APPLICATION OF ARTIFICIAL NEURAL NETWORKS AS A PREDICTIVE TOOL FOR THE ANALYSIS OF CHEMICAL ENGINEERING PROCESSES

BY

SIPHESIHLE PRAISE-GOD KHUMALO
STUDENT NUMBER: 203515674

MASTER OF SCIENCE IN CHEMICAL ENGINEERING
DISSERTATION

COLLEGE OF AGRICULTURE, ENGINEERING AND SCIENCE
DEPARTMENT OF CHEMICAL ENGINEERING

DATE: 23 NOVEMBER 2017

EXAMINERS COPY

SUPERVISOR: Dr David Lokhat

DECLARATION-PLAGIARISM

I, Siphesihle Praise-God Khumalo, declare that

1. The research reported in this disseratation, except where otherwise indicated, is my original research.
2. This dissertation has not been submitted for any degree or examination at any other university.
3. This dissertation does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This dissertation does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a. Their words have been re-written but the general information attributed to them has been referenced.

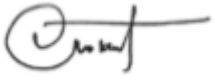
b. Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and refenced.

5. This dissertation does not contain text, graphics or tables copied and pasted from the Interrnet, unless specifically acknowledged, and the source being detailed in the dissertation and in the References sections.

Signed Date

……23/03/2018…….

As the candidate's Supervisor I agree to the submission of this dissertation:



Dr. Lokhat

ACKNOWLEDGEMENTS

- I would like to express my deepest gratitude to my supervisor Doctor David Lokhat for his help, guidance and support throughout the duration of this research.
- I would like to acknowledge Unilever Maydon Wharf and their employees.
- I would like to acknowledge financial support from National Research Foundation.
- I would like to thank the technical team that assisted with collecting data from the factory.
- I would like to offer my deepest gratitude to my wife, family and friends for their constant support, guidance, inspiration and encouragement.

"There is time for everything under the sun"

ABSTRACT

Artificial Neural Networks (ANN) is an example of black-box method that has been significantly contributing in the process optimization and modelling of industrial processes. This study focuses on the application of ANN as a modelling and predictive tool for heterogeneous catalytic reactions and batch processes.

The application of ANN in heterogeneous catalysis represents an efficient and easy technique that integrates factors affecting the rate of a catalytic reaction and catalytic performance, enabling prediction of the catalyst properties and reaction rate. The model developed was utilized in initial screening of different catalyst combinations for heterogeneous reactions of hydrogenolysis and hydrogenation of simple hydrocarbons before experimental laboratory studies. The catalytic hydrogenolysis reaction investigation resulted in an optimum four layered ANN model of (18-12-9-1) with the learning rate of 0.002, R^2 (0.802), SSE (0.0897), RMSE (0.0499) and MAD (0.0343) which provided good quality predictions of ethane, propane, n-butane and n-hexane specific catalytic rate. The catalytic hydrogenation reaction exploration produced an optimum four layered ANN (22-10-10-1) model with the learning rate of 0.002, R^2 (0.7513), SSE (0.1156), RMSE (0.0421) and MAD (0.0299) which indicate good quality predictions of rates for ethylene, benzene, propene and cyclopentane.

The aim of the batch study was to show that ANN could be used to determine the sensitivity of the process performance to various process parameters, both continuous and intangible. The inputs of the process were; quantities, and mass flowrate of different phases, type of variant and time durations for different stages in the process. The outputs were; cycle time, viscosity, pH and specific gravity and material utilized efficiency. A five layer ANN model (23-5-10-7-5) was the optimum model with the learning rate of 0.008 the sum of squared error of 0.0083. The process parameters that had the most significant effect on the process water phase inlet, sonolation time, oil phase inlet and the insensitive ones were aqueous phase inlet, time to raise temperature and addition of liquid ingredients. The entire study confirmed the relevant utilization of optimal ANN models for the prediction of heterogeneous catalyst performance and batch processes operation.

Table of Contents

LIST OF FIGURES

CHAPTER 2

CHAPTER 3

CHAPTER 4

APPENDICES

LISTS OF TABLES

Chapter 2

Chapter 4

.

NOMENCLACTURE

SSE	*Sum of squared error*
RMSE	*Root mean square error*
MAD	*Mean absolute deviatio*n

CHAPTER 1: INTRODUCTION

1. The background of the Study

Artificial Neural Network (ANN) is an example of a black-box approach that has gained reputation and has been much contributing in the process identification, optimization, troubleshooting and modelling of many industrial process during recent decades. ANN function to use in short measure of time compared to conventional methods can improve the model performance (Chen et al, 1991).

The research endeavors to show that ANN can be used to predict and derive meaning from complex non-linear chemical engineering processes and to deduce patterns or trends that are too difficult for other mathematical techniques to find. A final ANN model would enable the user to simulate, visualize and gain a proper understanding of the process, hence be able to optimize base on influential factors (i.e. inputs and outputs of the process).

The literature reveals that most of the previous was done on specific reactions mainly focusing on activity, selectivity and combinations of the catalyst. The work done by the Ibrahim H. et al (2010), dealt with the estimation of catalyst activity, which basically focused on the life span of the catalyst. Also Hernandez-Pichard et al (2017) worked on developing an ANN predictive model for the activity and selectivity of the catalyst. This study present the opportunity to use ANN to rapidly screen potential catalysts before carrying out an expensive and time consuming program.

1.1 The Aims of the Study

The aim was to use ANN as a predictive tool or model in both heterogeneous catalysis and batch process. There are two case studies considered in this work, namely; (1) Artificial Neural Networks as a Predictive Tool for Analysis of Heterogeneous Catalytic Reactions and (2) Artificial Neural Networks in Batch Processes. Case study 1 has part A (Hydrogenolysis) and part B (Hydrogenation).

Case study 1, the application of ANN in heterogeneous catalysis represented an efficient and easy technique that integrates factors affecting the rate of a catalytic reaction and catalytic performance, enabling optimization, development and troubleshooting of catalyst properties and reaction performance (turnover reaction rate). The conventional technique for learning the kinetics of catalyst reactions, entails analyzing experimentally individual catalyst with diverse reaction conditions. Consequently for the assessment of kinetics of several reaction setup result in hefty input and expenses.

Case study 2, the application of ANNs as a tool for modelling and optimization of chemical engineering batch processes. The aim was to show that the ANN could be applied to define the sensitivity of process performance to various process parameters, both continuous and intangible such as human factors. The inputs of this process are; quantities and type of ingredients, temperatures of each phases, mass flowrate of each phases, and sonolator operating conditions. The outputs are; cycle time, physical characteristics of the product (pH, viscosity and specific gravity) and material usage efficiency. The measure of performance was the product flowrate (kilograms per day), and quality of the product. An ANN model will be developed. A final ANN model would enable the user to simulate, visualize and gain a proper understanding of the process, hence be able to optimize based on influential factors (i.e. inputs and outputs of the process).

1.2 Limitations of the Study

The study had the challenge with ethnical issues about confidentiality and privacy of the company. This was the main setback as it delayed the start of data collection by 7 months and even when the data collection process was in progress it had to end earlier than the scheduled dates due to these issues raise by other staff members.

1.3 Dissertation Outline

Chapter 1: Introduction

- Background and Aims of the Study
- Limitations of the Study
- Dissertation Outline

Chapter 2: Literature Review

- Introduction
- Artificial Neural Network Technology
- ANN Operations
- How neural network differ from traditional computing
- Application of ANN and Types of ANN
- Selection of Specific Architecture of ANN
- Training an Artificial Neural Network and Network Selection
- Previous work on application of ANN

Chapter 3: Methodology

- Aims of the work
- Experimental Data
- Methodology
- Compilation and preparation of data
- Design of an ANN architectural Structure and Training of ANN
- ANN Software

Chapter 4: Results

- The results for the research, training and validation.
- Sensitivity

Chapter 5 Discussion

- Summary of findings for both case studies

Chapter 6: Conclusion and Recommendations

References

Appendices

CHAPTER 2: LITERATURE REVIEW

2.1 INTRODUCTION

ANN is an example of a black-box approach which has gained reputation and has been much contributing in latest decades in the modeling and system recognition of various system engineering systems. The superiority of ANN to conventional traditional methods for processes modeling and optimization has been clear especially where there was lack of data about the system dynamics or physics (Asgari et al, 2013).

Fundamentally ANN utilises mathematical procedure that endeavors to mimic the configuration and utilities of biological neural networks. It is empirically developed instead of performing process mass and energy balances. ANN comprises of a network of moderately linked processing fundamentals or nodes, organized in layers, similar to neurons in the brain. Since modeling of the processes is essential for the proper understanding of the operation and process. ANN's outstanding ability to develop significance from complex nonlinear or vague information, is used to excerpt configurations and perceive difficult trends compared to conventional means (e.g. a trained artificial neural network can be proficient in the group of material available to analyze). It can then be used to give projection particular for different conditions of interest. An ANN model, when compared to existing process models, is often more robust and precise in predicting abnormal behavior of a chemical process.

Process development, modeling and optimization entails transforming experimental or simulated information and essential physical and chemical associations into a predictive process model. The approaches used are categorized into three key categories, such as, "*(i) white-box, ii) black-box, and iii) gray-box models*" (Asgari et al, 2013). The use of black-box approach provides a practical method for process modeling and optimization, models are attained directly from simulated or experimental plant data . Black-box models bypass the complex internal structures in the system and present an association among the input and the output. When there is sufficient data about the mathematical dynamics and physics of the system a white-box model is used. When the certain understanding of the system is utilized on black-

box approach, the model is termed gray-box. Industrial systems are modeled for various scenarios for example, (i) situation monitoring, (ii) fault recognition and (iii) sensor justification, analysis, of the system (Asgari et al, 2013).

ANNs have a better filtering capacity than empirical models (Baughman, 2005). A node or neuron operates independently of the others. One can view each node as a processor in its own right, and these processors all operate in parallel. The development of conventional model techniques is time consuming and can be expensive due to various reactions, components and operating conditions (Chen et al, 1991). In addition, since batch reactors are utilized for minor scale production, surplus cost can not be compensated. The main advantages of ANN, (Anderson et al, 1992) unique contain generalization and learning capability of information, and fast calculation capability (Jain, 1996).

ANN are also used for condition monitoring on online industrial sites, process identification or control. As noted by (Himmelblau, 2000) when there is a lack of data in the real time simulation ANNs are very beneficial.

ANNs are widely applied in several areas of industries, and are suitable for chemical processes because to their capacity to generalize and represent multi-variable nonlinear models adequately (Chen et al., 1991). ANN permit superior understanding of the process systems and as a result a better theoretical foundation is achieved for application of scaling-up and automatic control.

2.2 Artificial Neural Technology

2.2.1 Definition of ANN (Similarity to the Brain)

ANNs are comparatively simple statistical models originated from the neural construction of the mind. Basically the mind studies from experience and this mirrors the approach of ANN. The brain modelling assures minimal technical way advance process explanations compared to the traditional technique.

The manner in which the brain stores information is very intriguing, as it has the ability to store very intricate patterns and permit humans to identify different distinct faces from various diverse angles. Hence ANN mimics the brain by recognizing even complex patterns.

2.2.2 How Biological and Artificial Neural Networks Operate

The basic element of a brain a neuron. These cells do not regenerate and they endow humans with capabilities to think, remember, and use prior knowledges to their actions. The neurons have a capability to connect 200,000 other neurons, although 1000 to 10000 is usual (Krisiel, 2005). The intrinsic power of human mind originates from these neurons and the networks between them. ANN endeavor to emulate the basic ability to a natural neural with the objective of solving problems.

ANN only try to mimic or replicate only the elementary parts of this complex, versatile and powerful organism. Ann is about a simpler and new way of solving problem. Fundamentally, a biological neuron obtains inputs from other sources, processes it to a normally nonlinear procedure, and then yields the final outcome (Anderson et al, 1992).

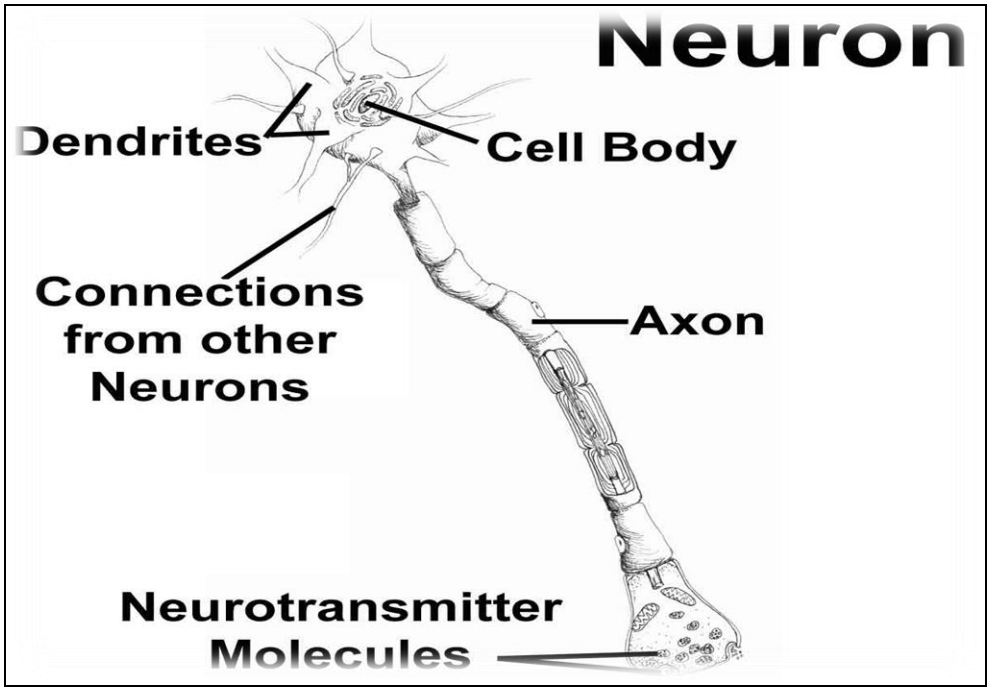


Figure 2.1 Natural Neuron (Catherine E. Myers, (2000))

Dendrites, soma, axon, and synapses are the biological names of four simple components of a natural neuron. The input channels are dendrites and they attained input by synapses of neurons. The data is processed by soma and then axon is the outpu channel. The objective of ANN is not the pretentious regeneration of the brain. Contrary, ANN investigators are investigating the brain to natural abilities to answer problems that are yet to be answered by conventional computing.

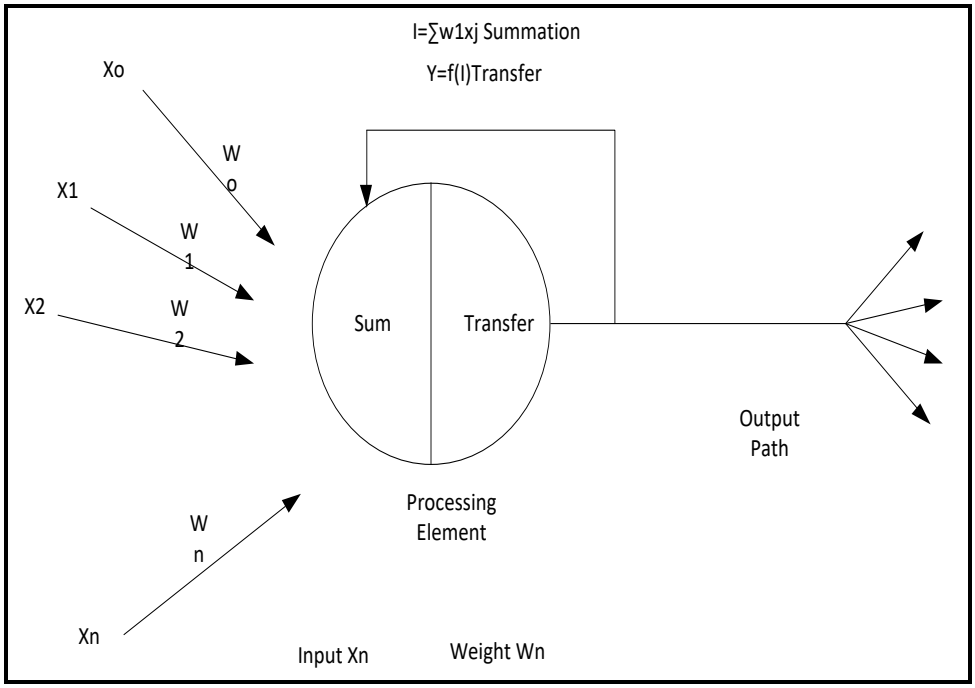


Figure 2.2 A Basic representation of an artificial neuron.

In Figure 2.2, a mathematical symbol *x(n)* represents the various inputs into the network. Each input is multiplied by connection weights and they are signified by *w(n).* In the basic example the yields after being summed are pass into a transfer function to create an outcome and output. There are actually different network structures which uses different transfer function and different summing functions

ANN works well where the resolution or solutions are not just one of the many recognized values and it need to have the capability of producing an infinite number of responses. For example the control behind robotic movements, this some sort of intelligence generates outputs which essentially source some maneuver to move.

