Evaluation of a Meteorological Prediction Model with the Correlation Between Rainfall and Sea Level using Particle Swarm Optimization

Wasfi Qordowi
School of Meteorology, Climatology and Geophysics Agency
STMKG
Tangerang Selatan, Indonesia
wasfi.qordowi@stmkg.ac.id

Abstract—**Rainfall in the sea is very difficult to predict. there are many factors that become parameters in determining the estimated rainfall in the sea, by taking the correlation between rainfall and sea level will be able to predict rainfall with particle swarm optimization, One such technique which can be conveniently used to determine the model parameters accurately. This robust optimization technique has already been applied to improve the performance of artificial neural networks for time series prediction. This study uses particle swarm optimization technique to determine the parameters of an exponential autoregressive model for time series prediction. The model is applied for annual rainfall prediction and it shows a fairly good performance in comparison to the statistical ARIMA model. The research was conducted in Semarang, using data from rainfall, precipitation, and sea level rise in 2016. Network training by using one unit of input layer, two hidden layer units, and one unit of output layer. The first hidden layer with 10 neurons and the second hidden layer used 5 neurons. The best results on the training and testing of the network by using the parameter learning rate 0.3 and a momentum 0.6. The results obtained in the training get a percentage value of correlation is 79.0% and in the testing process to get the percentage correlation is 77.5%.**

Keywords: particle swarm optimization, rainfall, sea level rise

I. INTRODUCTION

Linear time series models have drawn much attention due to their relative simplicity in understanding and implementation. The autoregressive integrated moving average (ARIMA) model [1] is one such model that has seen many applications in time series forecasting [2, 3, 4]. However, many practical time series including rainfall show nonlinear behaviour due to which nonlinear methods are employed for their prediction. A host of nonlinear statistical models have been described in the literature for predicting volatility changes in time series [5]. Artificial neural networks (ANNs) approach has been suggested as an alternative technique for nonlinear time series forecasting and it has gained immense popularity in the recent time [6, 7, 8]. Support vector regression (SVR) is another new approach being successfully applied in time series prediction [9]. Various hybrid techniques have also been tried in the recent years for efficient prediction. Particle swarm optimization (PSO) approach, in particular, has been utilized in combination with different time series models to improve the performances of these models. PSO has been effectively used by some researchers as an alternative to the Backpropagation (BP) algorithm for training ANN models [10, 11]. Asadi *et al.* [12] combined PSO with ARIMA model and reported that this hybrid method exhibited better prediction results compared to an ARIMA model itself. Cui and Jiang [13] used a binary Particle Swarm Optimization (BPSO) to increase the predictive accuracy of a local linear time series prediction model. In the present study, an updated PSO is employed for determining the parameters of an exponential regression model so as to improve its prediction accuracy for annual rainfall prediction.

II. PARTICLE SWARM OPTIMIZATION

PSO Particle swarm optimization (PSO) is a swarm intelligence class method which was invented in the mid 1990s [14]. The PSO is a population-based stochastic optimization algorithm and recently, it has acquired wide applications in optimizing design problems because of its simplicity and ability to optimize complex constrained objective functions in multimodal search spaces.

In the PSO each potential solution is referred as a particle and each set of particles composes a population. Each particle maintains the position associated with the best fitness ever experienced by it in a personal memory called pbest. Besides,the position associated with the best value obtained so far byany particle is called gbest. In any iteration, the pbest and gbest values are updated and each particle modifies its velocity to move toward them stochastically. This concept canbe formulated as:

$$v_i^{t+1} = \omega\, v_i^t + c_1\, r_1 (pbest_i - x_i^t) + r_2\, c_2\, (gbest - x_i^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}; i = 1,2,\ldots,n \quad (2)$$

v = particle velocity
x = particle position

ISBN 978-602-53524-0-9

n	=	number of particles
t	=	number of iterations
w	=	inertia weight factor
c1,c2	=	cognitive and social acceleration factors
n1,n2	=	uniformly distributed random numbers

