

**DOWNWARD APPROACH FOR STREAMFLOW ESTIMATION,
FORECASTING FOR SMALL-SCALE TO LARGE-SCALE
CATCHMENTS: LEARNING FROM DATA**

BASNAYAKE MUDIYANSELAGE LEKHANGANI ARUNODA BASNAYAKE
(B. Sc. Eng. (Hons), University of Peradeniya, Sri Lanka)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2012

ACKNOWLEDGEMENTS

First and foremost, I wish to express my sincere gratitude to my supervisor, Associate Prof. Vladan Babovic for his guidance, valuable advices, and constant support, which lead to the completion of my doctoral study. He has been an excellent advisor for me during my years in National University of Singapore.

I express my sincere appreciation to Dr. Rao Raghuraj, for his guidance, encouragements, and helpful suggestions during the initial stage of my research. I am also grateful to the other members of my dissertation committee, Prof. Cheong Hin Fatt, and Assistant Prof. Chui Ting Fong May, whose suggestions and constructive comments guided me through the research.

I am grateful to all my laboratory mates and my friends who have helped during my doctoral study at National University of Singapore. Heartfelt gratitude is extended for the entire family members of Civil Engineering Department. Very special thank goes for the entire family members of Singapore-Delft Water Alliances (SDWA). I would like to express my sincere thanks to all who, directly or indirectly, contributed in many ways to the success of my research.

I thankfully acknowledge the National University of Singapore for granting me research scholarship to pursue the degree of Doctor of Philosophy. I gratefully acknowledge the financial support of the Singapore-Delft Water Alliance (SDWA).

Last but not least, I would like to thank my parents and my husband for their love, inspirations and constant support during this intensive learning period and in every step of my life.

TABLE OF CONTENTS

	Page No.
ACKNOWLEDGEMENTS	i
TABLE OF CONTENTS	ii
SUMMARY	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF SYMBOLS	xiii
CHAPTER 1: INTRODUCTION	1
1.1 Rainfall-runoff (R-R) process modelling	1
1.1.1 Process-based models	2
1.1.2 Data driven models (DDMs)	2
1.2 Problem statement	3
1.3 Objectives of the study	5
1.4 Organization of the thesis	6
CHAPTER 2: LITERATURE REVIEW	8
2.1 Runoff generating processes	8
2.1.1 Process scale	10
2.1.2 Hydrological process scales	9
2.1.3 Observation (Measurement) scale	12
2.2 Rainfall-runoff (R-R) process conceptualization approaches	13

2.2.1 Upward approach	13
2.2.2 Downward approach	13
2.3 Rainfall-runoff (R-R) modelling with data driven techniques	15
2.4 Streamflow forecasting with data driven techniques	19
2.4.1 Distributed and lumped flow routing	21
2.4.2 Global and cluster-based flow routing	23
2.5 Effect of data resolution on rainfall-runoff (R-R) process approximation	26
2.6 Accuracy of multi-step-ahead forecasts	28
2.7 Artificial neural networks (ANNs)	29
2.7.1 Input determination	30
2.7.2 Training neural nets	32
2.7.3 Extrapolation capability	33
2.7.4 Optimal model complexity	33
2.8 Summary	37
CHAPTER 3: EFFECT OF DATA TIME INTERVAL ON RAINFALL-RUNOFF (R-R) MODELLING	38
3.1 Introduction	38
3.2 Case study	38
3.3 Input determination	39
3.4 Forecasting models	41
3.5 Performances of rainfall-runoff (R-R) models	42
3.5.1 Effect of data time interval on forecasting accuracy	44
3.5.2 Iterative and direct forecasting	48

3.6 Conclusions	50
CHAPTER 4: MODULAR DATA DRIVEN APPROACH FOR RAINFALL-RUNOFF (R-R) MODELLING	52
4.1 Introduction	52
4.2 Case study	53
4.3 Identification of hydrological regimes: Self-Organizing Maps (SOMs)	53
4.4 Forecasting models	54
4.4.1 Linear forecasting models	54
4.4.2 Nonlinear forecasting model: Artificial Neural Networks (ANNs)	55
4.5 Performances of global and modular rainfall-runoff (R-R) models	55
4.5.1 Model performance in rainfall-runoff (R-R) process representation	55
4.5.2 Linear and nonlinear model performances in global and modular model representations	65
4.5.3 Model performance in multi-step-ahead forecasts	69
4.5.4 Extrapolation capability of global and modular models	73
4.6 Conclusions	75
CHAPTER 5: FLOW ROUTING WITH DATA DRIVEN MODELS	77
5.1 Introduction	77
5.2 Description of the White river catchment	77
5.3 Input determination	78
5.4 Sequential flow routing method	81
5.5 Cluster-based flow routing	86

5.5 Conclusions	90
CHAPTER 6: CONCLUSIONS AND RECOMMENDATIONS	92
REFERENCES	95
LIST OF PUBLICATIONS	103

SUMMARY

Data driven models (DDMs) are recognized as models that offer computationally fast yet sufficiently accurate solutions for modelling complex dynamical systems. In so doing, DDMs are used in operational management systems. Current applications of DDMs on rainfall-runoff (R-R) process modelling are limited to finding a function for all runoff generating instances. These studies are rather general and not specific enough to capture the temporal and spatial variation of R-R processes. Therefore, from the operational perspective, it is highly imperative to find out the means of improving R-R process representation of DDMs and other influential factors on forecasting accuracy. The objectives of this research were: (1) to review the data driven streamflow estimation applications to understand the reasons for the model-attributed estimation errors, (2) to investigate the effect of data time interval and model complexities on streamflow estimation and forecasting, (3) to classify temporally dominant runoff generating processes, (4) to develop and evaluate a modular data driven model for estimating streamflow of lump catchments, (5) to develop and evaluate a sequential flow routing method, and (6) to investigate the applicability of cluster-based modelling for distributed flow routing. Artificial neural networks (ANNs) was the data driven modelling method in this research.

Orgeval catchment of France was chosen to illustrate the problems associated with lumped catchment R-R models. First, the effect of data time interval was investigated using 1 hour (hr), 2 hr, and 3 hr sampled data. Two analyses were performed using absolute discharge data (Q) and differenced discharge (dQ) data. Both analyses showed that accuracy improved with refined data and results were comparable. However, errors of ANN model trained with Q data were much higher in multi-step-ahead forecasts and in out-of-range forecasts. Models trained with dQ data

tend to generate more accurate forecasts. It was found that both improvements in runoff estimation, i.e., at one-step-ahead forecasts, and error accumulation property have significant impact on multi-step-ahead forecasts. The range of data time interval is not continuous and fine sampled data can deteriorate the model estimations due to the noise in data. This needs further investigation.

This thesis also presents a systematic approach for streamflow estimation in lump catchments; firstly to identify the temporally dominant processes and secondly to represent each local region by separate models; in an attempt to obtain improved estimation. Classification results showed that dQ and rainfall model inputs successfully identified the temporally dominant processes. Application of classified inputs to locally specialized models showed that the proposed modular model approach is feasible and effective. Improvement in predictability with modular model approach will depend on the degree of complexity of R-R process.

Finally, possibility of extending the research basis of lump catchment models into large-scale catchments was examined. A sequential flow routing model was developed for the West Fork of the White river, Indiana. In the first part of the study, single-station models were developed, firstly using the nearest upstream station data and secondly with all existing upstream flow data. Then, single-station models were sequentially applied to estimate the downstream flows. The model performance was evaluated with different data time intervals. Comparison of model results indicated that single river reach model performance could be improved with temporally refined data. In the second part of this study, cluster-based modelling was applied to improve the flow estimations. Simulation results of this analysis indicated that cluster-based modelling was a promising method to improve the streamflow forecasts. The proposed

approach was found to improve the forecasts over longer prediction horizon. This can be coupled with hydrological information to improve intra-catchment process variations.

It is believed that this research contribution will provide the basis for subsequent studies on data driven R-R process modelling and for other related data driven applications.

LIST OF TABLES

		Page
Table 3.1	Q-ANN model performance with data time interval.	46
Table 3.2	dQ-ANN model performance with data time interval.	46
Table 4.1	Parts of the hydrograph represented by each classification.	62
Table 4.2	Error accumulated due to the classification error in dQ-MNN models.	71
Table 5.1	Statistics of the streamflow time series data (m^3/s).	78
Table 5.2	Performances of single station models of Centerton and Newberry.	85
Table 5.3	Difference of statistical measures of GM_{SS} and GM_{MS} models with data time interval.	86

LIST OF FIGURES

		Page
Figure 2.1	Runoff generating processes (Maidment, 1993).	9
Figure 2.2	Process scale in time. (a) Duration (temporal extent of the process); (b) Temporal cycle; (c) Correlation time (Bloschl and Sivapalan, 1995).	10
Figure 2.3	Major controls on runoff generation mechanisms (Dunne, 1983).	10
Figure 2.4	Characteristic space-time scales of hydrological processes (Bloschl and Sivapalan, 1995).	11
Figure 2.5	Observation scale in time. (a) Temporal extent; (b) Integration time; (c) Data time interval (Bloschl and Sivapalan, 1995).	12
Figure 2.6	Dependency of observation scale and process scale (Bloschl and Sivapalan, 1995).	13
Figure 2.7	The representation of a process in data driven models (Solomatine and Ostfeld, 2008).	15
Figure 2.8	(a) Separation of sources of streamflow on an idealized hydrograph, (b) Sources of streamflow during a dry period, and (c) during a rainfall event. (Maidment, 1993).	18
Figure 2.9	Relative importance of the sub-processes at different times (Mays, 2005).	18
Figure 2.10	(a) Propagation of a flood wave, (b) Storage-discharge relationship.	24
Figure 2.11	Three-layered multi-layer perceptron (MLP).	30
Figure 2.12	Illustration of the bias/variance trade-off (Nelles, 2001).	35
Figure 2.13	Training and testing error variation with the model complexity.	36
Figure 2.14	Basic building block of MLP (Xiang et al., 2005).	37
Figure 3.1	The Orgeval catchment (Anctil et al., 2009).	39
Figure 3.2	Autocorrelation coefficient variation of absolute discharge (Q) data, and cross-correlation coefficient variation of absolute discharge (Q) and rainfall data for 1hr, 2hr, and 3hr sampled data.	40

Figure 3.3	Autocorrelation coefficient variation of differenced discharge (dQ) data, and cross-correlation coefficient variation of differenced discharge (dQ) and rainfall data for 1hr, 2hr, and 3hr sampled data.	40
Figure 3.4a	Performances of ANN models for hourly data.	43
Figure 3.4b	Performances of ANN models for 2 hr sampled data.	43
Figure 3.4c	Performances of ANN models for 3 hr sampled data.	43
Figure 3.5	Absolute error (scaled) produced by Q-ANN and dQ-ANN models.	44
Figure 3.6	Effect of data time interval (ΔT) on model error.	48
Figure 3.7	Iterative and direct forecasting performances of Q-ANN models.	49
Figure 3.8	Iterative and direct forecasting performances of dQ-ANN models.	50
Figure 4.1	Schematic representation of the proposed modelling approach.	54
Figure 4.2	Performances of the Q-MNN models.	56
Figure 4.3a	Position of classes in; (a) 2-class, and (b) 3-class classifications.	57
Figure 4.3b	Position of classes in; (a) 4-class, and (b) 6-class classifications.	58
Figure 4.4	Performances of the dQ-MNN models.	59
Figure 4.5a	Position of classes in; (a) 2-class, (b) 3-class, and (b) 4-class classifications.	60
Figure 4.5b	Position of classes in; (a) 6-class, and (b) 8-class classifications.	61
Figure 4.6	(a) Rainfall pattern; (b) dQ pattern.	63
Figure 4.7	Improvement in forecasts of local models compared to global models.	64
Figure 4.8	Performances of ARX and ANN models in global model (GM) and modular (MM) representations.	65
Figure 4.9	Improvement of forecasts in nonlinear local models compared to linear local models.	66

Figure 4.10	Flow duration curve for Orgeval catchment	67
Figure 4.11	Performances of the dQ-MNN models.	70
Figure 4.12a	Error accumulated due to the classification error in individual classes of (a) dQ-MNN-C2, (b) dQ-MNN-C3, and (c) dQ-MNN-C4 models.	72
Figure 4.12b	Error accumulated due to the classification error in individual classes of (a) dQ-MNN-C6, and (c) dQ-MNN-C8 models.	73
Figure 4.13	Performance of models for out-of range data.	74
Figure 5.1	White river catchment.	77
Figure 5.2	Streamflow time series of the year 1992.	79
Figure 5.3a	Corss-correlation coefficient and auto-correlation coefficient variation for Q data.	80
Figure 5.3b	Corss-correlation coefficient and auto-correlation coefficient variation for dQ data.	81
Figure 5.4	Contribution of upstream flows on the streamflow estimations at Newberry (N), Centerton (C), and Indianapolis (I).	82
Figure 5.5	Streamflow estimation at downstream stations.	83
Figure 5.6	Performance of Q-ANN and dQ-ANN models in estimating, forecasting flows at Newberry.	84
Figure 5.7a	Class positions in Centerton discharge time series for 2-class classification.	87
Figure 5.7b	Class positions in Centerton discharge time series for 4-class classification.	88
Figure 5.9	Performances of global model (GM) and modular neural network (MNN) models at Indianapolis (I), Centerton (C), and Newberry (N).	89

LIST OF SYMBOLS

R-R	Rainfall-runoff
ANNs	Artificial neural networks
DDMs	Data driven models
ARX	Auto Regressive with eXogeneous
ΔT	Data time interval
SOMs	Self-organizing maps
MLP	Multi-layer perceptron
Q	Absolute discharge
dQ	Differenced discharge
R	Rainfall
MAE	Mean absolute error
r	Correlation coefficient
FFNN	Feed-forward neural network
RNN	Recurrent neural network
Q_o	Observed discharge
Q_p	Predicted discharge
$\overline{Q_o}$	Mean observed discharge
$\overline{Q_p}$	Mean predicted discharge
AE	Absolute error
I	Inflow
K	Flow travel time

S	Storage
x	Weighting factor
LM	Local model
MM	Modular model
GM	Global model
SNN	Single neural network
MNN	Modular neural network

CHAPTER 1

INTRODUCTION

1.1 Rainfall-runoff (R-R) process modelling

Streamflow estimations are required over a wide range of discharge states, for example, for the design and operation of hydraulic structures, for real time management of the water resource systems, for the prediction of the effect of land-use and climate change, and as model inputs for other interacting process models like water quality models. The streamflow estimation models attempt to emulate the complex hydrological processes that transform rainfall into streamflow (runoff), with varying degrees of abstraction. Then, these rainfall-runoff (R-R) process models can be used to compute the streamflows, mainly at non-measurement stations and into the future. The decisions on planning and management of water resources are made based on the model forecasts and therefore depend on the accuracy and reliability of forecasts. Hydrological processes are nonlinear and complex processes. As a result, model approximations cannot reproduce the behaviour of those processes exactly. Error due to this process-model mismatch is known as bias error or model structure uncertainty. In addition to bias error, parameter errors and measurement errors collectively contribute for the uncertainties in hydrological predictions (Liu and Gupta, 2007). Model structure uncertainty is more likely to be dominant than other two types of errors and thereby identification and reduction are vital for operational modelling.

R-R process models are basically derived from the general principles of physical processes or measurement data itself. These modelling approaches are generally known as process-based models and data driven models (DDMs), respectively. The next two subsections will outline these approaches highlighting their merits and demerits.

1.1.1 Process-based models

Process-based models are derived from the descriptive equations of the hydrological processes. These equations that describe the temporal and spatial evolution of the sub-processes, are in general partial differential equations form that cannot be solved analytically. Therefore, solutions are found by finite difference representations, which involve form of discretization in space and time ordinates. This introduces errors which depends on the numerical method. Any model definition is an abstraction of knowledge what we have on hydrology. If some hydrological processes are not well understood those are represented by empirical generalizations. On the other hand, process-based models require large number of parameters that describe the physical characteristics of the catchment on a spatially distributed basis. Uncertainties in these parameters also contribute to the model error. Based on these, we can confirm that the incomplete understanding of the runoff generation processes and their representation lead to bias errors in process-based models. However, process-based models are distributed as equations involved space coordinates. Those are of great importance in understanding of the hydrological processes. Model simulations at short time steps are required to incorporate the nonlinearities and to maintain stable solutions. This makes computationally expensive model runs and limits their application in operational management systems.

1.1.2 Data driven models (DDMs)

In DDMs, like artificial neural networks (ANNs), regression equations, and genetic programming, a function is approximated using the system inputs and output without imposing a functional relationship. It is determined in the training process by optimising the number of possible functions.

Unlike process-based models, DDMs are computationally fast and therefore applicable for real-time applications (Proano et al., 1998). Those are widely applied to various hydrological problems (ASCE, 2000a, b; Babovic and Abbott, 1997a, b; Babovic and Keijzer, 2002; Babovic, 2005; Solomatine and Ostfeld, 2008). Most of these applications in R-R process modelling have been confined to identification of single input-output relation (Solomatine and Price, 2004) and therefore attempts should be made on improving the data driven representations to enhance their predictive capability. The primary focus of this research is given to reduction of model-attributed errors of DDMs.

The next section provides a brief review of the data driven streamflow estimation methods highlighting their limitations. A more detailed review is presented in Chapter 2. Finally, the objectives and the structure of the thesis are presented.

1.2 Problem statement

All models seek to simplify the complexity of the real world by presenting an approximated view of the reality; however, it should be complex enough to represent the system dynamics. More emphasis has been placed for identification of the major contributing processes to the runoff generation and their representation (Klemes, 1983; Sivapalan et. al., 2003), followed by progressive refinements.

Most primitive simplification made in R-R process modelling is lumping or spatial averaging. It is assumed that the variations in catchment properties and rainfall over the catchment are negligible. This type of conceptualization tends to be accurate, if the concentration time of the catchment is dominated by the hydrologic response time of the catchment, which holds for the small catchments (Anderson and Burt,

1985; Butts et. al., 2004). In such a situation, streamflow forecast can be based on catchment average rainfall and runoff data. Therefore, this approach is referred to as R-R modelling. It has been usual to approximate a function for streamflow estimation based on the antecedent rainfall and runoff values. However, hydrological rules are not similar for all runoff generating instances. Supervised classification of input-output data based on the magnitude of runoff as low, medium, and high runoff and approximating a function for each data cluster may not be applicable due to the presence of increases and decreases in flow. Instead, classification could be achieved with an unsupervised classifier. This is because the antecedent conditions are important in governing the subsequent processes. A few attempts have been made to classify the data, however, those studies failed to identify the different parts of the hydrograph effectively (Furundzic, 1998; Toth, 2009). Effective identification of the temporally dominant hydrological processes is one of the objectives in this research.

Research basis of small-scale catchments should be extended when it is applied for large-scale catchments. If the rainfall is not spatially uniform over the catchment, often in large catchments and in smaller catchments during intense convective storms, forecasts based on R-R models are inaccurate. For these applications streamflow forecasts can be based on the flow routing models as the total time of concentration is dominated by the flow travel time through the channel system (Anderson and Burt, 1985; Butts et. al., 2004). This is referred to as streamflow forecasting in the context of time series forecasting. Most of the data driven applications of streamflow forecasting are limited to point forecasts, where streamflow measurements at upstream gauging stations and/or at forecasting point are used to estimate streamflow at a downstream location (Khatibi et al., 2011; Kisi, 2008). Further refinement can be made by dividing the catchment into sub-catchments based on the spatially dominant processes. Studies

on this basis combined the sub-catchment runoff using a DDM (Chen and Adams, 2006; Corzo et al., 2009). A global model is not appropriate for flow routing, as it cannot capture local variations of flow. In addition, stage-discharge relationship is not similar for flow rising and flow recession. Several attempts have been made on cluster-based flow routing; however, those are limited to single stations (Abrahart and See, 2000; See and Openshaw, 1999; Wang et al., 2006). Therefore, there is a need to extend the cluster-based method for distributed flow routing.

From the above review, we can see that considerable errors in current data driven streamflow estimation procedures are model-attributed errors, which are due to the undefined process responses not included in the modelling procedure. Apart from the undefined processes, data resolution, both spatial and temporal, also introduces model error. Characteristic time and space scales of a process are threshold scales and these can only provide a partial picture of the process. To learn the process that occurs at characteristic space and time scales, data should be sampled at a fine resolution. This does not necessarily mean that data resolution can be chosen arbitrarily. This is because; fine sampled data can appear as a noise, deteriorating the models' predictability. Search for an optimal data resolution is difficult given that comparison has to be made at different time steps. This underlies the importance of interplay of data resolution and error accumulation of models, which has not been addressed so far.

1.3 Objectives of the study

Majority of data driven R-R process models are often insufficient to describe the inherently complex R-R processes. The overall objective of this research is to develop and evaluate techniques to improve the data driven estimation of catchment runoff. The specific objectives of the research are:

- (1) To review the data driven streamflow estimation applications to understand the reasons for the model-attributed estimation errors.
- (2) To investigate the effect of data time interval and model complexities on streamflow estimation and forecasting.
- (3) To classify temporally dominant runoff generating processes.
- (4) To develop and evaluate a modular data driven model for estimating streamflow of lump catchments.
- (5) To develop and evaluate a sequential flow routing method.
- (6) To investigate the applicability of cluster-based modelling for distributed flow routing.

This research is expected to accomplish the above listed objectives with following limitations. This study illustrates the application of the approaches using available rainfall and runoff data. It is also understood that several nonlinear data driven methods are available and the focus here is not to compare the accuracy of the methods available, but to improve the R-R process representation. Therefore, ANN is considered as the modelling method in this research.

1.4 Organization of the thesis

Chapter 2 introduces the subject of this research: stream flow estimation with DDMs. It provides a detailed review of the data driven flow estimation methods and addresses their issues that limit the accuracy of flow estimations. Based on the review, methodologies are outlined to represent the runoff generation processes in a better way for small to large-scale catchments.

Chapter 3 considers issues of R-R modelling based on DDMs. An example is chosen to illustrate the problems associated with data based R-R modelling. It

serves as a basis for highlighting particular constraints and implementation issues associated with R-R modelling.

Chapter 4 implements an input-output domain partition method using self-organizing maps (SOMs). Independent R-R relationships attached to each local region are approximated with ANNs and linear stochastic approach. Model results are compared to assess the improvement in nonlinear model approximations with input space decomposition.

Chapter 5 demonstrates the application of ANN in flow routing. A sequential flow routing method is then proposed and demonstrated. Applicability of cluster-based approach in distributed flow routing is also examined.

Chapter 6 presents a summary of the most important conclusions made in this thesis and gives a number of recommendations for further research.

CHAPTER 2

LITERATURE REVIEW

This chapter provides an overview of the developments in rainfall-runoff (R-R) process modelling with data driven techniques. More emphasis will be given to the methodologies that provide possible avenues for reducing the streamflow estimation errors.

The first section discusses the streamflow generating mechanisms together with some basic information on their process scales. The second section discusses relevance of model conceptualization approaches in process-based models to data driven models (DDMs). Then it reviews data driven applications in R-R process modelling and highlights their present limitations. Finally, artificial neural network (ANN), a machine learning technique used in this research is introduced with its implementation steps.

2.1 Runoff generating processes

Runoff integrates all hydrological processes upstream of the preferred point. The hydrological processes involved in the transfer of rainfall into runoff are shown in Figure 2.1. The water that eventually becomes streamflow comprises (1) baseflow (return flow from groundwater), (2) interflow (subsurface flow), (3) surface runoff or overland flow (Hortonian or infiltration-excess overland flow, saturated overland flow and throughflow), and (4) direct precipitation (Anderson and Burt, 1985; Maidment, 1993; Mays, 2005). These runoff generating mechanisms present arbitrary, spatially and temporally, depending on the significance of their major controls.

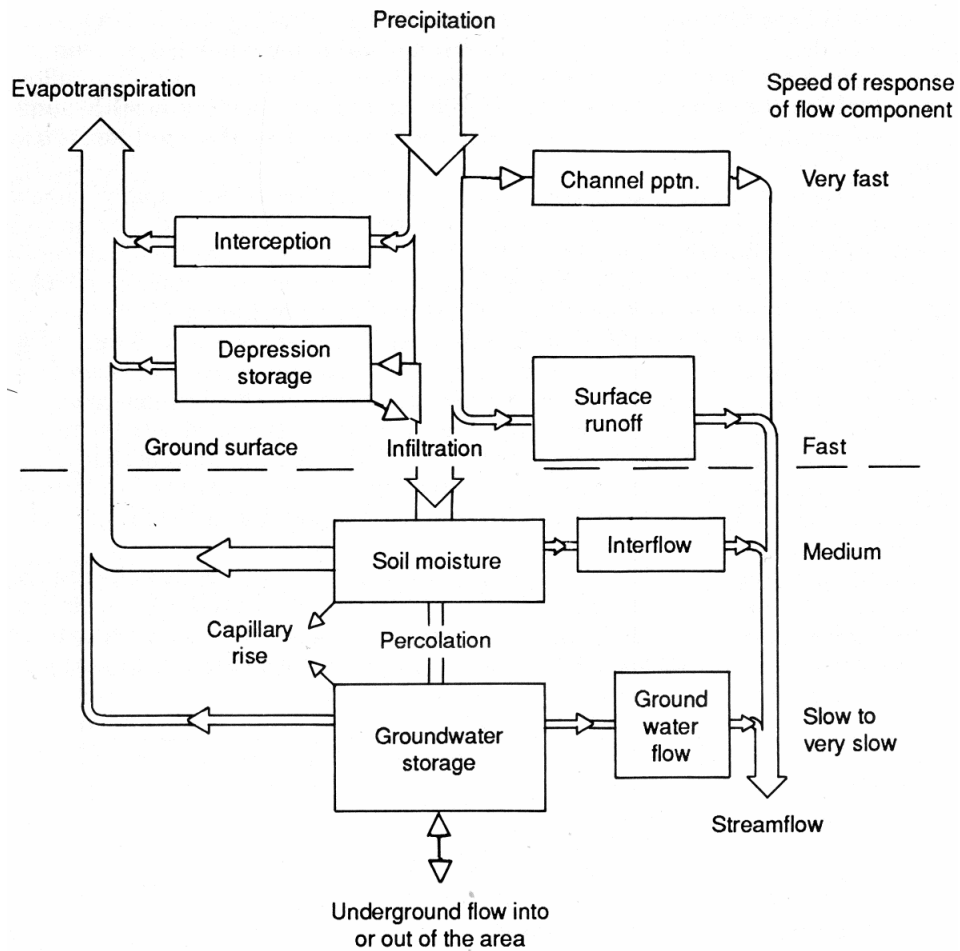


Figure 2.1: Runoff generating processes (Maidment, 1993).
 Note: width of the arrows indicates the average relative magnitudes of water transfer

2.1.1 Process scale

The process scale refers to the time (or length/area) required for a process to occur which is also referred to as characteristic time (space) scale. Characteristic time scale of a hydrological process is described using the process duration (for intermittent processes), the period or cycle (e.g., seasonal variation) and the correlation time (for a stochastic process). These are shown in Figure 2.2 a, b, and c, respectively. Similarly, characteristic space scales can be defined.

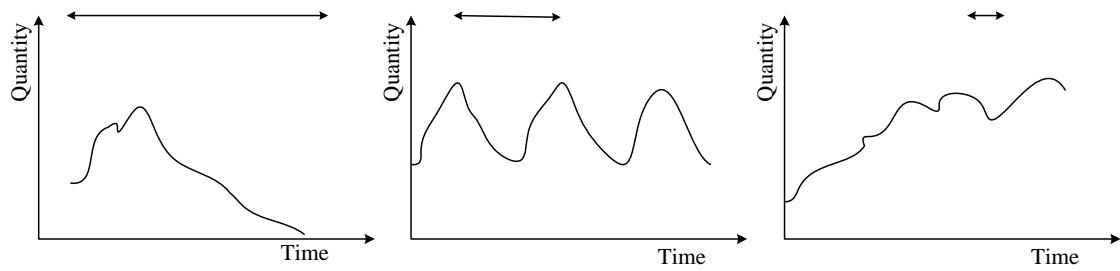


Figure 2.2: Process scale in time. (a) Duration (temporal extent of the process); (b) Temporal cycle; (c) Correlation time (Bloschl and Sivapalan, 1995).

2.1.2 Hydrological process scales

Dunne (1983) schematically represented the different environmental controls, i.e., climate, vegetation, land use, topography, and soils, on the runoff generation components (Figure 2.3).

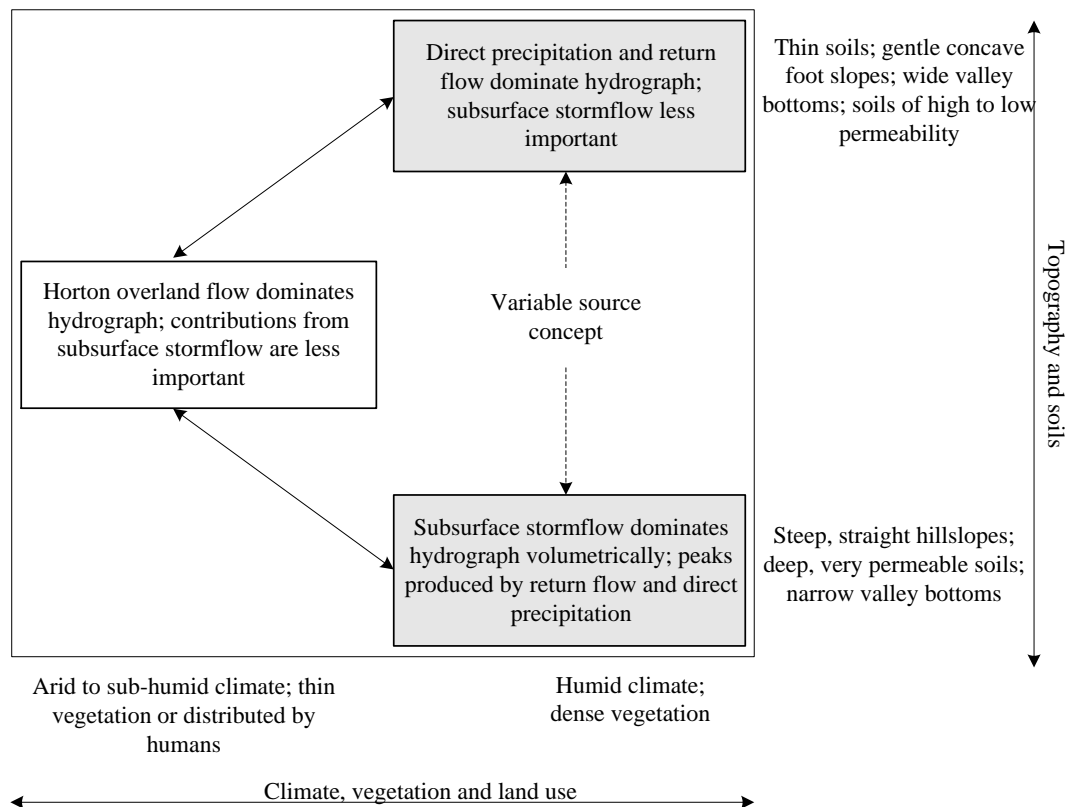


Figure 2.3: Major controls on runoff generation mechanisms (Dunne, 1983).

In addition, these sub-processes occur at different scales. Blosch and Sivapalan (1995) provided a more detailed classification of hydrological processes on possible spatial and temporal scales in their review paper on scale issues (Figure 2.4).