These networks come in non-continuous bursts because of limitations of sensors, although they want to smoothen their inputs. They do that by summing the data and then producing an output by using transfer function. In that custom the yield values are constant and they gratify further factual world boundaries.

Further applications by merely summing and comparing to a verge, they produce a one or a zero. Some functions create time-dependent networks by integrating input data over time (Anderson et al, 1992).

2.2.3 Detailed Description of Neural Network Components and Their Function.

2.2.3.1 Weighing Factors

The inputs are multiplied by the factor called weights. The inputs are given relative impact needed on the processing elements by their own relative weights. Weights express the greatness/strength of the input signal in the ANN. They actually accomplish the identical role as synaptic powers of natural neurons. Some inputs may have a better consequence on the neuron or processing element. Weights are altered in reply to the several training sets and with about to network's definite configuration or training rules (Anderson et al, 1992)

2.2.3.2 Summation Function

The primary phase of processing elements maneuver is calculating the sum of all the weighted inputs. The scientific representation of the inputs and the matching weights vectors are $(i_1, i_2 \ldots i_n)$ and $(w_1, w_2 \ldots w_n)$. The product is the entire input signal, inner or dot of the two vectors. By multiplying the each element of the i^{th} vector by the conforming component of the w vector and then totaling up all product, the outcome is a lone measure.

The inner product maximum occurs when the vectors point in the identical direction; The dot product is minimum if the vectors point in the opposite direction. This function can be more intricate as there are many different ways weighting coefficients can be combined. This can also select maximum, majority, minimum, product, or several normalization procedures in addition to simple product summing.

The chosen architecture and paradigm determine the precise procedure for combining neural inputs.

2.2.3.3 Transfer Function

The transformation of an a methodic process that transforms the outcome of the summation function to a operational output. The threshold is compared with summation total the value to define the neural output. The signal is created by the processing element if the totality is larger than the starting point value. No signal is generated if the sum of weight product and the input and is a smaller amount than the starting point. The transfer function or the starting point is commonly linear and non-linear functions because the yield is basically proportionate to the input. The transfer function simple depends upon whether the summation function result is negative or positive unit.. Basically a transfer function is a step function.

Sigmoid or S-shaped curve is yer another option. That curve at asymptoticapproaches a maximum and minimum number at the asymptotic. A sigmoid function while it varies between 0 and 1, and a hyperbolic tangent function varies among negative 1 and positive 1 and both their derivative are continuous.

Regularly dispersed unsystematic noise may be added, prior to applying the transfer function. The network paradigm or topology determine the source and amount of noise, *(*Anderson et al, 1992). The name temperature is results from the inability of people to think as they become too hot or cold. This process is electronically simulated by the addition of noise. (Gaussian noise source). The use of temperature is not being applied to many engineering application and is an ongoing research area.

2.2.3.4 Selecting Transfer Function

The selection of a proper activation function is very vital in data processing of a neural network. The input neurons do not process data, hence their activation function is fixed to the identity function, (Kriesel, 2005). The activation function of the is generally the same for hidden and output neurons. Although a choice of different activation function both output and hidden neurons respectively, can be made.

The use of hyperbolic tangent for the hidden neurons as activation function has been commended although a linear activation function is utilized in the output, (Kriesel, 2005). The output layer still processes data, unlike the input layer that uses linear activation function. As mentioned before that the hidden and output layers usually has the same activation function, nevertheless the linear activation functions in the output might result in enormous training steps and skipping over good minima in the error surface (error). The latter might be circumvented by setting the learning rate to appropriately smaller values, (Kriesel, 2005). The output interval is a bit larger, if the hyperbolic tangent is used. The shortcoming of sigmoid functions is that for values far from the threshold value they hardly learn something, except the network is improved.

2.2.3.5 Scaling and Limiting

The multiplication of the scale factor by the transfer worth and the addition of an offset is scaling. The limiting from here is in adding the one performed by the transfer function. It is mainly applied to test biological neuron models.

2.2.3.6 Output Function (competition)

Single output signal is allowed for each neuron to possible hundreds of further neurons and this is comparable to the natural neuron. Generally the output is equal to the transfer function's outcome. Other network configurations, nevertheless, alter the transfer outcome to assimilate rivalry between regional neurons. Neurons are permitted to challenge each other, hindering neurons except they have countless power. Rivalry can ensue at levels. The first one regulates which artificial neuron will be active or deliver an output and to determine which neuron will partake in the training process (Mohd et al, 2003).

2.2.3.7 Error Function and Back propagation Value

The variance among the existing output and the preferred output is called the raw error or current error and it is transformed by the error function to match a specific network architecture. Some architectures square the error however preserve it sign, some cube the error, other paradigms alter the error to their definite resolutions and

utmost elementary architecture use this error directly. The artificial neural error is habitually transmitted into the learning function of alternative processing function.

Normaly the existing error is propagated backwards to a prior layer. Typically, this back-propagated value, subsequently being scaled by the learning function, is multiplied alongside individually of the entering connection weights to adjust them afore the subsequent learning cycle (Anderson et al, 1992).

2.2.3.8 Learning Function

The target of this function is to change the adjustable linking weights on the inputs of for each neurons using some neural founded procedure. The adaption function or learning mode is a process of altering the weights of the input links to attain nearly preferred product, (Anderson et al, 1992).

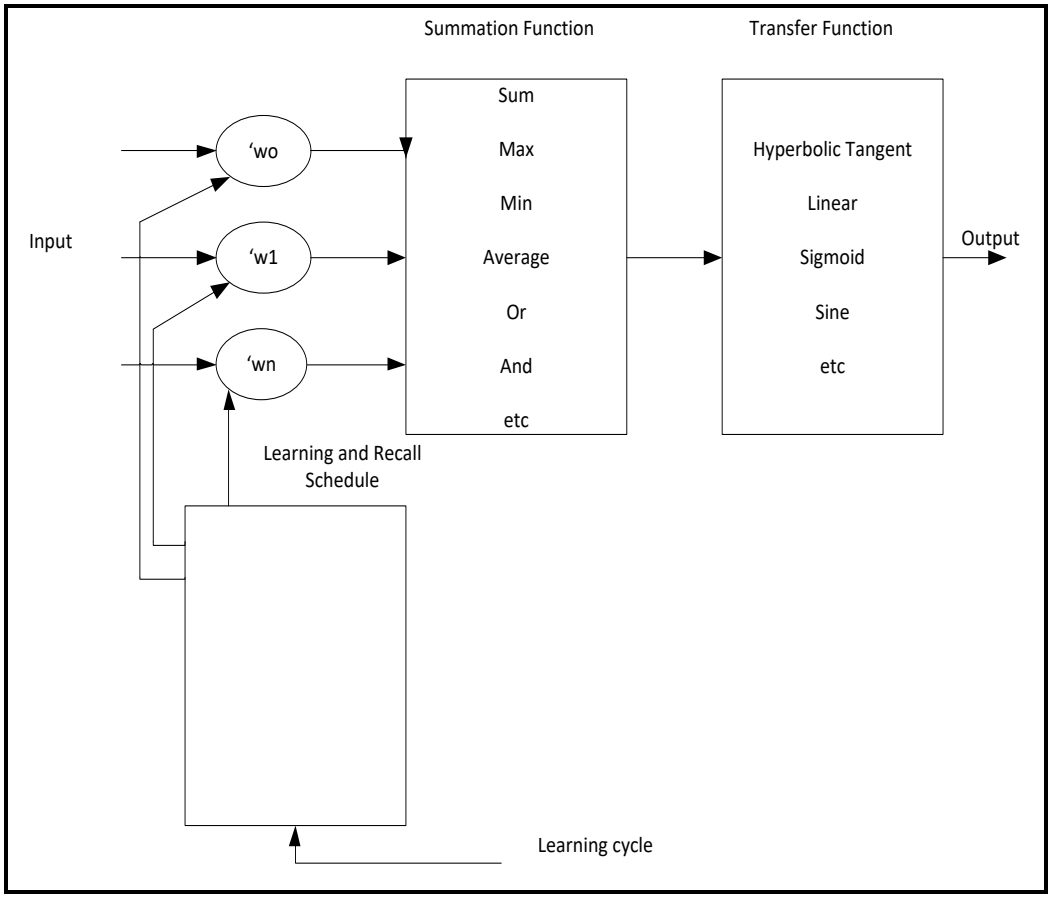


Figure 2.3 A Model of a "Processing Element" (Anderson et al, 1992)

In Figure 2.3, the upper left is the entrance of the inputs go into the neuron from the greater left. The multiplication of the inputs by their corresponding weighing factor ($w(n)$) is the first step. The summing function receive these altered inputs as a feed, where they are commonly summed. There are in fact a number of diverse kinds of that can be selected. Software engineers can generate their peculiar summing functions using advanced equal language.At times the summing function is additional intricate by the totaling of a transfer function which permits the summing function to maneuver in a time sensitive manner (Anderson et al, 1992).

Some algorithm this function turns this numeral into a actual output. The procedure changes the input and turns it into some other number or a zero or a one, a minus one or a one. Generally supported transfer functions are sine, sigmoid and hyperbolic tangent. This function scales the output if there is a need. Usually the outcome of this is the straight output of the neuron. An illustration of how a transfer function functions is presented in Figure 2.4.

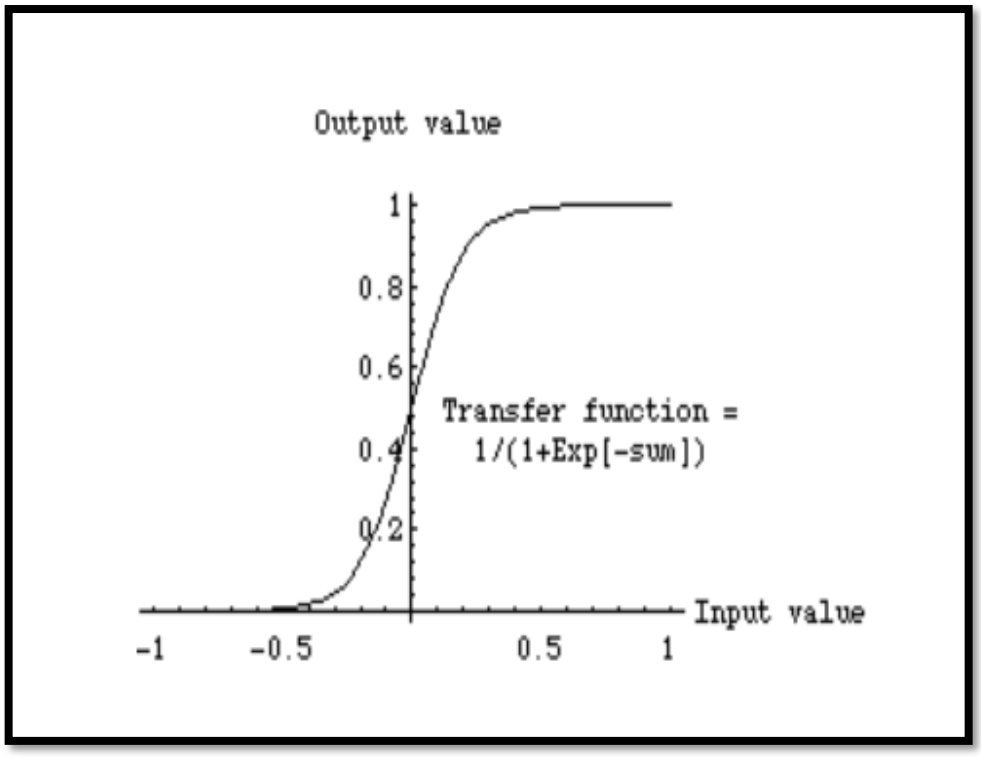


Figure 2.4 Transfer function (Anderson et al, 1992)

The outcome of the output becomes an input into other neurons, as per configuration of the network. Every ANN are made from this simple constructing block - the

processing element or the artificial neuron. As mentioned by (Kriesel, 2005), it is variation and the necessary alterations in these construction blocks which partly cause the executing of neural networks.

2.3 Artificial Neural Network Operations

The important factor that affect the utilization of neural networks entails the myriad or manifold of ways neurons might be arranged. The impact of this clustering is that it allow the information to be processed in an interactive, dynamic and ordered manner as in human minds. ANN are built in two dimensional compared to biological which are fashioned in three-dimensional domain see Figure 2.5.

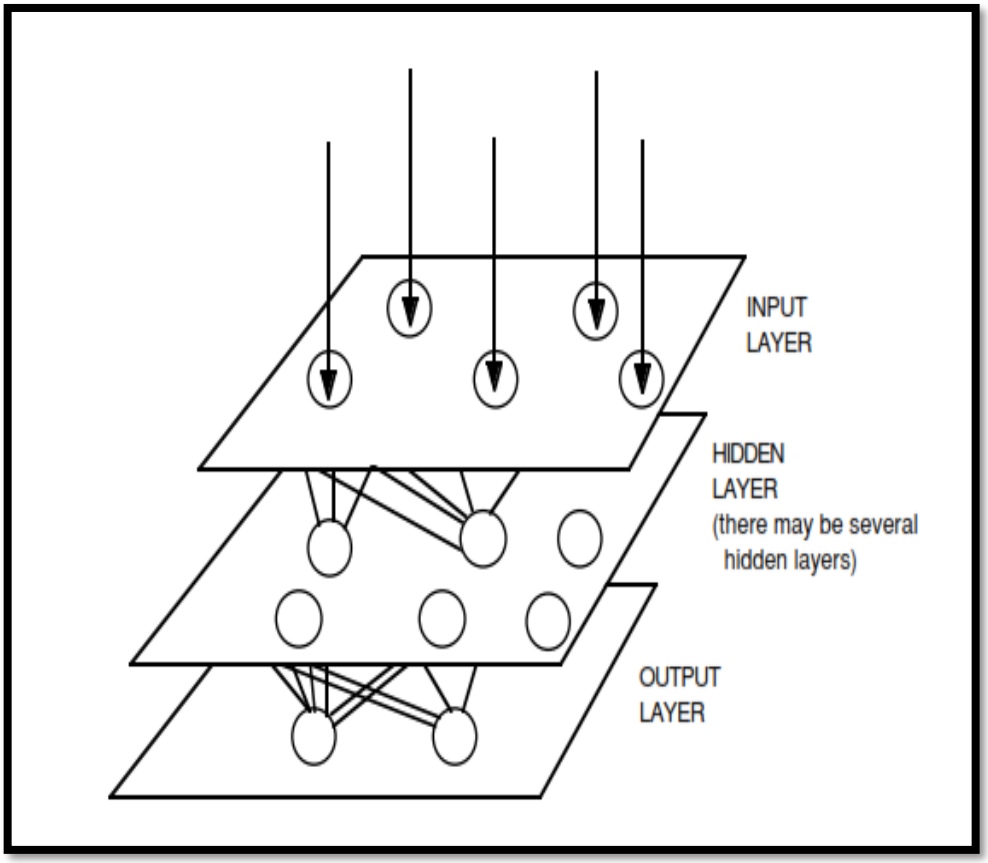


Figure 2.5 A Simple Neural Network Diagram, (Kriesel, 2005)

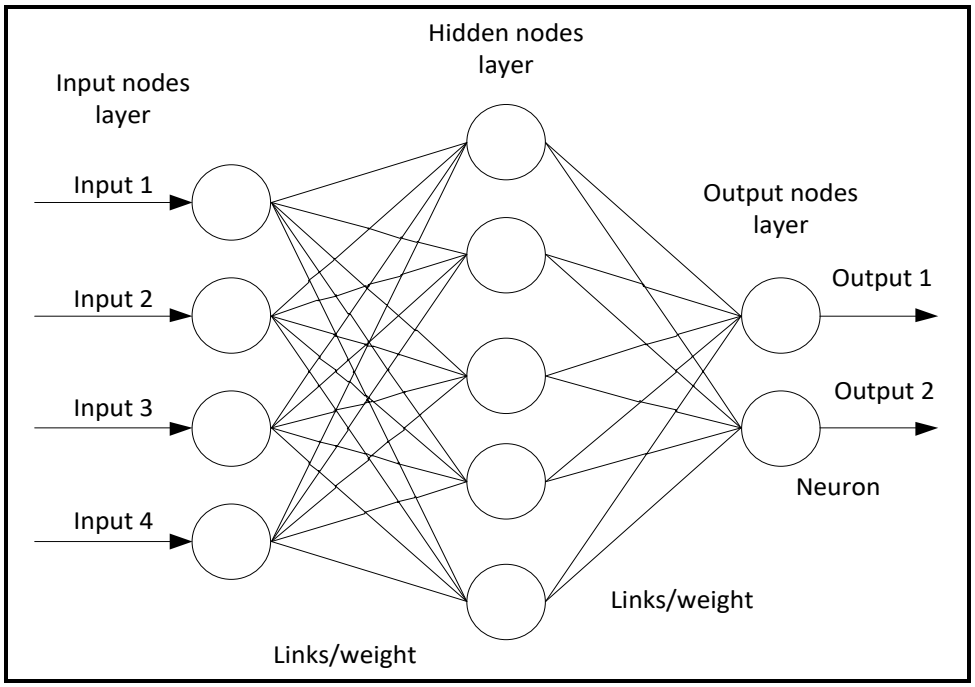


Figure 2.6 Artificial Neural Network

As shown in Figure 2.6, ANN also has similar systematic structure as biological neurons hence the inspiration. There set of input which are than weighted and converted the signals or activations of these neurons are then transfered on to other neurons. This process would be reiterated till an output neuron is activated.

