A Modular Technique for Monthly Rainfall Time Series Prediction

Jesada Kajornrit[1], Kok Wai Wong[2], Chun Che Fung[3]
School of Information Technology, Murdoch University
South Street, Murdoch, Western Australia, 6150
j_kajornrit@hotmail.com[1], k.wong@murdoch.edu.au[2], l.fung@murdoch.edu.au[3]

***Abstract*—Rainfall time series forecasting is a crucial task in water resource planning and management. Conventional time series prediction models and intelligent models have been applied to this task. Attempt to develop better models is an ongoing endeavor. Besides accuracy, the transparency and practicality of the model are the other important issues that need to be considered. To address these issues, this study proposes the use of a modular technique to a monthly rainfall time series prediction model. The proposed model consists of two main layers, namely, a prediction layer and an aggregation layer. In the prediction layer, Mamdani-type fuzzy inference system is used to capture the input-output relationship of the rainfall pattern. In the aggregation layer, Bayesian learning and nonlinear programming are used to capture the uncertainty in the time dimension. Eight monthly rainfall time series collected from the northeast region of Thailand are used to evaluate the proposed model. The experimental results showed that the proposed model could improve the prediction accuracy from the single model. Furthermore, human analysts can interpret such model as it contains set of fuzzy rules.**

***Keywords*—Modular Technique; Monthly Rainfall Data; Time Series Prediction; Fuzzy Inference System; Bayesian Learning Method; Northeast Region of Thailand**

I. Introduction

Rainfall time series forecasting is one of the most important issues in water resources planning and management. Forecasting of rainfall variable is used for flood and drought prevention, reservoir operation, contract negotiation, and irrigation scheduling [1]. Although, many researchers have dedicated much effort to improve the conceptual and empirical methods of rainfall forecasting, there is still a need to improve the operational forecasting systems.

In general, conventional models such as Autoregressive Moving Average (ARMA) [4], [6] and intelligent models like Artificial Neural Network (ANN) [4], [5], [6], [7] could provide accurate prediction. However, the prediction mechanisms of those models are in parametric form, which are difficult for human analysts to analyze [2], [3]. This could be viewed as a disadvantage of those models. Researchers realized that, not only improving the prediction accuracy is important, but also the issues of the transparency and practicality of the model need to be addressed.

In order to address the aforementioned issues, this study proposed the use of a modular technique to perform monthly rainfall prediction modeling. With the proposed modular technique, the complexity of the model decreases and thus the system could be easier to model.

The paper is organized as follows. Section 2 discusses some related works. Section 3 describes the case study area and datasets. The proposed model is presented in Sections 4. Section 5 shows the experimental results and analysis. Finally, Section 6 provides the conclusion.

II. Soft Computing Technique in Hydrological Time Series prediction

In hydrological studies, rainfall prediction is relatively difficult than other climate variables such as temperature. This is because of the highly stochastic nature in rainfall, which shows a lower degree of spatial and temporal variability. To address this issue, soft computing techniques have been adopted in the past decades.

Wu et al. [4] proposed the use of ANN with data preprocessing techniques to predict precipitation data in daily and monthly scale. They applied three preprocessing techniques, namely, Moving Average, Principle Component Analysis and Singular Spectrum Analysis to smoothen the time series data. Somvanshi et al. [5] confirmed in their work that ANN provided better accuracy than ARMA model for daily rainfall time series prediction.

Application of soft computing techniques to time series prediction does not limit only to predicting rainfall data but also allow the prediction of other hydrological variables such as streamflow modeling [6], [7] and rainfall-runoff modeling [8]. Adaptive Neuro-Fuzzy Inference System (ANFIS) is another popular technique that has been applied to hydrological time series [9], [10]. However, the disadvantages of ANFIS are the large number of parameters used and high computational cost.

The studies above show the successful cases of soft computing techniques used in the hydrological time series prediction problem. However, those models are difficult to understand due to the black-box nature of the prediction mechanism. Kajornrit et al. [2], [3] proposed the use of modular fuzzy inference system to make the prediction model more transparent and easily interpretable by human analysts. However, such models still lack of the capability to address the uncertainty in the time dimension. This study proposes a use of modular technique to perform monthly rainfall time series prediction. This study could be seen as an improvement of the work reported in [2].

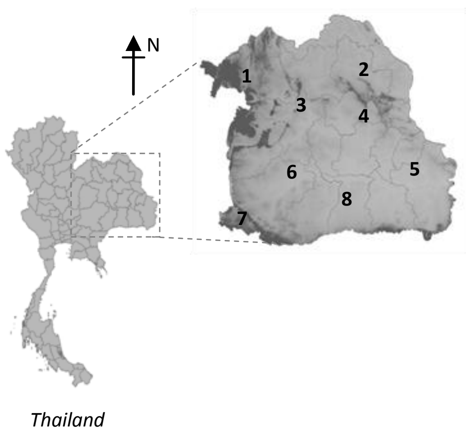


Fig 1. The case study area is located in the northeast region of Thailand. Eight monthly rainfall time series collected from eight rain gauge stations.

