Predicting Primary Water Levels Using Backpropagation and General Regression Neural Networks

Carlos Mendieta, Mario Garcia, Carl Steidley

Texas A&M University-Corpus Christi

Abstract

This project applied two Artificial Neural Network models (Backpropagation and the General Regression Neural Network (GRNN)) to predict primary water levels at a single port on the Texas coast. The data for this project was provided by the Division of Nearshore Research and is collected hourly from several ports along the Texas coast. Important variables needed for making tide prediction were determined. The networks were then built, trained, and tested. The results obtained from each neural network are presented.

1.0 - Introduction

Tide prediction requires knowledge of a number of subjects. Oceanography, Meteorology, Mathematics, and Physics are a few of the major disciplines that are involved. Fortunately, applying Artificial Neural Networks (ANN) to the problem of tide prediction (or more specifically, primary water level prediction) reduces most of the work down to simple pattern recognition.

2.0 - Tidal Analysis

A study of tidal patterns and what affects them was needed to determine the ANN inputs [6]. The primary tide generating forces are the gravitational pulls of celestial objects, namely the Sun and the Moon. Their movements cause large bodies of water to swell and recede according to their relative position to the earth and to each other. Though Newton's Law of Gravitation does not directly describe this phenomenon, it can help explain it. Basically, the law states that the attraction between objects has a force proportional to their masses and inversely proportional to the square of the distances between the objects. Simply put, the closer the Sun or the Moon is to the earth, the greater the force of attraction. Now recall that one of the primary properties of any liquid is its viscosity. Ocean water has a very low resistance to flow. Tides are the direct response of the Sun and Moon's exertion of a force against the Earth's large bodies of water.

The Moon's effects on the oceans are strongest at an interval of nearly two weeks. So, approximately twice a month, the Moon exerts the strongest gravitational pull on the Earth. Just after the full and new moon occur, tides generally have their greatest ranges from low to high water [1]. The heights of those ranges are called spring tides while the lowest of the ranges are called neap tides. The figures 1 and 2 are helpful in visualizing the effects of the moon on the earth's waters.

The Sun also plays a part in the generation of tides. Its gravitational strength is somewhat less than the Moon's pull. This is due to the fact that it is more distant than the Moon. Although its effect on the water is not as great as the Moon's, when acting in conjunction with the Earth and the Moon, it can clearly be seen to make a difference in tidal patterns. Attraction is strongest at times when the Sun, Moon, and Earth are lined up as seen on figure 3.

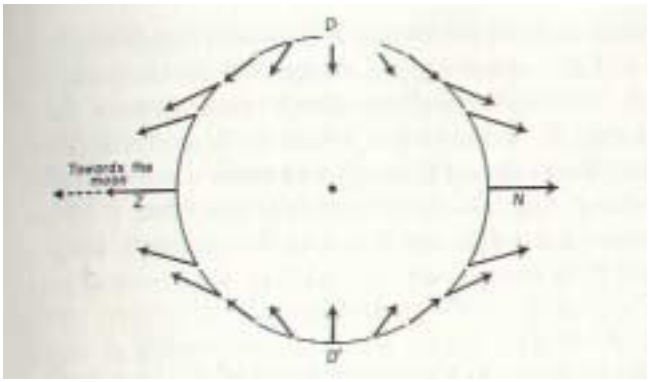

Figure 1. Gravitational pull of the Moon.

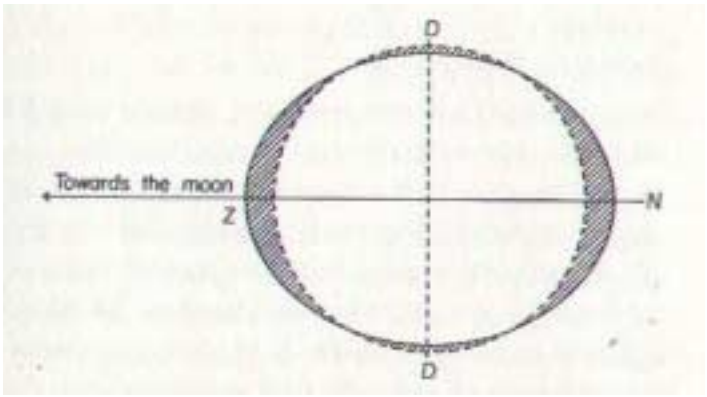

Figure 2. Effects of the Moon's gravitational pull on the earth's waters.

