Modern Applied Science; Vol. 9, No. 9; 2015
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

A Model Selection Procedure for Stream Re-Aeration Coefficient Modelling

David O. Omole[1,2], Julius M. Ndambuki[2], Adebola G. Musa[3], Ezechiel O. Longe[4], Adekunle A. Badejo[2] & Williams K. Kupolati[2]

[1] Department of Civil Engineering, Covenant University, Ota, Nigeria

[2] Department of Civil Engineering, Tshwane University of Technology, Pretoria, South Africa

[3] Department of Computer Science and Informatics, University of the Free State, Phuthaditjhaba 9866, South Africa

[4] Department of Civil & Environmental Engineering, University of Lagos, Lagos State, Nigeria

Correspondence: David O. Omole, Department of Civil Engineering, Covenant University, P.M.B. 1023, Ota, Nigeria. Tel: 234-804-400-6525. E-mail: david.omole@covenantuniversity.edu.ng

Received: February 16, 2015 Accepted: February 27, 2015 Online Published: August 30, 2015

doi:10.5539/mas.v9n9p138 URL: http://dx.doi.org/10.5539/mas.v9n9p138

Abstract

Model selection is finding wide applications in a lot of modelling and environmental problems. However, applications of model selection to re-aeration coefficient studies are still limited. The current study explores the use of model selection in re-aeration coefficient studies by combining several suggestions from numerous authors on the interpretation of data regarding re-aeration coefficient modelling. The model selection procedure applied in this research made use of Akaike information criteria, measures of agreement such as percent bias (PBIAS), Nash-Sutcliffe Efficiency (NSE) and root mean square error (RMSE) observation Standard deviation Ratio (RSR) and gragh analysis in selecting the best performing model. An algorithm prescribing a generic model selection procedure was also provided. Out of ten candidates models used in this study, the O'Connor and Dobbins (1958) model emerged as the top performing model in its application to data collected from River Atuwara in Nigeria. The suggested process could save software and model developers lots of time and resources, which would otherwise be spent in investigating and developing new models. The procedure is also ideal in selecting a model in situations where there is no overwhelming support for any particular model by observed data.

Keywords: model selection, information criteria, measures of agreement, re-aeration coefficient, stream, modelling

1. Introduction

Reaeration coefficient (k_2) modelling, as a relatively new and specialized field of study, has evolved over a period of ninety years through contributions by researchers from different parts of the world (Palumbo & Brown, 2013; Omole, 2012; Gayawan *et al*., 2009; Ye *at al*., 2008; Longe & Omole, 2008). This has resulted in the development of hundreds of k_2 models, often through processes that cost large sums of money, labour and time (Wang *et al*., 2013). Model developers agree that it is possible to save lots of resources by comparing existing models and selecting the most representative from a pool of carefully compiled models (Palumbo & Brown, 2013; Wang et al., 2013; Omole *et al*., 2013; Ritter & Munoz-Carpena, 2013). Indeed, some developed countries have provided guidance relating to the simulation and assessment of water quality in their respective environments by specifying certain models that have been found useful, thus setting the pace for developing countries to follow suit (Wang *et al*., 2013). In furtherance of this, hydrologic modellers have arrived at a consensus on the following modelling issues:

i. That it is necessary to standardize model evaluation procedures (Ritter & Munoz-Carpena, 2013; Moriasi et al., 2007).

ii. That the use of coefficient of determination (R2) and common error statistics such as standard error (SE) and normalized mean error (NME) are not sufficient for evaluating the performance of k_2 models (Palumbo & Brown, 2013; Ritter & Munoz-Carpena, 2013; Moog & Jirka, 1998).

iii. That in the process of evaluating models prior to selection, both graphical and error statistics should be considered (Harmel, et al., 2014). It is also popularly accepted that statistical evaluation of models must include both absolute error and dimensionless error indices in the analysis of goodness of fit (Omole et al., 2013; Moriasi et al., 2007; Harmel, et al., 2014; LeGates and McCabe, 1999).

iv. Finally, several literature agree that the Root mean square error (RMSE), percent bias (PBIAS) and RMSE observation Standard deviation Ratio (RSR) are good examples of absolute error statistic while Nash-Sutclife Efficiency (NSE) is acclaimed as the most widely used dimensionless error statistics (Ritter & Munoz-Carpena, 2013; Omole et al., 2013; Moriasi et al., 2007; Gupta & Kling, 2011; Ewen, 2011; Singh et al., 2005).

Hydrologic model developers, however, are yet to reach a consensus on the exact procedure to be adopted in the process of model selection. Also, there is no unanimity in the interpretation of some of the results from their analyses. In their article, Omole *et al.*, (2013) proposed the use of corrected Akaike Information Criteria (AIC*c*) in comparing the capacity of the models to interpret data from River Atuwara. The current study, however, takes a step further by quantitatively integrating graphic analysis into the procedure for model selection.

2. Methods

2.1 Theoretical Framework

The starting point in the model selection process is the short-list of candidate models. This should be carefully done to avoid wasted efforts. Basis of selection should be objective and based on researcher experience and scientific markers. This is because AIC would only select the most representative model out of the candidate models. This does not necessarily make the most representative model (among the candidate models) the best model for the data (Johnson & Omland, 2004). Information criteria should, in itself, be sufficient to select the best model. However when a single model does not provide overwhelming evidence of representation for real data, it becomes necessary to conduct further statistical and graphic analysis as proposed by Johnson & Omland, (2004). Overwhelming support for data being defined as $w_i > 0.9$ (Johnson & Omland, 2004), where w_i is the information criteria (IC) weight of model *i* obtained from a given set of candidate models. In the current study, both AIC*c* and BIC were used for comparison purposes even though AIC*c* would have been sufficient since all the models have the same parameters namely velocity and hydraulic radius. If some of the models included other known k_2 parameters such as slope, temperature, Froude number, time and/or discharge, then BIC would be more appropriate because it penalizes model complexity (parsimony) more than AIC. Both AICc and BIC are respectively defined by equation 1 and 2 (Omole *et al*., 2013; Burnham & Anderson, 2004; Johnson & Omland, 2004).

$$AIC_c = -2\ln\left[L\left(\hat{\theta}\middle|y\right)\right] + 2p\left(\frac{n}{n-p-1}\right) \tag{1}$$

and

$$BIC = -2\ln\left[L\left(\hat{\theta}\middle|y\right)\right] + p.\ln\left(n\right) \tag{2}$$

where n = sample size, p = count of free parameters; y = data; $L\left(\hat{\theta}\middle|y\right)$ = likelihood of model parameters.

Following the IC analysis, statistical analysis using measures of agreement was done. Ordinarily, based on the recommendation of Royall (1997), only the candidate model with the highest w_i, i.e. $\left(w_i^{\max}\right)$ and other candidate models having $w_i \geq 10\%$ of the value of $\left(w_i^{\max}\right)$ should be considered for further statistical tests. In this study, however, all the models were considered for both measures of agreement and graphic analysis since there was no model that had a distinct performance at any of the stages of analysis.

The measures of agreement used for this study are Percent BIAS (PBIAS), NSE and RSR. They are defined as:

$$Percent\ BIAS = \left[\frac{\sum_{i=1}^{n}\left(y_i^o - y_i^s\right)\times 100}{\sum_{i=1}^{n}\left(y_i^o\right)}\right] \tag{3}$$

$$NSE = 1 - \left[\frac{\sum_{i=1}^{n} \left(y_i^o - y_i^s \right)^2}{\sum_{i=1}^{n} \left(y_i^o - \bar{y} \right)^2} \right] \tag{4}$$

$$RSR = \frac{RMSE}{\sigma^2} \tag{5}$$

where y_i^o = observed data, y_i^s = simulated data, $\bar{y}$ is mean value of observed data and σ^2 = standard deviation.

Next is the graphic analysis. Each model was plotted as simulated data against observed data and the most visually representative model was allocated the highest weight of 10 (out of 10 candidate models), while the least representative model received the least weight allocation of 1. The allocation of the highest weight of 10 for the best performing model was also done at each stage of IC and measure of agreement analysis. At the end of all the analytical process (as detailed in the appendix), the average of all the weights were found for each model. The model with the highest score (in percent) emerged as the most representative model out of the ten candidate models.

Data used for analysis in this study was obtained during the rainy season (high stream velocity, depth and dilution) in July 2009 while data for the dry season (dry weather flow) was obtained in January 2010.

For the purpose of this study, the candidate models and the justification for their short-listing are presented in Table 1.