III. CASE STUDY AND RAINFALL TIME SERIES

The case study area used in this study is located in the northeast region of Thailand (see Fig 1). Eight monthly rainfall time series collected from the study area are used to evaluate the proposed model. The statistics of the eight datasets are shown in Table 1. The data from 1981 to 1998 were used to calibrate the models and data from 1999 to 2001 were used to validate the proposed models. This study used the models to predict one step-ahead, that is, one month. To validate the models, Mean Absolute Error (MAE) is adopted as given in equation (1).

$$MAE = \sum_{i=1}^{m} |Oi - Pi| / m \quad (1)$$

where O_i and P_i are the observed and the predicted value respectively, and m is the number of predicted data. The correlation coefficient of Fit (R) is also used.

TABLE I. DATASETS' STATISTICS

Statistics	Case 1	Case 2	Case 3	Case 4
Mean	929	1303	889	1286
SD	867	1382	922	1425
Kurtosis	-0.045	-0.100	0.808	0.532
Skewness	1.655	0.952	1.080	1.131
Minimum	0	0	0	0
Maximum	3527	5099	4704	6117
Latitude	17.25N	17.15N	16.66N	16.65N
Longitude	101.80E	104.13E	102.88E	104.05E
Altitude	283	176	164	155

Statistics	Case 5	Case 6	Case 7	Case 8
Mean	1319	981	1296	1124
SD	1346	976	1289	1153
Kurtosis	-0.224	1.229	1.590	1.725
Skewness	0.825	1.154	1.276	0.961
Minimum	0	0	0	0
Maximum	5519	4770	6558	6778
Latitude	15.50N	15.40N	14.63N	15.40N
Longitude	104.75E	102.35E	101.30E	103.40E
Altitude	129	152	476	152

IV. THE PROPOSED MODEL

Modular technique is a promising approach to decrease the complexity and computational cost of establishing prediction models. Through appropriate decomposition of the original problems, sub-modules could be individually modeled easier than the non-modular approach [11]. In the proposed model, outputs from individual sub-modules are subsequently aggregated to provide the final prediction values of the overall system.

Among several modular techniques, this study adopted the modular approach similar to the "*Multiple Expert Systems*" [11]. In this model, the input is fed into all relevant sub-modules and the final prediction value is aggregated from those sub-modules by a gating module. This architecture has been successfully applied in many areas [12], [13].

A. System architecture

The architecture of the proposed model is depicted in Fig 2. The model consists of an input layer, a prediction layer, an aggregation layer and an output layer. The input layer is used to feed the input data into the associated prediction modules. The prediction layer consists of twelve prediction modules (predictors) associated to the calendar months. The function of these modules is to generalize the input-output relationship of the rainfall pattern of the month. The aggregation layer consists of twelve aggregation modules (aggregator) associated to the calendar months, which are the same as the prediction modules. The function of these modules is to aggregate the outputs from the associated prediction modules by using the combination weights. The output layer is used to derive the final prediction of the system.

An example of the model operation is as follows - suppose the model is used to predict the rainfall value in February (see Fig 2). Firstly, the input selector feeds input data into the associated predictor (e.g. Feb), previous predictor (e.g. Jan) and next predictor (e.g. Mar). Secondly, the outputs from the predictors are aggregated by associated aggregator (e.g. Feb). Finally, the output selector receives the aggregated output from associated aggregator and provides the final output. Therefore, to perform the prediction, three consecutive predictors and one aggregator will be used. The principle concepts of this technique and operation are described as follow.

By taking the monthly rainfall time series into account, the data themselves are decomposed according to the calendar months. Therefore, twelve sub-modules are created to capture the input-output rainfall relationship in each month. When the single model is decomposed into smaller sub-modules, the complexity handling by a single model could be decreased. In this case, each sub-model handles only one month.

However, there is one drawback of using sub-modules. Since the sub-modules work independently, such model could loss the capability to capture the uncertainty in the time dimension. In order to address this problem, twelve aggregation modules are added. If the aggregation modules could capture the uncertainty in the time dimension efficiently, the prediction accuracy should increase.

