REMOTE SENSING WITH COMPUTATIONAL INTELLIGENCE MODELLING FOR
MONITORING THE ECOSYSTEM EVOLUTION IN A CONSTRUCTED WETLAND

by

GOLAM MOHIUDDIN
B.S. Bangladesh University of Engineering & Technology, Bangladesh, 2009

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Civil, Environmental, and Construction Engineering
in the College of Engineering and Computer Sciences
at the University of Central Florida
Orlando, Florida

Fall Term
2014

Major Professor: Ni-Bin Chang

ABSTRACT

Monitoring the heterogeneous aquatic environment such as the Stormwater Treatment Areas (STAs) located at the northeast of the Everglades is extremely important in understanding the land processes of the constructed wetland in its capacity to remove nutrient. Direct monitoring and measurements of ecosystem evolution and changing velocities at every single part of the STA are not always feasible. Integrated remote sensing, monitoring, and modeling technique can be a state-of-the-art tool to estimate the spatial and temporal distributions of flow velocity regimes and ecological functioning in such dynamic aquatic environments. In this presentation, comparison between four computational intelligence models including Extreme Learning Machine (ELM), Genetic Programming (GP) and Artificial Neural Network (ANN) models were organized to holistically assess the flow velocity and direction as well as ecosystem states within a vegetative wetland area. First the local sensor network was established using Acoustic Doppler Velocimeter (ADV). Utilizing the local sensor data along with the help of external driving forces parameters, trained models of ELM, GP and ANN were developed, calibrated, validated, and compared to select the best computational capacity of velocity prediction over time. Besides, seasonal images collected by French satellite Pleiades have been analyzed to address the seasonality effect of plant species evolution and biomass changes in the constructed wetland. The key finding of this research is to characterize the interactions between geophysical and geochemical processes in this wetland system based on ground-based monitoring sensors and satellite images to discover insight of hydraulic residence time, plant species variation, and water quality and improve the overall understanding of possible nutrient removal in this constructed wetland.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 <u>Constructed wetlands of the Everglades</u>

The Everglades, situated in South Florida, is one of the largest tropical wetlands. Historically, water flowed south through the Kissimmee River basin into Lake Okeechobee and then continued south entering the headwaters of the Everglades. The water then followed a southerly flow through the Everglades eventually discharging into Florida Bay. Due to the increasing anthropogenic activities, around 2,830 square kilometers of land situated south of Lake Okeechobee was declared as the Everglades Agricultural Area (EAA). As a consequence, much of the runoff through this area has been rerouted through drainage canals, altering local hydrodynamic properties. Human interference including increased fertilizer application and livestock has created a new source of nutrient loading entering EAA runoff, eventually leading into the Everglades.

Excessive nutrients entering the Everglades deteriorate the overall water quality and can cause eutrophication in the area. As a measure to prevent continuous deterioration of water quality due to nutrient loading, the South Florida Water Management District (SFWMD) developed constructed wetlands named Stormwater Treatment Areas (STAs) at the intersection between the EAA and Everglades headwaters. Nearly 230 square kilometers to the south of Lake Okeechobee has already been converted to such wetlands. Total area of STAs including the infrastructure components is approximately 68,000 acres, out of which 57,000 acres area is known as the effective treatment areas (SFWMD). Among all the constructed wetland areas, STA-3/4 with an area of more than 64 square kilometers is known as the world's largest

constructed treatment wetland [1]. The main objective of these constructed wetlands is to intercept runoff entering the Everglades and reduce phosphorus loading. Mechanisms involved in retaining phosphorus include plant nutrient uptake, litter decay, co-precipitation of minerals, settling and sorption along with sedimentation and microbial uptake. The whole procedure of phosphorus retention in treatment wetland is shown in Figure 1.1. Normally, STAs are divided into small interior cells and the demarcation of those cell boundaries are developed by levees. Water flows from the upstream towards downstream through different water control structures e.g., pump stations, gates and culverts. The plant communities situated inside of STAs are broadly categorized in to three types as emerged aquatic vegetation (EAV), submerged aquatic vegetation (SAV) and floating aquatic vegetation (FAV).
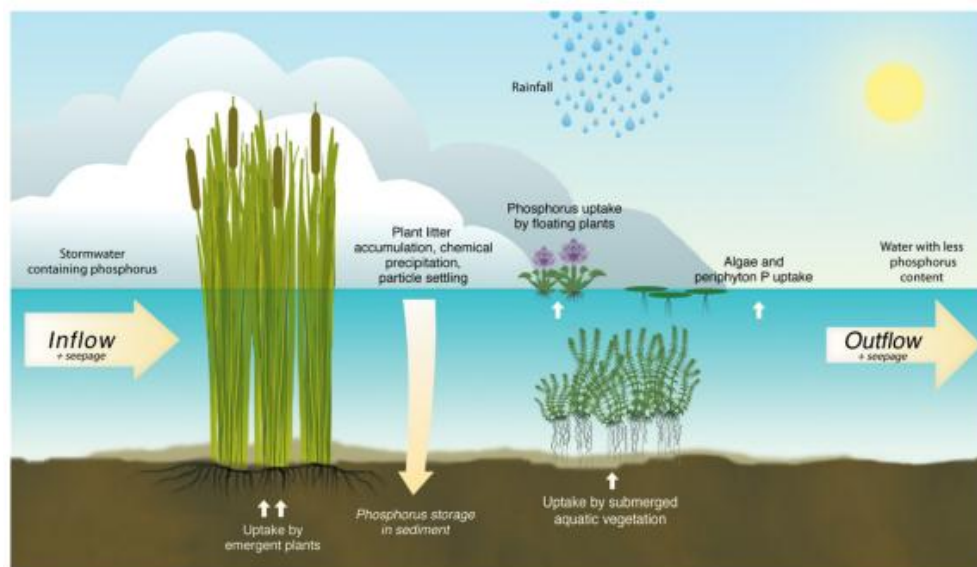
Figure 1.1: Nutrient removal through a constructed wetland system [1]

Each and every STAs around the Everglades has water quality based effluent limit (WQBEL) which represent the numeric discharge limit needs to be applied to all permitted discharges from Everglades STAs to the Environmental Protection Agency (EPA). On the year 2012, the Florida Department of Environmental Protection (FDEP) [2-3] issued Everglades Forever Act (EFA)

Watershed Permit (0311207) and National Pollutant Discharge Elimination System (NPDES) Watershed Permit (FL0778451) for operating the Everglades STAs to achieve the WQBEL limit. Such WQBEL pledges that the effluent discharge do not causes or exceeds the 10 microgram per liter (μg/l) (0.01 mg/l) total phosphorus (TP) (long-term geometric mean) established under 62-302.540, Florida Administrative Code [4]. TP concentrations are measured in the STA areas at the EPA designated locations in order to prevent the imbalances of aquatic flora and fauna. Discharge collected form each STA requires the following two limits [5]:

1. TP concentration will not exceed 13 μg/l (0.013 mg/l) as an annual flow-weighted mean in more than three out of five water years.

2. TP concentration will not exceed 19 μg/l (0.019 mg/l) as an annual flow-weighted mean in any water year.

1.2   Nutrient removal efficiency of treatment wetland

The most important parameters for any constructed wetland systems which affect the overall nutrient removal efficiency are the hydraulic loading rate, hydraulic retention time (HRT) and types of vegetation and its spatial distribution in the area [2-3]. HRT is strongly related to the flow velocity as high flow velocity may decrease the HRT which eventually decrease the treatment performance. Furthermore, the HRT can be increased by increasing the flow path in the treatment area. Vegetation species and species density can significantly increase the flow path by creating obstruction in front of the direction of natural flow of water and create complex flow circulations. Vegetation links wetland sediment to the water column, serving a unique role in wetland ecosystems and creating complex feedbacks between their biotic and abiotic components. Macrophyte communities and associated periphyton are acted as main vegetation

species in wetland. Emerged and submerged macrophytes along with macrophytic algae play a prevailing role in phosphorus cycling and deposition depending on their physical, chemical and biological interaction [1]. Major portions of the wetlands are covered with SAV, EAV such as chara, hydrilla, cattail, southern naiad and algae species. Presence of such aquatic plant mixtures imposes complex directional patterns on surrounding water pathways which affects the overall flow regime and velocity profile within the cell. In order to achieve a holistic view of any constructed wetland system, it is important to have an idea on the type of vegetation available in wetland and its density along with an idea of flow velocity and direction.

1.3   Study Area

In the current study, a constructed wetland treatment cell area, Cell 3B under STA-3/4 was selected for the analysis. The 3B cell area is shown in Figure 1.2. The total area of the cell is 8.4 square kilometers. The entire STA-3/4 area was topographically surveyed on 2008 by MACTEC Engineering and Consulting Inc on a 150m by 150m grid, following which a digital elevation model (DEM) was generated using a spline interpolation technique [8]. The DEM of Cell 3B is shown in Figure 3. The western side of the study cell area has a higher ground elevation than the eastern side. Cell 3B is primarily composed of SAVs with strips of EAV mainly comprised of cattail orientated in north-south and east-west directions. Flows enter STA-3/4 through a northern canal which transports runoff from the EAA, situated north of the STA and south of Lake Okeechobee. Inflow to the STA 3/4 area is controlled by two large pumping stations situated to the north, where inflow entering the individual cells is primarily managed by controllable weirs located at the influent and effluent sides.

The vegetation density information for the current study area for consecutive years 2010, 2011, 2012 were acquired from South Florida Environmental Reports (SFER), and years 2012, 2013 and 2014, respectively, published by SFWMD. Air borne digital imageries were used by SFWMD to calculate the vegetation density around different STA areas. Analyzing SFER reports it was found that the SAV coverage in Cell 3B decreased from 66 percent on year 2010 to 62 percent on year 2012, whereas, the EAV coverage increased from 34 percent to 38 percent for the same time period.
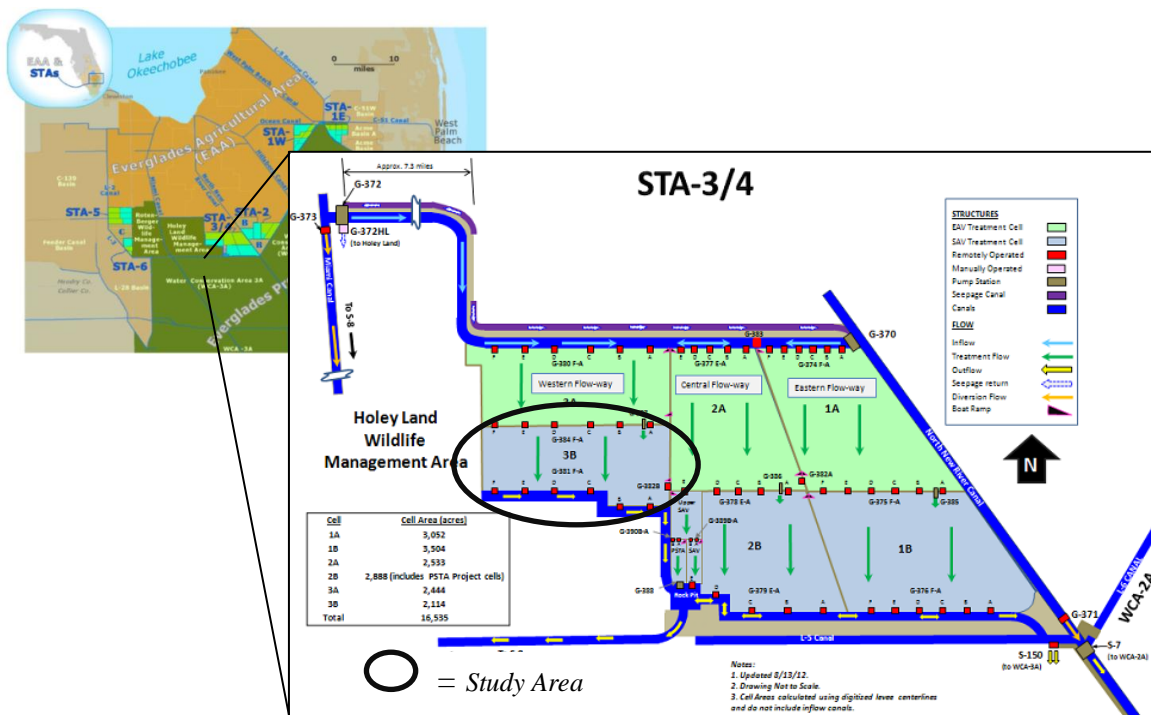
Figure 1.2: Study are (Cell-3B) [1]

Water samples were collected from different part of the study area on 21st May, 2014 in order to check whether the current cell condition could meet the WQBEL requirement established for TP concentration. Total 29 samples were collected and analyzed in the lab using HACH DR 5000 spectrophotometer. Following that, the TP concentration map was generated as shown in Figure 1.3. From the TP concentration map it was observed that, almost all part of the

area near to the southern effluent discharge side did not conform the prescribed WQBEL limit and the TP concentration in the effluent side went as high as 7.30 mg/l.
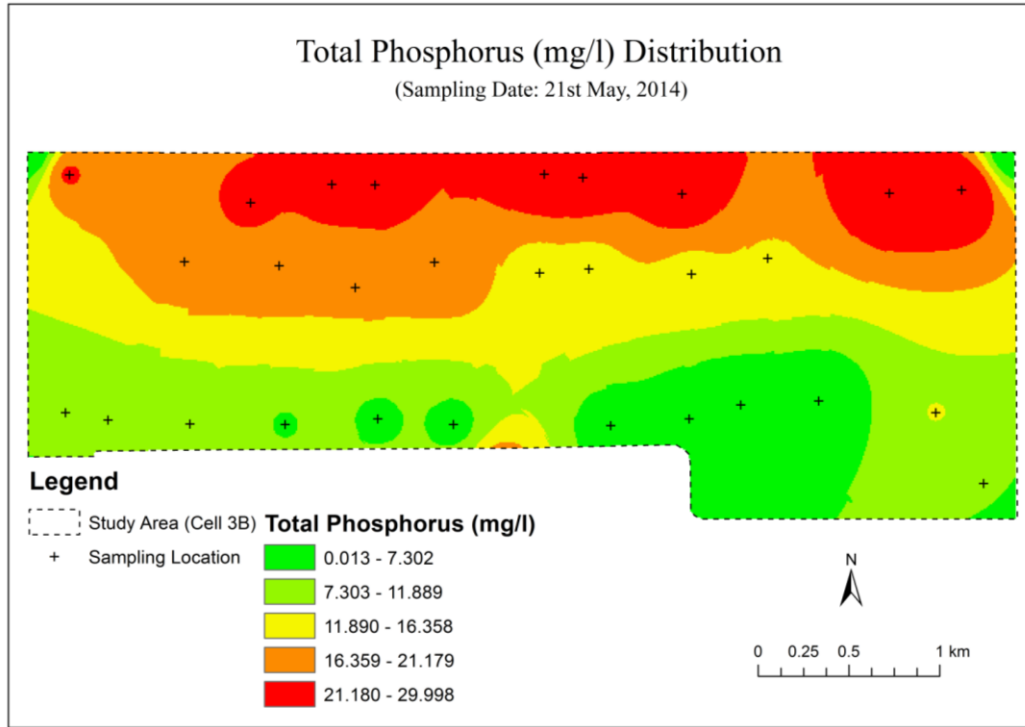
Figure 1.3: TP concentration (mg/l) map for cell-3B

1.4   Research objective

The main research efforts of this study is to investigate the ecosystem evolution of a constructed wetland system by predicting the biomass density, species classification and hydrodynamic parameters e.g., flow velocity and flow direction inside of a treatment cell area using different machine learning algorithms with the aid of satellite images. Scientific questions of this study include: 1.) Can machine learning algorithms comprehend the biomass density of a treatment wetland? [Chapter-2] 2.) Can the surface reflectance derived from visible bands of a multispectral satellite image perform well as an input parameter for machine learning based biomass prediction model? [Chapter-2] 3.) Can machine learning classification methods compete

6

with the traditional statistical methods? [Chapter-3] 4.) Under the same conditions, how are the learning capacities of ELM and ANN-BP compared in training a single-layer feed-forward neural network? [Chapter-3] 5.) Is it possible to predict the low flow velocity regime and HRT in a constructed wetland using computational intelligence models based on limited local measurements via a sensor network? [Chapter-4] 6.) Among different computational intelligence models, which model provides the most accurate estimation of low flow velocity regime and HRT for a constructed wetland system? [Chapter-4]

With these proposed scientific questions the objective of this study are to: 1.) Development of a machine learning based prediction model to predict biomass density 2.) Developing machine leaning based prediction model to predict vegetation species 3.) Developing machine learning based prediction model to predict flow velocity and flow direction and 4.) Comparative analysis between different machine learning algorithms, such as back propagation algorithm, extreme learning machine and genetic programming.

1.5 <u>References</u>

[1] SFWMD. 2013. Restoration Strategies Regional Water Quality Plan: Science Plan for the Everglades Stormwater Treatment Areas. South Florida Water Management District, West Palm Beach, FL. June 2013.

[2] FDEP. 2012a. STA National Pollutant Discharge Elimination System (NPDES) Watershed Permit (No.FL0778451) and Associated Consent Order (OGC 12-1148). Florida Department of Environmental Protection, Tallahassee, FL. September 10, 2012.

[3] FDEP. 2012b. STA Everglades Forever Act (EFA) Watershed Permit (No. 0311207) and Associated Consent Order (OGC 12-1149). Florida Department of Environmental Protection, Tallahassee, FL. September 10, 2012.

[4] SFWMD. 2012a. Restoration Strategies Regional Water Quality Plan. South Florida Water Management District, West Palm Beach, FL. April 27, 2012.

[5] SFWMD. 2014. South Florida Environmental Report. South Florida Water Management District, West Palm Beach, FL. June 2014.

[6] Kadlec, R. H. and Knight, R. L. (1996). Treatment wetlands, 1996. *Lewis, Boca Raton*, 893.

[7] Sakadevan, K. and Bavor, H. J. (1999). Nutrient removal mechanisms in constructed wetlands and sustainable water management. *Water Science and Technology*, *40*(2), 121-128.

[8] Piccone, T., MacBrian, J., Zhao, H., and Yan, Y. (2012). 2012 updated Everglades Stormwater Treatment Area average ground elevations, stage-Area/stage-volume relationships and effective treatment areas: Technical Publication. SFWMD.

CHAPTER 2: COMPARATIVE MACHINE INTELLIGENCE ANALYSIS FOR BIOMASS INVERSION OF EMERGED AND SUBMERGED AQUATIC PLANTS IN A CONSTRUCTED WETLAND

2.1   Introduction

Constructed wetland (CW) is a natural and efficient procedure of removing nutrient from a large area. The main purpose of such treatment wetland is to intercept nutrient laden runoff entering from the upstream and retain water for certain time period for nutrient removal, following that release water towards the downstream region. Almost all the nutrient removals in CW have been triggered by the submerged and emerged vegetation species situated inside of such treatment areas. As a part of their metabolic processes, plants within the wetland area uptake and store nutrient e.g., phosphorus within the plant cells, which eventually becomes partially retained by the sediment as the plants decay and sink to the bottom. Vegetation links wetland sediment to the water column, serving a unique role in wetland ecosystems and creating complex feedbacks between their biotic and abiotic components. In order to achieve significant removal rates of nutrients, larger hydraulic retention times (HRTs) with uniform slow velocities throughout the wetlands are required to fulfill sound restoration schemes that will be sustainable. Sparsely vegetated patches inside of wetland create some flow blockage and permit the rest of flow through the patch allowing minimal flow acceleration around it [01]. As stem density of vegetation becomes larger, flow through a patch may be increasingly diverted around the patch, resulting in higher flow acceleration in patch-adjacent areas [01, 02]. Due to the extreme prominence of vegetation for treatment wetland, it is indeed important to estimate the vegetation or biomass density in every location inside of such area. Estimating biomass is also necessary for nutrient allocation and carbon cycle for wetland areas [03, 04].

Remote sensing technology and satellite information have been considered as an efficient tool in delineating information collection from wetland areas for long time. Historically, both the multispectral and hyperspectral satellite sensors were used for vegetation mapping and biomass estimation for various wetland eco-systems. Landsat and SPOT satellite sensors were extensively utilized for such vegetation mapping. Common digital image classification techniques used in different studies include unsupervised and supervised classification [5, 6] and vegetation index clustering [7]. On the other hand, due to the greater spectral dimensionality, the hyperspectral sensors were used for in-depth discrimination of vegetation types [8, 9]. Out of several application of satellite information; estimation of different crop canopy variables or vegetation indices from remote sensing reflectance data in relation to the biochemical or physiological characteristics are the vivid ones [10, 11, 12]. Numerous numbers of studies have been conducted to examine the ecosystem and their annual variability over a large area of interest [13, 14]. Several studies showed the versatility of different vegetation indices such as; spectral vegetation index (SVI), simple ratio (SR), normalized difference vegetation index (NDVI), corrected normalized vegetation index (NDVIc) in predicting the leaf area index (LAI), biomass and productivity of different type of grassland and forest [15, 16, 17, 18]. Although vegetation indices do not provide the direct measure of biomass or primary productivity but those indices are closely correlated with biomass and LAI which eventually serve as a profound estimator of those parameters [19]. Among different vegetation indices, NDVI is the most common one used by ecologists for vegetation productivity [20], net primary production (NPP) [21, 22], percentage of absorbed photo-synthetically active radiations (APAR) [23], LAI [24] and plant biomass [25, 26, 27]. Several studies of field biomass estimation used vegetation indices based on the ratio of broadband red and near-infrared (NIR) reflectance. Relationship between canopy structure and

reflectances from Landsat TM (band 1-4) as well as from XMS SPOT (band 1-3) showed the complications in discerning mangrove species based on only the optical properties, but reveals the possibility of estimation of biomass using red and NIR bands [28]. Using the linear regression and statistical analysis between Landsat ETM reflectance (band 4, 3 and 2 false colors) and field biomass measurement it was shown that, the biomass has the highest positive correlation with difference vegetation index (DVI) [29].

In the current study, comparison between two artificial neural network (ANN) based machine learning algorithms known as back propagation (BP) and extreme learning machine (ELM) were conducted in predicting an important indices of constructed treatment wetland such as the biomass density. For more than several decades, BP algorithm combined with gradient decent optimization has widely been used for training different ANN models. ANN is inspired by the connections of human brain which uses algorithms to abstract away datasets and focus on the most important information. ANN models have been successfully applied to solve several water and environmental related problems [30]. ANN model was also used in several previous studies so as to build the relationships between lead concentration in grasses and permanent urban descriptors [31], leaf wetness prediction [32] to mapping tropical forest structure [33]. Iterative adjustment of all the network parameters e.g., input weights, hidden biases, output weights, etc is the major component of such approach in order to minimize the total error between the target and predicted output [34, 35]. As the gradient based optimization can lead towards slow training process and the convergence is highly dependent on the learning rate, the method often shows slow convergence or even divergence due to poor selection of learning rates. On the other hand, the recently developed ELM algorithm attracted attention in different field of optimization [36-40]. ELM is a fast and rigorous approach for training single-layer feedforward

neural networks (SLFN). In principle, ELM analytically finds suitable output weights by randomly assigning values to the input weights. This is in contrast with the standard learning algorithms which iteratively search for optimized input and output weights. As a result, the training process in the ELM is much faster than the other approaches. ELM is known to avoid over-fitting that is an issue in traditional methods [41]. It has been also shown that an ELM with N hidden units (also called random units) can learn exactly N distinct observations with zero error [41]. Recently ELM has been widely used for many applications from data mining in cancer diagnosis to loss analysis in electrical power systems and from sales forecasting to image processing [42, 43]. ELM has been also used for several multi-class classification studies [41-43].

In this study, 2m multispectral Pleiades satellite bands is utilized to develop different spectral indices which eventually used in the machine learning algorithms (ELM and BP) to predict the biomass density for a treatment wetland. For predicting biomass density, red band based NDVI ratio was used as the major spectral indices for both submerged and emerged vegetation. Neglecting the bottom reflectance effect for submerged vegetation was used as the hypothesis in calculating the NDVI values because of the extremely shallow depth and low turbidity water in the study area. During shallow depth condition, the visible band surface reflectance values can also be used as the reflectance for the bottom due to its penetration capability and negligible attenuation effect [44]. With this study objective, the following scientific questions are explored in this study: 1) Can machine learning algorithms comprehend the biomass density of a treatment wetland? 2) Can the surface reflectance derived from visible bands of a multispectral satellite image perform well as an input parameter for machine learning based biomass prediction model?.

2.2 Methodology

2.2.1 *Field biomass data collection and processing*

The term "biomass density" in this study refers to the dry weight of vegetation sample per unit area. A field trip was organized on 10th June, 2014 to collect the biomass samples from the treatment area. Both the SAV and EAV samples were collected using PVC made square quadrat of 0.25 m^2. In order to maintain a uniform area during calculation of biomass, a fixed quadrat was utilized in sample collection. Total 30 samples of vegetation were collected which include 13 SAV and 6 EAV samples. Remaining 11 samples were the samples of no aquatic vegetation (NAV) which mostly located at the middle and surrounding canal area. Every sampling location was recorded by taking specific position using hand-held Global Positioning System (GPS) (Figure 2.1). Average GPS error was fluctuated between 6 – 9 meters. After collecting those vegetation samples, all of them were weighted (wet weight) and following that oven dried at 60°C for 48 hours. The oven dried samples were weighted (dry weight) and finally the biomass was calculated by dividing the difference of weight between wet and dry sample with quadrat area of 0.25 m^2. Average moisture content in SAV and EAV samples were found around 40% and 20%, respectively.
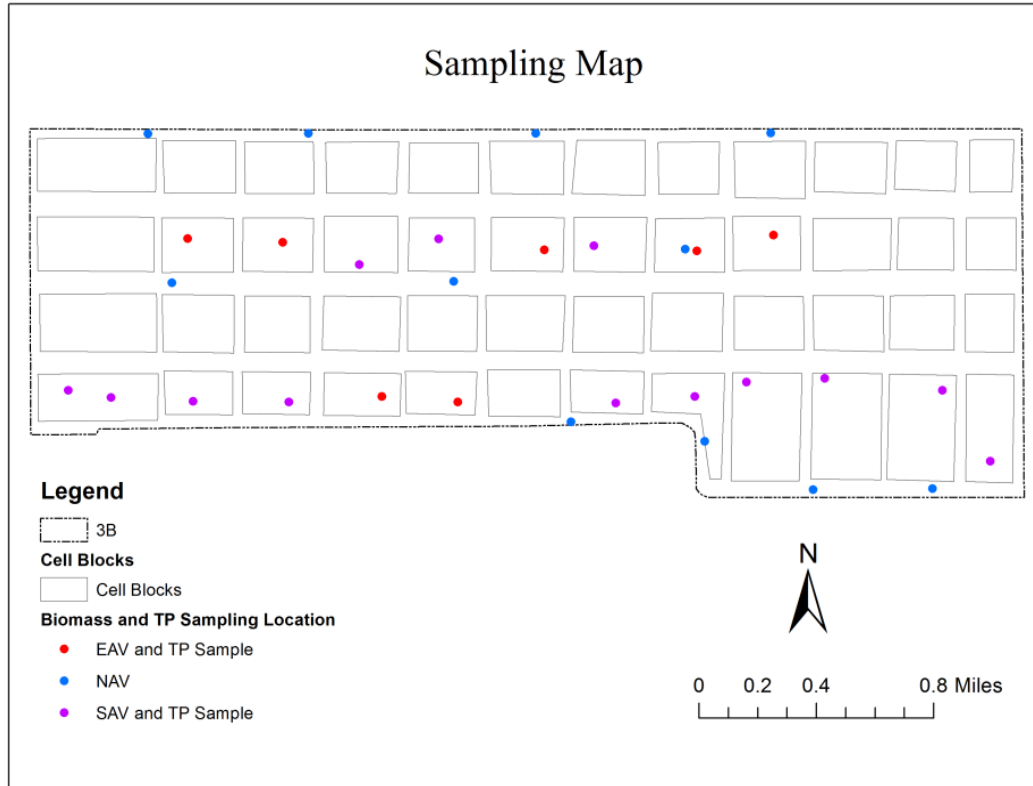
Figure 2.1: Biomass sampling location map

2.2.2 *Satellite data collection and processing*

2.2.2.1 *Data acquisition*

Pleiades satellite image was acquired for 21st May, 2014 which is around 20 days before collecting the field samples. As the change of vegetation density in cell 3B area is quite static (SFWMD), it was assumed that, 20 days' time lag will not affect the current analysis. Pleiades satellite is a cooperation program initiated by France and Italy which provides an earth observation system with metric resolution. Two Pleiades satellite were launched as Pleiades 1A and Pleiades 1B. The launching time of Pleiades 1A and Pleiades 1B satellite were at the fourth quarter of year 2011 and 2012 respectively. The satellite has a striking acquisition capacity with maximum theoretical acquisition capacity reaches up to 1 million square kilometers per day per

satellite. Pleiades coverage provides a larger image footprint than other satellites of same class of resolution partly due to its high swath of 20 kilometers. Such high coverage normally maximized information on target and its surrounding with diminishing work load for mosaicking over large areas. Pleiades bundle product comes with 0.50 m panchromatic and 2m multispectral bands. In this study, multispectral band of Pleiades 1B satellite image with a spatial resolution of 2m and zero cloud coverage was used. There are four bands in the multispectral band as red, green, blue and near-infrared. Detail information of the collected satellite image is shown in Table 2.1.

Table 2.1: Acquisition parameter of Pleiades satellite image used in the study

Acquisition date	Sun Elevation (^{o})	Sun Azimuth (^{o})
21st May, 2014	73.037	106.869
Band	Spectral Region (μm)	Centre Wavelength (μm)
Panchromatic	0.47 - 0.83	0.65
B1 - Blue	0.43 - 0.55	0.49
B2 - Green	0.50 - 0.62	0.56
B3 - Red	0.59 - 0.71	0.65
B4 - Near Infrared	0.74 - 0.94	0.84

2.2.2.2 *Satellite image quality, processing and bit depth*

Pleiades image qualities are normally ensured by using a 50 cm post-processing resampling algorithm. The entire process was developed and implemented by the French Space Agency. Resampling algorithm normally consists of three main steps as i) deconvolution-which enhance the image sharpness, ii) de-noising and iii) zooming factor on on-board resolution. By conducting a post processing algorithm a preservation of the acquired information can be

ensured along with an increasing in robustness during post processing phase like image rotation or re-projection. Ortho processing level was used for the satellite image. Ortho represents a geo-referenced image corrected from acquisition and terrain nadir effects. Along with the radiometric and geometric adjustments, a geometric process using a relief model known as ortho-rectification was deployed in order to eliminate the perspective effect and restore the geometry of vertical shot. Pixel depth of Pleiades at acquisition is 12 bits which means, at every spectral bands each pixel can take one values out of 4096. Such high value confirms any subtle nuisance in the beginning and end of spectrum. Besides that, due to the high pixel depth Pleiades can detect objects in the darkness of the shadow of building or mountain and also detect pale-colored elements in very light or bright environments (sand, ice, near-white ground) as more saturation problems are avoided.