An important side note: it is the Sun's and the Moon's attractions that considerably reduce the amount of data that can be used by a neural network for prediction in this project. The data being used does not include their effects, though they must be taken into account. A month's worth of data becomes too much. Variables, such as wind speed and direction could remain constant at a port for weeks (unlikely, but still possible), but the primary water levels would still shift. This is because of the gravitational pull. To reduce the range of this effect as much as possible, it is necessary to take smaller data samples for training and test sets. It turns out, for the most accurate predictions, about 24 - 36 hours for a training set is ideal. Beyond that, the error in prediction increases and accuracy suffers immensely.

On the other hand, should this project have been implemented on the effects of gravitational pull alone, the same sort of problems with errors would occur. The Earth will, approximately, reach the same position in its orbit every year, just as the Moon repeats its orbit around the Earth

every day. Were the primary water levels solely dependent on celestial pulls and movements, then it would be easy to make predictions for years and years to come. However, other conditions, as described below, are not as constant as the suns and moons movements.

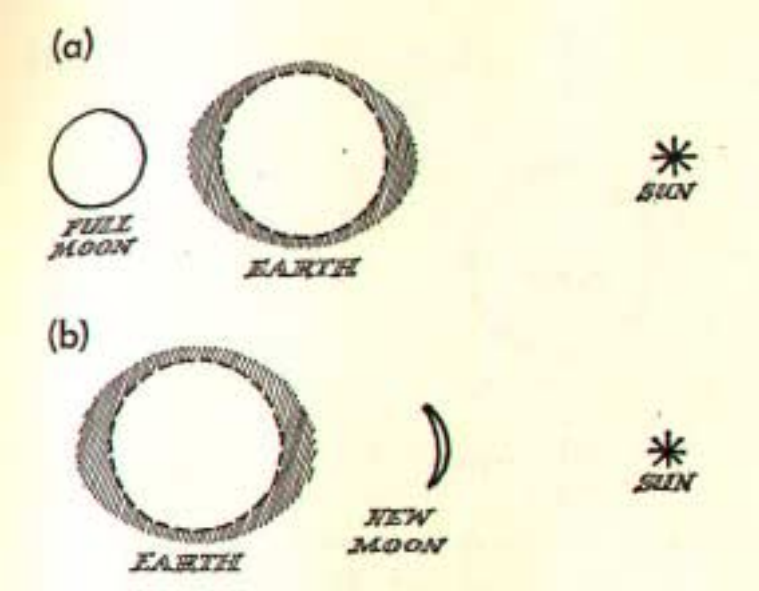

Figure 3. Gravitational pull is strongest when the Earth, Moon, and Sun are aligned.

Besides the obvious effect of gravity on the waves there are other tide affecting variables to consider. The wind plays a large role in tide generation, especially along the Texas coast. Just as gravity pulls large bodies of water, the wind can push water enough to affect tidal patterns. The more violent the weather, the stronger the winds, the greater the dramatic effect on the water. The wind's speed and direction are, in particular, important variables in determining water levels. Air and water temperatures also must be considered as they often provide some indication of weather.

Barometric pressure is also thought to have some noticeable sway over the tides. A high barometric pressure can be an indication of depressed sea levels. An inch change on a mercury barometer can correspond to sea level variation of around a foot [2]. There is clearly a relationship present between water levels and barometric pressure.

A tidal cycle for one day may appear to be identical to that of one a year before, but variable combinations, such as barometric pressure and air temperature, may be very different. This looks like a potential problem for a neural network that has already learned that certain combinations of values produce certain results; however the network learns combinations that consist of both inputs and outputs during its training. This means that the network can learn to recognize that there is more than one way to solve a problem.