There networks that work on complications where the decisions are of many values Hence they should be proficient of an countless amount of answers. They are usually used in robotic movements, where inputs are processed and the desired outputs are created that cause some device to move.

All ANNs have a like construction or topology. and there is significance in the arranging of neurons in a neural network structure. Creating layers of elements is one of the easiest way of creating a structure. The assembling of these layers comprises of the summation and transfer functions of a network, (Anderson et al, 1992). Utmost presentations of ANN necessitate networks encompass at minimum three normal kinds of layers: input, hidden and output.

The output layer is responsible for sending material straight to the external world, it may be a a control system. Hidden layers are among the input and output layers, they

contain many neurons in numerous connected arrangements. The inputs and outputs of each hidden neurons basically go to next neurons. Typically each neuron in the hidden layer accept the signals from other neurons in an input layer and then after performing a function it passes it to an output signal, which essentially provides a feed forward path to the output.

The communication between the neurons is actually a crucial aspects of neural networks. They make available a adjustable strength to the input. The connections are separated into two the one results in the summing mechanism of the subsequent neuron to increase whereas the other sources it to subtract. In a familiar term or language one excites while the other inhibits. A term called lateral inhibition comes from where some networks inhibit other neurons at same layer and this usually occur at the out layer. This perception may also be termed competition.

The kind of connection or configuration where the output of one layer paths rear to a preceding layer is called the feedback. An illustration of this is shown in Figure 2.7.

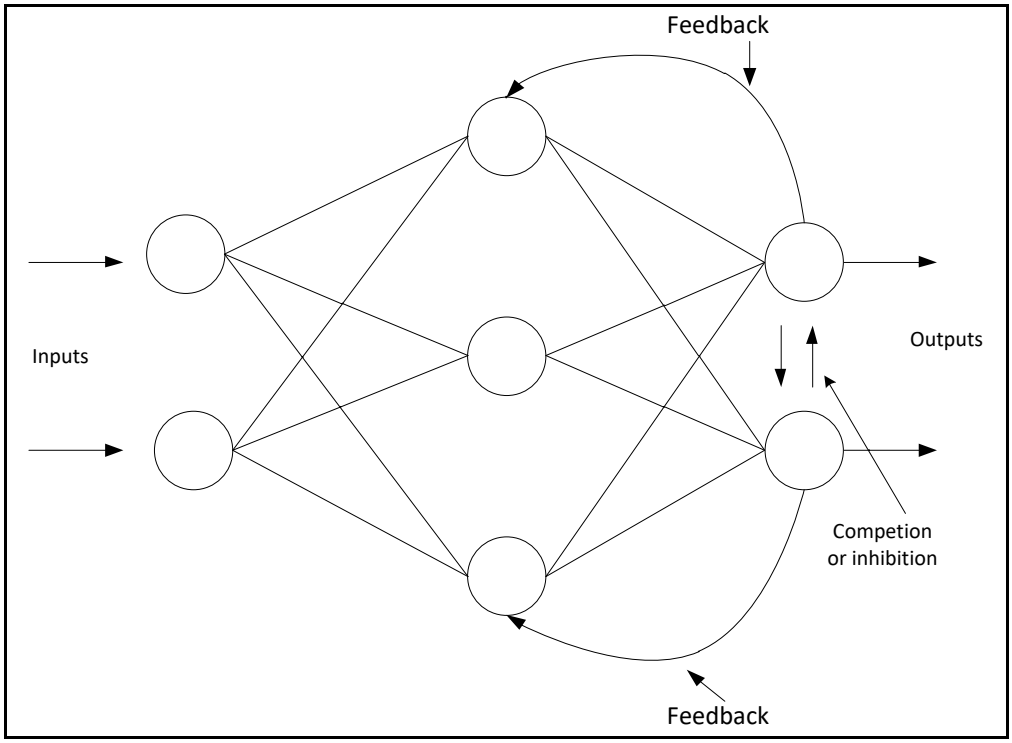


Figure 2.7 Simple network with feedback and competition.

The operation of the network is highly dependent on the manner in which the neurons are connected. The operator is allowed to control, add, and delete these connections at will and these connections can be manipulated to either excite or inhibit.

2.4 How Neural Networks Differ from Traditional Computing and Expert Systems

ANN actually bid a totally diverse method to problem solving. The tool that they produce both learns on its own and programs itself. Their nature gives them the ability to solve problems without programming and without the advantage of a professional. They are proficient of pursuing patterns in data that nobody knows are available. Table 2.1 illustrate the comparison of computing approaches

Table 2.1 Comparison of Computing Approaches

CHARACTERISTICS	TRADITIONAL COMPUTING (Including Expert Systems)	ARTIFICIAL NEURAL NETWORKS
Processing style	Sequential	Parallel
Functions	Logically (left brained) via Rules Concepts Calculations	Gestault (right brained) via Images Pictures Controls
Learning Method	By rules (didactically)	By example (Socratically)
Applications	Accounting, word processing, math, inventory, digital communications	Sensor processing, speech recognition, pattern recognition, text recognition

ANN has met challenges in areas such as synthesis, speech recognition and vision. It is also limited by the speed of the processor that it operate on. Ann's are also troubled by that professionals or expert do not always speak rules. Moreover, Ann are not the complete or ironclad solution.

The neural systems in addition to their demands, they necessitate their (User) to adhere to a numeral of conditions. The conditions are as follows:

- A data set that entails the knowledge that epitomise the problem.
- A sufficient data to be able train and validated the network.
- A comprehension of the elementary type of the problem, so that proper resolution on generating the network can be completed.
- An appreciative of the development tools.
- Satisfactory processing power (applications requirements are sometimes beyond the standard hardware).

The new way of computing necessitates skill above traditional computing. Now there are neural architects as compared to the past when there were system engineers, programmers and data base specialists. Neural architects need to be in a different way than his predecessors of traditional computing. For example it a prerequisite for him to know the statistics in orderto select and asses training conditions. The process of ANN system is definitely not logic, as it encompasses an experiential skill.

2.5 Application of ANN (Advantages and Disadvantages)

Advantages

- ANN extremely nonlinear.
- The structure can be highly difficult or complex, and hence more demonstrative, compared to empirical models,
- The structure need not to be pre-specified,
- They are fairly flexible models.

Disadvantages

- Incapacity to extrapolate
- Entails great training time for large neural networks.
- It’s a black box and it is impossible to explain how the result were calculated in any meaning full way.

2.6 Types of Artificial Neural Networks

2.6.1 Feed Forward Artificial Neural Network (FNN)

FNN are the ANN with the feed-forward topology, where the data flows in a single direction from input to output and there are no rear loops. There are no specific boundaries on the quantity of layers, expect for those imposed by the designer of an ANN topology. Therefore one is free to choose a quantity of connections between neurons and types of transfer function used. A single perceptron is the modest feed-forward ANN that is only proficient of learning linear independent problems (Mohd et al, 2003).

Specialized software and computers are used to construct, mathematically describe or explain and optimize every kind of ANN, due to that FNN can led to comparatively lengthy mathematical explanations, where solving for ANN optimization parameters by hand would be unfeasible.

2.6.2 Recurrent Neural Networks (RNN)

RNN are the ANN with the recurrent confirguration. Although it is analogous to feed forward neural network, it differs because it has back loops hence the information is transmitted in one direction and backwards. Basically an inner state of the network that permits it to display dynamic time-based behavior is created. RNN have the capability to use their inner retention to process any arrangement of inputs, (Mohd et al, 2003).

2.6.3 Hopfield

A kind of a RNN used to store one or extra constant target vectors is called a Hopfield. Constant vectors mentioned can be observed as recollections that the network evokes once supplied with like vectors that operate as a prompt (cue) to the network recollection (Krenker et al, 2010) To determine if the units surpasses their threshold, these binary units only take two different values, they can take either values 1 or -1, or values of 1 or 0. The prerequisite that weights need to be symmetric is naturally used, assure that the energy function drops monotonically whereas subsequent the activation rules, (Krenker et al, 2010).

2.6.4 Elman and Jordan Artificial Neural Networks

Another distinct case of a recurrent ANN is Elman network also known as basic recurrent network. It first layer has a current connection; hence it is different from conventional; two layer networks. Basically Elma network is a three-layer ANN that has rear loop from hidden layer to input layer channel (Mohd et al, 2003). This kind of ANN has a memory that is capable to mutually discover and produce time dependent patterns.

The Elman ANN has characteristically a linear artificial neurons in its output layer and sigmoid artificial neurons in its hidden layer (Mohd et al, 2003). Provided that there are enough artificial neurons in hidden layer, this grouping of artificial neurons transfer functions can estimate any function with random precision. As Elman artificial neural network are able to store information, they are also proficient of producing sequential and spatial patterns. Jordan network is alike to Elman network. They only differ in that the units are fed from the output layer alternative of the hidden layer.

2.7 Selection of the Specific Architecture of an ANN

2.7.1 Network Selection

There are similarities between configurations of neural networks and the major dissimilarities are derived from the several learning rules and how they change a network's typical confirguration. This section entails some of the most common ANN and they are presented simply to avoid some of the misperception over network architectures and their greatest equals to precise presentations.

2.7.2 Networks for Prediction

The prediction of what will most probable happen is the most mutual use for neural networks. In actual fact there are many examples where estimation can assist to set significance. Forecast of the future is what drove the making of networks of prediction, (Anderson et al, 1992).

2.8 Training an Artificial Neural Network

Training of ANN is basically the procedure to attain set of weights that reduce the errors among the predicted and the targeted output of the network. The initial weights are randomly chosen. The network training is preceded by choosing of a structure for a particular application. Training of ANN may also be referred as learning. There are two approaches of training: (i) supervised training and (ii) unsupervised training.

Supervised training entails a method of supplying the network with the preferred yield and working from that to minimize the error or by resulting in the desired yields with their inputs. In unsupervised training the network makes logic of the inputs apart the assistance from the outside. The supervised training is the most utilized. Unsupervised training is mostly used to achieve some preliminary classification on inputs. Although it is fair a possibility which is not fully understood and still a lab subject.

2.8.1 Supervised Learning

In supervised training, both the inputs and the outputs are specified. The network then develops the inputs and a comparison is performed between the resulting outputs against the desired outputs. The calculate errors are use in the system, to instigating the system to alter the weights that regulate the network. This process continues as the weights and or errors are persistently reduced. The set of data utilized in training maybe referred as training set.

The training set should be well prepared and enough in quantity to enable successful network learning. There are cases where some networks never learn. This may be attributed to the lack of information from the input data from which the preferred output is derived. The other reason may be network not converging due to the lack of data to ample training. Superlatively, there need to be adequate training data set so that a portion of the data might be set apart as a test. Since numerous layered networks with manifold nodes are able to memorize data, a set of data should be set aside be utilized to assess the system after training and to observe if the network is basically memorizing data in some non-important way.

There are current marketable network improvement packages that afford tools to observe how well an ANN is converges on the proficiency to calculate the accurate outcome. They permit the iteration to be lengthy, discontinuing provided the system reaches desired precision. The network designer has to review the architecture of a network when the network is unable to resolve the problem. The review of the structure entails reviewing the number of layers, input and outputs, the connections between the layers, the quantity of elements per layer, transfer, the summation, and training functions, and early weights themselves.

A further important portion of the designer's originality rules the rules of training. There are several rules utilized to execute feedback obligatory to alter the weights throughout training. Backward-error propagation, the utmost familiar method is more frequently recognized as back-propagation. The network can be locked and turned in hardware provided the system is appropriately trained, (Mohd et al, 2003). Other systems endure to train whereas used in process and don't lock themselves.

2.8.2 Unsupervised Learning

the network is only provided with the inputs an not desired outputs. The scheme has what may be denoted to as self-organization or amendment, as it must then resolve itself what features it will use to assembly the input data. There is actually limited understanding of unsupervised learning at the moment. This conversion to the environment is the assurance which would permit science fiction kinds of robots to persistently learn on their own as they meet different situations and fresh environments (Krenker et al, 2010) There are investigations that are carried out pertaining unsupervised learning which may lead to more powerful self-learning networks, yet at the present time supervised learning is achieving the results.

2.8.3 Reinforcement learning

A machine based training method that groups factors of an ANN is called reinforcement learning and data is typically not certain, but made by relations with the situation. Essentially reinforcement training or learning is apprehensive with how ANN should take movements in an environment consequently to maximize some concept of long-term remuneration. Initially the definition of a reoccurrence function

that need to be maximized takes place, and then the reinforcement learning utilizes numerous procedures to discover a procedure which generate the maximum yield. The algorithm first step is to compute reoccurrence function for each likely policy and indicates the policy with the largest return (Krenker et al, 2010). In case of particularly great or even immeasurable number potential policies, disadvantage of this algorithm is very apparent. *Reinforcement learning is mainly appropriate to complications that contain a long-term vs. short-term trade-off* (Krenker et al, 2010). Reinforcement application examples are in telecommunications, chess games and other chronological production tasks.

2.8.4 Learning Rates

There are actually various trade-offs to take into account when selecting the approach of ANN learning as the proportion at which they learn depends upon numerous manageable issues. In case of faster learning rate the system that learns more slowly may be challenged to create fine discriminations (Mohd et al, 2003). Whilst a gentler rate a portion further time exhausted in achieving the off-line learning to generate an sufficiently taught system. The best of worlds are yet to be enjoyed as researchers in the process of investigation. There several factors that needs to be deliberated while dealing with the off-line training duty as they have a major input on the duration of network training and accuracy thereof. These factors are the network size, architecture, complexity, pattern assortment and kind of learning regulation utilized**.**

For the learning rate of superior than one, it is likely for the learning technique to passes in improving the weights and that may result in an oscillating network. While the small values may be slow in correcting the current error, although there is a worthy possibility of attaining the greatest minimum convergence when small steps are taken in correcting the errors.

2.8.4.1 The Effect of Learning Rate Selection

The selection of the learning rate is very essential as it determines the learning procedure and behaviour of backpropagation. Another point to note is that the variation in the learning rate is proportional to change weight (w) .

The learning rate always controls and is always proportional to the speed and accuracy of the learning procedure. If the learning rate value chosen is greater, then the oscillations are too large on the error surface, (Kriesel, 2005). As a result the movements across the error surface would be uncontrolled, (Kriesel, 2005). Therefore a small value of the learning rate is the desired input, although it result to huge amount of training time. Generally one optimize an ANN model by starting with a large learning rate and gradually reduce the rate to help the model to focus on one particular local optimum.

2.8.4.2 Learning Rate Variation over Time

Throughout training, an additional technical device can be an adjustable learning rate: In the commencement of the ANN training, a large learning rate results to worthy outcome, nonetheless learning can be erroneous in the later. The result are more accurate fo a smaller rate though it is more time overwhelming, (Kriesel, 2005). Accordingly, throughout the learning process the rate desires to be reduced an order of one size once or repetitively. To just continually decrease the learning rate is a most common error, which results in the network getting. The key is to somewhat decrease the learning rate progressively as declared above.

2.8.5 Learning Laws

Although understanding of how neural processing operates especially limited and learning is particularly more complex, major laws were developed for simplified representation of natural neuron. These laws were developed from biological learning and or on how nature handles learning. Most of these laws are more or less variation of the Hebb's Rule. Only a couple of chief laws are mentioned in this study.

Hebb's Rule:

The weight between the neurons is strengthened If a neuron obtains an input from another neuron, and are both extremely energetic and they have the same mathematical sign. This rule is usually generalized as

$$\Delta w_i = \eta x_i y$$

The postsynaptic response *y* for a single neuron

$$y = \sum_i w_j x_j$$

Hopfield law:

It is relatively similar to Hebb's Rule. If both the input and desired output are either active or inactive, increase the links weight by the learning rate, then lessening the weight by the learning rate.

Delta Rule:

As mentioned previously that the learning laws are a variations of Hebb's Rule, there is also no exception with Delta Rule. The rule reduces the mean root squared error by the changes in the synaptic weights. This rule may also be denoted to as the Least Mean Square and Widrow-Hoff Learning Rule. This rule is established on the modest idea of constantly changing the strengths of the input connections to decrease the variance (the delta) among the anticipated output value and the real output of a neuron (Mohd et al, 2003).