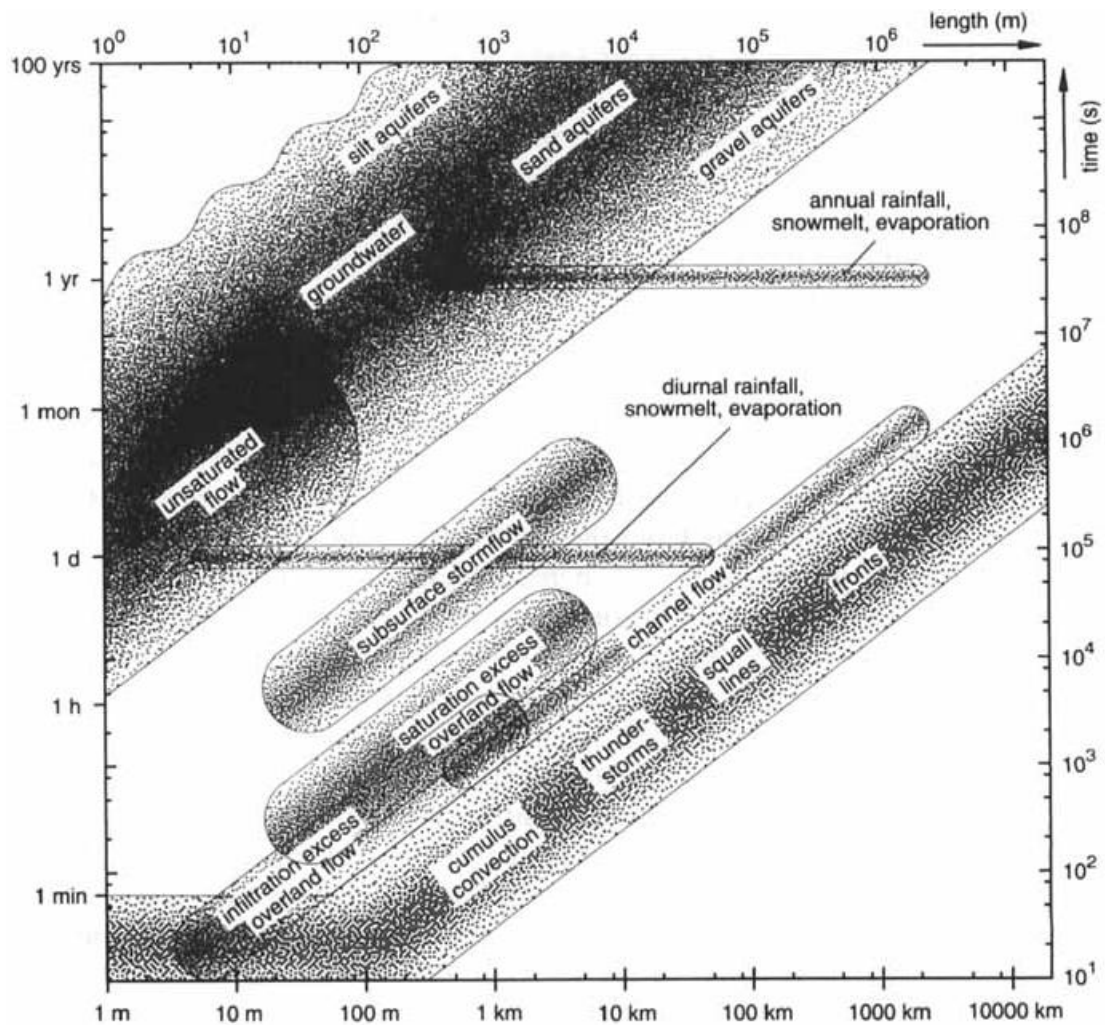


Figure 2.4: Characteristic space-time scales of hydrological processes (Bloschl and Sivapalan, 1995).

The rainfall mainly governs streamflow. The hydrological processes occur in response to rainfall and their time delays are clearly observable in Figure 2.4. For example, Hortonian overland flow adds to the streamflow quickly. It depends on the infiltration rate and the rainfall intensity, and can be defined at a small length scale. Saturation overland flow occurs subsequent to the Hortonian overland flow when soil is saturated. Subsurface and ground water flow components response slowly, which are operative over an area. We can also observe that the characteristic time scales of

sub-processes increase with the catchment scale. It indicates interplay of space and time scales, which needs to consider in model conceptualization.

2.1.3 Observation (Measurement) scale

The models are developed based on the observations made on the process variables. The observation scale is defined using the temporal extent of data set, the integration time of a sample, and the data time interval (Bloschl and Sivapalan, 1995). This is shown in Figure 2.5.

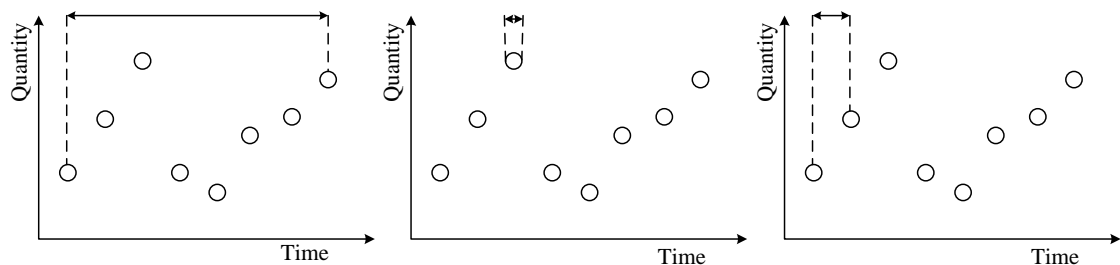


Figure 2.5: Observation scale in time. (a) Temporal extent; (b) Integration time; (c) Data time interval (Bloschl and Sivapalan, 1995).

Perfect match of the process scale and the observation scale is preferred to extract relevant information from data. If we observe a process at a larger scale, it can appear as a trend in data. On the other hand, a smaller scale can appear as a noise (Figure 2.6). The time and length scale that is considered in the modelling depends on the application. For real time control, we are interested in short-term forecasts. In that situation, event scales, which are typically order of days or less, are considered. Hydrological processes occur over a range of scales and whether to consider a combined scale or individual scales will depend on the model conceptualization.

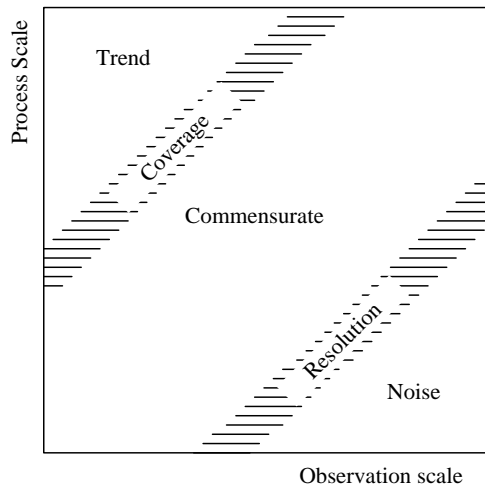


Figure 2.6: Dependency of observation scale and process scale (Bloschl and Sivapalan, 1995).

2.2 Rainfall-runoff (R-R) process conceptualization approaches

There are two ways to achieve a meaningful conceptualization, namely upward approach and downward approach (Klemes, 1983; Sivapalan et. al., 2003).

2.2.1 Upward approach

Upward approach is the conventional modelling approach in which the overall catchment response is estimated based on the knowledge on individual process components (Klemes, 1983; Sivapalan et. al., 2003). This is a theoretically perfect route, which advances our understanding of processes; however, for real time applications their usefulness will remain limited. Substantial amount of data needed for calibration and the excessive model complexity are other associated problems of the upward method. Unlike with process-based models, this type of formulation is unattainable with DDMs.

2.2.2 Downward approach

The model development from dominant processes to smaller scale processes is an alternative approach to upward approach. This is applied in a systematic way

starting from the first order controls of the overall catchment response and then further refinements are made in response to the deficiencies of the primary model. This is referred to as downward approach (Klemes, 1983). Simpler models that consider only the most important factors to the response are more appropriate for the management decisions.

Preliminary step of the downward approach will be to approximate a function based on past records of rainfall and runoff data. Transformation of rainfall into runoff is a result of many hydrological processes and it is shown that these occur at a wide range of spatial and temporal scales. The scales for the combined hydrological response are commonly determined using the time of concentration of the catchment and the spatial coverage of the rainfall. Catchment concentration time comprises the hydrologic and hydraulic response times. These are defined as the travel time of water from the most remote part of the catchment to the catchment outlet and flow travel time through the river system, respectively. Spatial scale is the ratio of the spatial coverage of the rainfall to the area of the catchment (Anderson and Burt, 1985). In small-scale catchments, generally less than 100 km², spatially uniform rainfall is assumed. In such situations, hydrologic response time of the catchment is significantly greater than the channel flow travel time. Then, forecasts are estimated based on the rainfall-runoff (R-R) models (Anderson and Burt, 1985; Butts et. al., 2004). However, in large catchments (spatial scale $< \sim 0.7$) flow travel time is much larger compared to the hydrologic response time. The streamflow forecasts are typically based on flow routing models in such situations (Anderson and Burt, 1985; Butts et. al., 2004). Further refinements can be made by dividing the catchment into sub-catchment areas.

In the present state, DDMs on R-R process consider how inputs and outputs are closely related without describing the internal processes and their interactions in a physical sense (Figure 2.7). This views the process externally and, thus the term ‘black-box’ is commonly used.

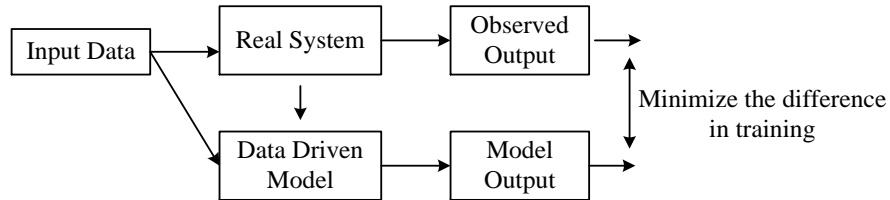


Figure 2.7: The representation of a process in data driven models (Solomatine and Ostfeld, 2008).

Through a better representation of the R-R process with further modifications models will improve the process approximation. This requires efforts to represent the basic processes in a way that can be applied in real time. The next two sections will discuss these possibilities according to research areas.

2.3 Rainfall-runoff (R-R) modelling with data driven techniques

In time series forecasting, historical observations of the same variable and forcing terms are considered to develop a model, which describes the underlying relationship. Then the developed model is used to compute the future time series values. The R-R model approximation can be presented as;

$$Q_{(t+1)} = f(Q_{(t)}, Q_{(t-1)}, \dots, Q_{(t-m)}, R_{(t)}, R_{(t-1)}, \dots, R_{(t-n)}) \quad (2.1)$$

Where, Q and R represent the discharge and rainfall values; m and n represent number of time lagged components of Q and R , respectively. The above function can be approximated with any DDM like ANNs, regression equations, and genetic programming (ASCE 2000a, b; Babovic and Keijzer, 2002; Liang et al., 2002; Solomatine and Ostfeld, 2008; Yu et al., 2004). Most of these applications in R-R

modelling have been confined to identification of single input-output relationship (Solomatine and Price, 2004). This type of model can be viewed as a global model that represents the whole domain. However, a global model might be adequate for approximating a distinct relationship for the entire input-output domain, which is not acceptable for the R-R process.

Due to inability of the exact model representation for the nonlinear complex R-R process, there is no single best model and only possibility is to have most likely outcomes. For this reason, many versions of independent model outputs can be combined together to reduce the approximation error. Example combination methods are simple averaging, weighted averaging, nonlinear combination, Bayesian model averaging, and generalized likelihood uncertainty estimation (Acar and Rais-Rohani, 2009; Baker and Ellison, 2008; Diks and Vrugt, 2010; Hashim, 1997; Kim et al., 2006). It was shown in literature that combined model performance is superior to that of single best model performance (Liu and Gupta, 2007; Sharkey, 1999). This type of model combinations falls into the static structure category of the committee machines (Haykin, 1999; Solomatine and Price, 2004). However, member models of ensemble model are global models that represent entire modelling domain and are incapable of capturing local variations of flow.

It is identified with the principle of divide and conquer, that a complex task can be solved by partitioning it into number of simpler tasks whose solutions then can be combined to obtain an overall solution to the complex problem (Haykin, 1999). The overall model comprising the simpler local models is referred to as a modular model in the literature (Jacobs and Jordan, 1993). Modular models have some advantages over global models, like simplicity and computational efficiency. Identification of the

simpler tasks or functionally different sub-processes is the main challenge in the application of this principle to physical processes. For example, in case of R-R process, interactions of sub-processes makes it difficult to identify the simpler tasks based on input-output data relations and thereby to separate corresponding inputs and outputs in a supervised manner. Depending on the feature of nonlinearity, usually a process could be divided, for example using thresholds, into a number of regimes and a model can be fitted to each regime (Sivapragasam and Liong, 2005; Zhang and Govindaraju, 2000; Solomatine et al., 2007). For example, Zhang and Govindaraju (2000) considered that hydrologic rules for generating runoff are different for low, medium, and high streamflows. They employed three different trained networks to represent each runoff subclass. Their results showed improvement over single global model. Modular models can be predictive than the global model. The question is whether we get improvement in forecasts for right reasons. In threshold-based approach, a local model learns rules for generating both increase in and decrease in flows, which is not justifiable. R-R models assume the lumped catchment concept; therefore, attempts should be made on identifying the temporal variation of dominant processes.

Runoff processes occur at different times during the progress of a rainfall event (Figures 2.8 and 2.9). As a result, depending on the main process that governs the runoff generation, the functional relationship is more likely to be different at different parts of the hydrograph.

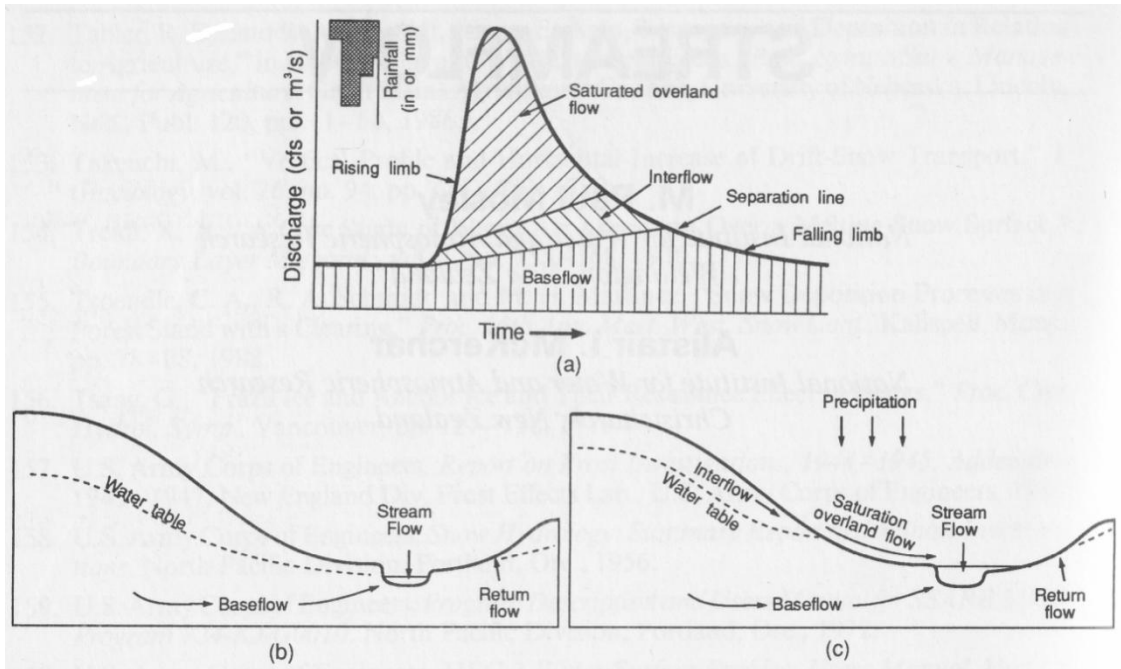


Figure 2.8: (a) Separation of sources of streamflow on an idealized hydrograph, (b) Sources of streamflow during a dry period, and (c) during a rainfall event. (Maidment, 1993).

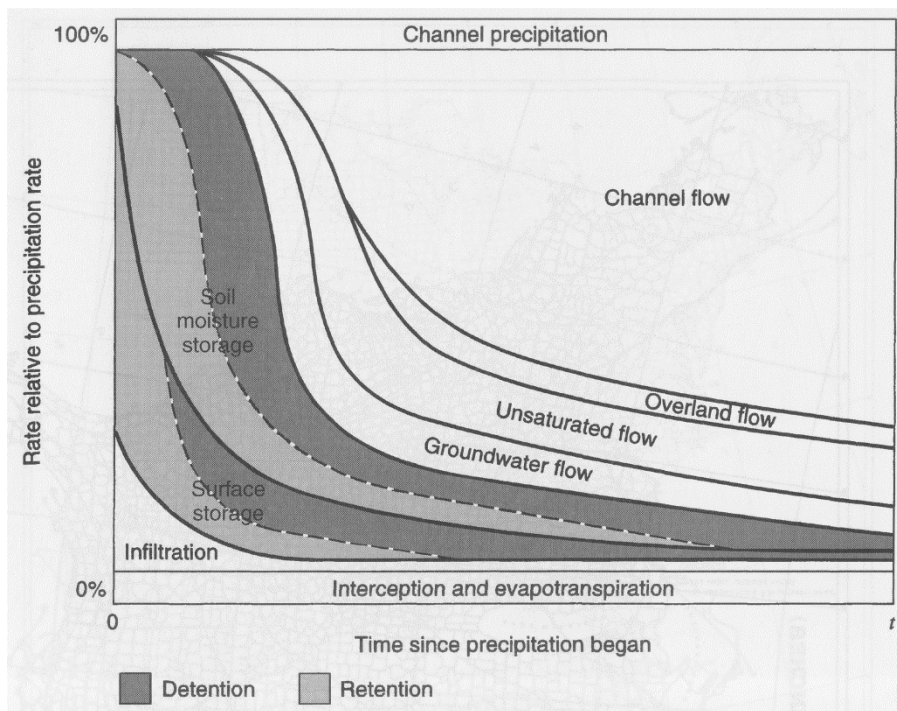


Figure 2.9: Relative importance of the sub-processes at different times (Mays, 2005).

Corzo and Solomatine (2007) applied the constant slope method (McCuen, 1998) and the filtering algorithm of Eckhardt (2005) to separate the baseflow and direct runoff (excess flow). Separate models were trained to learn the direct runoff and

the base flow relationships. They used the soft combination method to compute the final model output. The main drawback of this method is the use of constant weighting coefficients. Instead, time varying weights are more appropriate since the contribution of base flow and direct runoff varies from time to time. Successively, few studies considered unsupervised classifiers to partition the input space (Furundzic, 1998; Toth, 2009). Their idea was innovative for two reasons; (1) the antecedent conditions govern the catchment response, (2) possible partitions are not known for a particular catchment. In the hydrological context, the input pattern consists of rainfall depths and the output discharges at the catchment outlet. However, use of rainfall and runoff (cumulative) input patterns in domain classification seems to restrict the identification of rising limb and falling limb of a hydrograph. This can be a result of presenting the input pattern in a form that the classifier unable to identify. It is also known that the functional relationships are more likely to be different for decrease in and increase in flows. This is with the understanding that increases in flow are governed by the magnitude of rainfall. Conversely, previous discharge values or change in discharge values significantly affect the flow recession. Therefore, identification of rising limb and falling limb of a hydrograph may have significant effect on bias error. As a result, efforts should be made first to identify the change in discharge.

2.4 Streamflow forecasting with data driven techniques

Muskingum method is the conventional flow routing approach, which relates the inflow and outflow discharges of a river reach and water stored within it by the continuity equation and by an empirical storage equation (O'Donnel, 1985).

$$I - Q = \frac{dS}{dt} \tag{2.2}$$

$$S = K[xI_t + (1-x)Q] \quad (2.3)$$

Equations (2.2) and (2.3) can be expressed in finite difference form for an interval of time, ΔT , which results;

$$Q_{t+1} = C_1 I_t + C_2 I_{t+1} + C_3 Q_t \quad (2.4)$$

$$C_1 = \frac{\Delta T + 2Kx}{\Delta T + 2K(1-x)}; C_2 = \frac{\Delta T - 2Kx}{\Delta T + 2K(1-x)}; C_3 = \frac{-\Delta T + 2K(1-x)}{\Delta T + 2K(1-x)}$$

Where; $C_1 + C_2 + C_3 = 1$; I represents the inflow; Q stands for the outflow; S is the storage; K symbolizes flow travel time of the reach; and x is the weighting factor specifying relative importance of both the inflow to and the outflow from the reach in determining the storage. The two parameters, K and x are calculated by a trial-and-error graphical technique (Singh and McCann, 1980). If there are $(n+1)$ number of data, above equation can be applied simultaneously, which is represented in the matrix form;

$$|Q_{j+1}| = C_1 |I_j| + C_2 |I_{j+1}| + C_3 |Q_j| \quad ; j = 1, 2, \dots, n \quad (2.5)$$

This equation resembles to the linear ARX (Auto-Regressive with eXogenous) type of model with constraint coefficients (Masters, 1995). This method considers one time-lagged component of the inflow and outflow. However, if the data time interval (ΔT) is less than the flow travel time of the reach, the conventional approach will not extract the relevant information. Generally, ΔT should be less than the flow travel time in order to capture the essential dynamics of the process. The Muskingum method also assumes a linear relationship, which is not acceptable for nonlinear processes. Without imposing a relationship, it can be learned from the data itself using the machine

learning techniques, which are able to learn linear as well as nonlinear functions. If the flow travel time of the river reach is $n+1$, the formulation given in Equation (2.5) can be modified as;

$$Q_{(t+1)} = f(I_{(t+1)}, I_{(t)}, \dots, I_{(t-n)}, Q_{(t)}, Q_{(t-1)}, \dots, Q_{(t-n)}) \quad (2.6)$$

Data driven applications on flow routing can be grouped into two categories as; (i) distributed and lumped flow routing, and (ii) global and cluster-based flow routing. This differentiation is based on whether spatial and temporal variability of the process is considered in the modelling or not.

2.4.1 Distributed and lumped flow routing

Most of the data driven applications on flow routing have been confined to a single river reach, where discharge at a downstream location is estimated using the discharge data of an upstream location and streamflow data of the same location (Khatibi et al., 2011; Parasuraman and Elshorbagy, 2007; Wu et al., 2005). In this situation, predictability of the model deteriorates significantly when the forecasting horizon increases the flow travel time of the river reach. If the upstream location is distant from the downstream location, it will not provide useful information. This is because; there is an upstream characteristic length (similar to the temporal dependency) that affects the variations of the flow at a downstream location. Some other studies used only the auto-regressive streamflow data (Abrahart and See, 2000; Kisi, 2008; Wang et al., 2006). This will be the only possibility if the upstream data are not available.

The predictive capability of DDMs will be greatly enhanced if they are developed to learn the intra-catchment variation of the processes. For this purpose, the basin can be partitioned into sub-basins. Several spatial discretization methods are available in the literature. Some of early spatial discretization methods were based on stream order (Horton, 1945; Strahler, 1957), contours generated from digital elevation maps, and isochrones. These methods did not consider the spatial variability of the characteristics that govern the runoff generation. To overcome this limitation, researches attempted to develop indices for hydrological similarity (Wagener et al., 2007). Kirkby (1975) introduced the topographic index, which is the ratio of the upslope contributing area and the local surface topographic slope. Some other researchers used climatic classification schemes using the precipitation, potential evaporation, and the runoff variables. The Budyko curve is an example of climatic classification scheme, which represents wet, medium, and dry areas of the United States (Budyko, 1974). Some of other catchment discretization methods represented land-use heterogeneity. The existing spatial discretization methods can be integrated in a way to identify the distribution of the dominant runoff processes within a catchment. The next step will be to estimate the upstream channel inflows, i.e., small scale sub-catchment outflows, using the R-R models described in the section 2.3.

Few studies considered data at few upstream locations; however, a single model is not effective in identifying local variations of flow (Diamantopoulou et al., 2006; Liong et al., 2000; Liong and Sivapragasam, 2002). Chen and Adams (2006) applied semi-distributed form of conceptual models in estimating sub-catchment runoff and the estimated flows were used as ANN model inputs to predict the total runoff. In their study, entire catchment (8506 km²) was divided into three sub-catchments based on the river network characteristics. Corzo et al. (2009) followed a

similar approach except that the few sub-catchment models were replaced by DDMs. In these applications sub-catchment model outflows were nonlinearly combined to produce the catchment outflow. This type of global model will identify the most influential sub-catchment (s). More recently, Nourani and Kalantari (2010) proposed an integrated modelling approach for forecasting daily suspended sediment discharge at several locations. The inputs of the ANN model were the antecedent rainfall and runoff values of six gauging stations. The number of output neurons has been set to six. That was to provide the suspended sediment forecasts at the gauging stations. This type of model formulation has several drawbacks. First, the number of hidden neurons is determined based on the overall forecasting capability of the model. However, complexity of the process will differ from one location to another location. For this reason, a single integrated model will provide general solutions. Second, inclusion of inputs at all stations may provide superfluous information. Thus, potentially more reliable method will be the sequential application of the flow routing in which the outflow from one sub-reach becomes the inflow to the next sub-reach. Specifically, this flow routing method provides forecasts at number of locations.

2.4.2 Global and cluster-based flow routing

If we approximate a function for the wave propagation from one point to another, it follows that similar rules exist for increases or decreases in flow. In so doing, we assume a unique stage-discharge relationship for flow rising and flow recession. However, it is a loop-shaped curve during the passing of a wave as shown in Figure 2.10 (Wu et al., 2011). In addition, functionally different regions may exist like baseflow. For this reason, clustering of functionally similar input-output data and function approximation to those local regions may improve the forecasts.

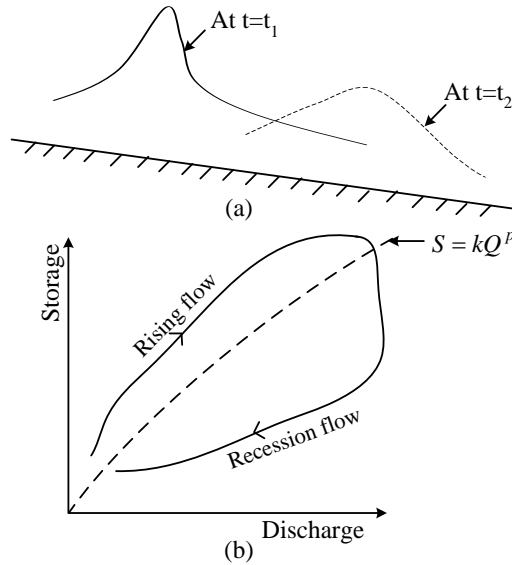


Figure 2.10: (a) Propagation of a flood wave, (b) Storage-discharge relationship.

Threshold-based models, which are based on the magnitude of the streamflows, are not logically correct, however, may provide improved forecasts due to the fact that they are trained on part of the data set. Instead, supervised classification of data can be applied to classify the input space. Parasuraman et al. (2006) integrated self-organizing maps (SOMs) and modular neural networks, and named the integrated model as spiking modular neural networks (SMNNs). They applied SMNN for monthly streamflow forecasting at Siox Lookout of English river, Canada using the upstream flow data at Umfreville. Similarly, Parasuraman and Elshorbagy (2007) applied k-means algorithm to cluster the streamflow data. In this approach, monthly streamflow data of the Little river were used to predict the flows at Reed Creek. However, this research considers short term forecasting.

Wang et al. (2006) developed cluster-based ANN model to forecast daily discharges at Tangnaihui, Yellow river, China. They classified the model input data into three clusters based on Fuzzy C-means clustering technique and found that those represent low flow, medium flow, and high flow. A possible reason for this may be the

use of absolute discharge (Q) data. Abrahart and See (2000) considered three single stations, one in Upper river Wye, Central Wales and two stations in river Ouse, Yorkshire. Classifier input variables of each station consisted of two seasonal factors, six antecedent Q data, six antecedent differenced discharge (dQ) data, and either Q or dQ value at time t . They found that use of all input variables classified data according to season, which might be a result of using seasonal factors in classifier inputs. They obtained reasonable differentiation with 64 SOM clusters using six antecedent Q values. In another study, See and Openshaw (1999) used hourly sampled water level data of Skelton and five other stations in the river Ouse, Yorkshire to forecast the water level at Skelton. Firstly, they classified the combined preceding water levels of six stations using SOMs. Initially, sixteen clusters were identified as suitable in identifying different events and those were manually classified into five main clusters: falling, rising, peaks, low-level flat, and medium level, based on their similarities. Secondly, fuzzy logic model was developed to identify the five clusters based on their inputs. Finally, specialized models were developed for each cluster. Application results were shown to improve the forecasts with cluster-based approach.

In summary, the studies on cluster-based flow routing are limited to single stations. Cluster-based flow routing models have been shown to improve the streamflow estimation and it is thus attempted to extend the cluster-based approach for streamflow estimation at multiple stations.

The next two sections will discuss effect of data resolution on R-R process approximation and factors affecting the accuracy of multi-step-ahead forecasts which are generally applicable to both R-R modeling and streamflow forecasting.

2.5 Effect of data resolution on rainfall-runoff (R-R) process approximation

As outlined in section 2.1, processes need threshold scales to occur. However, perfect selection of scales does not necessarily imply accurate representation of R-R process. This is because DDMs learn the R-R process dynamics based on past records of rainfall and runoff data. Reduction or magnification of data resolution have an effect on predictive capability of models. Characteristic time scale and space scale provide an upper bound to the data resolution. Data driven model applications on R-R modelling have been more commonly carried out using the existing sampled data. Model formulation at given data resolution may not be applicable. As temporal (spatial) variations are characteristic features of the process, an approach to improve the prediction accuracy will be enlarging the observation sample. However, the range of data resolution is not continuous.

Some attempts have been made to improve the forecasting capability by removing the noise in data (Elshorbagy et al., 2002; Jayawardena and Gurung, 2000; Karunasinghe and Liong, 2006; Porporato and Ridolfi, 1997; Sivakumar et al., 1999). The effectiveness of this approach is questionable in two aspects. Firstly, this is a subjective approach since the true signal is unknown. Secondly, the effect of noise depends on the data time interval (ΔT). Decrease in ΔT will improve the extraction of relevant information from data, while it also increases the possibility of capturing noise in data. As a result, unless the ΔT is too fine noise, removal will not improve the forecasts. It is also to be noted that training forces the network response to be smoother rather than fitting exactly to the training data.

Improvement in predictability with decrease in data sampling gap also reflects that models learn the nonlinear process dynamics. For example, some of the studies

suggested that ANNs perform well compared to linear models (Hsu et al., 1995; Sajikumar and Thandaveswara, 1999; Thirumalaiah and Deo, 2000), while few other studies reported that performance of ANN and linear models are comparable (Elshorbagy et al., 2000; Han et al., 2007). These studies considered one ΔT . Shamseldin (1997) applied linear models and nonlinear models to six catchments and analysis of his results showed that ANNs performed well for some catchments, while it was a linear model for some other catchments. In some instances, performances of both models were comparable. It is also to be noted that complexity and nonlinearity in the R-R process differ from one catchment to another. Nonlinear and linear models formulated for a process, which exhibits highly nonlinear dynamics, can perform comparably, if sparse data are considered in the model development. These considerations imply that nonlinear DDMs perform as good as or better than linear models depending on the degree of complexity of the process.