After the application of PSO algorithm, the algorithm continues with the evaluation phase. The function encoded by each particle in the population is evaluated on the fitness cases set. For each input set, the result of the function is compared with the expected output of that fitness case and the fitness of the particle is adjusted according to the results. In this paper, the fitness of a particle is computed as the sum of the absolute difference between the expected output value and the value returned by the particle, over all fitness cases:

$$\text{fitness} = \sum_{c\in C} |expected_c - computed_c| \qquad (3)$$

The best particles are the ones that return better approximations of the expected values the ones with a lower fitness. A particle A is better than another particle B if the fitness of A is lowers than the fitness of B or fitness of A is equal to the fitness of B and A has fewer nodes than B. The evolution-evaluation-selection cycle is repeated until a stopping criterion is met. We use a mixed stopping criterion: the algorithm stops when a maximum number of populations have been reached, or when an acceptable particle was found. The solution designated by a run of the PSO algorithm is the best particle throughout the search space.

III. PSO-EXPAR MODEL FOR RAINFALL PREDICTION

Among the different statistical methods for time series prediction, ARIMA model is by far the most popular one. However, this method has certain limitations including its inherent assumption of linearity. In most cases, a rainfall time series is nonlinear in nature. A nonlinear model namely, the exponential autoregressive (EXPAR) model [15], is used in this study to predict an annual rainfall time series. This model is capable of handling the non- Gaussian characteristics of a time series [16]. This model can be represented as:

$$y_t = \sum_{i=1}^{p} \{ \propto_i + \varphi_i \exp(-\lambda -_{t-1} 2)\} y_{t-1} + \varepsilon_t \qquad (4)$$

where yi (i = 1, 2,3......p) is a vector of the predictor variables and p is the order of the model; λ is a scaling constant in the range (0,1) and εt is a white noise operator with mean zero and variance one. In the above equation, if yt-1 is large then the model turns into an autoregressive model. The coefficients αi and ϕi are linear and hence the model may be termed as linear – in-the-parameters. Selection of appropriate model parameters is most crucial for efficient prediction of a time series. Further, choosing the best performing model (i.e. choosing the order p in this case) is a cumbersome task. In this study, the robustness of PSO algorithm is used to accomplish both the above tasks. The model parameters are iteratively optimized using the updated PSO algorithm with the objective of minimizing the sum of squared error (SSE) between the observed and predicted values of annual rainfall. The order p of the model is chosen on the basis of Akaike information criterion (AIC) and Bayesian information criterion (BIC) [17].

IV. RESEARCH METHOD

a. Data

Data consists of 2016. The data used are rainfall and obtained from BMKG Climatology Semarang, and sea level obtained from the NOAA.

b. Software

The software used in this study consisted of Microsoft Excel 2016 and Matlab R2016a. Microsoft Excel to process monthly average values and Matlab R2016a to model particle swarm optimization.

c. Research mind

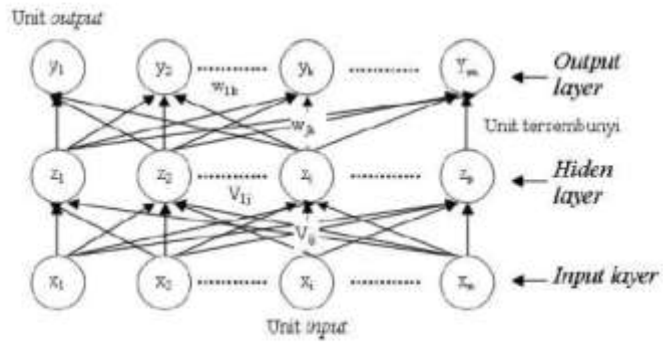

Figure 1. Step of Training System and Learning Prediction

Step 0: initialize weights (you should set a small random value)

Step 1: if conditions are not achieved, do steps 2 to 9

Step 2: for training, do steps 3 to 9

Forward propagation

Step 3: each input unit (xi, i = 1, ..., n) accepts the signal and delivers the signal to all of the above layer units including hidden layers

Step 4: each unit is hidden (zi, i = 1, ..., p) add the weight of the input signal

$$z\_inj = v1j + \Sigma xivijpi = 1 \qquad (4)$$

with, $v0j$ as the bias in the hidden unit apply the activation function to calculate the output signal, zj = f (z_inj), and send the signal to all units in the layer above it (output unit).