The delta error in the output layer is back-propagated into the preceding layers till the first layer is reached. The Feedforward, Back-propagation network derives its name from this technique of computing the error . It is imperative that the input data set is random when using this rule. The disadvantage of a well ordered arrangement of training data set, is that it results to a non converge network to the preferred accuracy hence the network will be unable of learning the problem.

For a activation function g(x) with a neoron *j*, the delta rule for *j*'s *i*th weight w_{ji} is given by

$$\Delta w_{ji} = \propto \left(t_j - y_j\right) g'(h_j) x_i$$

Where

α is a small constant called learning rate

g(x) if the neuron's activation function

t_j is the target output

h_j is the weighted sum of the neuron's inputs

y_j is the actual output

x_i is the $_i$th input

It holds that $h_j = \sum x_i w_{ji}$ and $y_i = g(h_j)$

The delta rule is commonly stated in simplified form for a neuron with a linear activation function as

$$\Delta w_{ji} = \propto (t_j - y_j)x_i$$

The Gradient Descent Rule:

It is comparable to Delta rule, the delta error is applied to the connection weights the before derivative of the transfer function is utilized to transform it. In this rule an extra proportionate continuous tangled to the learning rate is added to the final transforming factor performing upon the weight.

Although it is very slow to converge stability very slowly, this rule is commonly used. It is known that for the learning process converge faster using diverse learning rate for diverse layers of a network aid (Mohd et al, 2003). In these tests, the learning rates for those layers close to the output were set lower than those layers near the input).

For one output neuron

$$E = \frac{1}{2}(t - y)^2$$

Where

E is the squared error

T is the target output for a training sample, and

Y is the actual output of the output neuron

w_{ij} *denotes the weight between neurons i and j*

α denotes must choose a learning rate

$$\Delta w_{ij} = -\propto \frac{\delta E}{\delta w_{ij}}$$

The -1 is required in order to update in the direction of a minimum, not a maximum, of the error function. For a single-layer network, this expression becomes the Delta Rule.

Kohonen's Learning Law:

In this law the neuron contest for the chance to learn and modernize their weights. The neuron with the major output is professed the victor and has the proficiency of barring its contenders as well as stimulating its neighbors. Only the conqueror is allowed an output, and only the champion and its neighbors are permissible to regulate their connection weights. Additional, the degree of the region can differ through the training period. Typically one starts with a large region and decrease as the training process continues. Kohonen networks model the distribution of the inputs. This is deemed worthy for statistical modeling of the data and is occasionally denoted to as self-organizing maps.

2.9 Network Selection

2.9.1 Determining the number of layers and hidden layers in an ANN

ANN can have two or three layers but more are used conditional on the problem complexity. Those layers are one input layer and output layer and at least one hidden layer respectively. It can be mathematically proven if need be that MLP (multi-layer perceptron) with one hidden neuron layer is proficient of estimating random functions with any exactitude. But it is crucial that a perceptron can, in standard recognise mapping (representability) and can be taught (learnability). In this admiration since several problems are caused by hidden layers, practice displays that at least two hidden layers or there can be very beneficial to solve a problem, (Kriesel, 2005).

It is important that one should note that any extra layer generates further sub-minima of the error function in which a network can get stuck. Therefore it is advantageous to start with one hidden layer initially and then two layers, then consider more layer if it fails, as the computers are more than couple of handling more layers.

In instance of difficulties which deals with random resolution boundary to random precision with balanced activation functions then two or three hidden layer can be used. It is imperative that the quantity of neurons required in each hidden layer need to be computed. Under fitting may occur in a case where the number of neurons are fewer as related to the difficulty of the problem information. Basically under fitting happens when there are fewer neurons in the hidden layers to sufficiently recognize

the signals in a complex data set. For to many neurons present in the network then overfitting may occur, (Karsoyila, 2012).

Firstly approximation of number of hidden nodes, layers and neuron in each layer is crucial. Generally some standard approaches are used to determine the quantity of neurons in the hidden nodes.

- The quantity of hidden layer neurons are 70% to 90% of the magnitude of the input layer. If insufficient more neurons can be added, (Boger, 1997)
- The quantity of hidden layer neurons is fewer than double of the amount of neurons in input layer, (Berry, 1997)
- The magnitude of the hidden layer neurons is among the input layer magnitude and the output layer magnitude", (Blum, 1992)

The approximation of the number of hidden layer and nodes rely on the kind of the databank models for which the network is calculated. The more the number of hidden layers, that results in the increase of complexity of the problem. There it is imperative that afore designing the neural network, training data set samples must be evaluated so that guesstimate of quantity of neurons and hidden layers can be predicted appropriately, (Karsoyila, 2012).

Once one has chosen the type of ANN to utilize for an application, the following step is to divulge the specification with regard to the connections between the nodes (transfer function) and their structure. There are several the techniques that have been utilized to carry out this task (Krenker et al, 1992).

These are the characteristics of an appropriate size network:

1. Good generalization, when predicting new data and by circumventing under- and over-fitting.
2. Computational effectiveness; the smaller the network the less the parameters, fewer information is desired, and less identification time.
3. Better explanation of the input-output relative.

A proper or “acceptable" network is what one desire to make predictions or classify data. The result with considerable redundant information are expected after that training is complete, if one starts the training with further nodes and connections than then one plan to end up with. In case of redundancy in information, pruning of the nodes and or links from the network devoid of expressively demeaning performance. There are two methods of pruning and they both require parameter tuning to attain proper performance.

The sensitivity technique is the first, where the sensitivity of the error function is assessed after the network is achieved (Lee et al, 1991) and then the pruning take place to the nodes or weights with the lowest sensitivity. “*The second technique entails the addition of terms to the objective function that prunes the network by dividing some weights to zero during training”* (Kamruzzom, 1992); (Reed, 1993). The recommended alternative way in constructing a network is to commence with a smaller quantity of hidden nodes and build up by adding new nodes or divided existing nodes if the network is not adequate (Mohd et al, 2003). Pruning is identical to backward elimination and increasing to forward selection in deterioration (Mohd et al, 2003).

2.9.2 Feedforward Back-Propagation Network

In the early 1970’s the feedforward, back-propagation configuration was developed by Werber; Parker: Rumelhart, Hinton and Williams. This architecture is the most efficient and most used in many diverse applications. It has produced a great group of network kinds with numerous diverse training methods and topologies. The back-propagation network usually has an input layer, an output layer, and at least one hidden layer. From history there is normally one or two hidden layers while there is no actual limit. A work done by (Anderson et al, 1992) Indicate that a maximum of four layers are essential to answer problems of any difficulty.

The flow of data during the recall is indicated by the in and out layer of the figure above. Recall is basically a process of placing input data into a trained network and receiving the answer. The back-propagation is only used through training set and not during recall. The very important decisions entails the number of layers and the number of nuerons per layer. These parameters are the pinnacle of the network

designer, therefore very fragile. There is actually no one way or finest solution to the topology of the network, but general rules are picked with time of utilization.

Rule one: The amount of the neurons in the hidden layer should grow as the intricacy in the relationship between the input data and the desired output increases.

Rule Two: If the modelled process is divisible into manifold stages, the extra hidden layer(s) may be essential. In the case where the process is not discernable into phases, then extra layers may basically permit memorization and not a true genera solution.

Rule Three: An upper bound for the number of neurons in the hidden layer(s) is set by the quantity of the training data sets. To determine this upper bound, use the number of input and output pair example of the training set and split that number by the entire quantity of input and output neurons in the network. Then a scaling factor between 5 and 10 is used to divide the result. For relatively noisy data larger result are used.

A exact spotless input data and precise connection to the output require a factor about two, while exceedingly noisy data may necessitate a factor of twenty or more. It is imperative that the hidden layers have few neurons or memorization may occur. In that case no overview of the data trends will happen.

The learning starts after the rule to create a network have been applied. The learning of the feed forward network utilizes some variant of the Delta Rule, which begins with the computed difference among the actual outputs and the desired outputs. Then the error, connection weight are increased in ratio to the error multiply the scaling factor for total precision (Mohd et al, 2003).

The will be no need to change the weight of an inactive node and it will not contribute to the error. The challenge is tackled by applying the inputs to the input layer of the network, and desired output are compared at the output layer. A forward sweep is made during the learning process through the network, and the computation of the output of each element is one layer through layer. The alteration among the output of the last layer and the desired output is back-propagated to the preceding layer(s).

The learning rules for back-propagation networks has many variations. Different functions can be used such as, transfer functions, diverse error functions, and even the altering technique of the derivative function. The notion of "momentum error" was presented to permit for further rapid learning whereas reducing unbalanced performance. In engineering terms this performs, as a low-pass filter on the delta weight terms meanwhile common tendencies are strengthened whereas oscillatory conduct is annulled out. This lets a little, generally leisurelier, training coefficient to be used, but makes faster training .

Another method that has an influence on convergence rapidity is rather than updating weight after every presentation, but only apprise the weights after numerous pairs of inputs and their output are accessible to the network. This is termed as cumulative back-propagation since the delta weights are not accumulated till the whole set of pairs is obtainable. "Epoch" is term given to the amount of input-output pairs that are existing during the accumulation, and it may agree either to the whole training set pairs or to a subsection.

The restrictions to the feed forward back-propagation structural design, are that it necessitates loads of managed training with plenty of input-output samples. Moreover the inner mapping processes are confusing, and there is no conviction that the scheme might converge to a satisfactory solution. Sometimes, the learning becomes caught in a resident minima, restraining the superlative answer. Several learning uses increase a term to the calculations to knock or jerk the weights past low barricades and discover the real minimum rather than a provisional error pocket.

2.9.3 The Utilization of Training Data Set

It is fascinating to observe whether the network after the learning process has only memorized and actually use the training data sample to yield the accurate output nevertheless produce erroneous result for other similar data samples. The aforementioned can be due to undersize or oversize network, meaning too much storage capacity and insufficient storage capacity respectively. The training data set is separated into two, the training set used to train and the other one used for

verification (verification set). Usually the division relations are, for example 70-80% for training data and 30 to 20 for confirmation, and the data arbitrarily selected.

Tailoring the network to the verification data can occur when the verification data is included in the training data set and that can be avoided by having a third set of data for validation after fruitful training, depending on the size of data set available. By training less patterns, one clearly reserve data from the network and danger to deteriorate the learning presentation, (Kriesel, 2005). Nonetheless the chief aim is to attain successful generalization and approximation for the network and not 100 percent exact reproduction of the known samples. Generally the network stops learning when the user has confidence in results and most importantly when the error is small enough or all the time spreads the similar final error-rate for diverse random or repetitive training and initializations.

In general an ANN can have two or three layers but more are used reliant on the complexity of the problem. Those layers are one input layer and one output layer and at minimum one hidden layer. It can be mathematically proven if need be that MLP (multi-layer perceptron) with one hidden neuron layer is proficient of approaching random functions with any precision. But it is crucial that a perceptron can, in principle realize mapping (representability) and can be taught (learnability). In this respect, practice displays that two or three hidden neuron layers can be very beneficial to resolve a problem, (Kriesel, 2005).

It is important that one ought to note that any extra layer creates extra sub-minima of the error function in which a network can get stuck. Therefore it is advantageous to start with one hidden layer initially and then two layers, then consider more layer if it fails, as the computers are more than couple of handling more layers.

2.10 Previous Work on Application of ANN

Computational catalyst research has been a goal of screening and design of catalysts using first-principles techniques. Many worthwhile approaches to model heterogeneous catalytic reactions have been developed, the most successful involves the use and the development of "descriptors."

Descriptors are basic thermodynamic or kinetic parameters that are directly associated to the catalytic properties of the material and can be easily evaluated using

electronic structure calculations. In a representative descriptor-based catalyst exploration, an estimated practical relationship between the descriptor and the catalytic properties of interest, including activity, stability, and selectivity, Franklin Tao et al, (2015).

The literature also reveals that most of the previous work was done on specific reactions mainly focusing on activity, selectivity and combinations of the catalyst. The work done separately by the Ibrahim H. et al (2010) and Hernandez-Pichard et al (2017) dealt with the estimation of catalyst activity and selectivity.

CHAPTER 3: METHODOLOGY

There were two case studies considered in this work, namely; (1) Artificial Neural Networks as a Predictive Tool for Analysis of Heterogeneous Catalytic Reactions (2) Artificial Neural Networks in Batch Processes.

3.1 Case Study 1 (Entails Part A: Hydrogenolysis and Part B: Hydrogenation)

3.1.1 Aims of the Work

The chief aim was to evaluate the application of ANNs to predict the performance of a catalytic reaction (specific turnover rate) under different reaction conditions. These conditions were: temperature, molecular structure of reactants, reaction concentration and catalyst properties: (type and loading of the active metal component, type of support, surface morphology (surface arrangement of metal atoms) and pre-treatment. These relationships are illustrated in Figure 3.1.

Modelling and simulation is a prerequisite to optimizing process procedure hence it can be utilized for a superior appreciative of the process operation. ANNs can be very beneficial for the real time simulation particularly when there is lack data about the process dynamics. It was postulated that ANNs can be practically applied in reactions where there are no experimental results available or kinetic equations.

The objective of this work was to develop a predictive tool to be used for screening different catalysts for various hydrocarbon transformations. The tool can also be used to replace existing processes and reduce environmental impact**.** As catalytic reactions are favored in ecologically kindly green chemistry owing to the decreases quantity of waste produced compared to stoichiometric reactions in which all reactants are expended and additional side products are made, (Serra et al, 2003). The reactions studied were catalytic hydrogenation and hydrogenolysis of simple organic compounds. The study entailed developing and training of a neural network using data obtained from (Sormojai, 1994).

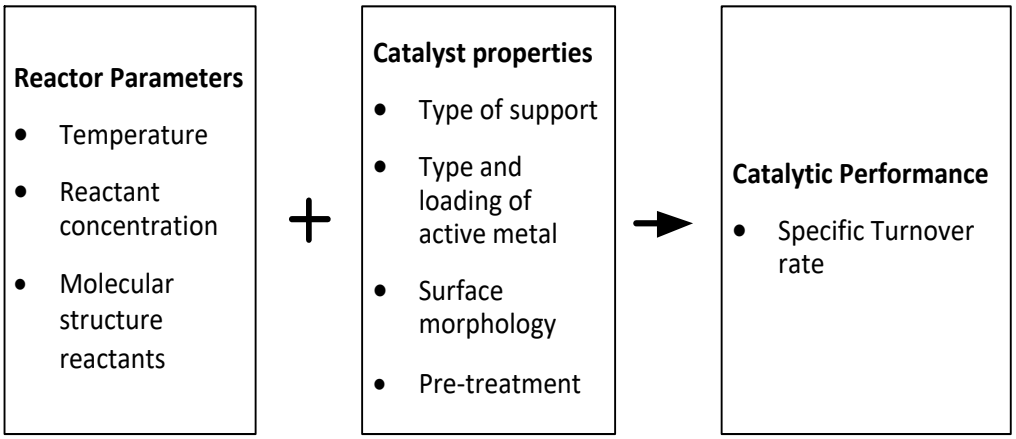


Figure 3.1. Catalytic Reactor model

3.1.2 Part A (Hydrogenolysis)

3.1.2.1 Experimental Data Utilized for ANN Training

Currently, ANN-based methodology were applied to model the n-alkanes (*n-ethane, n-propane, n-butane, n-pentane and n-hexane*) hydrogenolysis reaction over metal supported catalysts. These particular reactions were a consolidated data set by (Sormojai, 1994). The data taken from the literature entails, reaction conditions (temperature, molecular structure of reactants, reaction concentration), catalyst properties (type and loading of the active metal component, type of support, surface morphology (surface arrangement of metal atoms), pre-treatment) and output specific reaction turnover rate. The analogous experimental procedure details were given by the same author. The data utilized for the experiments is found in Appendix A. Since the ANN output signal falls between 0 and 1, all the data used was normalized between 0 and 1 before the training process. The normalization was performed by dividing all the values by the maximum value.

3.1.2.2 ANN Methodology for Catalysis Predictive Tool

ANNs were applied for the prediction catalytic performance in processes of hydrogenolysis of alkanes. The indicator of catalytic performance was specific turnover rate. The reaction conditions were reactant molecule concentration in molecules.cm^{-3}; temperature in Kelvins and the predicted variable is the specific turnover rate in molecules.cm^{-2}.sec^{-1}. Forty hydrogenolysis reaction sets were utilized for training: ethane (C_2), propane (C_3) and n-pentane (n-C_5). Both

hydrogenolysis reaction of n-butane (C_4) and n-hexane (C_5) we utilized for validation. Pertaining the molecular structure of reactants as input an integer value was assigned to each of the molecular moieties (0 or 1). The input would be an assigned integer multiplied by the number of the specific group divided by the total number of groups present. For example methyl group in n-propane, the input factor will be as follows. $1 \times 2 \div 3 = 0.667$, shown in Figure 3.2.