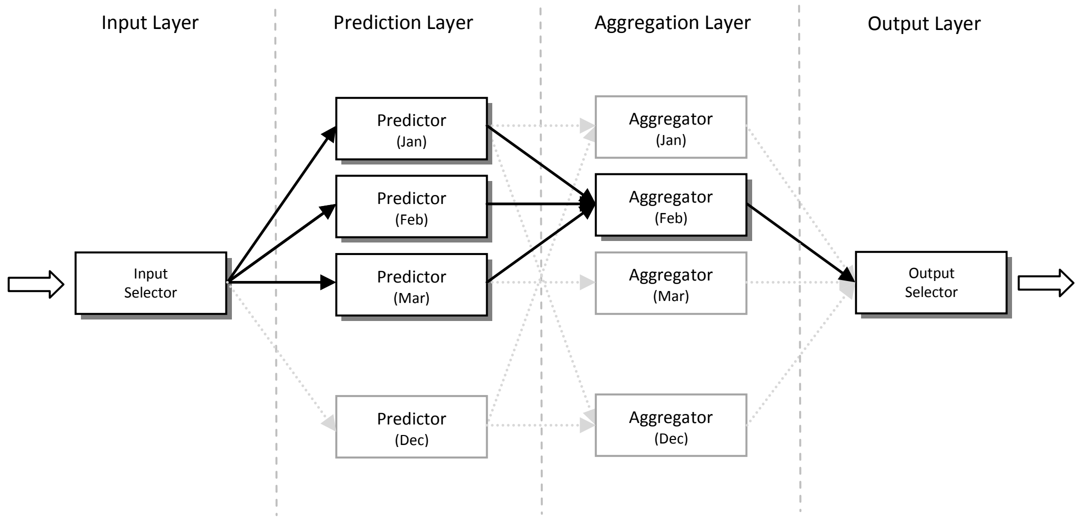


Fig 2. The architecture of the proposed model.

In the proposed model, the predicted values from three consecutive prediction modules are aggregated to provide the final value of the system. The hypothesis of this study is that the uncertainty in time will probably occur between the month immediately before and after the predicted month. Fig 3 shows an example of the overlapping of the two dimension input data for the three consecutive months (supposed that each prediction module has two dimensional input vectors). It can be seen that there are overlaps between two consecutive months. This overlapping area could be seen as the uncertainty in time dimension. The input pattern may locate closer to one cluster than the other to a certain degree. Therefore, aggregation is needed when the input data fall into overlapping area. Although there is overlapping of input data of further months, aggregation is not needed because it is separated naturally by time dimension.

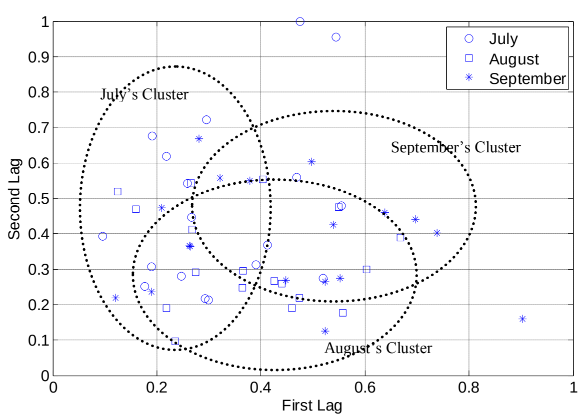


Fig 3. An example of two dimensional plot of input data of three consecutive months (July, August and September).

B. Input Identification

The objective on predicting rainfall using antecedent values is to generalize a relationship of the following form:

$$y = f(x^m) \tag{2}$$

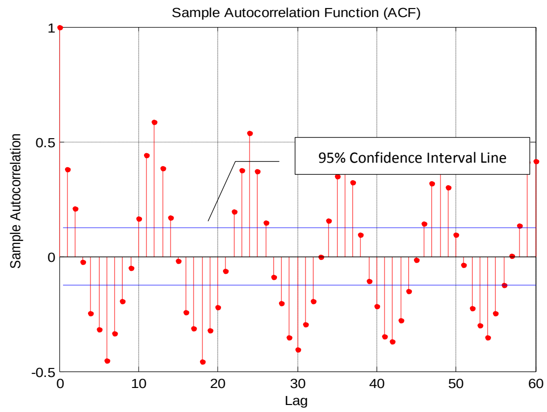
where x^m is a m-dimensional input vector representing rainfall value with different time lags. Generally, x^m is not known a priori. Furthermore, there are no fixed rules to define x^m for soft computing techniques [6].

In this study, two statistical methods (i.e. the autocorrelation function (ACF) and the partial autocorrelation function (PACF)) are employed to determine the dimension *m* of input vectors [6]. The ACF and PACF are generally used in diagnosing the order of the autoregressive process. Fig 4 shows an example of ACF and PACF of the dataset. ACF exhibits the peak value at Lag 12 and PACF showed a significant correlation at 95% confidence level interval up to lag 12. Therefore, twelve antecedent rainfall values have the most information to predict future rainfall.

However, for the proposed model, the whole system is decomposed into twelve sub-modules, 12-lag information may be redundant to the sub-module. This study proposed the use of first lag that cross confidence interval line as minimum information for each sub-module. Therefore, two antecedent rainfalls are considered as input for each sub-module. This 2-lag input has been proved in the work of [3] and [14] that it has sufficient information for modular model to provide accurate results.

