediction based on weather and water level historical data using recurrent neural networks: A case study of Jakarta flood incidents," *Int. J. Intell. Syst. Appl. Eng.*, vol. 10, no. 4, pp. 195–200, 2022.

[16] H. Kardhana, J. R. Valerian, F. I. W. Rohmat, and M. S. B. Kusuma, "Improving Jakarta's katulampa barrage extreme water level prediction using satellite-based long short-term memory (LSTM) neural networks," *Water*, vol. 14, no. 9, 2022, doi: 10.3390/w14091469.

[17] T. Jayathilake, R. Sarukkalige, Y. Hoshino, and U. Rathnayake, "Wetland water level prediction using artificial neural networks—A case study in the Colombo flood detention area, Sri Lanka," *Climate*, vol. 11, no. 1, 2023.

[18] A. S. Azad, R. Sokkalingam, H. Daud, S. K. Adhikary, H. Khurshid, S. N. A. Mazlan, and M. B. A. Rabbani, "Water level prediction through hybrid SARIMA and ANN models based on time series analysis: Red hills reservoir case study," *Sustainability*, vol. 14, no. 3, 2022, doi: 10.3390/su14031843.

[19] R. J. Pally and S. Samadi, "Application of image processing and convolutional neural networks for flood image classification and semantic segmentation," *Environ. Model. Softw.*, vol. 148, pp. 105285, Feb. 2022, doi: 10.1016/J.ENVSOFT.2021.105285.

[20] G. Scardino, G. Scicchitano, M. Chirivì, P. J. M. Costa, A. Luparelli, and G. Mastronuzzi, "Convolutional neural network and optical flow for the assessment of wave and tide parameters from video analysis (LEUCOTEA): An innovative tool for coastal monitoring," *Remote Sens.*, vol. 14, no. 13, 2022, doi: 10.3390/rs14132994.

[21] E. S. P. Wulandari and R. Z. A. Azis, "Model prediksi dengan artificial neural network untuk," in *Semin. Nas. Has. Penelit. dan Pengabdi. Masy. 2022*, pp. 77–84, 2022.

[22] W. Bao and W. Bin, "Real-time tide prediction based on an hybrid HA-WANN model using wind information," in *2018 14th IEEE Int. Conf. Signal Process.*, Beijing, China, 12-16 August 2018, doi: 10.1109/ICSP.2018.8652339.

[23] J. Lee and B. Kim, "Scenario-based real-time flood prediction with logistic regression," *Water*, vol. 13, no. 9, 2021, doi: 10.3390/w13091191.

[24] M. El-Rawy, W. M. Elsadek, and F. De Smedt, "Flash flood susceptibility mapping in Sinai, Egypt using hydromorphic data, principal component analysis and logistic regression," *Water*, vol. 14, no. 15, 2022, doi: 10.3390/w14152434.