2.2.2.3 *Spectral modeling in Pleiades*

Spectral modeling of Pleiades satellite image was carried out following different steps. At initial level the image contained only the digital numbers (DN) for every pixel. First the initial DN was converted to top of atmosphere (TOA) radiance and following that TOA was converted to reflectance values. For any respective image band, the TOA radiance was found using the following equation:

$$L(p) = \frac{X(p)}{A(b)} + B(b) \hspace{4cm} (\,2.1\,)$$

Where, $L(p)$ is the TOA radiance in $W.sr^{-1}.m^{-2}.\mu m^{-1}$ per pixel, $X(p)$ is the DN number for each pixel, $A(b)$ is the absolute radiometric calibration gain in $W.sr^{-1}.m^{-2}.\mu m^{-1}$ for the respective band ($= 12.04\ W.sr^{-1}.m^{-2}.\mu m^{-1}$) and $B(b)$ is the bias value which was considered as zero for the current panchromatic band.

Conversion of TOA radiance to spectral reflectance for each pixel of image was carried out using the following equation:

$$\rho_b(p) = \frac{\Pi \cdot L(p)}{E_o(b) \cdot \cos(\theta_s)}$$
(2.2)

Where, $\rho_b(p)$ is the reflectance for each pixel, $E_o(b)$ is the solar equivalent irradiance for respective band (= $1548\ W \cdot sr^{-1} \cdot m^{-2} \cdot \mu m^{-1}$), and θ_S is the sun zenith angle. Sun zenith angle can be found by deducting the sun elevation angel (= 73.03 degree) from 90 degree.

2.2.3 *Machine learning algorithm*

2.2.3.1 *Back propagation (BP) algorithm*

The three main components of the ANN model design are the input layer(s), hidden layer(s) and an output layer(s) of processing unit or neurons. The input layers introduce the inputs to the network, then the hidden layers transforms those inputs using non-linear activation functions. The hidden layers can work in one layer or in multilayers. Each layer of neuron possesses a distinct activation function and different layers of neuron possess different activation functions. The most simple neural network design can be represented as displayed in Figure 2.2. Here an input is multiplied with weight (w) and then the weighted input (wp) is multiplied with bias (b) to produce a net input (n). The net input is then passed through an activation function (f) which finally produces the output (o). Several types of activation functions may be used in generating output from the neural network model. The two most commonly used activation functions are linear transfer function and log-sigmoid transfer function.
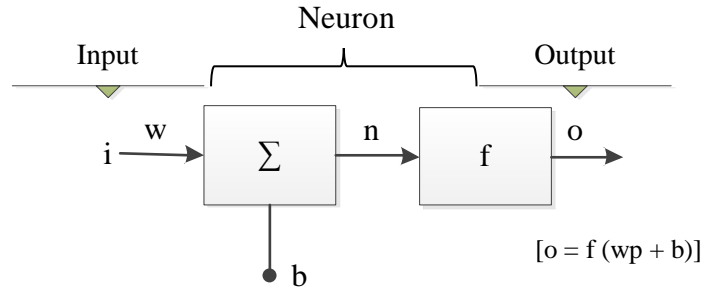
Figure 2.2: Simple neural network

Linear transfer functions are normally used in the final layer of multilayer neural networks as a function approximator. The log-sigmoid transfer functions are used in hidden layers of multilayer neural network where the function takes input ranging from minus infinity to plus infinity and finally generates an output in the range of 0 to 1. All ANN models applied a set of rules which change the connection weights in an iterative process during the learning phase until a certain stopping criteria is reached. Back propagation (BP) training algorithm was used as a learning algorithm which is known as a common algorithm utilized in more than 80% of previous neural network model studies [44]. In supervised learning process, the desired outputs or responses are normally available to guide the whole process. BP algorithms initially assign an arbitrary small value to the weights. With the progress of the training process the mean square error and coefficient of determination, or R-square value between the target output and network output, is calculated. The weights assigned previously are also updated methodically with the evolution of training process. The training process stops when a certain stopping criterion is reached. The whole data sets are divided into three parts as training, validation and test dataset. Training data are presented to the designed network during the training process and the network along with its weights and biases are adjusted according to its error. Validation datasets are normally used to measure the generalization of network and halt the training process when the

generalization process stops. Test data sets do not have any impact on the training process; rather it gives an independent measurement of network performance during and after the training process.

Several BP algorithms are available for training the dataset during the learning process, out of which a Levenberg-Marquardt BP algorithm was used for the current study. The Levenberg-Marquardt algorithm is a derivative of the Newton's method [45]. This multifaceted algorithm not only derives the gradient but also derives the Jacobian matrix as well. Considering the function V(m) which needs to be minimized with respect to the parameter vector m, then the Newton's method would be:

$$\Delta m = -[\nabla^2 V(m)]^{-1}\nabla V(m) \tag{2.3}$$

Here, $\nabla^2 V(m)$ represents the Hessian matrix and $\nabla V(m)$ represents the gradient. Assuming V(m) as the sum of square function or performance function it can be presented as below [46]:

$$V(m) = \sum_{i=1}^{N} e_i^2 (m) \tag{2.4}$$

Then the Hessian and gradient will be as below:

$$\nabla V(m) = J^T(m)e(m) \tag{2.5}$$

$$\nabla^2 V(m) = J^T(m)J(m) + S(m) \tag{2.6}$$

where, J(m) is the Jacobian matrix and

$$S(m) = \sum_{i=1}^{N} e_i(m)\nabla^2 e_i(m) \tag{2.7}$$

According to the Gauss-Newton method S(m) ≈ 0 then Equation 2.4 can be illustrated as:

$$\Delta m = [J^T(m)J (m)]^{-1}J^T(m)e(m) \tag{2.8}$$

Levenberg-Marquardt modification to the Gauss-Newton [47] method is:

$$\Delta m = [J^T(m)J (m) + \mu I]^{-1} J^T(m)e(m) \tag{2.9}$$

Considering Equation 2.9 the Levenberg-Marquardt algorithm uses the approximation towards the Hessian matrix as follows:

$$m_{k+1} = m_k - [J^T(m)J(m) + \mu I]^{-1} J^T(m)e(m) \qquad (2.10)$$

Following Equation 2.10, when μ approaches zero, it will turn into Newton's method. When μ is large this becomes a gradient descent with a small step size. In every BP algorithm the main target is to shift towards Newton's method quickly as Newton's method is much faster and accurate near minimum error. After every successful step, the value of μ decreased, whereas, μ increased when the step size increases the performance function. Following this procedure the performance function can reduce its size after each iteration.

2.2.3.2 *Extreme learning machine (ELM) algorithm*

Here we briefly review the formalism of Extreme Learning Machine (ELM). Consider a single-layer feed-forward neural network that as shown in Figure 2.3. The neural network is assumed to be fully-connected and containing N inputs $x_1, x_2, \dots x_N$, M outputs $y_1, y_2, \dots, y_M$ and L hidden units with the biases $b_1, b_2, \dots, b_L$ and activation functions $g_1, g_2, \dots, g_L$. The corresponding input weight of a connection between the nth input and the lth hidden unit is denoted with w_{nl}, while the output weight of a connection between the lth and the mth output is represented by β_{lm}. Assuming that K set of input/output samples is given as follows:
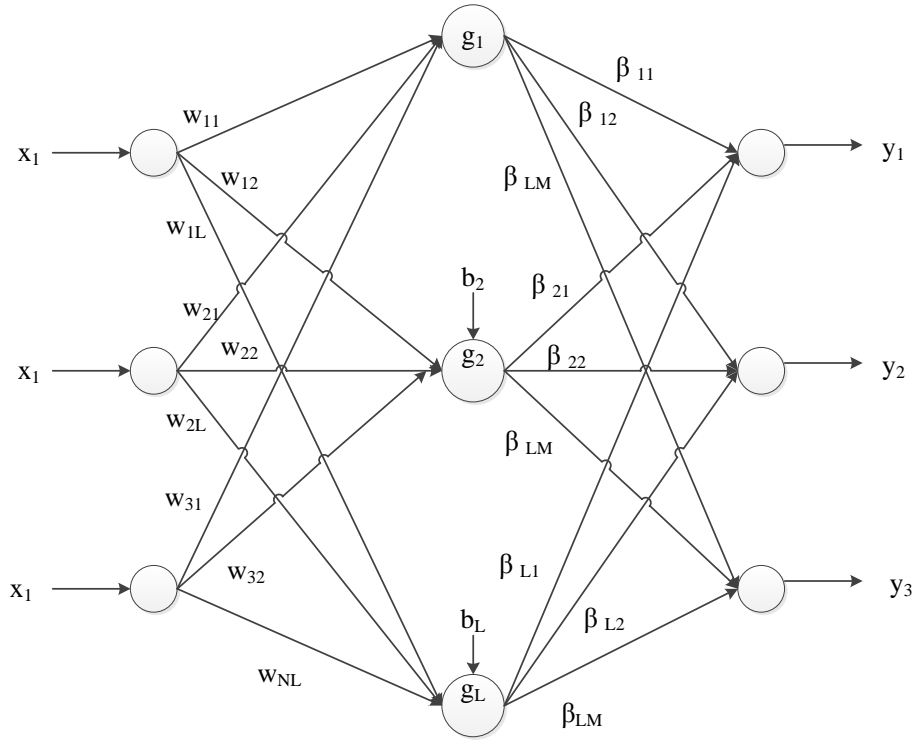
Figure 2.3: A single layer feed-forward neural network

$$X = \begin{bmatrix} x_1^{(1)}, x_2^{(1)}, \dots, x_N^{(1)} \\ \vdots \\ x_1^{(K)}, x_2^{(K)}, \dots, x_N^{(K)} \end{bmatrix} \qquad\qquad (2.11)$$

$$Y = \begin{bmatrix} y_1^{(1)}, y_2^{(1)}, \dots, y_M^{(1)} \\ \vdots \\ y_1^{(K)}, y_2^{(K)}, \dots, y_M^{(K)} \end{bmatrix} \qquad\qquad (2.12)$$

the forward propagation in the neural network demands that the following relation should hold between the inputs and outputs:

$$H\beta = Y \qquad\qquad (2.13)$$

where,

$$H = \begin{bmatrix} g\left(w_{11}x_1^{(1)} + \cdots + w_{N1}x_N^{(1)} + b_1\right) & \cdots & g\left(w_{1L}x_1^{(1)} + \cdots + w_{NL}x_N^{(1)} + b_L\right) \\ \vdots & \ddots & \vdots \\ g\left(w_{11}x_1^{(K)} + \cdots + w_{N1}x_N^{(K)} + b_1\right) & \cdots & g\left(w_{1L}x_1^{(K)} + \cdots + w_{NL}x_N^{(K)} + b_L\right) \end{bmatrix} \qquad (2.14)$$

contains the output of the hidden layers for different input samples and β describes the output weights:

$$\beta = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1M} \\ \vdots & \ddots & \vdots \\ \beta_{L1} & \cdots & \beta_{LM} \end{bmatrix} \qquad (2.15)$$

Now, if one randomly assigns the input weights and biases of the hidden units, the matrix H is fully determined, and therefore the unknown output weights can be determined through Equation 12:

$$\beta = H^\dagger Y \qquad (2.16)$$

where $H^\dagger$ represents the Moore-Penrose generalized inverse of H. Alternatively it has been shown that one can also use the following relation [48]:

$$\beta = H^T \left(\frac{I}{C} + HH^T\right)^{-1} Y \qquad (2.17)$$

where I represent a $K \times K$ identity matrix and C is a control parameter. Once the output weights are found, the output for any arbitrary number of inputs can be found simply by a forward propagation through the neural network. This can be done for many input sets at the same through Equation 11 while this time H should be calculated with the new data sets.

The standard procedure explained in Equation $2.11 - $ Equation 2.17 is for regression purposes. On the other the multi-class classification problem can be reduced to regression. Assuming that q different target classes $c_1, c_2, \ldots, c_q$ exist, we consider q output neurons where the target classes

are considered as binary vectors $[1,0,...,0]^T, [0,1,...,0]^T, ..., [0,0,...,1]^T$ respectively. When predicting the classes for new data, the outputs should be calculated as usual and depending on how close the predicted vector to the C target classes, a specific class will be assigned to the new data.

2.2.4   *Model development*

Two different modeling approaches were considered for predicting biomass. For biomass density prediction using different supervised computational intelligence algorithms, two different training datasets were prepared. First dataset contains the sampling location (latitude, longitude) and NDVI ratio as the input for model training. Whereas, the measured biomass density of the same locations were used as target dataset for model training. In the second dataset, a categorical variable was introduced along with previously used inputs (location and NDVI) to increase the dimensionality of the input dataset for supervised learning process. Three different categories were used for three different conditions as SAV, EAV and NAV. NDVI value of 0.25 was used as the demarcation value to differentiate the submerged and emerged vegetation. NDVI values less than zero was considered as the submerged part and NDVI higher than zero was considered as the emerged part of the study area. The Same measured biomass densities were also used as a target for the training dataset. Total 1294 sampling pixels were used as training dataset which correspond to 30 sampling points considering different GPS accuracy ranges during sample collection. The detail of model formulation is presented in a flowchart shown in Figure 2.4.
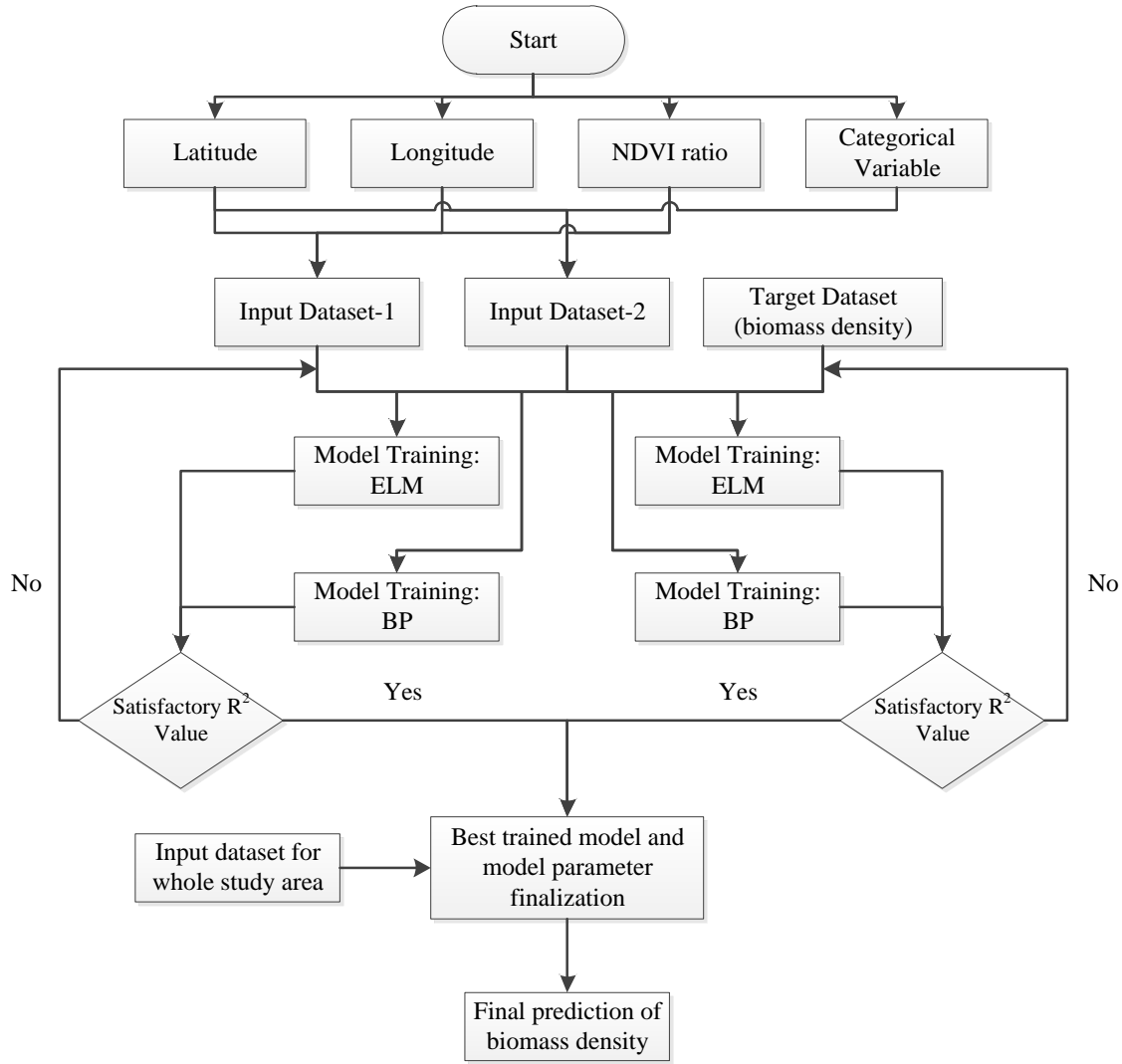
Figure 2.4: Flow chart for biomass density prediction

2.2.5 *Statistical indices*

For the current study, two statistical indices were considered to check the model performance. Those indices are coefficient of determination (R-square) and Pearson product moment correlation coefficient (RSQ). The indices R-square was used to evaluate the fitting performance during the initial training of the model, while the remaining two indices were used

to check the model performance during validation stage. The RSQ and R-square can be calculated using the Equation. 2.18 - 2.19 as below:

$$RSQ = \frac{\sum_{i=1}^{N}(Y_{oi} - \overline{Y_{oi}})(Y_{pi} - \overline{Y_{pi}})}{\sqrt{\sum_{i=1}^{N}(Y_{oi} - \overline{Y_{oi}})^2 \sum_{i=1}^{N}(Y_{pi} - \overline{Y_{pi}})^2}} \qquad (2.18)$$

$$R\text{-square} = RSQ^2 \qquad (2.19)$$

Here, Y_{pi} is the i-th predicted value, Y_{oi} is the i-th observed value, N is the total number of sample, $\overline{Y_{oi}}$ is the arithmetic mean of the observed values and $\overline{Y_{pi}}$ is the arithmetic mean of the predicted values. Any ideal model would depict an RSQ value of 1.

2.3   Result and discussion

2.3.1   *Model training using non-categorical input for ELM and BP algorithm (biomass density)*

Two different input datasets were used for both BP and ELM. First dataset was considered without using any categorical input in order to predict general biomass density for throughout the study area. The BP algorithm was simulated for different number of hidden neurons and training data splitting ratio. All the training samples were divided in to three sections known as the training, validation and test dataset. Training data are presented to the designed network during the training process and the network along with its weights and biases are adjusted according to its error. Validation datasets are normally used to measure the generalization of network and halt the training process when the generalization process stops. Test data sets do not have any impact on the training process; rather it gives an independent measurement of network performance during and after the training process. In this case the best result was found using the neuron number of 20 and the sampling data distribution of 70% for the training, 15% for validation and 15% for test data. The evolution of accuracy and simulation

time for both ANN-BP and ELM with different number of neurons is shown in Figure 5. All the

accuracies presented in Figure 2.5 show the R-square values.
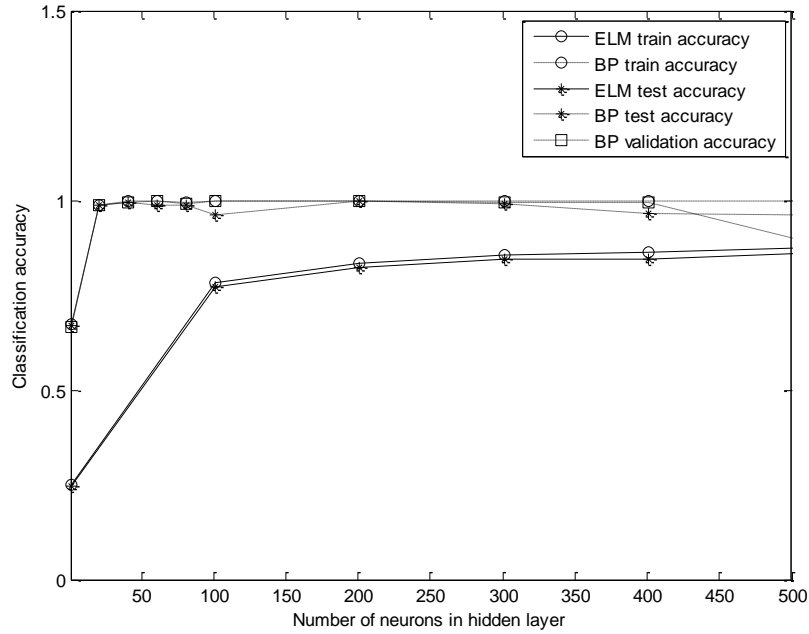
Figure 2.5: Evolution of accuracy with different number of neurons for BP algorithm and for ELM when non-categorical inputs are utilized.
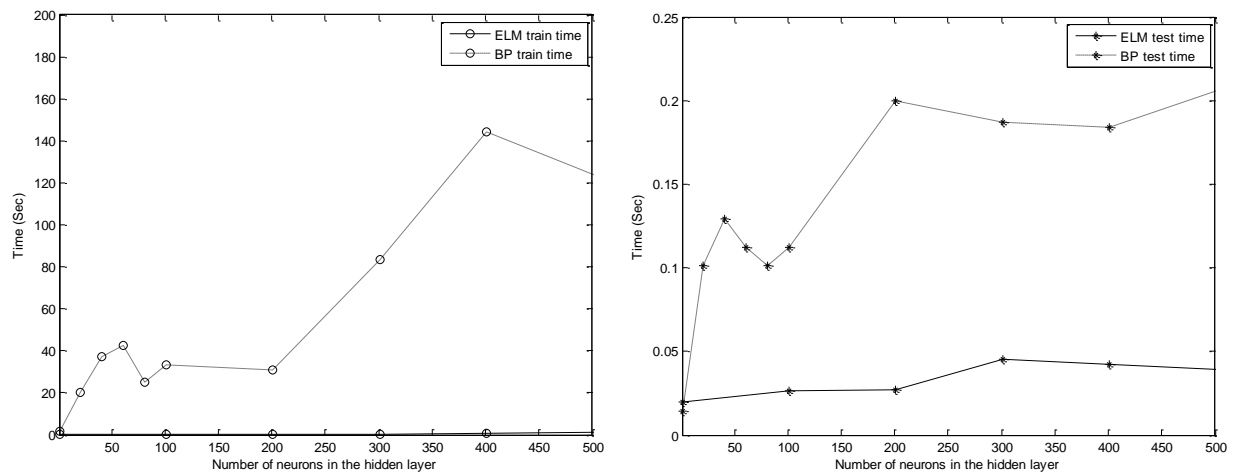
Figure 2.6: Evolution of the training (left) and test (right) time with different number of neurons for BP algorithm and for ELM when non-categorical inputs are utilized.

26

For ELM on the other hand, similar behavior is observed, however, the transition point where the accuracy becomes steady occurs at 100 hidden neurons. It clearly indicates that the computational time of ELM is much less than that of ANN-BP in both the training and test processes (Figure 2.6). However, comparisons of the training, validation and test accuracies for ANN-BP and for ELM as a function of the splitting ratio for training provide deeper insight of the reliability of the two inverse modeling techniques (Figure 2.7). For the case of ANN-BP, the training, validation and test accuracy reaches to maximum of 0.962, 0.973 and 0.966, respectively, at the 70% training data splitting ratio. For the case of ELM, on the other hand, the trend of the accuracy curve for training data was decreased gradually with the increase of training data splitting ratio as the increment in the model input normally increase the model complexity which affect the overall accuracy. Reverse trend was observed for the test dataset for the case of ELM. The optimum accuracy was found by considering both the training and test accuracy for training data splitting ratio of 70%.
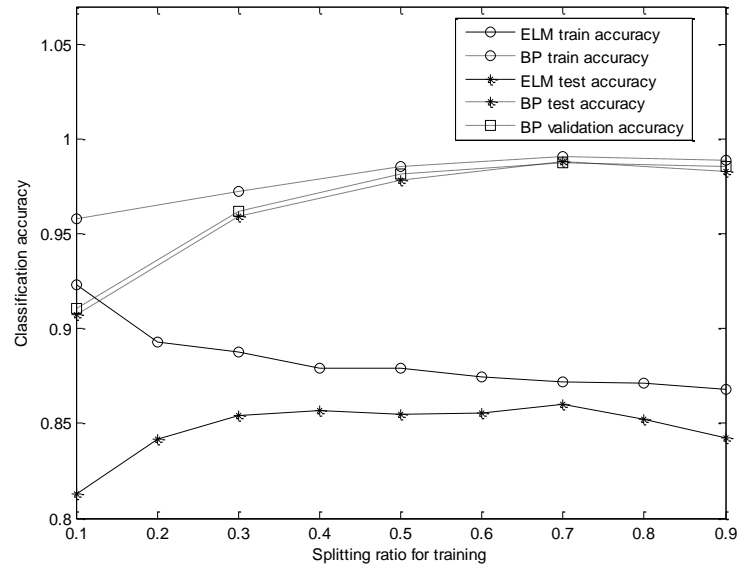
Figure 2.7: The training, validation and test accuracies for both ANN-BP and ELM as a function of the splitting ration of training.

2.3.2 *Model training using categorical input for ELM and BP algorithm (biomass density)*

In this simulation, a categorical variable was introduced as an input in the training dataset. The main objective of introducing such variable was to increase the dimensionality of the model input. Under this condition the learning capacities are discussed again in Figures 8-10. As shown in Figure 2.8, by gradually increasing the number of hidden neurons, the accuracies converge to a constant number. For ANN-BP, this saturation happens at 100 hidden neurons while for ELM it happens at around 200 hidden neurons. While both approaches offer a high accuracy, in general, ANN-BP better. On the other as expected, the computational time of the ANN-BP is much higher than the ELM (Figure 2.9).
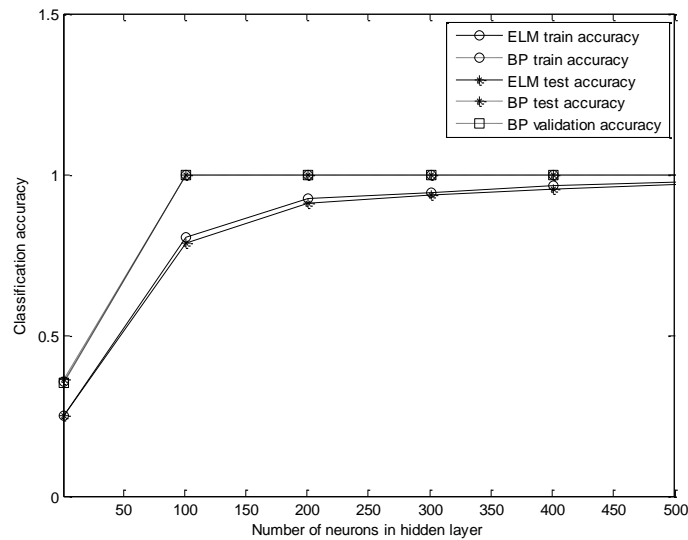
Figure 2.8: Evolution of accuracy with different number of neurons for BP algorithm and for ELM when categorical inputs are utilized.
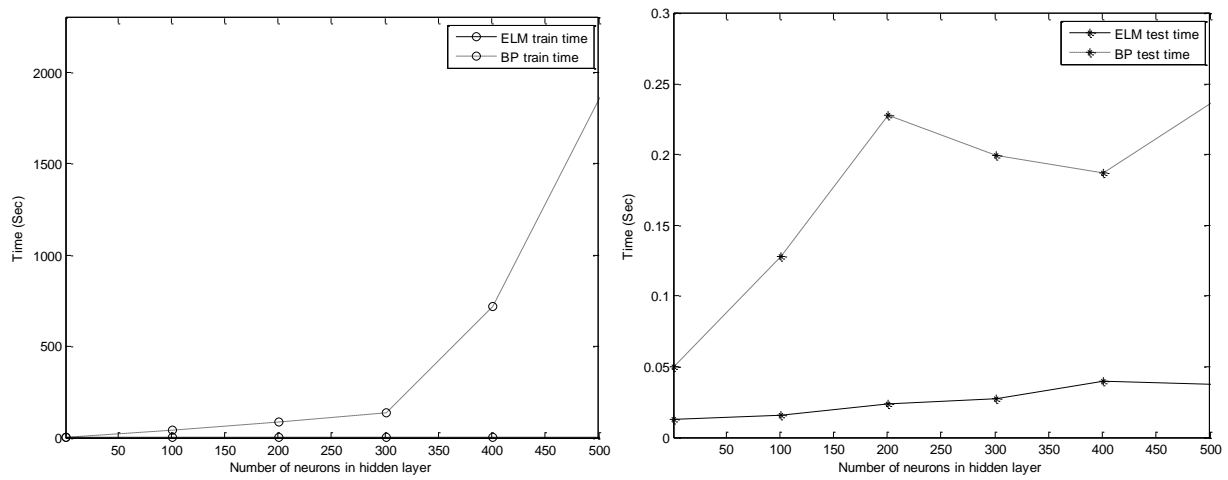
Figure 2.9: Evolution of the training (left) and test (right) time with different number of neurons for BP algorithm and for ELM when categorical inputs are utilized.
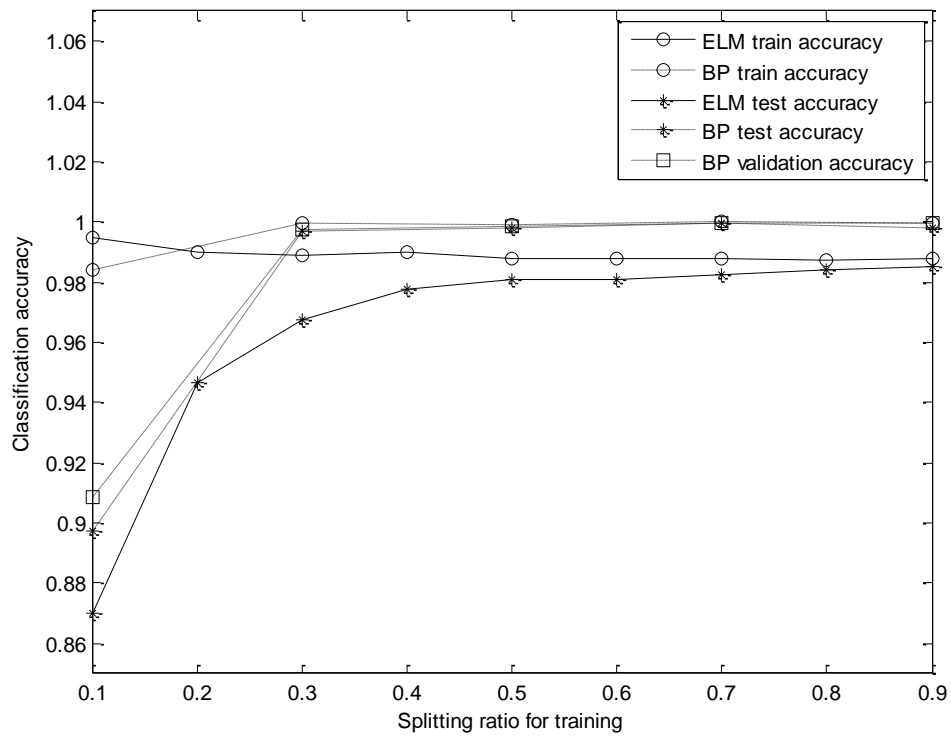
Figure 2.10: The training, validation and test accuracies for both ANN-BP and ELM as a function of the splitting ratio of training.