C. Prediction Module

Among various types of prediction models, Mamdani-type Fuzzy Inference System (MFIS) is seemed to be the most appropriate choice to the model because it is intuitive and well



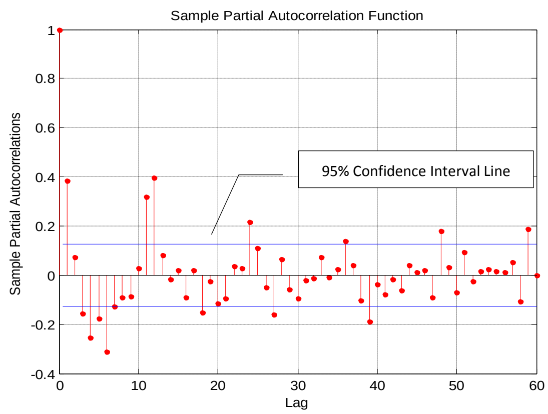


Fig 4. ACF and PACF of rainfall time series in case 1. (The ACF and PACF of other dataset are rather similar. They are not present here due to the limited space)

suited for human understanding. In some cases, although FIS is transparent to human analysts, it may not be appropriate if the input dimension is large, which will result in long antecedent part or too many fuzzy rules. In this model, when the information needed for sub-modules is a two dimensional input vectors. It is more appropriate to use the MFIS model. Therefore, in the proposed model, MFIS is adopted.

In order to generate prediction module (PM), fuzzy c-mean clustering method (FCM) is used to generate MFIS [22]. The rule extraction method uses the FCM to determine the number of rules and membership functions for the antecedents and consequents. For the inference properties, "min" function is used for implication, "max" function is used for aggregation, "centroid" function is used for defuzzification method, and Gaussian function is used for MFs.

The number of clusters in FCM method is determined by using subtractive clustering [20]. One parameter has to be defined in subtractive clustering is a vector that specifies a cluster center's range of influence in each of the data dimensions, assuming the data falls within a unit hyper box *("radii")*. To ensure that the range of the subtractive method examines at least half of the range of data in unit hyper box, this study then set the *radii* = 0.5.

D. Aggregation Module

In order to obtain the final output of the system, the aggregation module (AM) is used to combine the outputs y_i from $\{PM_i\}_{i=1}^{K}$ by using the combination weights. The combination formula is

$$y = \sum_{i=1}^{K} w_i y_i \tag{3}$$

where $w_i \geq 0$ and $\sum_{i=1}^{K} w_i = 1$. These weights could be viewed as the measure of "closeness" of rainfall pattern, in which the rainfall pattern is close to rainfall pattern of that PM. A lager combination weight indicated that the rainfall pattern is closer to that PM than others. This study examines two methods to evaluate the combination weights, that is, sequential method and non-sequential method. Bayesian learning is used for sequential method; whereas nonlinear programming is used in the non-sequential method.

E. Sequential Method

In [15], Wang et al. proposed the Bayesian learning method that aggregates information from modular neural networks in sequential way. This method is then adopted for the proposed model. Since the nature of time series data is sequential of time, the aggregation in the sequential fashion should be more appropriate than non sequential fashion. The steps to create combination weight from associated PM are as follows:

Step 1. Prepare all the sub-modules
Step 2. Ordering S calibration data records form oldest to newest
Step 3. For i = 1 to S:

Step 3a. Calculate Likelihood Function (*LF*) values, ω_j^i *(j = 1,2, ... ,K)* as

$$\omega_j = \frac{1/sse_j}{\sum_{k=1}^{K} 1/sse_k} \tag{4}$$

where sse_j is the training error of the j^{th} prediction module, K is the number of prediction modules aggregated (in this case $K = 3$).

Step 3b. Update the combination weights by using Bayesian reasoning as

$$W_j^i = \begin{cases} w_j^i = w_j^{i-1} & if \quad \sum_{j=1}^{K} w_j^{i-1}\omega_j^i = 0 \\ \frac{w_j^{i-1}\omega_j^i}{\sum_{j=1}^{K} w_j^{i-1}\omega_j^i} & otherwise \end{cases} \tag{5}$$

From step 3b, it can be seen that the calibration data are constructed in a sequential way so that each calibration data processes certain property of inheritance. The advantage of Bayesian decision analysis is that it can model uncertainty information via Bayesian reasoning process [15], which can help the analyst gain more insights into the system to be modeled.

F. Non-sequential Method

For non-sequential method, *constrained nonlinear optimization* (*constrained nonlinear programming*) [16] is used to find the optimal combination weights. The algorithm attempts to find a constrained minimum of a scalar function of several variables starting from an initial estimate. The algorithm uses a *Hessian*, the second derivatives of the Lagrangian [18]. The problem can be specified by

$$\min_x f(x) \; such\ that \left[\begin{array}{l} A.x \le b \\ Aeq.x \le beq \end{array} \right. \tag{6}$$

where $A.x \le b$ is set for constrain $w_i \ge 0$ and $Aeq.x \le beq$ is set for constrain $\sum_{i=1}^{K} w_i = 1$. For this case, A = [*-1 0 0; 0 -1 0; 0 0 -1*]; b = [*0; 0; 0*]; Aeq = [*1 1 1*]; beq = [*1*]. The initial estimate vector is set to $[0\ 1\ 0]^T$. In other word, the algorithm finds the optimal values of w_i that are better than no aggregation method. The cost function *f(x)*, which has to be minimized, is as follows.

$$sse = \sum_{i=1}^{S} [(w_1 z'_{1i} + w_2 z'_{2i} + w_3 z'_{3i}) - z_i] \tag{7}$$

where *sse* is error of calibration data, *S* is the number of calibration data, z'_i is the predicted value from PM_i and z_i is the observed value.