Table 1. Candidate models

s/n	Model	Authors	Symbol	Background
1	$k_2 = 46.2679 \frac{U^{1.5463}}{H^{0.0128}}$	(Omole & Longe, 2012; Omole, 2011)	OL	Developed from data obtained from River Atuwara, South-west Nigeria.
2	$k_2 = 12.9 \frac{U^{0.5}}{H^{1.5}}$	(Bowie *et al*., 1985; O'Connor & Dobbins, 1958)	OD	Developed for moderately deep to deep channels.
3	$k_2 = 11.632 \frac{U^{1.0954}}{H^{0.0016}}$	(Agunwanmba *et al.,* 2007)	AG	Developed from data obtained from creeks in the south-south part of Nigeria.
4	$k_2 = 5.792 \frac{U^{0.5}}{H^{0.25}}$	(Jha *et al*., 2001)	JH	Developed from data obtained from River Kali in India.
5.	$k_2 = 5.026 \frac{U^{0.969}}{H^{1.673}}$	(Bowie *et al*., 1985, Streeter *et al*., 1936)	SP	Developed from data gathered from River Ohio
6	$k_2 = 10.046 \frac{U^{2.696}}{H^{3.902}}$	(Baecheler & Lazo, 1999)	BL	Developed for rivers having slight slope in mountainous regions.
7	$k_2 = 21.7 \frac{U^{0.67}}{H^{1.5}}$	(Bowie *et al*., 1985; Owens *et al*, 1964)	OW	Developed from data taken from 6 different streams in England.
8	$k_2 = 4.67 \frac{U^{0.6}}{H^{1.4}}$	(Bowie *et al*., 1985; Bansal., 1973)	BS	Based on re-analysis of re-aeration data from numerous streams
9	$k_2 = 20.2 \frac{U^{0.607}}{H^{1.689}}$	(Bowie *et al*., 1985; Bennet & Rathbun, 1972)	BR	Developed from re-analysis of secondary data
10	$k_2 = 7.6 \frac{U}{H^{1.33}}$	(Bowie *et al*., 1985; Langbein & Dururn, 1972)	LD	Developed from the synthesis of data obtained from O'Connor and Dobbins (Bowie *et al.,* 1985, Churchill *et al*., (1962); Krenkel and Orlob (1962), Streeter *et al*., (1936).

3. Results

3.1 Information Criteria (IC) Analyses

Results of the AIC*c* and BIC analyses performed on the models listed in Table 1 are presented in Figures 1 – 2. The model having the lowest IC value is the most preferred model. The models are therefore ranked in order of IC value with the least IC value having the highest weight. Both AIC*c* and BIC were in agreement regarding the order of weights of the candidate models for each data set. Agunwamba *et al.*, (2007) model had the highest weight allocation for the dry season data while Bansal (Bowie et al., 1985) model emerged as the most preferred model for the rainy season. The ranking of the other models for either season are displayed in Figures 1 and 2 respectively.

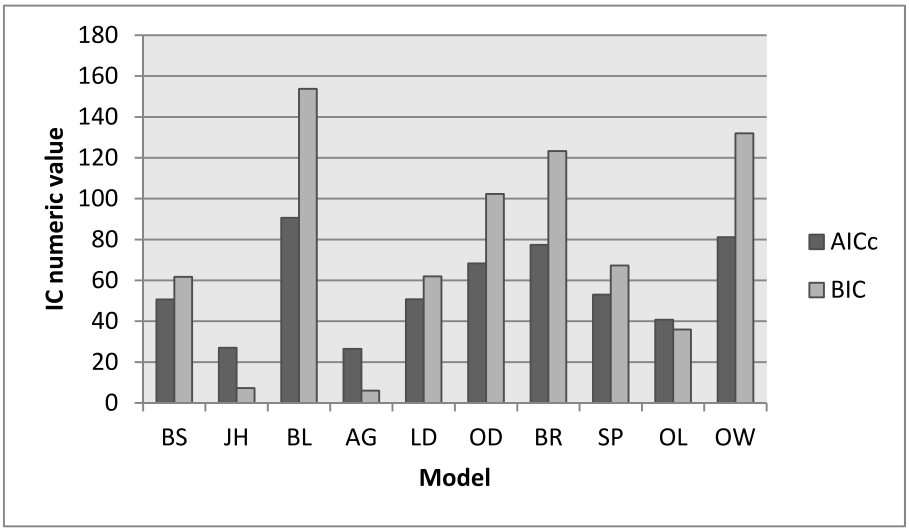


Figure 1. AIC*c* and BIC values for Dry season

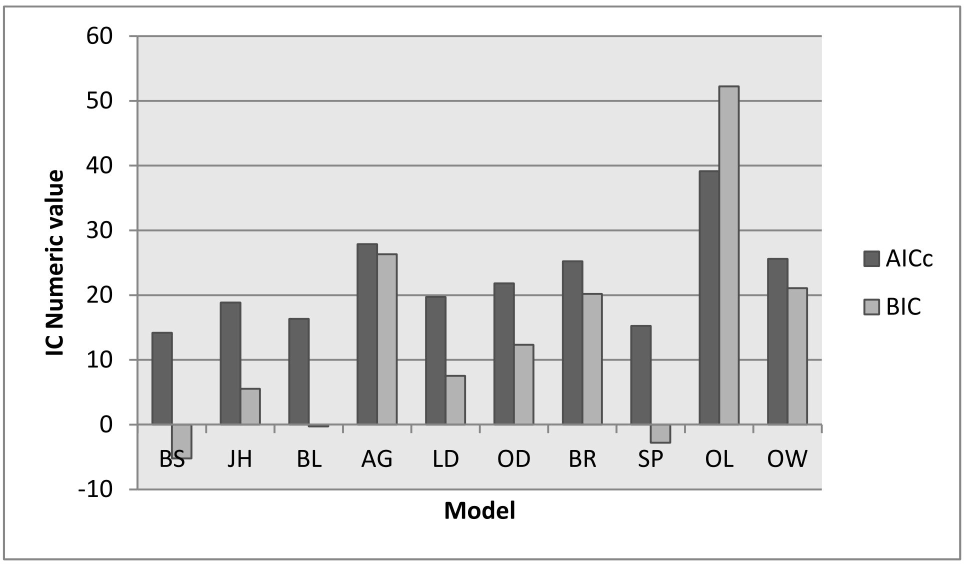


Figure 2. AIC*c* and BIC values for Rainy season

3.2 Measure of Agreement Analyses

Since the IC analysis did not give overwhelming support to any of the models considered in the study, it became necessary to conduct more analysis using recommended absolute and dimensionless error statistics in accordance with the recommendations of Johnson & Omland (2004). Results of the measure of agreement analyse are presented in Figures 3 - 8. Percent BIAS (PBIAS) is a measure of how accurately a model interprets observed data. The ideal PBIAS value is zero. Thus the closer a model PBIAS value is to zero, the better. However, when the value obtained is negative, it shows model overestimation and such value should be discountenanced. Using all 10 models, the PBIAS values obtained for the dry and rainy seasons are shown in Figures 3 and 4 respectively. Thus in the allocation of weights to the best performing models, all models that fall below zero were given zero weights while the other models were ranked according to their weights. For the dry season data, only five of the models were successful with Baecheler & Lazo (1999) model having optimum PBIAS value. For the rainy season, Bennet & Rathburn (1972) was the optimum model.