Finally, we also investigated the effect of the splitting ratio of the training on the classification accuracies in this case. As in the previous case, by increasing the splitting ratio, the training, validation and test accuracies of the ANN-BP increases (Figure 10). On the other hand for ELM, by gradually increasing the splitting ratio, the training accuracy decreases while the test accuracy increases until both converge to an acceptable accuracy of 98% which happens for a splitting ratio of about 70% (Figure 2.10). The training, test and validation accuracy found using both types of inputs are summarized in Table 2.2.

Table 2.2: Model accuracy training, validation and test dataset

Types	Algorithms	Training (R^2)	Validation (R^2)	Test (R^2)	Overall (R^2)
Non-Categorized	ELM	0.933		0.932	0.932
	BP	0.962	0.973	0.966	0.964
Categorized	ELM	0.963		0.962	0.962
	BP	0.999	0.999	0.999	0.999

2.3.3 *Prediction of biomass*

Depending on the high accuracy found using BP algorithm, the prediction of biomass was made for the whole study area. The prediction of biomass density is shown in Figure 2.11. The accuracy of the prediction was checked by using the R-square values for a separate number of pixels. Total 214 pixels were separated from the training sample in order to utilize them in measuring the prediction accuracy. The maximum R-square was found using BP algorithm with categorical variable input of 0.55. The reason of having such moderate accuracy in prediction

30

can be mainly due to the lower number of samples and absence of more EAV samplings due to the remote location of the site. From the predicted biomass density it was observed that, the north-west, south-west and central-southern part of the study area have high biomass density which outdone 10 kgm^{-2}. Linking the predicted biomass density information with field observation, it can be concluded that main biomass contributor of those areas are the EAV. The mid-central part of the area where the biomass density varied between 5-10 kgm^{-2}, the main contribution of biomass came from the SAV. A large part in the central region showed pretty low vegetation density which is below 4 5-10 kgm^{-2}.

Biomass Density (kgm^{-2}) for Cell-3B Area
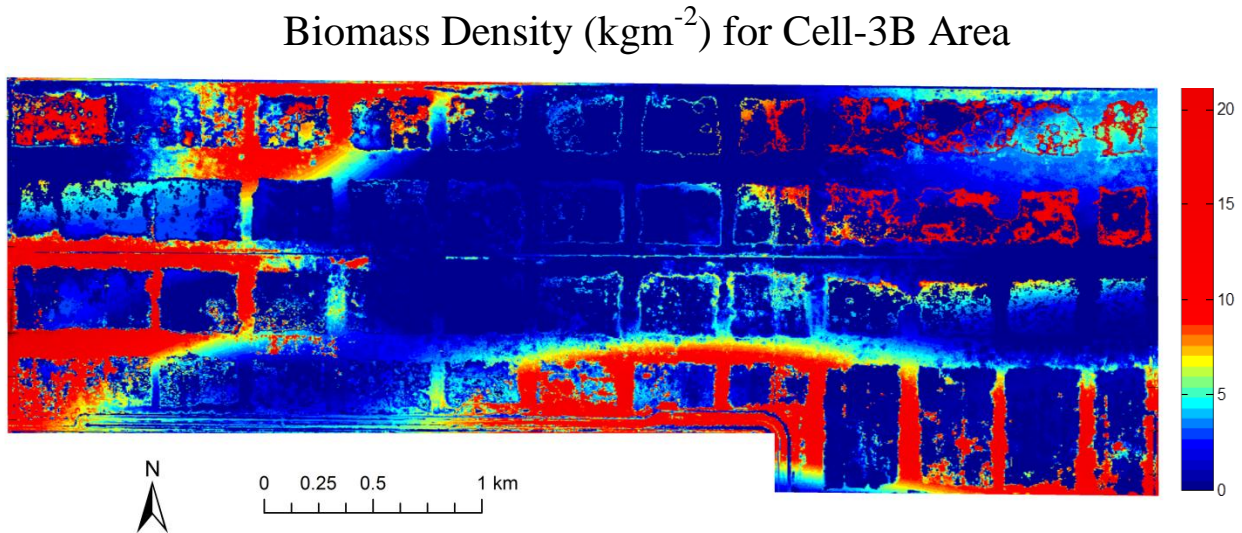
Figure 2.11: Biomass prediction for cell 3B

2.4   Conclusion

In this study, we presented the comparative capacity of machine learning algorithms with the use of multispectral satellite image spectral indices in predicting one of the imperative components of any constructed wetland system, the biomass density. Despite of numerous data requirement of machine learning languages, we still managed to find a moderate result using both BP and ELM algorithms. Implications of relatively new machine learning algorithm such as

ELM certainly open up a new set of insightful discoveries of its learning capacity in predicting biomass density for any eco-system on the Earth. Our observation is that a faster leaner such as ELM might not be a stable or accurate learner in the context of machine intelligence.

In this study many part of the emerged vegetation site couldn't be sampled due to its remote location and enormous size which hampered the sampling campaign. The prediction capacity of such models can be improved by adding more distributed and diversified *in situ* sampling information. Incorporating such information will surely improve not only the training accuracy of the model but also the prediction capacity of such models. Finally, the spatial accuracy of biomass sampling can be increased by deploying much higher accuracy GPS system rather than the traditional hand-GPS system used in this study.

2.5   <u>References</u>

[01] Bouma, T. J., Van Duren, L. A., Temmerman, S., Claverie, T., Blanco-Garcia, A., Ysebaert, T., & Herman, P. M. J. (2007). Spatial flow and sedimentation patterns within patches of epibenthic structures: Combining field, flume and modelling experiments. *Continental Shelf Research*, *27*(8), 1020-1045.

[02] Vandenbruwaene, W., Temmerman, S., Bouma, T. J., Klaassen, P. C., De Vries, M. B., Callaghan, D. P., ... & Meire, P. (2011). Flow interaction with dynamic vegetation patches: Implications for biogeomorphic evolution of a tidal landscape. *Journal of Geophysical Research: Earth Surface (2003–2012)*, *116*(F1).

[03] Zheng, D., Rademacher, J., Chen, J., Crow, T., Bresee, M., Le Moine, J., & Ryu, S. R. (2004). Estimating aboveground biomass using Landsat 7 ETM+ data across a managed landscape in northern Wisconsin, USA. *Remote Sensing of Environment*, *93*(3), 402-411.

[04] Mutanga, O., & Skidmore, A. K. (2004). Hyperspectral band depth analysis for a better estimation of grass biomass (< i> Cenchrus ciliaris</i>) measured under controlled laboratory conditions. *International journal of applied earth observation and geoinformation*, *5*(2), 87-96.

[05] McCarthy, J., Gumbricht, T., & McCarthy, T. S. (2005). Ecoregion classification in the Okavango Delta, Botswana from multitemporal remote sensing. *International Journal of Remote Sensing*, *26*(19), 4339-4357.

[06] Harvey, K. R., & Hill, G. J. E. (2001). Vegetation mapping of a tropical freshwater swamp in the Northern Territory, Australia: a comparison of aerial photography, Landsat TM and SPOT satellite imagery. *International Journal of Remote Sensing*, *22*(15), 2911-2925.

[07] Yang, X. (2007). Integrated use of remote sensing and geographic information systems in riparian vegetation delineation and mapping. *International Journal of Remote Sensing*, *28*(2), 353-370.

[08] Cochrane, M. A. (2000). Using vegetation reflectance variability for species level classification of hyperspectral data. *International Journal of Remote Sensing*, *21*(10), 2075-2087.

[09] Schmidt, K. S., & Skidmore, A. K. (2003). Spectral discrimination of vegetation types in a coastal wetland. *Remote sensing of Environment*, *85*(1), 92-108.

[10] Ahlrichs, J. S., & Bauer, M. E. (1983). Relation of agronomic and multispectral reflectance characteristics of spring wheat canopies. *Agronomy Journal*, *75*(6), 987-993.

[11] Fernandez, S., Vidal, D., Simon, E., & SOLl3-SUGRANES, L. (1994). Radiometric characteristics of Triticum aestivum cv, Astral under water and nitrogen stress. *International Journal of Remote Sensing*, *15*(9), 1867-1884.

[12] Thenkabail, P. S., Smith, R. B., & De Pauw, E. (2000). Hyperspectral vegetation indices and their relationships with agricultural crop characteristics. *Remote sensing of Environment*, *71*(2), 158-182.

[13] Goetz, S. J., Prince, S. D., Small, J., & Gleason, A. C. (2000). Interannual variability of global terrestrial primary production: results of a model driven with satellite observations. *Journal of Geophysical Research: Atmospheres (1984–2012)*, *105*(D15), 20077-20091.

[14] Running, S. W., Thornton, P. E., Nemani, R., & Glassy, J. M. (2000). Global terrestrial gross and net primary productivity from the earth observing system. In *Methods in ecosystem science* (pp. 44-57). Springer New York.

[15] Diallo, O., Diouf, A., Hanan, N. P., Ndiaye, A., & Prevost, Y. (1991). AVHRR monitoring of savanna primary production in Senegal, West Africa: 1987-1988. *International Journal of Remote Sensing*, *12*(6), 1259-1279.

[16] Fassnacht, K. S., Gower, S. T., MacKenzie, M. D., Nordheim, E. V., & Lillesand, T. M. (1997). Estimating the leaf area index of north central Wisconsin forests using the Landsat Thematic Mapper. *Remote sensing of environment*, *61*(2), 229-245.

[17] Jakubauskas, M. E. (1996). Thematic Mapper characterization of lodgepole pine seral stages in Yellowstone National Park, USA. *Remote sensing of environment*, *56*(2), 118-132.

[18] Steininger, M. K. (2000). Satellite estimation of tropical secondary forest above-ground biomass: data from Brazil and Bolivia. *International Journal of Remote Sensing*, *21*(6-7), 1139-1157.

[19] Tucker, C. J., & Sellers, P. J. (1986). Satellite remote sensing of primary production. *International journal of remote sensing*, *7*(11), 1395-1416.

[20] Pettorelli, N., Vik, J. O., Mysterud, A., Gaillard, J. M., Tucker, C. J., & Stenseth, N. C. (2005). Using the satellite-derived NDVI to assess ecological responses to environmental change. *Trends in ecology & evolution*, *20*(9), 503-510.

[21] Tucker, C. J., Holben, B. N., Elgin Jr, J. H., & McMurtrey III, J. E. (1981). Remote sensing of total dry-matter accumulation in winter wheat. *Remote Sensing of Environment*, *11*, 171-189.

[22] Box, E. O., Holben, B. N., & Kalb, V. (1989). Accuracy of the AVHRR vegetation index as a predictor of biomass, primary productivity and net CO2 flux. *Vegetatio*, *80*(2), 71-89.

[23] Sellers, P. J., Heiser, M. D., & Hall, F. G. (1992). Relations between surface conductance and spectral vegetation indices at intermediate (100 m2 to 15 km2) length scales. *Journal of Geophysical Research: Atmospheres (1984–2012)*, *97*(D17), 19033-19059.

[24] Gilabert, M. A., Gandía, S., & Melia, J. (1996). Analyses of spectral-biophysical relationships for a corn canopy. *Remote Sensing of Environment*, *55*(1), 11-20.

[25] Tucker, C. J., Vanpraet, C. L., Sharman, M. J., & Van Ittersum, G. (1985). Satellite remote sensing of total herbaceous biomass production in the Senegalese Sahel: 1980–1984. *Remote sensing of environment*, *17*(3), 233-249.

[26] Diallo, O., Diouf, A., Hanan, N. P., Ndiaye, A., & Prevost, Y. (1991). AVHRR monitoring of savanna primary production in Senegal, West Africa: 1987-1988. *International Journal of Remote Sensing*, *12*(6), 1259-1279.

[27] Persson, P., Hall-Könyves, K., Sjöström, G., & Pinzke, S. (1993). NOAA/A VHRR data for crop productivity estimation in Sweden. *Advances in Space Research*, *13*(11), 111-116.

[28] Ramsey, E. W., & Jensen, J. R. (1996). Remote sensing of mangrove wetlands: relating canopy spectra to site-specific data. *Photogrammetric Engineering and Remote Sensing*, *62*(8), 939-948.

[29] Tan, Q., Shao, Y., Yang, S., & Wei, Q. (2003, July). Wetland vegetation biomass estimation using Landsat-7 ETM+ data. In *Geoscience and Remote Sensing Symposium, 2003. IGARSS'03. Proceedings. 2003 IEEE International* (Vol. 4, pp. 2629-2631). IEEE.

[30] Maier, H. R. and Dandy, G. C. (2000). Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental modelling & software*, *15*(1), 101-124.

[31] Dimopoulos, I., Chronopoulos, J., Chronopoulou-Sereli, A., & Lek, S. (1999). Neural network models to study relationships between lead concentration in grasses and permanent urban descriptors in Athens city (Greece). *Ecological Modelling*, *120*(2), 157-165.

[32] Chtioui, Y., Panigrahi, S., & Francl, L. (1999). A generalized regression neural network and its application for leaf wetness prediction to forecast plant disease. *Chemometrics and Intelligent Laboratory Systems*, *48*(1), 47-58.

[33] Ingram, J. C., Dawson, T. P., & Whittaker, R. J. (2005). Mapping tropical forest structure in southeastern Madagascar using remote sensing and artificial neural networks. *Remote Sensing of Environment*, *94*(4), 491-507.

[34] Werbos, P. J. (1988, July). Backpropagation: Past and future. In *Neural Networks, 1988., IEEE International Conference on* (pp. 343-353). IEEE.

[35] Hecht-Nielsen, R. (1989, June). Theory of the backpropagation neural network. In *Neural Networks, 1989. IJCNN., International Joint Conference on* (pp. 593-605). IEEE.

[36] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2004, July). Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on* (Vol. 2, pp. 985-990). IEEE.

[37] Huang, G. B., & Siew, C. K. (2005). Extreme learning machine with randomly assigned RBF kernels. *International Journal of Information Technology*, *11*(1), 16-24.

[38] Zhu, Q. Y., Qin, A. K., Suganthan, P. N., & Huang, G. B. (2005). Evolutionary extreme learning machine. *Pattern recognition*, *38*(10), 1759-1763.

[39] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, *70*(1), 489-501.

[40] Liang, N. Y., Saratchandran, P., Huang, G. B., & Sundararajan, N. (2006). Classification of mental tasks from EEG signals using extreme learning machine. *International Journal of Neural Systems*, *16*(01), 29-38.

[41] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, *70*(1), 489-501.

[42] Zhang, R., Huang, G. B., Sundararajan, N., & Saratchandran, P. (2007). Multicategory classification using an extreme learning machine for microarray gene expression cancer diagnosis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, *4*(3), 485-495.

[43] Nizar, A. H., Dong, Z. Y., & Wang, Y. (2008). Power utility nontechnical loss analysis with extreme learning machine method. *Power Systems, IEEE Transactions on*, *23*(3), 946-955.

[44] Schweizer, D., Armstrong, R. A., & Posada, J. (2005). Remote sensing characterization of benthic habitats and submerged vegetation biomass in Los Roques Archipelago National Park, Venezuela. *International Journal of Remote Sensing*, *26*(12), 2657-2667.

[45] Maier, H. R. and Dandy, G. C. (2000). Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental modelling & software*, *15*(1), 101-124.

[46] Hagan, M. T. and Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *Neural Networks, IEEE Transactions on*, *5*(6), 989-993.

[47] Battiti, R. (1992). First-and second-order methods for learning: between steepest descent and Newton's method. *Neural computation*, *4*(2), 141-166.

[48] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and

applications. *Neurocomputing*, *70*(1), 489-501.

CHAPTER 3: LEARNING CAPACITY OF THE EXTREME LEARNING MACHINES FOR VEGETATION SPECIES DIFFERENTIATION VIA REMOTE SENSING IMAGES

3.1 Introduction

The ability to select a correct machine learning tool to maximize its minimum learning capacity is a challenging task in systems analysis. Artificial Neural Network (ANN) algorithms have been using efficiently as a learning tool for different prediction and classification studies. In the past three decades, the backpropagation (ANN-BP) algorithm, combined with the gradient-based optimizations, has been widely used for training ANN models for varying purposes. Such a technique relies on iterative adjustment of all the network parameters (input weights, hidden layer biases and output weights) in a direction that minimize the total error between the target and the predicted output values [1,2]. Due to the underlying gradient-based optimization, however, the training process is typically slow and the convergence depends highly on the initialization as well as the learning rate; a poor selection of the learning rate can lead to slow convergence or even divergence. On the other hand, the Extreme Learning Machine (ELM) has recently received a considerable attention in various fields of optimization [3-16]. ELM is a fast and rigorous approach for training single-layer feedforward neural networks (SLFN). In principle, ELM analytically finds suitable output weights by randomly assigning values to the input weights. This is in contrast with the standard learning algorithms which iteratively search for optimized input and output weights. As a result, the training process in the ELM is much faster than the other approaches. ELM is known to avoid over-fitting that is an issue in traditional methods [6]. It has been also shown that an ELM with N hidden units (also called random units) can learn exactly N distinct observations with zero error [6]. Recently ELM has

been widely used for many applications from data mining in cancer diagnosis to loss analysis in electrical power systems and from sales forecasting to image processing [8,9,11,12]. ELM has been also used for several multi-class classification studies [6-8,10,12,13,14,16].

In this study, the learning capacity of ELM method was compared with that of the traditional ANN-BP method in a classification problem. Furthermore, three traditional statistical methods including Discriminant Analysis (DA), Minimum Distance (MD) and Parallelepiped (PD) were utilized for further comparison of the learning capacities between the machine learning algorithms and the statistical methods group wise. It is known that these three statistical methods have been used in numerous previous studies [17-24]. First of all, DA is one of the oldest methods for data classification [25]. It has been applied to many classification problems in statistics and in pattern recognition [26-28]. The basic idea of DA is to assign a normal distribution to each class which describes the probability of a sample being in that class. The parameters of each normal distribution can be obtained through the training samples belonging to each class. Once the probability distribution functions of each class is determined, any new sample can then be categorized to one of the classes based on its membership probability to that class. On the other hand, MD method uses the mean vectors for each class and that calssification can be made possible by calculating the Euclidean distance from each of the unknown sample to the mean vector of each class. New samples are normally classified to a class corresponding to the nearest mean vector. It is also possible to deploy a threshold value during the classification process depending on which the standard deviation or distance threshold can be measured. Fixing such threshold often creates unclassified values during the supervised learning process which can be mentioned as the deficiency of the MD method [29]. In the end, PD method is one of the simplest and primary classification processes. In such a supervised classification process,

the model is trained by calculating the upper and lower brightness values within the spectral dimension. The process is conducted by scrutinizing the histogram of individual spectral component in the training dataset. By utilizing the upper and lower bound several multidimensional boxes have been formed and unknown samples are classified depending on the multidimensional boxes they belong to [29].

In the current study, an attempt was made for the very first time to compare the learning capacity of different training algorithms across these two broad-based categories described above. For the purpose of comparison, an issue of vegetative species classification in a constructed wetland (CW) system has been conducted based on remote sensing satellite images and ground truth data. Remote sensing technology can provide comprehensive advantages over traditional species mapping method due to its versatility over a larger meadow for monitoring purpose as spatial variability is a crucial point for detail mapping works [34, 35]. Both multispectral and hyperspectral data were used before to carry out species level classification of vegetation. High resolution multispectral satellite images like Landsat Thematic Mapper (TM) or SPOT were used for ecosystem monitoring at descriptive resolution scale [36-38]. Spatial resolution of different satellite images limits their application towards different objectives depending on the scale of the study area, requirement of species details and cost-effectiveness. With ample budget, accurate marine mapping was conducted using 4m IKONOS satellite image [39]. To explore the spectral power, an investigation was also conducted using hyperspectral data to find out species level identification in a tropical rain forest [40]. Landsat TM data was also used to detect changes in seagrass communities and species composition in a shallow (< 2m) coastal lagoon [41]. Nowadays, there are numerous high resolution satellites available for both multispectral and panchromatic bands to deal with small study areas. Worldview-2 and Pleiades

satellite can provide a high resolution of 2m for multispectral and 0.5m for panchromatic bands. Traditional classification techniques need to be conducted by specialist to interpret the initial classification and sometimes those methods are not capable to interpolate the classified data to a scale finer than the spatial resolution of the associated image pixel [42]. With satellite imageries and ground sampling point along with the traditional supervised or unsupervised classification machine learning algorithms, it is entirely possible to develop a vegetation or land cover pattern.

In this study, a 50 cm panchromatic band image of Pleiades satellite image is utilized. As the panchromatic band of Pleiades satellite image consists only single band, the available options in traditional statistical classification methods narrowed down to MD, PD and DA as other traditional algorithms need multiple bands to conduct classification process. The classification of the image was based on the four different classes. Out of four classes of interest in this study, three of them are the vegetation species and the remaining one is the clear water. With this study objective, the following scientific questions are explored in this study: 1) Can machine learning classification methods compete with the traditional statistical methods? 2) Under the same conditions, how are the learning capacities of ELM and ANN-BP compared in training a single-layer feed-forward neural network?

3.2   Methodology

3.2.1   *Classification using machine learning and statistical methods*

3.2.1.1   *Traditional classification using minimum distance method*

For a limited number of training samples per class it is advantageous to use the mean position of the spectral classes instead of using the covariance information in order to attain more accurate estimation. Minimum distance (MD) classifier algorithm is such an approach where it uses the mean position of the spectral classes. In this method, the training sample information was used only to determine the class means. Following that the classification was performed by employing pixels in the class region of the nearest mean.

If $p_i(i = 1,2,...,M)$ are the means of P classes calculated from the training data samples, N is the threshold value for each class and x is the position of the pixel in spectral space to be classified then the Euclidean distances of the unknown pixel to each of the class mean can be found as [29]:

$$d(x,p_i)^2 = \left(x - p_i\right)^N (x - p_i) = (x - p_i) \cdot (x - p_i) \qquad\qquad (3.1)$$

Expanding the dot products form gives

$$d(x,pi)^2 = x \cdot x - 2p_i \cdot x + p_i \cdot p_i \qquad\qquad (3.2)$$

Classification is performed using the decision rule

$$x \in \omega_i \text{ if } d(x,p_i)^2 < d\left(x,p_j\right)^2 \text{ for all } j \neq i \qquad\qquad (3.3)$$

Since $x.x$ is common to all squared distance calculations it can be removed from both sides in the decision rule. The sign of the remainder can then be reversed so that the decision rule can be written as:

$x \in \omega_i$ if $g_i(x) > g_i(x)$ for all $j \neq i$ $\qquad$ (3.4)

where,

$g_i(x) = 2p_i \cdot x - p_i \cdot p_i$ $\qquad$ (3.5)

Eq. (5) is known as the discriminant function for the minimum distance classifier.

The implicit surfaces in spectral space separating adjacent classes are defined by the respective discriminant functions being equal. The surface between the i-th and j-th spectral classes is given by:

$g_i(x) - g_j(x) = 0$ $\qquad$ (3.6)

Substituting from Eq. (5) it gives:

$2(p_i - p_j) \cdot x - (p_i \cdot p_i - p_j \cdot p_j) = 0$ $\qquad$ (3.7)

This defines a linear surface, called a hyper-plane in more than three dimensions. The surfaces between each pair of classes define a set of first degree separating hyper-planes that partition the spectral space linearly. The minimum distance classifier nevertheless is effective when the number of training samples is limited or if linear separation of the classes is suspected.

3.2.1.2 *Traditional classification using parallelepiped method*

The parallelepiped classifier is the simplest form of classification under supervised classifier. It defines each class as an n-dimensional parallelepiped, where n is the number of spectral bands in the image. The method is a widely used decision based on the simple Boolean "and/or" logic. Each pixel's brightness values of certain satellite imagery can be used to produce n-dimensional mean vectors, $M_c = (\mu_{ck1}, \mu_{bck2}, \mu_{ck3}, \ldots, \mu_{ckn})$; where, μ_{ck} is the mean value of the training data derived for any class c in band k out of m possible classes. ENVI 5.0

software was used to perform the traditional supervised classifications based on the panchromatic image band. An example of two dimensional parallelepiped is shown in Figure 3.1.
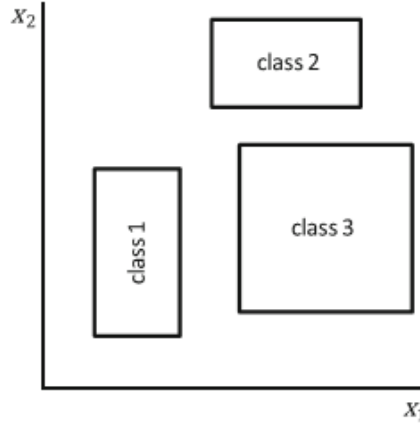
Figure 3.1: A set of two dimensional parallelepipeds

3.2.1.3   *Traditional classification using DA method*

Assuming there are K different N-dimensional samples $x^{(1)} = \left(x_1^{(1)}, x_2^{(1)}, \dots, x_N^{(1)}\right), \dots, x^{(K)} = \left(x_1^{(K)}, x_2^{(K)}, \dots, x_N^{(K)}\right)$, each belonging to one of the q different classes namely $c_1, c_2, \dots, c_q$. Of our interest would be to assign one of these classes to a new data point. We assumed that the samples are normally distributed to each class:

$$p(x \in c_i) = f(x; \mu_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_i|}} \exp\left(-\frac{1}{2}(x - \mu_i)\Sigma_i^{-1}(x - \mu_i)\right); \quad i = 1,2,\dots,q \qquad (3.8)$$

where μ_i and Σ_i represent the mean vector and the covariance matrix of the ith class respectively. These two latter can be found from the training samples which belong to the ith class. Now a new sample x' is classified in class c_j if the argument $(x' - \mu_i)\Sigma_i^{-1}(x' - \mu_i)$ is minimized for $i = j$.

3.2.1.4 *Neural network based classification*

ANN-BP and ELM algorithm were used for the neural network based classification. Detail methodologies of both algorithms are described in Chapter- 02.

3.2.2 *Statistical analysis*

In order to compare the agreement between the predicted species class and sampling species class a statistical inter-rater agreement called Kappa coefficient (K) and confusion matrix were used. Kappa measures the degree of agreement of the nominal or ordinal assessments made by multiple appraisers when assessing the same samples. Confusion matrix represents the number of correctly classified and misclassified samples of any classification process. Agreement is quantified by the Kappa (K) or Weighted Kappa (Kw) statistic [43] as K is 1 when there is perfect agreement between the classification system; K is 0 when there is no agreement better than chance; and K is negative when agreement is worse than chance. K does not take into account the degree of disagreement between observers and all disagreement is treated equally as total disagreement. Therefore when the categories are ordered, it is preferable to use Kw, and assign different weights w_i to subjects for whom the raters differ by i categories, so that different levels of agreement can contribute to the value of Kappa. All the inter-rater agreement analyses were performed using statistical software MedCalc [44]. Two sets of weights, called linear and quadratic were considered in that software. In the linear set, if there are k categories, the weights are calculated as follows:

$$w_i = 1 - \frac{i}{k-1} \qquad (3.9)$$

For the quadratic set the weights are calculated as below:

$$w_i = 1 - \frac{i^2}{(k-1)^2} \qquad\qquad (3.11)$$

When there are 5 categories, the weights in the linear set are 1, 0.75, 0.50, 0.25 and 0 when there is a difference of 0 (=total agreement) or 1, 2, 3 and 4 categories respectively. In the quadratic set the weights are 1, 0.937, 0.750, 0.437 and 0. Linear weights are normally used when the difference between the first and second category has the same importance as a difference between the second and third category, etc. If the difference between the first and second category is less important than a difference between the second and third category, etc., use quadratic weights. The coefficient K can be described in word as,

$$K = \frac{\text{probability of correct classification} - \text{probability of chance agreement}}{1 - \text{probability of chance agreement}} \qquad (3.12)$$

In any confusion matrix the probability of correct classification is given by the sum of the diagonal elements divided by the global total. If an entry of such matrix is n_{ij}, total number of samples is N and number of classes in M, and then the probability of correct classification is

$$p_o = \frac{1}{N}\sum_{i=1}^{M} n_{ii} \qquad\qquad (3.13)$$

The sum over the rows and columns of a confusion matrix is given by the following equation:

$$n_{+i} = \sum_{k=1}^{M} n_{ki} \quad n_{i+} = \sum_{k=1}^{M} n_{ik} \qquad\qquad (3.14)$$

then the probability that the sampling data and classifier respectively place a sample at random into class i are

$$p_{+i} = \frac{1}{N}\sum_{k=1}^{M} n_{ki} \qquad p_{i+} = \frac{1}{N}\sum_{k=1}^{M} n_{ik} \qquad\qquad (3.15)$$

so that the probability of their agreeing by chance on any of the available labels for a sample is:

$$p_c = \sum_{i=1}^{M} \left\{ \frac{1}{N} \sum_{k=1}^{M} n_{ki} \ \frac{1}{N} \sum_{k=1}^{M} n_{ik} \right\} \qquad\qquad (3.16)$$

According to the definition of coefficient, K:

$$K = \frac{p_o - p_c}{1 - p_c} \qquad\qquad (3.17)$$

3.2.3 *Study site and satellite image*

Constructed wetland system developed by the South Florida Water Management District (SFWMD) at the intersection between the Everglades Agricultural Area (EAA) and Everglades headwater plays a dominant role in maintaining the water quality around the large Everglades area. Around 230 square kilometers of the south of Lake Okeechobee has been converted to CW. Among all the CWs, STA-3/4 which comprises an area more than 64 square kilometers is known as the world's largest CW system. The main purpose of treatment wetland is to intercept the runoff entering the Everglades and reduce nutrient loading. Overall treatment performance depends on various factors such as land use, nutrient loading, vegetation condition, soil type, cell topography, cell size and shape, hydropattern (continuous flooding versus periodic dryout), maintenance and enhancement activities and regional operations (SFWMD, 2013). Vegetation condition and its distribution inside of CW area serve as a main biological mechanism of removing excess nutrients like phosphorus. Macrophyte communities and allied periphyton are the principal types of vegetation inside of such wetland system. Emergent and submerged macrophytes and macrophytic algae such as muskgrass (*Char sp.*) are responsible for phosphorus recycling and deposition through their physical, biological and biological interaction [30, 31]. Phosphorus retention and outflow concentration inside of CW are affected by the bidirectional movement and flux of phosphorus across the sediment-water interface. On the other hand,

phosphorus flux is affected by the movement of water through different vegetation species and nutrient uptake or release by different species [31, 32]. Macrophytes normally uptake nutrients from the sediment root zone and surrounding water column which eventually transported towards canopy part of the plant to continue the regular growth rate. Sometimes the macrophyte community move phosphorus towards the bottom part of the root and, thus increase the nutrient loading below ground surface than above ground [32]. Moreover, after completing the life cycles, the plant biomass and associated root zone release nutrients in the water column by decomposition. Besides that, mineralization of macrophytes also creates residual load of nutrient by their burial in the sediment. As different vegetation species plays their own role of extracting nutrients inside the wetland system, it is indeed important to understand the spatial distribution of such species. Collecting species information manually from all part of a wetland is not a feasible option as most of the wetlands are situated in under preserved natural condition. Manual data collection procedure like diver survey, underwater videography or acoustic transect method can produce erroneous result as such methods require interpolation of dataset [33]. Mapping the vegetation species based on satellite remote sensing technology can be an efficient technology to overcome the manual species data collection process.