V. Experimental Results and Analysis

To evaluate the prediction accuracy, the proposed model was compared to hydrological common-used prediction models, namely, Autoregressive Moving Average (ARMA), Artificial Neural Network (ANN) [2],[3],[4],[5] as well as Fuzzy Inference System (FIS) [19],[20]. Furthermore, the proposed model is also compared to the model without aggregation method [2]. From now on, the "*Mod* FIS" stands for the proposed model without aggregation method, "*Mod* FIS–*BSM*" stands for the Bayesian method and "*Mod* FIS–*HSA*" for the Hessian method.

A. Model calibration

In order to select the optimal ARMA model, Akaike Information Criterion (AIC) is adopted [2], [4]. This study generated ARMA models from calibration data by replacing parameters *p* and *q* of ARMA model from 0 to 12. The parameters that gave lowest AIC value are used for ARMA model. Table 2 shows the ARMA models for eight datasets.

TABLE II. The selected parameters and AIC values

Case	(p,q)	AIC	Case	(p,q)	AIC
1	(4,4)	13.417	5	(5,3)	13.751
2	(10,9)	13.982	6	(12,1)	13.536
3	(6,3)	13.379	7	(12,0)	14.334
4	(8,11)	14.182	8	(11,2)	13.850

For ANN and FIS, there is no consistent theory to select the number of input. However, the work of [4], [14] recommended the use of ACF and PACF to investigate the appropriate inputs. Considering ACF and PACF of time series data (Fig. 4), it pointed out that time series show autoregressive process up to lag 12. Therefore, 12-lag inputs seem to be sufficient information for the model [4].

The ANN used in this study is an one hidden layer Back-Propagation Neural Network (BPNN). The architecture of BPNN is twelve input nodes and one output node. The optimal number of hidden node is selected by trial and error procedure. To investigate the optimal number of hidden nodes, calibration data are separated into two parts. The first part is use to train BPNN and the second part is used to test BPNN. The experiments varied the number of hidden nodes from 2 to 6 and the experiments are repeated 100 times to ensure the results. An example of the results is shown in Fig 5.

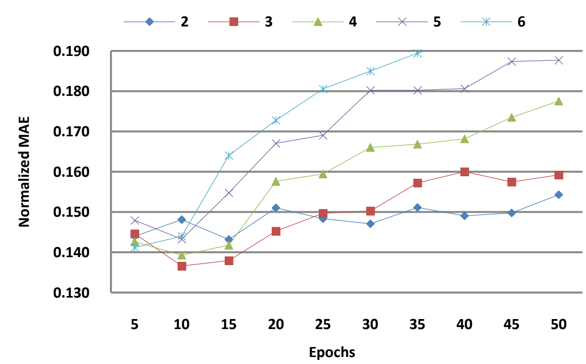


Fig 5. An example of MAE of testing data in BPNN process

From the experiment, the number of two or three hidden nodes could provide the minimum error. Table 3 summarizes the architecture of BPNN for eight datasets (p, q and r are referred to number of input, hidden, output nodes). Furthermore when the number of training epoch is larger than 15, the error from testing data start to increase. Therefore, the number of epoch was limited to 15. Since the accuracy of BPNN depends on the initial random weights, to create the BPNN model for comparison, this study generated 100 BPNN models and selected the model that provided MAE closest to the average MAE of those 100 models.

TABLE III. The architecture and epoch of BPNNs

Case	(p,q,r)	Case	(p,q,r)
1	(12,3,1)	5	(12,2,1)
2	(12,2,1)	6	(12,3,1)
3	(12,3,1)	7	(12,2,1)
4	(12,3,1)	8	(12,3,1)

The popular FIS model used in hydrological studies is Sugeno-type FIS model created by subtractive clustering techniques [20]. One parameter has to be defined in creating the FIS is a vector that specifies a cluster center's range of influence in each of the data dimensions (or "*radii")*, assuming the data falls within a unit hyper box. The optimal radii value was investigated in the same procedure as BPNN. Fig 6 shows the testing error by increasing radii form 0 to 1. The optimal radii of eight datasets are shown in Table 4.

TABLE IV. THE SELECTED RADII VALUES OF FIS MODELS

Case	Radii	Case	Radii
1	0.5	5	0.3
2	0.4	6	0.4
3	0.4	7	0.4
4	0.3	8	0.3

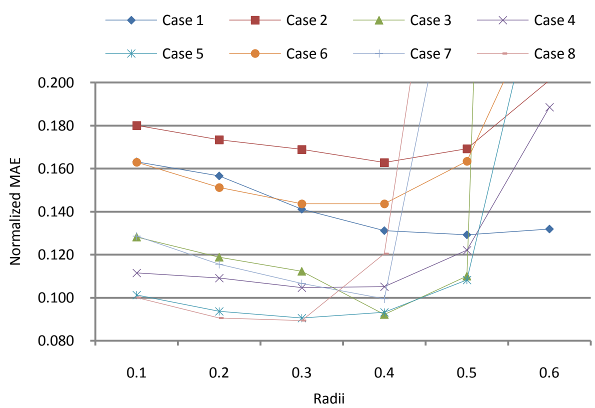


Fig 6. MAE of testing data in FIS trial and error process

B. Model evaluation

The MAE and R measure of validation period are shown in Table 5 and Table 6. To evaluate the overall results, the last column in Table 5 shows the average of normalized MAE by their mean rainfall value or Relative Mean Absolute Error (RMAE) and the last column in Table 6 is the average of R values or Relative Correlation Coefficient (RR). Since the results from RMAE and RR measures are consolidated, these experimental results are rather consistent.