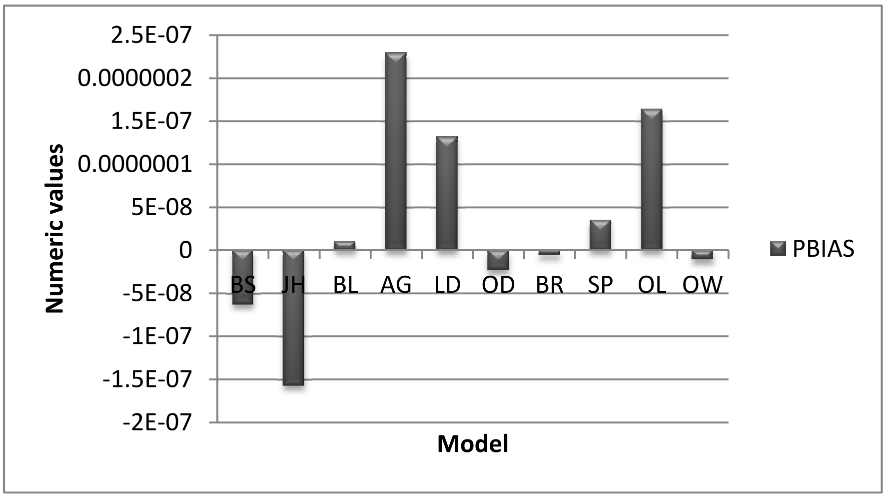


Figure 3. PBIAS for **Dry season**

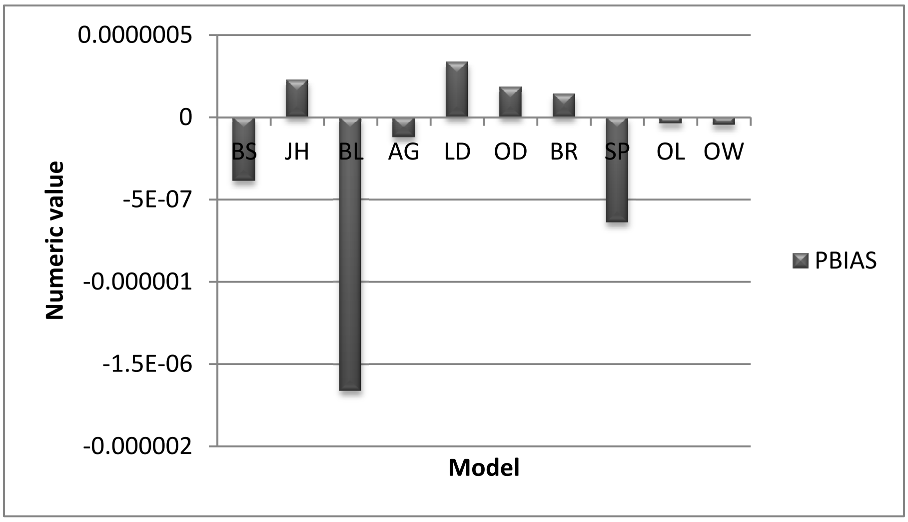


Figure 4. PBIAS for **Rainy season**

Similarly, lower RSR values are preferred. Thus, the model with the lowest RSR value was allocated the highest weights. Results of the RSR analysis for both dry and rainy seasons are presented in Figures 5 and 6 respectively. RSR is an absolute error statistic defined as the ratio between root mean square error (RMSE) and standard deviation. For the dry season, Baecheler & Lazo (1999) model had the best RSR values while Omole & Longe (2012) model had the best RSR values for the rainy season.

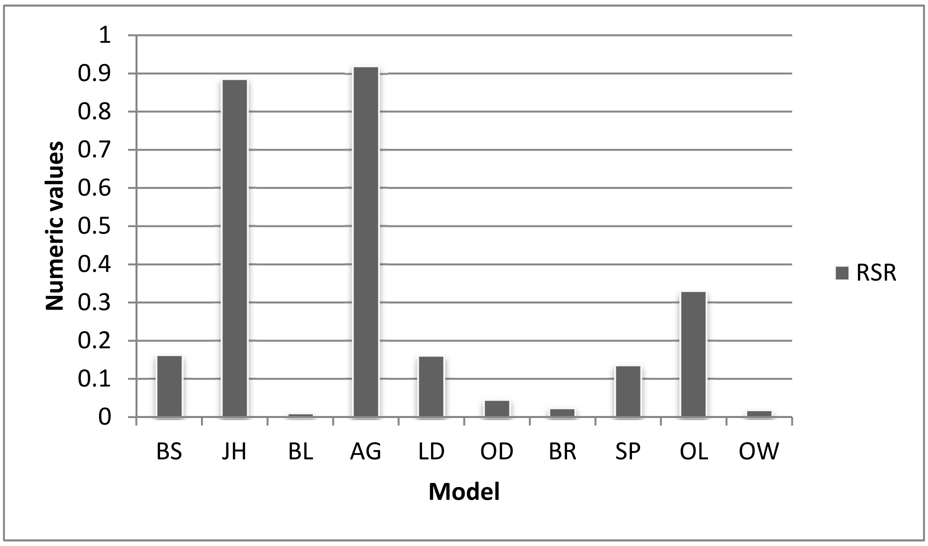


Figure 5. RSR for Dry season

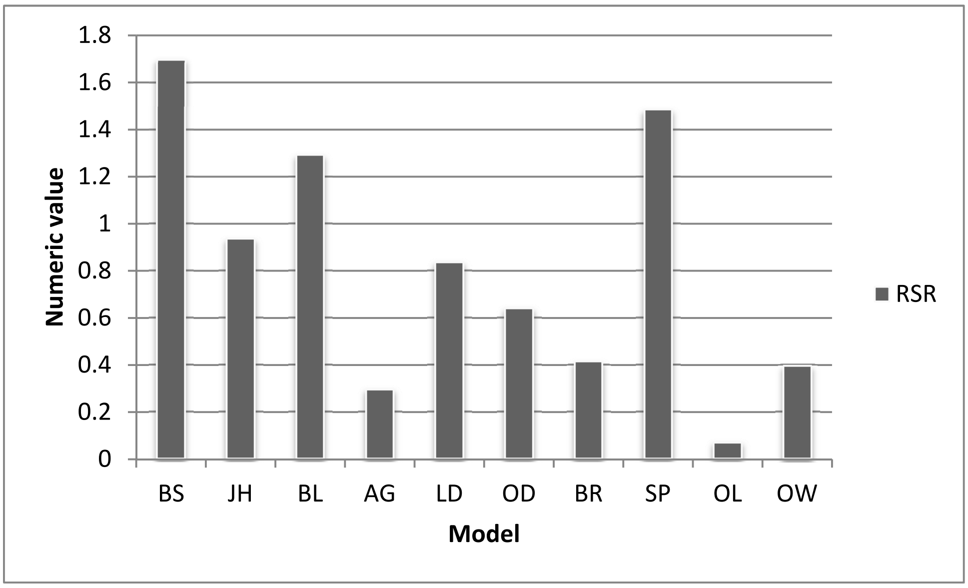


Figure 6. RSR for Rainy season

The Nash-Sutcliffe Efficiency (NSE), which is a dimensionless error statistic, measures the variance between noise and information in simulation problems. Values between 0.0 and 1.0 are optimal. However, NSE values closer to 1.0 are preferred. The results for the NSE tests for both the dry and rainy seasons are presented in Figures 7 and 8. It shows that the model with the best output among the candidate models for the dry season is Omole & Longe (2012) model while the best model for the rainy season is Owens *et al.,* (1964) model.