3.2.4   *Data collection*

3.2.4.1   *Vegetation species and species data collection*

Two emerge and one submerge vegetation species where identified in the Cell-3B area. Emerge species are primarily two cattail (*Typha spp.*) types named as broadleaf cattail (Class-01) and southern cattail (Class-02). Broadleaf cattails are found in all around the United States and in temperate and tropical locations around the world. It is a kind of erect, semi-aquatic or aquatic

and perennial herb. The species is also a rhizomatous type of plant where the rhizomes are located 7.62 – 10.16 centimeters below the soil surface. It can grow up to 0.70 meter height and the diameter ranges between 0.5 to 3.05 centimeters. Broadleaf cattail can reproduce in both sexually and asexually. Reproduction normally occurs through the rhizome system which is accountable for the spreading out of prevailing stands. The species can invade in to the new area through sexual reproduction via seed dispersal. For germinating seeds the species needs to have a favorable condition like moist/ wet substrate, warm temperature, low oxygen concentration and an exposure of long day light hour. Southern cattail is also a rhizomatous, perennial plant with long green stalks and can grow up to 2.74 – 3.05 meters in height. This species is the dominant emerge vegetation in the current study area. The species consist of separate pistillate and staminate flowering spikes on the same axis. Both of the spikes can grow up to 15.24 – 25.40 centimeters. Southern cattail contains a spindle-shaped achene fruit which can be easily deciduous and dehiscent in water. The species is normally found at the edge of ponds, lakes, freshwater marshes across the entire southern tier of the states to the Atlantic Coast, the Caribbean, South America and Europe. The only submerge species (Class-03) found in the study area is muskgrass (*Chara spp.*). It has foul, musty and garlic-like odor. It is a gray-green branched multicellular algae and normally do not extend beyond the water surface. The species doesn't contain any flower and has some kind of rough texture. The body of this species forms a cylindrical shape with whorled branches coming out from the nodes. Number of branches associated with each nodes can goes from 6 -16. The submerged portion of the aquatic plant normally provides the habitats for micro and macro invertebrates. All three species are shown in Figure 3.2.

Figure 3.2: Broadleaf cattail-[Class-01] (left), southern cattail-[Class-02] (middle) and muskgrass-[Class-03] (right) species

A field trip was organized on 21st May, 2014 to collect the species information mentioned above. Using an air boat almost all parts of the study area was observed and different species pictures were taken using a digital camera along with their positions using a hand held GPS. At each sampling location the average GPS accuracy was around 9 meters. Total forty five (45) species information was collected from the field trip. In addition to that, by utilizing the field information, sixty four (64) species information was collected using Google Earth. Finally, total 109 species information was organized for subsequent analysis. All field species sampling locations are shown in Figure 3.3.
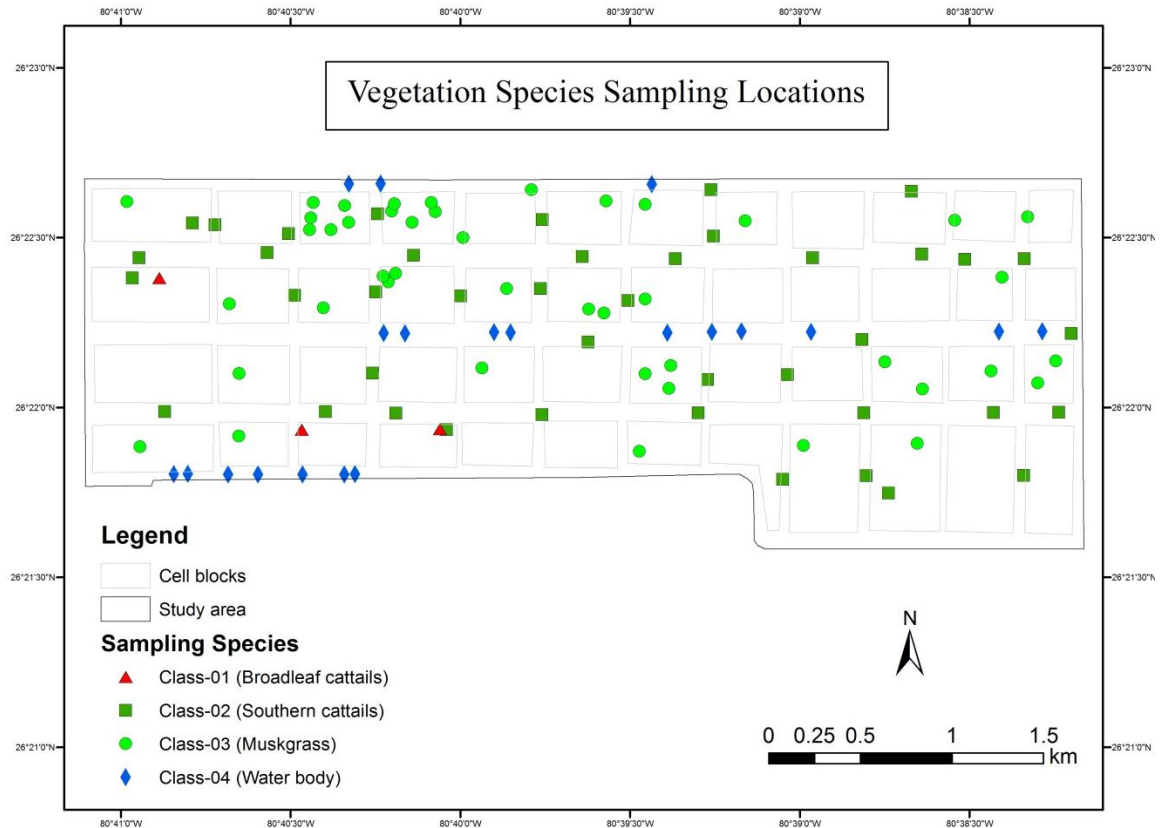
Figure 3.3: Vegetation species sampling locations

3.2.4.2 *Satellite image collection*

Pleiades satellite panchromatic image was acquired for the same day as the field data sampling campaign. Pleiades was a cooperation program initiated by France and Italy which provides an earth observation system with metric resolution. Two Pleiades satellite were launched as Pleiades 1A and Pleiades 1B. The launching time of Pleiades 1A and Pleiades 1B satellite were at the fourth quarter of year 2011 and 2012 respectively. In this study Pleiades 1B satellite image with zero cloud coverage was used. Pleiades offer 0.50 m resolution for its panchromatic band and 2 m resolution for its multispectral bands. The panchromatic band of the

satellite image includes only the black and white band. The wavelength for this band ranges between 0.47 to 0.83 μm of the visible spectrum.

3.2.4.3   *Data arrangement for predicting vegetation species*

Two different data samples were prepared for this study. Point based dataset was used for training different algorithms and finding the efficiency in the learning capacity. Total 109 data samples were used for that purpose. Another training dataset was prepared depending on the pixel based information. For developing a pixel based training sample, a circle with diameter of 9 meter (average GPS accuracy) was drawn around each of the 109 sampling points. Following that, those circles were overlaid on the satellite image. And finally, the pixel information corresponding to each circle was cropped out along with their associated positions (latitude and longitude) to form a pixel based dataset for subsequent species classification. Both point based and pixel based training samples are shown in Table 3.1.

Table 3.1: Number of samples associated to each class

Species	Class	Number of training samples used for comparing learning capacity	Number of new training samples used for developing prediction map
Broadleaf Cattails	Class-01	3	3318
Southern Cattails	Class-02	42	46539
Muskgrass	Class-03	44	48607
Water	Class-04	20	22049

3.3   Result and Discussion

    In this section we present the classification results obtained from different approaches that we discussed in this paper. In this regard, a particular attention has been paid to a comparison between ANN-BP and ELM since both methods are learning algorithms for feed-forward neural networks. Therefore a detailed comparison between the learning capacities of these two approaches is presented in this section. On the other hand, the classification capabilities of DA, Minimum distance method and Parallelepiped which are all among the traditional classifiers and by nature different from the neural-network-based methods are also discussed in detail. According to our results, in general, neural network-based methods outperform the traditional schemes and between the two neural networks training algorithms ELM performs slightly better than ANN-BP.

In this work, as a major criterion for presenting our results, we define the classification accuracy as fraction of correct class predictions over the total predictions. All the results are based on 109 vegetation samples which are from 4 different classes. Each sample has three properties; the latitude and longitude coordinates as well as the reflection. In most cases we divide these observations into training and test samples based on a splitting ratio of 0.7 for the training. This means that 70% of the available samples are used as training data while the remaining samples are used to test the accuracy of the trained network. The effect of this splitting ratio on the classification performance is also studied at the end of this section.

3.3.1   ANN-based methods

    In order to investigate the effect of the number of hidden neurons on the operation of the neural-network-based methods (ANN-BP and ELM), first 70% of the

total data were chosen randomly to be used for the training and the remaining 30% for the test. The training and test accuracies are recorded for a given number of hidden units. The number of hidden neurons is then changed from 1 to 110, and for each case the same process is repeated to calculate the training and test accuracies. Figure 6 depicts the classification accuracies of both ANN-BP and ELM as a function of the number of hidden units and for both the training and test data. As shown in this figure none of the two models can produce acceptable accuracies for less than 10 hidden units. As the number of hidden neurons increases in the given range, ELM produces a better accuracy for the training and test data. On the other hand, both the training and test accuracies obtained from ANN-BP, while fluctuating around an average value, relatively decrease for a large number of hidden units.
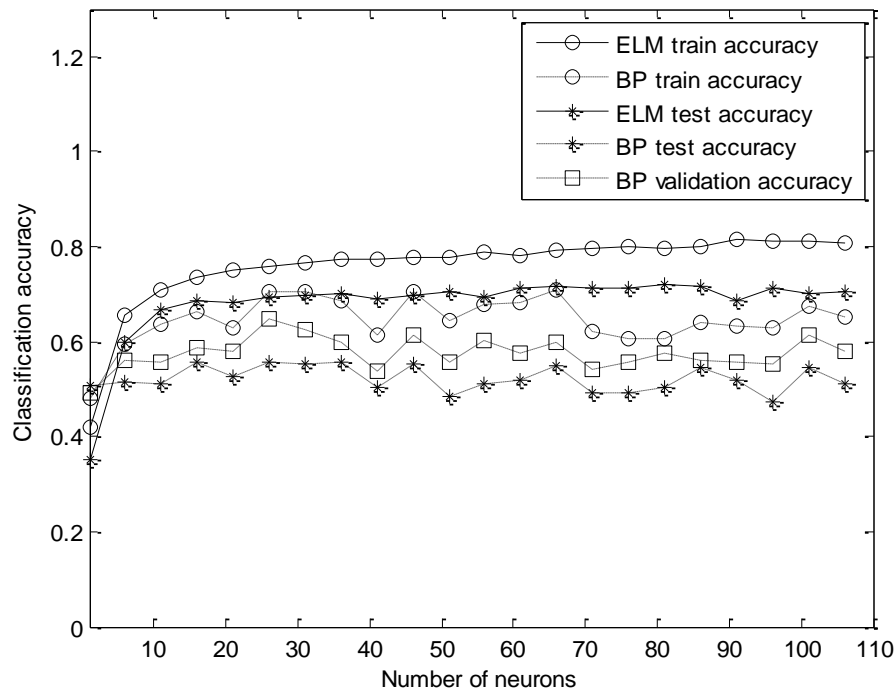
Figure 3.4: The classification accuracies of ANN-BP and ELM for the both the training and test samples and for different numbers of hidden units.

In general in a single-layer fully connected feed-forward neural network trained with BP, the learning capacity is governed by $L \leq K \leq \left(1 + \frac{N}{M}\right) L$ [48, 49]. This rule of thumb simply states that for a given number of training samples K, the number of hidden units L should be more than a lower bound $L_{min} = K/\left(1 + (N/M)\right)$ and still less than an upper bound that is $L_{max} = K$. As a result selecting a large number of hidden units does not always result in a better accuracy. In fact, in such case, even though the training accuracy may increase, the test accuracy will decrease due to the over-fitting. Interestingly, however, this problem is circumvented in ELM as shown in Figure 3.4.

As mentioned in the methodology section, a major advantage of ELM over other training algorithms is the simplicity of its algorithm. From a computational point of view, the only time-consuming part of ELM is the calculation of the Moore-Penrose inverse of the matrix H (which exhibits $K \times L$ elements). In ANN-BP, on the other hand, all the input weights, hidden neurons and output weights $((N + M + 1) \times L$ elements in total) need to be updated iteratively as long as their values converge. Meanwhile each update depends on the partial derivative of the total error which itself is obtained in terms of a summation containing the entire K samples. Therefore ELM is expected to be a much faster algorithm compared to ANN-BP and many other machine learning algorithms. Figure 3.5 compares the computational times of ELM and ANN-BP as a function of the number of the hidden neurons. These results are obtained in a desktop computer with a Core i7 processor at 3.30 GHz, an installed memory (RAM) of 32 GB and the operating system is 64-bit windows seven. As this figure clearly indicates, by increasing the number of hidden units the computational time of the

ANN-BP training algorithm grows rapidly and its difference with that of ELM becomes more prominent. This becomes even more evident when a large number of samples $(K > 100,000)$ are to be used. In such conditions eventually ANN-BP becomes impractical while ELM can still be used.
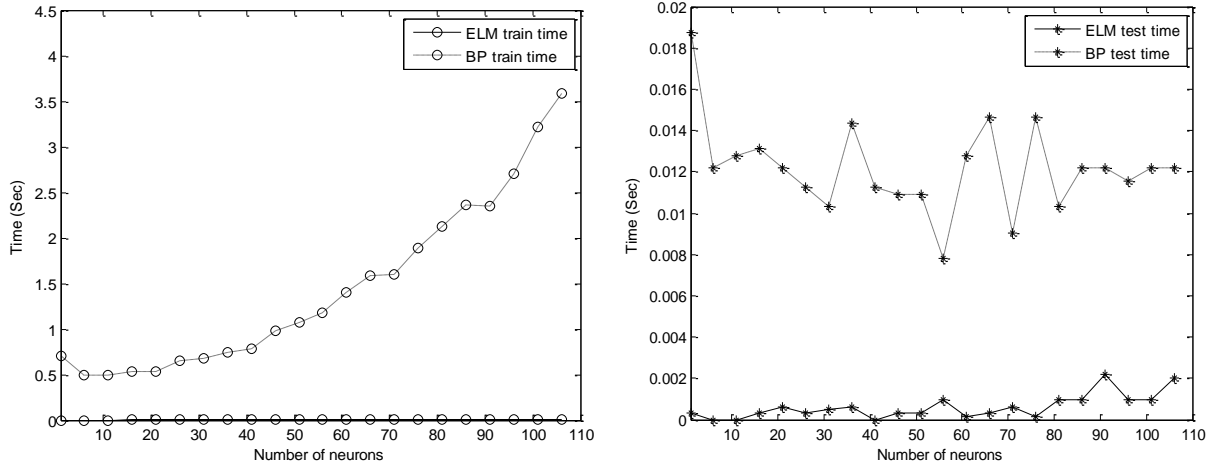
Figure 3.5: Computational time for the training (left) and test (right) processes of ELM and ANN-BP as a function of the number of the hidden units.

Note that for the single-layer feed-forward neural network discussed in this paper, several different activation functions can be utilized in the hidden units and both ANN-BP and ELM can utilize many types of activation functions. Even though ANN-BP training algorithm, because of the error gradients, can admit only differentiable activation function, ELM is not limited to this restriction and instead can utilize any bounded activation function. Here we used the sigmoid function $(g(x) = (1 + \exp(-x))^{-1})$, the sinusoidal function $(g(x) = \sin(x))$ and finally the Heaviside's step function $(g(x) = 0, 1$ for negative and positive values of x respectively) as the activation functions. The training and test accuracies versus different activation functions are shown in Figure 8. For these results, the number of hidden neurons and the splitting ratio for training are set to be 80 and 0.7 respectively. As it can be seen in Figure 3.6, among the aforementioned activation functions, the sinusoidal function gives the best accuracy.

58
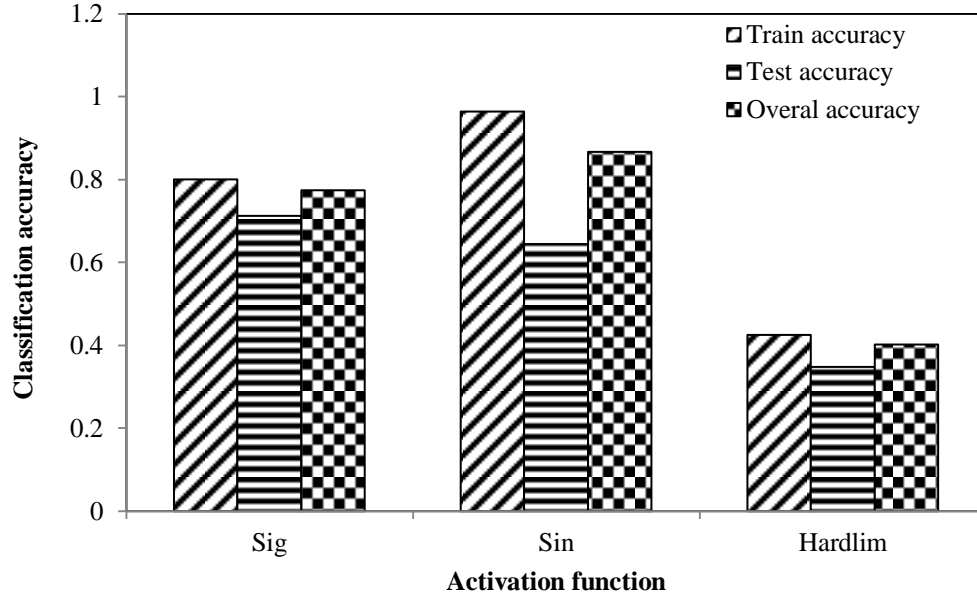
Figure 3.6: Training and test accuracies of the ELM model for different activation functions

It is also worth noting that in general there are several different types of the back-propagation algorithms each using a different optimization technique for minimizing the backward propagation error [50-54]. Evidently of interest would be to see how such schemes compete with each other and with ELM in the vegetation species classification problem. So far in this work, we presented the results from the Levenberg-Marquardt back-propagation (LMBP) method. Here we compare the classification results obtained from three other different ANN-BP algorithms; Bayesian Regulation back-propagation (BRBP), Scaled Conjugate Gradient back-propagation (SCGBP) and Resilient back-propagation (RPROP). All these methods are used form the Neural Network Toolbox of MATLAB 2012b [45]. Perhaps this should be mentioned somewhere else. Figure 3.7 compares the classification accuracies of these four methods under the same conditions. In addition the associated computational times are also shown in Figure 3.8.
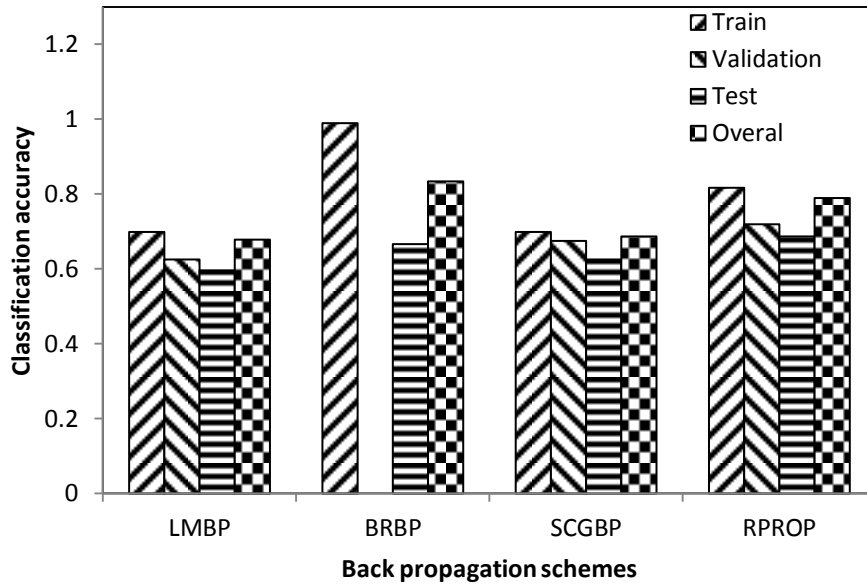
Figure 3.7: Train and test accuracies for different back propagation methods
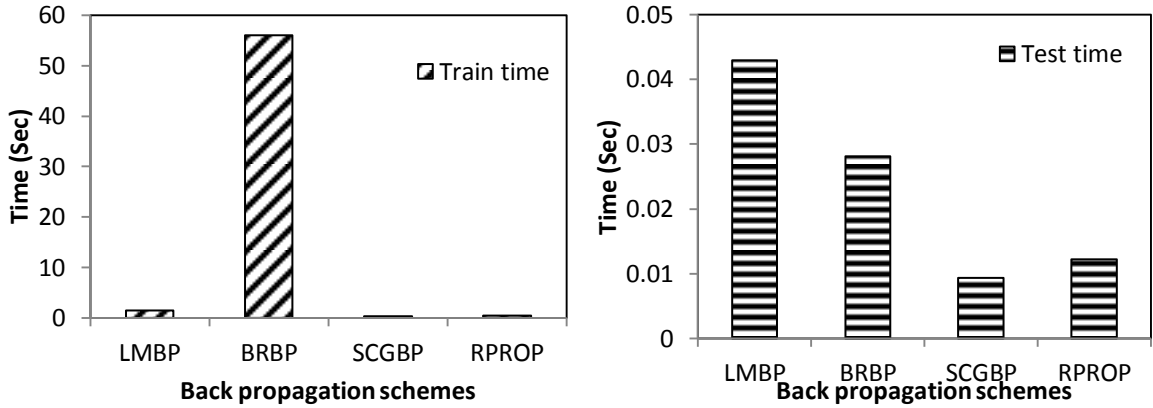
Figure 3.8: Train and test computational times for different back propagation methods

Before ending this part, it is worth noting that for each method, even for a fixed number of hidden units, different classification accuracies may be obtained at different runs and under the exact same conditions. This is due to the random weight initializations in both methods. In order to minimize the effect of this random starting point, each run is repeated for 100 time and the average value was presented in all results presented so far. To investigate the effect of random

initializations on classification accuracies, their variations in different runs are plotted in Figure 3.9 for both ANN-BP and ELM. Obviously the results obtained from ANN-BP vary in a wider range while ELM is more robust with respect to the random initializations especially for the training data.
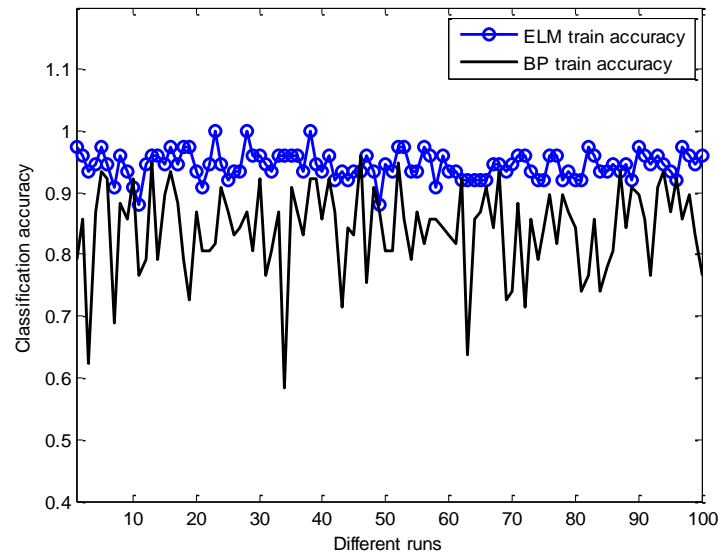
(a)
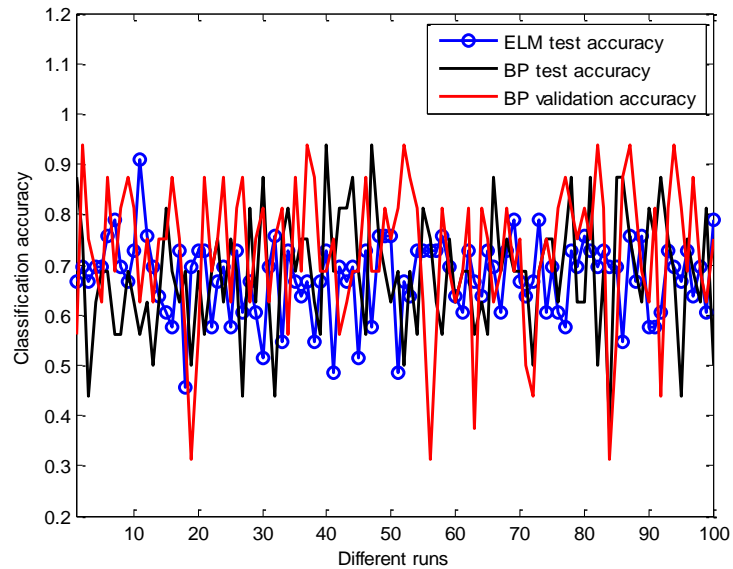
(b)

Figure 3.9: Training (a) and test (b) classification accuracies of ELM and ANN-BP for a specific neural network but for different runs. Here 20 and 80 hidden neurons are used for the ANN-BP and ELM respectively (inputs and outputs data are chosen randomly).

3.3.2 *Traditional classification method*

As described in the methodology, the discriminant analysis is by nature different from the neural network models. On the other hand it can also be considered as a machine learning approach since it involves the training and test processes. As we will show here, DA can be used as a fast and accurate method which is to some extent comparable with the neural-network-based models. Here, we performed the classification of the 109 observations with five different variations of the discriminant analysis methods which are the linear (LDA), diagonal linear (DLDA), quadratic (QDA), diagonal quadratic (QDA) and the discriminant analysis based on Mahalanobis distance (MDA) [26-28]. The main difference between these methods lies with the covariance matrices of the underlying multivariate normal distributions assigned to each class. For example in QDA each normal distribution exhibits a different covariance matrix while in LDA all classes share the same matrix of covariance. The changes of the training and test accuracies for different DA schemes are investigated and shown in Figure 3.10. For all cases the splitting ratio for training is assumed to be 0.7. According to this figure, among these five different approaches the Mahalanobis distance (MDA) results in the highest accuracy.
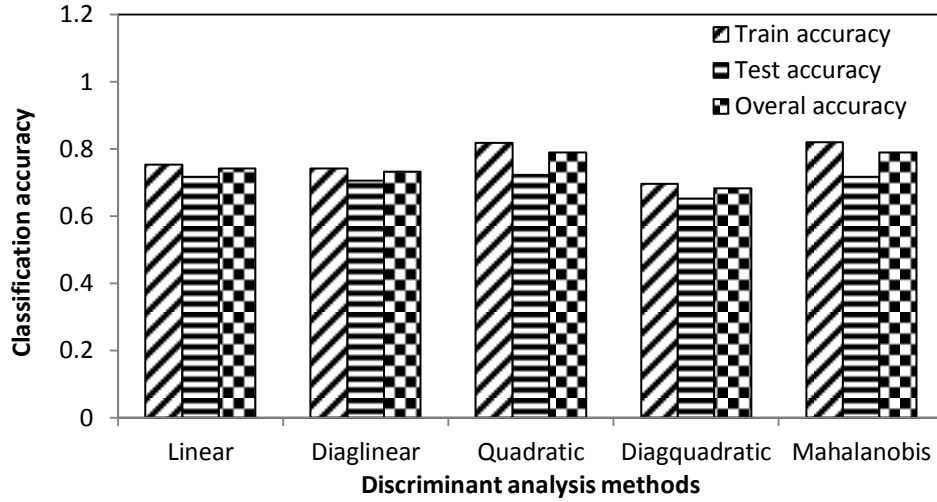
Figure 3.10: The training and test accuracies for different discriminant analysis methods

For the MD and Parallelepiped methods, four type of splitting ratios were considered as 20%, 50%, 70% and 80%. The classification accuracy results for both training and test dataset shown in Table 3.2. For the MD method of classification, the model showed a surprising accuracy of 100% for all the training data ratios. The reason behind such results lies in the model structure of the MD method. During the classification of stage, MD method normally used two parameters known as the mean vector (m) and covariance matrix for each spectral class. In order to estimate the covariance matrix accurately, there needs to be a sufficient number of sampling datasets. Calculating covariance matrix using limited number of sampling data will lead towards low accuracy of training model. However, when the sampling data is limited, the MD algorithm does not make use of the covariance information rather it calculates the mean position of spectral classes. Considering only the mean position of spectral classes generally increase the training accuracy significantly which was also observed in the current study. The results for the test dataset using MD method showed a moderate accuracy because, this method always assigns a value for each point

depending on the mean value comparison which might not always give a similar result for independent test dataset; especially when the sampling data are limited.

For the PD method, both the training and test data showed pretty low accuracies. The reason of having such low accuracy using this method is due the unclassified pixels in the training process. In PD method, unclassified pixels are developed due to the gaps situated between the parallelepiped of the spectral classes in spectral space. On the other hand, overlapping between the parallelepiped also creates some missing classification information. Both of those phenomena decrease the overall accuracy of the classification using PD method especially when the number of sampling dataset is limited for the analysis.