Among the single models, the order of prediction accuracy is FIS > ARMA > ANN in general. In term of MAE, both FIS and ARMA can provide the best prediction in 4 cases. However, in terms of R, FIS provided the best in 5 cases; whereas ARMA can provide the best prediction accuracy only in 2 cases and compatible to FIS in 1 case. In this experiment, it seems that ANN provided the lowest prediction accuracy because it provided the highest MAE in 5 cases and lowest R in 6 cases. Based on the ANN model, the accuracy of FIS and ARMA is improved by 6.2% and 3.4% respectively.

Considering *Mod* FIS model, such model could improve the prediction accuracy from single model significantly. In term of MAE and R, *Mod* FIS could provide better prediction than the best of single model in all cases. Overall, the accuracy of *Mod* FIS is improved from ANN 23%, from ARMA 20% and from FIS 18%.

Among the modular models, the order of prediction accuracy is *Mod* FIS-*BSA* > *Mod* FIS-*HSA* > *Mod* FIS in general. In term of MAE, *Mod* FIS-*BSA* provided the best prediction results in 5 cases and provided compatible accuracy to *Mod* FIS-*HSA* in 1 case. In term of R, *Mod* FIS-*BSA* showed the best accuracy in 5 cases and showed compatible accuracy to *Mod* FIS-*HSA* in 2 cases. In this experiment, *Mod* FIS provided the lowest prediction accuracy in 6 cases in term of MAE and 5 cases in term of R. Based on *Mod* FIS model, the accuracy of *Mod* FIS-*HSA* and *Mod* FIS-*BSA* is improved 4.5% and 6% respectively.

Overall, the prediction accuracy ordered by *Mod* FIS-*BSA* > *Mod* FIS-*HSA* > *Mod* FIS > FIS > ARMA > ANN. It can be seen that, major improvement comes from the use of modular technique (*Mod* FIS) and the minor improvement comes from the use of aggregation layer. In turn, the sequential method provided more improvement than non-sequential method.

C. Analysis Discussion

In the single model, ANN is not appropriate to the problem since it provided the lowest prediction accuracy. The weak point of ANN for this problem is that ANN needs a lot of training data. In this case study, the calibration data is considered small. This could be the reason why ANN did not perform well for this case. In the work of [3], they showed that increasing the number of training set could improve the accuracy of ANN. Another possible reason is that the time series used in this study is periodic. Wu et al. [2] has also provided similar observation in their study, in which ANN performed well on daily rainfall data but not in the monthly rainfall data

In this study, one may conclude that FIS is more suitable than ARMA for two reasons. First, FIS provided better prediction accuracy than ARMA. Second, FIS model is more understandable to human analyst than ARMA. As the fuzzy rules are closer to human reasoning, an analyst could understand how the model performs the estimation. However, in this case, the second reason could not be claimed as the advantage of FIS.

Considering the MFs in Fig 7 (top), this figure shows an example of the MFs in the first input of one FIS model. In this FIS model, the number of inputs is twelve and the number of fuzzy rules generated is 195. This large number of model's parameters may not be appropriate for human analysts to analyze. Although the FIS model could be created by other algorithms (i.e. Fuzzy C-mean clustering) and could reduce the number of MFs and rules, however, twelve-dimension input is still difficult to understand. To address this problem, *the curse of dimensionality* [21], hierarchical FIS or modular FIS are the promising technique. This study selected the modular FIS to address the problem.

Considering the *Mod* FIS models, the prediction accuracy is improved significantly from single models. As the single model is divided into sub-modules, the heterogeneity of the input data has decreased. Consequently, each sub-model handles only a part of the data. The advantage of *Mod* FIS is that it takes only two lags as input information for each PM. This

TABLE V. MEAN ABSOLUTE ERROR (MAE) OF VALIDATION PERIOD

Model	TS353001	TS356010	TS381010	TS388002	TS407005	TS431008	TS431020	TS432004	RMAE
ARMA	688	626	529	707	823	560	671	471	0.562
ANN	526	631	551	793	806	648	736	592	0.581
FIS	443	619	630	769	781	529	699	487	0.545
Mod FIS	430	501	406	609	601	486	636	404	0.447
Mod FIS - *HSA*	458	457	364	602	602	456	549	392	0.427
Mod FIS - *BSA*	417	432	354	593	581	456	597	398	0.420