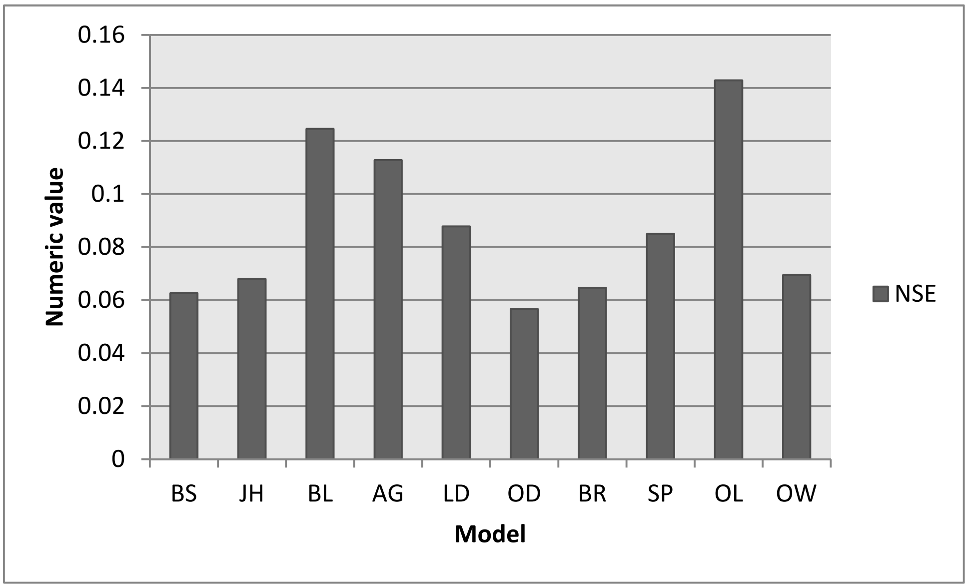


Figure 7. NSE for Dry season

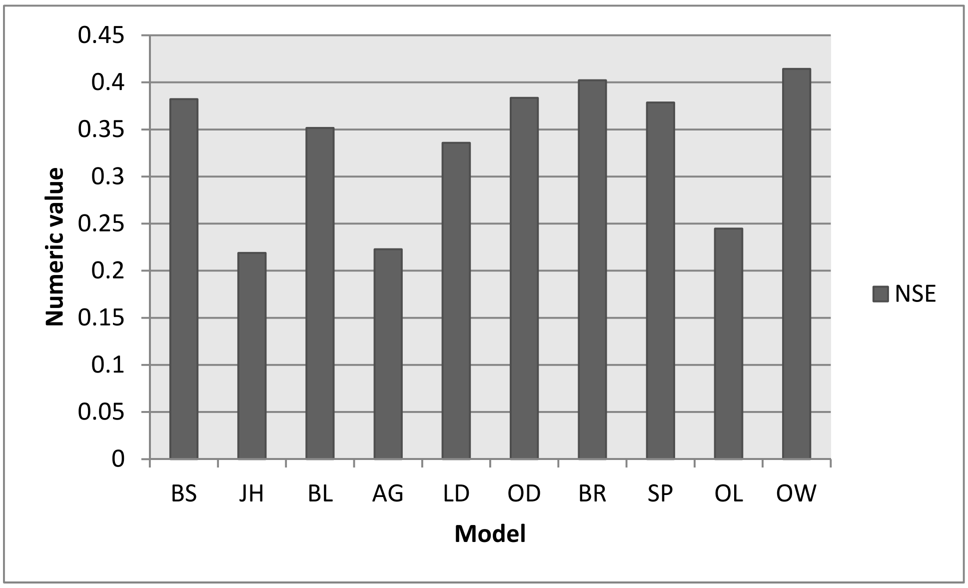


Figure 8. NSE for Rainy season

3.3 Graphic Analysis

The plots of all the models against observed data for both the dry and rainy seasons are shown in Figures 9 and 10 respectively. By visual inspection, the most representative graph was allocated the highest weight. The results of the inspection of the graphs for each model in both seasons are presented in Table 2. The graphs show that O'Connor and Dobbins (1958) model was more representative of the dry season observed data while Omole and Longe (2012) model was more representative of the rainy season data.

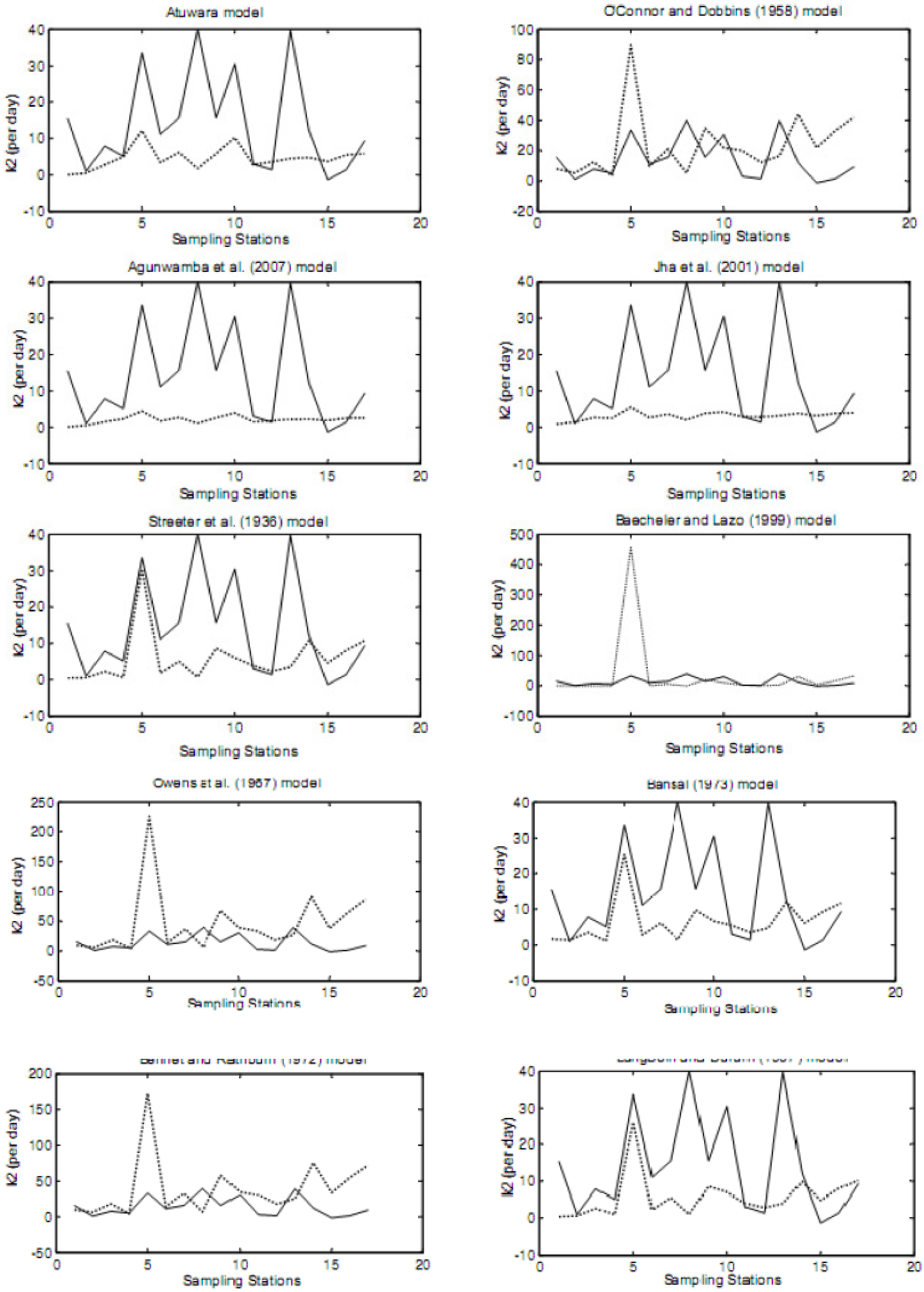


Figure 9. Plot of observed and simulated k_2 values for dry season (reproduced with permission from Omole and Longe, 2012)

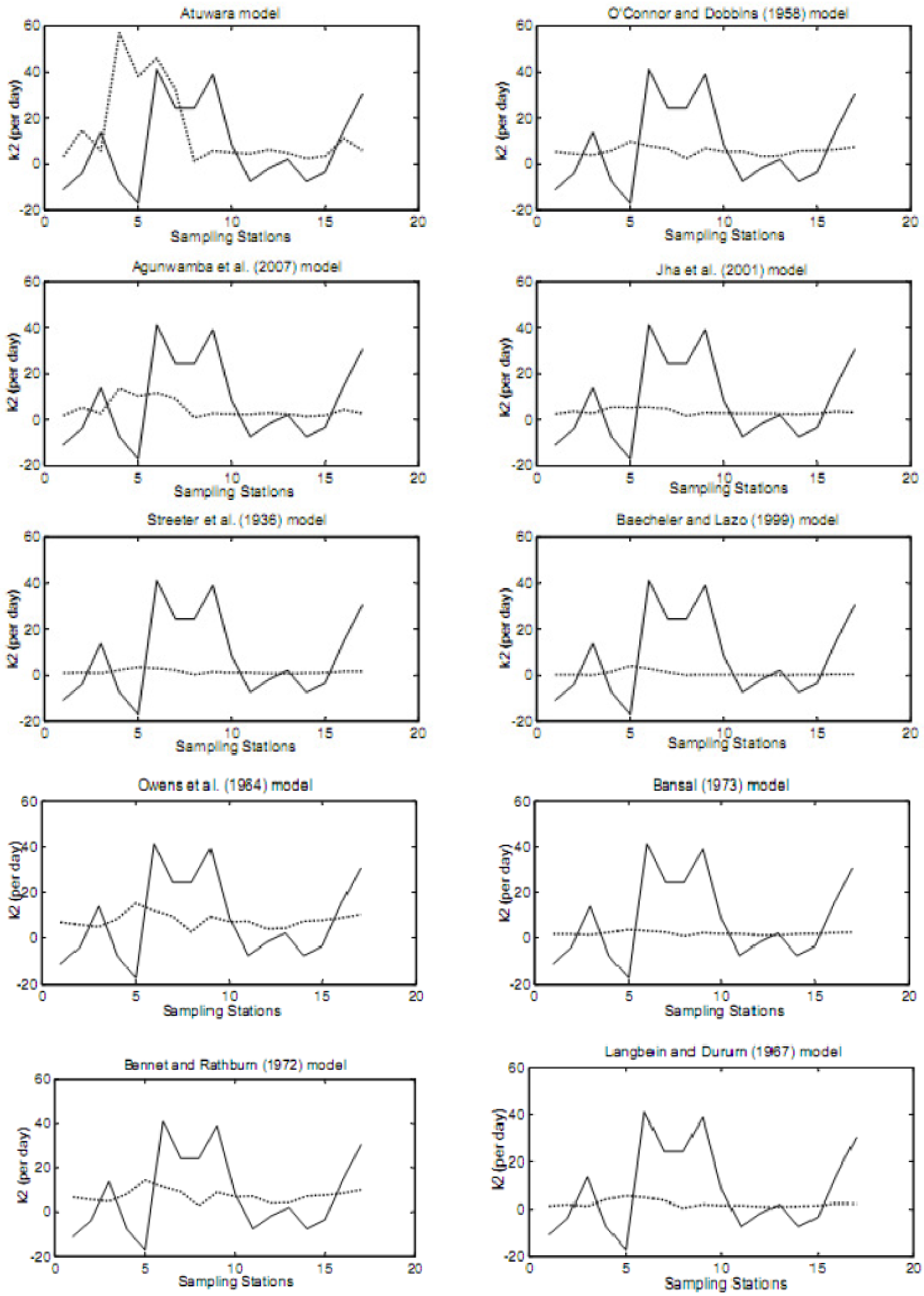


Figure 10. Plot of observed and simulated k_2 values for rainy season (reproduced with permission from Omole and Longe, 2012)