Table 3.2: Classification accuracy using MD and PD method

Splitting Ratio for Training	Classification Accuracy (MD)		Classification Accuracy (PD)	
	Training	Test	Training	Test
0.2	100.00	64.37	36.36	49.43
0.5	100.00	61.11	45.45	40.74
0.7	100.00	57.58	65.36	39.60
0.8	100.00	59.09	72.36	36.36

Finally, we have also considered the effect of different splitting ratios for training on the classification accuracies of the discussed methods. This is depicted in (b)

Figure 3.11 for ANN-BP, ELM, DA, MD and PP while a constant number of hidden neurons (20 for ANN-BP and 80 for ELM) are used for the first two approaches. According to this figure, for the ELM algorithm, at low splitting ratios the classification accuracy of the

training data is very high while it is relatively poor for the test data. By increasing the splitting ratio, the training accuracy decreases while at the same time the test accuracy increases until reaching a proper ratio when both accuracies are acceptable. The behavior of ANN-BP and DA are relatively the same in this figure.
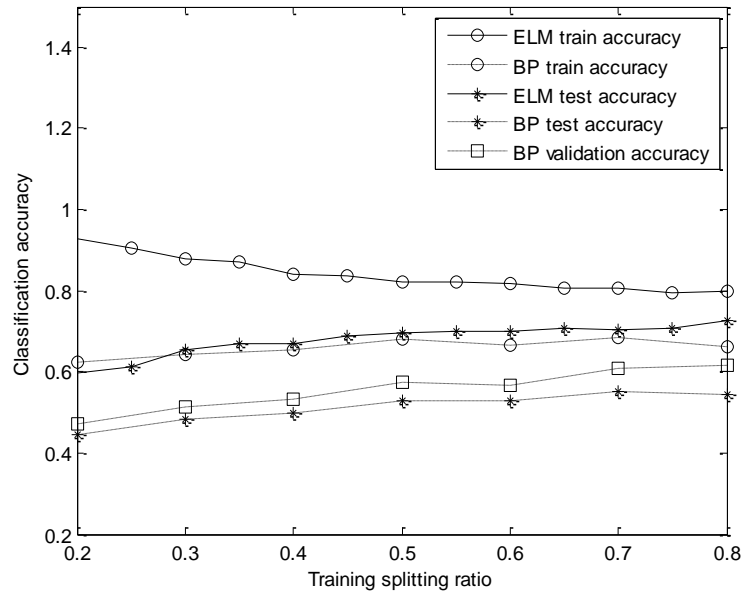
(a)
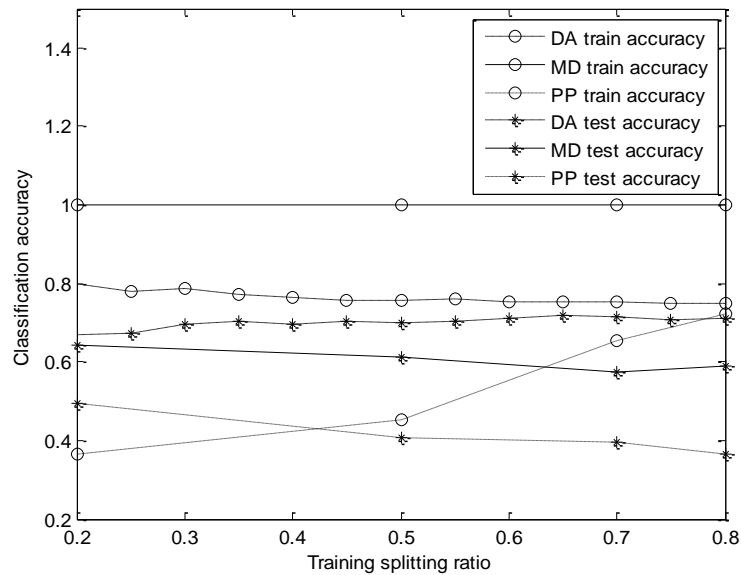
(b)

Figure 3.11: The effect of different splitting ratio for training on classification accuracies of ANN-BP and ELM (a), DA, MD, and PP (b)

Based on our studies so far, the best results of each model are summarized in Table 3.3. According to this table, in general, ANN-based models perform better than the traditional approaches while ELM also outperforms ANN-BP both in classification accuracy and in computational time.

Table 3.3: The classification accuracies of the best results obtained from all classification method

Model	Classification accuracy				Computational time (sec)		
	Training	Validation	Test	Overall	Training	Test	Total
ELM	0.94	----	0.67	0.86	0.004	0.001	0.005
ANN-BP	0.84	0.72	0.68	0.81	0.731	0.018	0.748
DA	0.82	----	0.72	0.79	0.005	0.002	0.007
MD	1.00	----	0.57	0.87	27.09	2.5	29.59
PP	0.65	----	0.39	0.68	37.71	3.0	40.71

3.3.3 *Statistical comparison*

In order to conduct a statistical comparison between different methods the inter-rater agreement coefficient Kappa (K) along with the confusion matrix for predicted and sampling species were developed. The confusion matrix for each method and associated K values are shown in Table 3.4. As ANN-BP, ELM and DA methods needed to go through several iterations, the value of K and confusion matrix for those methods were generated by taking an average over the iterations. Total 100 iterations using 109 samples were considered for ANN-BP, ELM and DA which comprised 10900 (100×109) samples in total. Final value of K and confusion matrix for MD and PD were developed by taking an average of those values for different training data ratio. The diagonal elements of the confusion matrix represents the number of correctly classified samples whereas, the remaining values in the matrix represents the number of mis-classifications. Considering the confusion matrix for two machine learning algorithms, ANN-BP

66

method correctly predicted 71, 3085, 3505 and 1877 number of samples for species-1, species-2, species-3 and species-4 respectively. Whereas, ELM method correctly predicted 227, 3467, 3902 and 1875 number of samples for species-1, species-2, species-3 and species-4 respectively. Besides that, among the traditional classification methods, prediction accuracy for DA method was much better comparing to the prediction result found using MD and PD. The percentage of correctly predicted samples for different methods is shown in Table 3.5. ELM method showed the highest prediction accuracy for all the species classes except water, though the percentage of correctly classified water sample is still more than 90% which was quite satisfactory. Similar trend was also observed with K values. Though all the K values for ANN-BP, DA, ELM and MD fell inside of a good agreement [46] zone, the ELM outclassed the other methods by having a K value of 0.80. The lowest K value was observed for PD method of 0.39 mainly due to the unclassified samples during the training process.

Table 3.4: Confusion matrix and kappa coefficient for different classification methods

Machine Learning							
Method	Predicted Species	Actual Species				Total	Kappa
		Species-1	Species-2	Species-3	Species-4		
ANN-BP	Species-1	**71**	27	3	4	105	0.67
	Species-2	53	**3085**	849	15	4002	
	Species-3	138	1028	**3505**	104	4775	
	Species-4	38	60	43	**1877**	2018	
	Total	300	4200	4400	2000	10900	
ELM	Species-1	**227**	109	30	9	352	0.80
	Species-2	57	**3467**	451	55	4053	
	Species-3	16	611	**3902**	61	4590	
	Species-4	0	13	17	**1875**	1905	
	Total	300	4200	4400	2000	10900	
Traditional Methods							
Method	Predicted Species	Actual Species				Total	Kappa
		Species-1	Species-2	Species-3	Species-4		
DA	Species-1	**224**	0	0	0	224	0.67
	Species-2	41	**3353**	1392	13	4799	

Machine Learning							
Method	Predicted Species	Actual Species				Total	Kappa
		Species-1	Species-2	Species-3	Species-4		
	Species-3	35	847	**3008**	39	3929	
	Species-4	0	0	0	**1948**	1948	
	Total	300	4200	4400	2000	10900	
MD	Species-1	**6**	0	36	0	42	
	Species-2	0	**128**	15	0	143	
	Species-3	6	40	**123**	8	177	0.75
	Species-4	0	0	2	**72**	74	
	Total	12	168	176	80	436	
PD	Species-1	**9**	36	43	1	89	
	Species-2	2	**126**	73	4	205	
	Species-3	1	6	**59**	23	89	0.39
	Species-4	0	0	1	**52**	53	
	Total	12	168	176	80	436	

Table 3.5: Prediction accuracies (%) for all classes using different classification methods

Types	Class-01	Class-02	Class-03	Class-04
ANN-BP	23.67	73.45	79.66	93.85
ELM	75.67	82.55	88.68	93.75
DA	74.67	79.83	68.36	97.40
MD	50.00	76.20	69.89	90.00
PD	75.00	75.00	33.52	65.00

3.3.4   *Vegetation prediction over the whole constructed wetland*

The satellite images of the constructed wetland of this case study include the reflectance value for 3923×10597 pixels. Clearly, classification of such huge number of data pixels requires a considerable amount of training samples. As a result, the pixel based training samples which consist of 120333 pixels from different parts of the constructed wetland. In this case the ANN-BP algorithm can no longer be used due to its huge computational type. Here based on the

training samples, we can predict the vegetation species for all these pixels by using ELM as depicted in Figure 3.12. Form the classification result it was observed that, most of the areas in the wetland are covered with submerged aquatic vegetation species or class-03 (muskgrass). Numbers of pixels predicted as class-01 were extremely poor as the numbers of training samples under such class were very limited comparing to the other classes.
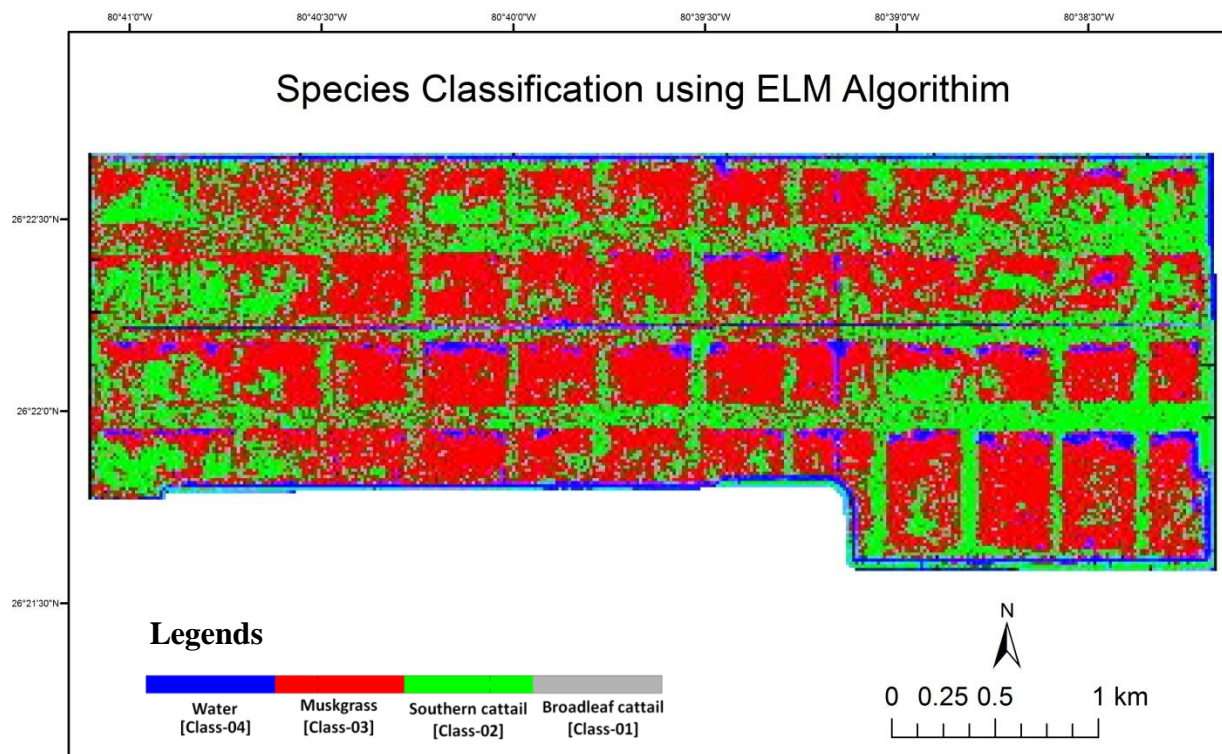
Figure 3.12: Species classification map using ELM algorithm

In order to conduct a statistical comparison between different methods the inter-rater agreement coefficient Kappa (K) along with the confusion matrix for predicted and sampling

3.4 Conclusion

In this work, we exploited different machine learning and statistical schemes for the classification of vegetation species in a constructed wetland (CW). CWs are certainly a diversified eco-system which consist several types of vegetation species. Different species has its

own distinct nutrient intake limit. From the view point of treatment efficiency of such eco-system, it is indeed important to signify the types of vegetation species in order to achieve a nutrient removal target. The implication of machine learning algorithm in quantification of vegetation species certainly open up a new way on that regards.

We performed a detailed comparison between the learning capacities of BP and ELM in training a single-layer feed-forward neural network. The effect of different number of hidden neurons, activation functions and the splitting ratio of training where investigated while various BP algorithms where also considered. According to this study, in general ELM performs better than BP both in classification accuracy and in computational time. In addition when the training data are selected randomly from a given set of samples, at different runs the classification accuracies of the ELM vary in a smaller range compared to BP. This counterintuitive result shows the robustness of ELM with respect to initializations in spite of the fact that in each run the input weights and biased are assigned randomly. In our study we have also considered three traditional classification algorithms (DA, MD, and PD). Even though such methods can be easily implemented, their classification accuracies are in general less than the machine learning algorithms. Our results reveal the great potential of the extreme learning machine training algorithm in environmental data mining applications. In particular, this method can be very useful in cases where large sets of training samples are concerned. In regards to the sampling data collection, the accuracy can be increased by using a more robust GPS system e.g., RTK or differential GPS instead of hand held GPS. Moreover increasing the number of sampling points will always help to train a machine learning model.

3.5 References

[1] Werbos, P. J. (1988, July). Backpropagation: Past and future. In *Neural Networks, 1988., IEEE International Conference on* (pp. 343-353). IEEE.

[2] Hecht-Nielsen, R. (1989, June). Theory of the backpropagation neural network. In *Neural Networks, 1989. IJCNN., International Joint Conference on* (pp. 593-605). IEEE.

[3] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2004, July). Extreme learning machine: a new learning scheme of feedforward neural networks. In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on* (Vol. 2, pp. 985-990). IEEE.

[4] Huang, G. B., & Siew, C. K. (2005). Extreme learning machine with randomly assigned RBF kernels. *International Journal of Information Technology*, *11*(1), 16-24.

[5] Zhu, Q. Y., Qin, A. K., Suganthan, P. N., & Huang, G. B. (2005). Evolutionary extreme learning machine. *Pattern recognition*, *38*(10), 1759-1763.

[6] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, *70*(1), 489-501.

[7] Liang, N. Y., Saratchandran, P., Huang, G. B., & Sundararajan, N. (2006). Classification of mental tasks from EEG signals using extreme learning machine. *International Journal of Neural Systems*, *16*(01), 29-38.

[8] Zhang, R., Huang, G. B., Sundararajan, N., & Saratchandran, P. (2007). Multicategory classification using an extreme learning machine for microarray gene expression cancer diagnosis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, *4*(3), 485-495.

[9] Nizar, A. H., Dong, Z. Y., & Wang, Y. (2008). Power utility nontechnical loss analysis with extreme learning machine method. *Power Systems, IEEE Transactions on*, *23*(3), 946-955.

[10] Rong, H. J., Ong, Y. S., Tan, A. H., & Zhu, Z. (2008). A fast pruned-extreme learning machine for classification problem. *Neurocomputing*, *72*(1), 359-366.

[11] Sun, Z. L., Choi, T. M., Au, K. F., & Yu, Y. (2008). Sales forecasting using extreme learning machine with applications in fashion retailing. *Decision Support Systems*, *46*(1), 411-419.

[12] Suresh, S., Venkatesh Babu, R., & Kim, H. J. (2009). No-reference image quality assessment using modified extreme learning machine classifier. *Applied Soft Computing*, *9*(2), 541-552.

[13] Huang, G. B., Ding, X., & Zhou, H. (2010). Optimization method based extreme learning machine for classification. *Neurocomputing*, *74*(1), 155-163.

[14] Suresh, S., Saraswathi, S., & Sundararajan, N. (2010). Performance enhancement of extreme learning machine for multi-category sparse data classification problems. *Engineering Applications of Artificial Intelligence*, *23*(7), 1149-1157.

[15] Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., & Lendasse, A. (2010). OP-ELM: optimally pruned extreme learning machine. *Neural Networks, IEEE Transactions on*, *21*(1), 158-162.

[16] Huang, G. B., Zhou, H., Ding, X., & Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *42*(2), 513-529.

[17] Laine, A., & Fan, J. (1993). Texture classification by wavelet packet signatures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *15*(11), 1186-1191.

[18] Bellman, R., Kalaba, R., & Zadeh, L. (1966). Abstraction and pattern classification. *Journal of Mathematical Analysis and Applications*, *13*(1), 1-7.

[19] Winters, J. H., & Rose, C. (1989). Minimum distance automata in parallel networks for optimum classification. *Neural Networks*, *2*(2), 127-132.

[20] Kettig, R. L., & Landgrebe, D. A. (1976). Classification of multispectral image data by extraction and classification of homogeneous objects. *Geoscience Electronics, IEEE Transactions on*, *14*(1), 19-26.

[21] Lin, H., & Venetsanopoulos, A. N. (1993, September). A weighted minimum distance classifier for pattern recognition. In *Electrical and Computer Engineering, 1993. Canadian Conference on* (pp. 904-907). IEEE.

[22] Albert, T. H. (2002). Evaluation of remote sensing techniques for ice-area classification applied to the tropical Quelccaya Ice Cap, Peru. *Polar Geography*, *26*(3), 210-226.

[23] Turker, M., & Derenyi, E. (2000). GIS assisted change detection using remote sensing. *Geocarto International*, *15*(1), 51-56.

[24] Miranda, F. P., MacDonald, J. A., & Carr, J. R. (1992). Application of the semivariogram textural classifier (STC) for vegetation discrimination using SIR-B data of Borneo. *International Journal of Remote Sensing*, *13*(12), 2349-2354.

[25] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, *7*(2), 179-188.

[26] Cacoullos, T. (1973). Discriminant analysis and applications. New York: Academic Press.

[27] Seber, G. A. (2009). *Multivariate observations* (Vol. 252). John Wiley & Sons.

[28] McLachlan, G. (2004). *Discriminant analysis and statistical pattern recognition* (Vol. 544). John Wiley & Sons.

[29] Richards, J. A., & Richards, J. A. (1999). *Remote sensing digital image analysis* (Vol. 3). Berlin et al.: Springer.

[30] Brix, H. 1997. Do macrophytes play a role in constructed treatment wetlands? Water Sci and Tech., 35:11-17.

[31] Kadlec, R.H. and S.D. Wallace. 2009. Treatment Wetlands, 2nd Edition. CRC Press, Taylor and Francis Group, Boca Raton, FL. 1016 pp.

[32] Reddy, K.R. and R.D. DeLaune. 2008. Biogeochemistry of Wetlands: Science and Applications. CRC Press, Boca Raton, FL

[33] Norris, J. G., & Wyllie-Echeverria, S. (1997). Estimating maximum depth distribution of seagrass using underwater videography.4th Int. Conf. Remote Sens. Mar. Coastal Environ., Orlando, Florida, 17 – 19 March 1997, Vol. I(pp. 603 – 610)

[34] Duarte, C. M., Alvarez, E., Grau, A., & Krause-Jensen, D. (2004). Which monitoring strategy should be choosen? In J. Borum, C. M. Duarte, D. Krause-Jensen, & T. M. Greve (Eds.),European seagrasses: An introduction to monitoring and management, The M&MS project, September 2004, Internet version at:http://www.seagrasses.org.

[35] Krause-Jensen, D., Quaresma, A. L., Cunha, A. H., & Greve, T. M. (2004). How are seagrass distribution and abundance monitored? In J. Borum, C. M. Duarte, D. Krause-Jensen, & T. M. Greve (Eds.),European seagrasses: An introduction to monitoring and management(pp. 345 – 350), The M&MS project, September 2004, Internet version at: http://www.seagrasses.org.

[36] Ferguson, R. L. & Korfmacher, K. (1997). Remote sensing and GIS analysis of seagrass meadows in North Carolina, USA.Aquatic Botany, 58(3–4), 241 – 258

[37] Green, E. P., Mumby, P. J., Edwards, A. J., & Clark, C. D. (1996). A review of remote sensing for the assessment and management of tropical coastal resources.Coastal Management, 24(1), 1 – 40.

[38] Mumby, P. J., Edwards, A. J., Green, E. P., Anderson, C. W., Ellis, A. C., & Clark, C. D. (1997). A visual assessment technique for estimating seagrass standing crop.Aquatic Conservation: Marine and Freshwater Ecosystems, 7, 239 – 251.

[39] Hochberg, E. J., Andre´fouJt, S., & Tyler, M. R. (2003). Sea surface correction of high spatial resolution IKONOS images to improve bottom mapping in near-shore environments.IEEE Transactions on Geoscience and Remote Sensing, 41, 1724 – 1729.

[40] Cochrane, M. A. (2000). Using vegetation reflectance variability for species level classification of hyperspectral data. *International Journal of Remote Sensing*, *21*(10), 2075-2087.

[41] Dekker, A. G., V. E. Brando & J. M. Anstee, 2005. Retrospective seagrass change detection in a shallow coastal tidal Australian lake. Remote Sensing of Environment 97: 415–433

[42] Joy, S. M., Reich, R. M., & Reynolds, R. T. (2003). A non-parametric, supervised classification of vegetation types on the Kaibab National Forest using decision trees. *International Journal of Remote Sensing*, *24*(9), 1835-1852.

[43] Fleiss JL, Levin B, Paik MC (2003) Statistical methods for rates and proportions, 3rd ed. Hoboken: John Wiley & Sons.

[44] MedCalc Statistical Software version 13.3.3 (MedCalc Software bvba, Ostend, Belgium; http://www.medcalc.org; 2014)

[45] Matlab (Student Version) [Computer software]. (2013). Retrieved from http://www.mathworks.com/ academia/ student_version/

[46] Altman, D. G. (1990). *Practical statistics for medical research*. CRC Press.

[47] Piccone, T., MacBrian, J., Zhao, H., and Yan, Y. (2012). Everglades Stormwater Treatment Area average ground elevations, stage-Area/stage-volume relationships and effective treatment areas: Technical Publication. SFWMD.

[48] Huang, G. B., & Babri, H. A. (1998). Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. *Neural Networks, IEEE Transactions on*, *9*(1), 224-229.

[49] Narasimha, P. L., Manry, M. T., & Maldonado, F. (2008). Upper bound on pattern storage in feedforward networks. *Neurocomputing*, *71*(16), 3612-3616.

[50] Pan, X., Lee, B., & Zhang, C. (2013, November). A comparison of neural network backpropagation algorithms for electricity load forecasting. In *Intelligent Energy Systems (IWIES), 2013 IEEE International Workshop on* (pp. 22-27). IEEE.

[51] Moré, J. J. (1978). The Levenberg-Marquardt algorithm: implementation and theory. In *Numerical analysis* (pp. 105-116). Springer Berlin Heidelberg.

[52] Møller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural networks*, *6*(4), 525-533.

[53] MacKay, D. J. (1992). A practical Bayesian framework for backpropagation networks. *Neural computation*, *4*(3), 448-472.

[54] Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *Neural Networks, 1993., IEEE International Conference on* (pp. 586-591). IEEE.

CHAPTER 4: DIAGNOSIS ANALYSIS OF WATER POLLUTION CONTROL FUNCTIONALITY OF A CONSTRUCTED WETLAND BY THE ANN-BASED PREDICTION OF HYDRAULIC PATTERN

4.1   Introduction

A well-designed constructed wetland should be capable to uphold two major properties, namely hydraulic loading rate (HLR) and hydraulic retention time (HRT). These two parameters significantly affect the treatment performance of the wetland [1]. It has been found that nutrient treatment efficiency can be improved by decreasing the HLR and increasing the HRT [2]. HRTs will vary from one wetland to another, depending on the vegetation condition, inflow flow rate, outflow flow rate and the nutrient removal rate. Effective HRTs have been described in the range from 4 to 15 days [3], whereas shorter HRTs within a range from 3 to 6 days was reported as efficient for removing disease causing bacteria and viruses [4]. In order to achieve significant removal rates of nutrients, larger hydraulic retention times (HRTs) with uniform slow velocities throughout the wetlands are required to fulfill sound restoration schemes that will be sustainable. For the constructed wetland systems near the Everglades area, the HRT within the wetland also affects the upstream control of water flow from Lake Okeechobee. Most of the constructed wetlands possess a low velocity regime ranging from $0 - 5$ cm•s^{-1} [5] which is a dominant factor for achieving the required HRT and supporting the attraction and removal of nutrient via plant species uptake.

Very limited numbers of studies had been conducted in order to capture the distribution of flow velocity and HRT in a wetlands system using modeling techniques. Data driven models eg., ANN and GP models can be an efficient tool in capturing correlations and trends when limited data is available for a system with respect to interactions between hydrological,

climatological, and physical processes. ANN is a data driven model inspired by the connections within a human's brain which uses algorithms to abstract away datasets and focus on the most important information. ANN models have been successfully applied to help solve several water and environmental problems [6]. For example, the effects of climate change on the discharge of dissolved organic carbon and nitrogen from a river basin were modeled and assessed using ANN [7]. ANN models were also used to forecast water quality parameters like salinity [8], residual chlorine concentration in urban water system [9], and prediction of wind velocity in a particular location using reference set of data [10]. However, only a few studies were conducted in constructed wetlands to predict nutrient concentrations in effluent waters [11-13]. Besides that, the Extreme Learning Machine (ELM) has recently received a considerable attention in various fields of optimization (Chapter 2 and Chapter 3). On the other hand, genetic algorithm-based models are inspired by Darwinian process of natural selection and biological operation. The special kind of evolutionary computing algorithm, namely GP, was first presented by Koza [14] as a development or extension of the genetic algorithm. GP models operate by preparing populations which consist of random numbers (also known as chromosomes). The fitness of each chromosome is then evaluated by comparison with a target value. Using genetic crossover, or sexual recombination number(s), computer programs are genetically reproduced with the help of Darwin's principle of survival of the fittest. Some examples of GP models previously used include: hydraulic studies on velocity prediction in lab scaled vegetated floodplains [15], developing rainfall runoff relationship from synthetic data along with sediment transport models, salt intrusion in estuarine environments, analyzing flow over a flexible bed covered with vegetation [16-17], emulating rainfall-runoff process [18], and monitoring total organic carbon concentration in a lake [19].

Correlations between hydrological, climatological factors, and flow behavior has not been well established for complex hydraulic systems such as constructed wetlands. As there was no sufficient previous research which could establish the application of data driven models in predicting hydraulic parameters of a wetland system, this study presents a unique attempt to demonstrate the application of such models for a wetland system. The question remains as to whether it is possible to develop a reliable prediction of hydraulic properties using data driven models. We hypothesize that: 1) velocity field is uniform in the constructed wetland and 2) HRT is uniformly distributed in the constructed wetland given the strict control of inflow and outflow with the gates. The objective of this study is thus to test the computational intelligence modeling techniques and predict the hydraulic parameters, namely flow magnitude, flow direction and HRT. With this study objective, the following scientific questions are explored in this study: 1) Is it possible to predict the low flow velocity regime and HRT in a constructed wetland using computational intelligence models based on limited local measurements via a sensor network? 2) Among different computational intelligence models, which model provides the most accurate estimation of low flow velocity regime and HRT for a constructed wetland system?

4.2   Methodology

4.2.1   *Sensor network and primary data collection*

4.2.1.1   *Sensor Network Development*

A sensor network was established within the cell 3B area to monitor the flow magnitude and flow direction. The whole sensor network was comprised of two different components. The first component corresponds to the measurement network developed by SFWMD and their

database known as DBHYDRO. DBHYDRO is SFWMD's environmental database which stores hydrologic, meteorological, hydrogeological, and water quality data. Parameters considered from the DBHYDRO system for cell 3B were inflow, outflow, cell upstream stage, cell downstream stage, wind speed, and wind velocity. The locations of all the measurement stations are shown in Figure 4.1. All measurement data were stored by SFWMD after ensuring sufficient quality checks. For the inflow and outflow computation, instantaneous values of upstream and downstream stage along with the control level were stored in a database known as Data Collection and Verification Program (DCVP). Water stage data was based on elevation above mean sea level, or National Geodetic Vertical Datum (NGVD), located at the upstream and downstream control structures. Different types of recorders such as graphic, telemetry, solid state, punch tape and daily water readings were used to collect the instantaneous stage data. In addition, daily water readings were done by field staff for further verification and quality check. Such data recording procedure had a recording frequency ranging from 6 to 24 hours. The fine time resolution data (also known as Breakpoint data) or continuous time series data of wind speed and wind direction were collected from a Rotenberger Tract weather station situated approximately 20 kilometers from cell 3B. CR10 data logger from *Campbell Scientific Inc.* was used for recording such weather parameters. The term break point was derived from data reduction which was used to minimize the redundant points in the database. Such algorithms were typically attached with the corresponding data logger software or can be done during post processing of data.

The duration of data collected from the DBHYDRO database system began 20th August, 2013 and ending 26th October, 2013. The sensor measurements collected from the DBHYDRO

database system along with the corresponding station ID, temporal resolution, recorder type and control structure type are summarized in Table 4.1.

