

Application of Nonlinear Autoregressive Neural Network Model to Forecast Local Mean Sea Level

Yeong Nain Chi

Department of Agriculture, Food and Resource Sciences, University of Maryland Eastern Shore, MD, USA

Abstract. The primary purpose of this study was to apply the nonlinear autoregressive neural network to model the long-term records of the monthly mean sea level from January 1978 to October 2020 at Grand Isle, Louisiana, as extracted from the National Oceanic and Atmospheric Administration Tides and Currents database. In this study, the empirical results revealed that the Bayesian Regularization algorithm was the best-suit training algorithm for its high regression R-value and low mean square error compared to the Levenberg-Marquardt and Scaled Conjugate Gradient algorithms for the nonlinear autoregressive neural network. Understanding past sea levels is important for the analysis of current and future sea level changes. In order to sustain these observations, research programs utilizing the existing data should be able to improve our understanding and significantly narrow our projections of future sea-level changes.

Keyword: Time series, monthly mean sea level, nonlinear autoregressive neural network, levenberg-Marquardt algorithm, bayesian regularization algorithm, scaled conjugate gradient algorithm, Grand Isle, Louisiana.

Received 18 June 2022 | Revised 24 July 2022 | Accepted 31 July 2022

1 Introduction

Climate change is the long-term change in the average weather patterns of the earth. It is primarily triggered by human activities like burning fossil fuels, deforestation, etc., or natural events like volcanic eruptions. The National Oceanic and Atmospheric Administration (NOAA) 2019 Global Climate Annual Report (<https://www.ncdc.noaa.gov/sotc/global/201913>) summarized that the global annual temperature has increased at an average rate of 0.07°C (0.13°F) per decade since 1880. This rate (+0.18°C / +0.32°F) of increase has doubled since 1981.

*Corresponding author at: 1 College Backbone Road, 1102 Trigg Hall, Department of Agriculture, Food and Resource Sciences, University of Maryland Eastern Shore, Princess Anne, MD 21853, USA

E-mail address: ychi@umes.edu

Two major factors related to climate change have caused sea levels to rise globally: the added water from ice melting from land (ice sheets and glaciers) into the ocean, and the thermal expansion of warming waters. Locally, the amount and speed of sea level rise vary by location. In particular, the slowing Gulf Stream and the sinking land affect some areas at varying rates in the United States (<https://sealevelrise.org/>). The potential impacts of sea level rise include, but are not limited to, increasing coastal flooding and erosion, damaging agricultural land cover and crops, damaging coastal/urban settlements and infrastructures, and damaging coastal flora and fauna ecosystems [1].

According to NOAA Climate.gov, the global average sea level has risen about 8-9 inches (21-24 cm) since 1880. Church et al. [2] used TOPEX/Poseidon satellite altimeter data combined with historical tide gauge data to compute the rate of global average sea level rise from the reconstructed monthly time series as 1.8 ± 0.3 mm/year from 1950 to 2000. Church and White [3] extended the reconstruction of sea level back to 1870 and found a sea level rise from January 1870 to December 2004 of 195 mm, a 20th century rate of sea level rise of 1.7 ± 0.3 mm/year and a significant acceleration of sea level rise of 0.013 ± 0.006 mm/year.

Furthermore, Church and White [4] revealed that the estimated rate of sea level rise was 3.2 ± 0.4 mm/year from the satellite data and 2.8 ± 0.8 mm/year from the in situ data. The global average sea level rise from 1880 to 2009 was about 210 mm. The linear trend from 1900 to 2009 was 1.7 ± 0.2 mm/year and since 1961 has been 1.9 ± 0.4 mm/year. They also documented that there was considerable variability in the rate of sea level rise during the 20th century but there has been a statistically significant acceleration since 1880 and 1900 of 0.009 ± 0.003 mm/year² and 0.009 ± 0.004 mm/year², respectively.

The Intergovernmental Panel on Climate Change (IPCC) [5] estimated that the sea level has risen by 26-55 cm (10-22 inches) with a 67% confidence interval. If emissions remain very high, the IPCC projected sea level could rise by 52-98 cm (20-39 inches). In its Fourth National Climate Assessment Report [6] the U.S. Global Change Research Program (USGCRP) estimated that the sea level has risen by about 7-8 inches (about 16-21 cm) since 1900, with about 3 of those inches (about 7 cm) occurring since 1993 (very high confidence). Relative to the year 2000, the sea level was very likely to rise by 1.0-4.3 feet (30-130 cm) in 2100, and 0.3-0.6 feet (9-18 cm) by 2030.

Many studies have pointed out that the sea level is rising at an increasing rate [7] [8] [9] [10] [11] [12]. Thus, understanding past sea levels is important for the analysis of current and future sea level changes. Modeling sea level changes and understanding their causes have considerably improved in the recent years, essentially because new in-situ and remote sensing observations have become available [13] [14] [15] [16]. Despite the importance of the sea level rise and its consequences, there is a lack of studies in the technical literature available on forecasting schemes at the local level of consideration.

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics from the data, while time series forecasting uses a model to predict future values based on previously observed values. Neural networks have become one of the most popular trends in machine learning for time series modeling and forecasting. Recently, there is increasing interest in using neural networks to model and forecast time series, especially in sea level rise issues [17] [18] [19] [20]. Hence, the primary purpose of this study was to apply the nonlinear autoregressive (NAR) neural network to model the long-term records of monthly mean sea level from January 1978 to October 2020 at Grand Isle, Louisiana.