2.1- Data Collection

Collection of data was the next step in predicting primary water levels. The Division of Nearshore Research (DNR) is part of the Conrad Blutcher Institute for Surveying and Science at Texas A&M – Corpus Christi. One of their activities is the collection of real-time data from several ports along the Texas coastline. Some of the data series descriptions include primary water levels, air temperature, wind speed, wind direction, and many others. The data has been collected for a number of years and can be accessed from an online database [8].

As this project emphasizes the prediction ability for a single port, it was necessary to look through the data to determine which ports had the most consistent entries. Some ports were found to have entries missing for a number of data series. Morgan's Point was found to be one of the more reliable sources of data collection and is the port that is used in all further process and data descriptions in this paper.

Unfiltered, a resulting query from the online database would have some basic port information and variable descriptions at its head. Figure 4 shows what the data looks like after removing some of the unnecessary information.

# Time	503-pwl	503-atp	503-wtp	503-wsd	503-wdr	503-bpr
2000264+0000	1.804	28.1	35.3	4.600	145	758
2000264+0100	1.821	27.9	35.4	3.400	158	758
2000264+0200	1.841	27.8	35.5	2.100	151	758

Figure 4. Raw data from online database [8].

2.2 - Data Analysis

After obtaining the data from the online database, it had to be examined for quality. While Morgan's Point is one of the better data collection stations, there were still sporadic areas of inadequacy. There are, in some sections of the data, areas that are incomplete. Supplying an Artificial Neural Network (ANN) with insufficient data when training can lead to erroneous prediction during testing, so it was necessary to find reasonable methods that could be used to reduce such potential problems.

If gaps in data were large, for example a few days worth of missing wind speed data, those days were discarded. Use of these data as training or test sets in an artificial neural network would corrupt the learning process since over a long period of time a variable can display a large range of values. In the space of a day alone, water levels can go from one meter to two meters and back again.

If the gaps were less substantial, a few hours, maybe even up to a day, then an average between the last recorded instance and the next recorded instance would be taken for that attribute. For example, figure 5 shows that the air temperature, wind temperature, and barometric pressure are missing between 1:00 PM and 4:00 PM.

# Time	503-pwl	503-atp	503-wtp	503-wsd	503-wdr	503-bpr
2000029+1200	1.250	2.3	10.9	6.100	327	770
2000029+1300	1.202	NA	NA	6.900	330	NA
2000029+1400	1.167	NA	NA	7.500	335	NA
2000029+1500	1.103	NA	NA	7.700	341	NA
2000029+1600	1.117	2.1	10.8	7.900	346	772

Figure 5. Data set with missing elements.

Filling in the data with the averages between the two times would produce the following results (see figure 6):

# Time	503-pwl	503-atp	503-wtp	503-wsd	503-wdr	503-bpr
2000029+1200	1.250	2.3	11.0	6.100	327	770
2000029+1300	1.202	2.1	10.9	6.900	330	771
2000029+1400	1.167	2.1	10.9	7.500	335	771
2000029+1500	1.103	2.1	10.9	7.700	341	771
2000029+1600	1.117	2.1	10.8	7.900	346	772

Figure 6. Corrected data set.

This method operates under the assumption that, from one hour to the next, values of certain variables (like wind speed) will not change drastically. Instead they will follow a smooth rate of increase or decrease. This will, generally, hold true for any of the variables over a small period of time.

The folly in this method is the lack of ability to comprehend natural phenomenon such as wind gusts or other unpredictable, yet natural occurrences. For the most part, however, it can produce believable data where there was previously none. Attempting to apply the same methods to larger sections of missing data could result in larger error. As the length of time stretches between data collection, accounting for wider ranges of missing values need to be made. It has been established that values do not, generally, change drastically over a period of twenty-four hours; however, it can be seen that in that time period, values for the variables in question can rise and fall several times. Neural Networks need accurate data in order to make accurate predictions. Large amounts of sequentially missing data will produce erroneous results.

3.0 - Building the Backpropagation Neural Network

The first networks built contained a single layer with only 3 input nodes and a single output node. These were designed to be a prototype to determine the optimal size of the training sets. The initial input variables were wind speed, wind direction, and air temperature. The output value was primary water levels. Once the validity of the network was confirmed, two more input variables were added (water temperature and barometric pressure) and predictions were recorded.