In some other studies, real world systems are assumed as rarely linear or nonlinear and proposed two-step hybrid procedure; firstly to capture the linear effects with a linear model and secondly to approximate a nonlinear relationship with the residuals of the linear model (Jain and Kumar, 2007; Khashei and Bijari, 2011; Sallehuddin and Shamsuddin, 2009; Díaz-Robles et al., 2008; Zhang, 2003). This hybrid method was inspired by the little difference in predictability of linear and nonlinear models observed by some of the researchers (Elshorbagy et al., 2000; Gaume and Gosset, 2003; Han et al., 2007; Shamseldin 1997). Another reason may be the inadequate representation of the process to learn the nonlinear variations of the process. We can argue that it is inappropriate to use a linear model to approximate a nonlinear process. The above hybrid approach also can be viewed as a type of error correction method. The error correction models were applied in number of studies

(Babovic et al., 2001; Lekkas et al., 2001; Shamseldin and O'Connor, 2001; Solomatine et al., 2007).

Moreover, nonlinear models generally outperform linear models in approximating nonlinear processes, if those are strictly unique relationships and not a result of several sub-processes. R-R process is a result of several sub-processes and this might be a reason for satisfactory results with global linear model approximations. These suggest that more efforts have to be made to fully utilize the nonlinear models' predictive skill. This research focuses on the model-attributed errors due to improper representation of the R-R process.

2.6 Accuracy of multi-step-ahead forecasts

The iterative method and the direct method are the two ways of computing multi-step-ahead forecasts. The iterative approach iteratively uses immediate preceding data including the forecasted values, while the other method employs only past rainfall and runoff data. Theoretically, former method is more appropriate as state at any time depends on the immediate preceding values and therefore improved predictions are expected with iterative forecasting. Several researchers have applied iterative forecasting procedure (Van den Boogard et al., 1998; Khondker et. al., 1998; Babovic, 1998; Daimantopoulou et al., 2006). Study carried out by Khondker et. al., (1998) compared direct forecasting with iterative forecasting. However, their results showed no improvements to the forecasting accuracy.

The forecasting accuracy deteriorates with the forecasting horizon. Even a small runoff estimation error at the beginning can accumulate deteriorating the quality of forecasts. This effect can be significant for complex and nonlinear systems which

are poorly understood. Accurate representation of the R-R process will reduce the error accumulation caused by the previous forecasts. In addition, response of the model to errors may depend on the complexity of the function, which has not received the attention of researchers.

2.7 Artificial neural networks (ANN)

ANNs are designed to model the way in which the brain performs a particular task or function of interest (McCulloch and Pitts, 1943; Rosenblatt, 1958; Haykin, 1999). ANNs ability to learn a nonlinear complex relationships and their capability to produce reasonable outputs for unseen data make it as a sophisticated tool to solve classification as well as regression problems. There are numerous time series forecasting applications of ANN in the field of water resources management. Several ANN based models have been proposed to forecast runoff including Multi-layer perceptron (MLP), Support vector machines, Generalized regression neural networks, and Radial basis functions. From all the available neural network types, MLP has been most widely used in the water resources field (Minns and Hall, 1996; Van den Boogaard et al., 1998; Thirumalaiah and Deo, 2000 ASCE 2000a,b) and MLP with a single hidden layer have the ability to approximate any bounded continuous function (Universal approximation theorem).

MLP is characterized by its architecture and the direction of information flow. It can be classified by the number of layers as single layer, bilayer, three layer, and multilayer. In Figure 2.11, schematic diagram of the three-layered network is shown. Typically, nodes are arranged in layers. As such, ANN has an input layer, from which input vector is fed to the network, output layer and one or more intermediate layers (hidden layers) comprised with computational nodes. In each layer (except in

output layer), inputs are weighted by corresponding connection weights and sum up together which is then transformed by an activation function. ANN is trained adjusting the parameters to bring the output of a network to the desired output.

Another way is to classify by the direction of information flow as feed-forward neural networks (FFNN) or recurrent neural networks (RNN). In FFNN, information flow from input layer to the output layer without feeding back to the precedent layers, whereas, in RNN direction of flow can be in both directions. Several feedback architectures considered in the literature. However, in the context of the water resources, FFNNs are more widely used.

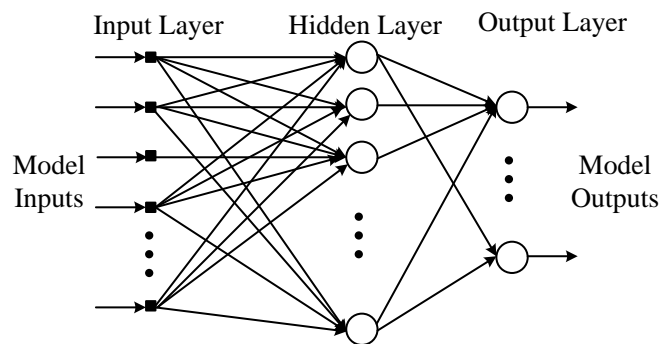


Figure 2.11: Three-layered multi-layer perceptron (MLP).

Several steps should be considered in implementing the ANN. These are discussed in the following subsections.

2.7.1 Input determination

The first step is to determine the appropriate inputs. Good physical understanding of the process being modelled can help in selecting the input vector. Selection of appropriate inputs is primarily based on the system knowledge (ASCE, 2000a). Then analytical techniques like correlation analysis, average mutual

information can be used to find the number of lags for each input variable (ASCE, 2000a; Bowden et al., 2005; Maier and Dandy, 1997; Masters, 1995).

The significance of input variables to output variables may differ to each other. However, ANNs treat all the input variables equally. Therefore, it is important to normalize the selected input variables such that they have similar ranges. That is to bring all the variables into similar ranges. There are several approaches. One approach is to standardize the data using the mean and standard deviation of the training set. Another approach is to normalize the input variables to the range of either to $[0, 1]$ or to $[-1, 1]$. However, normalizing the inputs to the range of $[0, 1]$ is not efficient for updating the weights. That is because updates of the weights will have the same algebraic sign resulting decrease or increase in weights. In general, any shift of the average input away from the zero will bias the updates in a particular direction and thus slow down the training. This strategy is much helpful in choosing the activation function for the hidden layers. As output of the hidden layers are inputs to the next layer, choosing a activation function that gives normalized output will automatically provide normalized inputs. With relevant to the above discussion, hyperbolic tangent function is preferable from the available continuous activation functions. In some situations, the dimension of the input vector is large, but the components of the vectors are highly correlated. It is useful in this situation to reduce the dimension of the input vectors. An effective procedure for performing this operation is principal component analysis (Hu et al., 2007).

The data set, which is used to build the neural network model, is partitioned into three categories as training data, cross-validation data, and testing data. The training data set is used to find the optimal weights and bias values. The data allocated for

training should be sufficient to learn the underlying relationship between inputs and outputs (ASCE, 2000a). In other words, training data should be representative.

2.7.2 Training neural nets

The main objective of training is to reduce the process approximation error by adjusting the model parameters. With the breakthrough finding of the back-propagation algorithm (Gradient descent method) by Rumelhart in 1986, it has been the most commonly used method for training the multi-layer FFNNs in many fields. This standard back propagation algorithm updates the network weights and biases in the direction in which the negative gradient of the performance function decreases most rapidly. Summary of this algorithm is given in Haykin (1999). A Momentum constant (forgetting factor) was introduced to this method to avoid instability. These methods are often too slow for practical problems. As a result, several high performance algorithms have been developed such as conjugate gradient methods, Levenberg-Marquardt algorithm. All those are upgrade to the standard back propagation algorithm to provide faster convergence. In many cases, Levenberg-Marquardt algorithm is able to obtain lower mean square errors. In addition, learning rate also makes an impact on learning speed. Small learning rate is desirable to avoid instability. However, it imposes a slow learning.

After appropriate training, ANN is able to generate satisfactory results within few seconds. The generalization capability of ANNs depends on the strategies used in the training procedure.

2.7.3 Extrapolation capability

ANN is one of the machine learning tools that has been successfully applied in time series forecasting providing potentially better results (ASCE, 2000a, b). However, it is a well-known fact that ANN is not a good extrapolator (ASCE, 2000b). The extreme events may be encountered in real world systems and forecasts provided by ANN models are not reliable in such situations. Few attempts have been made to improve the extrapolation capability of ANNs. As highlighted by Karunanithi et al. (1994) use of linear transfer function in the outer layer helps to improve the extrapolation ability, however, bounds of the hidden neuron transfer function (sigmoid function or hyperbolic function) undermine extrapolation level. Minns and Hall (1996) suggested scaling the input data to 0.2 to 0.8 rather than to -1 to 1. In a later study by Varoonchotikul (2003), modification to standardization function was proposed in which maximum value of the raw data was multiplied by a factor, greater than one, to provide a room for larger values. However, this method might distort the relationship of input and output data as increase in all parameters is not expected in the same order of magnitude. Hettiarachchi et al. (2003) applied another approach in which the estimated maximum flood in the river basin was computed to train the ANN model. However, this approach has limited application, as it required long period of record data to estimate the maximum value. In addition, there is an uncertainty in the estimation. Besides the above approaches, model complexity reduction might add value to the extrapolation ability.

2.7.4 Optimal model complexity

A simple model would not be able to capture the process behaviour. On the other hand, a model should not be too complex. This is because fitting a function that

passes through all the training data points, or smaller bias, does not always guarantee that it learns the underlying relationship. This causes large errors for new data sets, which is referred to as overfitting. A network that is just large enough to provide an adequate fit provides better results. In this respect, it is necessary to determine the optimal model complexity. This can be better explained with bias-variance trade-off (Breiman, 1998; Nelles, 2001). The cost function, which used to define the model error, can be decomposed into bias and variance as;

$$\text{Model error} = \text{Variance} + \text{Bias}^2 \quad (2.7)$$

Variance and bias refer to the variance of the model estimate and deviation of the mean model estimate from the desired response, respectively. Bias describes the systematic deviation of the process and the model that exists due to the model structure. In other words, it represents the structural instability of the model. Models are not exact representations of the physical processes. As a result, individual forecasting models may be subject to deviation from the exact. A nonlinear process usually cannot be modelled without a bias error due to process complexity. Generally, bias error approaches to zero with increasing the model complexity as shown in Figure 2.12. On the other hand, error due to the deviation of the estimated parameters from their optimal values is known as variance error. Model parameters are found using finite and noisy data set. In reality, it is not possible to have a representative data set and it is just a realization. As a result, it is expected to deviate from their optimal values. Increase in model complexity allows the model to fit training data perfectly and it precisely represents the noise contained in training data. This results poor generalization and variance error increases with model complexity as in Figure 2.12.

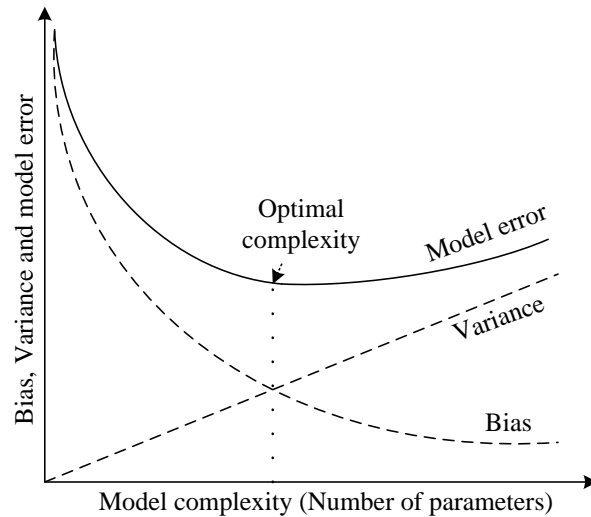


Figure 2.12: Illustration of the bias/variance trade-off (Nelles, 2001).

Our goal in modelling is to get close to the optimal model complexity or in other words to have a model with low bias and low variance. Generally, growing and pruning techniques are used to determine the number of hidden nodes (ASCE, 2000 a). An alternative yet effective approach has been proposed based on the bias variance trade-off. In this method, part of the training data set is used for estimating the model parameters. The training error does not contain the variance part of the error decomposition. As a result, error on the training data decreases with the model complexity. The rest of the training data, data set with different noise realization, is used to detect the variance error. The optimal model complexity is the one that gives minimum error on that testing data (Figure 2.13).

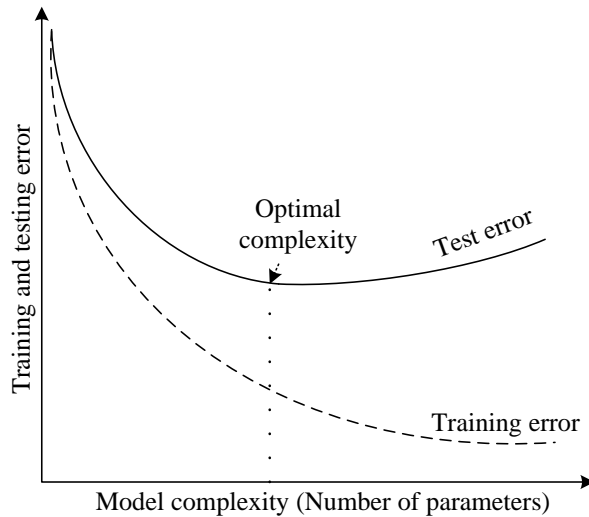


Figure 2.13: Training and testing error variation with the model complexity.

The other effective techniques include cross-validation and regularization. Cross-validation prevents overfitting during the training. In the beginning of the training, errors of the both training and the cross-validation data sets decrease. After parameters reached to the optimal values, training data set error continues to decrease, while the cross-validation data set error starts rising. This gives an indication that overfitting is occurred. Cross-validation stops training once it starts to over train. On the other hand, regularization is a smoothing approach, which can be explained, based on the study of Xiang et al. (2005). According to their geometrical interpretation of MLP network, the approximated function is a superposition of piecewise linear functions with a bias. Its geometrical shape is similar to the piecewise linear activation function (Figure 2.14).

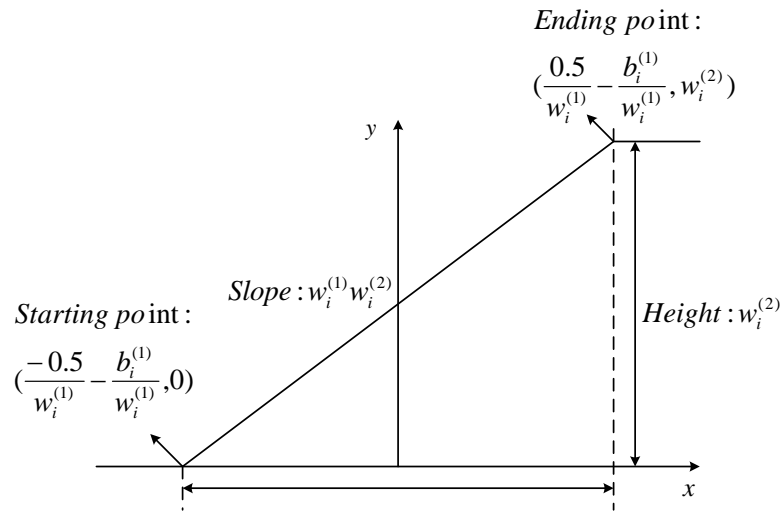


Figure 2.14: Basic building block of MLP (Xiang et al., 2005).

It is shown that the number of hidden neurons corresponds to the number of piecewise linear functions. The slope of the basic building block depends on the product of weights connecting the input neuron to the hidden neurons and the weights connecting the hidden neurons to the output neurons. Minimization of the slope will reduce the overfitting. The cost function can be modified by adding cost on weights.

2.8 Summary

This chapter reviewed the data driven applications in R-R process modelling of small-scale to large-scale catchments. It appears that the considerable errors in current DDM applications are due to the insufficient representation of the runoff generating processes. Global representation of temporally and spatially dominant processes is identified as the major problem preventing the improvement in runoff estimation. Data resolution and model complexity are other basic factors that have an effect on streamflow estimation and multi-step-ahead forecasting. This research will address the number of approaches to reduce these model-attributed errors.

CHAPTER 3

EFFECT OF DATA TIME INTERVAL ON RAINFALL-RUNOFF (R-R) MODELLING

3.1 Introduction

Data driven models (DDMs) are widely recognized as an important tool for decision support systems. Nonlinear time series techniques are therefore widely applied for rainfall-runoff (R-R) modelling. Data driven models are primarily based on observations. Therefore, time series data should be sufficiently refined to capture the essential dynamics of the process. This will provide accurate forecasts at one-step lead-time. Besides, in practice, we would prefer accurate forecasts in the longer forecast lead-time. Accuracy of multi-step-ahead forecasts, i.e., forecasts several time steps into the future, mainly depends on the models' predictability in one-step-ahead forecasts and on their error accumulation properties. This chapter examines the effect of data time interval (ΔT) on forecasting accuracy. This study also discusses the importance of rainfall and corresponding change in runoff as model inputs compared to commonly applied rainfall and runoff inputs. All the methods and procedures are tested with the artificial neural network (ANN) models.

3.2 Case study

Hourly sampled rainfall and runoff data of the Orgeval catchment, France (Figure 3.1) were considered in this study. The Orgeval is a secondary tributary of the Marne river. It has a drainage area of 104 km². The basin is relatively flat and there is a sharp drop near to the river mouth. It is located entirely in rural areas where agriculture takes place on 80% of area and remain is forested (shaded areas in the Figure 3.1). The annual average rainfall is 706 mm.

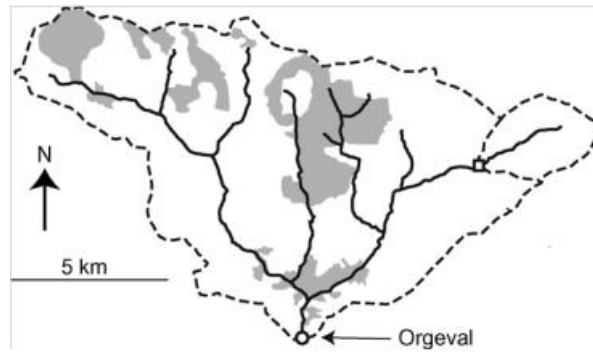


Figure 3.1: The Orgeval catchment (Anctil et al., 2009).

Three years of hourly rainfall and runoff data were used in this study: 80% for training and 20% for testing the models. Two analyses were performed with absolute discharge (Q) and differenced discharge (dQ) values. The statistical measures, mean, standard deviation, minimum, and maximum of the discharge time series are $0.7 \text{ m}^3/\text{s}$, $1.5 \text{ m}^3/\text{s}$, 0.03 m^3 , and $28.8 \text{ m}^3/\text{s}$, respectively. Testing data were within the range, i.e. in between the minimum and maximum values of the training data. This is to avoid any misinterpretation with under-predictability of ANN models for out-of-range data.

3.3 Input determination

The preferred approach for determining appropriate inputs and time lags of inputs involves a combination of prior knowledge and analytical approaches. In case of R-R process, dynamics vary within the catchment concentration time and necessary hydrologic information can be extracted from the data if the data time interval is less than the catchment concentration time. Correlation analysis is the most commonly applied analytical technique for selecting the appropriate inputs. Figures 3.2 and 3.3 show the correlation coefficient variation with the time lag for Q and dQ data, respectively.

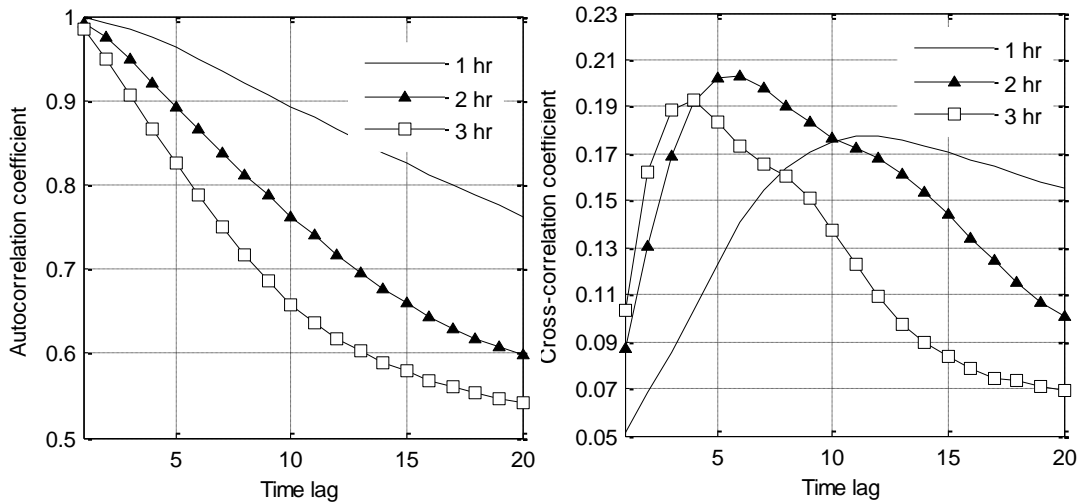


Figure 3.2: Autocorrelation coefficient variation of absolute discharge (Q) data, and cross-correlation coefficient variation of absolute discharge (Q) and rainfall data for 1hr, 2hr, and 3hr sampled data.

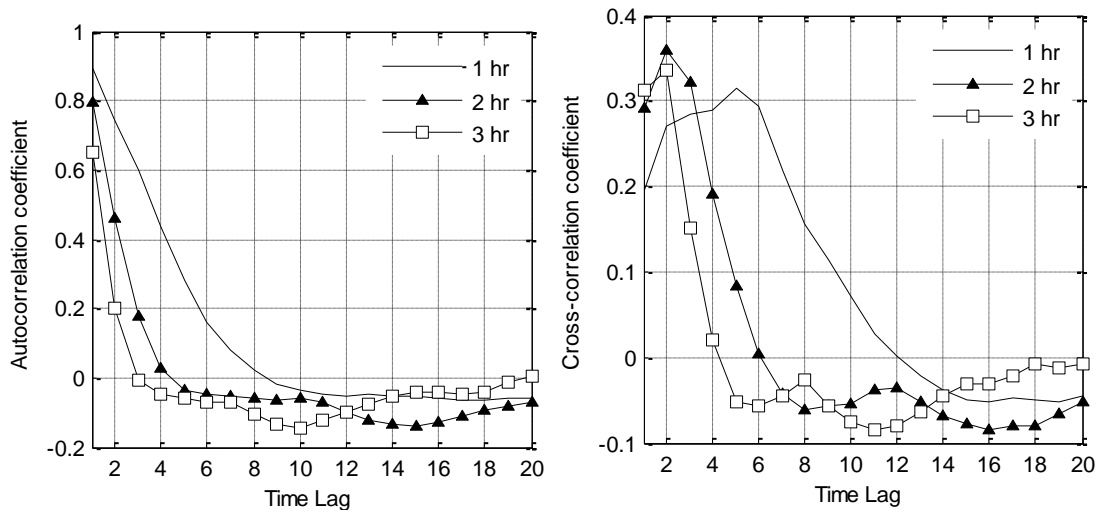


Figure 3.3: Autocorrelation coefficient variation of differenced discharge (dQ) data, and cross-correlation coefficient variation of differenced discharge (dQ) and rainfall data for 1hr, 2hr, and 3hr sampled data.

Magnitude of correlation coefficient determines the strength of the linear relationship. Cross-correlation function gives its maximum when peak rainfall coincides with peak absolute discharge (Figure 3.2) or peak positive change in discharge (Figure 3.3). Correlation analysis showed that the concentration time of the catchment is around 6 hours. Thus, in addition to 1 hr sampled data, 2 hr, and 3 hr sampled data were considered for the analysis. Correlation analysis indicated that 6, 3,

and 2 time lagged components of rainfall and runoff would be sufficient for 1 hr, 2 hr, and 3 hr sampled data, respectively.

As can be observed from Figure 3.3, autocorrelation coefficient of dQ data drops to zero after a few numbers of time lags (< 9); however, it is still greater than 0.65 for Q data (Figure 3.2). In addition, immediate cross-correlation coefficients are slightly increased with adjacent differencing. These suggest that the linear dependencies were significantly reduced with adjacent differencing.

3.4 Forecasting models

R-R relationship was approximated with ANNs. Three-layered multilayer perceptron (MLP) network, the most commonly applied ANN architecture in function approximation (ASCE 2000a, b), was used to approximate the R-R relationship (Equation 3.1).

$$Q_{(t+1)} = f\left(R_{(t+1)}, R_{(t)}, \dots, R_{(t-n)}, Q_{(t)}, Q_{(t-1)}, \dots, Q_{(t-n)}\right) \quad (3.1)$$

Where Q and R represent discharge and rainfall values and n represents number of time-lagged components.

The activation function of the hidden neurons was hyperbolic tangent function. The number of hidden layer neurons was determined for different ΔT s, and for Q and dQ data. In addition, cross-validation was used to prevent the over-fitting problem. The iterative approach was utilized to compute the forecasts at different forecasting horizons. Forecasting horizon was 12 hours.

3.5 Performances of rainfall-runoff (R-R) models

Mean absolute error (MAE) and correlation coefficient (r) were used in this study to evaluate the model performance.

$$MAE = \frac{\sum_{i=1}^n AE_i}{n} \quad \text{with} \quad AE_i = |(Q_{o,i} - Q_{p,i})| \quad (3.2)$$

$$r = \frac{\sum_{i=1}^n (Q_{o,i} - \overline{Q_o})(Q_{p,i} - \overline{Q_p})}{\sqrt{\sum_{i=1}^n (Q_{o,i} - \overline{Q_o})^2 \sum_{i=1}^n (Q_{p,i} - \overline{Q_p})^2}} \quad (3.3)$$

Where Q_o is the observed discharge, Q_p is the predicted discharge, $\overline{Q_o}$ is the mean observed discharge, $\overline{Q_p}$ is the mean predicted discharge, and n is the number of data samples. The notation Q-ANN stands for ANN model trained with Q and rainfall data. Symbol Q is replaced with dQ if the model was trained dQ data.

Figure 3.4a, b, and c show the forecasting performances of Q-ANN and dQ-ANN models for 1 hr, 2 hr, and 3 hr sampled data, respectively. Dashed line represents the model accuracy without the accumulated error.

Comparison of one-step-ahead forecasts shows that dQ-ANN model performs slightly better for hourly data, while it is Q-ANN model for 2 hr and 3 hr sampled data. However, MAEs of dQ-ANN models are much lower than the corresponding MAEs of Q-ANN models at extended forecasting horizons. This is because of the simplicity of the function approximated with the dQ data. This means that models trained with dQ data can issue more reliable forecasts.

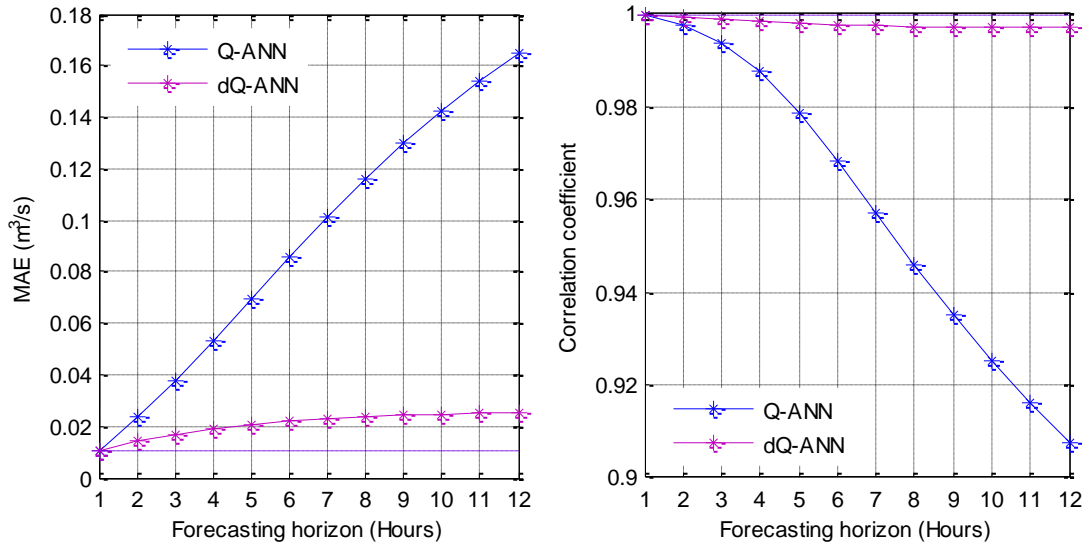


Figure 3.4a: Performances of ANN models for hourly data.

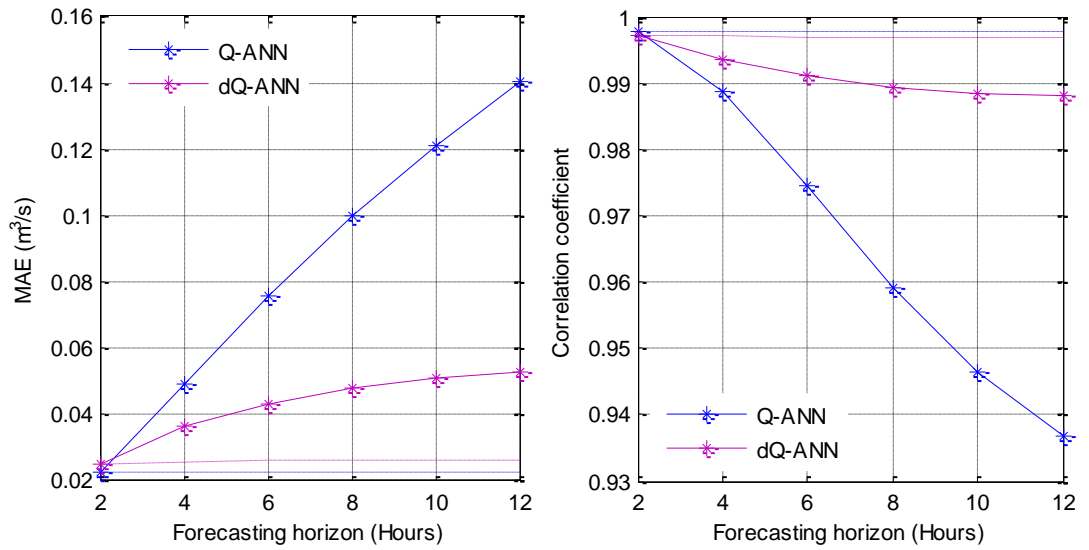


Figure 3.4b: Performances of ANN models for 2 hr sampled data.

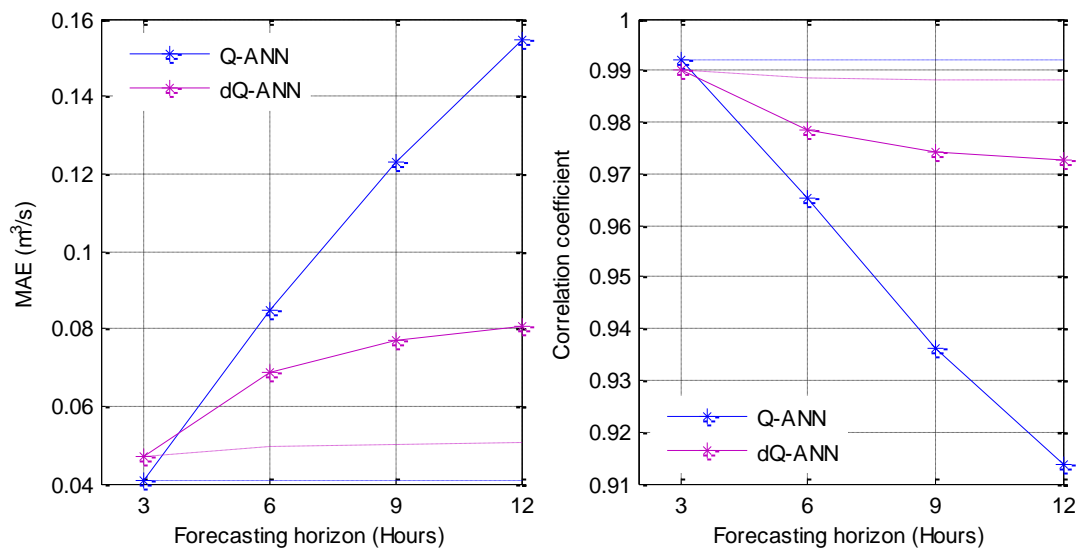


Figure 3.4c: Performances of ANN models for 3 hr sampled data.