Shown below is a hydrogenolysis reaction of n-propane.

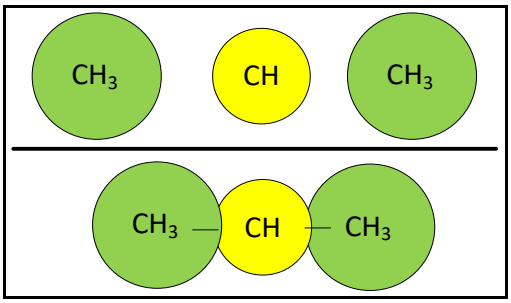


Figure 3.2. The molecular structure of propane.

$$C_3H_8 + H_2 \longrightarrow C_2H_6 + CH_4$$

The reactants are n-propane and hydrogen and the resulting products after catalytic hydrogenolysis is ethane and methane. The training and testing of all the alkanes hydrogenolysis was obtained from literature by Somorjai, (1994). As forty sets of training data was used and five sets of hexane and butane data were utilized for validation.

3.1.3 Part B (hydrogenation)

3.1.3.1. ANN Methodology for Catalysis Predictive Tool

ANN was applied for the prediction specific turnover rate as a catalytic performance in processes of hydrogenation of alkanes. The reaction conditions were: reactant molecule concentration in molecules.cm^{-3}; temperature in Kelvins and the predicted variable is the specific turnover rate in molecules.cm^{-2}.sec^{-1}. The data of the reaction was divided into three categories namely; training (68), testing (14) and validation

set (6), shown in Appendix B. Eighty eight hydrogenation reaction sets were utilized for training: Ethylene, benzene, propene and cyclopentane. Both hydrogenation reaction of benzene and ethylene we utilized for testing and only ethylene reaction were utilized for validation. The molecular structure as input was similar to the one of section 3.1.1 part A for hydrogenolysis. The training and testing of all the organic hydrogenation was obtained from literature by Somrjai, (1994). As 88 sets of training data was used and 22 sets of data were utilized for testing and validation.

3.2 Case Study 2

The research was conducted on a soap making factory. ANN model is a black-box approach, focusing on inputs and outputs, therefore it is very crucial to identify the significant inputs and outputs of the entire process.

3.2.1 Aims of the Work.

The aim was to show that the ANN could be used to define the sensitivity of the process performance to various process parameters, both continuous and intangible (such as human factors). The inputs of this process are; Quantities, and mass flowrate of individual phases, type of variant and time durations for different stages involved in the process. The outputs are; cycle time, physical characteristics of the product (viscosity, pH and specific gravity) and material usage efficiency. The measure of performance would be the production total cycle time (hours), and physical qualities of the product. An ANN model would be developed. A final ANN model would enable the user to simulate, visualize and gain a proper understanding of the process, hence be able to optimize base on influential factors (i.e. inputs and outputs of the process).

Modelling and simulation is a prerequisite to optimizing operation of the process hence it can be utilized for a improved considerate of the process operation. ANN is established empirically ccompared to process mass and energy, as noted by Himmelblau (2000). ANNs are useful for the real time simulation mainly when there is lack of data about the dynamics fo the system.

3.2.2 Compilation and Preparation of Data

The data was collected at the Unilever Maydon Wharf factory. Batch processes are common in food, specialty chemicals and pharmaceuticals manufacturing factories. The basic goals of batch process optimization are to maximize utilization of equipment on the plant and to minimize the batch cycle time. (ANNs) are authoritative modelling tools that can capture trends in historical plant data and use the developed model to calculate the performance of the plant for variations in the plant structure, schedule or operating parameters.

Since Unilever is one of the leading player in FMCG (fast moving and consumable goods) it was point of interest for the study. The first meeting was held between all stakeholders i.e. Unilever employees and UKZN representatives. Engineers representing Unilever at the first meeting were; James Conridie, Shivania Naicker and Ukash and those representing UKZN were the interns, Doctor David Lokhat and myself. The meeting entailed the presentation and proposal from our side and then collectively identifying a suitable process for our project.

Ukash and Nkululeko were the engineers that were available from Unilever to assist us with process study and data collection. There was an induction done, which specifically explained what was expected from us, for example safety issues, rule and regulations. The major limitation was that most of the documents needed could not be easily accessed due to company policies and procedures, hence there were delays pertaining documentation access. Therefore this negatively impacted the process of data collection to a point that the superior ordered the intern to stop collecting data.

The data collecting team or interns comprised of third year and fourth year UKZN chemical engineering students. They all have been previously exposed to chemical engineering factory or industrial environments before the Unilever project. The team had the following qualities: ability to analyze data, cognitive thinking, and engineering acumen. The lack of experience in data collection and limitations imposed by the company procedures were the major challenges they encountered. It was difficult to acquire the sonolation batch process historical data with all the crucial inputs for instance batch cycle time. Some interns had challenges of attaining

necessary data from some of the operators, possible due to communication challenges.

The sonolation process under personal care production was selected. The sonolation process shown in Figure 3.4 utilize a sonolator. The Sonolator in Figure 3.3 is a high shear static mixer used by manufactures to mix or emulsify multiphase fluids by compelling them through a nozzle with a small orifice at high pressure (Ryan, 2015). The high velocity jet formed encroaches on a secure blade which separates the flow and is believed to cause cavitation. The pressure in the core compartment is maintained by backpressure valve downstream.

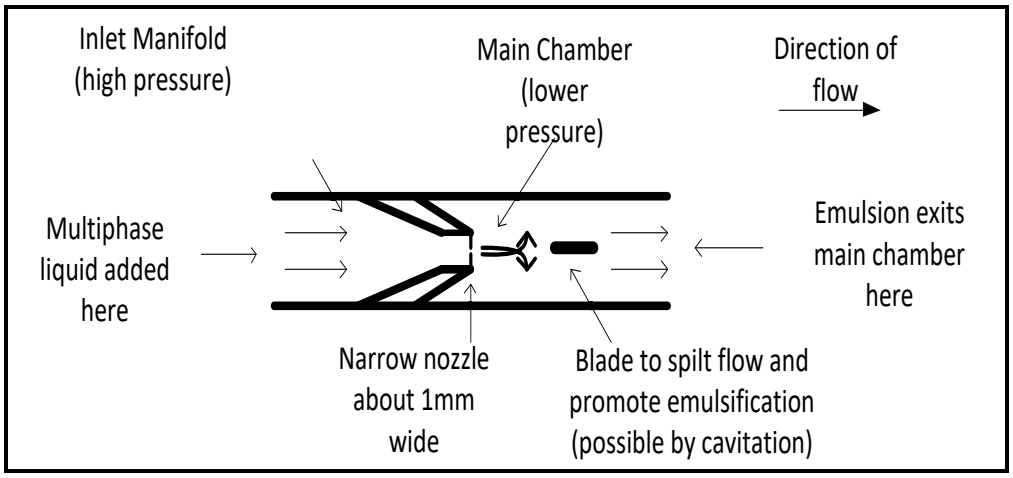


Figure 3.3 Sonolator device (Ryan, 2015)

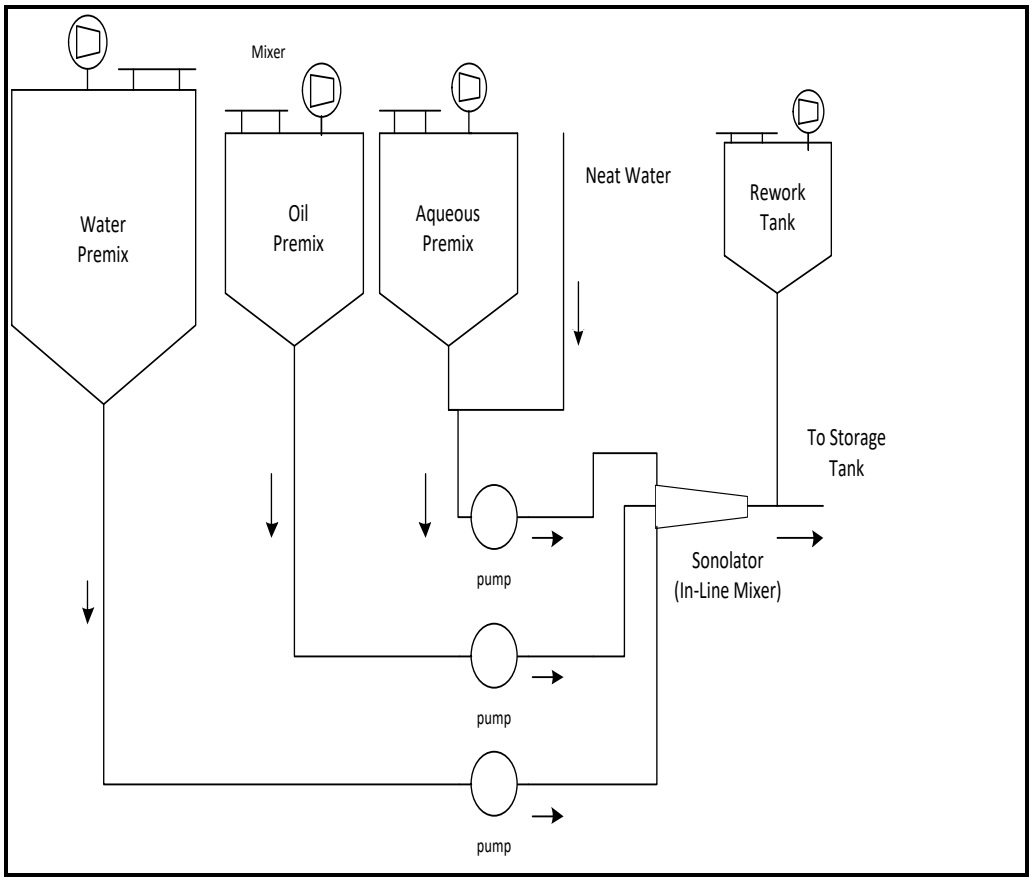


Figure 3.4 Sonolation Process

The objective for data gathering is to capture quality confirmation that allows investigation that result to the formulation of undoubted and trustworthy solutions to the problems at hand. Penalties from incorrectly composed data comprise: Incapability to solve the research problems exactly; Incapability to authenticate and repeat the study.

The following steps were followed when collecting the data: Process study and analysis, identification of the significant inputs, outputs and critical steps and parameters in the process. The observation method was used to collect data, it involves systematically selecting, watching and recording behaviour and characteristics of a process and humans.

Advantages:

- Collect information when and where an occasion or action is happening.
- Does not depend on people's preparedness to deliver data.

Disadvantages:

- Vulnerable to viewer biasness.
- Hawthorne effect – individuals commonly act well when they know they are being perceived.
- Does not intensify considerate of why individuals act in certain way.

The challenges encountered when the data was collected:

(i) **Ethnical issues regarding confidentiality and privacy**. This was the main setback as it delayed the start of data collection by 7 months and even when the data collection process was in progress it had to end earlier than the scheduled dates due to these issues raise by other staff members.

(ii) **Uncooperative Staff**. Some of the research assistance experienced lack of help of the staff, which off has negative effect on the data collected.

(iii) **Lack of experience**. The element of lack of experience negatively impacted data collection abilities.

Only 9 sample data sets were attained due to the regulations and logistical issues of the Factory, see Appendix B Table B.3.1 to B.3.5. The training data and testing data samples were randomly selected. Thus, circumventing either repetation or shortage of appropriate data points for training.

Normalization method before presenting the input data to the network is mostly a decent drill. Mixing variables with large amounts and small amounts blur the learning algorithm on the prominence of a individual variable and may drive it to finally discard the variable with the lesser amount (Tymvios et al., 2008). To ensure that the selected data for modeling distinguish distinctive operational varieties and off data set will be separated from the list. The data set will be grouped in two i.e. training, and testing data sets.

3.2.3 Design of an ANN Architectural Structure

This process is strongly dependent on the data used for training. Therefore the ANN structure will be examined and altered based on desired performance.

3.2.4 Training of an ANN

It is an optimization procedure where an error function is decreased by regulating ANN weights. In the training period, a set of recognized input-output configurations are continually offered to the network in order to demonstrate the network. The weights are constantly adjusted in order to decrease the error. The system learns how to handle different situations so that the network model can appropriately handle the similar scenarios in the future. Thus error is deduced from comparing output of an ANN with the actual output provided by the user. The training is to be conducted until error is reduced and better performance of ANN is achieved. A both stable and convergent ANN is desired from a training phase

3.2.5 Testing of ANN

A different data set will be utilized to validate the performance of ANN after the learning or training process. The sum of squared error (SSE), root mean square error (RMSE) and mean absolute deviation (MAD) of the network prediction would be calculated to find the best model. Therefore, the supreme valuable method is to firstly train with limited neurons and to continually train new networks with more neurons till the outcome meaningfully progresses and, predominantly, the simplification performance is not exaggerated (bottom-up approach), Kriesel 2005.

3.2.6 ANN Software

The software used was Visual Gene Developer 1.7 which is primarly used for gene development and testing. It has the ANN toolbox that allows for architecture generation, training, validation and prediction, which interfaces with Microsoft excel. It allos the user to select from various transfer fuctions (gaussian, sigmoid, hyperbolic tangent, modified sigmoid and modified gaussin), adjust the momentum coefficient and learning rate. Since the neuron output signal falls between 0 and 1, all the data need to be normalized between 0 and 1 before the training process.

CHAPTER 4: RESULT

4.1 Case Study 1 (Hydrogenolysis)

4.1.1 Inputs to the ANN Model.

The success of the ANN models greatly rely in appropriate selection of all inputs entering the input layer, as the wrong selection may result to an unsuccessful ANN model. The following were selected as inputs; the performance of a catalytic reaction depended on different reaction conditions (temperature, molecular structure of reactants, reaction concentration), catalyst properties (type and loading of the active metal component, type of support, surface morphology (surface arrangement of metal atoms), pre-treatment). Henceforth all the previously mentioned were selected as inputs into the ANN model. The choice of 17 inputs was made and it does not permit for many hidden neurons that may reduce the network's predictive proficiency. The development and simulation of ANNs was carried out using the visual gene development 1.7 software.

4.1.2 Training and Validation of ANN.

The training input data of 17x45 is given in Appendix A. The data was divided into two segments, the first part being the training set and the remaining segment was for testing of the network model. This is the stage where the model has to be monitored when training to avoid overtraining. Since the neuron output signal falls among 0 and 1, all the data used was normalized between 0 and 1 before the training process by dividing the values by the maximum value. The learning advancement of the present ANNs was appropriately supervised by the use of Visual Gene Developer 1.7 observing competences that permit the visual display of the error pasts of both the test and training cases.

4.1.3 Trained ANN

The suitable optimal training algorithm and neural network topology, was determined from literature and the data available see Figure 4.1. ANN optimal topology had an input layer of 17 nodes, 12 nodes in the in the primary hidden layer, 9 nodes in the subsequent hidden layer and 1 in the output layer. The hyperbolic tangent functions was a preferred transfer function in the hidden nodes in hidden

layers. ANN was first trained and then tested or validated with diverse sets of ethane *(C_2), propane (C_3), hexane (C_6), butane (C_4) and n-pentane (n-C_5)* samples and it delivered lesser error for the test samples when trained with 45 samples and it was evident that the neural network can create adequate predictions from 45 training samples. The training data set was separated into two parts the 36 data set for learning (80 percent of the data set) and 9 data set for validation, which were randomly selected from the data set and is 20 % of the data set. The validation set was used without the output variable in order to verify that the ANN model has learnt well from the training set. The inouts for the validation set were used to predict outputs using the trained ANN model and that these outputs were compared to the known outputs of the validation set in order to verify that the learning of the ANN model was satisfactory. Additionally, the use of the hyperbolic tangent function reduces the number of training samples required compared to the use of other functions. From the outcomes shown in Figure 4.1, 4.2 and Table 4.1 the trained ANN can predict the specific catalytic rate with minimal errors.

The input to the ANN model were; temperature (temp.) molecular structure of reactants (methyl; CH_3 and Methylene CH_2), reaction concentration (Hydrogen, H_2 and hydrocarbon), catalyst properties; type and loading of the active metal component (transition metals: Fe, Ir, Os, Re, Pd, Rh, Ru, Cu, Ni and Co, type of support (Al_2O_3 and SiO_2), surface morphology; surface arrangement of metal atoms) and output (specific turnover rate) for a specific reactions, see Figure 4.1. Accordingly, a multi-layered feed-forward back-propagated with a hyperbolic transfer function were designed. Four variables explicitly pressure, the feed inlet temperature, the feed flow rate, the concentration of feed salt aqueous solution and as input parameters accessible to the ANN model. AN optimum ANN model had 17 node input layer, two hidden layers (12 and 9 node respectively) and the output layer had 1 node, shown in Figure 4.1.