Step 5: each output unit (yk, k = 1, ..., m) add the weight of the input signal.

$$y\_ink = w0k + \Sigma zjwjkmj = 1 \qquad (5)$$

with, $w0k$ as the bias on the output unit k and apply the activation function to calculate the output signal, yk = f (y_ink)

Reverse propagation

Step 6: each output unit (yk, k = 1, ..., m) accepts an interconnected target pattern in inputting the training pattern, calculating the misinformation

ISBN 978-602-53524-0-9

$\delta k = (tk - yk)\, f'\ (yink)$ (6)

Calculate the weight correction (used to update wjk later),

$\Delta\ wjk = \alpha\ \delta k\ zj$ (7)

Calculate the bias correction (used to update wok), and send it to the units in the layer below it.

Step 7: each layer unit is hidden (zj, i = 1, ..., p) add the result of the change of input (from the layer units above).

$\delta\ \_inj = \Sigma \delta k\ wjknj = 1$ (8)

Multiply by the activity function derivative to calculate the error information,

$\delta j = \delta\_injf'\ \ (zinj)$ (9)

Calculate the correlation of the weights (used to update voj)

Step 8: each output unit (yk, k = 1, ..., m) updates the bias and weight (j = 0, ..., p):

wjk (new) = wjk (old) + $\Delta\ wjk$ (10)

Each layer unit is hidden (zj, i = 1, ..., p) updates its bias and weight (I = 0, ..., n):

vij (new) = vij (old) + $\Delta\ vij$ (11)

V. SIMULATION AND RESULT

a. Training and testing network

Training and testing uses 1 output layer, 2 hidden layers, and 1 output layer. The hidden layer consists of 10 neurons and 5 neurons. In this study the best training model was obtained at 0.1 learning rate and 0.7 momentum. In Figure 3 the results of the training are illustrated with a linear graph, so the training results are obtained as follows:

The best line gradient (m1)	=	0.609
Constant (a1)	=	76,415
Correlation Coefficient (r1)	=	0.790

In the training, the percentage of correlation was 79.0%. The training results include good, percentage of the correlation is in the range 0.600-0.799 which is included in the strong category [18].

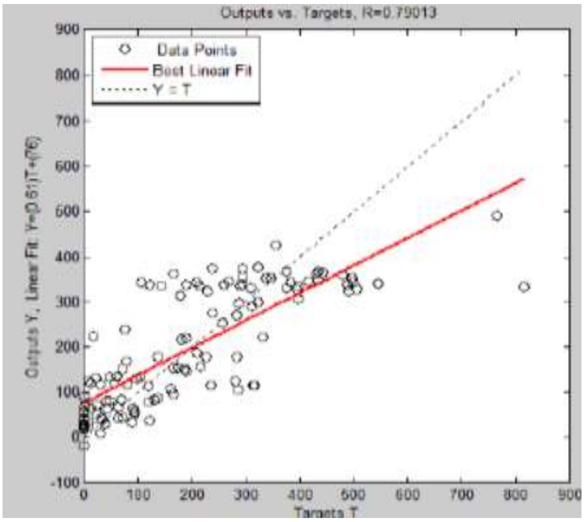

Figure 2. Training results are depicted in a linear graph

Figure 3 is a test result with a linear graph. In the testing process carried out as follows:

The best gradient (m2) = 0.615

Constant (b1) = 106,723

Correlation Coefficient (r2) = 0.775

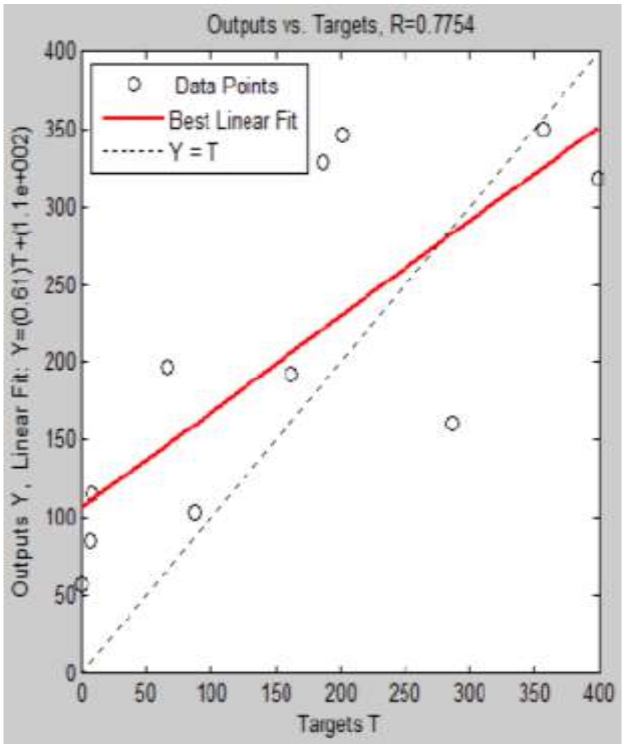

Figure 3. Training results are depicted in a linear graph

ISBN 978-602-53524-0-9

Comparison between the values of observation, particle swarm optimization (PSO), and ARIMA prediction results is done to find out that modeling with particle swarm optimization is used for prediction, which is presented in table 1. For PSO results obtained RMSE value of 101.846 and ARIMA prediction results with RMSE value of 145.948. The RMSE value for PSO is smaller than that of ARIMA, it can be said that the PSO prediction results are better used compared to the predictions with the ARIMA method.

Table 1. Comparison of rainfall values between observation data, PSO results, and ARIMA results in Semarang

Month	Observation	PSO	ARIMA
Jan-16	356	425	376
Feb-16	142	335	437
Mar-16	323	299	237
Apr-16	267	344	185
May-16	496	353	104
Jun-16	107	342	95
Jul-16	34	40	18
Aug-16	92	52	45
Sep-16	280	124	77
Oct-16	237	374	206
Nov-16	228	323	273
Dec-16	396	306	301
RMSE	-	101.846	145.946

b. Relationship between Sea Level and Rainfall

The results of the graph between rainfall and sea level data are illustrated in figure 4.

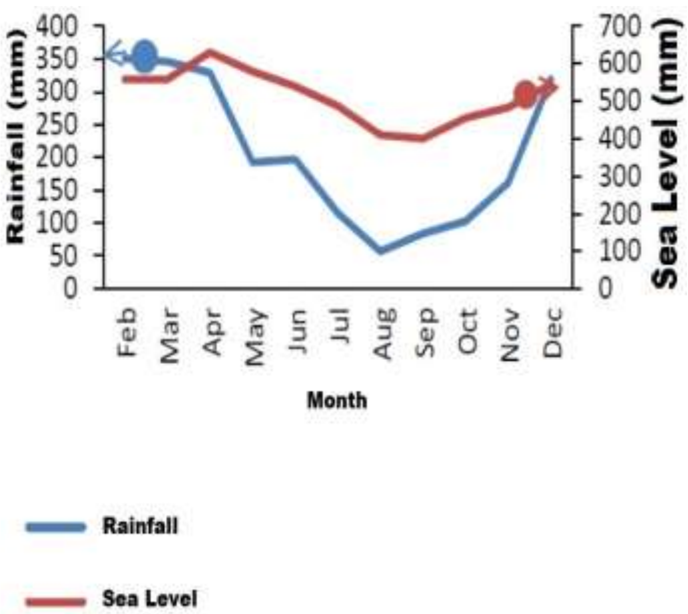

Figure 4. Graph between rainfall from PSO test results with sea level data

In Figure 5 shows the suitability of the data between the value of rainfall from the test results with sea level data. There are nine appropriate data and two data do not match in April and September.

For the appropriate data occurs due to the occurrence of global warming which resulted in rising sea surface temperatures which will affect the high rainfall that falls. Increased rainfall intensity will result in a rise in sea water so that sea level increases.

For data that is not suitable can be caused by global factors, namely the increase in global temperature which results in an increase in greenhouse gases. Air temperature will increase and cause polar ice to melt and increase the volume of sea water throughout the world.

This is corroborated by Wirasatriya's research, which explains that sea level rise in Semarang is not a dominant factor because it only causes an increase in sea level of 2.65 mm per year. Sea level rise in Semarang is due to global factors and local factors [19].