The approximated models are global representations of the R-R process. Complexity of the R-R process is not similar over the model domain. Thus, global model errors are expected to be large with the process complexity. This is clear from Figure 3.5, which represents the scaled errors for a particular section of the discharge time series. For this reason, same data samples were considered in order to compare the forecasting performance for 1 hr, 2 hr, and 3 hr sampled data.

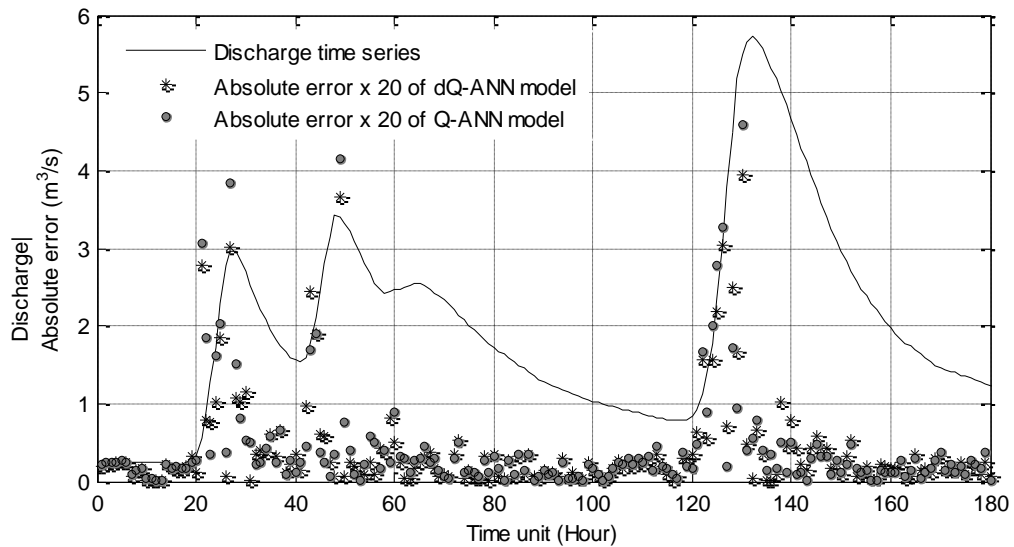


Figure 3.5: Absolute error (scaled) produced by Q-ANN and dQ-ANN models.

3.5.1 Effect of data time interval on forecasting accuracy

Q-ANN and dQ-ANN model performances for the same data samples are presented in Table 3.1 and 3.2, respectively. Comparison of Q-ANN model results of 1 hr and 2 hr sampled data shows that 2 hr sampled data improve the predictions slightly and this is more prominent at extended forecasting horizons (Table 3.1). In case of 1 hr and 3 hr sampled data, 3 hr sampled data perform slightly better at 3 hr ahead forecasts (Table 3.2). On the contrary, *dQ* data produce improved predictions at fine sampled data over the prediction horizon. This inconsistency could be due to two possible reasons.

(1) Error accumulation properties of the models

The R-R process approximation error accumulates during the iterative steps and it can be quite significant when the number of steps is large. For example, 6 hr ahead forecast is composed of 1×6 hr, 2×3 hr or 3×2 hr forecast. Similarly, 12 hr ahead forecast can be based on 1×12 hr, 2×6 hr or 3×4 hr forecast. As can be seen in Figure 3.4, Q-ANN model error increases at a rate greater than the dQ-ANN model in subsequent iterative steps. This is observable even if the Q-ANN model performs well in one-step-ahead forecasts. This indicates that the Q-ANN model's sensitivity to model approximation error is higher compared to the dQ-ANN model and it affects the prediction accuracy at extended forecasting horizons.

(2) Linear dependencies and noise in time series data

Small ΔT might capture random effects, including the noise effect. Thus, it affects overall prediction accuracy. This approximation error is accumulated during the iterative steps. In addition, we expect that the linear dependencies of the autoregressive components would dominate the nonlinear variations, which affect the effective extraction of the information relevant to nonlinear dynamics. Adjacent differencing reduces the linear dependencies and noise in data, and this might be a possible reason for improved predictions at fine sampled data. It is known that ANN can learn linear as well as nonlinear relationships.

If we compare the Q-ANN and dQ-ANN models R-R process approximation errors, it can be observed from the Figure 3.4 and Table 3.1 that the Q-ANN model performs well for 3 hr and 2 hr sample data, while it is comparable for 1 hr sampled data. Improvement in predictability with differenced data increases with decrease in

ΔT . Moreover, rainfall measured over a period of time results an increase in runoff. As a result, the functional relationship is more likely to exist between rainfall and dQ data.

Table 3.1: Q-ANN model performance with data time interval.

Mean absolute error (MAE) $\times 10^{-3}$ (m ³ /s)								
ΔT (Hours)	Forecasting horizon (Hours)							
	2	3	4	6	8	9	10	12
1	23		55	95	140		186	232
2	19		42	67	91		112	130
1		37		97		164		231
3		32		69		103		129
1				91				226
2				70				129
3				73				132

Correlation coefficient (r)								
ΔT (Hours)	Forecasting horizon (Hours)							
	2	3	4	6	8	9	10	12
1	1.00		0.99	0.97	0.95		0.92	0.91
2	1.00		0.99	0.98	0.96		0.95	0.94
1		0.99		0.97		0.93		0.91
3		0.99		0.97		0.94		0.92
1				0.98				0.91
2				0.98				0.95
3				0.96				0.92

Table 3.2: dQ-ANN model performance with data time interval.

Mean absolute error (MAE) $\times 10^{-3}$ (m ³ /s)								
ΔT (Hours)	Forecasting horizon (Hours)							
	2	3	4	6	8	9	10	12
1	12		17	20	21		22	22
2	21		34	40	42		43	45
1		15		20		22		23
3		38		53		59		63
1				20				22
2				40				45
3				53				64

Correlation coefficient (r)								
ΔT (Hours)	Forecasting horizon (Hours)							
	2	3	4	6	8	9	10	12
1	1.00		1.00	1.00	1.00		1.00	1.00
2	1.00		0.99	0.99	0.99		0.99	0.99
1		1.00		1.00		1.00		1.00
3		0.99		0.98		0.97		0.97
1				1.00				1.00
2				0.99				0.99
3				0.97				0.97

The results showed that the decrease in ΔT improves the R-R process approximation. However, this property is beneficial if models generate improved forecasts at extended forecasting horizon, which is not true for Q-ANN models. Remesan et al. (2010) also studied the effect of ΔT on forecasting accuracy using 15 min, 30 min, 60 min, and 120 min sampled rainfall and Q data of Brue catchment, England. Forecasting lead times were 2 hr, 4 hr, and 6 hr. Four time-delayed components of rainfall and runoff were considered for the 15 min data sampling rate. This suggests that the concentration time of the catchment is around 1 hr. Their results showed that the 30 min sampled data provided the lowest error.

The R-R process is a result of several sub-processes with dynamics varying over a range of temporal scales. For this reason, decrease in ΔT less than the concentration time of the catchment will improve the learning of the process dynamics (Figure 3.6). However, optimum time scale, which captures essential dynamics of the process, is not known. Further discretization of time series into finer steps is not advantageous and models trained with such data are more susceptible to capture the noise in data. From this point of view, we can conclude that the hourly rainfall and runoff data are not too fine to capture the noise. It might be possible to improve the model predictability with dQ data, if more refined data are available. Prior information on future flows at small intervals is very useful to operational forecasting systems.

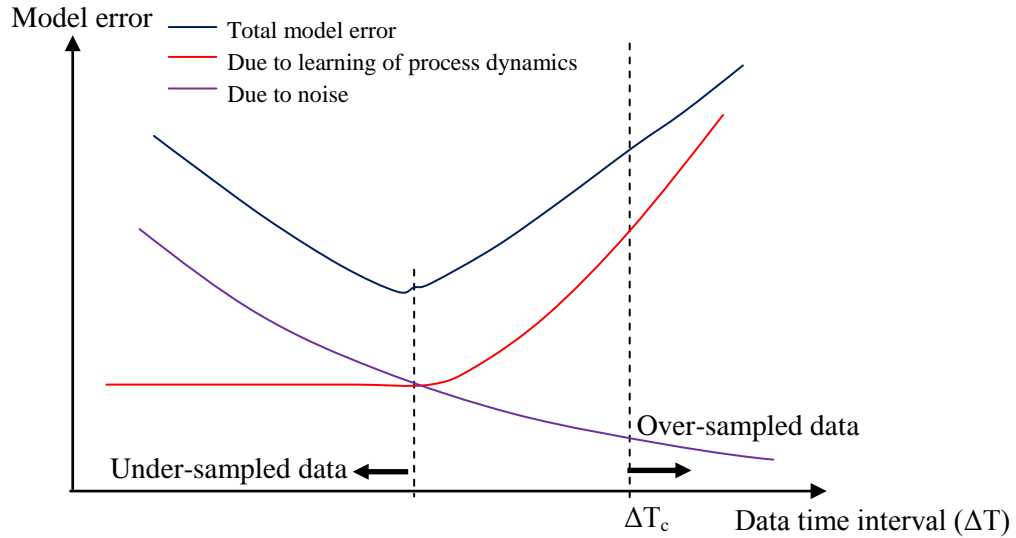


Figure 3.6: Effect of data time interval (ΔT) on model error.
 ΔT_c : Catchment concentration

The above forecasting approach iteratively used the previous forecasted values in successive time steps. This can be considered as more suitable as the future state depends on the immediate preceding values. Direct forecasts, which use only past information to forecast multi-step-ahead forecasts, might be predictive, if the error accumulation in iterative forecasting is significant. This depends on the predictive capability of the models and their sensitivity to the errors. The next sub-section will discuss the iterative and direct forecasting performance for hourly sampled data.

3.5.2 Iterative and direct forecasting

Figures 3.7 and 3.8 show the performances of iterative forecasts (IF) and direct forecasts (DF) of Q-ANN and dQ-ANN models, respectively. Direct forecasting performance is slightly better in short term predictions, especially at forecasting horizons less than the catchment concentration time. As forecasting horizon increases information is not given by the immediate precedence values as those are left one by one. In simply, this is because initial conditions are washed out after 6 hrs. As a result, direct forecasting accuracy deteriorates. This is clearly observable in Figures 3.7 and

3.8. Moreover, one trained network can be used for all time steps with iterative forecasting, while individual trained networks are required in direct forecasting. The results also show that direct forecasting does not reduce the forecasting error significantly in Q-ANN models and the dQ-ANN models perform well over the forecasting horizon compared to Q-ANN models.

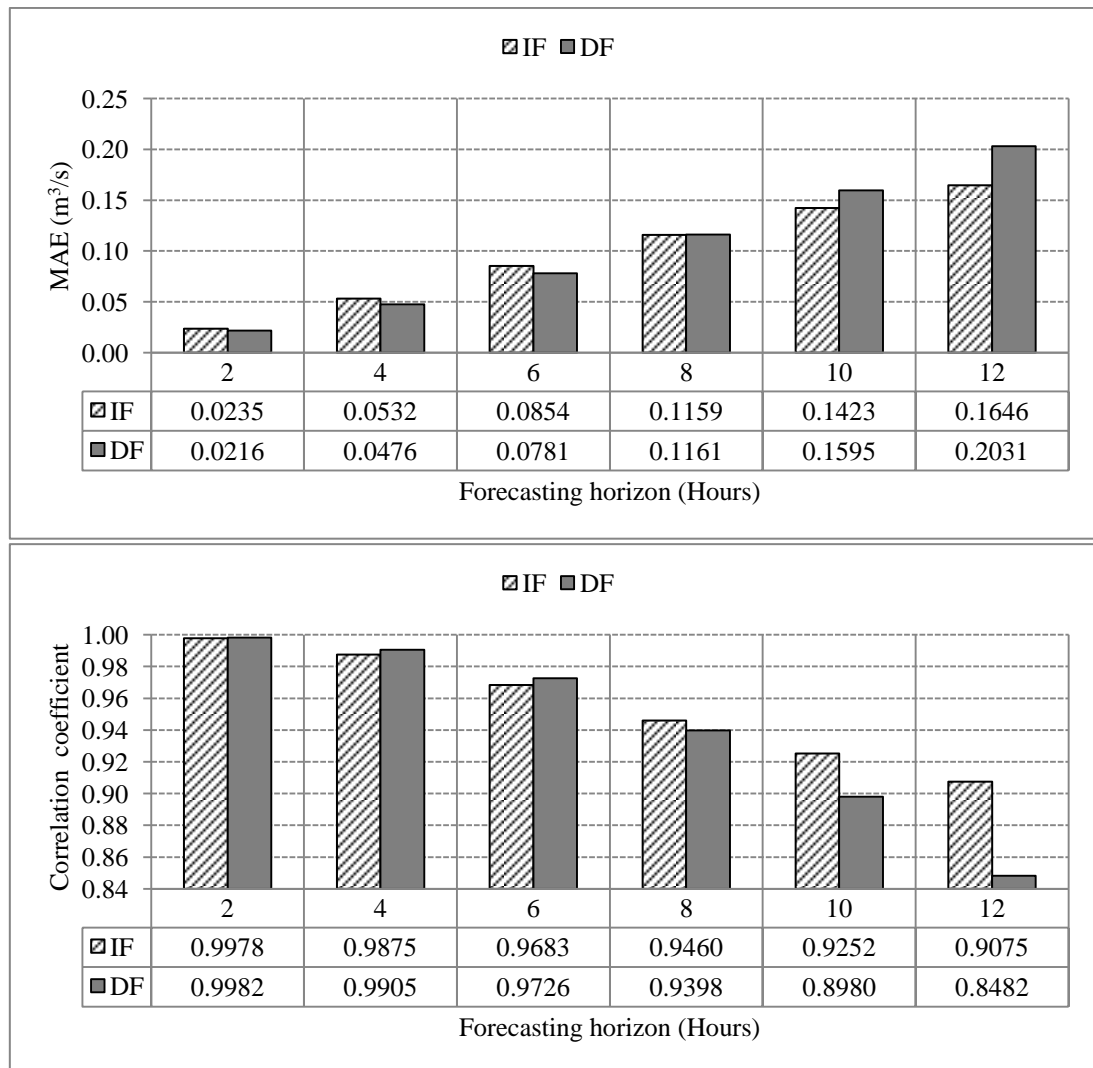


Figure 3.7: Iterative and direct forecasting performances of Q-ANN models.

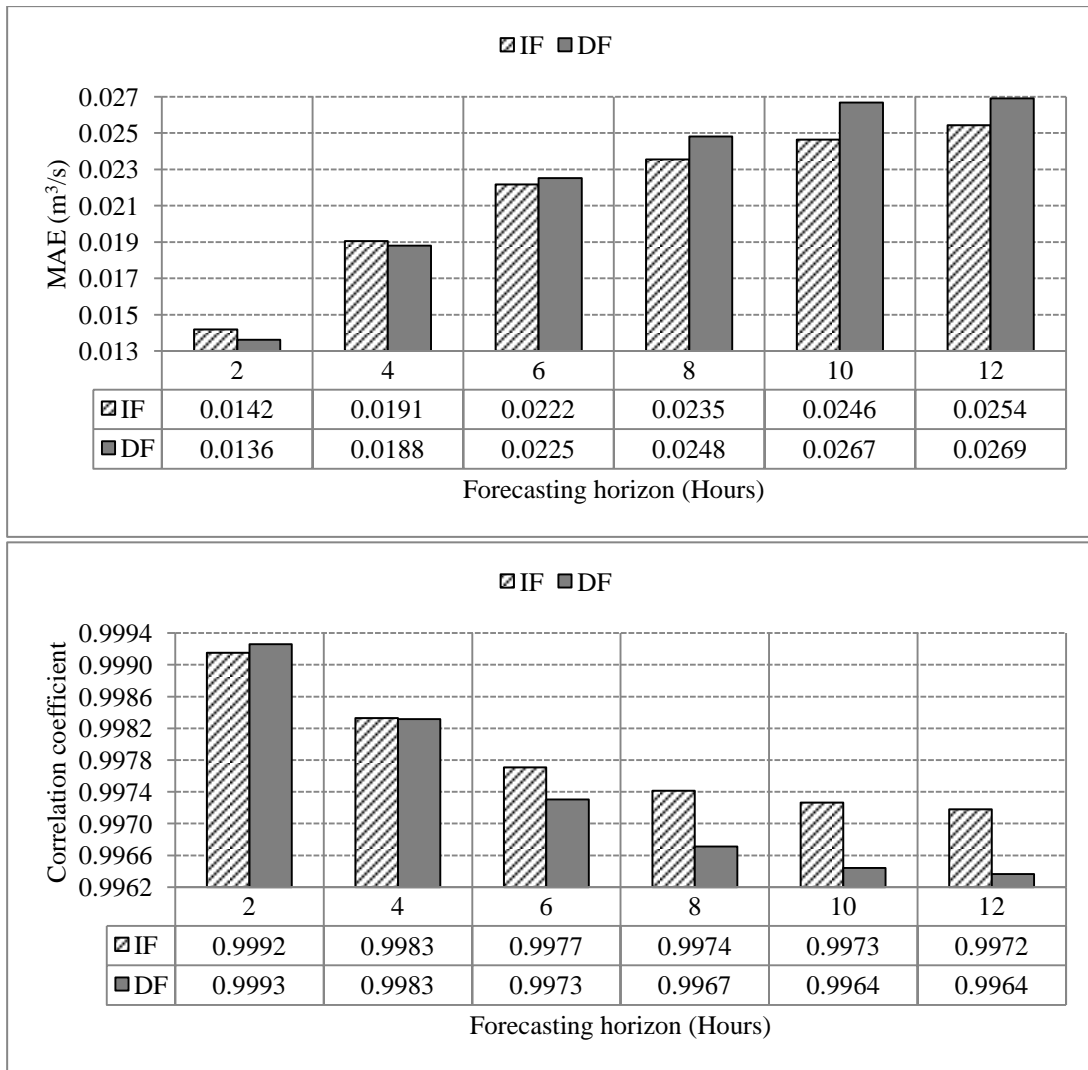


Figure 3.8: Iterative and direct forecasting performances of dQ-ANN models.

3.6 Conclusions

This chapter examined the effect of data time interval on forecasting accuracy. Both R-R model approximation error and the sensitivity of a model to estimation errors were identified as factors affecting the accuracy of long-lead forecasts. Adjacent differencing provided improved predictions at extended forecasting horizons compared to Q data. This suggests that models trained with dQ data tend to generate more reliable forecasts. This study could not evaluate the effect of noise in data due to the limited data. However, the effects of data time interval and differencing on predictive

capability of R-R model were successfully established offering a basis for further evaluations.

Identification of change in functional relationship for different magnitudes of runoff will further improve the R-R process approximation. The next chapter will look into the possibility of applying modular model approach for R-R process approximation.

CHAPTER 4

MODULAR DATA DRIVEN APPROACH FOR RAINFALL-RUNOFF (R-R) MODELLING

4.1 Introduction

Data driven models (DDMs) are widely used to approximate the rainfall-runoff (R-R) relationship (ASCE 2000). Most of these studies considered single input-output relationship. However, the functional relationship is not similar for all runoff generation instances over the modelling domain (Solomatine and Price, 2004; Zhang and Govindaraju, 2000). Incorporation of R-R process knowledge into the modelling process may improve the model accuracy. This chapter presents an input-output domain partition method using self-organizing maps (SOMs).

The first step involves the search of modularity-associated features of hydro-meteorological input data. In the second step, functional relationship of each local region is approximated with artificial neural networks (ANNs). In this way, for a particular forecasting instance, classifier determines the local domain and ANN model assigned for that local domain provides the forecast. In this study, results of single neural nets (SNN) and the modular models (MM) are compared to assess the improvement in nonlinear model approximations with input space decomposition. Further, classifying input variables into number of hydrological regimes and fitting a function for each regime might improve the ability of identifying nonlinearity. Performances of ANN and linear model representations are therefore compared. Finally, extrapolation capabilities of all models are discussed.

4.2 Case study

This study also used the hourly sampled rainfall and runoff data of the Orgeval catchment. The R-R model approximation can be presented as;

$$Q_{(t+1)} = f\left(R_{(t+1)}, R_{(t)}, \dots, R_{(t-n)}, Q_{(t)}, Q_{(t-1)}, \dots, Q_{(t-n)}\right) \quad (4.1)$$

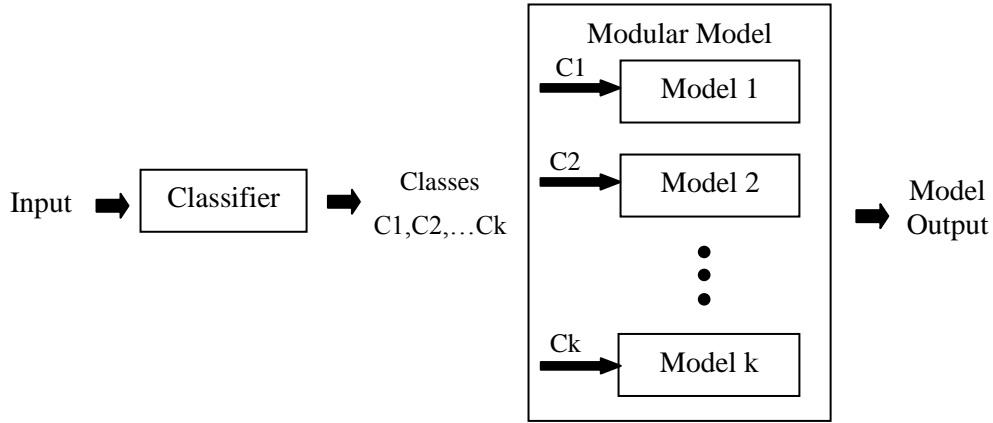
Where Q and R represent the discharge and rainfall, n represents the time-lagged components of discharge and rainfall. Six lagged components were considered based on the correlation analysis. Similar to the study presented in Chapter 3, this study also considered two analyses with absolute discharge (Q) data and differenced discharge (dQ) data. Next section briefly discusses the SOM classification approach used in this study.

4.3 Identification of hydrological regimes: Self-Organizing Maps (SOMs)

In SOM approach, developed by Kohonen (1982), input variables are mapped into a discrete map space, consisting of map neurons, grouping similar patterns together. Architecture of SOM comprises two layers, an input layer, and an output (map) layer. These two layers are completely connected. Initial weight vectors of the map neurons are randomly selected. In each iteration step, best matching map unit is chosen for the selected input vector. The best matching map unit is the one with the weight vector that most closely matches the training example. Then, weight vectors of all map neurons in the neighbourhood of the winning neuron are updated. The above procedure is repeated until there are no noticeable changes in the weight vectors. Detail explanation of the algorithm is given in Haykin (1999).

Generally, the number of map neurons should be greater than or equal to the number of clusters. However, this is not known. The number of classes was varied by

setting the number of map neurons to 2,3,4,6 and, 8. Each map neuron was considered to represent a hydrological regime. Modular neural networks (MNNs) were then trained in a supervised mode using the hard classification rule. Schematic diagram of the modular approach is presented in Figure 4.1.



k: Number of clusters or local models in the modular model

Figure 4.1: Schematic representation of the proposed modelling approach.

4.4 Forecasting models

This section describes the linear and nonlinear modelling approaches used in the study to approximate the R-R relationship.

4.4.1 Linear forecasting model

The model input consists of exogenous inputs, i.e., time lagged components of rainfall and runoff. Thus, it resembles to the ARX (Auto Regressive with eXogenous) type of linear stochastic model (Equation 4.2). Coefficients of the model are determined using linear least square method.

$$Q_{(t+1)} = \sum_{i=0}^n b_i Q_{(t-i)} + \sum_{j=0}^m c_j R_{(t-j)} + a_0 \quad (4.2)$$

4.4.2 Nonlinear forecasting model: Artificial Neural Networks (ANNs)

Three-layered MLPs were trained with normalized inputs. The activation function of the hidden neurons was hyperbolic tangent function. The number of hidden layer neurons was determined for global and local models. Cross-validation was also used to prevent the over-fitting problem. ANN is an unstable predictor, which provides different forecasts when trained with different initial parameters. Hence, ensemble of fifty model forecasts was considered in each case. Simple average method was used to combine the model outputs. This improves the generalization error unlike in SNNs (Sharkey, 1999). The iterative approach was utilized to compute the forecasts at different forecasting horizons.

4.5 Performances of global and modular rainfall-runoff (R-R) models

Performances of the forecasting models were evaluated based on mean absolute error (MAE) and correlation coefficient (R).

The following notation was used to refer models. First symbol indicates the use of either Q or dQ inputs. The second symbol is to differentiate the global model (S) from modular model (M) followed by type of model, i.e., NN for neural network and ARX for linear model. Final symbolization is to identify the number of local models in a modular model.

4.5.1 Model performance in rainfall-runoff (R-R) process representation

The statistical performances of the Q-MNN models in one-step-ahead forecasts are presented in Figure 4.2. Plotting class positions in the discharge time series offers a fast way to get insight of the models' predictability. Figure 4.3 represents the visualization of classes in 2-class, 3-class, 4-class, and 6-class classifications.

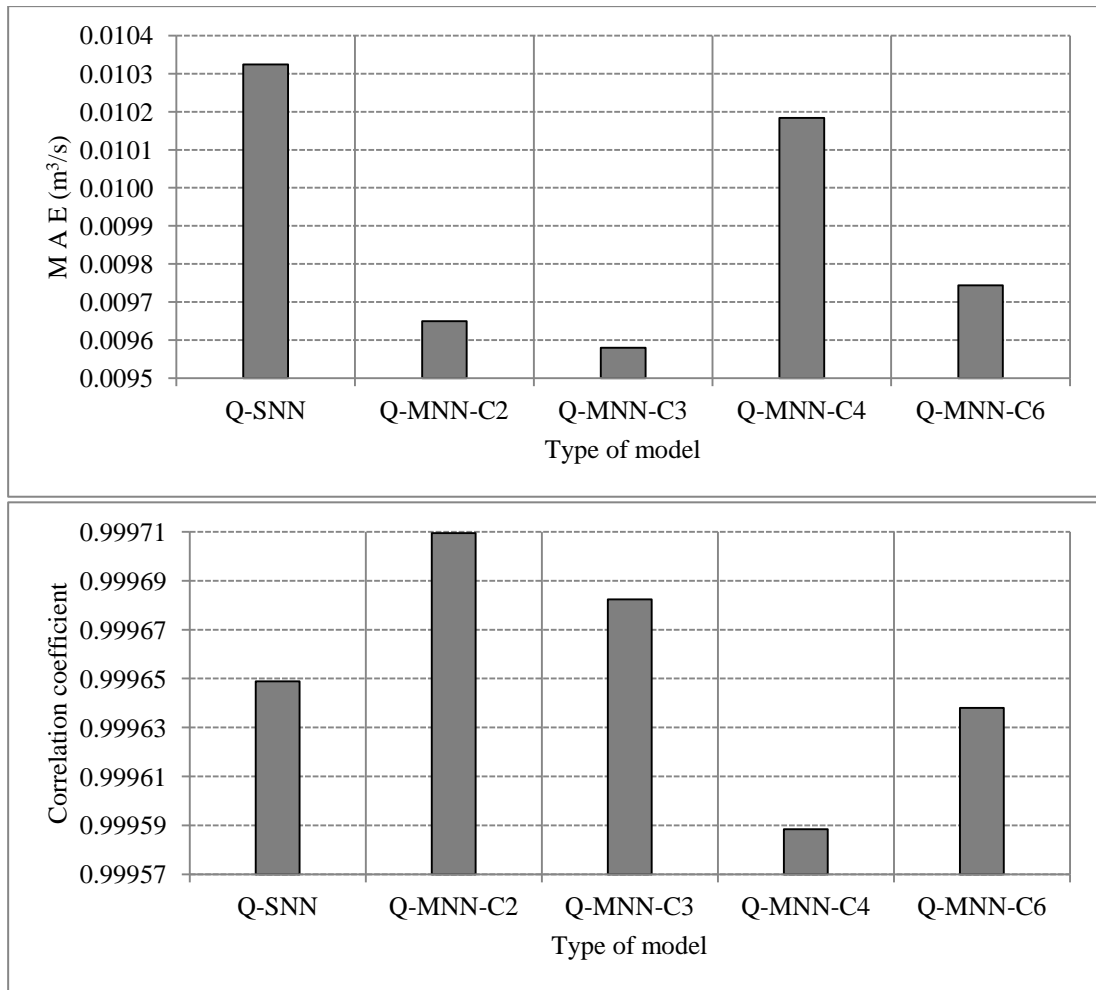


Figure 4.2: Performances of the Q-MNN models.

It can be observed from Figure 4.3 that classification based on Q and rainfall inputs mainly represents low flows and high flows. As a result, the results show no significant improvement in model predictability with increase in number of partitions; however, Q-MNN models produced lower MAEs than Q-SNN model (Figure 4.2). This is attributed to the different runoff generation processes for the high flows and low flows. We can speculate that threshold based MNN models representing low, medium, and high absolute discharges (Zhang and Govindaraju, 2000) would also provide comparable forecasts.