Table 2. Graphic Goodness of fit for the two data sets

s/n		OL	OD	AG	JH	SP	BL	OW	BS	BR	LD
1	JANUARY	4	10	3	3	7	1	9	6	9	6
2	JULY	10	7	9	8	3	1	7	3	7	4
3	AVERAGE SCORE FOR 2 MONTHS	7.0	8.5	6.0	5.5	5.0	1.0	8.0	4.5	8.0	5.0
4	AVERAGE SCORE FOR 2 MONTHS (%)	11.97	14.53	10.26	9.40	8.55	1.71	13.68	7.69	13.68	8.55

A summary of the result of all the three analyses were obtained by summing the weights obtained from each analysis and finding the cumulative average. This was used to rank the models in the order of performance (column 8 of Table 3). This process suggested that O'Connor and Dobbins (1958) model is the preferred model among the candidate models.

Table 3. Order of model performance in the different analysis

s/n	MODEL	MODEL SYMBOL	MODEL RANKING IN ORDER OF PERFORMANCE FOR AIC	MODEL RANKING IN ORDER OF PERFORMANCE FOR MEASURES OF AGREEMENT	MODEL RANKING IN ORDER OF PERFORMANCE FOR GRAPHICAL ANALYSIS	Cumulative percentage	AVERAGE SCORE FOR AIC, MEASURE OF AGREEMENT & GRAPH (%)
1	O'Connor & Dobbins (1958)	OD	6th	6th	1st	11.08	1st
2	Bennett & Rathburn (1972)	BR	9th	1st	2nd	10.88	2nd
3	Langbein & Dururn (1962)	LD	4th	3rd	7th	10.57	3rd
4	Omole & Longe model (2012)	OL	6th	4th	4th	10.46	4th
5	Jha *et al*., (2001)	JH	2nd	9th	6th	10.14	5th
6	Streeter *et al*., (1936)]	SP	3rd	7th	7th	10.38	5th
7	Agunwamba *et al*., (2007)	AG	4th	8th	5th	9.99	7th
8	Owens *et al.*, (1964)	OW	10th	5th	2nd	9.70	8th
9	Bansal (1973)	BS	1st	10th	9th	9.30	9th
10	Baecheler & Lazo (1999)	BL	6th	1st	10th	7.49	10th

The selection of O'Connor and Dobbins model appeals to sense for a few reasons. Butts *et al.*, (1970; p.7] believe the model was developed based on a more general theory than most other models. The model also finds wide applicability because it was designed for rivers having depths between 0.3 – 9.14 m and sluggish velocity ranging between 0.15 – 0.49 m/s [Omole et al., 2013, p. 87). River Atuwara had an average dry weather depth of 1.03 m and a dry weather flow of 0.22 m/s, which makes it to fall within the model constraints of O'Connor and Dobbins (1958) model.

4. Conclusion

The procedure for model selection procedure used in this paper was based on a combination of suggestions by different authors on the subject. The study suggested a procedure that used statistical tools (information criteria and measures of agreement) and graphical tools to rank the capacity of ten different models to predict observed stream data (Appendix). The procedure produced the top performing model which in this case was O'Connor and Dobbins (1958) model. When compared to Jha *et al.*, (2001) model which was the recommended model in Omole *et al*., (2013), it could be seen that the Jha *et al*., (2001) model was the preferred model when the test is only statistically based. However, when statistics and graphic analysis is quantitatively combined, the output differed. The procedure described in this research is appropriate for model selection in situations where there is no clear evidence of support for observed data by any particular model among competing candidate models. Although the original proponents of information criteria believe in its use as a self-sufficient model selection tool, this study has demonstrated that use of information criteria may not necessarily be the ultimate model selection tool as the different tests ranked the models differently. It is therefore recommended that re-aeration coefficient modelling scientist and software programmers research more into finding a means of compiling qualified candidate models in order to obtain more reliable results.

References

Agunwamba, J. C., Maduka, C. N., & Ofosaren, A. M. (2007). Analysis of pollution status of Amadi Creek and its management. *J. of Water Supply: Research and Technology-AQUA*, *55*(6), 427-435.

Baecheler, J. V., & Lazo, O. L. (1999). Evaluation of Water Quality Modeling Parameters: Reaeration Coefficient. IAHR, Madrid, 1999. Retrieved February 22, 2012, from http://www.iahr.org/membersonly/grazproceedings99/doc/000/000/267.htm

Bansal, M. K. (1973). Atmospheric reaeration in natural streams. *Wat. Res. Bull.*, *11*, 491-504.

Bennett, J. P., & Rathbun, R. E. (1972). Reaeration in Open Channel Flow. Professional paper, 737. USGS, Reston, VA, USA.

Bowie, G. L., Mills, W. B., Porcella, D. B., Campbell, C. L., Pagenkopf, J. R., Rupp, G. L., … Chamberlin, C. E. (1985). Rates, Constants, and Kinetics Formulations in Surface Water Quality Modeling (2nd Ed.), United States Environmental Protection Agency, Athens, GA, USA.

Burnham, K. P., & Anderson, D. R. (2004). Multimodel inference: understanding AIC and BIC in model selection. *Sociological Methods and Research, 33,* 261–304.

Butts, T. A., Schnepper, D. H., & Evans, R. L. (1970). *Dissolved oxygen resources and waste assimilative capacity of the LaGrange Pool, Illinois River.* Report of Investigation 64, Illinois State Water Survey, Urbana, USA. Retrieved November 17, 2014, from http://www.tsws.uiuc.edu/pubdoc/RI/ISWSRI-64.pdf

Churchill, M. A., Elmore, H. L., & Buckingham, R. A. (1962). The prediction of stream re-aeration rates. *Journal of Sanitary Eng. Division*, *ASCE 88*(SA4), 1-46.

Ewen, J. (2011). Hydrograph matching method for measuring model performance. *J. of Hydrol.*, *408*, 178–187.

Gayawan, E., & Ipinyomi, R. A. (2009). A Comparison of Akaike, Schwarz and R Square Criteria for Model Selection Using Some Fertility Models. *Australian Journal of Basic and Applied Sciences,* 3(4), 3524-3530.

Gupta, H. V., & Kling, H. (2011). On typical range, sensitivity, and normalization of Mean Squared Error and Nash-Sutcliffe Efficiency type metrics. Water Resour. *Res., 47,* W10601. http://dx.doi.org/10.1029/2011WR010962

Harmel, R. D., Smith, P. K., Migliaccio, K. W., Chaubey, I., Douglas-Mankin, K. R., Benham, B., … Robson, B. J. (2014). Evaluating, interpreting, and communicating performance of hydrologic/water quality models considering intended use: A review and recommendations. *Environmental Modelling & Software*, *57*(2014), 40-51.

Jha, R., Ojha, C. S. P., & Bhatia, K. K. S. (2001). Refinement of predictive reaeration equations for a typical India river. *Hydrological Processes*, *15*(6), 1047–1060.

Johnson, J. B., & Omland, K. S. (2004). Model Selection in Ecology and Evolution. *Trends in Ecology and Evolution*, *19*(2), 101-108.

Krenkel, P. A., & Orlob, G. T. (1962). Turbulent diffusion and the re-aeration coefficient. *Journal of Sanitary Eng. Division, ASCE 88*(SA2), 53-83.

Langbein, W. B., & Ddurn, W. H. (1962). *The aeration capacity of streams*. Circular S42, U.S. Geological Survey, Reston, VA, USA.