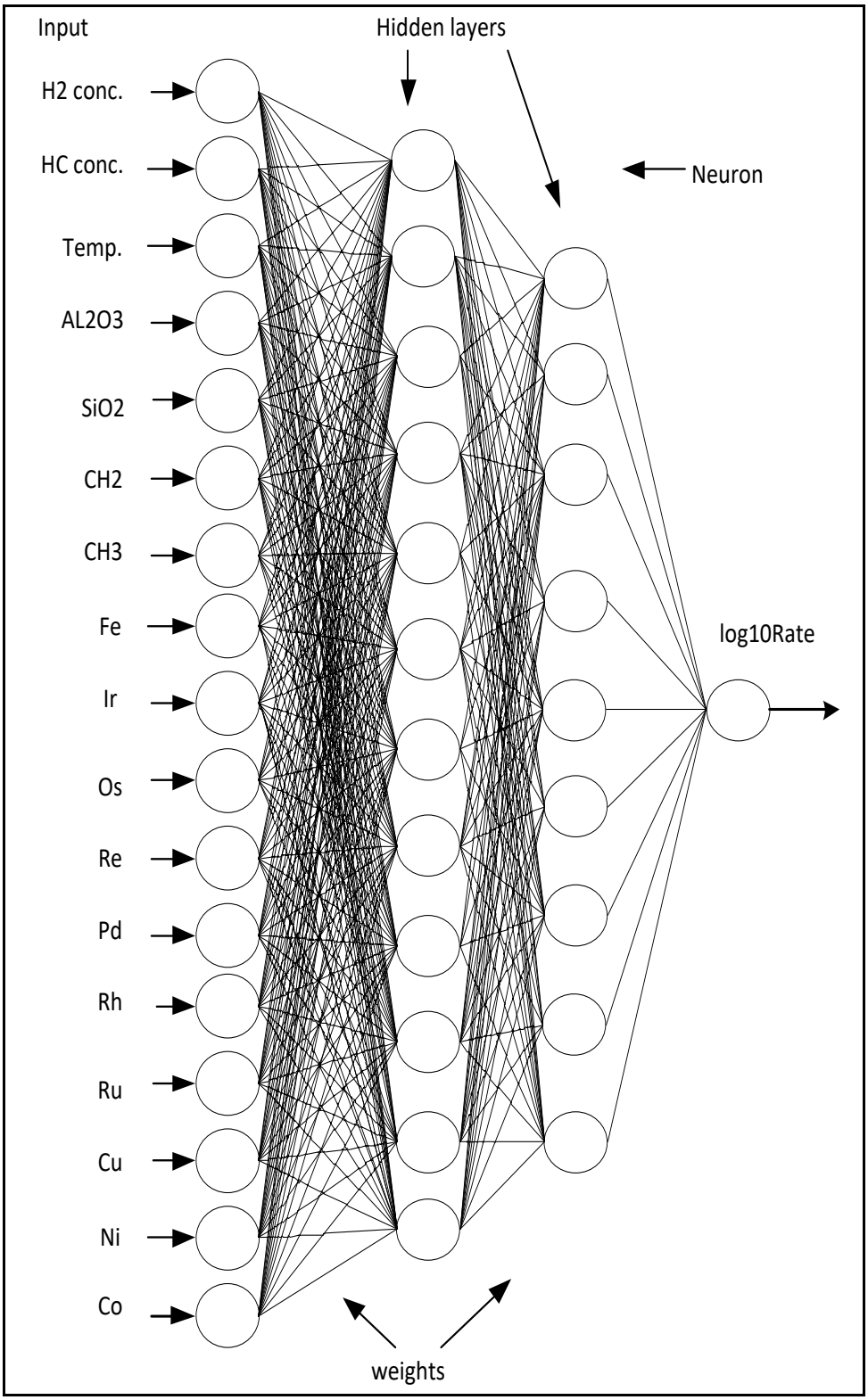


Figure 4.1 Optimized Artificial neural network topology (Hydrogenolysis)

All architectures in Table 4.1have 18 neurons input and 1 neuron output. Four multilayered feed-forward backpropagation ANNs were constructed, the configuration of each ANN is given in Table 4.1, namely; quantity of layers, quantity of hidden layers, number of neurons per layer, learning rate, R^2, root mean square error (RMSE), mean absolute deviation (MAD) and sum of squared error (SSE). The network chosen was the one with the smallest SSE (0.0897), RMSE (0.0499), MAD (0.0343) and configuration 18-12-9-1 shown in figure 4.1.

Table 4.1 Hydrogenolysis results. (The optimal ANN highlighted in blue)

ANN	# 1	# 2	# 3	# 4
Network	17-8-5-1	17-12-9-1	17-15-5-1	17-4-5-1
No. of layers	4	4	4	4
No. of hidden nodes	2	2	2	2
No. of neurons 1st hidden layer	8	12	12	4
No. of neurons 2nd hidden layer	5	9	9	5
Learning rate	0.004	0.002	0.002	0.002
R^2	0.7665	0.8024	0.7541	0.7541
SSE Training	0.1056	0.0897	0.0941	0.0995
SSE Test	0.0052	0.0056	0.0055	0.0049
RMSE	0.0541	0.0499	0.05112	0.0526
MAD	0.04	0.0343	0.0391	0.0405

Figure 4.2 illustrate a plot presenting ANN # 2 which was the superlative neural model the contrast between the predicted data set and training data set using ANN #2 from table 4.2 for turnover rate, both training data set (blue dots) and verification (prediction) data set (black dots) were considered. The plot in figure 4.2 divulges that the popular of the points fall close to the regression line. The results corroborates the precision and excellence of the neural model established. A feed forward ANN with back propagation training function was developed one and hidden layers.

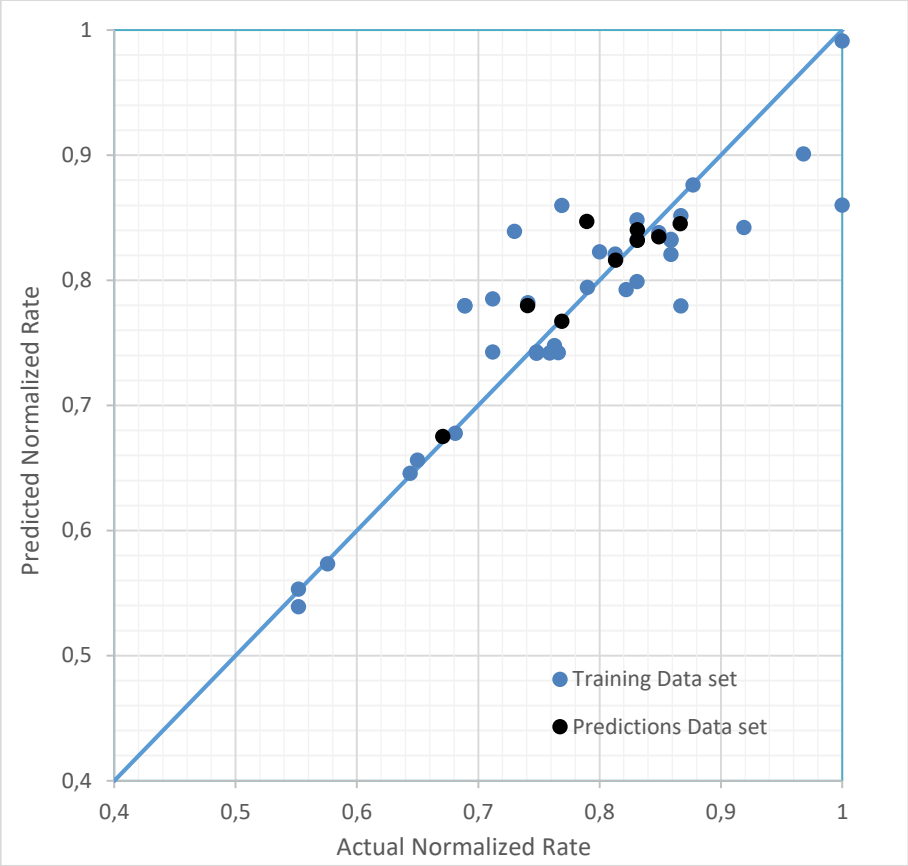


Figure 4.2. Graph of normalized actual and predicted specific hydrogenolysis catalytic reaction rates.

4.2 Part B (Hydrogenation)

4.2.1 Inputs to the ANN Model .

The selected inputs for hydrogenation reactions, that enhances the catalytic reactor performance are; (temperature, molecular structure of reactants, reaction concentration), catalyst properties (type and loading of the active metal component, type of support, surface morphology (surface arrangement of metal atoms), pre-treatment). Hence all the previously mentioned were selected as inputs. The choice of 22 inputs are reasonable, as it does not permit for many hidden neurons that may reduce the network's predictive ability. The visual gene development 1.7 software was utilized for development and simulation of ANNs. Thus, two multi-layered feed-forward back-propagated.

4.2.2 Training and Validation of ANN

The accessible data was divided into two segments, the first part being the training set (69) and the remaining segment was for testing or validation of the neural network model (13). This is the stage where the model has to be monitored when training to avoid overtraining, the learning and validation data set were 84% and 16 % of quantity correspondingly which is well within the range endorsed by literature. Successively the neuron output signal drops between 0 and 1, all the data used was normalized between 0 and 1 before the training process. The learning progress of the current ANNs was appropriately supervised by the use of Visual Gene Developer 1.7 observing competences that permit the visual display of the error histories of both the training and assess cases.

4.2.3 Trained ANN

The suitable optimal training algorithm and neural network topology, was determined from literature and the data available see Appendix B. The optimal ANN architecture had an input layer of 22 nodes, 10 nodes in the in the first hidden layer, 5 nodes in the second hidden layer and 1 in the output layer and the hyperbolic tangent functions in the hidden nodes in hidden layers. ANN was trained and validated with different sets of ethylene, benzene samples and it delivered lesser error for the test samples when trained with 68 samples and it was evident that the neural network can make adequate predictions from 68 training samples. Additionally, the use of the hyperbolic tangent function reduces the number of training samples required compared to the use of other functions. From the results shown in Figure 4.3, 4.4 and Table 4.2 the trained ANN can predict the specific catalytic rate with minimal errors.

The input to the ANN model are; temperature (temp.) molecular structure of reactants (methyl; CH_3, Methanetriyl CH and Methylene CH_2), reaction concentration (Hydrogen, H_2 and hydrocarbon), catalyst properties; type and loading of the active metal component (transition metals: Fe, Ir, Os, Re, Pd, Rh, Ru, Cu, Ni, Tn, Sn, Au, Pt and Co, type of support (Al_2O_3 and SiO_2), surface morphology; surface arrangement of metal atoms) and output (specific turnover rate) for a specific reactions, see Figure 4.3.

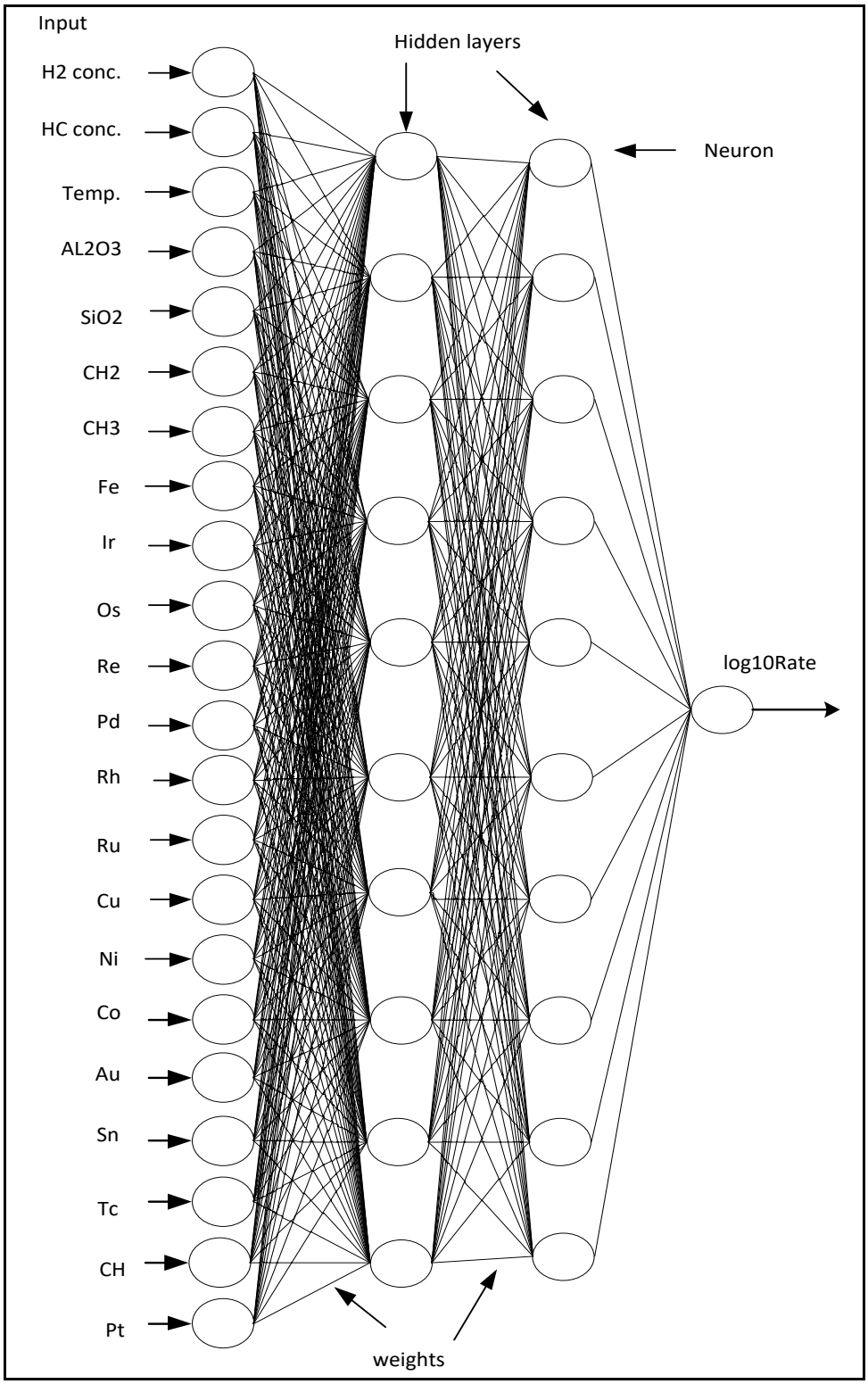


Figure 4.3. Optimized Artificial neural network topology (hydrogenation)

Four multilayer feed-forward backpropagation ANNs were constructed, the configuration of each ANN is given in Table 4.2, namely; number of layers, number of neurons per layer, number of hidden layers, learning rate, R^2, mean absolute deviation (MAD), root mean square error (RMSE) and sum of squared error (SSE). The network chosen was the one with the smallest SSE both for training and test data, MAD and RMSE with configuration 22-10-10-1 shown in Figure 4.4.

Table 4.2 Hydrogenation results (the optimal ANN highlighted in blue)

ANN	# 1	# 2	# 3	# 4
Network	22-10-10-1	22-10-10-1	22-10-10-1	22-10-10-1
No. of layers	4	4	4	4
No. of hidden nodes	2	2	2	2
No. of neurons 1st hidden layer	10	10	10	10
No. of neurons 2nd hidden layer	10	10	10	10
Learning rate	0.009	0.007	0.005	0.005
R^2	0.7102	0.739	0.7426	0.7513
SSE Training	0.1383	0.1309	0.1231	0.1186
SSE Test	1.0785	1.0788	0.4966	0.3299
RMSE	0.0453	0.0457	0.0429	0.0421
MAD	0.0322	0.0234	0.0307	0.0299

As mentioned above ANN # 4 was the superlative neural model. Figure 4.4 indicate a plot presenting the comparison between the predicted data set and training data set using ANN #4 from Table 4.2 for turnover rate, both training data set (blue dots) and prediction data set (black dots) were considered. The plot in figure 4.4 shows that the predicted data set are comparable to training data set. This definitely authenticates the exactness and quality of the neural model established.