2 Materials

2.1 Study Site

Grand Isle is a town at latitude 29.2366 and longitude -89.9873 in Jefferson Parish, Louisiana, located on a barrier island at the mouth of Barataria Bay where it meets the Gulf of Mexico. According to the United States Census Bureau, the town covers a total area of 7.8 square miles (20 km²), of which 6.1 square miles (16 km²) is land and 1.7 square miles (4.4 km²) is water. Grand Isle has a humid subtropical climate bordering on a tropical monsoon climate, with mild winters and long, hot, humid summers.

2.2 Data Source

The long-term records of the monthly mean sea level from January 1978 to October 2020 at Grand Isle, Louisiana, were used for this study and are available to the public from NOAA Tides and Currents (<https://tidesandcurrents.noaa.gov/>). Average monthly mean sea level was -0.1356 mm/year with the standard deviation of 0.1177 mm/year (Minimum: -0.431 mm/year, Maximum: 0.212 mm/year, and Median: -0.1455 mm/year) at Grand Isle, Louisiana, from January 1978 to October 2020 (Figure 1). According to NOAA Tides and Currents, the term mean sea level can refer to a tidal datum, which is locally-derived based on observations at a tide station, and it is typically computed over a 19-year period, known as the National Tidal Datum Epoch (NTDE).

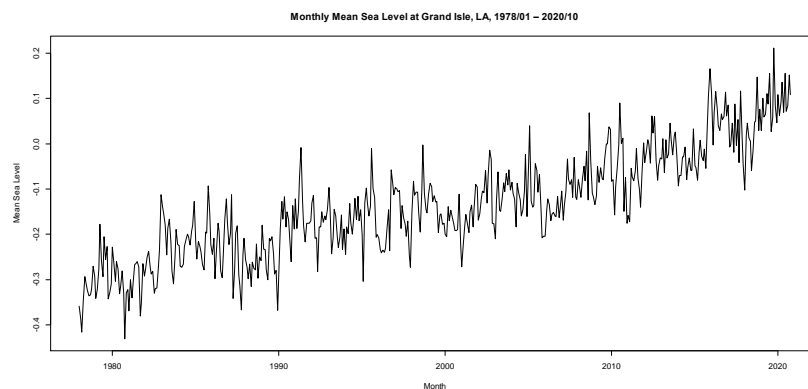


Figure 1. Time Series Plot of Monthly Mean Sea Level at Grand Isle, Louisiana, January 1978 ~ October 2020 (Source: own work)

3 Methods

In the neural network frameworks, multilayer feedforward neural networks, trained with the backward propagation algorithm, are the most popular neural networks. A simple multilayer feedforward neural network consists of neurons that are ordered into layers. The first layer is called the input layer, the last layer is called the output layer, and the layer between is the hidden layer.

With time series data, lagged values of the time series can be used as inputs to a neural network, the so-called nonlinear autoregressive (NAR) neural network model. A NAR neural network, applied to time series prediction using its past values of a univariate time series, can be expressed as follows:

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-d)) + e(t) \quad (1)$$

where $y(t)$ = the value of the time series at time t , d = the time delay, and $e(t)$ = the error of the approximation of the time series at time t , $y(t)$. This formula describes how a NAR neural network is used to predict the future value of a time series $y(t)$ using the past values of the time series. The function $f(\cdot)$ is a nonlinear function, and the training of the neural network aims to approximate the function by means of the optimization of the network weights and neuron bias [21].

The most common learning rules for the NAR neural network are the Levenberg-Marquardt, Bayesian Regularization, and Scaled Conjugate Gradient training algorithms. Training is the process of determining the optimal network weights and bias points of the *multilayer feedforward* neural network. This is done by defining the total error function between the network's output and the desired target and then minimizing it with respect to the weights.

3.1 Levenberg-Marquardt (LM) Algorithm

The LM algorithm, first published by Levenberg [22] and then rediscovered by Marquardt [23], is a commonly used iterative algorithm to solve non-linear minimization problems. These minimization problems arise especially in least squares curve fitting. This curve-fitting method is a combination of the gradient descent and the Gauss-Newton. It works without computing the exact Hessian matrix. Instead, it works with the gradient vector and the Jacobian matrix, thereby increasing the training speed while having stable convergence [24].

3.2 Bayesian Regularization (BR) Algorithm

The BR algorithm was introduced by MacKay [25] automatically sets the best possible performance function to accomplish the excellent generalization on the basis of Bayesian inference approach. The BR algorithm is based on the probabilistic interpretation of network parameters. Bayesian optimization of regularization parameters depends upon the calculation of the Hessian matrix at the minimum point. Therefore, the BR algorithm includes a probability

distribution of network weights and the network architecture can be identified as a probabilistic framework [26].

3.3 Scaled Conjugate Gradient (SCG) Algorithm

The SCG algorithm, developed by Moller [27], is based on the Conjugate Gradient Method, but this algorithm does not perform a line search at each iteration. Unlike many other standard backward propagation algorithms, the SCG algorithm is fully-automated, includes no critical user-specific parameters, and avoids a time-consuming line search.