Legates, D. R., & McCabe, G. J. (1999). Evaluating the use of ‘‘goodness-of-fit’’ measures in hydrologic and hydroclimatic model validation. *Water Resour. Res.*, *35*, 233–241.

Longe, E. O., & Omole, D. O. (2008). Analysis of Pollution Status of River Illo, Ota, Nigeria. *The Environmentalist, 28*(4), 451-457.

Moog, D. B., & Jirka, G. H. (1998). Analysis of reaeration equations using mean multiplicative error. *J. Environ. Eng.*, *2*(104), 104–110.

Moriasi, D. N., Arnold, J. G., Van Liew, M. W., Bingner, R. L., Harmel, R. D., & Veith, T. L. (2007). Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. *American Society of Agricultural and Biological Engineers*, *50*(3), 885–900.

O’Connor, D. J., & Dobbins, W. E. (1958). Mechanism of Re-aeration in Natural Streams. *Transactions of the American Society of Civil Engineers*, *123,* 641-666.

Omole, D. O. (2012). Composite Goodness of Fit in Reaeration Coeffcient Modeling. *Environment and Natural Resources Research*, *2*(3), 71-83.

Omole, D.O., & Longe, E.O. (2012). Reaeration Coefficient Modeling: A Case Study of River Atuwara in Nigeria. *Research Journal of Applied Sciences Engineering and Technology*, *4*(10), 1237-1243.

Omole, D. O. (2011). *Reaeration Coefficient Modelling: Case study of River Atuwara, Ota, Nigeria.* LAP Lambert Academic Publishing GmbH & Co. KG, Saarbrücken, Germany. ISBN: 978-3-8443-3177-6.

Omole, D. O., Longe, E. O., & Musa, A. G. (2013). An Approach to Reaeration Coefficient Modeling in Local Surface Water Quality Monitoring. *Environmental Modeling and Assessment*, *18*(1), 85-94.

Owens, M., Edwards R.W., & Gibbs, J. W. (1964). Some reaeration studies in Streams. *Int'tl J. Air and Water Pollutio*n, *8*, 469-486.

Palumbo, J. E., & Brown, L. C. (2013). Assessing the Performance of Reaeration Prediction Equations. *J. Environ. Eng.*. http://dx.doi.org/10.1061/(ASCE)EE.1943-7870.0000799

Ritter, A., & Muñoz-Carpena, R. (2013). Performance evaluation of hydrological models: Statistical significance for reducing subjectivity in goodness-of-fit assessments. *Journal of Hydrology*, *480,* 33–45.

Royall, R. M. (1997). *Statistical evidence: a likelihood paradigm*, First edition; Chapman and Hall, New York, USA.

Singh, J., Knapp, H. V., Arnold, J. G., & Demissie, M. (2005). Hydrologic modelling of the Iroquous River watershed using HSPF and SWAT. *Journal of American Water Resources Association*, *41*(2), 361-375.

Streeter, H. W., Wright, C. T., & Kehr, R. W. (1936). Measures of natural oxidation in polluted streams. *Sewage Works J., 8*, 282–316.

Wang, Q., Li, S., Jia, P., Qi, C., & Ding, F. (2013). A Review of Surface Water Quality Models. *The Scientific World Journal Volume*, Article ID 231768, 7 pages.

Ye, M., Meyer, P. D., & Neuman, S. P. (2008). On model selection criteria in multimodel analysis. *Water Resources Research*, *44*, W03428. http://dx.doi.org/ 10.1029/2008WR006803

Appendix A

Algorithm for the analysis

Data structure:

1. DataSetName: one dimensional array([1…NoOfDatasets]) of Strings

	1	2
DataSetName	Jan	July

2. ModelName: one dimensional array ([1…NoOfModels]) of Strings

	1	2	3	4	5	6	7	8	9	10
ModelName	BS	JH	BL	AG	LD	OD	BR	SP	OL	OW

3. ModelQuantityID: one dimensional array ([1…NoOfModelQuantity]) of Strings

	1	2	3	4	5
ModelQuantityID	AICc	PBIAS	RSR	NSE	GGof

4. Model: three dimensional array([1…NoOfDatasets][1…NoOfModels][1…NoOfModelQuantity]) of Double

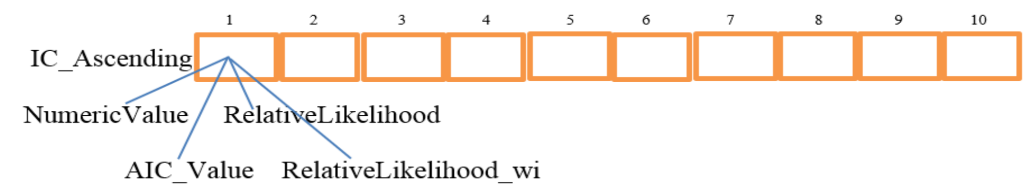
Model 1

50.65	-6.24E-08	0.1618	0.0626	6
27.02	-1.57E-07	0.8857	0.0680	3
90.63	-9.93E-09	0.0091	0.1246	1
26.48	2.29E-07	0.9202	0.1128	3
50.77	-1.32E-07	0.1603	0.08784	6
68.28	-2.28E-08	0.0455	0.0566	10
77.4	-5.76E-09	0.0236	0.0647	9
53.07	3.47E-08	0.1359	0.0850	7
40.77	1.64E-07	0.3293	0.1429	4
81.16	-1.1E-08	0.0180	0.0695	9

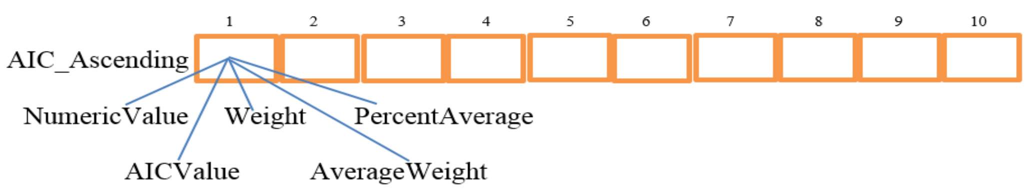
Model 2

14.19	-3.87E-07	1.6978	0.3822	3
18.85	2.24E-07	0.9352	0.2190	8
16.33	-1.66E-06	1.2919	0.3517	1
27.87	-1.20E-07	0.2950	0.2229	9
19.73	3.35E-07	0.8366	0.3359	4
21.81	1.81E-07	0.6411	0.3835	7
25.22	1.40E-07	0.4143	0.4022	7
15.24	-6.38E-07	1.4855	0.3787	3
39.14	-4.03E-08	0.0698	0.2448	10
25.6	-4.65E-08	0.3945	0.4143	7

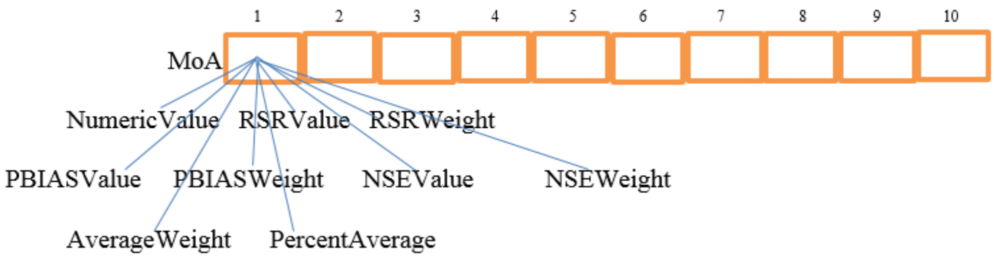
5. IC_Ascending: one dimensional array ([1…NoOfModels]) of Records. Each record has four fields namely: NumericValue (Integer), AIC_Value (Double), RelativeLikelihood (Double), RelativeLikelihood_wi (Double).



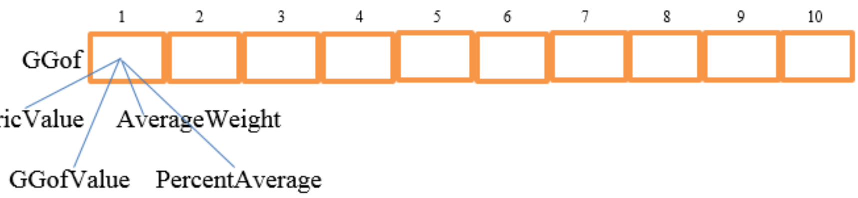
6. AIC_Ascending: one dimensional array ([1…NoOfModels]) of Records. Each record has five fields namely: NumericValue (Integer), AICValue (one dimensional array ([1…NoOfDatasets]) of Double), Weight (one dimensional array ([1…NoOfDatasets]) of Double), AverageWeight (Double), PercentAverage (Double).