Actual predictions were made with several Backpropagation networks that had 5 input nodes and one or two hidden layers. The basic approach was to attempt the network with one hidden layer and compare its performance with a network with two hidden layers. Had there been a significant increase in performance of the one with the hidden layer, more processing elements would have been added to the layers. This would have been re-tested and compared for performance. As it was, the only significance in the resulting predictions was the time it took to train the networks.

The first network, shown in figure 7, was tested several times using different learning rules and squashing functions. The learning rule is a guideline for performing the weight (connection strength) updates, while the squashing function is a method for transforming the input (Hyperbolic Tangent Transfer Function). For the small samples that this network was limited to, the difference was not highly noticeable between the different learning rule options. The difference was noticed when looking at the networks' learning capabilities with more layers and the transfer functions. The number of cycles given to the network between weight updates (the epoch) was set to 16. This means that the network evaluated 16 data entries before the network was able to adjust its connection strength. The test set consisted of 12 hours worth of data.

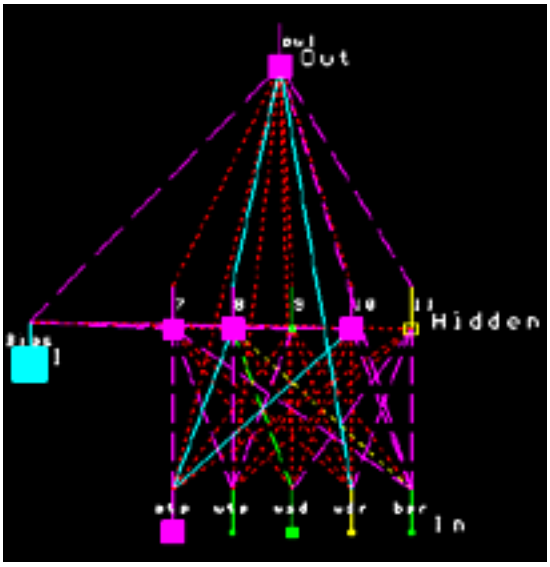

Figure 7. Screenshot of a Backpropagation network from NeuralWare's NeuralWorks [3,4,5].

Feeding the network random data entries (24 hours worth of data) for a number of cycles results in the error shown in figure 8. The Root Mean Square (RMS) is an error measuring technique that takes the sums of the squares of each processing element's (each node in the input, output, and hidden layers) error. It then takes the square root of that average [5]. An ideal RMS error approaches zero. Correlation is another tool that is used here. The closer to one the correlation value gets, the more similar the predicted and the actual water levels are to each other.

Figure 8. Error tools from NeuralWare's NeuralWorks.

This graph shows a high correlation. The actual results can be better seen in figure 9 where a graph that represents predicted and actual values is presented. For the most part, the predicted water levels follow, rather closely, the actual water levels. The most noticeable error, between the fourth and sixth hours can be explained by looking at the training and test sets that were used. Figure 10 shows the results of the same network attempting to predict the next 24 hours after the training set.