4 Results

In this study, the NAR neural network was applied to model the time series monthly mean sea level: The network structure contains one input (corresponding to monthly mean sea level at time $t-1$, $y(t-1)$), and one output (the next value of the time series, $y(t)$, to be predicted). The number of delays was set experimentally after a data pre-processing and analysis stage. The extracted features were trained with MATLAB (2019a) Neural Network Toolbox using the LM, BR, and SCG training algorithms respectively for the target time series data: 514 timesteps of one element, the monthly mean sea level from January 1978 to October 2020 at Grand Isle, Louisiana.

Three kinds of target timesteps were set aside for the training, validation and testing in this case study. The training target timesteps are presented to the network during training, and the network is adjusted according to its error. The validation target timesteps are used to measure network generalization, and to halt training when generalization stops improving. The testing target timesteps have no effect on training and so provide an independent measure of network performance during and after training [21]. The division of the time series in this analytical work was 70% for the training, 15% for the validation, and 15% for the testing. Randomly, 514 data samples were divided into 360 data for the training, 77 data for the validation, and 77 data for the testing.

Open loop architecture was used to train the NAR neural network. Figure 2 showed a block diagram of the NAR neural network, generated during MATLAB processing. In Figure 2, the block $y(t)$ is the input series consisting of monthly mean sea level observations. The number “1” at the bottom of the block indicates univariate time series.

The hidden layer of the network is illustrated in the second block, namely ‘Hidden Layer with Delays.’ The inner boxes “w” and “b” represent input-hidden weights and bias respectively for a single neuron in the hidden layer. The term “1:2” denotes the number of delays used (2). The larger box after the summation sign indicates the sigmoid transfer function of each neuron. The number “10” at the bottom of the “Hidden” block denotes the number of hidden neurons.

The “Output” block represents the output layer of the network. The inner boxes “w” and “b” represent the hidden-output weights and biases respectively. The transfer function of the output layer is linear. There is only one output neuron, which is denoted below as the “Output” block. The last block $y(t)$ represents the predicted output. This output $y(t)$ is different from the input $y(t)$. Since the output of the network is a prediction of the input time series, MATLAB signifies both with the same variable [21].

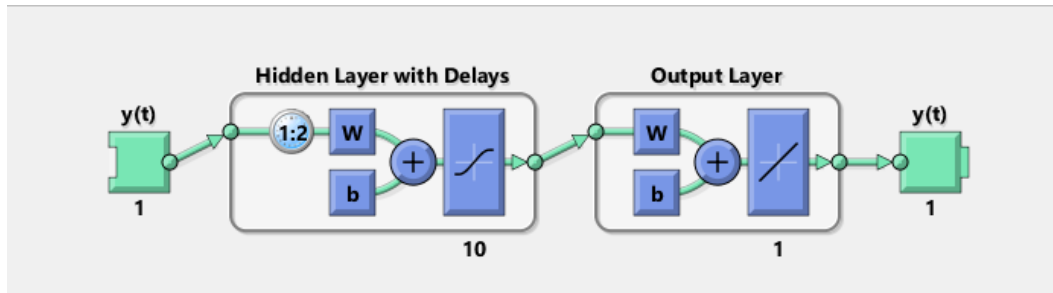


Figure 2. NAR Neural Network Architecture (Source: own work)

4.1 NAR Neural Network Training Output

The LM algorithm typically requires more memory but less time. Training automatically stops when generalization stops improving, as indicated by an increase in the mean square error of the validation samples [21]. The training progress using the LM algorithm stopped when the validation error increased for six iterations with Performance = 0.00281, Gradient = 8.36e-05, and Mu = 1.00e-05 at epoch 10. The term “epoch” represents the number of iterations during the training in which minimizing the error function was attempted.

The BR algorithm typically requires more time but can result in good generalization for difficult, small or noise datasets [21]. The training progress using the BR algorithm stopped according to adaptive weight minimization (regularization) with Performance = 0.00290, Gradient = 5.20e-05, Mu = 1.00e+10, Effective Number Parameters = 6.15, and Sum Squared Parameters = 2.56 at epoch 233.

The SCG algorithm requires less memory. Training automatically stops when generalization stops improving, as indicated by an increase in the mean square error of the validation samples [21]. The training for the network, in this case, was done using the SCG algorithm. The training progress using the SCG algorithm stopped when the validation error increased for six iterations with Performance = 0.00310, and Gradient = 0.00153 at epoch 29.

4.2 NAR Neural Network Best Validation Performance

The performance is evaluated after the training is completed, and then the values are generated. The performance using the LM algorithm was evaluated by taking epochs and mean square error (MSE), the average squared difference between outputs and targets. The best validation was taken at the point where both training and testing are equal. The best performance occurred at epoch 4

and was equal to 0.0029872. This showed that training and validation errors decreased until the highlighted epoch. It did not appear that any overfitting had occurred, because the validation error did not increase before this epoch.

The performance using the BR algorithm was described the performance of error at 233 epochs. In this case, the best validation was at epoch 44 and the value of validation performance was 0.0029002. The performance using the SCG algorithm was shown the performance of error at 29 epochs. The best validation was at epoch 23 and the value of validation performance was 0.0031863.

4.3 NAR Neural Network Training State

The gradient and validation check results using the LM algorithm was evident that the gradient was a locally decreasing function of epochs, which meant that if more iterations were involved, then the error should decrease. On the other hand, the validation checks increased when the number of epochs increased at epoch 10. The gradient result using the BR algorithm were showed that the gradient was a locally decreasing function of epochs. The gradient and validation check results were using the SCG algorithm were revealed that the gradient was a locally decreasing function of epochs, and the validation checks increased when the number of epochs increased at epoch 29. The term μ is the control parameter for the algorithm used to train the neural network. The choice of μ directly affects the error convergence. In case of Least Mean Square algorithm, μ depends on the maximum eigen value of input correlation matrix.