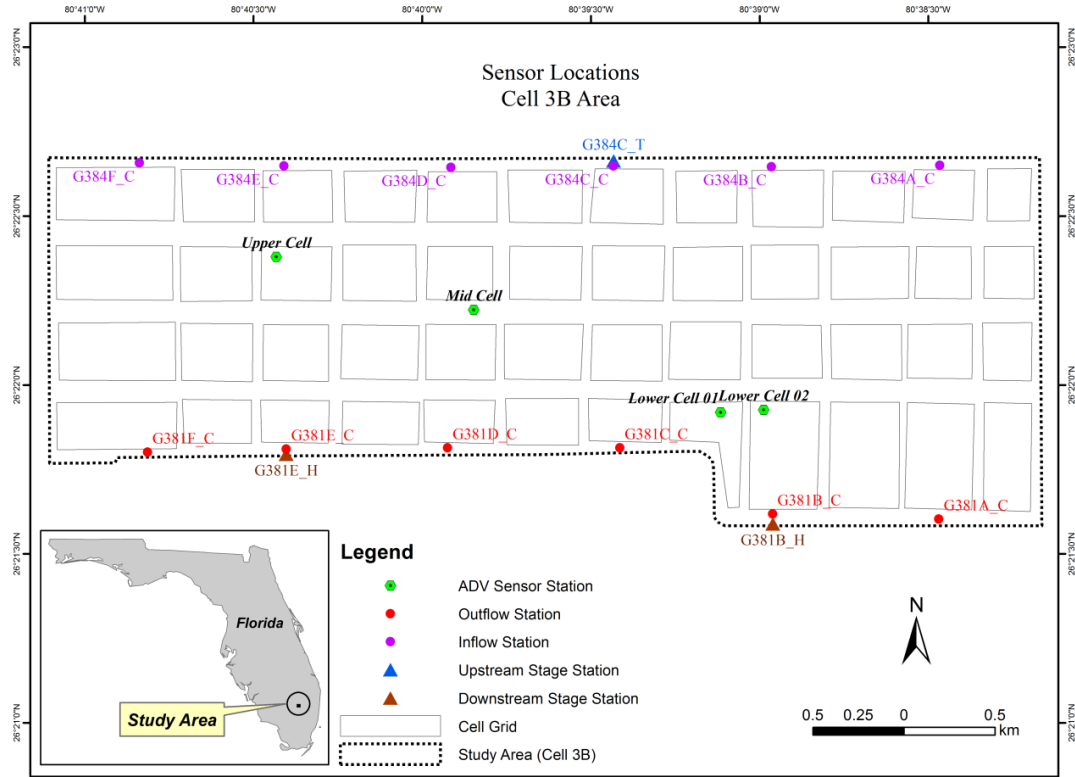Figure 4.1: Sensor locations

Table 4.1: Details of DBHYDRO dataset

Station ID	DBHYDRO Database ID	Parameter	Recorder Type/ Control Structure	Temporal Resolution	Data Unit
G384A_C	W1927	Inflow	Gated Culvert	Daily Average	Cubic feet per sec
G384B_C	W1928				
G384C_C	VV483				
G384D_C	W1929				
G384E_C	W1930				

81

G384F_C	W1931				
G381A_C	TA296				
G381B_C	TA297				
G381C_C	TA299				
G381D_C	TA298	Outflow	Gated Culvert	Daily Average	Cubic feet per sec
G381E_C	TA300				
G381F_C	TA301				
G382B_C	T9992				
G384C_T	VV482	Upstream Stage	Telemetry Recording	Hourly	Feet NGVD29
G381E_H	T1069	Downstream Stage	Telemetry Recording	Hourly	Feet NGVD29
G381B_H	T1062				
ROTNWX	KV253	Wind Speed	CR10 Data Logger	15 min	Miles per hour
ROTNWX	IY276	Wind Direction	CR10 Data Logger	15 min	Degree clockwise from North direction

The second component of the sensor network was established in the study area using Vector Acoustic Doppler Velocimeters (ADVs), developed by Nortek. ADVs measure the water velocity by using a physical principle called the "Doppler Effect". ADVs transmit short pairs of sound pulses through an emitter and measure the echoed frequency reflected off of suspended particles in the water. Assuming the suspended particles have the same velocity as the water, the water velocity is calculated by relation to the Doppler shift measured from the reflected pulse. Each ADV has one emitter and three receivers angled from the central axis, creating a sampling volume located approximately 5-10cm from the emitter. Having all three receivers concentrated on the same sample volume, the ADV creates a three dimensional velocity measurement with

directional components. The ADV sensor and its velocity measurement principle are shown in Figure 4.2.

Figure 4.2: ADV sensor (left) and velocity measuring principle (right) (Nortek, 2005)

ADV sensors were installed on a wooden support frame with the sensor orientated in a downward facing position. The structure was built to withstand high wind velocities and other abrupt weather conditions so as to minimize any damage to the ADV sensor. To minimize Cartesian coordinate system errors during velocity measurements, the supportive structure was built firm enough to remain stable in the horizontal and vertical direction. To decrease turbulent flow impacts, the supporting legs were sufficiently spaced 5 feet apart from each other. A velocity range of 10 cm•sec^{-1} was fixed by conducting an initial flow velocity check using the ADV. The entire ADV sensor support structure is shown in Figure 4.3.

Two ADV sensors were deployed at four locations periodically around the Cell 3B area for monitoring flow velocity and direction. Site locations were selected considering adequate depth in order to depict velocities at mid-depth. The four locations are termed as "upper cell", "mid cell", "lower cell-01" and "lower cell-02" as shown in Figure 4.1. The "upper-cell" location was surrounded primarily with dense SAV with a strip of EAV located at a distance of

30m to the north side. For the mid-cell location, dispersed EAV was observed around the sensor location. The lower cell-01 location was situated in a submerged north-south "ditch", which had water depths characteristically deeper than the surrounding areas. Dense EAV was located 3.5m upstream from the sensor and some dispersed EAV and SAV were observed at the eastern and western side of the sensor location. The surroundings of the lower cell-02 sensor location was predominantly dispersed SAV, though some dense EAV was located at a distance of 6m and 3m towards the northern and western sides, respectively. The southern part of the lower cell-02 location was comprised mainly of dense and dispersed SAV.
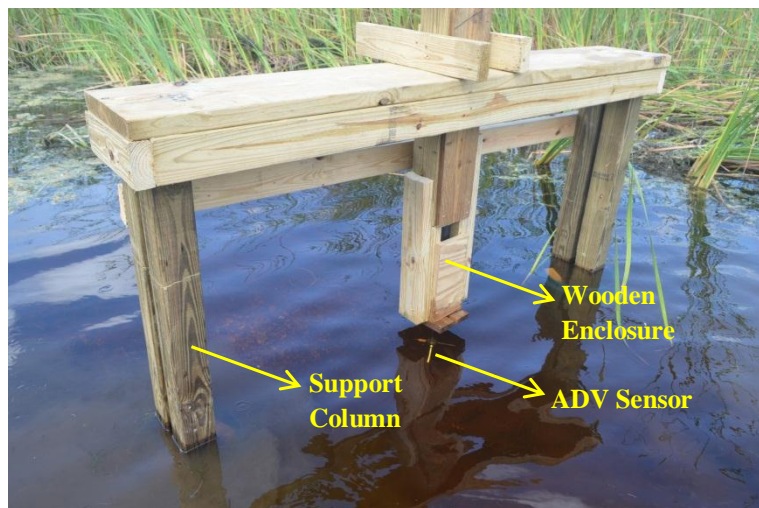

Figure 4.3: Sensor support structure

4.2.1.2 *ADV data collection and processing*

During ADV data collection, the averaging duration and sampling duration were properly maintained to reduce the effect of Doppler noise. Averaging duration represents the amount of time required by the ADV to collect one sample, whereas the sampling duration represents the amount of time required for conducting the measurements. ADVs measure the velocity by taking the average of many estimated velocities known as "pings". By averaging many pings, the

uncertainty behind every ping or short-term error can be reduced significantly. The reduction in uncertainty depends on a limit known as the long-term bias. Internal signal processing and beam geometry can affect the long-term bias. Depending on the short-term error due to a ping and number of pings averaged together, the error in the measurement is estimated. The reduction in error follows the following formula:

$$\sigma_{v(mean)} = \frac{\sigma_{v(ping)}}{\sqrt{N}} \qquad\qquad (4.1)$$

Where, σ is the standard deviation and N is the number of pings averaged together. For data collected in the Cell 3B area, the number of pings recorded in each velocity measurement was set in-between 10 to 30, depending on digital space requirements. Sampling intervals were set to 30 min for the upper cell location while the sampling interval for the mid cell, lower cell-01 and lower cell-02 was set to 60 min.

The data collection period of four sampling locations along with their geographical positions are listed out in Table 4.2. During the post processing of ADV sensor data, two quality check parameters were set to ensure data quality, the signal-to-noise ratio (SNR) and correlation coefficient. In the steady flow condition, it is important to remove all data samples with communication errors with average correlation below 60 percent or SNR falls below 5-10 dB [20]. In the current study, sampling data having correlation coefficient below 60 percent and SNR below 5 dB were removed from the analysis. Besides that, spikes which occurred due to the physical interference such as aquatic life and submerged floating vegetation were removed by visual inspection of the dataset.

Table 4.2: Data collection period

Sampling Location	Latitude	Longitude	Data Collection Start	Data Collection End
Upper Cell	26°22'23.21"N	80°40'26.58"W	30th August, 2013	17th September, 2013
Mid Cell	26°22'13.33"N	80°39'51.17"W	22nd August, 2013	30th August, 2013
Lower Cell -01	26°21'54.75"N	80°39'8.63"W	22nd August, 2013	22nd September, 2013
Lower Cell -02	26°21'56.00"N	80°39'0.50"W	11th October, 2013	25th October, 2013

4.2.2 *Model design and hydraulic parameter prediction in cell 3B*

4.2.2.1 *ANN Model Network Architecture*

The three main components of the ANN model design are the input layer(s), hidden layer(s) and an output layer(s) of processing unit or neurons. The input layers introduce the inputs to the network, then the hidden layers transforms those inputs using non-linear activation functions. Several types of activation functions may be used in generating output from the neural network model. The two most commonly used activation functions are linear transfer function and log-sigmoid transfer function. Linear transfer functions are normally used in the final layer of multilayer neural networks as a function approximator. The log-sigmoid transfer functions are used in hidden layers of multilayer neural network where the function takes input ranging from minus infinity to plus infinity and finally generates an output in the range of 0 to 1. All ANN models applied a set of rules which change the connection weights in an iterative process during the learning phase until a certain stopping criteria is reached. Back propagation

(BP) training algorithm was used as a learning algorithm which is known as a common algorithm utilized in more than 80% of previous neural network model studies [6]. Several BP algorithms are available for training the dataset during the learning process, out of which a Levenberg-Marquardt BP algorithm [21] was used for the current study.

In this study, a two-layer feed-forward network with sigmoid hidden neurons and linear output neurons was used as a supervised network in the analysis. Twenty five hidden neurons were used for the network. Out of four ADV sensor sampling stations, two station's dataset were used as target dataset in the ANN training process including those two stations located at the upper cell and lower cell-01. Temporal distribution ranging from 30th August, 2013 – 17th September, 2013 was used for both stations. A total of 703 time steps were considered within the time range. In order to incorporate the spatial information, the whole Cell 3B area was divided into a matrix form with dimension of 4 by 12 which corresponds to 48 small blocks as shown in Figure 4.4. Each block had a unique row and column number for identification. Block denoted as "C2-8" corresponds to the row 2 and column 8 of the matrix. The ADV sampling stations used as target data for the training process were located at the C2-3 and C4-8 blocks. A total of nineteen inputs corresponding to eight types of parameters were used as neural network training processes. These parameters included; wind speed, wind direction, inflow, outflow, upstream stage, downstream stage, block row number and block column number. The ANN model architecture for predicting velocity magnitude and direction is shown in Figure 4.5. A similar ANN model was also developed for predicting stages around the wetland system by considering the three stage information as target data set during training process to support HRT estimation.
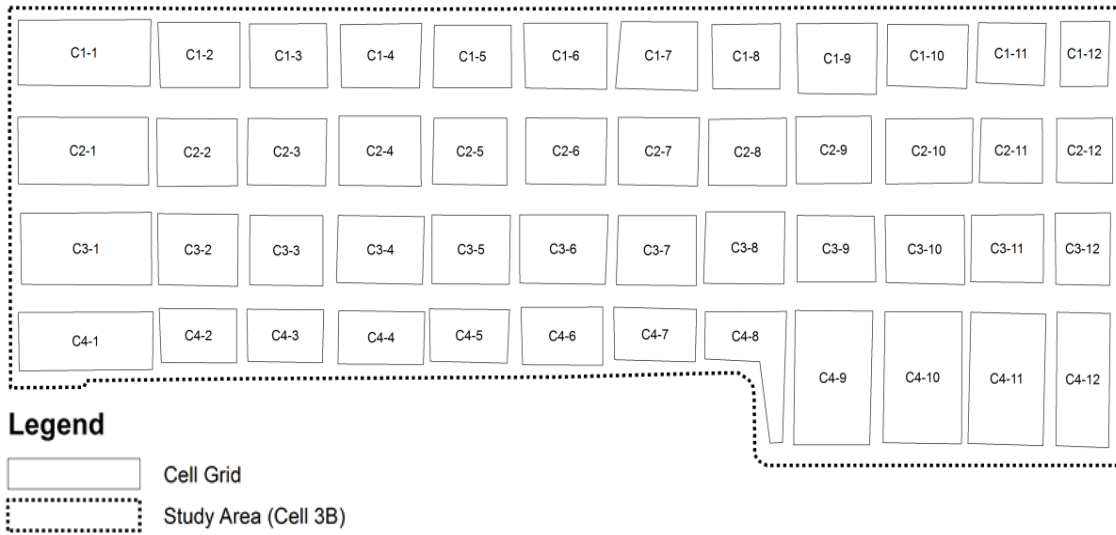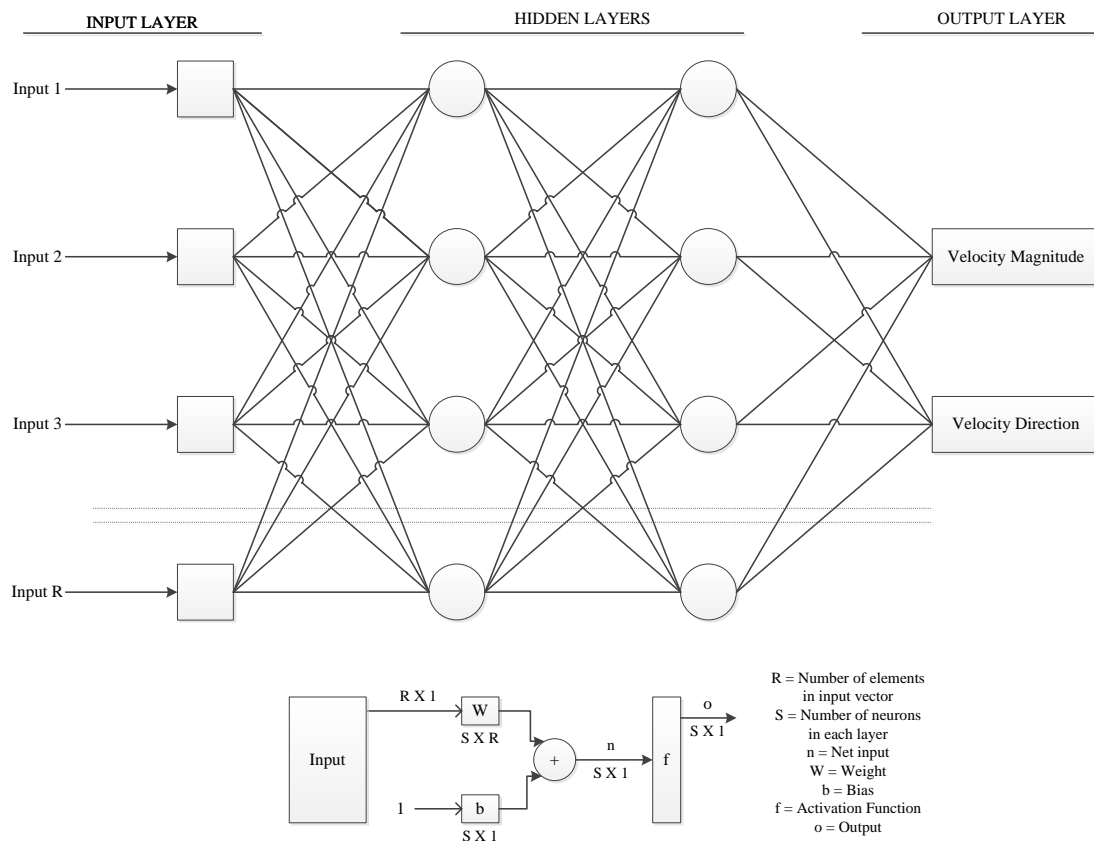
Figure 4.4: Cell locations

Figure 4.5: ANN Architecture to Predict Velocity Magnitude and Direction

88

4.2.2.2   GP Model

The methodology of the GP modeling practice has three distinct steps (Figure 9). Those are (i) initialization of randomly generated population of programs, (ii) evaluation and ranking of programs depending on their fitness, and, (iii) application of genetic operators to replace bad programs (if error threshold is exceeded). In the first step, programs were generated to relate the surrounding parameters (wind direction, magnitude, inflow, outflow, stages) to the flow velocity and direction. The program size and number of programs were fixed manually. Due to the random generation of programs, it was required to rank all the programs depending on the fitness. Programs with low fitness were discarded from the consideration. Best performing programs were used for the application of genetic operators that are shown in Figure 4.6. During reproduction, all the original programs were directly copied, whereas in crossover, the solution procedures were modified within the programs of best fitness value. The child program was generated using the operations contained within two parent programs of high fitness. The mutation processes changed the solution procedure randomly within the program. It should be noted the original fitness model should not be changed until the genetic operators can produce much higher fitness value program. The overall process was continued until a minimum threshold error was attained.

For the current study, Discipulus software package was used to develop the GP model. Discipulus uses the supervised learning process to improve the programs. The whole input datasets were divided into three segments as training, validation and applied data as ratio of 80%, 10% and 10% respectively. Splitting of input datasets was conducted in a manner so as to have the same range of velocity magnitude and velocity direction exposed towards the training,

89

validation and applied dataset. The operators considered for training and validating the GP model were as follows: mathematic, arithmetic and trigonometric. Default values for mutation and crossover of 95% and 50% in defined by Discipulus were kept. The programs were initialized within a size range of 80MB to 256MB and the computational time (CPU time) is one day for all scenarios.
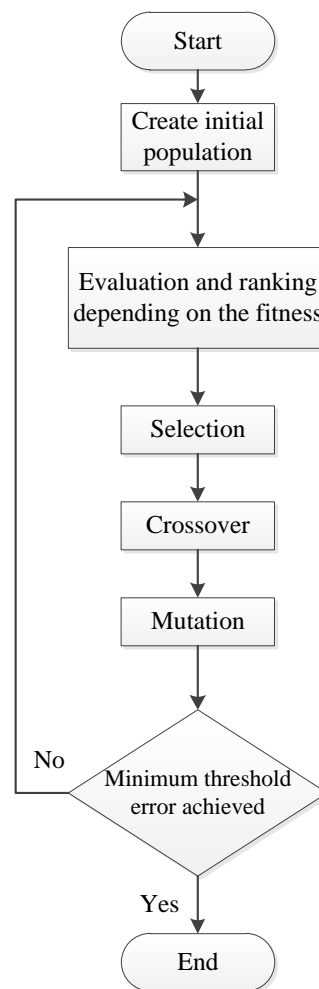

Figure 4.6: Genetic operators used in GP model

4.2.2.3   ELM Model

Methodology for ELM model has been described in Chapter-02.

4.2.2.4   Prediction of velocity magnitude and direction

All ANN training simulations were conducted using a Matlab software package. After selecting the input and target parameters the whole network was trained for both velocity magnitude and direction. The training process was continued until a certain stopping criterion was reached; in this case it is the coefficient of determination (R-square). A properly trained ANN model was used to predict the velocity direction and magnitude for the remaining blocks. During the prediction step, input dataset for the remaining 46 blocks was fed into the previously trained ANN model to generate the output. Input parameters for the remaining blocks were the same as the input parameters used for the training of the ANN model, except the position of the block in the block matrix. So for each block, a unique row and column number was assigned during the prediction process. After simulating the trained ANN model with 46 new sets of input for both velocity magnitude and direction, the final predicted time series was generated for all the block locations. The predicted velocity magnitudes and directions were then classified into three segments depending on the upstream stage values. Three distinct upstream stages were selected to demonstrate the change of velocity pattern within the cell area. The stages included; (i) high inflow stage condition (3.60m NGVD29), (ii) medium inflow stage condition (3.51m NGVD29), and (iii) low inflow stage condition (3.46m NGVD29). For each of the mentioned conditions, a particular time step was selected to develop a velocity prediction surface for the whole area. Using the unique value of velocity for all 48 blocks at a certain time step, a velocity surface for each three upstream stage conditions was generated using natural neighbor (NN)

91

interpolation techniques [22]. The steps for predicting two dimensional velocity magnitude surface and direction are shown in Figure 4.7.
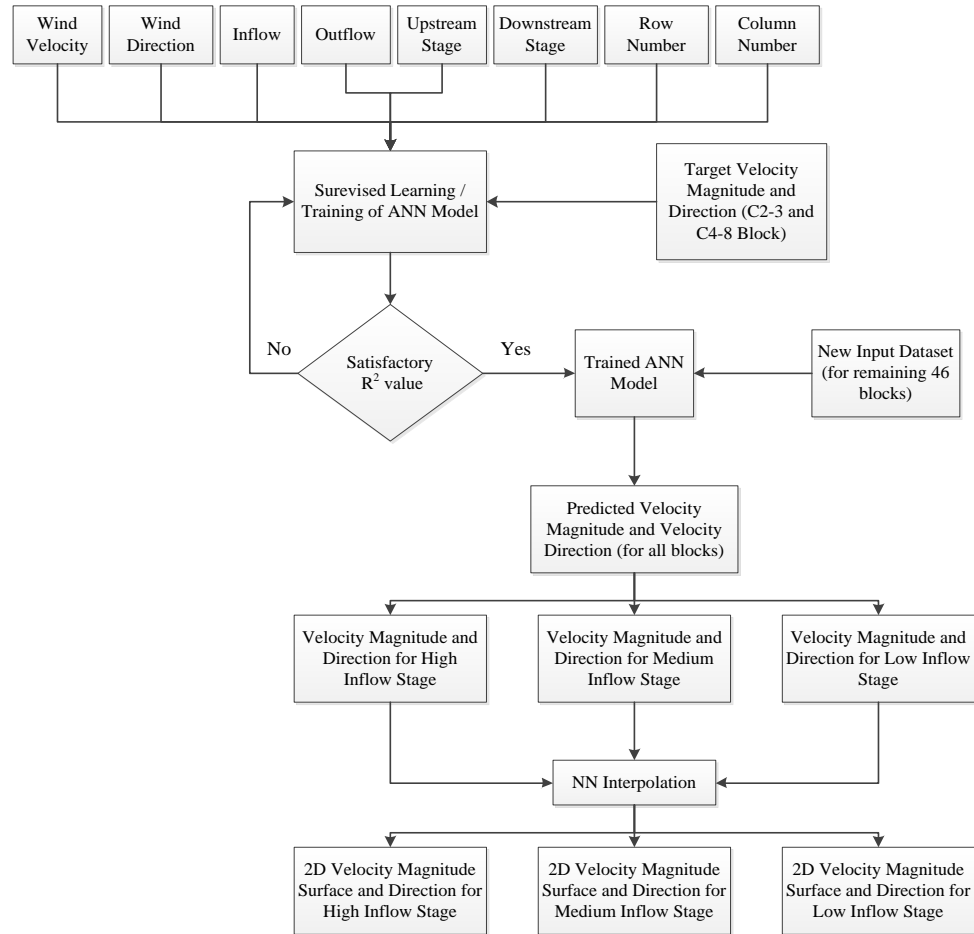

Figure 4.7: Flow chart for developing two dimensional surface mapping of velocity magnitude and direction

4.2.2.5 *Prediction of HRT*

As mentioned earlier, the HRT represents the duration for which water remains in a particular place and is a crucial parameter for any constructed wetland area. The main objective of the constructed wetland is to provide a sufficient HRT for water to remain within the wetland,

so as to allow nutrient uptake by the vegetation, thereby decreasing soluble nutrient concentrations in the effluent. In order to organize the spatial distribution of vegetation inside the wetland, and making a proper channeling of water to flow towards downstream, it is important to have knowledge of the HRT inside the wetland. The HRT for an individual block can be calculated using the following formula:

$$HRT = \frac{VO}{Q} = \frac{BA*WD}{XA*V}$$ (4.2)

Where VO = volume of water inside of a cell block, Q = flow rate of water, BA = Block surface area, WD = Water depth, XA = Cross-sectional area, and V = Predicted flow velocity. The depth of water (WD) for each block was calculated by deducting the land elevation (Figure 3) from the predicted stage. During calculating the cross sectional area of individual block, it was assumed that the major flow direction was towards the south and the cross-sectional area for each block was calculated considering the section perpendicular to the north-south direction. Finally, the HRTs found for 48 blocks were used to generate a 2D surface map using NN interpolation techniques for the same three upstream stage scenarios considered during flow velocity predictions. The steps of predicting stage for different cell blocks are shown in Figure 4.8.
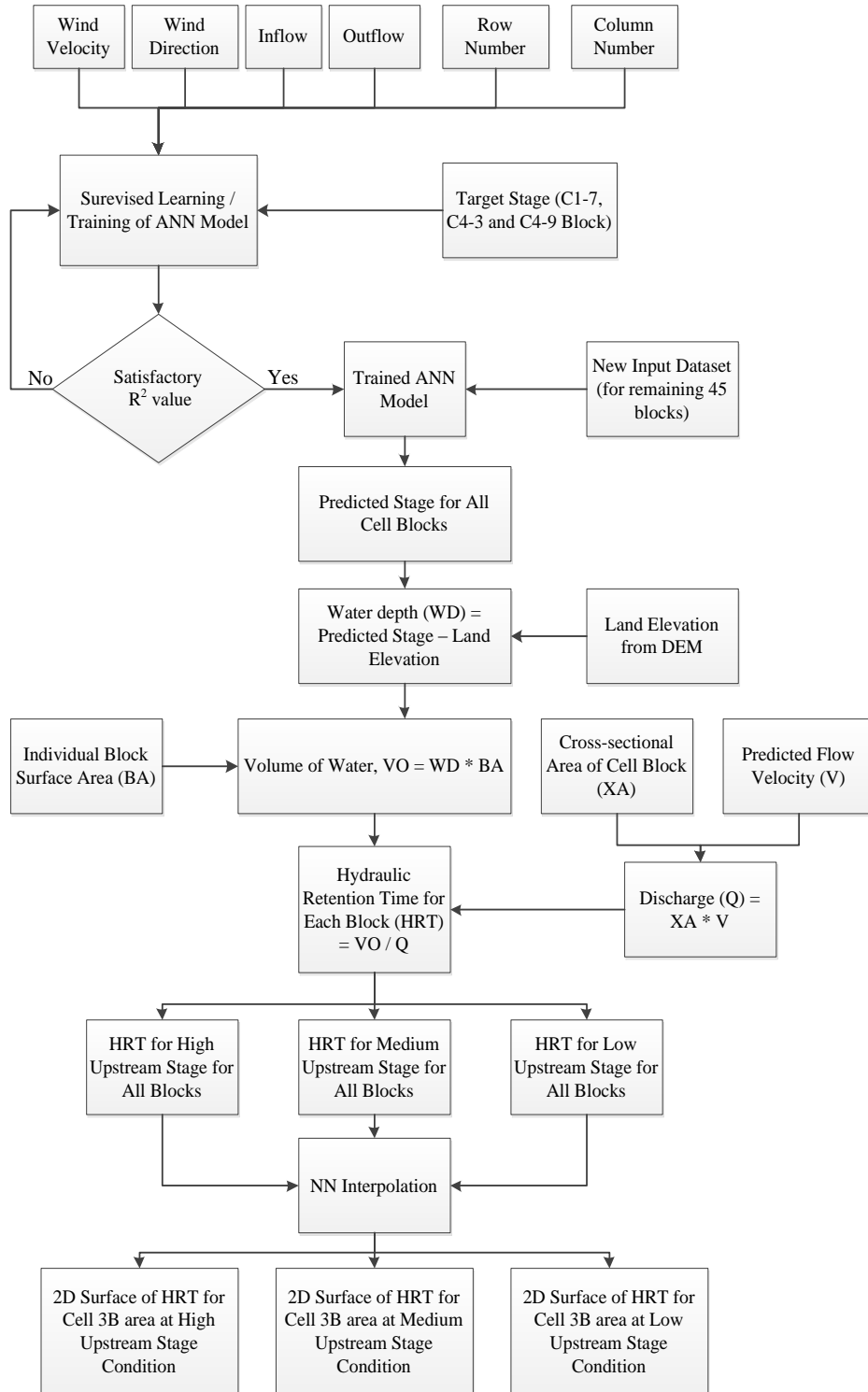
Figure 4.8: Flow chart for developing two dimensional surface of stage for Cell 3B area

4.2.2.6   Statistical Analysis

For the current study, three statistical indices were considered to check the model performance. Those indices are coefficient of determination (R-square), Pearson product moment correlation coefficient (RSQ), and root mean square error (RMSE). The indices R-square was used to evaluate the fitting performance during the initial training of the model, while the remaining two indices were used to check the model performance during validation stage. The RMSE, RSQ and R-square can be calculated using the Equation 3-5 as below:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Y_{pi} - Y_{oi})}{N}} \qquad\qquad (4.3)$$

$$RSQ = \frac{\sum_{i=1}^{N}(Y_{oi} - \overline{Y_{oi}})(Y_{pi} - \overline{Y_{pi}})}{\sqrt{\sum_{i=1}^{N}(Y_{oi} - \overline{Y_{oi}})^2 \sum_{i=1}^{N}(Y_{pi} - \overline{Y_{pi}})^2}} \qquad\qquad (4.4)$$

$$R\text{-}square = RSQ^2 \qquad\qquad (4.5)$$

Here, Y_{pi} is the i-th predicted value, Y_{oi} is the i-th observed value, N is the total number of sample, $\overline{Y_{oi}}$ is the arithmetic mean of the observed values and $\overline{Y_{pi}}$ is the arithmetic mean of the predicted values. In an ideal situation, the value of RMSE should be zero. Any ideal model would depict an RSQ value of 1.

4.3   Results and discussion

4.3.1.1   Sensor data analysis

The upper cell sensor location was situated between the inflow station G384E_C and outflow station G381E_C. Southern ADV flow velocity fluctuated between -1.05 cm•sec^{-1} (northerly) to 2.21 cm•sec^{-1} (southerly) with an average velocity of 0.20 cm•sec^{-1}. Whereas, western flow velocity fluctuated between -1.24 cm•sec^{-1} (easterly) to 1.79 cm•sec^{-1} (westerly)

with an average velocity of 0.03 cm•sec^{-1}.   Fluctuation of velocity in upper cell location along with the inflow and outflow is shown in Figure 4.9. No significant relation was observed between the inflow, outflow and the fluctuation of flow velocities in the upper cell location.