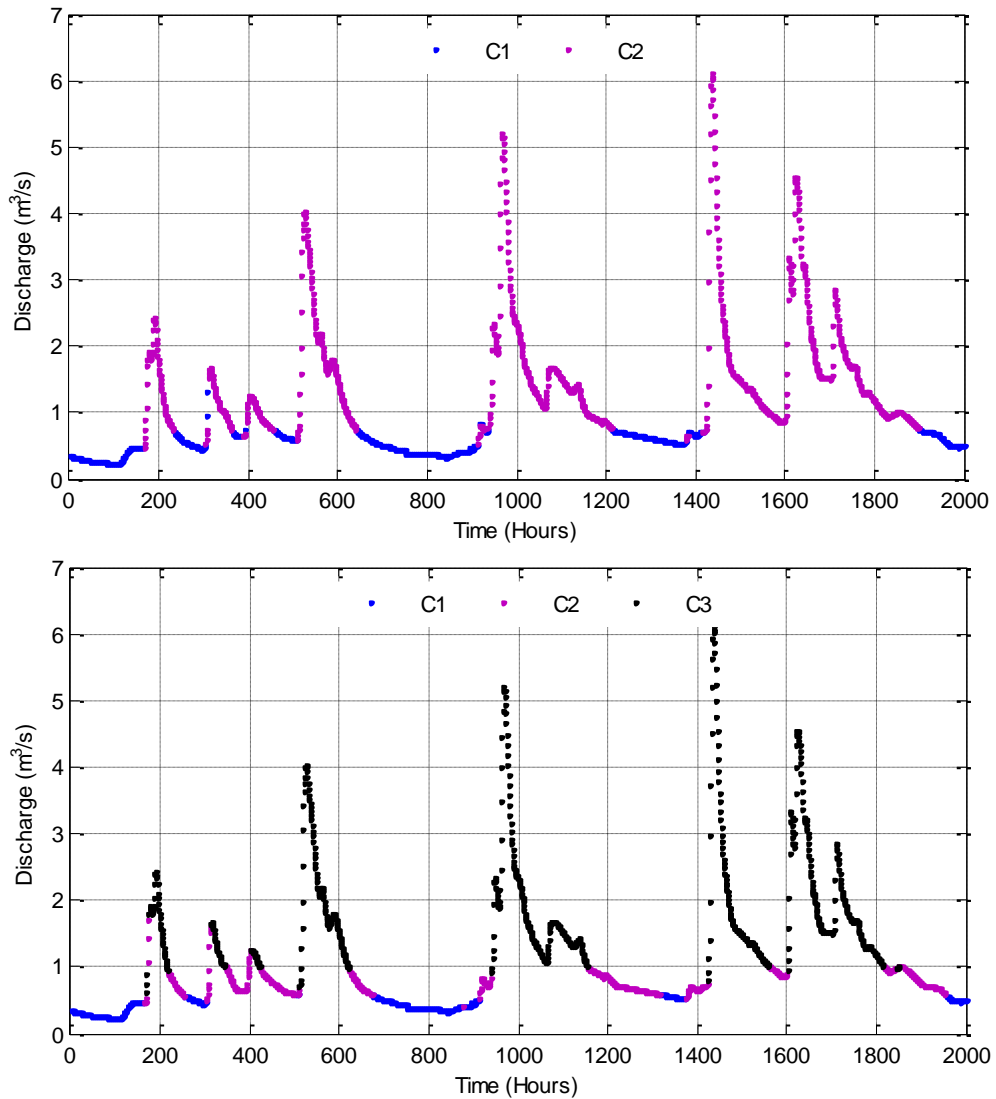


Figure 4.3a: Position of classes in (a) 2-class, and (b) 3-class classifications.
 Note: classifier inputs: Q and rainfall data

Classes C2, C3, C4, and C3 in 2-class, 3-class, 4-class, and 6-class classifications, respectively include high flows. It is understood that functional relationship of flow recession is mainly governed by preceding discharges, while it is by previous rainfall values for an increase in flow. Function approximation for both changes in flows will provide an average of individually approximated functions. In addition, rules for generating low flows and high flows are different. Thus, firstly, efforts should be made on differentiating the change in flow and secondly based on magnitudes of flow. The results indicate that the domain partition method based on Q

data and rainfall data is not effective in identifying the similar flow generating instances.

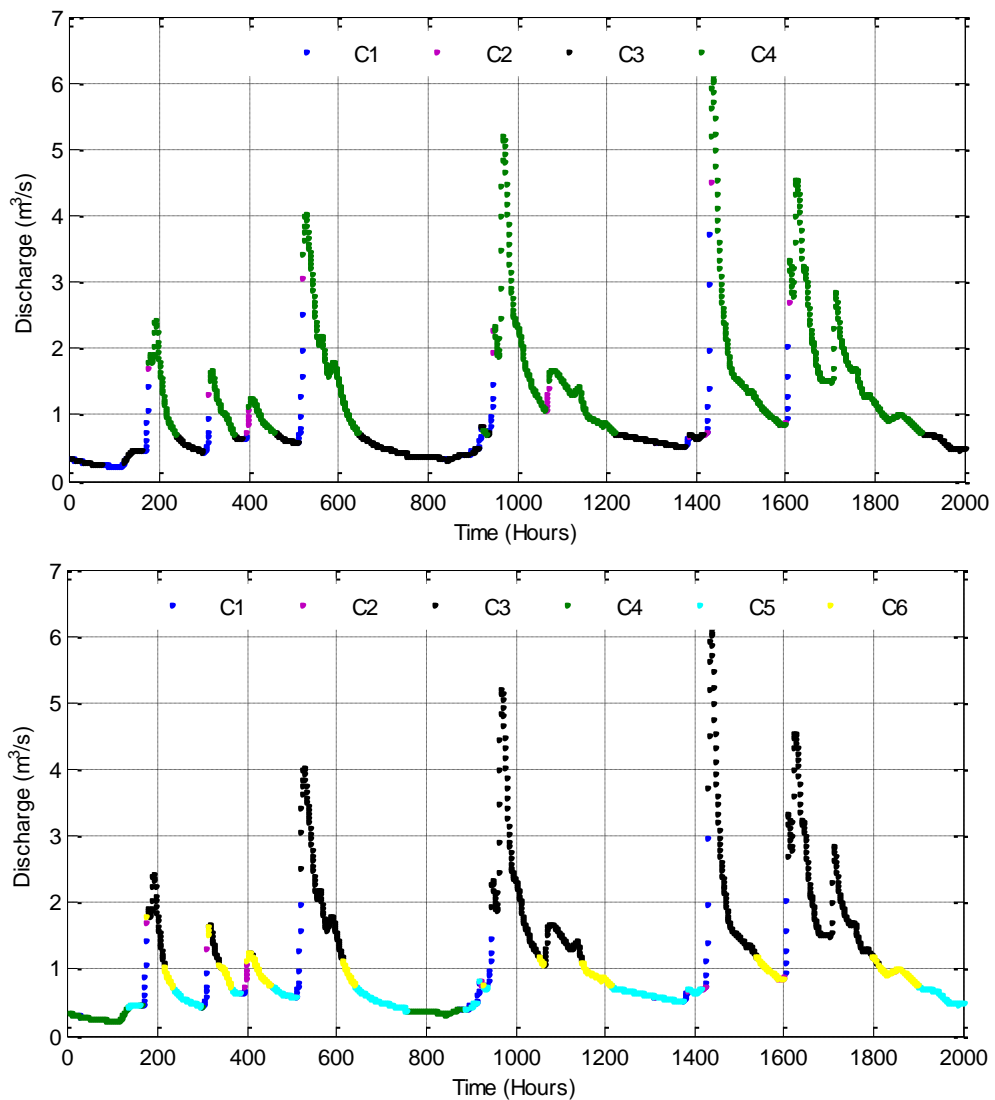


Figure 4.3b: Position of classes in; (a) 4-class, and (b) 6-class classifications.
Note: classifier inputs: Q and rainfall data

Measured rainfall over a specified time interval corresponds to increase in runoff over same time interval. Therefore, adjacent differencing can be used to differentiate the change in flow, which introduces the ‘plus’ and ‘minus’ values to the dQ . With this understanding, unsupervised classification of $dQ_{(t+1)}$ data was tested using rainfall and dQ inputs. Figure 4.4 presents the statistical performances of the dQ-MNN models in estimating runoff.

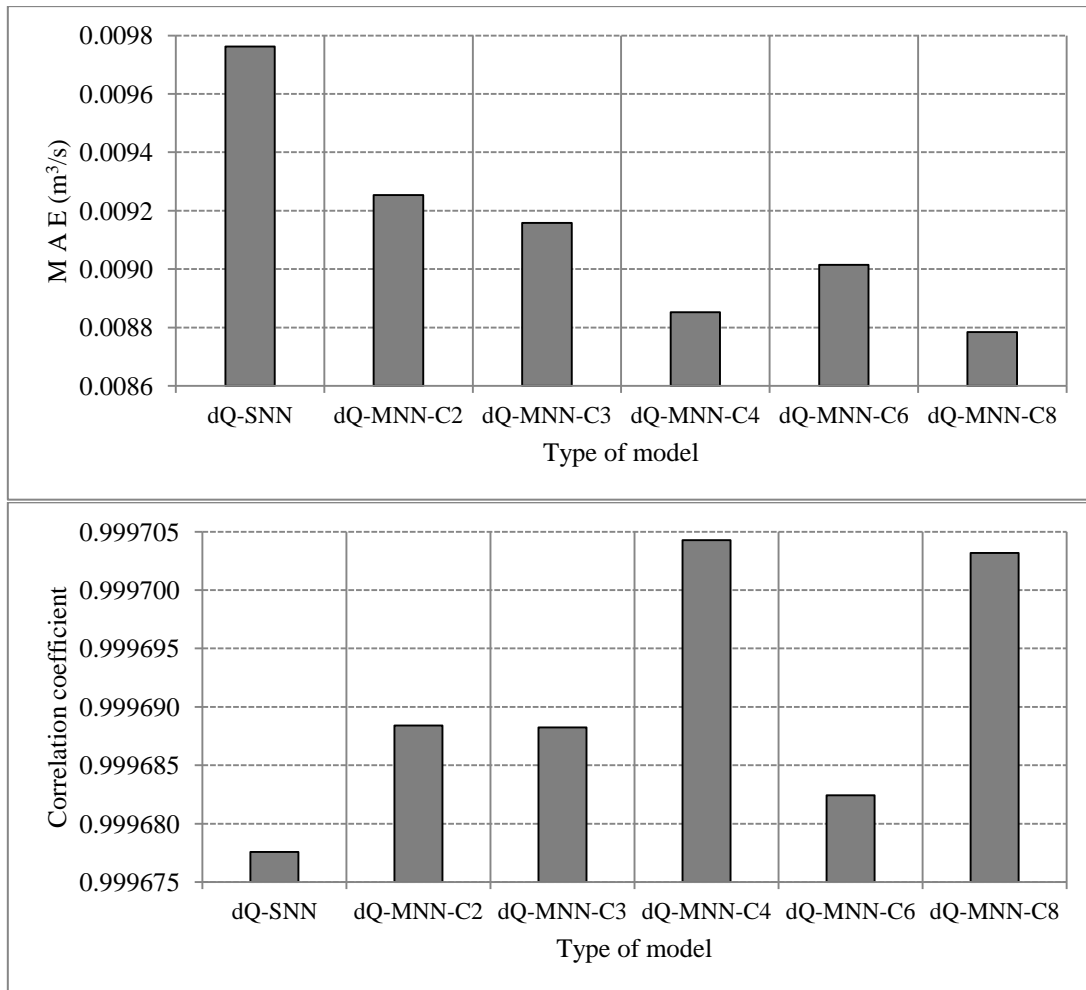


Figure 4.4: Performances of the dQ-MNN models.

The results show that the MNN models produced better testing accuracy compared to SNN model. Increase in number of hydrological regimes was lead to lower MAEs. Figure 4.5 shows the class positions in discharge time series. The color and symbols stand for the classes. It can be observed that, unlike in Figure 4.3, flows are classified according to change in flows.

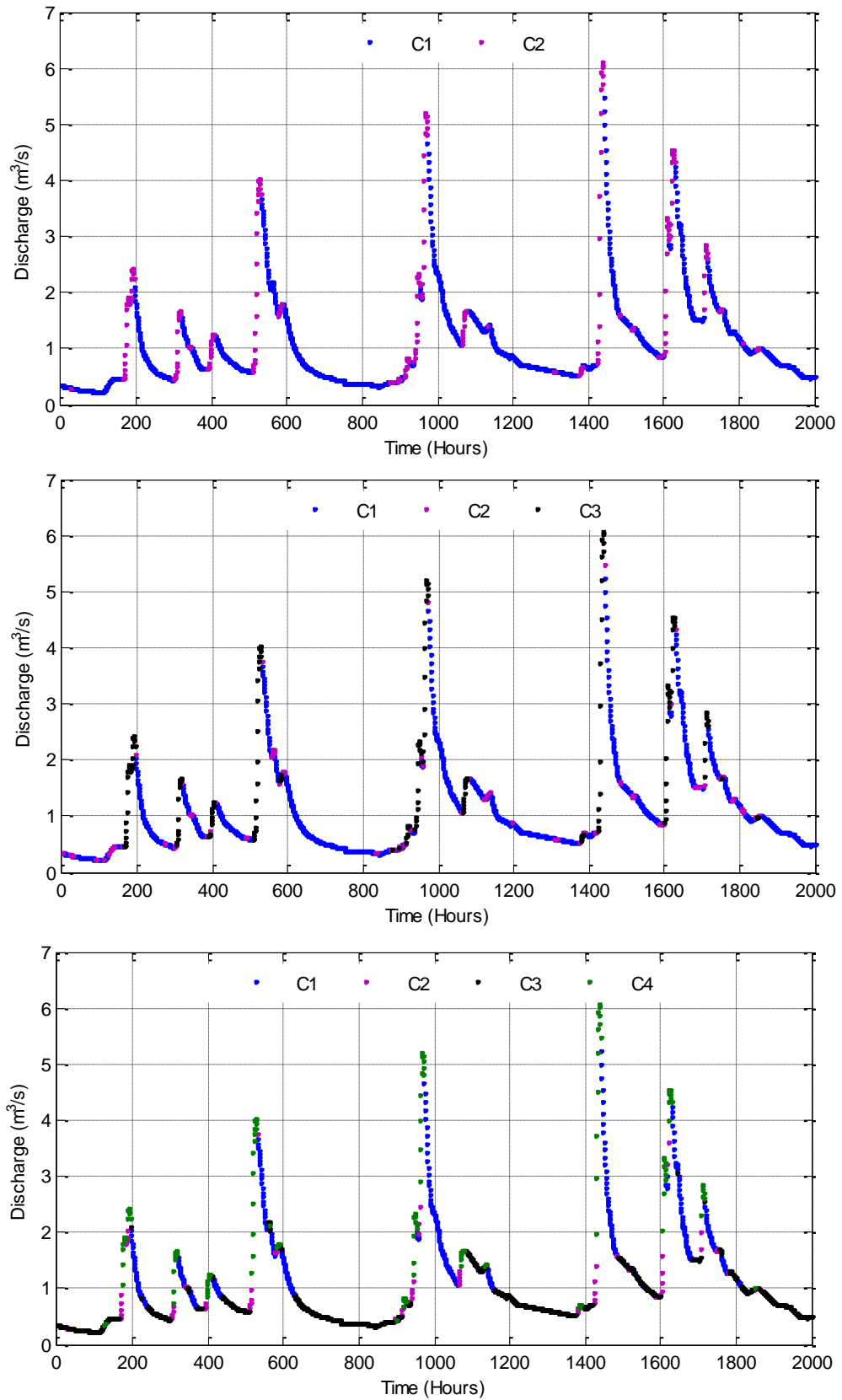


Figure 4.5a: Position of classes in; (a) 2-class, (b) 3-class, and (c) 4-class classifications.
 Note: classifier inputs: dQ and rainfall data

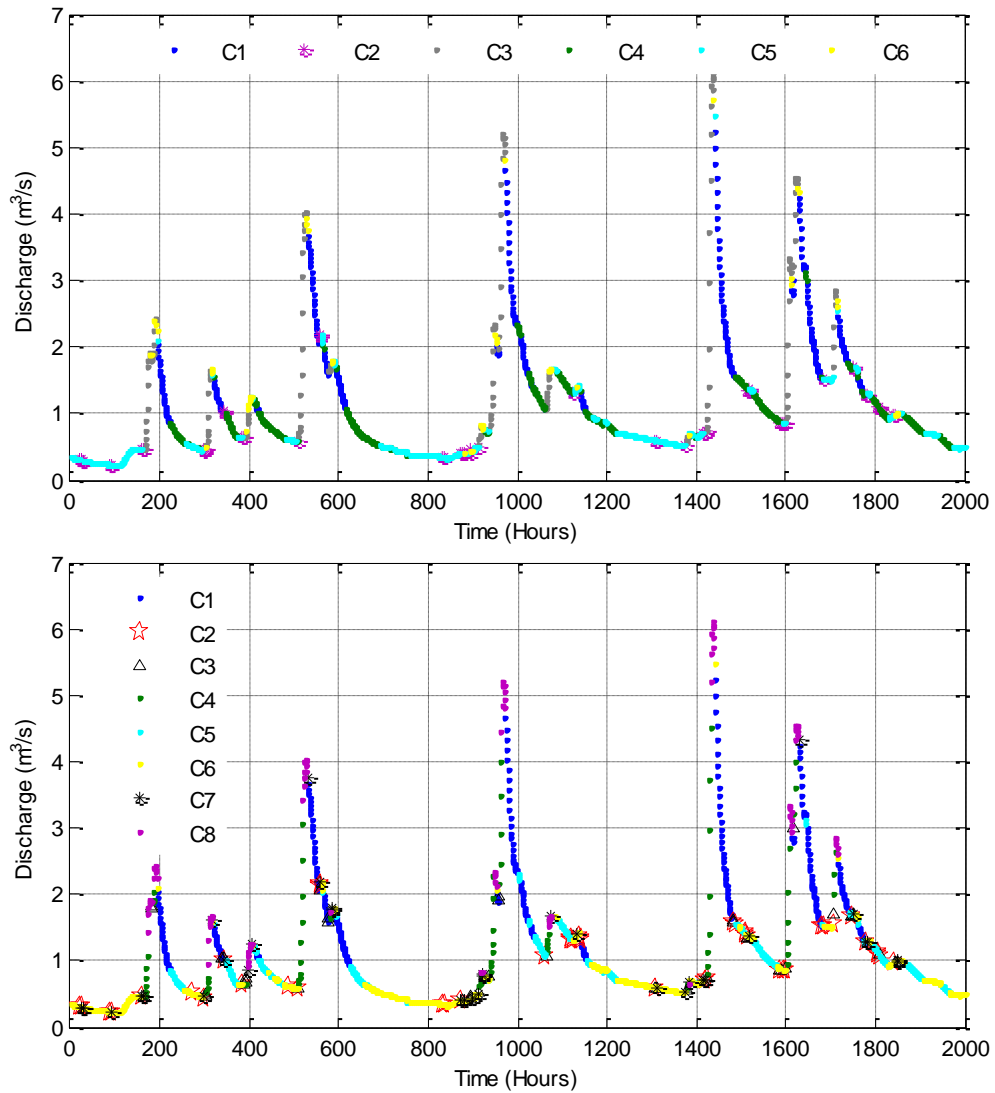
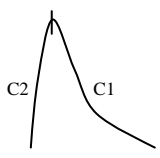
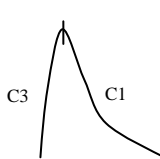
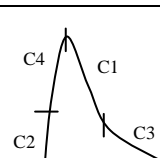
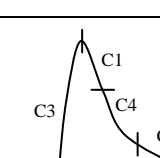
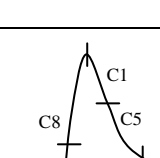


Figure 4.5b: Position of classes in in; (a) 6-class and (b) 8-class classifications.
 Note: classifier inputs: dQ and rainfall data

Table 4.1 summarizes the parts of the hydrograph represented by each classification based on the visualizations of Figure 4.5. It shows that use of dQ instead of Q identifies time variability of hydrological processes. Based on this information, we can confirm that dQ and rainfall classifier inputs in 2-class classification divide the input space, based on whether the future estimate causes decrease in or increase in flow. Further classification will subdivide those two regions. The classes, which are not included in Table 4.1 are arbitrarily present in the discharge time series. It is to be noted that the SOM classifier determines the class for a particular forecasting instance.

It is not possible to provide the threshold discharge values for the classes as these values differ for different storm events.

Table 4.1: Parts of the hydrograph represented by each classification.

Classification	Parts of the hydrograph
Two-class	 A hydrograph curve with a peak. The rising limb is labeled C2 and the falling limb is labeled C1.
Three-class	 A hydrograph curve with a peak. The rising limb is labeled C3 and the falling limb is labeled C1.
Four-class	 A hydrograph curve with a peak. The rising limb is divided into C4 (upper) and C2 (lower). The falling limb is divided into C1 (upper) and C3 (lower).
Six-class	 A hydrograph curve with a peak. The rising limb is labeled C3. The falling limb is divided into C1 (top), C4 (middle), and C5 (bottom).
Eight-class	 A hydrograph curve with a peak. The rising limb is divided into C8 (top) and C4 (bottom). The falling limb is divided into C1 (top), C5 (middle), and C6 (bottom).

Additionally, Figure 4.6 shows the past rainfall and dQ patterns of classes, in order of their occurrences in 8-class classification. This shows that time evolution of rainfall pattern resembles to the Q (or dQ) variation.

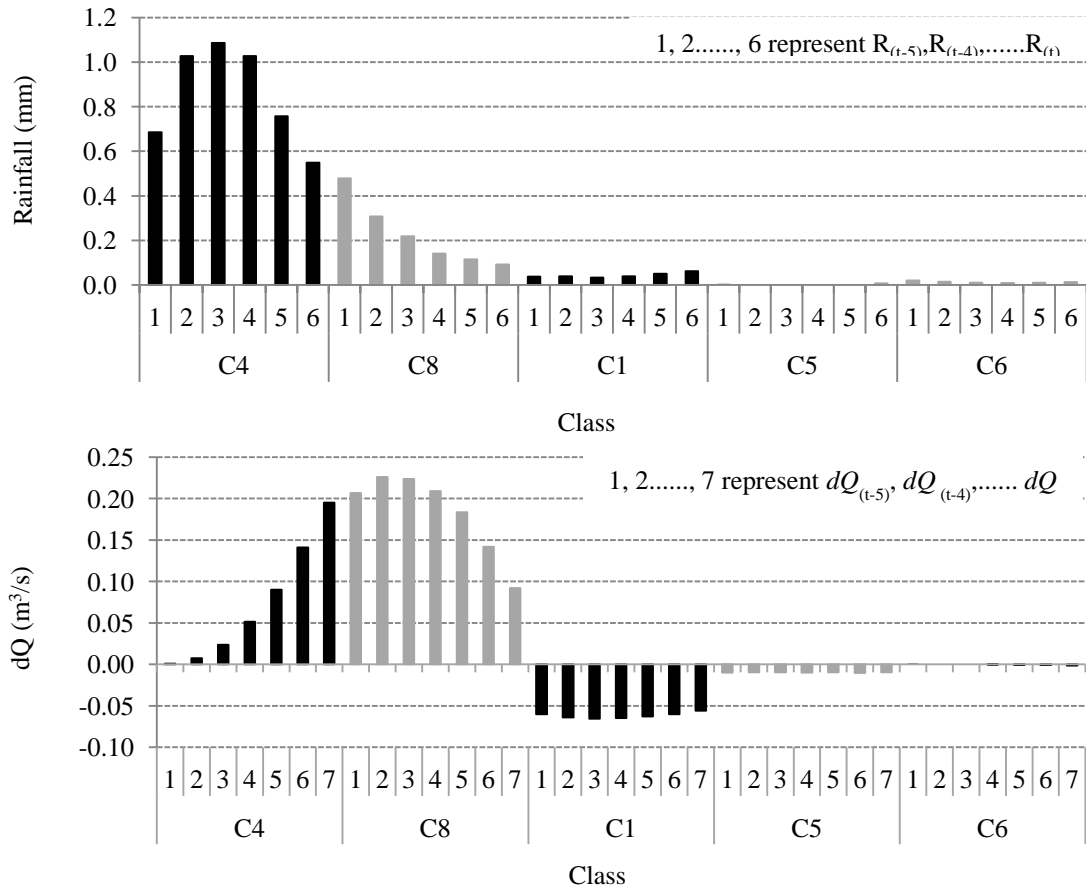


Figure 4.6: (a) Rainfall pattern. (b) dQ pattern.

For a quantitative evaluation, performances of local models and performances of global models in local domains (LD) were compared. Figure 4.7 shows the relative improvement, in MAE, of forecasts in local models compared to their performances in global model representation. We can observe that the LD2, LD3, LD3, and LD7 in 2-class, 3-class, 6-class, and 8-class representation, respectively failed to contribute to the forecast improvement. Those domains represent the rising limb of the hydrograph. Modular model results show that the improvement is considerable in 4-class and 8-class classifications (Figure 4.4). This is attributed to the improvement in forecast accuracy of local models (Figure 4.7). Those two classifications subdivide the rising limb into two or more local regions.

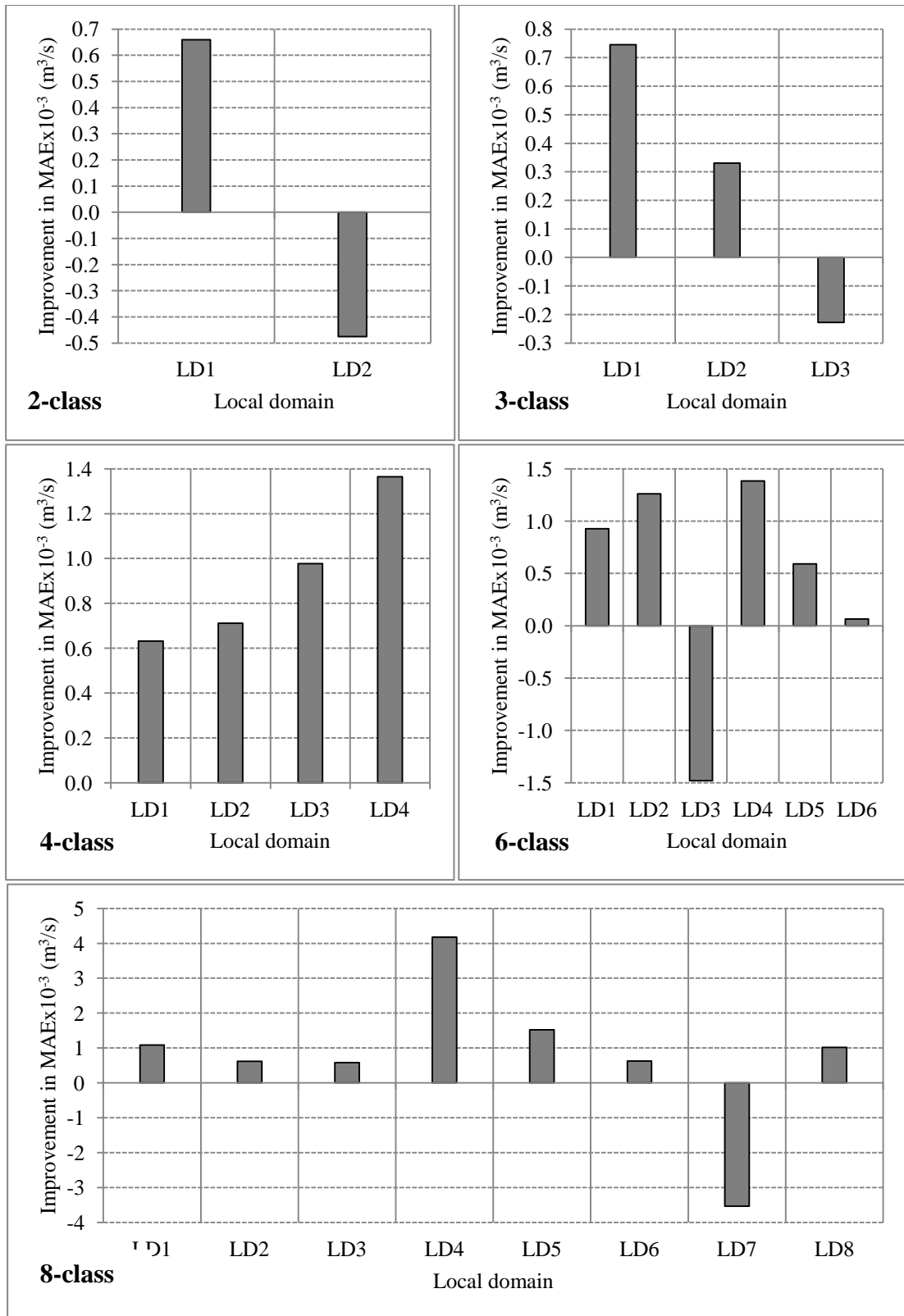


Figure 4.7: Improvement in forecasts of local models compared to global models.

Based on above discussion, dQ-MNN models appear to be good candidates. Classification of model input data enables to decompose functionally different regions

of discharge time series. Some of these functionally different regions might be better approximated with nonlinear functions, while others might be represented by linear functions. Next section will discuss the performances of linear and nonlinear models.

4.5.2 Linear and nonlinear model performances in global and modular model representations

Performances of dQ-ARX and dQ-ANN models are shown in Figure 4.8. Global neural network model performs well compared to ARX model. MAEs of MARX models are much higher than the ANN models. This is because of the higher errors introduced by fitting linear functions for nonlinear variations.

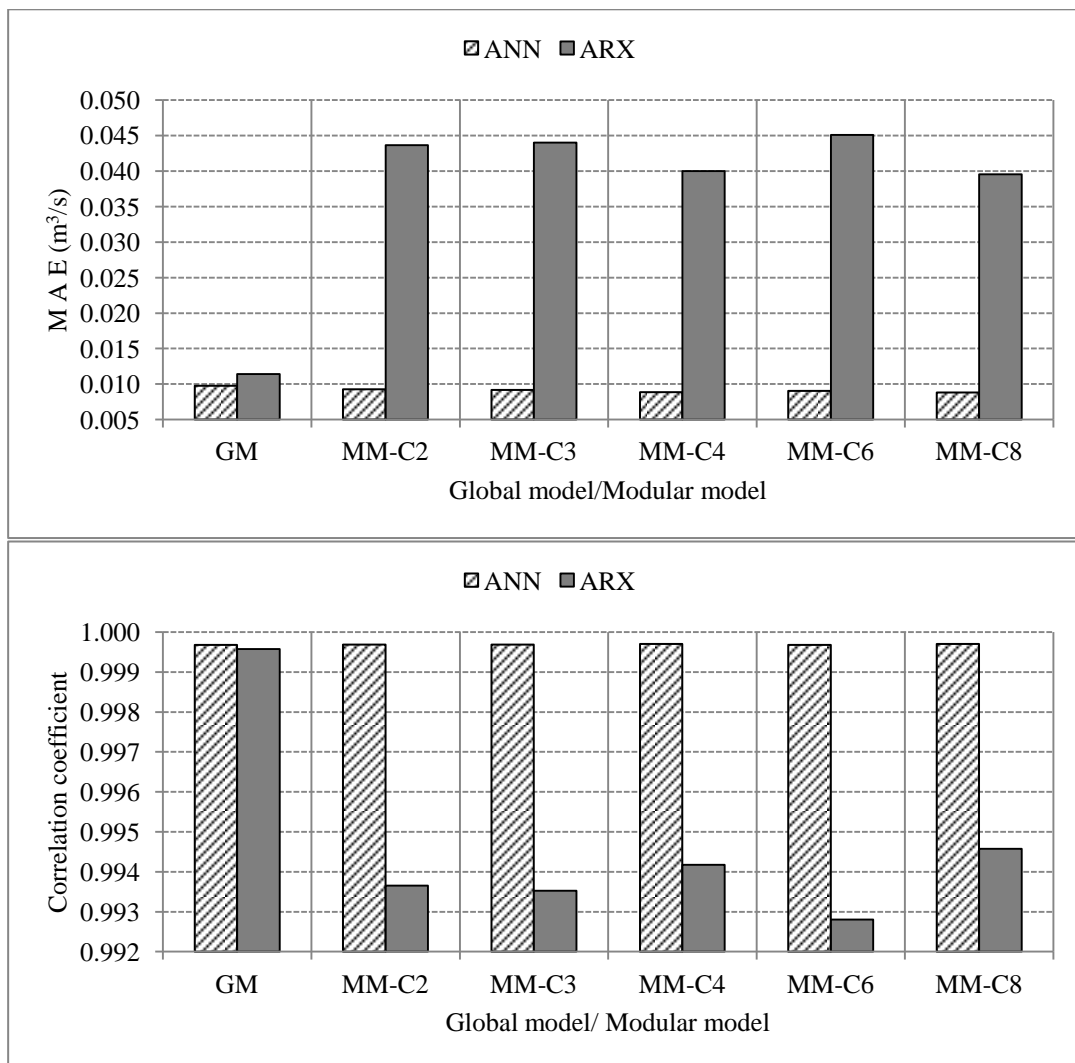


Figure 4.8: Performances of ARX and ANN models in global model (GM) and modular model (MM) representations.

For a quantitative evaluation, we compared the performances of local linear and nonlinear models. Figure 4.9 presents the improvement of forecasts in local domains with ANN.