4.4 NAR Neural Network Error Histogram

The error histogram can give an indication of outliers, which are data points where the fit is significantly worse than that of most of the data. Using the LM algorithm, there were some training points and a few test points outside of the range. Using the BR algorithm, similarly some training points and a few test points were outside of the range. Using the SCG algorithm, some training points and a few test points were outside of the range as well. If the outliers are valid data points but are unlike the rest of the data, then the network is extrapolating for these points. This means more data similar to the outlier points should be considered in training analysis and that the network should be retrained.

4.5 NAR Neural Network Regression of the Training

The plots in Figure 3 demonstrated the regression of the training, validation, testing, and all data using the LM algorithm. The dashed line in each plot represents the perfect result outputs = targets, which can be seen on the regression diagrams. The solid line in each plot represents the best fit linear regression line between outputs and targets. On top of each plot, the regression R value measures the correlation between the outputs and the targets. If $R = 1$, this indicates that there is an exact linear relationship between the outputs and the targets. If R is close to zero, then there is no linear relationship between the outputs and the targets. In this case, the test results

using the LM algorithm showed that the regression R values were close to 0.9, which showed a reliable fit.

The plots in Figure 4 demonstrated the regression of the training, testing, and all data using the BR algorithm. In this case, the test results using the BR algorithm showed that the regression R values were close to 0.9, which showed a reliable fit. The plots in Figure 5 demonstrated the regression of the training, validation, testing, and all data using the SCG algorithm. In this case, the test results using the SCG algorithm showed that the regression R values were close to 0.87, which showed a reliable fit.

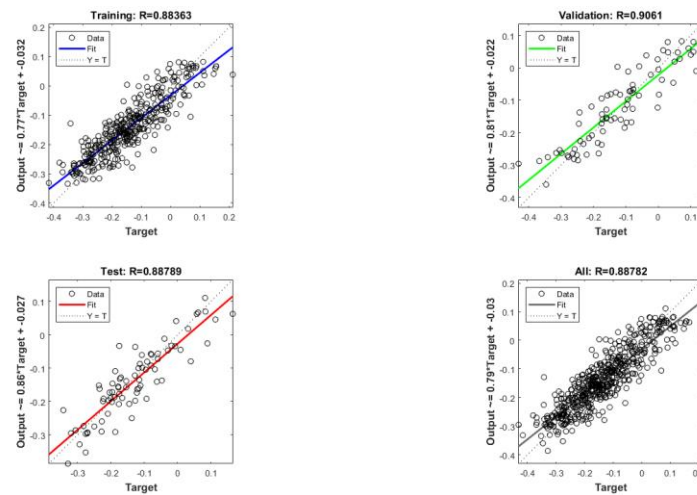


Figure 3. Regression Using Levenberg-Marquardt Algorithm (Source: own work)

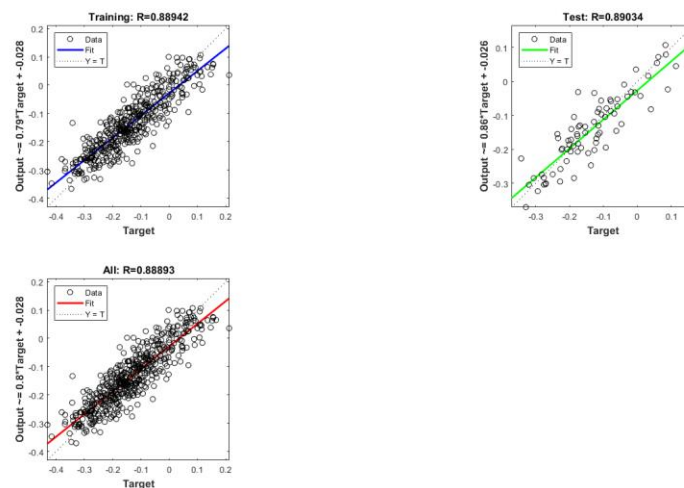


Figure 4. Regression Using Bayesian Regularization Algorithm (Source: own work)

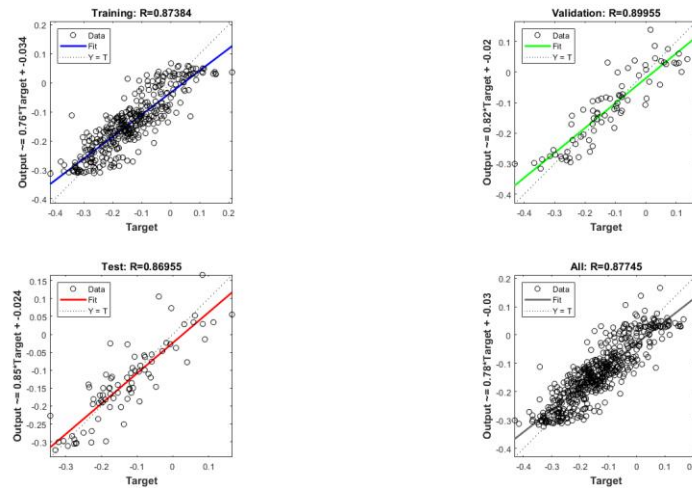


Figure 5. Regression Using Scaled Conjugate Gradient Algorithm (Source: own work)

4.6 NAR Neural Network Time-Series Response

The dynamic network time-series response plots were displayed in Figure 6 using the LM algorithm, showing that the outputs were distributed evenly on both sides of the response curve and the errors versus time were small in the training, testing, and validation subsets, indicating that the model reliably reflected the data. The dynamic network time-series response plots in Figure 7 using the BR algorithm and Figure 8 using the SCG algorithm displayed similar patterns, showing that the model reliably reflected the data.