VI. CONCLUSIONS

Particle Swarm Optimization applications model can be used for prediction. Prediction with PSO model is better than ARIMA, where PSO model is obtained with RMSE value of 101,846. The best predictive model with PSO is obtained by using two hidden layer units consisting of 10 neurons and 5 neurons. Network parameters such as learning rate are 0.3 and momentum is 0.6. Network training resulted in a percentage of correlation of 79.0% and network testing resulted in a correlation percentage of 77.5%. Correlation between rainfall prediction model PSO with sea level data get a suitable graph pattern, where there is a relationship between high rainfall and sea level rise in Semarang.

REFERENCES

[1] Box, G.E.P. and Jenkins, G.M. Time Series Analysis, Forecasting and Control. Holden-Day, CA, San Francisco, 1970.

[2] Leite, S. , Mand, J.P. and Peixoto, S. The autoregressive model of climatological time series an application to the longest time series in Portugal. Intern. Journ. of Climatology, 16, 1165-1173, 1996.

[3] Chander, R.E. and Wheater, H.S. Analysis of rainfall variability using generalized linear models: A case study from the west of Ireland. Water Resour. Res., 38(10), 1– 11, 2002.

[4] Shamsnia, S.A. , Amiri, S.N. and Pirmoradian, N. Drought simulation in Fars Province using standardized precipitation index and time series analysis (ARIMA Model). Intern. Journ. Appl. Math. (IJAM), 22(6), 869-878, 2009.

[5] Parrelli, R. Introduction to ARCH & GARCH models. Optional TA Handouts, Econ 472 Department of Economics, University of Illinois, 2001.

[6] French, M. N., Krajewski, W. F., and Cuykendall, R. R. Rainfall forecasting in space and time using neural network, J. Hydrol., 137, 1–31, 1992.

[7] Abrahart, R. J. and See, L. Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecast in two contrasting catchments. Hydrol. Proc., 14, 2157–2172, 2000.

[8] Hung, N. Q., Babel, M.S., Weesakul, S. and Tripathi, N.K. An artificial neural network model for rainfall forecasting in Bangkok, Thailand. Hydrol. Earth Syst. Sci., 13, 1413–1425, 2009.

[9] Samui, P., Mandla, V. R. , Krishna A. and Teja T. Prediction of Rainfall Using Support Vector Machine and Relevance Vector Machine. Earth Sc. India, 4(IV), 188 – 200, 2011.

ISBN 978-602-53524-0-9

[10] Jha, G.K., Thulasiraman, P. and Thulasiram, R.K. PSO based neural network for time series forecasting. IEEE International Joint Conference on Neural Networks (IJCNN), June 14–19, USA, 1422–1427, 2009.
[11] Adhikari, R. and Agrawal, R.K. Effectiveness of PSO based neural network for seasonal time series forecasting. Indian International Conference on Artificial Intelligence (IICAI), Tumkur, India, 232–244, 2011.
[12] Asadi, S., Tavakoli, A., and Hejazi, S.R. A new hybrid for improvement of auto-regressive integrated moving average models applying particle swarm optimization. Expert Systems with Applications , 39, 5332–5337, 2012.
[13] Cui, C. and Jiang, M. Chaotic Time Series Prediction Based On Binary Particle Swarm Optimization. AASRI Procedia, 1, 377–38, 2012.
[14] J. Kennedy, R. C. Eberhart, Y. Shi, Swarm Intelligence, 2001, Pages 369-392
[15] Haggan, V. and Ozaki, T. Modeling nonlinear vibrations using an amplitude dependent autoregressive time series model. Biometrika, 68, 189- 196, 1981.
[16] Gurung, B. Fitting Nonlinear Time-series Model Using Swarm Optimization Technique. Advance in Electronic and Electric Engineering, 4(6), 537-540, 2014.
[17] Akaike, H. A new look at the statistical model identification. IEEE Trans. Autom. Control, AC-19,716-723, 1974
[18] Sugiyono, 2007, Statistika Untuk Penelitian, Alfabeta, Bandung
[19] Wirasatriya, Anindya, Agus Hartoko, Suripin, 2006, Kajian Kenaikan Muka Laut Sebagai Landasan Penanggulangan Rob di Pesisir Kota Semarang, Jurnal Pasir Laut, Vol. 1, No. 2, Januari 2006, 31-42