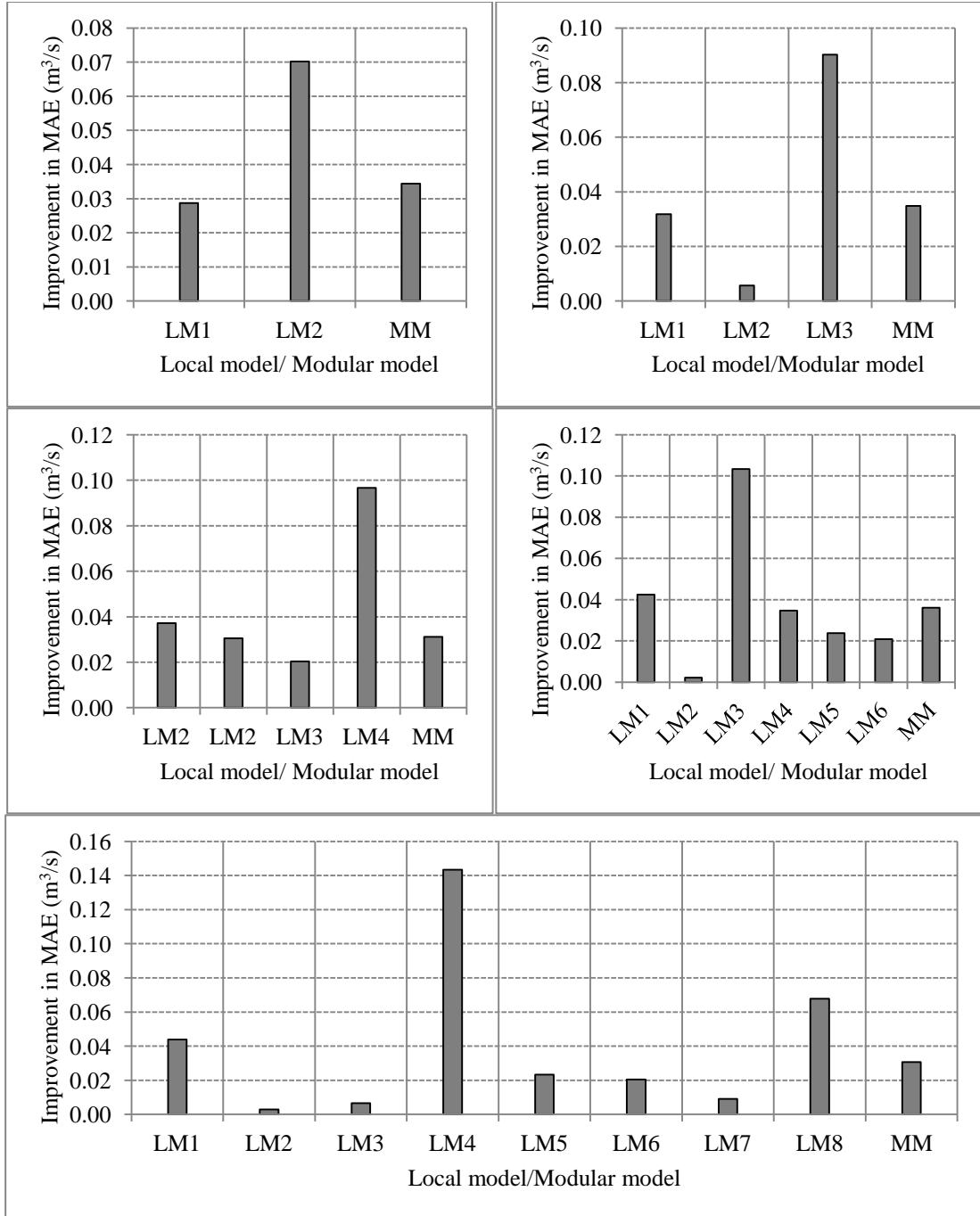


Figure 4.9: Improvement of forecasts in nonlinear local models compared to linear local models.

The results show that the performances of local nonlinear models are at least as good as or better than corresponding local linear models. This is because of the ability

of ANN in learning linear as well as nonlinear functions. Improvement in forecasts depends on the degree of complexity in the flow generating processes.

Analysis of runoff data shows that low flows exist for long period. Figure 4.10a shows the flow duration curve, which summarizes the chances of exceeding a given streamflow. Similar curve was produced for dQ data (Figure 4.10b). Continuous flow records were available for two-year period and only those were used for this analysis.

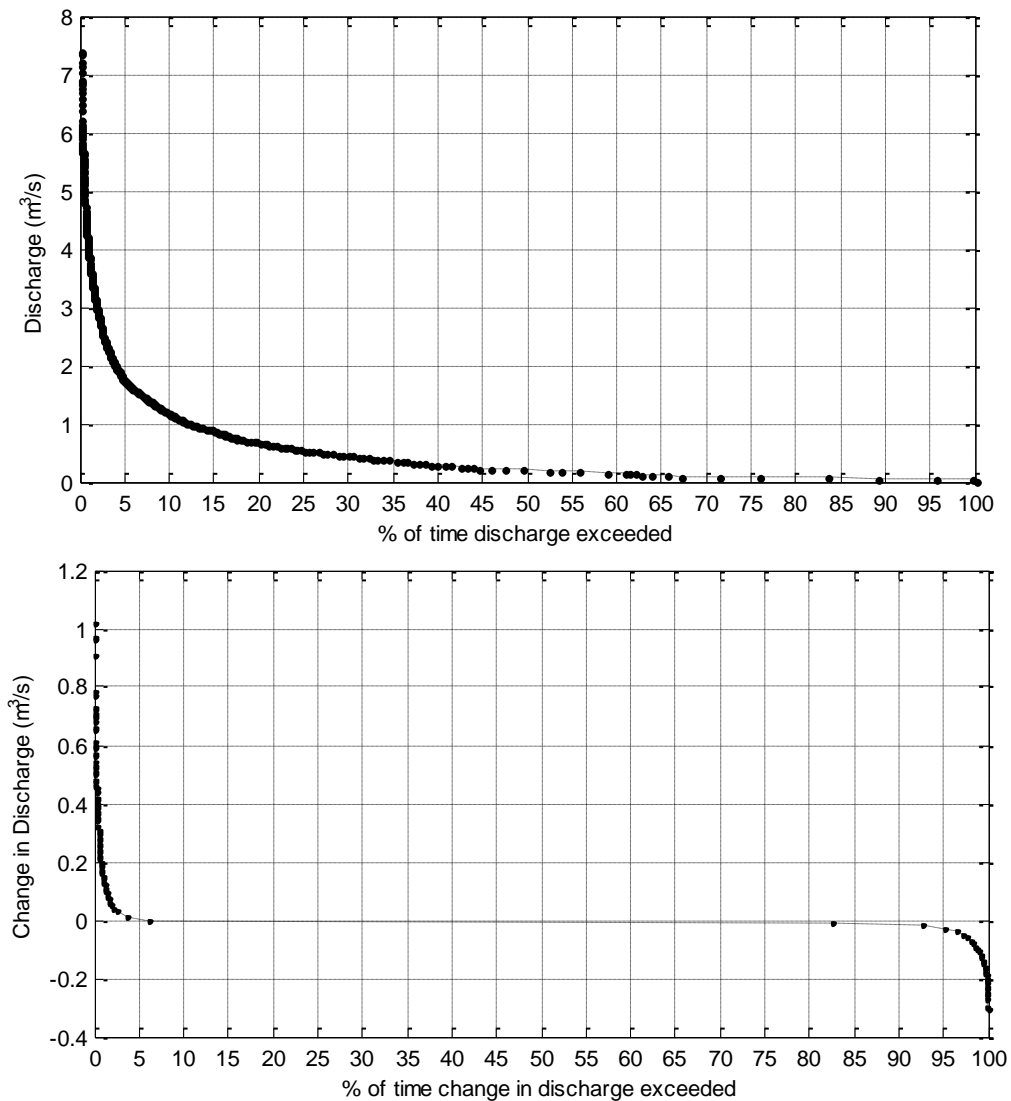


Figure 4.10: Flow duration curve for Orgeval catchment.

Approximately 75% of the data do not show significant variability with preceding values ($-0.01 \text{ m}^3/\text{s} \leq dQ \leq 0.01 \text{ m}^3/\text{s}$). Seventy five percent of the remaining

data correspond to the flow recession. Flow recession is described by mathematical expressions, which are derived from theoretical equations of the drainage aquifer flow, i.e., linearized Dupuit-Boussinesq equation and its variants (Thallaksen, 1995). These equations, which are exponential functions, have the time as a variable. Instead, rate of change in flow is expressed as a function of Q eliminating the variable t as, $-dQ/dt = a.Q^b$, where a and b are constants. A curve is fitted to the observed recession curves to find the coefficients of the mean curve. It has been found that the hydraulics of flow and heterogeneity in catchment characteristics can give rise to nonlinear recession curves (Clark et al., 2009; Harman et al., 2009). The recession rates and shape of recession curves depend on the catchment geology, distance from catchment boundary to its outlet, infiltration characteristics of soils, river and aquifer hydraulic characteristics, frequency and amount of recharge, vegetation characteristics, topography, and climate (Clark et al., 2009; Harman et al., 2009; Thallaksen, 1995). These factors collectively contribute to the losses and gains of flow during the recession. Clark et al. (2009) showed that shape of the recession curve varies with the catchment scale from a linear reservoir type, i.e., exponent with 1, for individual hillslope (0.001 km^2) to nonlinear situations at larger scales (0.1 km^2 and 0.41 km^2). The values of exponent b will be different for various antecedent conditions. For example, higher peak discharge leads to a steeper recession slope. The time variability in recessions can be handled with the modular model approach. As quickflow leaves the catchment, a sharp drop in flow is observable. This will flatten out with delayed supply of subsurface stores. Then flow will become nearly constant if it is sustained by groundwater storage. Recession behaviour varies in these three segments of the recession curve.

The complexity of hydrological processes varies considerably among catchments and effectiveness of modular model approach is thereby subjective.

The above sub-section mainly considers the representation of the R-R process. The next section will analyse the model performance at different forecasting horizons.

4.5.3 Model performance in multi-step-ahead forecasts

Multi-step-ahead forecast errors are due to the process approximation errors and the sensitivity of model for errors. In Chapter 3, it was shown that Q-ANN model error accumulates at a greater rate than the dQ-ANN models. This section focuses on the error accumulation properties of the dQ-MNN models compared to dQ-SNN model.

Figure 4.11 shows the MAE and correlation coefficient variation of dQ-ANN models with the forecasting horizon. Modular models perform well compared to the global model in multi-step-ahead forecasts. This is attributed to the improvement in R-R process approximation and model complexity reduction with modular models. Improvement is significant in dQ-MNN-C4 and dQ-MNN-C8 models.

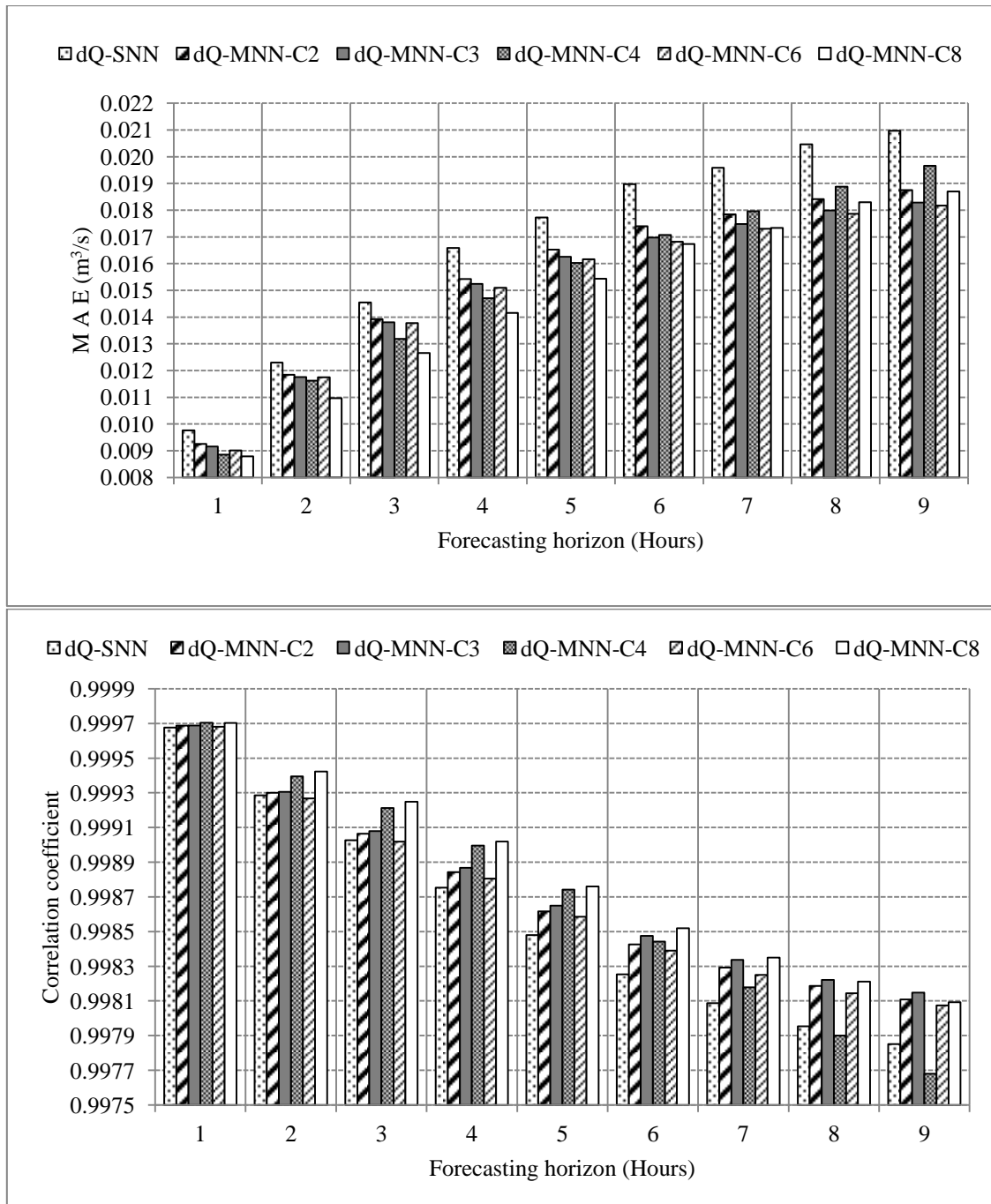


Figure 4.11: Performances of the dQ-MNN models.

Accumulated error in iterative steps might result in errors in input data classification, which finally causes reduction in forecasting performance. This can be significant at large steps. Error due to misclassification can be calculated as the difference of the actual model error and the model error for correctly classified data. This is presented in Table 4.2. It shows that forecast errors due to misclassification are

much higher for lead times greater than the catchment concentration time. It can be observed that it is much higher in dQ-MNN-C4 and dQ-MNN-C8 models.

Table 4.2: Error accumulated due to the classification error in dQ-MNN models.

(MAE of actual model- MAE with correctly classified inputs) $\times 10^{-3}$								
Model	Forecasting horizon (Hours)							
	2	3	4	5	6	7	8	9
dQ-MNN-C2	-0.001	-0.014	-0.018	-0.015	-0.007	-0.002	0.007	0.016
dQ-MNN-C3	-0.018	-0.007	-0.030	-0.049	-0.025	-0.005	0.043	0.073
dQ-MNN-C4	0.057	0.023	0.073	0.230	0.377	0.609	0.817	1.137
dQ-MNN-C6	0.007	0.034	0.018	0.049	0.029	0.069	0.047	0.053
dQ-MNN-C8	-0.023	0.044	0.088	0.087	0.186	0.363	0.287	0.296

(Corr. coeff. of actual model- Corr. coeff. with correctly classified inputs) $\times 10^{-3}$								
Model	Forecasting horizon (Hours)							
	2	3	4	5	6	7	8	9
dQ-MNN-C2	0.000	0.001	0.000	0.001	0.002	0.006	0.009	0.011
dQ-MNN-C3	-0.001	-0.002	-0.008	-0.009	-0.010	0.006	0.013	0.015
dQ-MNN-C4	0.004	0.004	0.027	0.058	0.159	0.210	0.327	0.422
dQ-MNN-C6	0.001	0.000	-0.002	0.000	-0.003	0.000	-0.004	0.001
dQ-MNN-C8	0.030	0.058	0.070	0.062	0.052	0.032	-0.008	-0.070

Classification based on dQ and rainfall model inputs divides the functionally different regions of the modelling domain. Therefore, approximated functions for different sub-classes have different complexities; thereby the changeover of exemplar classification can have a negative or positive effect on the forecasting accuracy of local models. More complex local models will response negatively for the misclassified data. Depending on these local model error fluctuations, overall modular model will have fluctuations. For example, errors are much higher in subclasses correspond to the increases in flow (Table 4.2 and Figure 4.2).

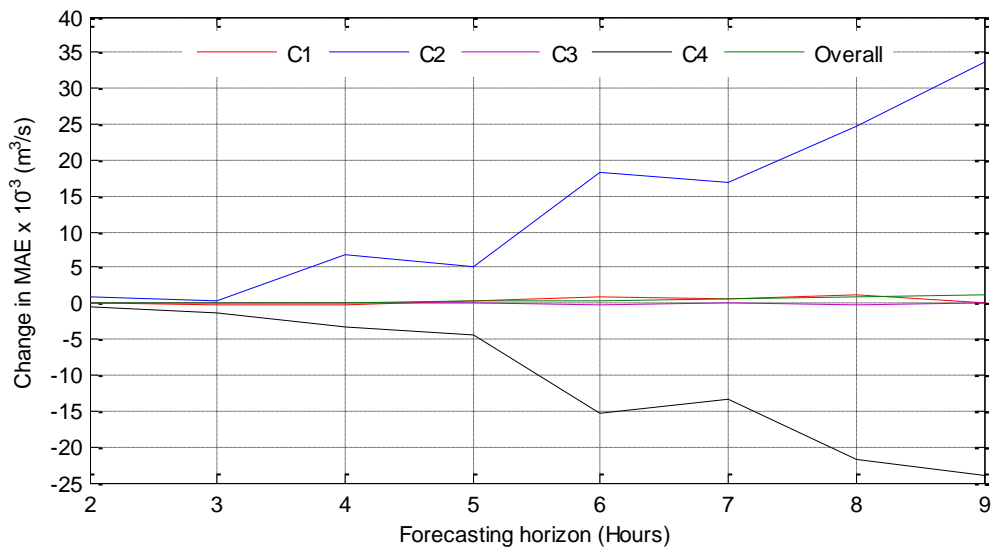
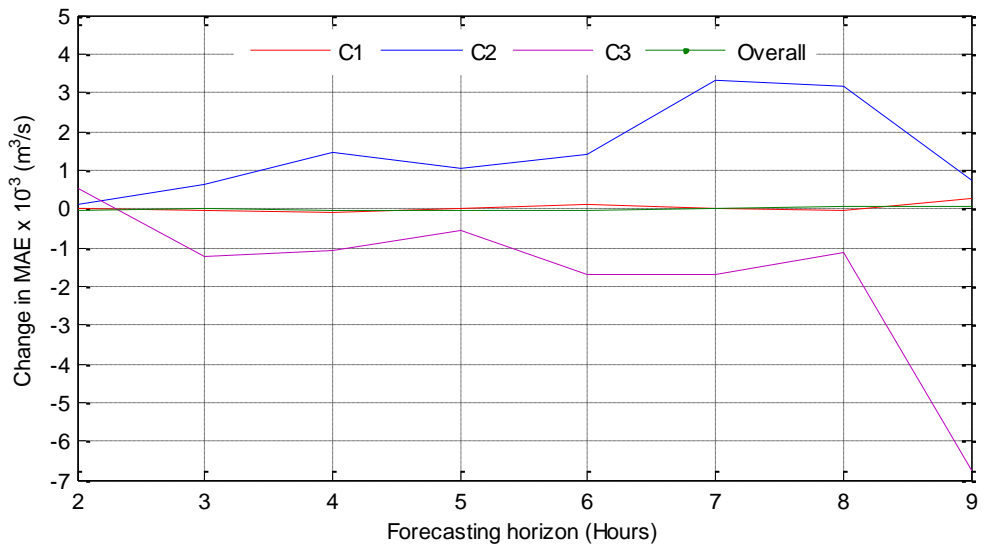
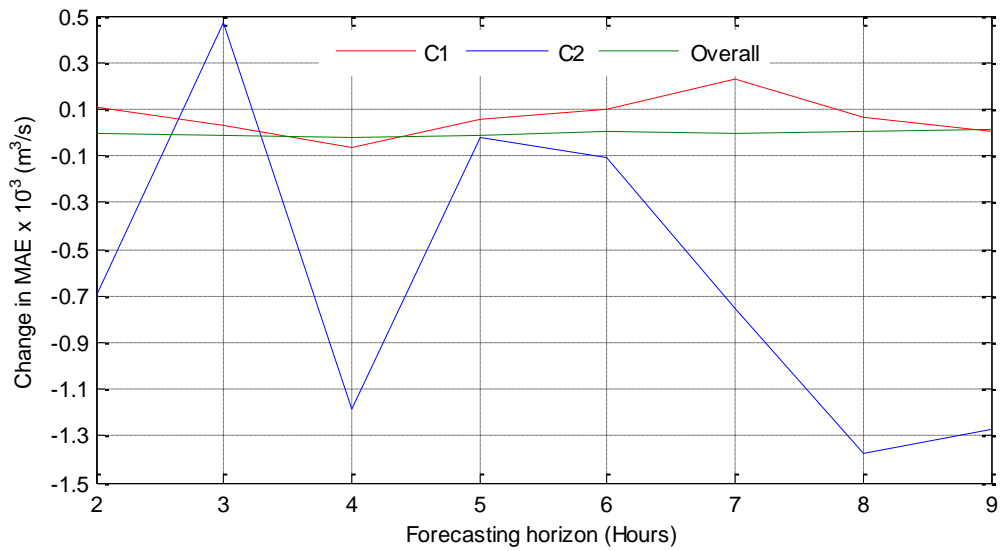


Figure 4.12a: Error accumulated due to the classification error in individual classes of (a) dQ-MNN-C2, (b) dQ-MNN-C3, and (c) dQ-MNN-C4 models.

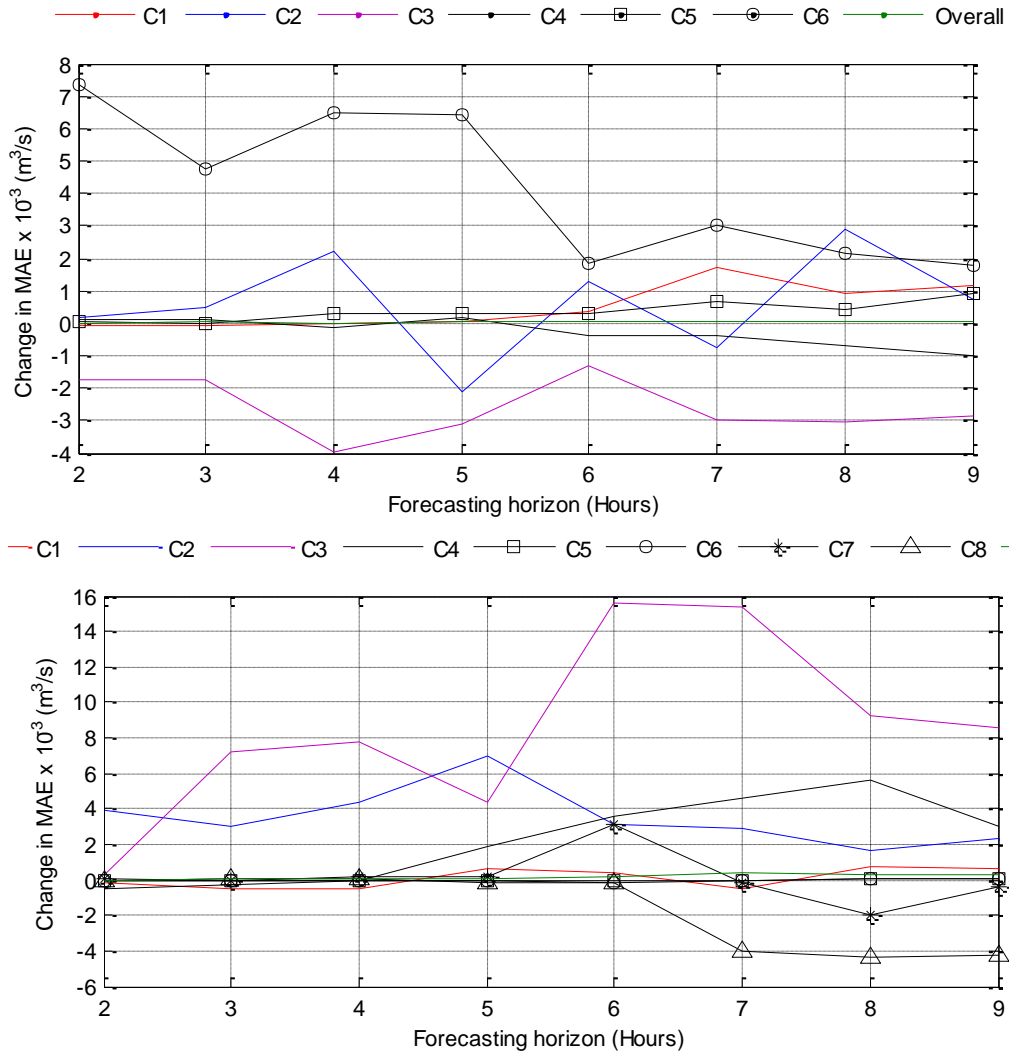


Figure 4.12b: Error accumulated due to the classification error in individual classes of (a) dQ-MNN-C6, and (b) dQ-MNN-C8 models.

Predictability of models for extreme events is also important if the application is for flood forecasting.

4.5.4 Extrapolation capability of global and modular models

Figure 4.13 presents the error (actual discharge-predicted discharge) generated by global and modular ANN and ARX models for out-of-range data, i.e., for discharges greater than the training discharge data.

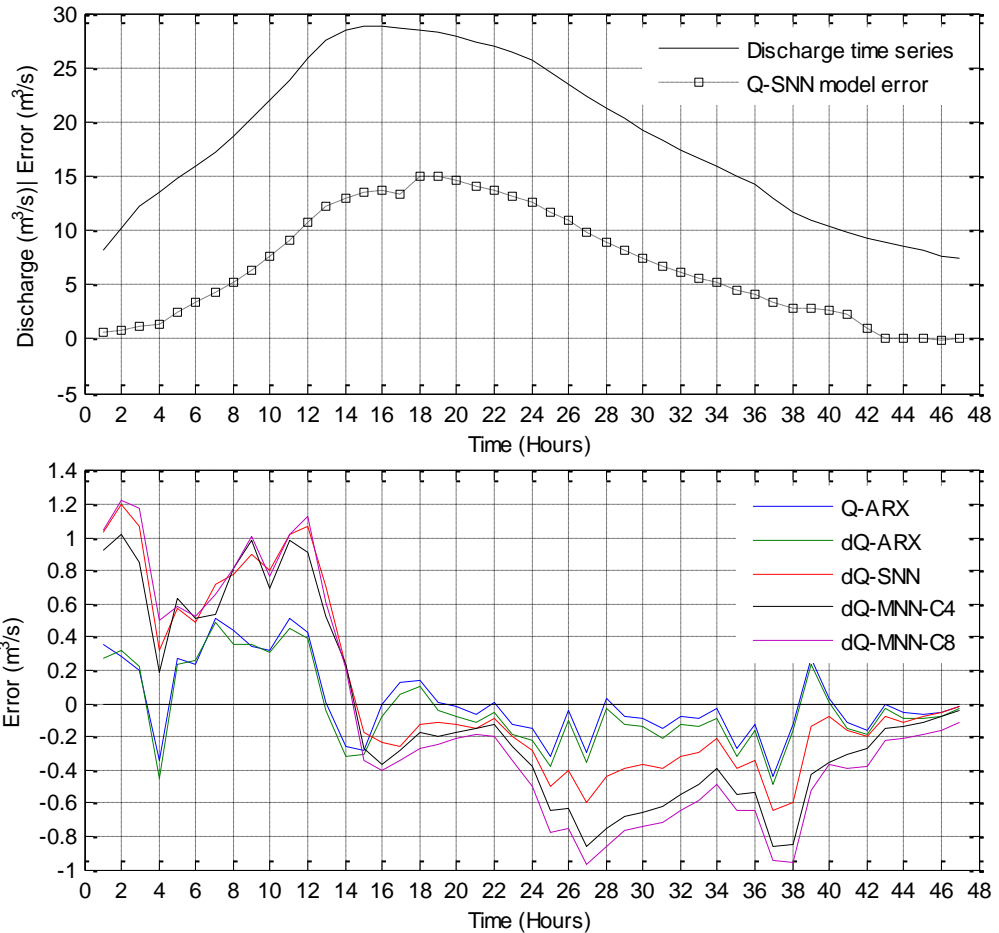


Figure 4.13: Performances of models for out-of-range data.

It can be observed from Figure 4.13a that Q-ANN model errors are much higher. This is attributed to the ANNs' under-predictability of high flows. In case of dQ data, $\pm dQ$ s greater than the training $\pm dQ$ s produced under predicted dQ s. Magnitudes of dQ s are positive for rising limb, negative for falling limb, and closer to zero near peak discharges. As a result, dQ-ANN model produced under predicted values for rising limb of a hydrograph and over predicted values for falling limb of a hydrograph. However, peak discharge errors are closer to zero (Figure 4.13b). dQ-models have the greatest tendency to yield lower MAEs. Higher errors for out-of-range discharges are expected, since the nonlinear processes cannot be approximated exactly. Moreover, dQ-MNN models' MAEs are higher than the dQ-SNN. Fitting a function to a particular data range might increase the forecasting error for out-of-range data. The lower errors

of ARX models are due to the linear dependencies of autoregressive components of Q/dQ time series data.

Overall, class locations in discharge time series and local model MAEs suggest that input data classification mainly categorizes input-output domain based on complexity of the runoff generation process. Runoff generation processes are different for rising limb, falling limb, and base flow. In addition, it varies for different magnitudes. Based on this discussion, local model approximations, like locally weighted regression (Cleveland, 1979), would reduce the bias error (i.e. error due to process/model mismatch). This is the reason for improved forecasting performance.

4.6 Conclusions

This chapter considered approximation of R-R process with DDMs, which was based on modularity. The first step involved search of modularity-associated features of hydro-meteorological input data. Rainfall and dQ inputs clearly recognized the parts of the hydrograph. In addition, adjacent differencing was useful for improving forecasting accuracy. It was also shown that by applying modular based approach forecasting error could be reduced. This indicates modular models are more robust to temporal evolution of rainfall-runoff process than global model. It is to be noted that the number of hydrological regimes is subjective. This might depend on the range of the change in discharge. The higher the range, the more hydrological regimes may persist. One limitation of the modular model approach is that the large amount of data is required for training phase to avoid the use of same data set twice. Modular models are comprised of specialized modules performing individual specialized tasks. Instead, a nonlinear function approximation method that can capture temporal variation of

rainfall-runoff process would be the promising modelling approach. Implementation of this type of modelling approach is challenging.

CHAPTER 5

FLOW ROUTING WITH DATA DRIVEN MODELS

5.1 Introduction

The earlier two chapters were on prediction related issues of lump catchment models. Possible extension of the research basis of lump catchment into large-scale catchments was discussed in Chapter 2. This chapter considers the flow routing with data driven models. Prediction improvement methods are illustrated using the streamflow data of the White river catchment, Indiana.