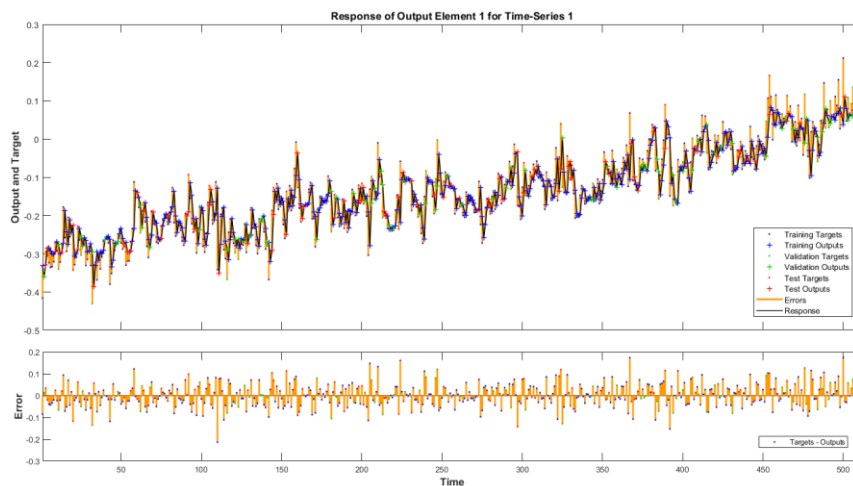


Figure 6. Network Time-Series Response Using Levenberg-Marquardt Algorithm (Source: own work)

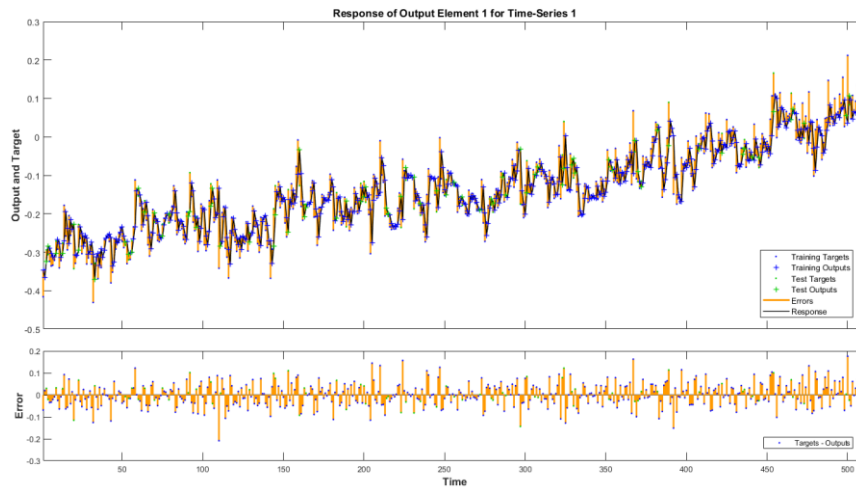


Figure 7. Time-Series Response Using Bayesian Regularization (BR) Algorithm (Source: own work)

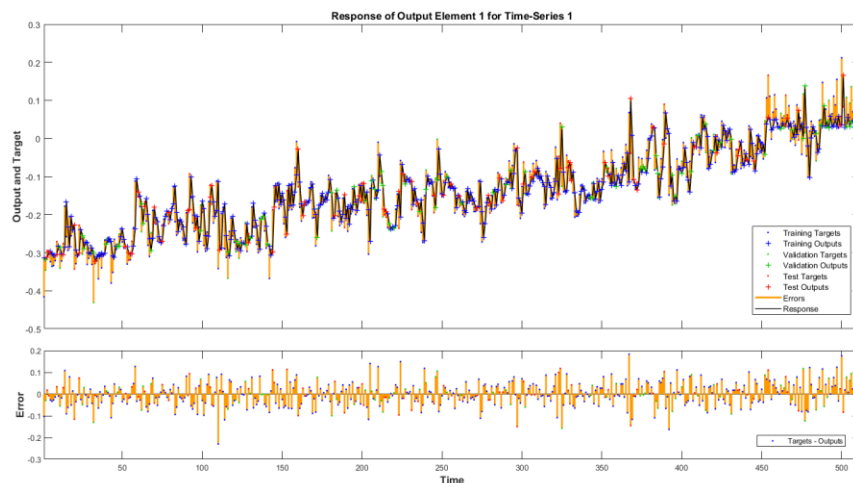


Figure 8. Time-Series Response Using Scaled Conjugate Gradient Algorithm (Source: own work)

4.7 NAR Neural Network Error Autocorrelation

The error autocorrelation function describes how the prediction errors are related in time. For a perfect prediction model, there should only be one nonzero value of the autocorrelation function, and it should occur at zero lag (this is the mean square error). This would mean that the prediction errors are completely uncorrelated with each other (white noise). If there is significant correlation in the prediction errors, then it should be possible to improve the prediction, perhaps by increasing the number of delays in the tapped delay lines.