TABLE VI. CORRELATION COEFFICIENT (R) OF VALIDATION PERIOD

Model	TS353001	TS356010	TS381010	TS388002	TS407005	TS431008	TS431020	TS432004	RR
ARMA	0.539	0.787	0.543	0.797	0.666	0.587	0.466	0.776	0.645
ANN	0.731	0.761	0.572	0.740	0.656	0.465	0.371	0.664	0.620
FIS	0.764	0.787	0.605	0.747	0.681	0.600	0.496	0.755	0.679
Mod FIS	0.813	0.872	0.696	0.873	0.791	0.681	0.663	0.824	0.777
Mod FIS - *HSA*	0.816	0.859	0.721	0.877	0.791	0.693	0.683	0.830	0.784
Mod FIS - *BSA*	0.825	0.886	0.725	0.877	0.809	0.692	0.657	0.835	0.788

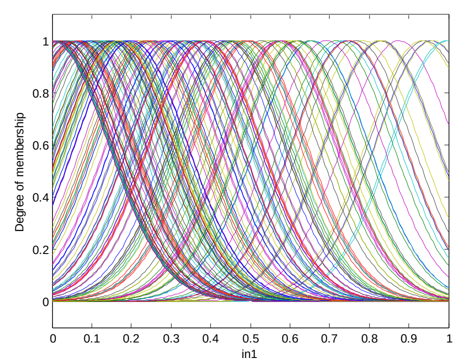
also allows the dimension of input data to decrease to a level that human analysts can understand. Fig 8 shows an example of the input-output relationship in 3 dimensional spaces. Another advantage of *Mod* FIS is the number of fuzzy rules and MFs significantly decrease, consequently it is more practical. Fig 7 (bottom) shows an example of MF in sub-module and the number of fuzzy rules related to the example is only 10.

However, even though the modular model in *Mod* FIS could address the accuracy and practicality problem. One weak point of *Mod* FIS is that each sub-module works independently. This makes the model weak in handling the uncertainty in the time dimension. To address this problem, aggregation layer is then added. In *Mod* FIS-*HSA*, weights in aggregation unit are generated from non-sequential method. The overall accuracy has been improved from *Mod* FIS. This points out that the uncertainty in time could be captured by using the aggregation layer.

The week point of using non-linear optimization technique to create combination weights is that the algorithm treats all the calibration data with the same significance. However, in time series data, the sequence of data does matter. Therefore, sequential method was more appropriate. In the Bayesian reasoning technique, the weights are evaluated sequentially form likelihood function. So this method treats the calibration data in ordered significance. Consequently, combining weights evolved across the time could model the dynamic of time series data better than *Mod* FIS-*HSA*. Therefore, accuracy has been improved. Fig 9 shows an example of the combination of weights in one aggregation module that are inherited across the time dimension.

VI. Conclusion

Accurate rainfall forecasting is crucial for reservoir operation, flood prevention and contract negotiation because it can provide an extension of lead-time for the flow forecasting. This study proposed the use of modular techniques to crate the prediction model for monthly rainfall time series data. The proposed model consists of two main layers, that is, prediction layer and aggregation layer. Mamdani-type fuzzy inference system was used in the prediction layer whereas the Bayesian



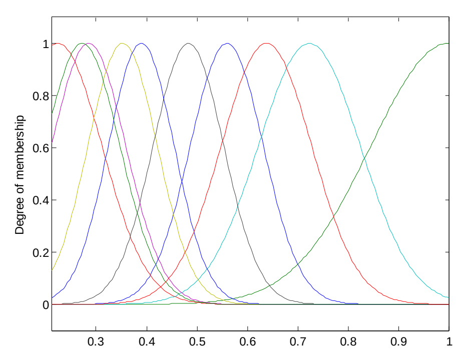


Fig 7. (Top) An example of MFs of the first input of the FIS model crated by subtractive algorithm. (Bottom) An example of MFs of the first input of one module in Mod FIS.

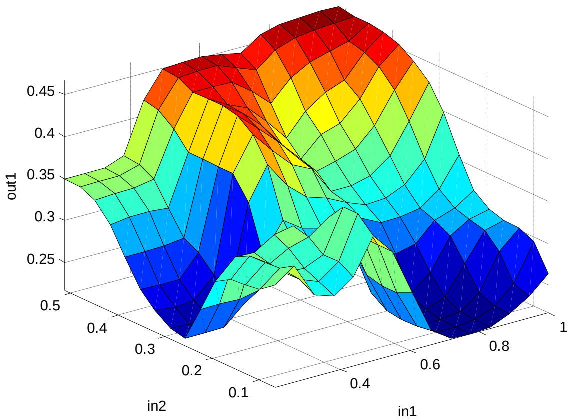


Fig 8. An example of input-output relationship of rainfall time series of one prediction module