The mid cell sensor was located 4.5m north from a canal that ran along the east-west direction. The north-south velocity at mid cell area varied between -1.11 cm•sec^{-1} to 4.22 cm•sec^{-1} with an average velocity of 1.00 cm•sec^{-1}. East-west velocity was varied between -2.03 cm•sec^{-1} to 2.21 cm•sec^{-1} with an average velocity of 0.21 cm•sec^{-1}. The three dimensional distribution of flow velocity at the mid cell location is shown in Figure 4.10. A gradual upward trend in flow velocity was observed for both the southern and western direction. Rainfall and inflow to the cell area were two predominant factors which influenced the upward velocity trend in both southern and western directions. A strong correlation was observed between the inflow from G384D_C station and velocity at the mid cell location. The impact of rainfall and inflow on the velocity measurement at mid cell is shown in Figure 4.11 and Figure 4.12.
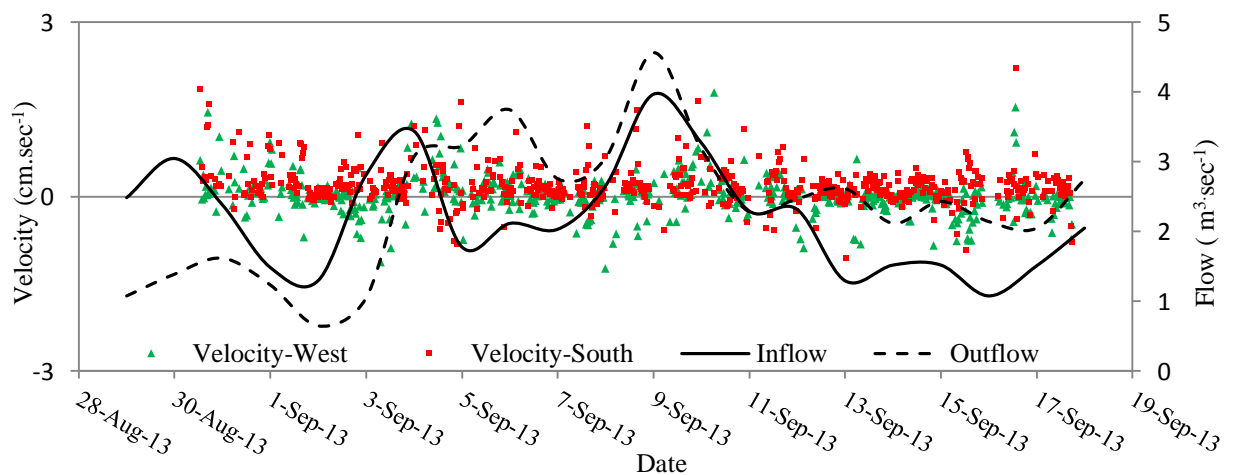

Figure 4.9: Velocity and flow measured in the southern and western direction at upper cell location
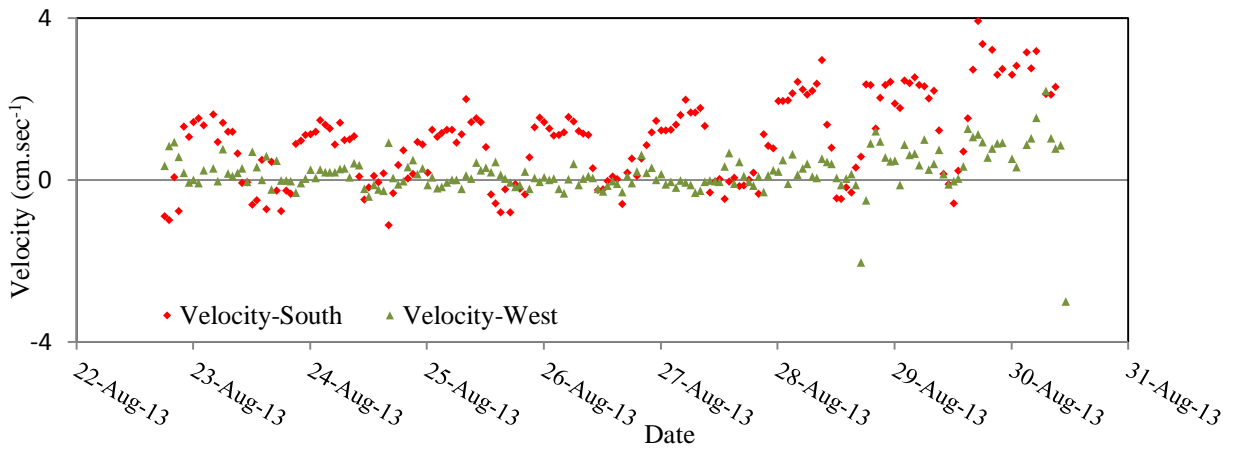
Figure 4.10: Velocity measured in the southern and western direction at mid cell location
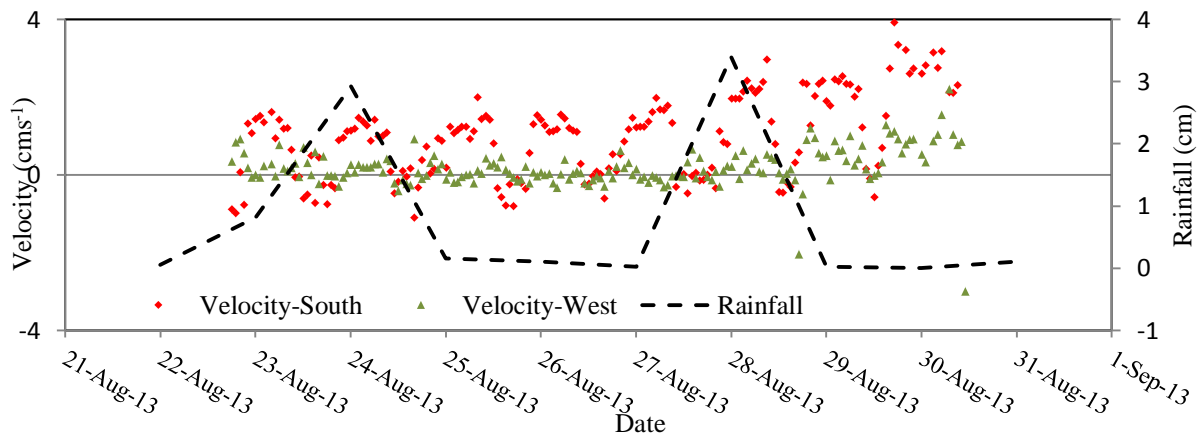

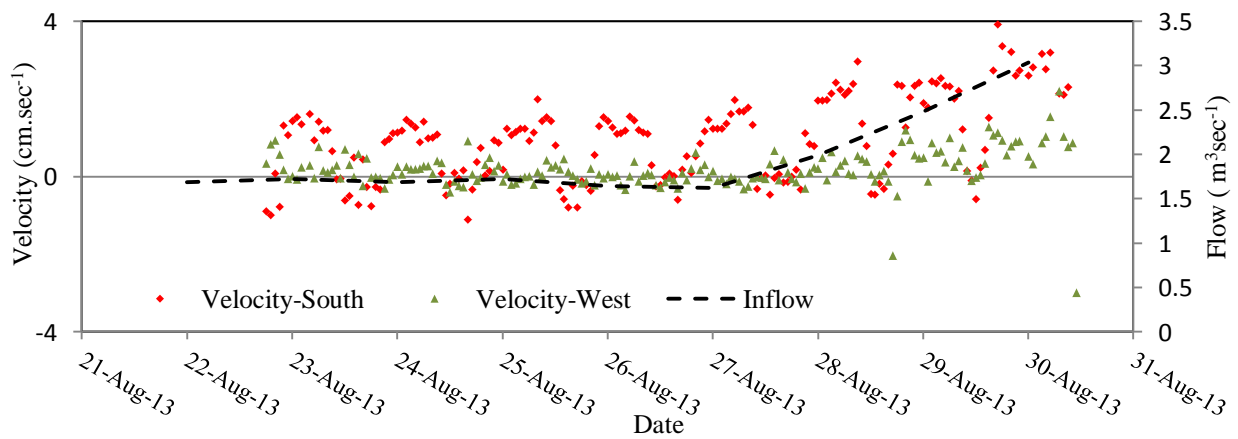Figure 4.11: Impact of rainfall on flow velocity at mid cell location

Figure 4.12: Upward trend in both inflow and measured velocity at mid cell

97

In the lower cell-01, the flow regime was highly governed by a southern flow which varied between -0.5 cm•sec^{-1} to 7.8 cm•sec^{-1}, with an average velocity of 3.90 cm•sec^{-1}. High southern velocities were quite evident due to the short circuiting effect caused by the ditch-like bathymetry located at the sensor location. The western velocity varied between -1.85 cm•sec^{-1} to 3.57 cm•sec^{-1}, with an average velocity of 0.71 cm•sec^{-1}. The flow velocity in the southern and western side is shown in Figure 4.13. The fluctuation in the southern velocity was perfectly harmonized with the rainfall, inflow, outflow and stage information near the sensor location. Impact of rainfall on the southern velocity is displayed in Figure 4.14. After every significant rainfall event the southern velocity showed an upward response trend. High correlation was also observed between the stage information (G381B_H), near to the lower cell-01 location, and southern velocity data as shown in Figure 4.15.
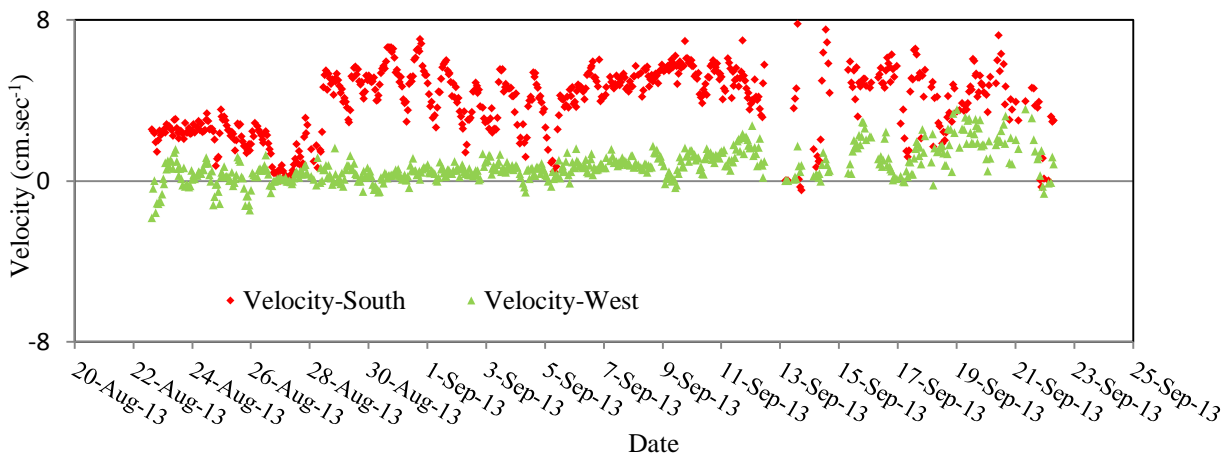

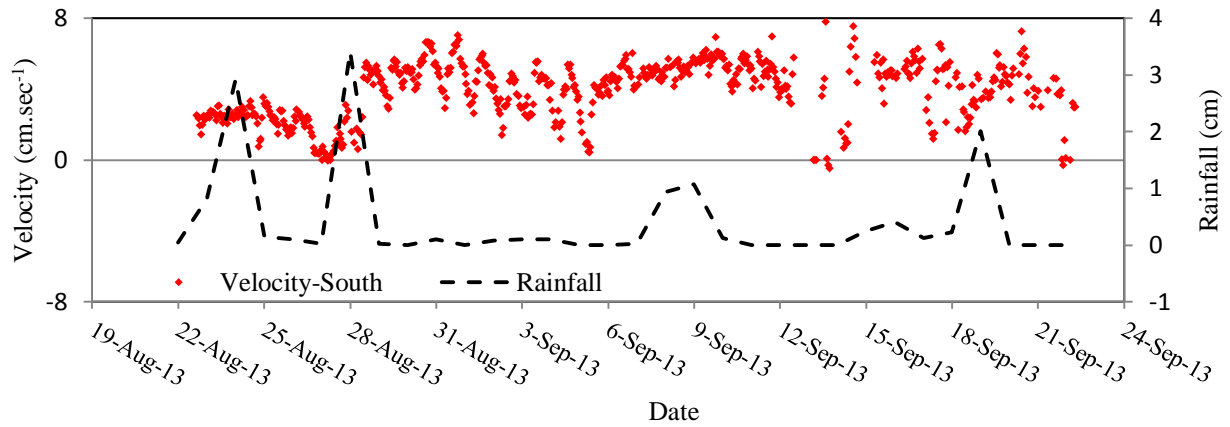Figure 4.13: Southern and western flow velocity at lower cell-01 location

Figure 4.14: Impact of rainfall on the southern velocity at lower cell-01 location
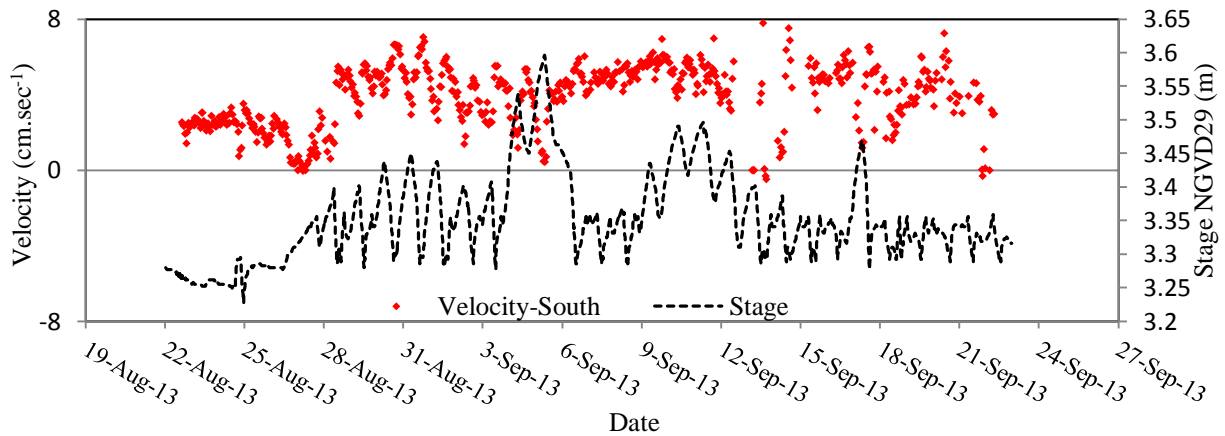
Figure 4.15: Correlation between stage and southern velocity at lower cell-01 location

In the lower cell-02, the southern flow velocity varied between -0.81 cm•sec^{-1} to 0.76 cm•sec^{-1} whereas the western flow velocity varied between -0.88 cm•sec^{-1} to 2.37 cm•sec^{-1}. A decreasing trend in flow velocity was observed in both southern and western flow velocity after 15th October, 2013 which was due to the closing of gate at the outflow (G381B_C) situated right at the southern side of the sensor location after that date. The change of southern and western flow velocity with inflow and out flow is shown in Figure 4.16. Analyzing the velocity in all four

sampling stations, the directional distribution of flows in east-west and north-south direction is summarized in a bar chart as shown in Figure 4.17.
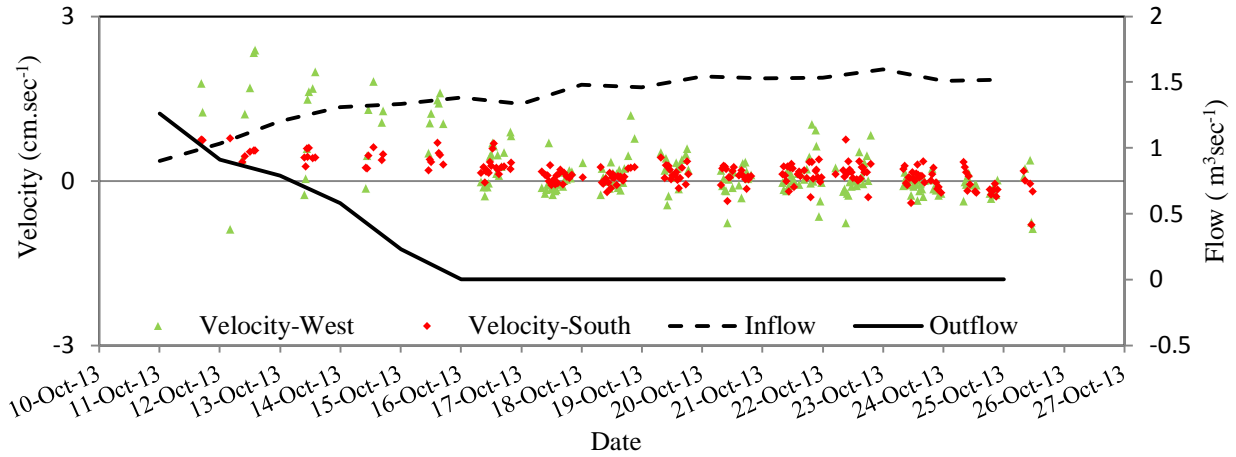
Figure 4.16: Change of flow velocity with inflow and outflow at the lower cell-02 location
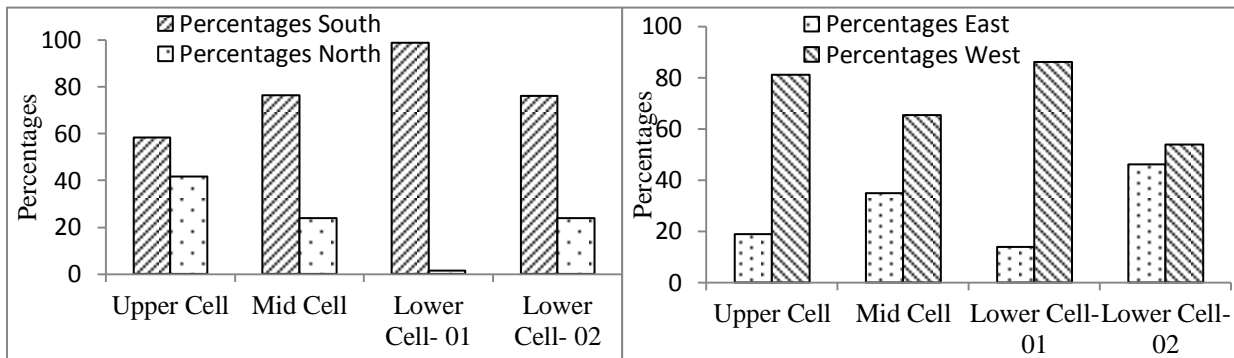
Figure 4.17: Flow distribution in north-south (left bar chart) and east-west (right bar chart) direction at all sensor location

From the field observations as described above, vegetation that may increase flow resistance and diminish flow conveyance plays a critical role in wetland systems; while having a positive influence on water quality by removing nutrients and producing oxygen in stagnant regions [23-24]. The patches of vegetation with spatial heterogeneity that provides a diversity of habitat also promote biodiversity in the flow [25]. By damping the flow, vegetation can create

100

local regions that entrain sediment and uptake nutrients [26]. Within sparse beds of vegetation, however, both bed stress and turbulent impulse may affect entrainment and sediment transport within emergent and submerged vegetation clamps, while regional wind patterns, inflow dynamics and near-bed turbulence structures around vegetation may collectively impact the flow direction and magnitude deeply. In particular, the deflection of flow away from the high drag of a vegetated region results in enhanced flow locally at its edges based on momentum and force balances, while both stem diameters and stem density impact the threshold of motions from the turbulent to the laminar flows providing scale linkages. This entails the phenomenon in statistics, as summarized in Figure 4.17.

4.3.1.2   *Model training for velocity magnitude and direction*

Initial training of the model is a key step for every supervised learning process. ANN, GP and ELM model were trained using the available sensor data sets and collected data from DBHYDRO. The ANN network's performance can be improved significantly by training the network repetitively, increasing the number of neurons and finally getting a larger training dataset for both velocity magnitude and direction. For the current study, the ANN model was trained continuously until a certain R-square value can be achieved. In the current study area, the change of magnitude of velocity was quite straight forward, which depends on the surrounding hydraulic condition such as inflow, outflow and stage data. As the prediction of velocity magnitude is relatively easier than that of the velocity direction, the limiting R-square value was set as 0.90 as a stopping criterion for the training process of velocity magnitude. The change in velocity direction was very complex as it was not only dependent on the surrounding hydraulic condition but also dependent on the weather parameter such as wind speed, wind direction and most importantly the distribution of vegetation around the area. Absence of

information regarding the vegetation pattern led to select a more liberal R-square value of 0.50 as a stopping criterion for the velocity direction training process. During the training process the input dataset was divided into three segments as training, validation, and test datasets. Several trials were conducted with different ratios between training, validation, and test datasets until a satisfactory R-square was achieved in all stages to avoid the over fitting issue. For ELM model, similar numbers of inputs were used for model training and the number of hidden unit was fixed as 900 for velocity magnitude and 2000 for velocity direction after several trials to attain the best possible training accuracy. The R-square values and simulation time for ANN, GP and ELM models for velocity magnitude and direction are shown in Table 4.3.

Table 4.3: R-square values for velocity magnitude and velocity direction

Statistical Parameter	ANN Model (BP)		GP Model		ELM	
	Magnitude	Direction	Magnitude	Direction	Magnitude	Direction
R-square (Training)	0.96	0.45	0.90	0.17	0.90	0.99
R-square (Validation)	0.93	0.35	0.84	0.39		
R-square (Test)	0.91	0.24	0.88	0.001	0.97	0.03
R-square (Overall)	**0.95**	0.42	0.87	0.28	0.92	**0.76**
Time	24 hour	24 hour	24 hour	24 hour	0.15 sec	0.54 sec

For the prediction of velocity direction with the ANN model, input dataset with 80% overallsamples gave the maximum overall R-square value of 0.42 at the training stage whereas the same percentage of training samples gave the maximum overall R-square value of 0.17 with the GP model and 0.76 with ELM model. The increment of data samples used for training was stopped after reaching the 80% ratio, as further increment in training stage would result in an insufficient dataset for validation and testing. Comparatively, the ELM model was considered better due to its high R-square values for velocity direction.

The same data ratio was also used for training the ANN model for velocity magnitude. The overall R-square value for velocity magnitude was found as 0.95. The regression plot for both velocity magnitude and direction are shown in Figure 4.18. The regression plot shows the ANN modeling outputs with respect to the targets for training, validation and test datasets. In the current study, the fit for velocity magnitude was resonably good with an overall R-value of 0.97 (R-square = 0.95). Fit for velocity direction was pretty low with an overall R-value of 0.65 (R-square = 0.42). The error histogram for the training of velocity magnitude and direction is shown in Figure 4.19. Thr error histogram is an indication of outliers. Outliers are the data points which fit significantly worse than the major part of the dataset. For the velocity magnitude, most of the errors fall between -1 to 1. Due to the low regression performance, most of the errors in the velocity direction fall between high ranges of values from - 40 to 40.
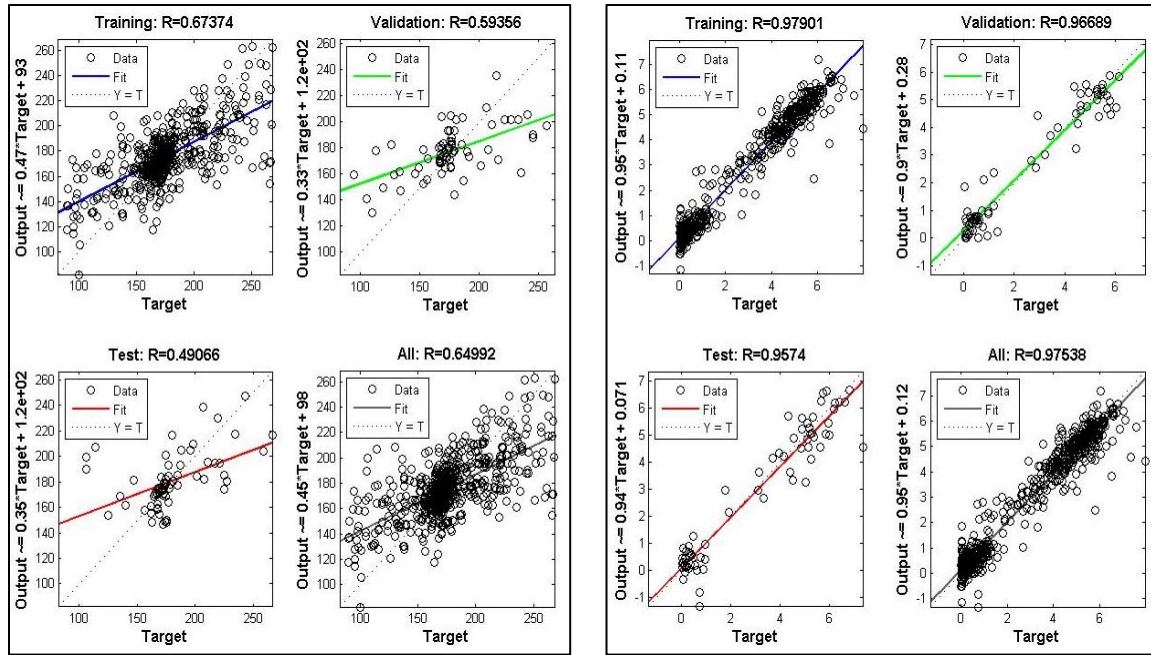
Figure 4.18: Regression plot for velocity direction (on left) and velocity magnitude (on right) for ANN-BP model
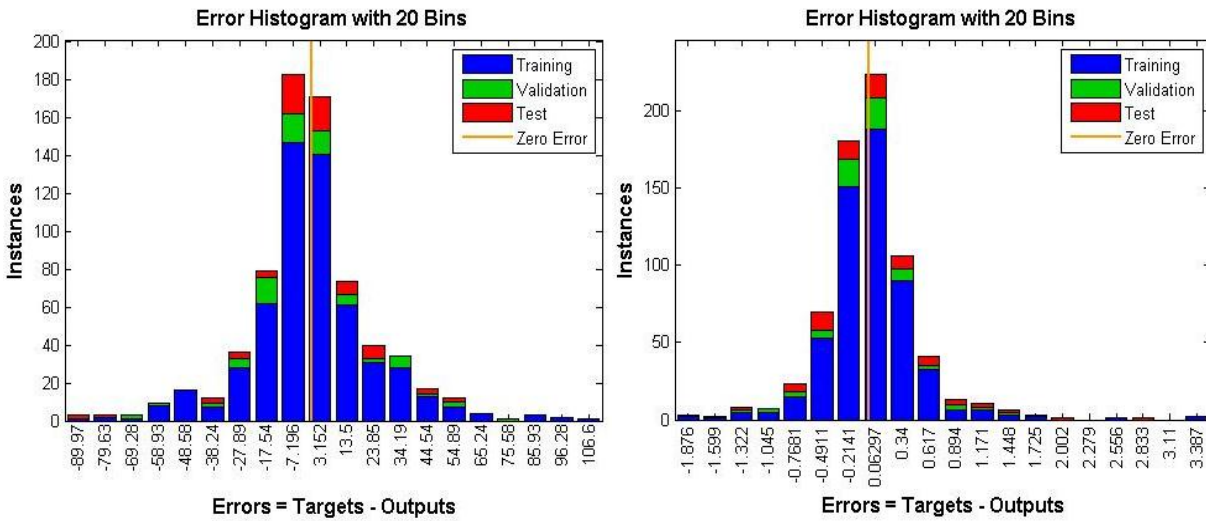

Figure 4.19: Error histogram for velocity magnitude (on right) and velocity direction (on left) for ANN-BP model

4.3.1.3   Model validation for velocity magnitude and direction

In the current study, the ANN model was developed utilizing two ADV sampling stations's data known as the upper cell and lower cell-01. Prediction of velocity magnitude and direction was made using the developed ANN model for those two sampling stations. In order to check the model fidelity, two types of validation were employed. First the modeling outputs generated for the upper cell and lower cell-01 were compared with the observed data from the same stations. This step was taken to determine how closely the model can predict the velocity magnitude and direction to the observed counterparts at the same stations which were used at the training stage. Secondly, the modeling outputs was verified for the remaining two independent datasets at mid-cell and lower cell-02. A statistical comparison between the measured and predicted velocity magnitude and direction is summerized in Table 4.4. For the sampling stations used at the training stage, the model showed a resonable correlation and RMSE for velocity magnitude. However, despite having the stations in the training, the model showed a high RMSE for velocity direction based on the same stations. It can be concluded that the current ANN model performed not quite well at validation stage based on two independent sampling stations.

For the stations not included during the training stage (mid-cell and lower cell-02), the model showed weak correlations in terms of both the velocity magnitude and direction based on the same stations. The RMSE was also quite high in terms of both the predicted velocity direction and magnitude. The likely reason for weak correlations and high RMSE during the validation stage was due to the insufficient number of sampling locations for initial training of the ANN model. In addition, different characteristics between specific locations may have had an impact. The mid cell sampling station was located on the canal that's runs across the cell area.

Large differences in location characteristics during training samples and validation samples, along with the relatively low availability of number of sampling data for training process, collectively deteriorate the validation result of the ANN model. This point enhances the need for a multitude of measuring locations when working in such heterogenous environments.