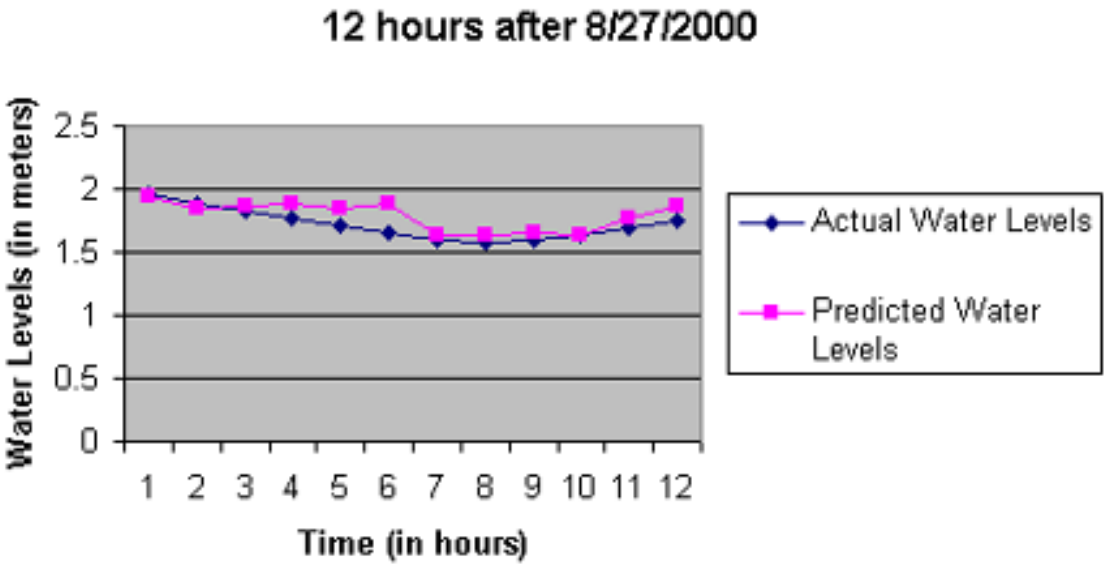

Figure 9. Prediction graph (for 12 hours) from a Backpropagation network.

Adding one more hidden layer to the network decreased the training time by half. The accuracy of such a network, while slightly less than that of the original network, is still very good. Increasing the training time would improve the accuracy of predictions and still manage to make the training of the network considerably more efficient than the original. Figure 11 shows the results for a 24 hours prediction.

4.0 - Building the General Regression Neural Network (GRNN)

The General Regression Neural Network excels at prediction [9]. Its background has deep roots in statistics and uses a Gaussian-based technique in order to determine a conditional mean Y given X, where Y is the output and X the input [3]. To briefly describe the inner process: the GRNN uses a technique called Parzen estimation in order to approximate a joint probability density function (pdf). A probability density function is the probability distribution associated with a continuous random variable. The pdfs are used to determine which of the processing elements "win" when an input vector is put through the network and matches that pdf.

Processing elements are competitive and only the ones that win produce an output (Network Selection).

One of the main advantages of using a GRNN is its speed during its training phase. It trains much faster than the Backpropagation network. The initial test set produced good results after a brief training period and shows the network to be about as accurate as the Backpropagation network as well.

Convergence to an optimal regression surface with an increasingly large number of samples is simply saying that the more training data the GRNN is given, the better it will perform in its estimations. However, another of the advantages mentioned above claims that the network will also be effective with sparse data as well. Regression, as a general statistical technique has the ability to make decent estimations for relationships with non-dense or data that is few and far apart. The same principles apply to the GRNN model. Figure 12 shows the GRNN predictions for 24 hours after the training set from 8/27/2000.