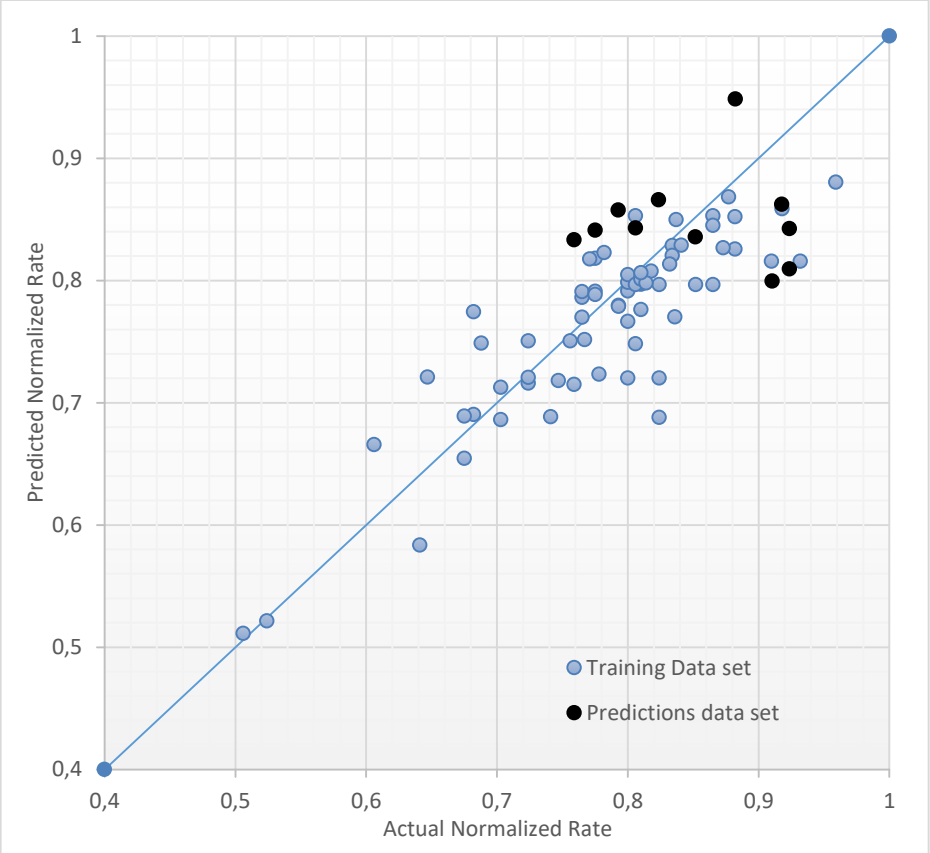


Figure 4.4. Graph of normalized actual and predicted specific catalytic hydrogenation reaction rate.

4.3 Case Study 2

4.3.1 Inputs to the ANN Model

The accomplishment of the ANN models really rely in general, appropriate selection of all inputs entering the input layer, as the wrong selection may consequence to an unsuccessful ANN model. The inputs to the process are; quantities, and mass flowrate of different phases, type of variant and time durations for different stages in the process. The outputs are; cycle time, viscosity, pH and specific gravity and material utilized efficiency. Hence all the previously mentioned were selected as inputs. The choice of 23 inputs was made and 9 data set available was separated into 5 training set and the remaining 4 was used for validation. The development and simulation of ANNs was carried out using the visual gene development 1.7 software.

4.3.2 Training and Validation of ANN

All the above mentioned trained and tested using the 28x9 input data given in Appendix B Table B.3.1 to B.3.5. The accessible data was divided into two segments, the first part being the training set and the remaining segment was for assessing or validation of the neural network model. This is the stage where the model has to be monitored when training to avoid overtraining. Since the neuron output signal falls between 0 and 1, all the data used was normalized between 0 and 1 before the training process. The learning process of the current ANNs was conveniently supervised by the use of Visual Gene Developer 1.7 monitoring capabilities that allow the visual readout of the error histories of both the training and test cases.

4.3.3 Trained ANN

An optimum training algorithm and neural network topology, was determined from the data available Appendix B Table B.3.1 to B.3.5. An optimal ANN had an input layer of 23 nodes, 5 nodes in the in the first hidden layer, 15 nodes in the second hidden layer, 7 nodes in the third layer and 5 in the output layer and the hyperbolic tangent functions in the hidden nodes in hidden layers. ANN was trained and validated with diverse sets of Vaseline and dawn samples and it delivered lower sum of squared error for the test samples when trained with five samples and it was evident that the ANN can make adequate predictions from 40 training samples. Additionally, the use of the hyperbolic tangent function reduces the number of training samples required compared to the use of other functions. From the results shown in Figure 4.5.1 to 4.5.5 and Table 4.3 the trained ANN can predict the required output with minimal errors.

The input to the ANN model are; type of variant, aqueous phase (min) (addition of liquid ingredients (min), addition of dry ingredients (min), addition of perfume (min) and transfer of premix (min)), oil phase (min) (addition of Pj (min), addition liquid ingredients (min) and time to raise temperature (min)), (premix 1 (min) , addition of dry ingredients (min) , addition of liquid (min) and flushing (min)), time before sonolation (min), sonolation time (min), aqueous phase into the sonolator (kg), aqueous phase out of the sonolator (kg), oil phase into the sonolator (kg), water phase

into the sonolator (kg), water phase into the sonolator (kg) and batch size (kg) and output are; total cycle time, pH, viscosity, specific gravity and material used efficiency, see Figure 4.5.

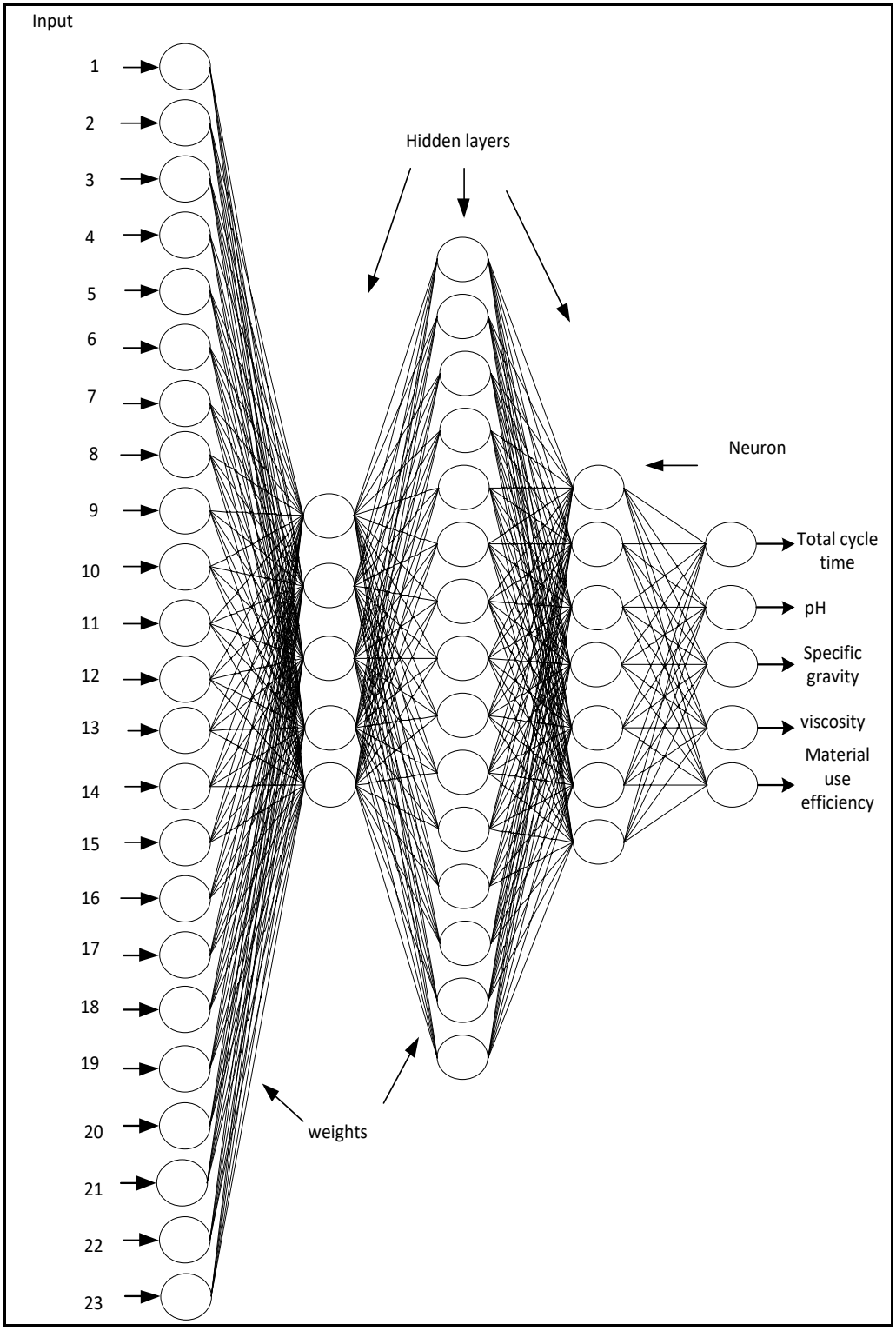


Figure 4.5 Optimized Artificial neural network topology (Batch process)

Therefore, a multi-layered feed-forward back-propagated with a hyperbolic transfer function were designed. All of above architectures have 23 neurons input and 5 neurons output. Four multilayer feed-forward backpropagation ANNs were created, the configuration of each ANN is given in table 4.3, namely; number of neurons per layer, number of layers, number of hidden layers, learning rate, R^2, root mean square error (RMSE), mean absolute deviation (MAD) and sum of squared error (SSE). The network chosen was the one with the smallest SSE (0.0083), MAD (0.0112) and RMSE (0.0182) with configuration 23-5-15-7-5 shown in Figure 4.5.

Table 4.3 Batch process results (the optimal ANN highlighted in blue)

ANN	# 1	# 2	# 3	# 4
Neuron configuration	23-10-15-7-5	23-5-10-7-5	23-5-15-7-5	23-5-15-7-5
No. of inputs	23	23	23	23
No. of outputs	5	5	5	5
No. of hidden layers	3	3	3	3
No. of neurons 1st layer	10	5	5	5
No. of neurons 2nd layer	15	10	15	11
No. of neurons 3rd layer	7	7	7	7
Transfer function	Hyperbolic	Hyperbolic	Hyperbolic	Hyperbolic
Learning rate	0.01	0.007	0.01	0.008
SSE Training	0.0155	0.0264	0.127	0.0083
SSE Test	0.3284	0.3724	0.1207	0.0002
RMSE	0.0249	0.0325	0.0712	0.0182
MAD	0.0179	0.0206	0.041	0.0112

Table 4.4 Batch process optimal ANN

ANN	**23-5-15-7-5**				
Graph	Total Cycle Time	pH	Specific Gravity	Viscosity	Material Used Efficiency
SSE	0.0692	0.0118	0.0002	0.1161	0.0014
RMSE	0.1315	0.0542	0.0073	0.1704	0.0184

Table 4.4 illustrate SEE and RMSE of the graphs in Figure 4.5.1 to Figure 4.5.5.

Figure 4.5.1 show a plot presenting the comparison between the predicted data set and training data set using ANN #4 from Table 4.3 for total cycle time, both training (blue dots) and verification (maroon dots) data set were considered. It is evident from the plot that the results are comparable with the majority of the points decreasing within minimal deviation with the SSE of targeted values of 0.0692. These result also indicate that with more training data samples the results will definitely improve.

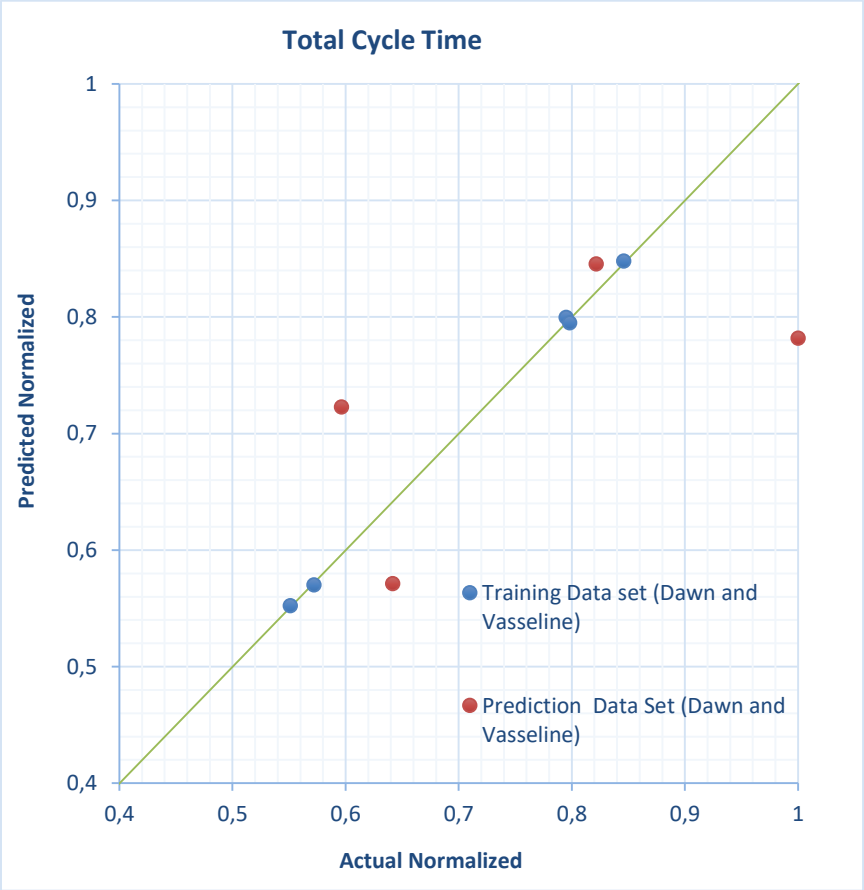


Figure 4.5.1 Graph of actual and predicted normalized total cycle time

Figure 4.5.2 below indicates a plot presenting the comparison between the predicted data set and training data set using ANN #4 from table 4.3 for pH, both training data set (blue dots) and prediction data set (maroon dots) were considered. There was not a great variability in the pH over data set considered and the results are within a minimal deviation with the SSE of targeted values of 0.0118. This substantiates the accuracy and quality of the neural model established.

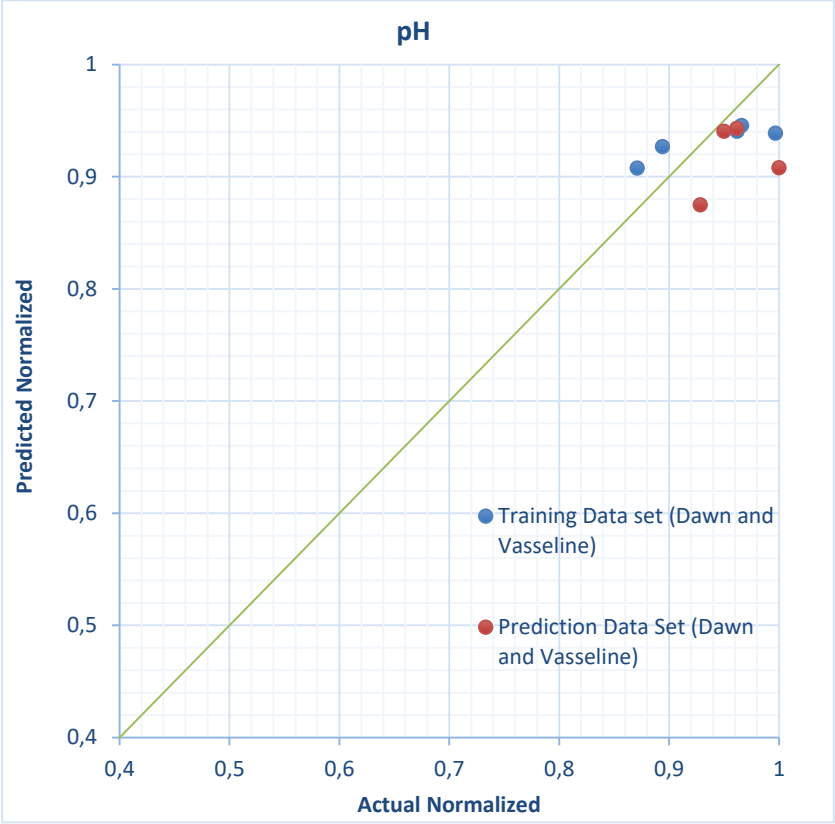


Figure 4.5.2 Graph of actual and predicted normalized pH

Figure 4.5.3 demonstrate a plot presenting the comparison between the predicted data set and training data set using ANN #4 from Table 4.3 for specific gravity, both training data set (blue dots) and prediction data set (maroon dots) were considered. There was less variability in the data set considered hence the results are comparable with all points falling within minimal deviation with the SSE value of 0.0002.