5.2 Description of the White river catchment

The White river has two tributaries namely, the West Fork and the East Fork. The West Fork is the main and the longest tributary (583 km), which originates from north-western Indiana. The East Fork, 309 km in length, starts at Columbus. The two tributaries join very near to the end of the watershed at Petersburg, Indiana. The White river basin has an area of 14,880 km². Figure 5.1 shows the river map with approximate locations of the measurement stations.

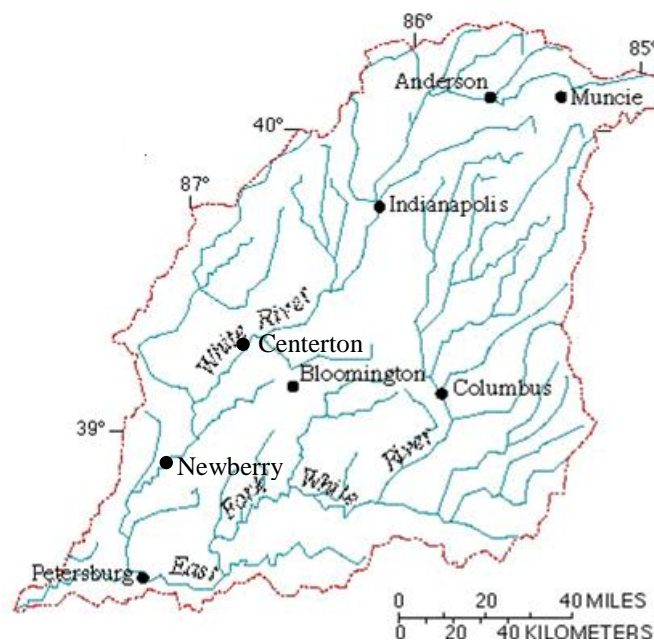


Figure 5:1: White river catchment.

5.3 Input determination

Thirty-five years (1957-1992) of daily data were obtained from the United States Geological Survey (USGS) website. Flow travel times of each river reach were determined using the cross-correlation analysis. The flow travel times suggested that the temporal resolution of flow data was large to consider all the measurement points for modelling. For this reason, West Fork tributary was considered for the analysis and measurement points at Anderson (A), Indianapolis (I), Centerton (C), and Newberry (N) were included. The statistics of the streamflow time series data are given in Table 5.1.

Table 5.1: Statistics of the streamflow time series data (m^3/s).

Measurement station	Minimum	Maximum	Mean	Standard deviation
Anderson	1	704	18	31
Indianapolis	2	1551	64	103
Centerton	10	1985	109	153
Newberry	15	3093	216	265

Figure 5.2 shows the flow time series of the year 1992. We can observe that the flow is accumulated with increasing the distance from the source of the West Fork. The flow at a particular point includes flows from upstream measurement stations and that from the intermediate area.

Discharge fluctuations at a downstream location are a result of changes in upstream flows. Moreover, adjacent differencing reduces linear dependencies and noise in data (Babovic and Keijzer, 2002). Therefore, model applications were demonstrated with differenced discharge (dQ) data. However, both absolute discharge (Q) data and dQ data were used for the analysis. Comparison will only be made to justify the advantages of dQ data for brevity.

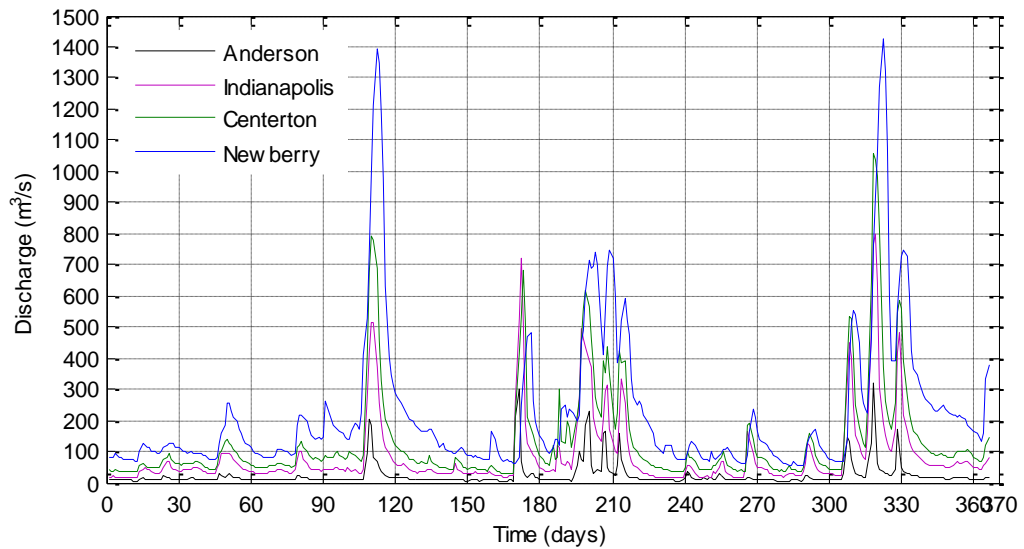


Figure 5.2: Streamflow time series of the year 1992.

Figure 5.3 presents the cross-correlation coefficient and autocorrelation coefficient variation for Q and dQ data. We can observe that the flow travel times of river reaches A-I, I-C, and C-N are approximately 1 day.

Number of autoregressive components in the models were 2, 3, 4, and 5 for A, I, C, and N, respectively. Preceding upstream flows within flow travel times were employed, if upstream stations were considered in the modelling.

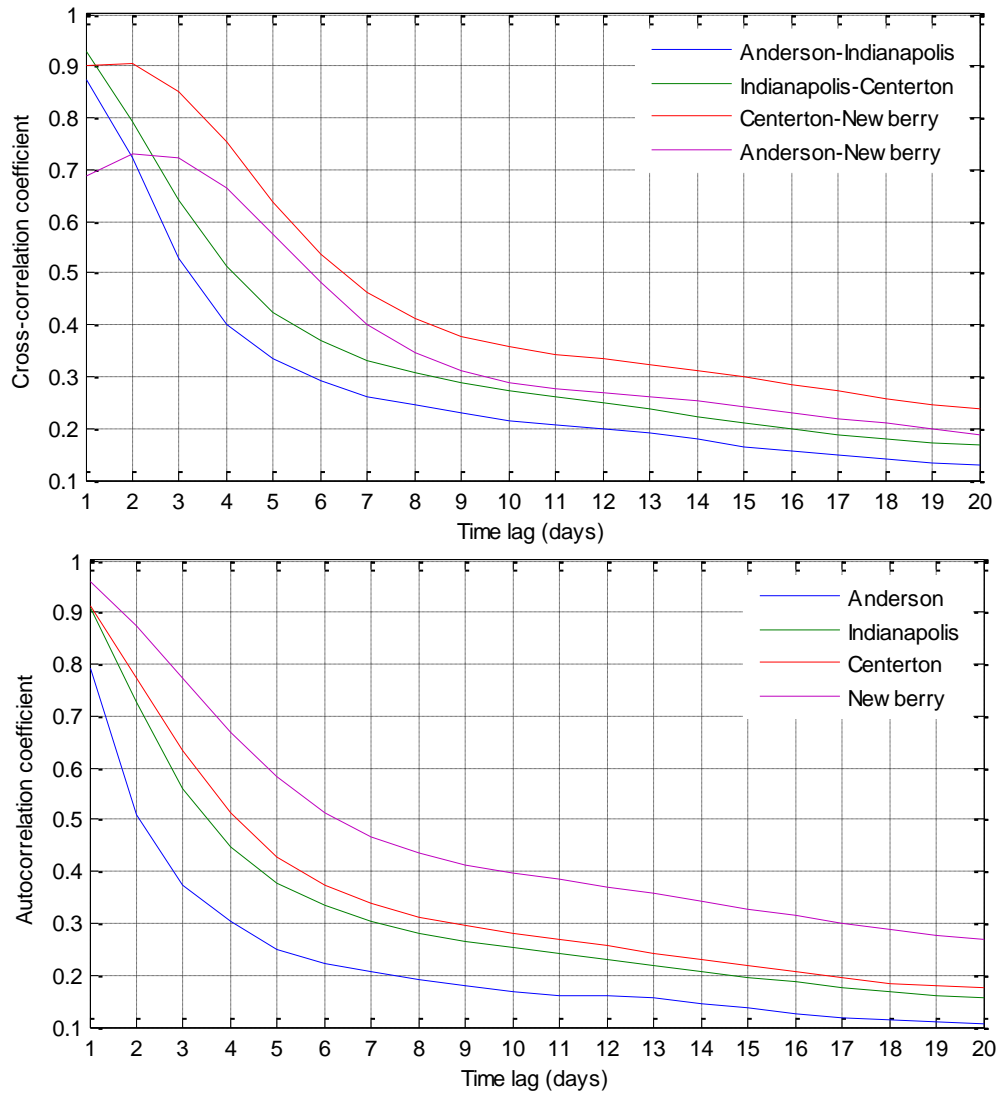


Figure 5.3a: Cross-correlation coefficient and autocorrelation coefficient variation for Q data.

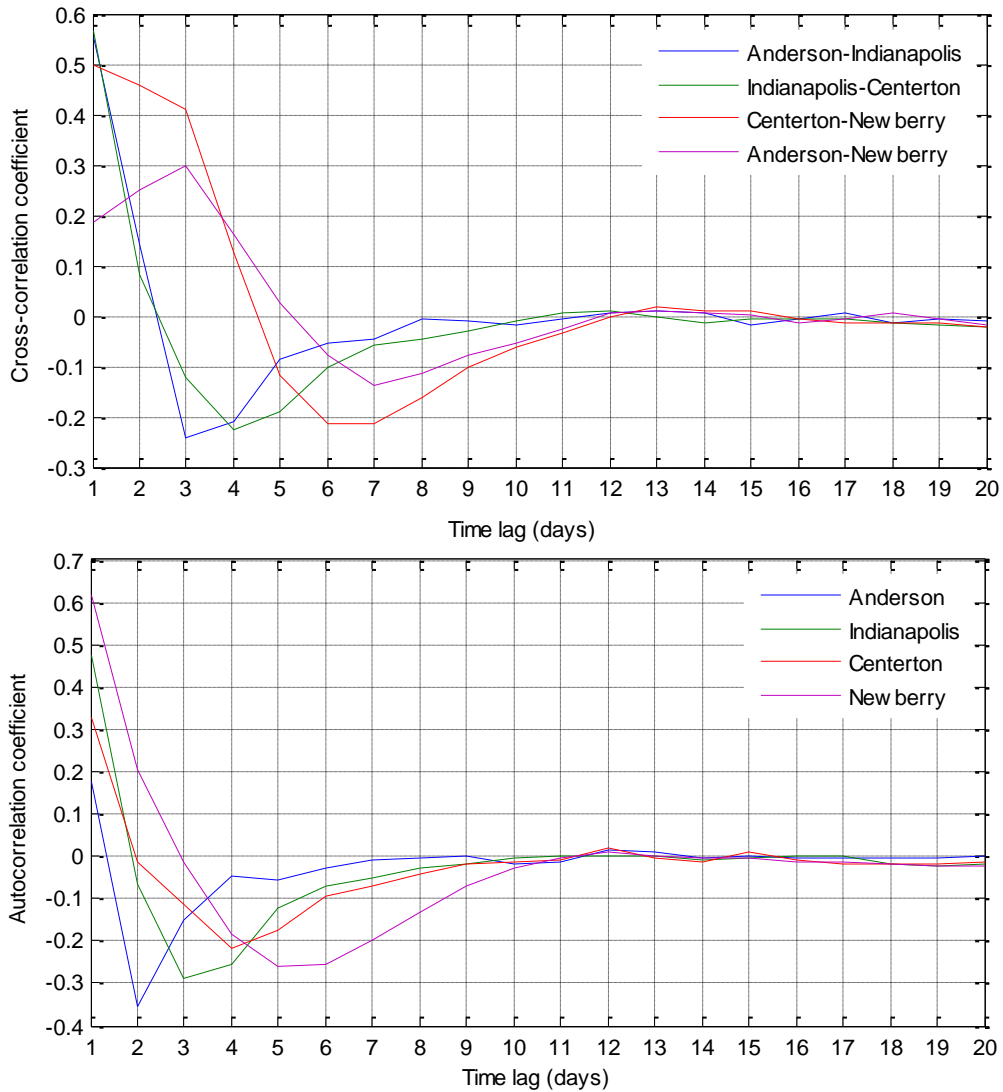


Figure 5.3b: Cross-correlation coefficient and autocorrelation coefficient variation for dQ data.

5.4 Sequential flow routing method

This study also considered three-layered MLPs in developing the ANN models. For each model, 80% of data were used for training and 20% for testing. Optimal model complexity was determined for each situation. Iterative approach was used to compute the forecasts at different forecasting horizons.

First, a sensitivity analysis was performed to find the spatial dependency of time series data. The purpose of this analysis was to find the relevant lag-space, i.e., input data for the time series models. Single-station ANN models were first developed

using the auto-regressive components. Antecedent flow components of the adjacent upstream station were added successively to the model inputs. Figures 5.4 a, b, and c present the MAEs of the models in estimating flows at Newberry, Centerton, and Indianapolis, respectively.

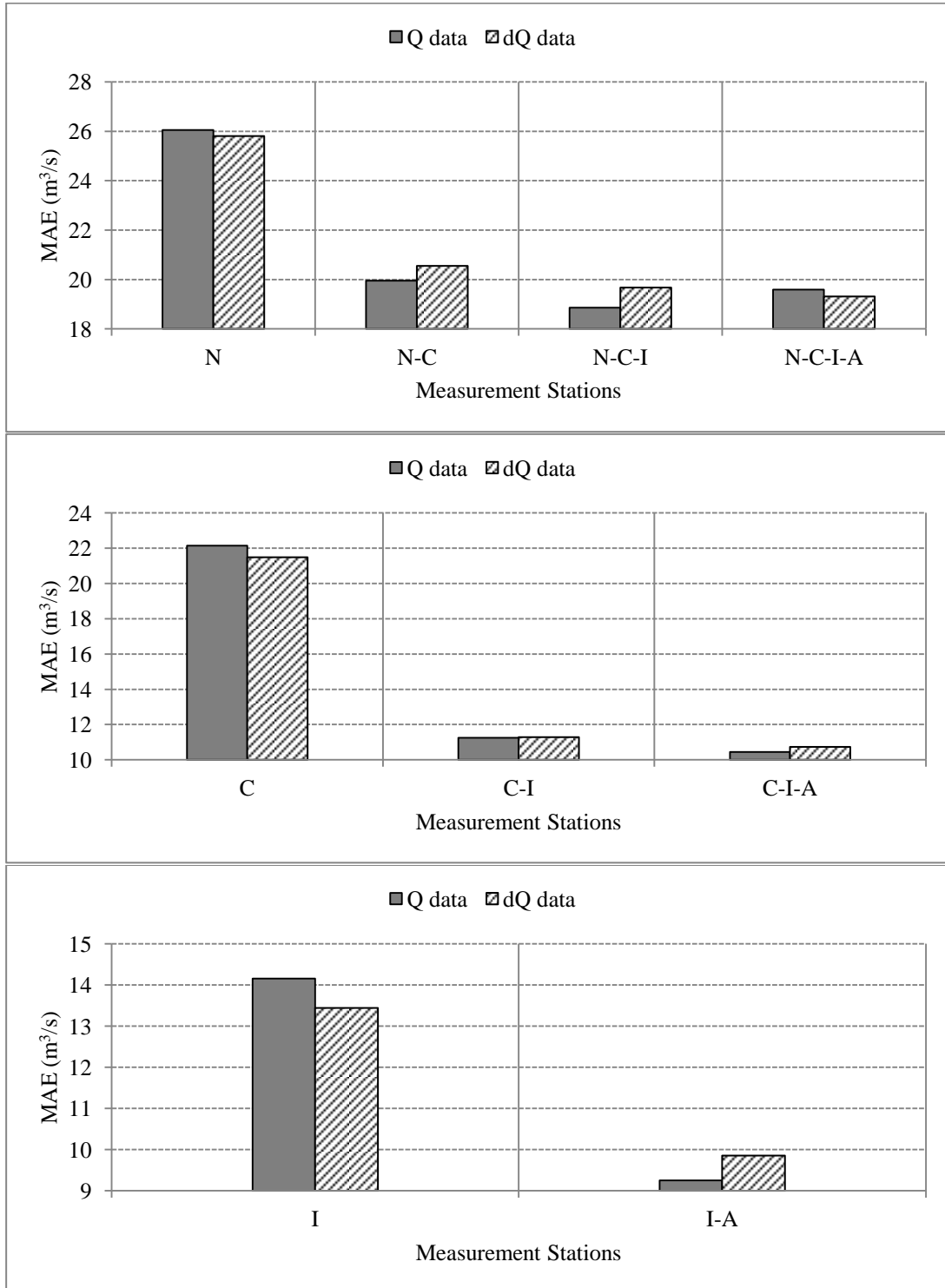


Figure 5.4: Contribution of upstream points on the streamflow estimations at Newberry (N), Centerton (C), and Indianapolis (I).

Results show that inclusion of one upstream location would be sufficient for estimating flows at downstream locations. It can be observed that the spatial dependency gets worse with the distance. This is because more information is given by the nearby flow data. However, flow travel times are higher for the distant upstream stations; thereby provide the means of improving forecasts to a longer time horizon. For this reason, flow routing was performed sequentially from Anderson to Newberry using the method outlined in Figure 5.5.

Anderson		Indianapolis		Centerton		Newberry
$Q_{(t-1)}$		$Q_{(t-1)}$		$Q_{(t-1)}$		$Q_{(t-1)}$
$Q_{(t)}$		$Q_{(t)}$		$Q_{(t)}$		$Q_{(t)}$
$Q_{F(t+1)}$		$Q_{F(t+1)}$		$Q_{F(t+1)}$		$Q_{F(t+1)}$
...		$Q_{F(t+2)}$		$Q_{F(t+2)}$		$Q_{F(t+2)}$
$Q_{F(t+12)}$	→	→	→
	
	
	
	
Assumed forecast with a rainfall-runoff model		$Q_{F(t+12)}$		$Q_{F(t+12)}$		$Q_{F(t+12)}$

Figure 5.5: Streamflow estimation at downstream stations.
 Note: Subscript F denotes forecasted discharge

Two methods were used to estimate the flows at downstream stations. In the first approach, single river reaches were considered. That is streamflow at a downstream location was estimated using the streamflow data of adjacent upstream location and streamflow data of the same location. Q-ANN and dQ-ANN model results of this approach are shown in Figure 5.6. Comparison of model results shows that the accuracy of one-step-ahead forecasts is comparable; however, MAEs of the Q-ANN models were much higher in multi-step-ahead forecasts. This is attributed to the sensitivity of the Q-ANN models for errors.

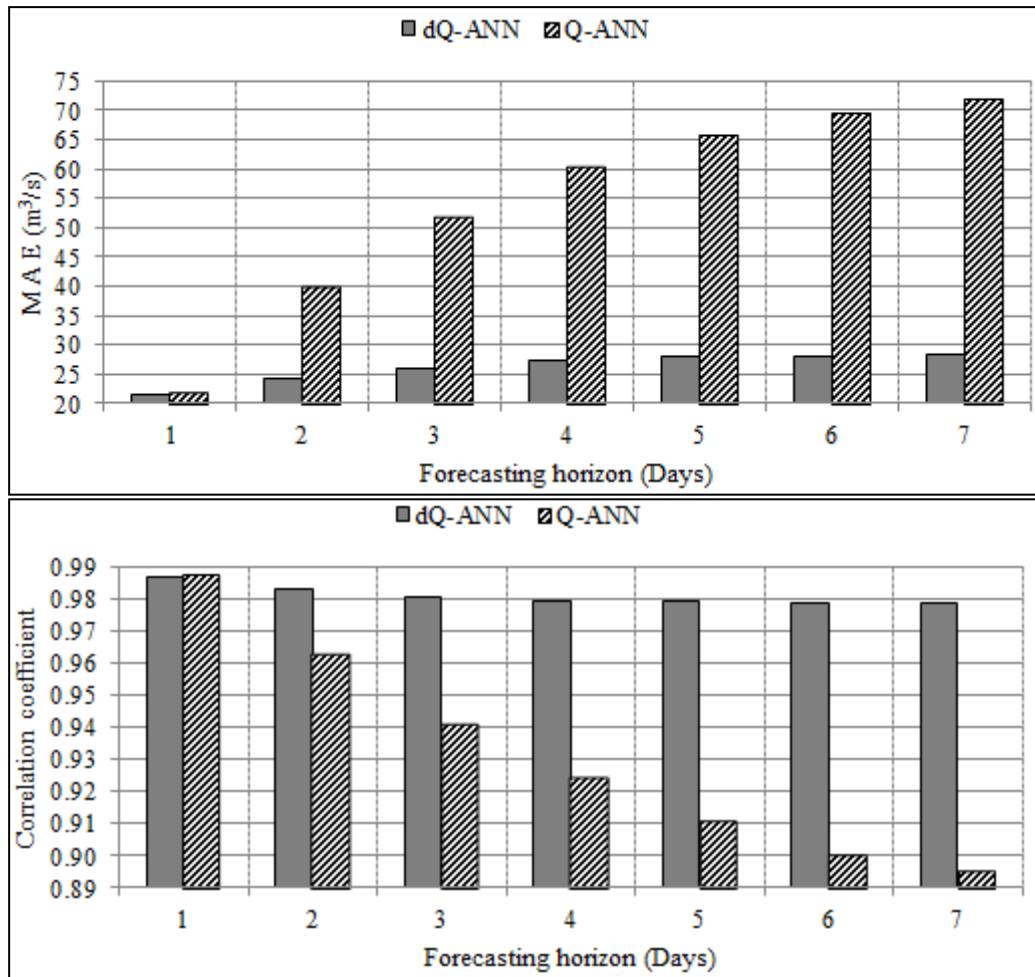


Figure 5.6: Performances of Q-ANN and dQ-ANN models in estimating flows at Newberry.

In the second method, all the upstream points were considered in estimating the future flows at a downstream location. In both methods, a function was approximated for the whole domain, which can be viewed as a global model (GM). As such, the notation GM_{SS} and GM_{MS} will be used to refer global single-station models developed with first and second methods, respectively. Table 5.2 shows the performances of the sequentially applied GM_{SS} and GM_{MS} models developed for Centerton and Newberry.

Table 5.2: Performances of single-station models of Centerton and Newberry.

Mean absolute error (MAE) (m ³ /s)		
Single-station Model	Station	
	Centerton	Newberry
GM _{SS}	14.80	21.68
GM _{MS}	14.09	20.47
Correlation coefficient		
GM _{SS}	0.970	0.986
GM _{MS}	0.973	0.988

Comparison of results indicates that the GM_{MS} model formulation provides slightly better forecasts than the GM_{SS} model formulation. This can be attributed to the limited data and the inadequate process representation. Process scale and the observation scale should be matched for a better process conceptualization (Bloschl and Sivapalan, 1995). If the observation scale is too fine, the model might capture the noise in data. On the other hand, sparse data may not provide process dynamics. In this case study, data were not refined enough, spatially and temporally, to capture the process dynamics. Data time interval is one day, equivalent to the flow travel time of river reaches. Therefore, the original discrete time series was enlarged to demonstrate the effect of data time interval. Similar to the model formulations with daily flow data, models were implemented with 12 hr and 6 hr sampled streamflow data. Table 5.3 presents the difference of MAEs and correlation coefficients of GM_{SS} and GM_{MS} models with data time interval. It can be observed that the difference in MAE and correlation coefficient could be effectively reduced with refined data. This is a result of improved extraction of process dynamics. Similarly, spatially refined data will improve the data driven process approximation. This implies that GM_{SS} models developed with sufficiently refined data will perform well. This finding provides the possible avenues for coupling the hydrologic and hydraulic information.

Table 5.3: Difference of statistical measures of GM_{SS} and GM_{MS} models with data time interval.

Data time interval	MAE of GM _{SS} -MAE of GM _{MS}	
	Station	
	Centerton	Newberry
Daily	0.71	1.21
12 hr	0.45	0.58
6 hr	0.16	0.41
(Corr. coeff. of GM _{SS} -Corr. coeff. of GM _{MS}) x 10 ⁻²		
Daily	0.35	0.17
12 hr	0.11	0.04
6 hr	0.02	0.01

Above models were applied to approximating a function for all streamflow generation instances, i.e., to the global domain. Single ANN model may be biased on the most occurring instances. For example, Sajikumar and Thandaveswara (1999) found that errors are small for the low flows. In the next step, cluster-based flow routing models were developed to enhance the process approximation accuracy.

5.5 Cluster-based flow routing

First step involved the classification of GM_{SS} model input data using SOMs. Figure 5.7 shows the class positions in Centerton flow time series data for 2-class and 4-class classifications. Use of Q classifier inputs mainly identified low flows and high flows. However, we expected to classify the flow data as flow rising and flow recession. This could be achieved with dQ data. Then further classification subdivided those two regions. Similar classification results were obtained for stations, Indianapolis and Newberry (results are not shown).

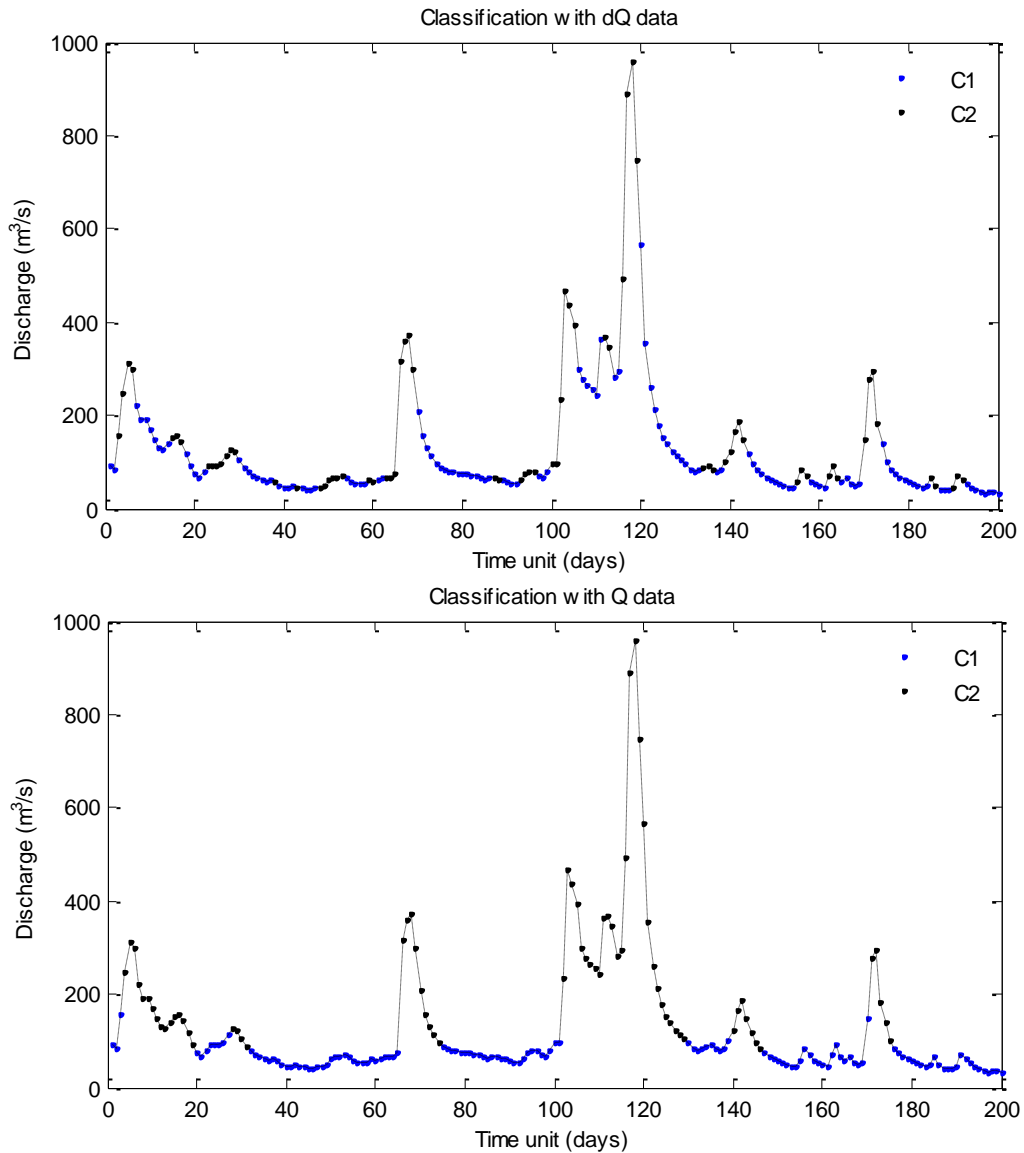


Figure 5.7a: Class positions in Centerton discharge time series for 2-class classification.

Abrahart and See (2000) reported that dQ model inputs added little to the clustering process. It is to be noted that the upstream flow data were not included in their approach, which might limit the identification of functionally different regions. Studies that applied Q data for classification, achieved reasonable differentiation with large number of classes (Abrahart and See, 2000; See and Openshaw, 1999). This approach required manual grouping of classes and another classifier to identify those manually grouped classes. Successful identification of temporally varying regions with dQ data will eliminate this intermediate step.

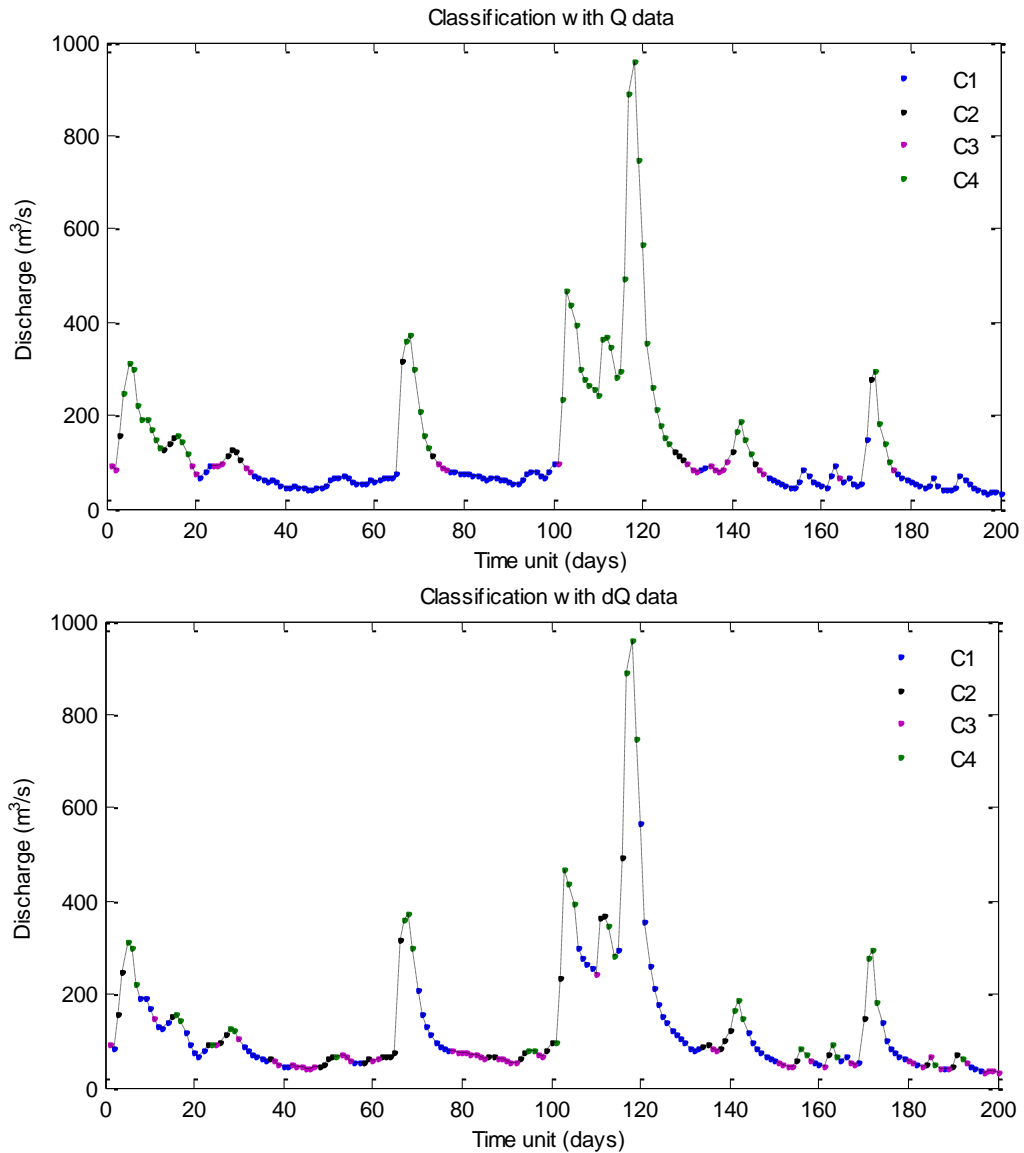


Figure 5.7b: Class positions in Centerton discharge time series for 4-class classification.