The correlations using the LM algorithm (Figure 9), the BR algorithm (Figure 10), and the SCG algorithm (Figure 11), except for the one at zero lag, all fell approximately within the 95% confidence limits around zero, so the models seemed to be adequate. There are however some exceptions that suggest that the created neural network can be improved by retraining it or by

increasing the number of neurons in the hidden layer. If even more accurate results are required, retraining the network will change the initial weights and biases of the network and may produce an improved network after retraining.

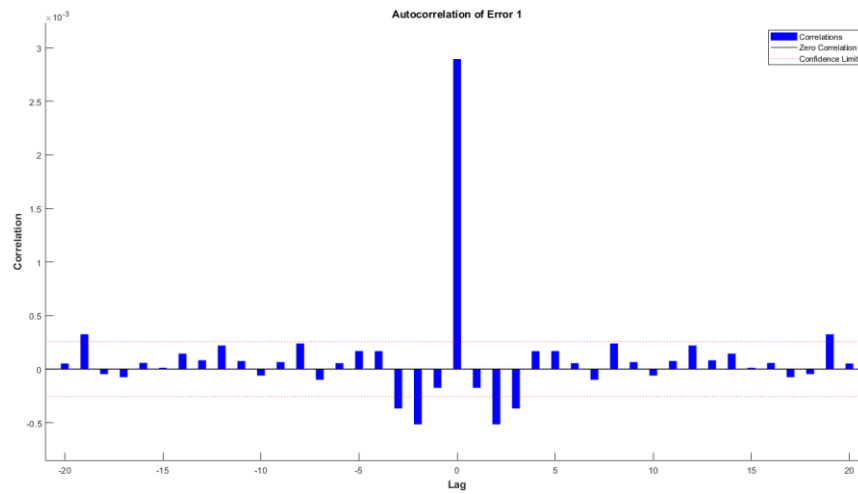


Figure 9. Error Autocorrelation Using Levenberg-Marquardt Algorithm (Source: own work)

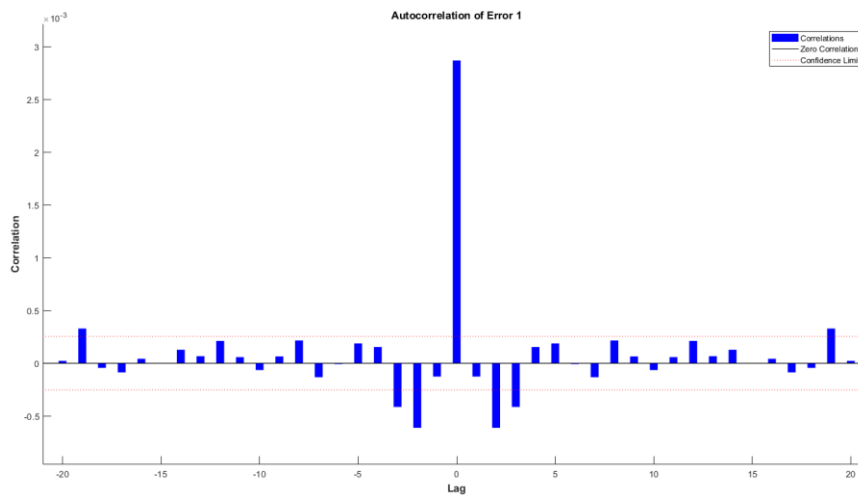


Figure 10. Error Autocorrelation Using Bayesian Regularization Algorithm (Source: own work)

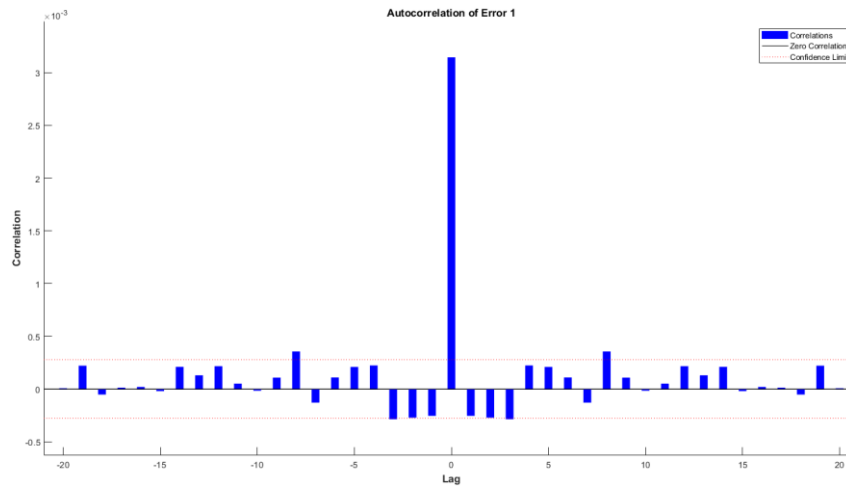


Figure 11. Error Autocorrelation Using Scaled Conjugate Gradient Algorithm (Source: own work)

4.8 Comparisons of the LM, BR, and SCG Algorithms

The purpose of this study was to evaluate the accuracy of the LM, BR, and SCG algorithms used in monthly mean sea level prediction. The performance was tested with the MSE and regression R values. Results and comparisons were summarized in Table 1. In this analytical work, the division of the time series dataset was 70% for the training data, 15% for the validation data, and 15% for the testing data. Randomly, 514 data samples were divided into 360 data for training, 77 data for validation, and 77 data for testing.