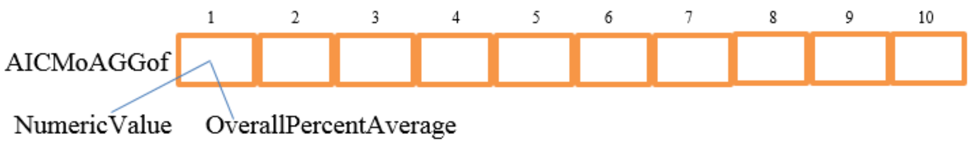
7. MoA: one dimensional array ([1…NoOfModels]) of Records. Each record has nine fields namely: NumericValue (Integer), PBIASValue (one dimensional array ([1…NoOfDatasets]) of Double), PBIASWeight (one dimensional array ([1…NoOfDatasets]) of Double), RSRValue (one dimensional array ([1…NoOfDatasets]) of Double), RSRWeight (one dimensional array ([1…NoOfDatasets]) of Double), NSEValue (one dimensional array ([1…NoOfDatasets]) of Double), NSEWeight (one dimensional array ([1…NoOfDatasets]) of Double), AverageWeight (Double), PercentAverage (Double).



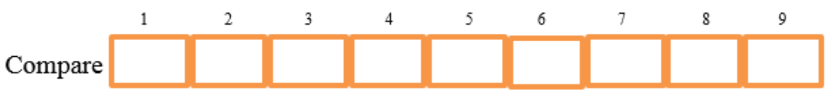
8. GGof: one dimensional array ([1…NoOfModels]) of Records. Each record has four fields namely: NumericValue (Integer), GGofValue (one dimensional array ([1…NoOfDatasets]) of Double), AverageWeight (Double), PercentAverage (Double).



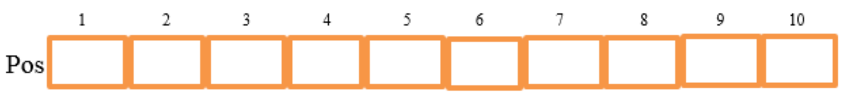
9. AICMoAGGof: one dimensional array ([1…NoOfModels]) of Records. Each record has two fields namely: NumericValue (Integer), OverallPercentAverage (Double).



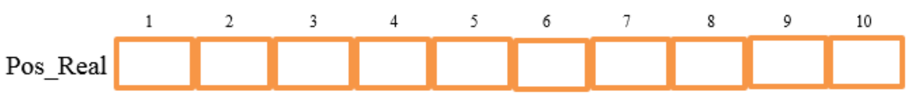
10. Compare: one dimensional array ([1…(NoOfModels-1)]) of Float



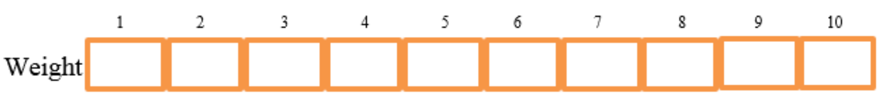
11. Pos: one dimensional array ([1…NoOfModels]) of Integer



12. Pos_Real: one dimensional array ([1…NoOfModels]) of Integer



13. Weight: one dimensional array ([1…NoOfModels]) of Integer



Algorithm:

STEP 1:

// **Initialize** all variables

i=0, j=0, k=0, m=0, DeltaI=0, SumOfRelativeLikelihood=0, TotalWeight=0, SumOfAllAverageWeight=0, DataSetName[],ModelName[], ModelQuantityID[], Model[][][], IC_Ascending[], AIC_Ascending[], MoA[], GGof[], AICMoAGGoF[], Compare[], Pos[], Pos_Real[], Weight[]

STEP 2: **Input** NoOfDatasets, NoOfModels, NoOfModelQuantity

STEP 3:

// Compute or **Store all values** for all Model quantities in Model[i][j][k]

For i = 1 to NoOfDatasets

Begin

For j = 1 to NoOfModels

Begin

For k = 1 to NoOfModelQuantity

Begin

Compute and Store Model[i][j][k]

End

End

End

$$// AIC_c = -2\ln\left[L\left(\hat{\theta}|y\right)\right] + 2p\left(\frac{n}{n-p-1}\right), \quad PBIAS = \left[\frac{\sum_{i=1}^{n}\left(y_i^{obs} - y_i^{sim}\right)\times 100}{\sum_{i=1}^{n}\left(y_i^{obs}\right)}\right]$$

$$// RSR = \frac{RMSE}{\sigma^2}, \quad NSE = 1 - \left[\frac{\sum_{i=1}^{n}\left(y_i^{obs} - y_i^{sim}\right)^2}{\sum_{i=1}^{n}\left(y_i^{obs} - \bar{y}\right)^2}\right]$$

STEP 4:

// Check for model with **overwhelming support** for all Datasets

// Extract AICc values into array IC_Ascending

For i = 1 to NoOfDatasets

Begin

k =1 // 1^{st} Model Quantity ie AICc

For j = 1 to NoOfModels

Begin

IC_Ascending[j].NumericValue = j // Model numeric values: BS=1, JH=2, etc

IC_Ascending[j].AIC_Value = Model[i][j][k] // Model AICc value

End

Sort IC_Ascending in Ascending order of its IC_Ascending[].AIC_Value

// Compute RelativeLikelihood_wi

For j = 1 to NoOfModels

Begin

DeltaI = IC_Ascending[j].AIC_Value - IC_Ascending[1].AIC_Value // Model perf based on minimum value

IC_Ascending[j].RelativeLikelihood = $e^{0.5*DeltaI}$

SumOfRelativeLikelihood = SumOfRelativeLikelihood + IC_Ascending[j].RelativeLikelihood

End

For j = 1 to NoOfModels

Begin

IC_Ascending[j].RelativeLikelihood_wi = IC_Ascending[j].RelativeLikelihood/SumOfRelativeLikelihood

End

For j = 1 to NoOfModels

```
    Begin
      If (IC_Ascending[j].RelativeLikelihood_wi ≥ 0.9)
        Begin
          print ModelName[IC_Ascending[j].NumericValue] "has overwhelming support"
          stop
        End
    End
  End      // End of overwhelming support for all Datasets
```

// **AIC Analysis** for all Datasets

STEP 5:

```
// Extract AICc values for all Datasets unto array AIC_Ascending
For i = 1 to NoOfDatasets
Begin
k =1 // 1st Model Quantity ie AICc
  For j = 1 to NoOfModels
    Begin
      AIC_Ascending[j].NumericValue = j                  // Model numeric values: BS=1, JH=2, etc
      AIC_Ascending[j].AICValue[i] = Model[i][j][k]     // Model AICc value
    End
End
```

STEP 6:

```
// Sort and Allocate Weight for AICc
For i = 1 to NoOfDatasets
Begin
  Sort AIC_Ascending in Ascending order of AIC_Ascending[].AICValue[i]
  Call Compare&PositionAlg(AIC_Ascending)       //Compares & Position AIC_Ascending wrt
AIC_Ascending[].AICValue[i]
  Call WeightAlg(AIC_Ascending) //Allocate weight with proper positioning based on output of Compare&PositionAlg &
store weight in AIC_Ascending[].Weight[i]
End
```

STEP 7:

```
// Compute AICc Average
For j = 1 to NoOfModels
Begin
 For i = 1 to NoOfDatasets
  Begin
   TotalWeight = TotalWeight + AIC_Ascending[j].Weight[i]
  End
 AIC_Ascending[j].AverageWeight = TotalWeight/NoOfDatasets
 SumOfAllAverageWeight = SumOfAllAverageWeight + AIC_Ascending[j].AverageWeight
End
```

STEP 8:

```
// Compute AICc %tage Average
For j = 1 to NoOfModels
Begin
 AIC_Ascending[j].PercentAverage = (AIC_Ascending[j].AverageWeight/SumOfAllAverageWeight) * 100
End
```

STEP 9:

```
// To measure model perf based of AICc with positioning, sort AIC_Ascending in Descending order of
// AIC_Ascending[].PercentAverage & pass the sorted AIC_Ascending[] to Compare&PositionAlg and PositionAlg
// respectively ie Sort AIC_Ascending in Descending order of AIC_Ascending[].PercentAverage
Call Compare&PositionAlg(AIC_Ascending)  // Compares & Position AIC_Ascending wrt AIC_Ascending[].PercentAverage
Call PositionAlg(AIC_Ascending)          //Based on output of Compare&PositionAlg,it properly position models in AIC_Ascending wrt Ascending[].PercentAverage
                                         // highest PercentAverage => 1st position. If there are two 1st positions, then next is 3rd position, ie no 2nd position
print ModelName[AIC_Ascending[1].NumericValue] "is the best AICc model"
```