Table 4.4: Statistical comparison for model validation

Station Type	Independent Stations				Stations Used in Training Process			
Statistical parameter	Lower Cell- 02		Mid Cell		Lower Cell- 01		Upper Cell	
	Velocity Magnitude	Velocity Direction	Velocity Magnitude	Velocity Direction	Velocity Magnitude	Velocity Direction	Velocity Magnitude	Velocity Direction
Correlation Coefficient	0.26	0.32	0.41	0.16	0.92	0.46	0.60	0.67
Root Mean Square Error (RMSE)	4.53	124.12	2.34	121.11	0.65	13.87	0.30	32.86

4.3.1.4 *Two dimensional (2D) map for velocity magnitude and direction*

With the trained ANN model, the prediction of velocity magnitude and direction was made possible for all the cell blocks in the Cell 3B area. Three different scenarios were selected depending on the upstream water stage conditions. The upstream stage was located at the C1-7 block of the cell area. From the predicted time series of velocity magnitude and direction, three time steps were selected corresponding to the three upstream stages to develop a two dimensional velocity map for the Cell 3B area. The time steps considered for generating the velocity map were; (i) 7th Sep, 2013 at 06:27:16 am (for high upstream stage of 3.60m NGVD29), (ii) 3rd Sep, 2013 at 08:27:50 am (for medium upstream stage of 3.51m NGVD29), and (iii) 2nd Sep, 2013 at 11:27:16 am (for low upstream stage of 3.46m NGVD29). The velocity magnitudes found for each block were finally interpolated using NN interpolation techniques to develop three velocity surfaces. The generated velocity maps for different inflow stage

conditions are shown in Figure 4.20, Figure 4.21 and Figure 4.22. Obviously, velocity magnitude and direction are not uniformly distributed in our study and the hypothesis has to be refuted.
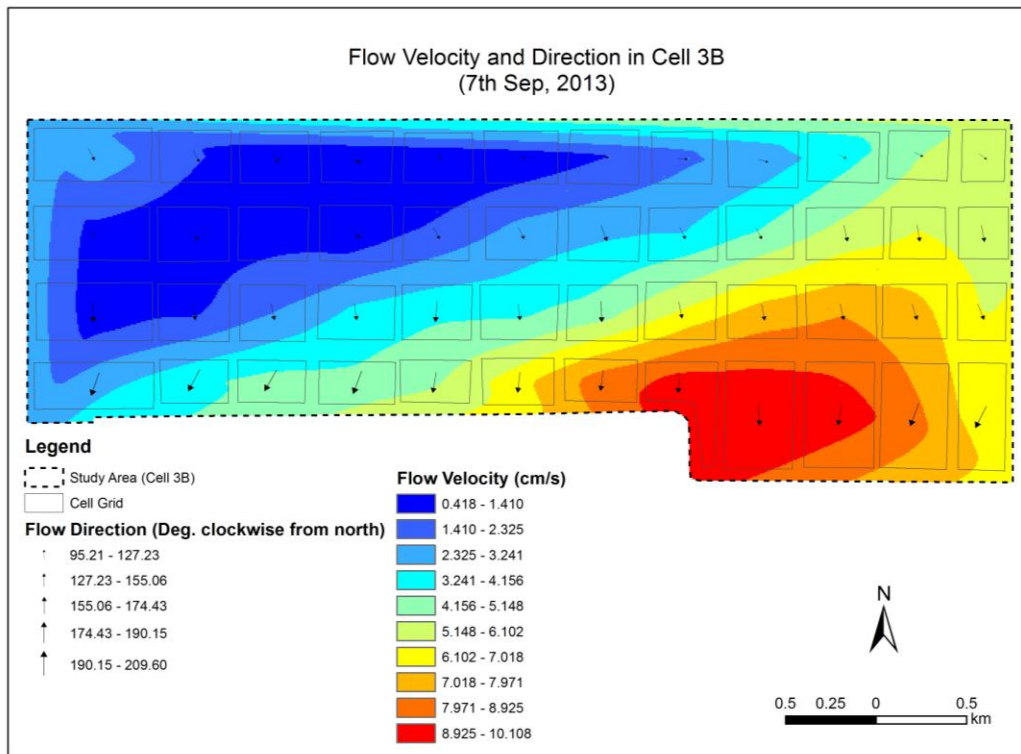

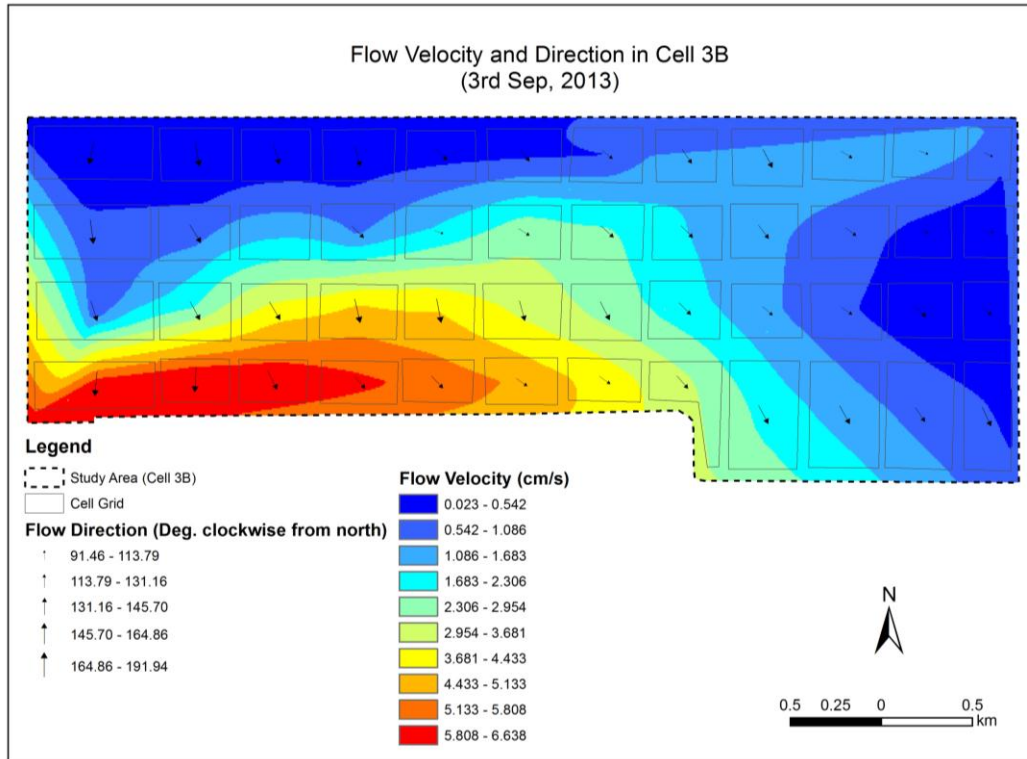Figure 4.20: Velocity magnitude and direction at high stage condition

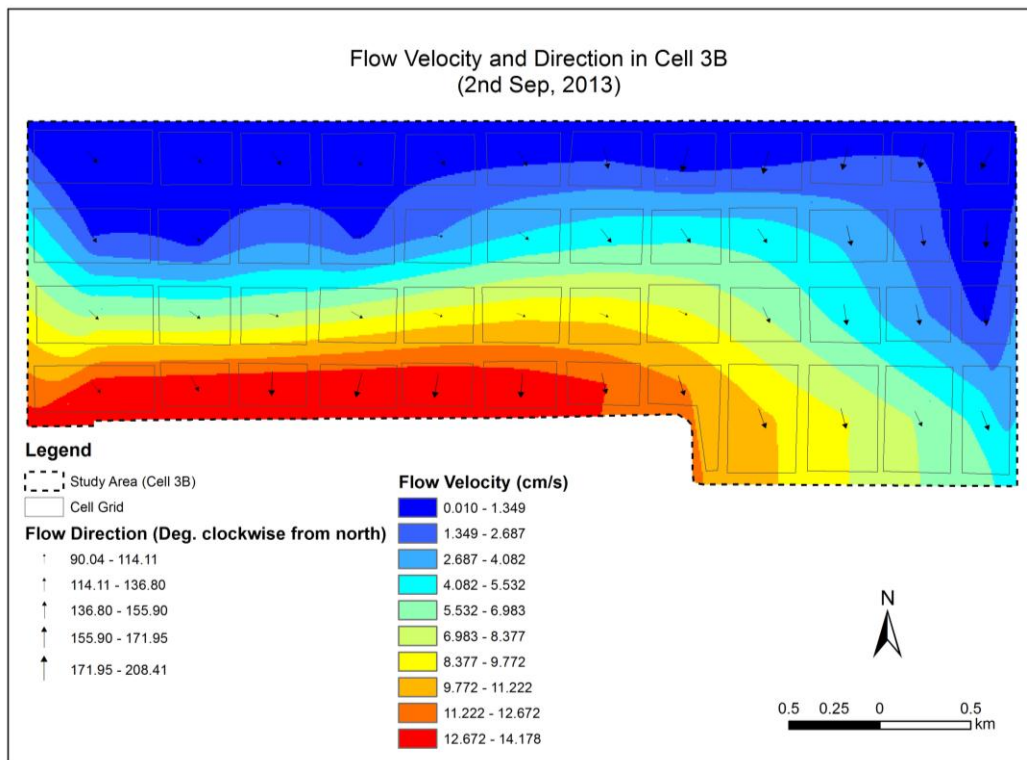Figure 4.21: Velocity magnitude and direction at medium stage condition

Figure 4.22: Velocity magnitude and direction at low stage condition

108

Observing the three velocity maps for different upstream stage conditions revealed some interesting velocity patterns in the Cell 3B area. During the high stage condition, the magnitude of velocity was higher at the eastern side of the study area, while during the medium and low stage condition, the velocity magnitude was higher at the western side of the cell area. The velocity magnitude in the study area was highly correlated with the incoming and outgoing flow from the area. All the inflow and outflow stations were orientated in pairs and every pair was aligned on the same vertical line directed towards north-south direction (Figure 4). The difference between the inflow and outflow was the main driving force on the water to flow from the northern side to the southern side of the area. The change of difference between inflow and outflow, upstream stage with time is shown in Figure 4.23. The flow stations started with an identification of A_C from the eastern side of the area and ended with F_C towards the western side. According to Figure 16, the higher negative value of "inflow – outflow" will dominate the flow velocity magnitude inside the cell area.
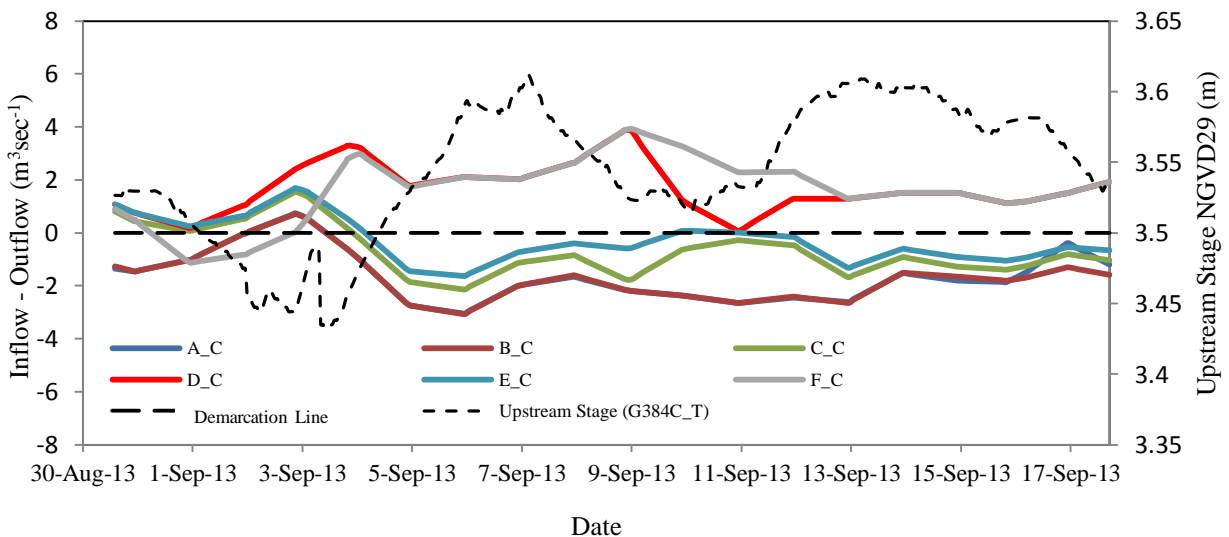

Figure 4.23: Change of flow difference and stage with time

At the high inflow stage condition, the time step considered for developing velocity surface was on 7th Sep, 2013 at 06:27. Considering the same time window in Figure 16, it shows the highly negative value of "inflow-outflow" for stations A_C, B_C and C_C. All these stations were situated at the eastern side of the area and thus validate the model approximation of high velocity towards that side during high stage condition. For the medium and low upstream stages, the time steps considered for velocity surface were 3rd Sep, 2013 at 08:27 and 2nd Sep, 2013 at 11:27. During that time period the F_C station shows higher negative value of "inflow-outflow" than A_C and B_C stations. As F_C station was situated at the western side of the area, the high velocities found using network model prediction towards that side for both medium and low upstream stage condition were quite justifiable.

4.3.1.5   *Two dimensional (2D) map for HRT*

A separate ANN model was developed for predicting stages around the wetland after training the model using three existing stage stations. Using the predicted stage values the HRT for all the cell blocks was generated. Following, a 2D HRT map for the entire study area was developed considering three case scenarios. The time steps considered for those three scenarios were (i) 7th Sep, 2013 at 06:27:16 am (for high upstream stage of 3.60m NGVD29), (ii) 3rd Sep, 2013 at 08:27:50 am (for medium upstream stage of 3.51m NGVD29), and (iii) 2nd Sep, 2013 at 11:27:16 am (for low upstream stage of 3.46m NGVD29). HRT maps for three conditions are shown in Figure 4.24, Figure 4.25, Figure 4.26. Theoretically, HRT increases with the decrease of flow velocity. The same situation was also observed between predicted HRT and flow velocity for all conditions. Obviously, HRT is not uniformly distributed in our study and the hypothesis made before has to be refuted. Distribution of HRT for a wetland system has great importance in its design and operation. Proper design of a constructed wetland system requires

numerous multidisciplinary inputs. Optimal flow hydrodynamics and controlled hydrologic regime are two key parameters within the wetland. A poor hydrodynamic condition inside of a wetland often creates the overall management problem. Full utilization of water storage and flow condition can be ensured by developing high hydraulic efficiency. High hydraulic efficiency depends on the HRT which in turn is affected by the arrangement of vegetation and channels inside the wetland. According to the predicted HRT found for the current study area, a significant portion of the area, especially the central part, possesses HRT < 4 days. To maximize the nutrient removal efficiency using the constructed wetland system in the current study area, it is important to make proper channelization of water through the well-oriented vegetation so that the HRT in the central region can reach beyond 4 days effective HRT limit.
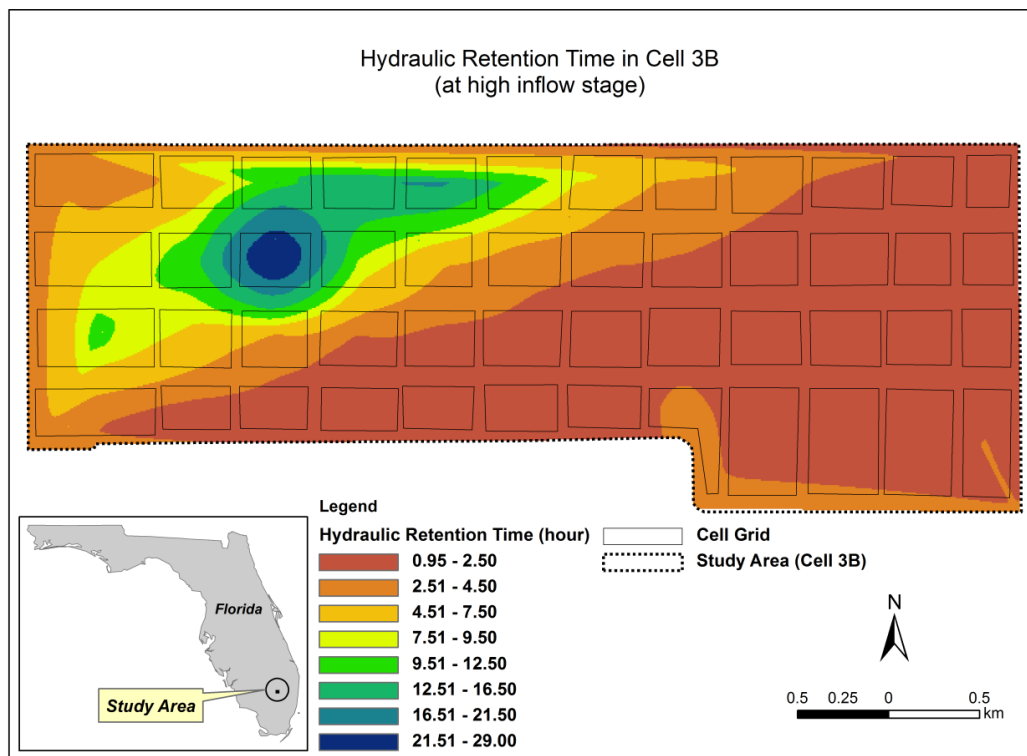
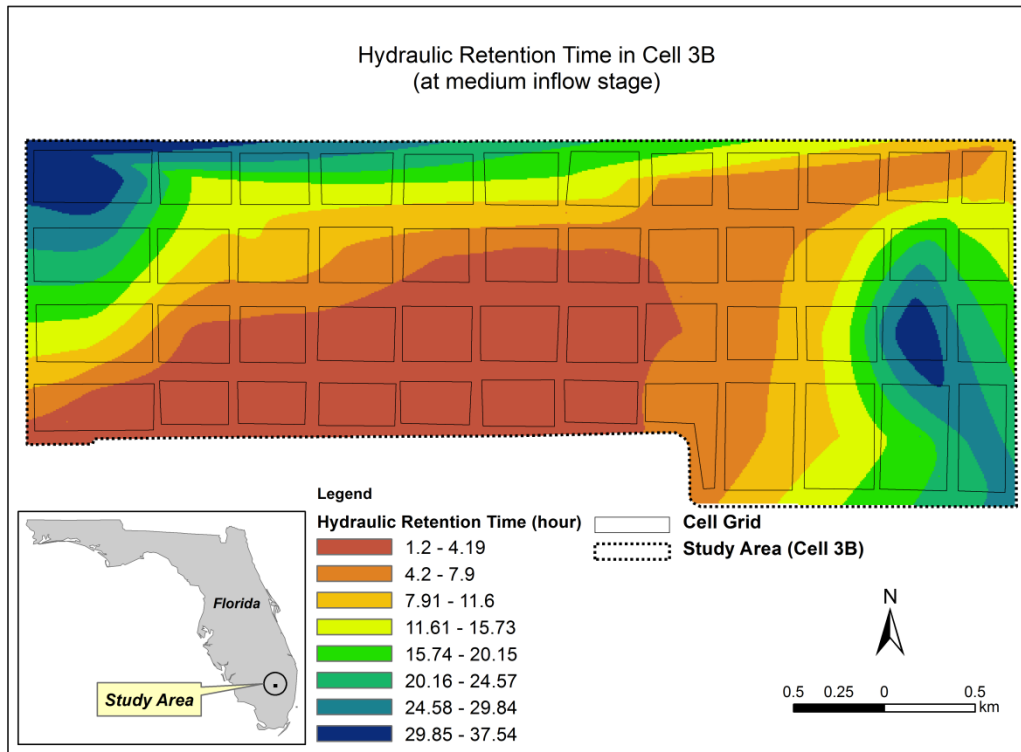Figure 4.24: 2D map for HRT (at high upstream stage condition)

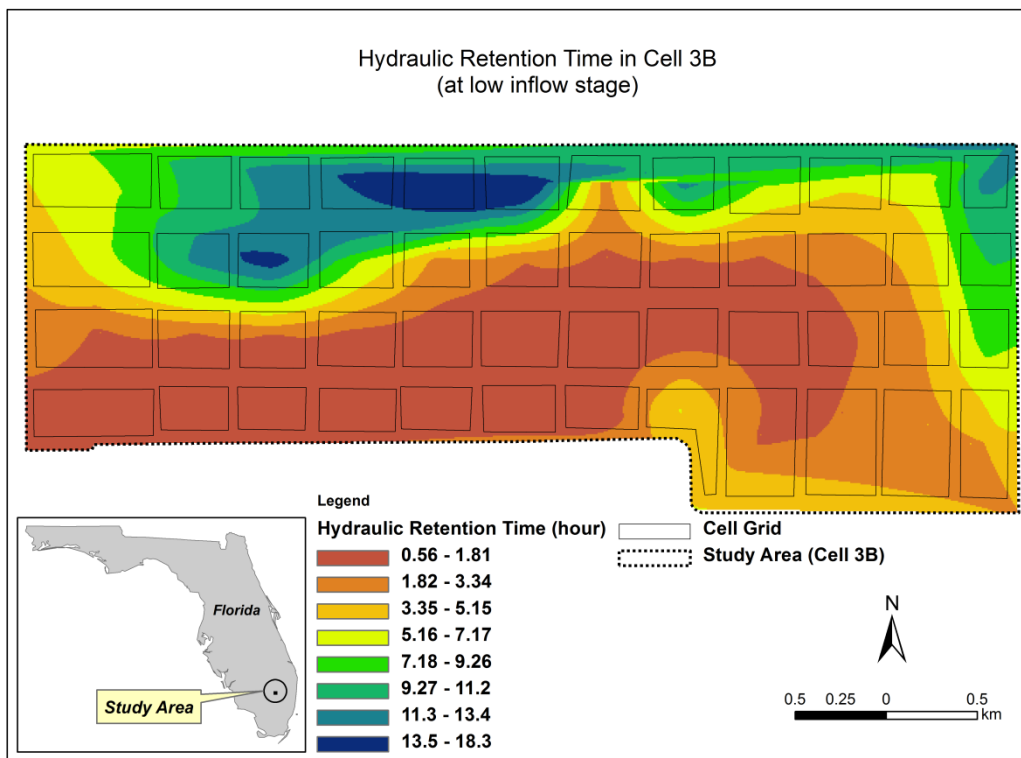Figure 4.25: 2D map for HRT (at medium upstream stage condition)

Figure 4.26: 2D map for HRT (at low upstream stage condition)

112

4.4   Conclusion

With the aid of computational intelligence models, this study creates a lucid linkage between velocity field and HRT such that stage and nutrient removal rates can be conceptually connected in a predictable manner to repeated unsteady flow conditions over time. All the three models, however, are data-driven models that may be functional under limited observations. ANN model generally outperforms GP model in terms of sevral indices. The prediciton of velocity magnitude is betetr than that of velocity direction due to sporadic and sparse vegetation in the wetlands. With only four stations available in the local sensor network, the more parameters fed into the ANN model the more accurate the ANN model. Incrasing the number of observed dataset will not only improve the reliability of the data driven model, but also significantly improve the overall prediction processes. For a complex vegetative area such as Cell 3B, the velocity direction, which are tied to the local turbulence, are dependent on the spatial distribution and stem density of the surrounding vegetation in each block resulting in highly non-uniform velocity field even though the control of gate is in place.

4.5 <u>References</u>

[1] Kadlec, R. H. and Knight, R. L. (1996). Treatment wetlands, 1996. *Lewis, Boca Raton*, 893.

[2] Sakadevan, K. and Bavor, H. J. (1999). Nutrient removal mechanisms in constructed wetlands and sustainable water management. *Water Science and Technology*, *40*(2), 121-128.

[3] Metcalf & Eddy. (1991). *Wastewater Engineering: Treatment, Disposal, and Reuse*. G. Tchobanoglou, & F. L. Burton (Eds.). McGraw-Hill.

[4] Gersberg, R. M., Gearheart, R. A., and Ives, M. (1989). Pathogen removal in constructed wetlands. *Constructed Wetlands for Wastewater Treatment: Municipal, Industrial and Agricultural. Lewis Publishers, Chelsea Michigan. 1989. p 431-445, 5 fig, 4 tab, 42 ref.*

[5] Ball, M. H. and Schaffranek, R. W. (2000). Flow-velocity data collected in the wetlands adjacent to canal C-111 in South Florida during 1997 and 1999. US Geological Survey Open File Report, 00-56.

[6] Maier, H. R. and Dandy, G. C. (2000). Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental modelling & software*, *15*(1), 101-124.

[7] Clair, T. A. and Ehrman, J. M. (1996). Variations in discharge and dissolved organic carbon and nitrogen export from terrestrial basins with changes in climate: a neural network approach. *Limnology and Oceanography*, *41*, 921-927.

[8] Maier, H. R. and Dandy, G. C. (1996). The use of artificial neural networks for the prediction of water quality parameters. *Water resources research*, *32*(4), 1013-1022.

[9] Rodriguez, M. J. and Sérodes, J. B. (1998). Assessing empirical linear and non-linear modelling of residual chlorine in urban drinking water systems. *Environmental Modelling & Software*, *14*(1), 93-102.

[10] Bilgili, M., Sahin, B. and Yasar, A. (2007). Application of artificial neural networks for the wind speed prediction of target station using reference stations data. *Renewable Energy*, *32*(14), 2350-2360.

[11] Pastor, R., Benqlilou, C., Paz, D., Cardenas, G., Espuña, A., and Puigjaner, L. (2003). Design optimisation of constructed wetlands for wastewater treatment. *Resources, conservation and recycling*, *37*(3), 193-204.

[12] Tomenko, V., Ahmed, S., and Popov, V. (2007). Modelling constructed wetland treatment system performance. *Ecological modelling*, *205*(3), 355-364.

[13] Akratos, C. S., & Tsihrintzis, V. A. (2007). Effect of temperature, HRT, vegetation and porous media on removal efficiency of pilot-scale horizontal subsurface flow constructed wetlands. *Ecological Engineering*, *29*(2), 173-191.

[14] Koza, J. R. (1992). *Genetic programming: on the programming of computers by means of natural selection* (Vol. 1). MIT press.

[15] Harris, E. L., Babovic, V., and Falconer, R. A. (2003). Velocity predictions in compound channels with vegetated floodplains using genetic programming. *International Journal of River Basin Management*, *1*(2), 117-123.

[16] Babovic, V. (1996). Emergence, evolution, intelligence; hydroinformatics: a study of distributed and decentralised computing using intelligent agents.

[17] Babovic, V. and Abbott, M. B. (1997). The evolution of equations from hydraulic data Part II: Applications. *Journal of Hydraulic Research*, *35*(3), 411-430.

[18] Whigham, P. A. and Crapper, P. F. (2001). Modelling rainfall-runoff using genetic programming. *Mathematical and Computer Modelling*, *33*(6), 707-721.

[19] Chang, N. B., Vannah, B. W., Yang, Y. J., and Elovitz, M. (2014). Integrated data fusion and mining techniques for monitoring total organic carbon concentrations in a lake. *International Journal of Remote Sensing*, *35*(3), 1064-1093.

[20] Goring, D. G., & Nikora, V. I. (2002). Despiking acoustic Doppler velocimeter data. *Journal of Hydraulic Engineering*, *128*(1), 117-126.

[21] Hagan, M. T. and Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *Neural Networks, IEEE Transactions on*, *5*(6), 989-993.

[22] Sibson, R. (1981). A brief description of natural neighbour interpolation. *Interpreting multivariate data*, *21*.

[23] Wilcock, R., Champion, P., Nagels, J., and Crocker, G. (1999). The influence of aquatic macrophytes on the hydraulic and physicochemical properties of a New Zealand lowland stream, *Hydrobiologia*, *416*(1), 203-214.

[24] Schultz, M., Kozerski, H.-P., Pluntke, T., and Rinke, K. (2002). The influence of macrophytes on sedimentation and nutrient retention in the lower River Spree (Germany), *Water Res.*, *37*, 569-578.

[25] Crowder, D. and Diplas, P. (2002). Vorticity and circulation: spatial metrics for evaluating flow complexity in stream habitats. *Can. J. Fish. Aquat. Sci., 59*, 633-645.

[26] Gurnell, A., van Oosterhout, M., de Vlieger, B., and Goodson, J. (2006). Reach-scale interactions between aquatic plants and physical habitat: River Frome, Dorset. *River Res. App.* 22(6), 1535-1467.

CHAPTER 5: CONCLUSION AND FUTURE DEVELOPMENT

In the whole study the main research objective was to accomplish the implementation of several machine learning algorithms to predict different parameters of constructed wetland by using remote sensing. Often, the remote location of treatment wetlands and insufficient data availability hamper the overall monitoring of nutrient removal efficiency. Traditional hydrodynamic models using differential manning roughness coefficient would have done the similar prediction of flow velocity and direction but such models require numerous number of calibration and validation dataset inside of treatment wetland. Absence of such information might generate erroneous prediction results. For the current study, considering the limited number of available dataset, machine learning algorithms are the most efficient option in order to capture the complex relationships between different eco-system parameters of wetland system.

Machine learning models are normally data driven models. It is always advantageous to incorporate large number of samples to train such models in order to increase the prediction accuracy of such models. Moreover, high training model accuracy not often guarantees high prediction accuracy. In order improve the prediction accuracy, the model train with independent dataset using the same weights and neurons used for initial model training. Implementing such approach will broaden the capacity of machine learning based models with respect to the validation and calibration accuracy.

In this study, the number of machine learning algorithms were limited to ANN based back propagation, extreme learning machine and genetic algorithm. For handling a larger ANN to improve the predictability, a varity of additional algorithms such as conjugate gradient method, deep Boltzmann machine, resilient backpropagation or random forest algorithm may be utilized in the future.

APPENDIX A: EXTREME LEARNING MACHINE ALGORITHM

```matlab
clear
clc
close all

%%
load('data_corrected.mat')
M1=1;
M2=300;
K=M2-M1+1;%size(X,1); %number of data sets
X=XX(M1:M2,:);
Y=YY(M1:M2,:);

%%
N=size(X,2); %number of inputs
M=size(Y,2); %number of outputs
L=900; %number of hidden units

%%
b=rand(1,L); %biases for the hidden layer
W=rands(N,L); %the input weights

%%

T=X*W+repmat(b,K,1);
H=(1+exp(-T)).^-1;

%%
% Hdag=pinv(H,1);
% beta=Hdag*Y;

C=0.9;
beta=H'*inv(eye(K)/C+H*H')*Y;

% beta=inv(H'*H)*H'*Y;

%%
m1=1;
m2=400;
k=m2-m1+1;
x=XX(m1:m2,:);
y=YY(m1:m2,:);
t=x*W+repmat(b,k,1);
t=(1+exp(-t)).^-1;
yh=t*beta;
%%
%Rsquared

% SStot=sum((y-mean(y)).^2);
% SSreg=sum((yh-mean(y)).^2);
% SSres=sum((y-yh).^2);
% R2=1-SSres/SStot
```

```
% R2=SSreg/SStot

m=sqrt(1/k*sum((y-yh).^2))

r=sum((y-repmat(mean(y),size(y,1),1)).*(yh-
repmat(mean(yh),size(yh,1),1)))./sqrt(sum((y-
repmat(mean(y),size(y,1),1)).^2).*sum((yh-
repmat(mean(yh),size(yh,1),1)).^2));
r2=r.^2
```

APPENDIX B: BACK PROPAGATION ALGORITHM

```matlab
% Solve an Input-Output Fitting problem with a Neural Network
% Script generated by NFTOOL
% Created Thu Jan 23 14:37:39 EST 2014
%
% This script assumes these variables are defined:
%
%   Input_STA - input data.
%   Output_V_STA - target data.

inputs = textread('E:\UCF\Proposal\Everglades\ANN Model\ANN
Result\Input_STA.txt')';
targets = textread('E:\UCF\Proposal\Everglades\ANN Model\ANN
Result\Output_V_STA.txt')';

% Create a Fitting Network
hiddenLayerSize = 10;
net = fitnet(hiddenLayerSize);

% Choose Input and Output Pre/Post-Processing Functions
% For a list of all processing functions type: help nnprocess
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};


% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide
net.divideFcn = 'dividerand';  % Divide data randomly
net.divideMode = 'sample';  % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% For help on training function 'trainlm' type: help trainlm
% For a list of all training functions type: help nntrain
net.trainFcn = 'trainlm';  % Levenberg-Marquardt

% Choose a Performance Function
% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse';  % Mean squared error

% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
  'plotregression', 'plotfit'};


% Train the Network
[net,tr] = train(net,inputs,targets);
```

```matlab
% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)

% Recalculate Training, Validation and Test Performance
trainTargets = targets .* tr.trainMask{1};
valTargets = targets   .* tr.valMask{1};
testTargets = targets  .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance = perform(net,testTargets,outputs)

r = corrcoef(targets, outputs);
r2 = r(1,2)^2

% View the Network
%view(net)

% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, plotfit(net,inputs,targets)
%figure, plotregression(targets,outputs)
%figure, ploterrhist(errors)
```