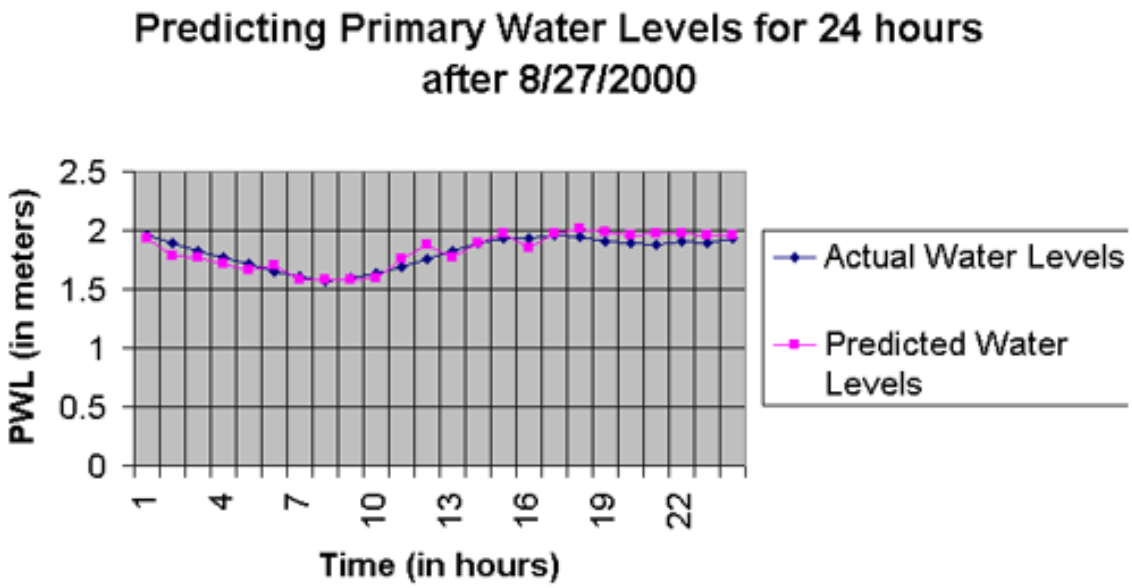

Figure 10. Prediction graph (for 24 hours) from a Backpropagation network

5.0 – Conclusions

Each of the networks used had definite and noticeable advantages in both speed and accuracy. The Backpropagation network, even when built with a single layer produced exceptional results that were close to the actual primary water levels. Adding a layer to the same network resulted in a network that learned approximately twice as fast, proving that efficiency is increased with more complex networks of this type.

The General Regression Neural Network's main advantage over the Backpropagation model is the incredible speed at which it learns. Its accuracy is only slightly less than the end result of the

Backpropagation's predictions of the same data and this problem can be eliminated with additional training.

For the amount of data being used for the training of these networks, either model would be appropriate for practical use; however, as larger and larger data sets are employed (say, eventually, years) the General Regression Neural Network may provide decent results in less time.