The training based on the LM algorithm yielded 88.78% accuracy for the all samples, 88.36% for training, 90.61% for validation, and 88.79% for testing. MSE reached $2.7738\text{e-}3$ during the testing. The training based on the BR algorithm yielded 88.89% accuracy for the all samples, 88.94% for training, and 89.03% for testing. MSE reached $2.6880\text{e-}3$ during the testing. The training based on the SCG algorithm yielded 87.75% accuracy for the all samples, 87.38% for training, 89.96% for validation, and 86.96% for testing. The MSE reached $3.1828\text{e-}3$ during testing.

The BR algorithm yielded low MSE compared to the LM and SCG algorithms. The regression R value computed by the BR algorithm was higher compared to the LM and SCG algorithms for monthly mean sea level forecast. These comparative results showed that the BR algorithm yielded higher accuracy than the LM and SCG algorithms. In terms of processing time, the BR algorithm (0:00:01) took more time compared to the LM (0:00:00) and SCG (0:00:00) algorithms during training but it converges faster.

Table1. Comparisons of the LM, BR, and SCG Algorithms

	Target Values	MSE			Regression R		
		LM	BR	SCG	LM	BR	SCG
Training	360	2.9016e-3	2.9002e-3	3.1281e-3	0.8836	0.8894	0.8738
Validation	77	2.9872e-3	---	3.1863e-3	0.9061	---	0.8996
Testing	77	2.7738e-3	2.6880e-3	3.1828e-3	0.8879	0.8903	0.8696
All	514				0.8878	0.8889	0.8775

(Source: own work)

5 Discussion and Conclusion

Many studies have concluded that the global sea level is rising at an increasing rate. While sea level rise is a relatively slow process, understanding past sea levels is important for the analysis of current and future sea level changes. In order to sustain these observations, research programs utilizing the existing data should improve our understanding and significantly narrow our projections of future sea level changes.

Sea level changes are driven by a variety of mechanisms operating at different spatial and temporal scales [28]. Therefore, forecasting future relative sea level changes at specific locations requires not just an estimate of global mean sea level changes but also estimates of the different processes contributing to global mean sea level changes, as well as of the processes contributing exclusively to regional or local mean sea level changes [29].

Tides, for example, are predictable because of the occurrence by a combination of forces, particularly the earth and the moon in relation to the sun. Local mean sea level can be measured at tide stations, which refers to the height of the water as measured along the coast relative to a specific point on land. To understand tides, therefore, it is important to understand the movements of the earth and the moon relative to the sun. However, most of the time, and in most places, tides are much more complex.

Prediction is a kind of dynamic filtering, in which the past values of one or more time series are used to predict future values. Currently, a neural network is one of the most popular machine learning methods as it is able to do prediction tasks in a more reliable manner. With time series data, lagged values of the time series can be used as inputs to a neural network, the NAR neural network was applied to time series prediction using its past values of a univariate time series in this study.

Empirically, the results revealed that the BR algorithm was the best training algorithm at its high regression R value and low mean square error compared to the LM and SCG algorithms for the NAR neural network. Hence, the NAR neural network can provide information important in the decision-making process related to the future sea level change impacts, and can be employed in forecasting the future performance for local mean sea level change outcomes.

For the further research tasks, the nonlinear autoregressive exogenous (NARX) neural network can be used to understand not only past information of the same time series (monthly mean sea level), but also current and past information of the externally determined time series that influences the time series of interest (i.e., temperatures, tides, etc.).