In the second step, a function was approximated with ANN for each data cluster. In this way, for a particular forecasting instance, classifier determines the data cluster and ANN model associated to that data cluster provides the streamflow forecast. As in global model formulation, single-station modular neural network models (MNNs) were first developed and those were sequentially applied to estimate the flows at each station. Figure 5.8 shows the global model and modular model performances. Final symbolization in MNN model is to identify the number of local models.

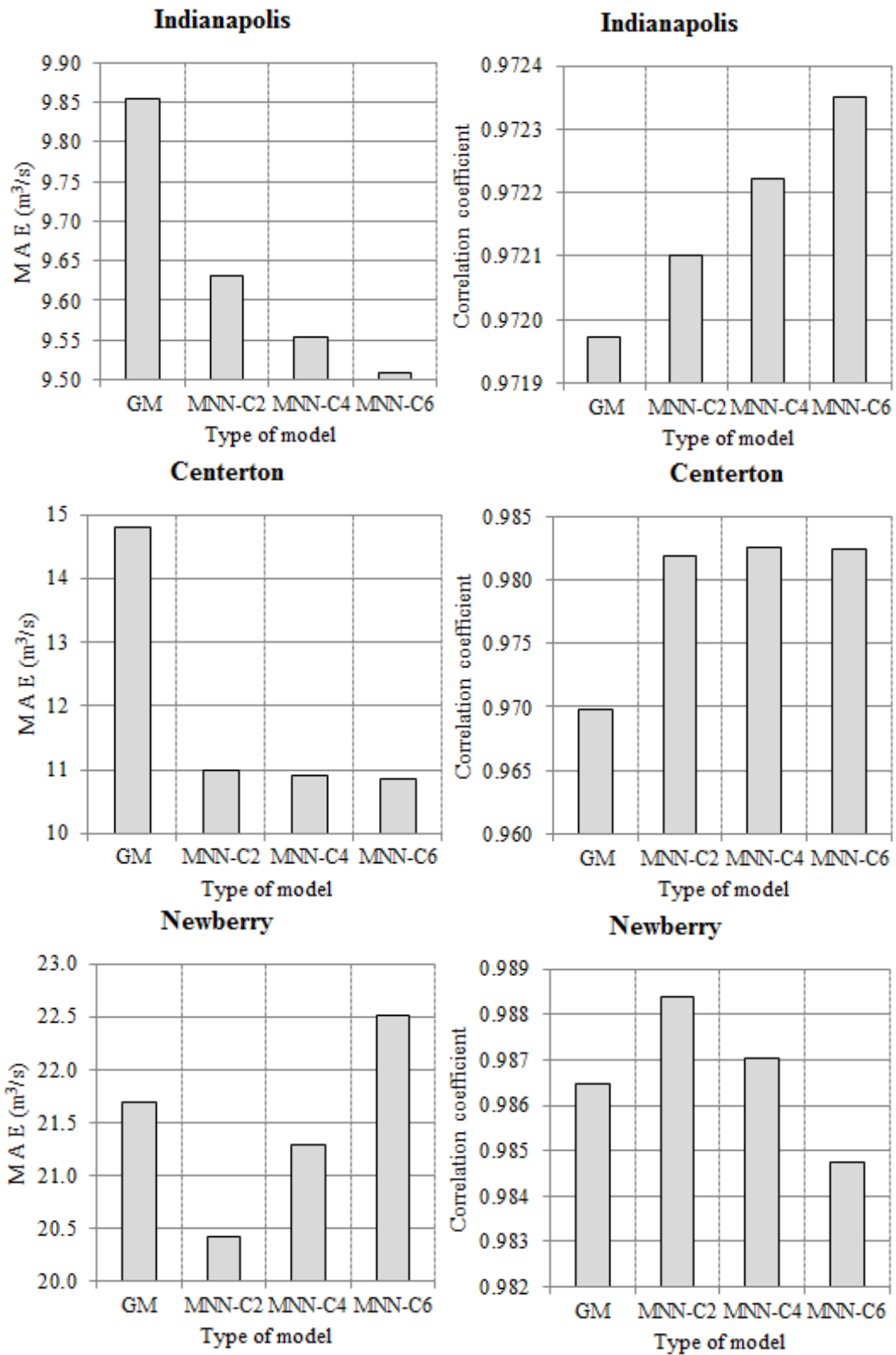


Figure 5.8: Performances of global model (GM) and modular neural network (MNN) models at Indianapolis (I), Centerton (C), and Newberry (N).

At all stations, MNNs with two local modes (MNN-C2) provided lower MAEs than the corresponding global models. The improvement in MAEs for the 2-classes

were 0.120 m³/s, 0.346 m³/s for Indianapolis; 0.694 m³/s,-0.093 m³/s for Centerton; and 0.321 m³/s, 0.367 m³/s for Newberry station. At Indianapolis, further increase in number of classes results in decrease in MAE. Comparison of local model performance and global model performance in local domains shows that local models improved the forecasts accuracy except one local model in 4-class classification and two local models in 6-class classification. Class positions indicate that those represent baseflow and rising flow. On the other hand, significant improvement in MAEs was not observed with number of classes at Centerton. At further downstream location, Newberry, increase in MAE is observed. MAEs of local models were higher than the global model MAEs in corresponding local domains. The possible reasons for this can be explained as follows. Two-class classification mainly identified the rising limb and falling limb, while 4-class and 6-class classifications subdivided the rising and falling limb into two or more classes. A wave is generally subjected to translation and attenuation conserving the volume of flow. However, streamflow at a particular point includes flow from the upstream as well as that from the intermediate area. In this study, contributions of rainfall and lateral flows were not considered due to the lack of data. As a result, local models might not improve the approximation. It was also observed that few numbers of data were available for the high flows (Figure 5.7). Moreover, accuracy of flow forecasts at downstream locations depends on the accuracy of upstream flow estimations.

5.5 Conclusions

This study applied MLP neural networks for estimating the future flows at multiple stations of White river, Indiana. Single-station models were first developed using the upstream streamflow data. Single-station models were also implemented

with all available upstream data. These models were sequentially applied to find the streamflows at downstream locations. It was found that single river reach models performed well for sufficiently refined data. The study was extended to examine the applicability of cluster-based modelling for distributed flow routing. The modelling was not entirely successful. Data were not refined enough, spatially and temporally, to capture the variations. Further, contribution of rainfall in generating runoff was not included. Therefore, performance of the distributed cluster-based flow routing method can be further improved by coupling the hydraulic and hydrologic information. The findings and research basis of this study will provide the possible avenues for extending the distributed cluster-based modelling.

CHAPTER 6 CONCLUSIONS AND RECOMMENDATIONS

Data driven rainfall-runoff (R-R) process models in their current form provide fair results despite their practical significance. This study has identified the means of extracting relevant information from data and thereby to improve the prediction accuracy of data driven models (DDMs).

Lump catchment models were first developed to study the effect of data time interval on streamflow estimation using 1 hr, 2 hr, and 3 hr sampled rainfall and runoff data of the Orgeval catchment, France. Two analyses were performed using absolute discharge (Q) data and differenced discharge (dQ) data. Forecasts were iteratively computed at different time horizons, 2 hr ahead to 12 hr ahead. It was found that the fine sampled data improved the streamflow estimation and results were comparable in both analyses. However, significantly higher MAEs were observed in multi-step-ahead forecasts for Q data than for dQ data. This is because sensitivity of the Q -models is high, which results higher errors at subsequent iterative steps. An important feature of the dQ -models is that a significant increase in error was not observed even after the lead-time greater than the catchment concentration time. Error accumulation property was found to have significant impact on the multi-step-ahead forecasts' accuracy, which made the prediction improvement with refined data, unresponsive in Q - models. These results provide valuable information on the multi-step-ahead forecasts' accuracy, since those indicate that in addition to the improvement in streamflow estimation, i.e., accuracy of one-step-ahead forecasts, error accumulation property of the model is an important factor. Due to the fact that accumulated error in iterative forecasting is significant, direct forecasting approach was employed to compute the multi-step-ahead forecasts. It was found that direct forecasts were slightly better than the iterative

forecasts, when forecasting horizon is less than the catchment concentration time. This is expected, because direct forecasting uses only past information. This study was not able to describe the effect of noise due to the fact that fine sampled data were not available. Further research is therefore needed to evaluate the effect of noise and its removal with data time interval.

This study also examined the possibility of identifying temporally dominant processes of the lump catchment concept by classifying the antecedent conditions, i.e., model inputs. The number of classes varied from 2 to 8. For each situation, modular model was developed to compare the accuracy of forecasts. Local domain for a forecasting instance was found with the SOM classifier and the inputs were presented to corresponding local domain model to produce the final model output. The analysis was first performed on rainfall and Q model inputs. The classification results showed that the change in discharge could not be successfully identified with the Q data. Consequently, increase in number of classes did not result any improvement in predictability. Secondly, the same procedure was applied for rainfall and dQ model inputs. It was shown that the use of dQ data effectively identified the different parts of the discharge time series. Modular models also performed well compared to global model. Improvement in model representation also has an effect on identifying nonlinear dynamics of the process. To investigate this, performances of modular ANN models were compared with linear modular model results. Linear models did not perform well in all local domains. This is because of the different complexities associated with each local domain. As a result, local linear model errors were much higher compared to ANN models. However, the overall improvement in predictability with nonlinear models depends on the complexity of the R-R process. Application of

modular model approach for catchments with different complexities will be an interesting research topic.

It was also found that dQ-models have the greatest tendency to yield lower errors for out-of-range data compared to Q-models. In case of modular models, slightly higher errors were observed. This effect is unavoidable due the fact that approximating a function to a particular data range tends to produce higher errors for out-of-range data.

Lump catchment concept is not valid for large-scale catchments and urban catchments. It can be extended to capture the spatial variation of hydrological processes. This research demonstrated a sequential data driven approach for flow routing, which can be used in distributed R-R process models. Use of upstream information to predict flows at downstream could improve the forecasts to a possibly longer horizon. In the second part of this study, cluster-based modelling was applied to improve the flow estimations. Simulation results of this analysis indicated that it is a promising method to improve the streamflow forecasts. Inclusion of contribution of rainfall will improve the predictive capability further.

The results of this research suggested that estimation errors could be effectively reduced by more detailed representation of the R-R process. This research will provide a basis for subsequent studies on data driven R-R models and for other relevant data driven applications.

References

1. Abrahart, R. J. and L. See. Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. *Hydrological Processes*, *14*, pp.2157- 2172. 2000.
2. Acar, E. and M. Rais-Rohani. Ensemble of metamodels with optimized weight factors, *Structural and Multidisciplinary Optimization*, *37(3)*, pp.279-294. 2009.
3. Anctil, F., Filion, M. and J. Tournebize. A neural network experiment on the simulation of daily nitrate-nitrogen and suspended sediment fluxes from a small agricultural catchment, *Ecological Modelling*, *220(6)*, pp.879-887. 2009.
4. Anderson, M. G. and Burt, T. P. (ed.). *Hydrological forecasting*, New York: John Wiley & Sons.1985.
5. ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. Artificial neural networks in hydrology I: Preliminary concepts, *Journal of Hydrologic Engineering*, *5(2)*, pp.115-123. 2000a.
6. ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. Artificial neural networks in hydrology II: Hydrologic applications, *Journal of Hydrologic Engineering*, *5(2)*, pp.124-137. 2000b.
7. Babovic, V. A data mining approach to time series modeling and forecasting. *Hydroinformatics' 98*, pp. 847-856. Babovic & Larsen (eds). 1998.
8. Babovic, V. and M. B. Abbott. The evolution of equations from hydraulic data Part I: Theory, *Journal of hydraulic research*, *35 (3)*, pp.397-410. 1997a.
9. Babovic, V. and M. Keijzer. Rainfall-runoff modelling based on genetic programming, *Nordic Hydrology*, *33(5)*, pp.331-346. 2002.
10. Babovic, V. Data mining in hydrology, *Hydrological processes*, *19 (7)*, pp.1511-1515. 2005.
11. Babovic, V., Canizares, R., Jensen, H. R. and A. Klinting. Neural networks as routine for error updating of numerical models, *Journal of Hydraulic Engineering*, *127(3)*, pp.181-193. 2001.
12. Babovic, V. and M. B. Abbott. The evolution of equations from hydraulic data Part I: Applications, *Journal of Hydraulic Research*, *35 (3)*, pp.411-430. 1997b.
13. Baker, L. and D. Ellison. The wisdom of crowds-ensembles and modules in environmental modeling, *Geoderma*, *147(1-2)*, pp.1-7. 2008.
14. Blöschl, G. and M. Sivapalan. Scale issues in hydrological modelling: A review, *Hydrological Processes*, *9(3-4)*, pp.251-290. 1995.

15. Bowden, G. J., Maier, H. R. and G. C. Dandy. Input determination for neural network models in water resources applications. Part 2. Case study: Forecasting salinity in a river, *Journal of Hydrology*, *301(1-4)*, pp.93-107. 2005.
16. Breiman, L. Bias-variance, Regularization, Instability and Stabilization. In *Neural networks and machine learning*, ed by C.M. Bishop, pp.27-56, London: Springer. 1998.
17. Budyko, M. I. *Climate and life*. New York: Academic Press. 1974.
18. Butts, M. B., Madsen, J. H. and J. C. Refsgaard. Hydrologic forecasting, *Encyclopedia of Physical Science and Technology*. pp.547-566. 2004.
19. Chen, J. and B. J. Adams. Integration of artificial neural networks with conceptual models in rainfall-runoff modelling, *Journal of Hydrology*, *318(1-4)*, pp.232-249. 2006.
20. Clark, M. P., D. E. Rupp., R. A. Woods., H. J. T. Meerveld., Peters, N. E. and J. E. Freer. Consistency between hydrological models and field observations: Linking Processes at the hillslope scale to hydrological responses at the watershed scale, *Hydrological Processes*, *23(2)*, pp.311-319. 2009.
21. Cleveland, W. S. Robust locally weighted regression and smoothing scatterplots, *Journal of the American Statistical Association*, *74 (368)*, pp.829-836. 1979.
22. Corzo, G. A. and D. P. Solomatine. Baseflow separation techniques for modular artificial network modeling in flow forecasting, *Hydrological Sciences Journal*, *52(3)*, pp.491-507. 2007.
23. Corzo, G. A. Solomatine, D. P., Hidayat, De Wit, M., Werner, M., Uhlenbrook, S. and R. K. Price. Combining semi-distributed process-based and data driven models in flow simulation: A case study of the Meuse river basin, *Hydrology and Earth System Sciences*, *13(9)*, pp.1619-1634. 2009.
24. Diamantopoulou, M. J., Georgiou, P. E. and D. M. Papamichail. A time delay artificial neural network approach for flow routing in a river system, *Hydrology and Earth System Sciences*, *3(5)*, pp.2735-2756. 2006.
25. Díaz-Robles, L. A., Ortega, J. C., Fu, J. S., Reed, G. D., Chow, J. C., Watson, J. G. and J. A. Moncada-Herrera. A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: The case of Temuco, Chile, *Atmospheric Environment*, *42 (35)*, pp.8331-8340. 2008.
26. Diks, C. G. H. and J. A. Vrugt. Comparison of point forecast accuracy of model averaging methods in hydrologic applications, *Stochastic Environmental Research Risk Assessment*, *24(6)*, pp.809-820. 2010.
27. Dunne, T. Relation of field studies and modelling in the prediction of storm runoff, *Journal of Hydrology*, *65(1-3)*, pp.25-48. 1983.

28. Eckhardt, K. How to construct recursive digital filters for baseflow separation. *Hydrologic Processes*, 19(2), pp.507-515. 2005
29. Elshorbagy, A., Simonovic, S. P. and U.S. Panu. Noise reduction in chaotic hydrologic time series: Facts and doubts, *Journal of Hydrology*, 256(3-4), pp.147-165. 2002.
30. Elshorbagy, A., Simonovic, S. P. and U.S. Panu. Performance evaluation of artificial neural networks for runoff prediction, *Journal of Hydrologic Engineering*, 5(4), pp.424-427. 2000.
31. Furundzic, D. Application example of neural networks for time series analysis: Rainfall-runoff modelling, *Signal Processing*, 64(3), pp.383-396. 1998.
32. Gaume, E. and R. Gosset. Over-parameterization, a major obstacle to the use of artificial neural networks in hydrology, *Journal of Hydrology and Earth System Sciences*, 7(5), pp.693-706. 2003.
33. Han, D., Kwong, T. and S. Li. Uncertainties in real-time flood forecasting with neural networks, *Journal of Hydrological Processes*, 21(2), pp.223-228. 2007.
34. Harman, C. J., Sivapalan, M. and P. Kumar. Power law catchment-scale recessions arising from heterogeneous linear small-scale dynamics, *Water Resources Research*, 45(9), art.no. W09404. 2009.
35. Hashim, S. Optimal linear combinations of neural networks, *Neural Networks*, 10(4), pp.599-614. 1997.
36. Haykin, S. *Neural networks: A comprehensive foundation*. In Prentice Hall, New Jersey. 1999.
37. Hettiarachchi, P., Hall, M. J. and A.W. Minns. The extrapolation of artificial neural networks for the modelling of rainfall-runoff relationships, *Journal of Hydroinformatics*, 7(4), pp.291-295. 2005.
38. Horton, R. E. Erosional development of streams and their drainage basins: hydrophysical approach to quantitative morphology, *Bulletin of the Geological Society of America*, 56, pp.275-370. 1945.
39. Hsu, K., Gupta, H. V. and S. Sorooshian. Artificial neural network modeling of the rainfall-runoff process, *Water Resources Research*, 31(10), pp.2517-2530. 1995.
40. Hu, T., Wu, F., and X. Zhang. Rainfall-runoff modeling using principal component analysis and neural network, *Nordic Hydrology*, 38(3), pp.35-248. 2007.
41. Jacobs, R. A. and M. I. Jordan. Learning piecewise control strategies in a modular neural network architecture, *IEEE Transactions on Systems, Man, and Cybernetics*, 23(2), pp.337-345. 1993.

42. Jain, A. and A. M. Kumar. Hybrid neural network models for hydrologic time series forecasting, *Applied Soft Computing*, 7(2), pp.585-592. 2007.
43. Jayawardena, A. W. and A. B. Gurung. Noise reduction and prediction of hydro-meteorological time series: dynamical systems approach vs. stochastic approach, *Journal of Hydrology*, 228(3-4), pp.242-264. 2000.
44. Karunanithi, N., Grenney, W. J., Whitely, D. and K. Bovee. Neural Networks for river flow predictions, *Journal of Computing in Civil Engineering–ASCE*, 8(2), pp.201-220. 1994.
45. Karunasinghe, D. S. K. and S. Y. Liong. Chaotic time series prediction with a global model: Artificial neural network, *Journal of Hydrology*, 323(1-4), pp.92-105. 2006.
46. Khashei, M. and M. Bijari. A novel hybridization of artificial neural networks and ARIMA models for time series forecasting, *Applied Soft Computing Journal*, 11(2), pp.2664-2675. 2011.
47. Khatibi, R., Ghorbani, M. A., Kashani, M. H. and O. Kisi. Comparison of three artificial intelligence techniques for discharge routing, *Journal of Hydrology*, 403(3-4), pp.201-212. 2011.
48. Khondker, M. U.H., Wilson, G. and A. Klinting. Application of neural networks in real time flood forecasting. *Hydroinformatics'98*, ed by Babovic & Larsen (eds). pp.777-781, 1988.
49. Kim, Y.-O., Jeong, D. and I.H. Ko. Combining rainfall-runoff model outputs for improving ensemble streamflow prediction, *Journal of Hydrologic Engineering* 11(6), pp.578-588. 2006.
50. Kirkby, M. Hydrograph modeling strategies. In *Processes in physical and human geography*, ed by R. Peel., Chisholm, M. and Haggett, P. London: Heinemann, pp69-90. 1975.
51. Kisi, O. River flow forecasting and estimation using different artificial neural network techniques, *Hydrology Research*, 39(1), pp.27-40. 2008.
52. Klemes, V. Conceptualization and scale in hydrology, *Journal of Hydrology*, 65(1-3), pp.1-23. 1983.
53. Kohonen, T. Self-organized formation of topologically correct feature maps, *Biological Cybernetics*, 43(1), pp.59-69. 1982.
54. Lekkas, D. F., Imrie, C. E. and M. J. Lees. Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting, *Journal of Hydroinformatics*, 3(3), pp.153-164. 2001.

55. Liong, S. Y., Gautam, T. R., Khu, S. T., Babovic, V., Keijzer, M. and Muttill, N. Genetic programming: A new paradigm in rainfall runoff modelling, *Journal of the American Water Resources Association*, 38 (3), pp. 705-718. 2002.
56. Liong, S. Y. and C. Sivapragasam. Flood stage forecasting with support vector machines, *Journal of the American Water Resources Association*, 38(1), pp. 173-186. 2002.
57. Liong, S.Y., Lim, W. H. and G. N. Paudyal. River stage forecasting in Bangladesh: Neural network approach, *Journal of Computing and Civil Engineering*, 14(1), pp.1-8. 2000.
58. Liu, Y. and H. V. Gupta. Uncertainty in hydrologic modeling: Towards an integrated data assimilation framework, *Water Resources Research*, 43(7), W07401, doi:10.1029/2006WR005756. 2007.
59. Maidment, D. R. *Handbook of hydrology*, New York: McGraw-Hill, 1993.
60. Maier, H. R. and Dandy, G. C. Determining Input for Neural Network Models of Multivariate Time Series, *Microcomputers in Civil Engineering*, 12(5) , pp.353-368. 1997.
61. Masters, T. *Novel and Hybrid Algorithms for Time Series Prediction*. In *Neural*. 1995.
62. Mays, L. W. *Water resources engineering*, USA: John Wiley & Sons.2005.
63. McCuen, R. H. *Hydrologic analysis and design*, In Prentice-Hall, Inc., USA: New Jersey. 1998.
64. McCulloch, W. S. and W. Pitts. A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, 5, 115-133. 1943.
65. Minns, A. W. and M. J. Hall. Artificial neural networks as rainfall-runoff models, *Hydrological Sciences-Journal*, 41(3), pp. 399-417. 1996.
66. Nelles, O. *Nonlinear System Identification*. Berlin: Springer- Verlag. 2001.
67. Nourani, V. and O. Kalantari. Integrated artificial neural network for spatiotemporal modeling of rainfall-runoff-sediment processes. *Environmental Engineering Science*, 27(5), pp.411-422. 2010.
68. O'Donnell, T. A direct three parameter Muskingum procedure incorporating lateral inflow, *Hydrological Sciences Journal*, 30 (4), pp.479-496. 1985.
69. Parasuraman, K. and A. Elshorbagy. Cluster-based hydrologic prediction using genetic algorithm-trained neural networks. *Journal of Hydrologic Engineering*, 12(1), pp.52-62. 2007.

70. Parasuraman, K., Elshorbagy, A. and S. K. Carey. Spiking modular neural networks: A neural network modeling approach for hydrological processes. *Water Resources Research* 42, w05412. 2006.
71. Porporato, A. and L. Ridolfi. Nonlinear analysis of river flow time sequences, *Water Resources Research*, 33(6), pp.1353-1367. 1997.
72. Proano, C. O., Minns, A. W., Verwey, A. and H. F. Van den Boogaard. Emulation of a sewerage system computational model for the statistical processing of large numbers of simulations. *Proceedings of the 3rd International Conference on Hydroinformatics*, 1998, pp.1145- 1152.
73. Remesan, R., Ahmadi, A., Shamim, M. A. and D. Han. Effect of data time interval on real-time flood forecasting, *Journal of Hydroinformatics*, 12(4), pp.396-407. 2010.
74. Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65, 386-408. 1958.
75. Rumelhart, D. E., Hinton, G. E. and R. J. Williams. Learning representations by back-propagating errors, *Nature*, 323(6088), pp.533 – 536, doi:10.1038/323533a0, 1986.
76. Sajikumar, N. and B. S. Thandaveswara. A non-linear rainfall-runoff model using as artificial neural network, *Journal of Hydrology*, 216(1-2), pp.32-55. 1999.
77. Sallehuddin, R. and S. M. Hj. Shamsuddin. Hybrid grey relational artificial neural network and auto regressive integrated moving average model for forecasting time-series data, *Applied Artificial Intelligence*, 23 (5), pp.443-486. 2009.
78. See, L. and S. Openshaw. Applying soft computing approaches to river level forecasting, *Hydrological Sciences Journal*, 44(5), pp.763-778. 1999.
79. Shamseldin, A. Y. and K. M. O'Connor. A non-linear neural network technique for updating of river flow forecasts, *Hydrological Earth Science Systems*, 5(4), pp. 577-597. 2001.
80. Shamseldin, A. Y. Application of a neural network technique to rainfall-runoff modeling, *Journal of Hydrology*, 199(3-4), pp.272-294. 1997.
81. Sharkey, A. J. C. Combining artificial neural nets: ensemble and modular multi-net systems. London: Springer. 1999.
82. Singh, V. P. and R. C. McCann. Some notes on Muskingum method of flood routing, *Journal of Hydrology*, 48(3-4), pp.343-361. 1980.
83. Sivakumar, B., Phoon, K. K., Liong, S. Y., and C. Y. Liaw. A systematic approach to noise reduction in chaotic hydrological time series, *Journal of Hydrology*, 219(3-4), pp.103-135. 1999.

84. Sivapalan, M., Blöschl, G., Zhang, L. and R. Vertessy. Downward approach to hydrological prediction, *Hydrological Processes*, 17(11), pp.2101-2111. 2003.
85. Sivapragasam, C. and S. Y. Liong. Flow categorization model for improving forecasting, *Nordic Hydrology*, 36(1), pp.37-48. 2005.
86. Solomatine, D. P. and A. Ostfeld. Data-driven modeling: some past experiences and new approaches, *Journal of Hydroinformatics*, 10(1), pp.3-22. 2008.
87. Solomatine, D. P. and R. K. Price. Innovative approaches to flood forecasting using data driven and hybrid modelling. In Proc. 6th International Conference on Hydroinformatics, June 2004, Singapore, pp. 21-24.
88. Solomatine, D. P., Maskey, M. and D. L. Shrestha. Instance-based learning compared to other data driven methods in hydrological forecasting, *Hydrological Processes*, 22(2), pp.275-287. 2007.
89. Strahler, A. N. Quantitative analysis of watershed geomorphology. *Transactions of the American Geophysical Union*, 38(6), pp.913-920. 1957.
90. Tallaksen, L. M. A review of baseflow recession analysis, *Journal of Hydrology* 165(1-4), pp.349-370. 1995.
91. Thirumalaiah, K. and M. C. Deo. Hydrological forecasting using neural networks, *Journal of Hydrologic Engineering*, 5(2), pp.180-189. 2000.
92. Toth, E. Classification of hydro-meteorological conditions and multiple artificial neural networks for streamflow forecasting, *Hydrological Earth System Sciences*, 13(9), pp.1555-1566. 2009.
93. Van den Boogaard, H. F., Gautam D. K., and A. E. Mynett. Auto-regressive neural networks for the modeling of time series. *Hydroinformatics '98*, pp.741-748. 1998.
94. Wagener, T., Sivapalan, M., Troch, P. and Woods, R. Catchment classification and hydrologic similarity, *Geography Compass*, 1(4), pp.901-931. 2007.
95. Wang, W., Van Gelder, P. H. A. J. M., Vrijling, J. K. and J. Ma. Forecasting daily streamflow using hybrid ANN models. *Journal of Hydrology*, 324, pp.383-399. 2006.
96. Wu, J. S., Han, J., Annambhotla, S. and S. Bryant. Artificial neural networks for forecasting watershed runoff and stream flows. *Journal of Hydrologic Engineering* 10(1), pp.85-88. 2005.
97. Wu, S. -J., Ho L. -F. and J. -C. Yang. Application of modified nonlinear storage function on runoff estimation. *Journal of Hydro-environment Research*, 5, pp.37-47. 2011.

98. Xiang, C., Ding, S. Q., and T. H. Lee. Geometrical interpretation and architecture selection of MLP, *IEEE Transactions on Neural Networks*, *16(1)*, pp.84-96. 2005.
99. Yu, X., Liong, S. Y. and V. Babovic, EC-SVM approach for real-time hydrologic forecasting, *Journal of Hydroinformatics*, *6 (3)*, pp.209-223. 2004.
100. Zhang, B. and R. S. Govindaraju. Prediction of watershed runoff using Bayesian concepts and modular neural networks, *Water Resources Research*, *36(3)*, pp.753-762. 2000.
101. Zhang, G. P. Time series forecasting using a hybrid ARIMA and neural network model, *Neurocomputing*, *50*, pp.159–175. 2003.

LIST OF PUBLICATIONS

International Journals

1. Basnayake, L.A. & V. Babovic. Rainfall-runoff modelling with data driven techniques: Constraints and proper implementation. IAHS Red Book Series 357. Floods-From Risks to Opportunity, pp. 273-282. 2013.
2. Basnayake, L.A. & V. Babovic. Flow routing with data driven techniques. Submitted for the possible publication in the Journal of Hydro-Environmental Research.
3. Basnayake, L.A. & V. Babovic. Modular data driven approach for rainfall-runoff modelling (in preparation).

International Conferences

1. Basnayake, L.A., Raghuraj, R. & V. Babovic. Water quality model emulation with artificial neural networks, 9th International Conference on Hydroinformatics (HIC 2010), Tianjin, China. pp.991-999.
2. Basnayake, L.A. & V. Babovic. Rainfall-runoff modelling with data driven techniques: Constraints and proper implementation (Abstract), 5th International Conference on Flood Management, Sep. 2011, Tokyo, Japan.
3. Basnayake, L.A. & V. Babovic. Integration of domain knowledge and analytical techniques for improving rainfall-runoff modelling, 18th Congress of the Asia and Pacific Division of the International Association for Hydro-Environment Engineering and Research (IAHR-APD), Aug. 2012, Jeju, Korea.
4. Basnayake, L.A. & V. Babovic. Flow routing with data driven techniques, 18th Congress of the Asia and Pacific Division of the International Association for Hydro-Environment Engineering and Research (IAHR-APD), Aug. 2012, Jeju, Korea.