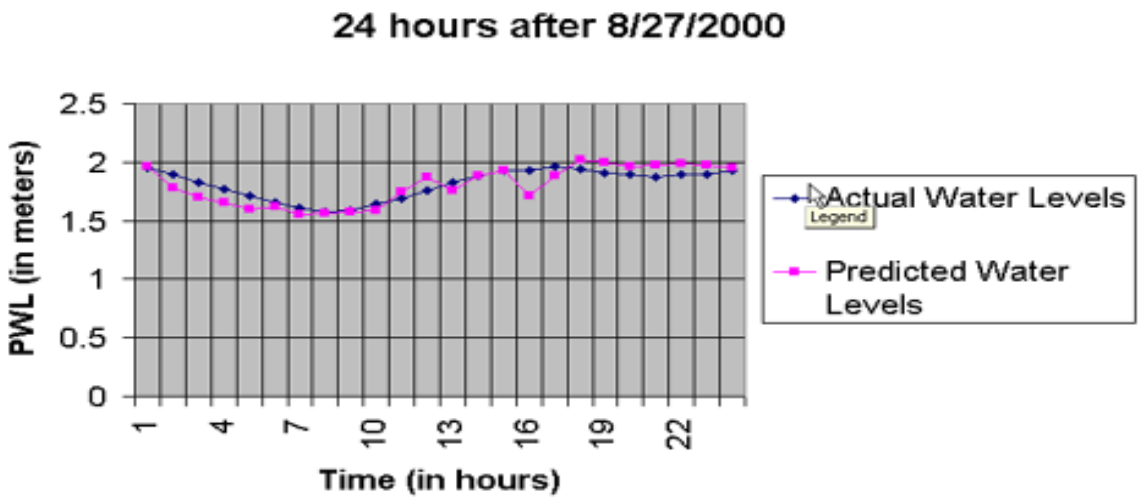

Figure 11. Prediction graph (for 24 hours) from a multi-layer network

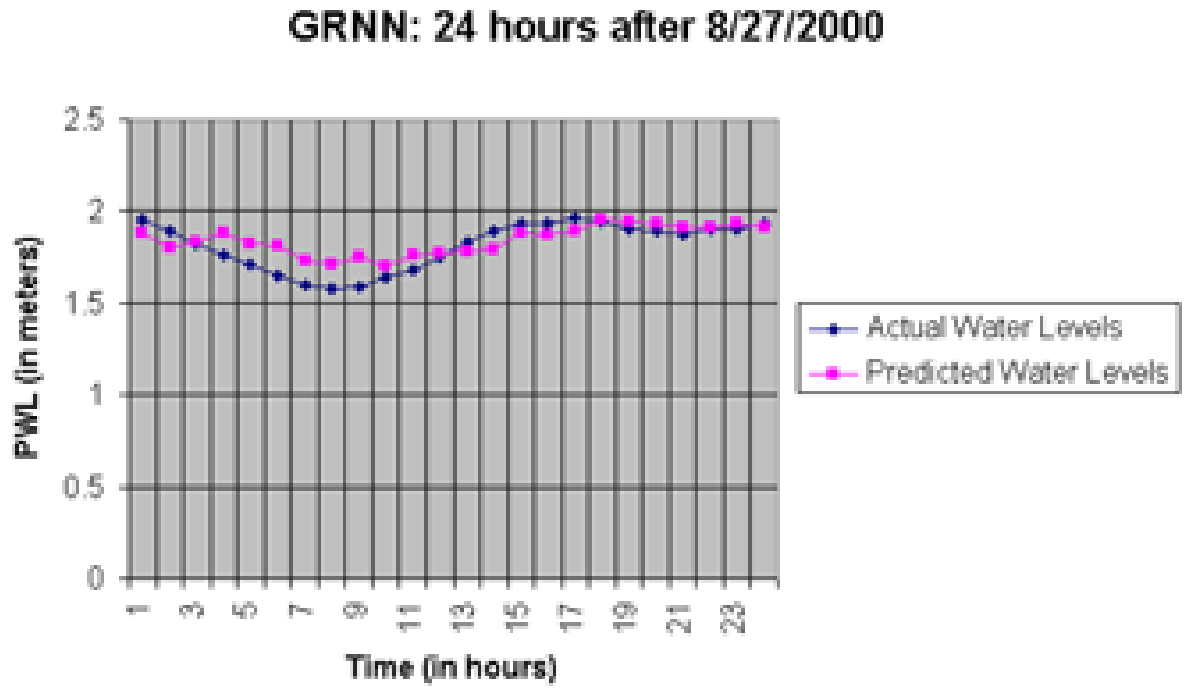

Figure 12. Prediction graph (for 24 hours) from a GRNN.

6. Bibliography

[1] Clancy, Edward P. *The Tides*. Doubleday & Company, NY, 1968.
[2] Defant, Albert. *Ebb and Flow*. The University of Michigan Press, 1958.
[3] NeuralWorks. *Neural Computing*. NeuralWare, Inc. 1997.
[4] NeuralWorks. *NeuralWorks Reference Guide*. NeuralWare, Inc. 1997.
[5] NeuralWorks. *Using NeuralWorks*. NeuralWare, Inc. 1997.
[6] Pugh, D.T. Tides, *Surges and Mean Sea-Level*. John Wiley and Sons. Great Britain. 1987.
[7] Schalkoff, Robert J. *Artificial Neural Networks*. McGraw-Hill. 1997.
[8] Conrad Blutcher Institute: Division of Offshore Research.
http://dnr.cbi.tamucc.edu/pquery,
[9] Properties of GRNNs
http://www.irecall.com/vve/grnn.htm

Biographical Information

MARIO A GARCIA
Mario A Garcia is an assistant professor at Texas A&M University-Corpus Christi. Dr. Garcia received a B.S. degree in Electrical Engineering from Tecnologico de Saltillo, Mexico, He received a M.S. in Electrical Engineering from Tecnologico de la Laguna, Mexico, He received a M.S. in Artificial Intelligence from ITESM, Mexico, and he received his Ph.D. in Computer Science from Texas A&M University. garciam@falcon.tamucc.edu

CARL STEIDLEY
Carl Steidley is Professor of Computer Science and Chair of Computing and Mathematical Sciences. His interests are in the applications of artificial intelligence, real-time computing, and robotics. He taught computer science at Southeastern La. Univ., Central Washington Univ., and Oregon Institute of Technology. He has research at NASA Ames Research Center, Oak Ridge Natl. Labs, and Electro Scientific Industries in Portland.

CARLOS MENDIETA
Carlos Mendieta received a M.S. from Texas A&M University-Corpus Christi. He is working as a software analyst in a software development company in Corpus Christi TX