REFERENCES

- [1] J. E. Neumann, G. Yohe, R. Nicholls, and M. Manion, *Sea-Level Rise & Global Climate Change: A Review of Impacts to U.S. Coasts*, The Center for Climate and Energy Solutions prepared for the Pew Center on Global Climate Change, 38 pages, 2020. [Online] Available: <https://www.c2es.org/document/sea-level-rise-global-climate-change-a-review-of-impacts-to-u-s-coasts/> [Accessed: January 25, 2022]
- [2] J. A. Church, N. J. White, R. Coleman, K. Lambeck, and J. X. Mitrovica, Estimates of the regional distribution of sea level rise over the 1950-2000 period, *Journal of Climate*, 17(13), pp. 2609-2625, 2004.
- [3] J. A. Church, and N. J. White, A 20th century acceleration in global sea-level rise, *Geophysical Research Letters*, 33, L01602, pp. 1-4, 2006.
- [4] J. A. Church, and N. J. White, Sea-level rise from the late 19th to the early 21st century, *Surveys in Geophysics*, 32, pp. 585-602, 2011.
- [5] IPCC, *Climate Change 2014: Synthesis Report*, Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Core Writing Team, R.K. Pachauri and L.A. Meyer (eds.)], IPCC, Geneva, Switzerland, 151 pages, 2014. [Online] Available: <https://www.ipcc.ch/report/ar5/syr/> [Accessed: January 25, 2022]
- [6] U.S. Global Change Research Program (USGCRP), *Climate science special report: Fourth National Climate Assessment*, Volume I, [D. J. Wuebbles, D. W. Fahey, K. A. Hibbard, D. J. Dokken, B. C. Stewart, and T. K. Maycock (eds.)], U.S. Global Change Research Program, Washington, DC, USA, 470 pages, 2017. [Online] Available: <https://science2017.globalchange.gov/> [Accessed: January 25, 2022]
- [7] J. A. Church, N. J. White, T. Aarup, W. S. Wilson, P. L. Woodworth, C. M. Domingues, J. R. Hunter, and K. Lambeck, Understanding global sea levels: past, present and future, *Sustainability Science*, 3, pp. 9-22, 2008.
- [8] A. Cazenave, and W. Llovel, Contemporary sea level rise, *Annual Review of Marine Science*, 2, pp. 145-173, 2010.
- [9] A. Cazenave, and G. Le. Cozannet, Sea level rise and its coastal impacts, *Earth's Future*, 2, pp. 15-34, 2013.
- [10] B. P. Horton, R. E. Kopp, A. J. Garner, C. C. Hay, N. S. Khan, K. Roy, and T. A. Shaw, Mapping sea-level change in time, space, and probability, *Annual Review of Environment and Resources*, 43, pp. 481-521, 2018.
- [11] S. A. Kulp, and B. H. Strauss, New elevation data triple estimates of global vulnerability to sea-level rise and coastal flooding, *Nature Communications*, 10:4844, pp. 1-12, 2019.
- [12] M. Haasnoot, J. Kwadijk, J. van Alphen, D. Le Bars, B. van den Hurk, F. Diermanse, A. van der Spek, G. O. Essink, J. Delsman, and M. Mens, Adaptation to uncertain sea-level rise; how uncertainty in Antarctic mass-loss impacts the coastal adaptation strategy of the Netherlands, *Environmental Research Letters*, 15, 034007, pp. 1-15, 2020.
- [13] G. Foster, and P. T. Brown, Time and tide: analysis of sea level time series, *Climate Dynamics*, 45(1-2), pp. 291-308, 2014.
- [14] H. Visser, S. Dangendorf, and A. C. Petersen, A review of trend models applied to sea level data with reference to the “acceleration-deceleration debate”, *Journal of Geophysical Research: Oceans*, 120(6), pp. 3873-3895, 2015.
- [15] D. Bolin, P. Guttorp, A. Januzzi1, D. Jones, M. Novak, H. Podschwit, L. Richardson, A. S’arkk’a, C. Sowder, and A. Zimmerman, Statistical prediction of global sea level from global temperature, *Statistica Sinica*, 25, pp. 351-367, 2015.
- [16] P. K. Srivastava, T. Islam, S. K. Singh, G. P. Petropoulos, M. Gupta, and Q. Dai, Forecasting Arabian sea level rise using exponential smoothing state space models and ARIMA from

- TOPEX and Jason satellite radar altimeter data, *Meteorological Applications*, 23, pp. 633–639, 2016.
- [17] O. Makarynsky, D. Makarynska, M. Kuhn, and W. E. Featherstone, Predicting sea level variations with artificial neural networks at Hillarys Boat Harbour, Western Australia. Estuarine, *Coastal and Shelf Science*, 61(2), pp. 351–360, 2004.
- [18] A. Braakmann-Folgmann, R. Roscher, S. Wenzel, B. Uebbing, and J. Kusche, *Sea level anomaly prediction using recurrent neural networks*, In Proceedings of the 2017 conference on Big Data from Space, pp. 297–300, 2017. [Online] Available: <https://arxiv.org/abs/1710.07099> [Accessed: January 25, 2022]
- [19] W. Wang, and H. Yuan, A Tidal Level Prediction Approach Based on BP Neural Network and Cubic B-Spline Curve with Knot Insertion Algorithm, *Mathematical Problems in Engineering*, Volume 2018, Article ID 9835079, 9 pages, 2018. [Online] Available: <https://doi.org/10.1155/2018/9835079> [Accessed: January 25, 2022]
- [20] N. Bruneau, J. Polton, J. Williams, and J. Holt, Estimation of global coastal sea level extremes using neural Networks, *Environmental Research Letters*, 15 (7), 074030, 11 pages, 2020.
- [21] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Deep learning Toolbox™: getting started guide*, Natick, MA: The MathWorks, Inc., 2019.
- [22] K. Levenberg, A method for the solution of certain non-linear problems in least squares, *Quarterly of Applied Mathematics*, 2(2), pp. 164–168, 1944.
- [23] D. W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, pp. 431–441, 1963.
- [24] H. P. Gavin, *The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems*, Department of Civil and Environmental Engineering Duke University, 19 pages, 2020. [Online] Available: <http://people.duke.edu/~hpgavin/ce281/lm.pdf> [Accessed: January 25, 2022]
- [25] D. J. C. MacKay, Bayesian Interpolation, *Neural Computation*, Vol. 4, No. 3, pp. 415–447, 1992.
- [26] E. Sariev, and G. Germano, Bayesian regularized artificial neural networks for the estimation of the probability of default, *Quantitative Finance*, Vol. 20, No. 2, pp. 311–328, 2020.
- [27] M. F. Moller, A scaled conjugate gradient algorithm for fast supervised learning, *Neural Networks*, Volume 6, Issue 4, pp. 525–533, 1993.
- [28] R. E. Kopp, B. P. Horton, A. C. Kemp, and C. Tebaldi, Past and future sea-level rise along the coast of North Carolina, USA, *Climatic Change*, 132, pp. 693–707, 2015.
- [29] R. E. Kopp, C. C. Hay, C. M. Little, and J. X. Mitrovica, Geographic variability of sea-level change, *Current Climate Change Reports*, 1, pp. 192–204, 2015.