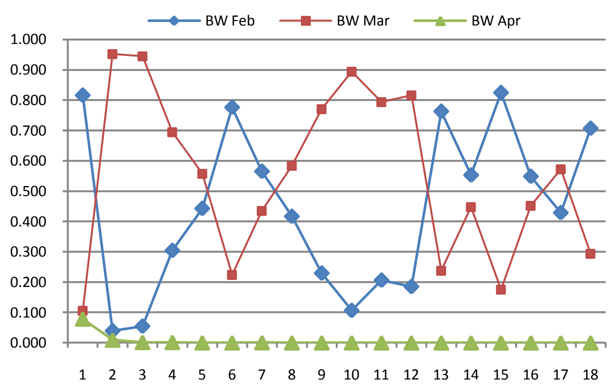


Fig 9. An example of the combination weights for March's AM inherit across time dimension

learning method and the nonlinear programming method were used in the aggregation layer. The experimental results pointed out that the use of modular model could improve the prediction of the single models in term of quantitative and qualitative. In the modular model, the prediction accuracy has significantly increased. The model established is more practically interpretable by human analysts. In the modular models, in turn, the aggregation module that uses Bayesian technique to create weights in the sequential fashion provided more accurate prediction results than those using non-sequential method and no aggregation method.

References

[1] S. Araghinejad, M. Azmi, M. Kholghi, "Application of artificial neural network ensembles in probabilistic hydrological forecasting." Journal of Hydrology, vol. 407, pp. 94-104, 2011.

[2] J. Kajornrit, K. W. Wong, C. C. Fung, "Rainfall Prediction in the Northeast Region of Thailand using Modular Fuzzy Inference System", In Proc. IEEE World Congress on Computational Intelligence, 2012.

[3] J. Kajornrit, K. W. Wong, C. C. Fung, "Rainfall Prediction in the Northeast Region of Thailand using Cooperative Neuro-Fuzzy Technique", In Proc. The 8th International Conference on Computing and Information Technology, 2012.

[4] C. L. Wu, K. W. Chau, C. Fan, "Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques." Journal of Hydrology, vol. 389, pp.146-167, 2010.

[5] V. K. Somvanshi, et al., "Modeling and prediction of rainfall using artificial neural network and ARIMA techniques." J. Ind. Geophys. Union, vol. 10, no. 2, pp. 141-151, 2006.

[6] W. Wang, K. Chau, C. Cheng, L. Qiu, "A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series." Journal of Hydrology, vol. 374, pp. 294-306, 2009.

[7] A. Jain, A. M. Kumar, "Hybrid neural network models for hydrologic time series forecasting." Applied Soft Computing, vol. 7, pp. 585-592, 2007.

[8] A. K. Lohani, N. K. Goel, K. K. S. Bhatia, "Comparative study of neural network, fuzzy logic and linear transfer function techniques in daily rainfall-runoff modeling under different input domains." Hydrological Process, vol. 25, pp. 175-193, 2011.

[9] P. C. Nayak, et al., "A neuro-fuzzy computing technique for modeling hydrological time series." Journal of Hydrology, vol. 291. pp. 52-66, 2004.

[10] M. Z. Kermani, M. Teshnehlab, "Using adaptive neuro-fuzzy inference system for hydrological time series prediction." Applied Soft Computing, vol. 8, pp. 928-936, 2008.

[11] H. ChrisTseng, B. Almogahed, "Modular neural networks with applications to pattern profiling problems." Neurocomputing, vol. 72. pp. 2093–2100, 2009.

[12] K. W. Wong, P. M. Wong, T. D. Gedeon, C. C. Fung, "Rainfall Prediction Model Using Soft Computing Technique." Soft Computing, vol 7, issue 6, pp. 434-438, 2003.

[13] C. C. Fung, K. W. Wong, H. Eren, R. Charlebois, H. Crocker, "Modular Artificial Neural Network for Prediction of Petrophysical Properties from Well Log Data," In Proc. IEEE Instrumentation & Measurement Technology Conference, 1996.

[14] H. Raman, N. Sunilkumar, "Multivariate modeling of water resources time series using artificial neural network." Hydrological Sciences –Journal- des Sciences Hydroligiques, vol. 40, pp.145-163, 1995.

[15] P. Wang et al., "A novel Bayesian learning method for information aggregation in modular neural networks." Expert Systems with Applications, vol. 37, pp. 1071-1074, 2010.

[16] Optimization toolbox™ – User's guide R2008b, Matlab.

[17] Fuzzy Logic toolbox™ – User's guide R2008b, Matlab.

[18] Byrd, R.H., J. C. Gilbert, and J. Nocedal, "A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming," Mathematical Programming, Vol 89, No. 1, pp. 149–185, 2000.

[19] A. K. Lohani, N. K. Goel, K. K. S. Bhatia, "Comparative study of neural network, fuzzy logic and linear transfer function techniques in daily rainfall-runoffmodelling under different input domains." Hydrological Processes, vol. 25, pp. 175-193, 2011.

[20] P. C. Nayak, K. P. Sudheer, "Fuzzy model identification based on cluster estimation for reservoir inflow forecasting." Hydrological Processes, vol. 22, pp. 827-841, 2008.

[21] S. Marsland, "Machine Learning An Algorithmic Perspective" CRC Press, 2009.

[22] M. Sugeno, T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling." IEEE Transaction on Fuzzy System, vol.1, issue.1, pp. 7-31, 1993.