// **MoA Analysis** for all Datasets

STEP 10:

```
// Extract PBIAS, RSR, NSE values for all Datasets unto array MoA
For i = 1 to NoOfDatasets
Begin
  For j = 1 to NoOfModels
    Begin
      k = 1
       MoA[j].NumericValue = j                     // Model numeric values: BS=1, JH=2, etc
       MoA[j].PBIASValue[i] = Model[i][j][k+1]     // Model PBIAS value
       MoA[j].RSRValue[i] = Model[i][j][k+2]       // Model RSR value
       MoA[j].NSEValue[i] = Model[i][j][k+3]       // Model NSE value
    End
End
```

STEP 11:

```
// Sorting and Weight Allocation for PBIAS
For i = 1 to NoOfDatasets
m = 0
Begin
  Sort MoA in Ascending order of MoA[].PBIASValue[i]
  For j = 1 to NoOfModels
   Begin
    If (MoA[j].PBIASValue[i]< 0)
      Begin
       MoA[j].PBIASWeight[i] = 0
      End
```

```
    Else
     Begin
      MoA[j].PBIASWeight[i] = NoOfDatasets – m
      m++
     End
  End
End
```

STEP 12:

```
// Sorting and Weight Allocation for RSR
For i = 1 to NoOfDatasets
Begin
  Sort MoA in Ascending order of MoA[].RSRValue[i]
  Call Compare&PositionAlg(MoA)      // Compares & Position MoA wrt MoA[].RSRValue[i]
  Call WeightAlg(MoA)                //Allocate weightBased on output of Compare&PositionAlg,& store weight in
MoA[].RSRWeight[i]
End
```

STEP 13:

```
// Sorting and Weight Allocation for NSE
For i = 1 to NoOfDatasets
Begin
  Sort MoA in Ascending order of MoA[].NSEValue[i]
  Call Compare&PositionAlg(MoA)      // Compares & Position MoA wrt MoA[].NSEValue[i]
  Call WeightAlg(MoA)                //Allocate weightBased on output of Compare&PositionAlg,& store weight in
MoA[].NSEWeight[i]
End
```

STEP 14:

```
// Compute MoA Average
SumOfAllAverageWeight = 0
For j = 1 to NoOfModels
TotalWeight = 0
Begin
 For i = 1 to NoOfDatasets
 Begin
   TotalWeight = TotalWeight + MoA[j].PBIASWeight[i]+ MoA[j].RSRWeight[i]+ MoA[j].NSEWeight[i]
 End
 MoA[j].AverageWeight = TotalWeight/NoOfDatasets
 SumOfAllAverageWeight = SumOfAllAverageWeight + MoA[j].AverageWeight
End
```

STEP 15:

```
// Compute MoA %tage Average
For j = 1 to NoOfModels
Begin
```

```
 MoA[j].PercentAverage = (MoA[j].AverageWeight/SumOfAllAverageWeight) * 100
End
```

STEP 16:

```
// To measure model perf based of MoA with positioning, sort MoA in Descending order of
// MoA[].PercentAverage & pass the sorted MoA[] to Compare&PositionAlg and PositionAlg
// respectively ie Sort MoA in Descending order of MoA[].PercentAverage
Call Compare&PositionAlg(MoA)   // Compares & Position MoA wrt MoA[].PercentAverage
Call PositionAlg(MoA)           //Based on output of Compare&PositionAlg,it properly position models in MoA wrt MoA[].PercentAverage
                                // highest PercentAverage => 1st position. If there are two 1st positions, then next is 3rd position, ie no 2nd position
print ModelName[MoA[1].NumericValue] "is the best MoA model"
```

// **GGof Analysis** for all Datasets

STEP 17:

```
// Extract GGof values for all Datasets unto array GGof
For i = 1 to NoOfDatasets
Begin
  For j = 1 to NoOfModels
    Begin
     k = 5                                // 5th Model Quantity is GGoF
      GGof[j].NumericValue = j            // Model numeric values: BS=1, JH=2, etc
      GGof[j].GGofValue[i] = Model[i][j][k]  // Model GGof value
    End
End
```

STEP 18:

```
// Compute GGof Average
SumOfAllAverageWeight = 0
For j = 1 to NoOfModels
TotalWeight = 0
Begin
 For i = 1 to NoOfDatasets
 Begin
   TotalWeight = TotalWeight + GGof[j].GGofValue[i]
 End
GGof[j].AverageWeight = TotalWeight/NoOfDatasets
SumOfAllAverageWeight = SumOfAllAverageWeight + GGof[j].AverageWeight
End
```

STEP 19:

```
// Compute GGof %tage Average
For j = 1 to NoOfModels
Begin
 GGof[j].PercentAverage = (GGof[j].AverageWeight/SumOfAllAverageWeight) * 100
```

End

STEP 20:

// To measure model perf based of GGof with positioning, sort GGof in Descending order of

// GGof[].PercentAverage & pass the sorted GGof[] to Compare&PositionAlg and PositionAlg

// respectively ie Sort GGof in Descending order of GGof[].PercentAverage

Call Compare&PositionAlg(GGof) // Compares & Position GGofwrt GGof[].PercentAverage

Call PositionAlg(GGof) //Based on output of Compare&PositionAlg,it properly position models in GGoF wrt GGof[].PercentAverage

// highest PercentAverage => 1st position. If there are two 1st positions, then next is 3rd position, ie no 2nd position

print ModelName[GGof[1].NumericValue] "is the best GraphicalGoodness of fit model"

// **AICc, MoA &GGofMerging: Final Analysis**

STEP 21:

// Sort AIC_Ascending, MoA & GGoF in Ascending order of NumericValue (model name) because as at the last time

// these arrays are processed, they may not be in order or may be in different order

Sort AIC_Ascending in Ascending order of AIC_Ascending[].NumericValue

Sort MoA in Ascending order of MoA[].NumericValue

Sort GGof in Ascending order of GGof[].NumericValue

STEP 22:

// Extract AICc PercentAverage, MoA PercentAverage& GGof PercentAverage. Then calculate the Overall

// Percentage Average for all models

For j = 1 to NoOfModels

Begin

AICMoAGGof[j].NumericValue = j // Model numeric values: BS=1, JH=2, etc

AICMoAGGof[j].OverallPercentAverage = (AIC_Ascending[j].PercentAverage + MoA[j].PercentAverage + GGof[j].PercentAverage)/3

End

STEP 23:

// Sorting & Positioning based on overall model performance

// Sort AICMoAGGof in Descending order of AICMoAGGof[].OverallPercentAverage

Call Compare&PositionAlg(AICMoAGGof) // wrt AICMoAGGof[].OverallPercentAverage

Call PositionAlg(AICMoAGGof) // highest OverallPercentAverage => 1st position. If there are two 1st positions, then next is 3rd position, ie no 2nd position

print ModelName[AICMoAGGof[1].NumericValue] "is the best overall model"

<u>Compare&PositionAlg(Array) Algorithm:</u>

```
For j = 1 to (NoOfModels-1)
Begin
  If (Array[j+1] = Array[j])
    Begin
      Compare[j] = 0
    End
```

```
  Else
    Begin
      Compare[j] = 1
    End
End
Pos[1] = 1
For j = 2 to NoOfModels
Begin
 If (Compare[j-1] = 1)
  Begin
    Pos[j] = Pos[j-1]+1
  End
 Else
  Begin
   Pos[j] = Pos[j-1]
  End
End
```

WeightAlg Algorithm:

```
Similar = 1
Weight[1] = NoOfModels
For j = 1 to (NoOfModels-1)
 Begin
  If (Pos[j] ≠Pos[j+1])
   Begin
    If (Similar ≠ 1)
     Begin
       Weight[j+1] =Weight[j] – Similar
       Similar = 1
      End
    Else
     Begin
      Weight[j+1] = Weight[j] – 1
      Similar = 1
     End
    End
  Else
    Begin
      Weight[j+1] = Weight[j]
       Similar++
    End
End
```

PositionAlgAlgorithm:

```
Similar = 1
Pos_Real[1] = 1
For j = 1 to (NoOfModels-1)
Begin
  If (Pos[j] ≠ Pos[j+1])
    Begin
      If (Similar ≠ 1)
        Begin
            Pos_Real[j+1] = Pos_Real[j] + Similar
         Similar = 1
        End
      Else
        Begin
         Pos_Real[j+1] = j + 1
         Similar = 1
        End
    End
  Else
    Begin
      Pos_Real[j+1] = Pos_Real[j]
      Similar++
    End
End
```