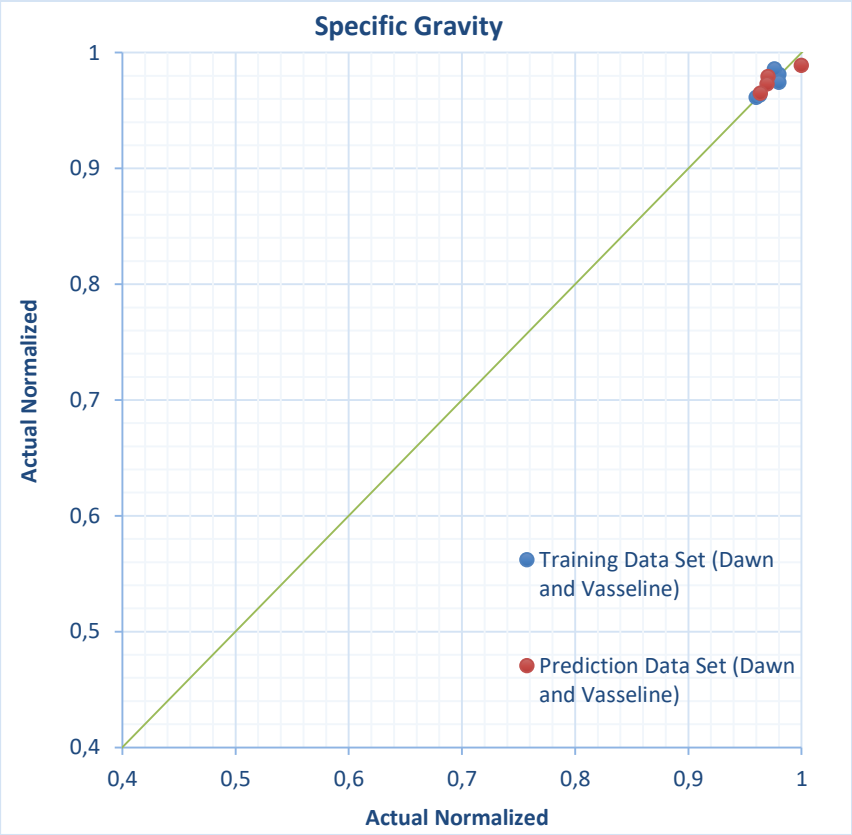


Figure 4.5.3 Graph of actual and predicted normalized specific gravity

Figure 4.5.4 illustrate a plot presenting the comparison between the predicted data set and training data set using ANN #4 from Table 4.3 for viscosity, both training data set (blue dots) and prediction data set (maroon dots) were considered. It is evident from the plot that the results are comparable with the majority of the points falling within minimal deviation and the SSE of targeted values of 0.1161. This definitely substantiates the accuracy and quality of the neural model developed. There is also an outlier and this may due to the great variability in the training data set.

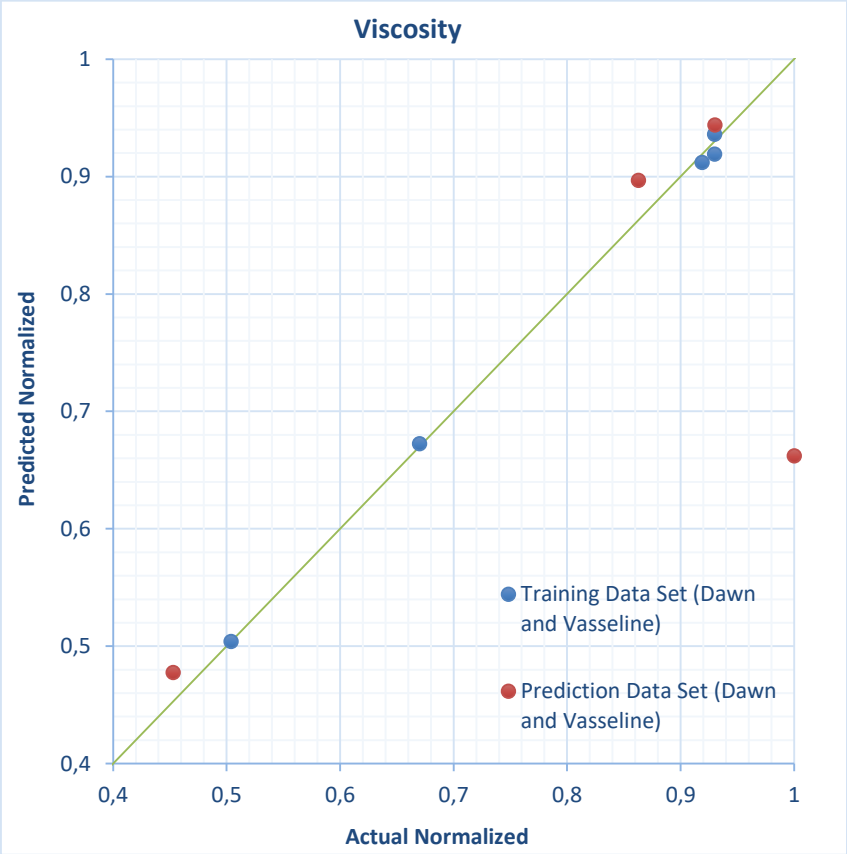


Figure 4.5.4 Graph of actual and predicted normalized viscosity

Figure 4.5.5 illustrate a plot presenting the comparison between the predicted data set and training data set using ANN #4 from Table 4.3 for material utilized efficiency, both training data set (blue dots) and prediction data set (maroon dots) were considered. It is evident from the plot that there was not much variability in the parameter over the data sets considered and the results are comparable with the majority of the points falling within minimal deviation and the SSE of targeted values of 0.0014. This validates the accuracy and quality of the neural model developed. The plot also indicate that the process was good from the operability point of view.

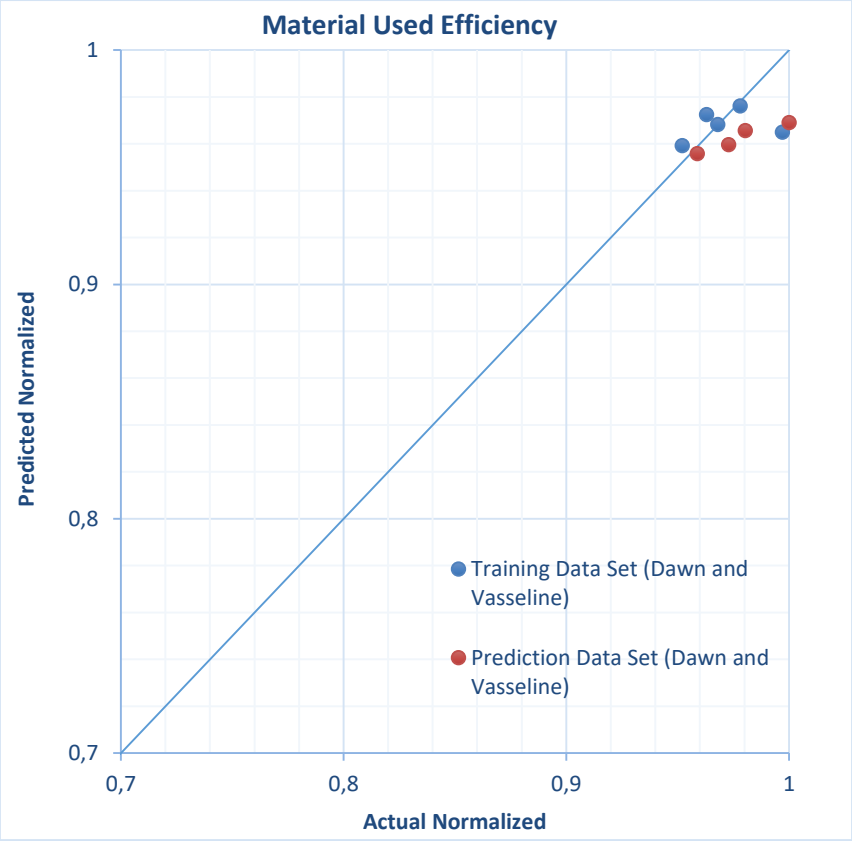


Figure 4.5.5 Graph of actual and predicted normalized material utilized efficiency

4.3.4 Sensitivity

Sensitivity analysis was conducted to understand the relationships between input and output variables. Sensitivity analysis is basically a method used to determine how different values of an independent variable impact a particular dependent variable. The sensitivity analysis assist in identifying the key variable that have a major influence in the production, cost and benefit of the process. The graphs in Appendix C illustrate the sensitivity test that were done to determine the degree of influence of inputs on the outputs. The inputs values were adjusted 0.5 and 1 in order to evaluate the sensitivity of the output to change in inputs.

The properties such as specific gravity, pH, and material used efficiency are some of the outputs to the batch process investigated but were used as an input to the model during ANN training. Even though there is small variability in the input values for these variables, they are still very important in characterizing the final product of the sonolation process and for quality control purposes. Hence, they were retained in the ANN model.

CHAPTER 5: DISCUSSION

The Visual Gene Developer 1.7 software was used to appropriately supervise the learning progress of the presented ANN models, monitoring proficiencies that permit the visual display of the error histories of both the training and validation cases. The Visual Gene Developer 1.7 software expressed these errors as sum of squared errors (SSE) between the target outcome and the network's outcome. The learning progress also the process of divergence or convergence are well monitored by the sum of squared error, the network reaches convergence for as long as SSE approaches the minimum value. The overtraining of the network is at the point when the error starts cumulating after having extended its global minimum and past this point the network is memorizing. Therefore for a worthy quality of outcome and decent ANN based model should only be trained to the point of the test set's global minimum error.

The use of a hyperbolic tangent function as the function in the hidden nodes ensured that the training data used was adequate as it reduces the number of training samples required compared to both the sigmoid and logistic functions. The number of hidden layers was within the limit as it was less than the maximum of four recommended by literature Mohd, (2003). The hyperbolic tangent function was chosen as a transfer function for nearly all the network model. Preceding the training procedure, all input and target data were normalized between 0 and 1, dividing all the values by maximum values since each neuron's output signal falls between 0 and 1.

5.1 Case Study 1

An optimized ANN model constructed based on the results of previous studies with similar reactants and catalysts, can be utilized to screen catalysts for hydrogenolysis and hydrogenation catalytic reactions. An optimized ANN model based also has a potential in development of the new catalysts to improve existing processes, replace existing processes and reduce environmental impact. The catalytic reaction turnover rate is mostly affected by the reaction conditions: reactant concentration, temperature, the absence or presence of the catalyst and catalyst properties. Hence all the inputs in this study had a significant effect to the sensitivity of turnover reaction rate.

5.1.1 Part A (Hydrogenolysis)

The ANN model was trained to predict the output which was the measure of the catalytic performance for this study, see Figure 4.2. Table 4.1 briefly illustrate the configuration of ANN models that were constructed and ANN # 2 (18-12-9-1) was the optimum model with the learning rate of 0.002, R^2 (0.802), SSE (0.0897), RMSE (0.0499) and MAD (0.0343) which indicate good quality predictions of ethane, propane, n-butane and n-hexane specific catalytic rate.

Figure 4.1 depicts the comparison of training and predicted data for the turnover rate in catalytic hydrogenolysis and the predictions are of good quality. Base on the results one can safely conclude that the input to the network were properly selected. Out of the data that was available 80% was utilized for training dedications and the remaining 20% was set aside and used to validate, authenticate the integrity of the neural network model and to verify possible memorizing or overtraining.

Most of the outliers in Firure 4.2 were from pentane and ethane alkanes that was due to different metal catalyst combinations . The turnover reaction rate was sensitive to the metal catalyst types and combinations. The ANN model developed can be useful in utilization to screen different alkane catalyt combinations before the actual reaction is perfomed.

5.1.2 Part B (Hydrogenation)

The ANN model designed and trained in Figure 4.4 indicates the prediction of the output (turnover reaction rate) from input values. Table 4.2 briefly illustrate the configuration of ANN models that were constructed and ANN # 4 (22-10-10-1) was the optimum model with the learning rate of 0.002, R^2 (0.7513), SSE (0.1156), RMSE (0.0421) and MAD (0.0299) which indicate good quality predictions. The transfer function chosen was hyperbolic tangent function for nearly all the network model.

Figure 4.3 depicts the comparison of training and predicted data for the turnover rate in catalytic hydrogenolysis and the predictions are of good quality. Base on the results one can safely conclude that the input to the network were properly selected. Out of the data that was available 84% was used for training purposes and the

remaining 16% was set aside and used to validate, authenticate the integrity of the neural network model and to verify possible memorizing or overtraining.

The values that had a noticeable variation of turnover rate in Figure 4.4 were from benzene and ethylene that was due to that the input to an ANN model had different metal catalyst combinations. The reaction turnover rate was most sensitive to change in catalyst. Therefore the optimized ANN model can be utilized for screening different catalyst combinations.

5.2 Case Study 2

The ANN model was designed and trained, the results are presented in Figure 4.3.1 to 4.3.5. The hyperbolic tangent function was chosen as a transfer function for the network model. ANN model were tested, varying the learning rate and the optimum was selected. The sum of squared error for the optimum model between the output and the normalized predicted output was 0.0083 see Table 4.3 for an ANN model, which indicate good quality predictions of the process.

Table 4.3 briefly illustrate the configuration of ANN models that were constructed and ANN # 4 (23-5-10-7-5) was the optimum model with the learning rate of 0.008 which is well within the literature recommended values and the sum of squared error of this model was 0.0083 which was the smallest and had good quality results. Figure 4.3.1 to 4.3.5 depicts the comparison of training and predicted data for the turnover rate in catalytic hydrogenolysis and the predictions are of good quality.

Base on the results one can safely conclude that the input to the network were properly selected. Out of the data that was available 55.6 % was utilized for training determinations and the residual 44.4 % was set apart and used to validate, authenticate the integrity of the neural network model and to verify possible memorizing or overtraining. Since each neuron's output signal falls between 0 and 1, all input and target data were normalized between 0 and 1, preceding the training process.

5.2.1 Sensitivity

The graphs in Appendix C illustrate the sensitivity tests that were performed to determine the degree of influence of inputs on the outputs of the process. The inputs values were from original normalized prediction and to the normalized prediction of 0.5 and 1 as inputs in order to evaluate the sensitivity of the output to change in inputs.

Material utilized efficiency (MUT) was most sensitive to all input and outputs streams of the process that was due to it on the material entering and leaving the process. The graphs in Figures C.11 to C.4 illustrate the sensitivity of MUT to inputs and outputs of the sonolation process. The MUT was least sensitive for all the time based inputs graphs are shown in Figure C.5 to sC.10. That was due that MUT is less dependent on time taken to add inputs or ingredients to the process than the actual amount of inputs added.

Total cycle time was most sensitive to sonolation time illustrated in Figure C.5 that was due to the direct dependence of total cycle time on the duration of sonolation, meaning the change in one results on the change on the other and due to that sonolation is the crux of the entire process. Basically all the time based inputs had a significant effect on the sensitivity of total cycle time, as they had direct effect on total cycle time, see graphs in Figures C.1 to C.10. Total cycle time was least sensitive to oil phase inlet shown in Figure C.4.

Specific gravity was most sensitive to changes in the amounts of inlet streams to the process compared to the time based inputs. The graphs depicting the inlet streams that had a notable effect on sensitivity of specific gravity are shown in Figures C.21 to C.30. Specific gravity was least sensitive to time taken to raise the temperature of the liquid phases shown in Figure C.30.

Viscosity was most sensitive to the changes in oil phase inlet and water phase inlet shown in Figure C.33 and C.34 respectively. This is caused by the basic principle that a change in the solvent concentration directly affects viscosity. The addition of water phase decreases viscosity, whilst addition of oil phase increases viscosity. The

other inputs had fairly negligible effect on the sensitivity of viscosity these are shown in Figures C.31 to C.40.

The pH was mostly invariant physical property as there was very minimal to no changes observed to it with the change in the most of the inputs to the process. The graphs are shown in Figure C.41 to C.50. This was caused by the fact that pH is affected by the change in the acidity and alkalinity of the solvent.

All physical properties were generally sensitive to material inputs, but it can be concluded that there was not much that can be done on the operability point of view. Material utilized efficiency was sensitive to all inputs but the change was small about if one notice the change in numbers shown in Figures C.11 to C20. Sonolation time proved to be the core of the process since the proper functioning of the sonolator and duration of sonolation determined the batch quality and total cycle time.

From the operability point of view total cycle time was to highly sensitive to changes sonolation time shown in Figure C.5 to C10. The redundancy of sonolator would eliminate any batch delay or loss, in way that if there were issues with one the other can be utilized. All the above observation were obtained by the use of an ANN.

CHAPTER 6: CONCLUSION AND RECOMMENDATIONS

6.1 Case Study 1 Part A: (Hydrogenolysis) and Part B: (Hydrogenation)

The optimal ANN architecture was chosen from three diverse multi-layer feed-forward ANN architectures that were designed and trained using literature data for hydrogenolysis. The hydrogenation process optimal ANN architecture was selected from four diverse multi-layer feed-forward ANN architectures were design and trained using literature data for hydrogenation.

The agreement between the comparison of predicted and targeted values was good thus indicating the capabilities of an optimal ANN as a predictive tool for analysis of heterogeneous catalytic reactions. ANNs are appropriate alternative to the mechanism based kinetics models used to model kinetics for screening large sets of potential catalyst-substrate combinations when developing a new catalyst for hydrogenolysis and hydrogenation of catalytic reactions.

The ANN model present an ideal black box model that simple relate the effect of input variables to the output one, with a tolerable and reasonable degree of error. Based on the results predicted ANN model the artificial neural network architecture had been designed with desired input and output information.

ANN and available surface rate data can be used to construct a beneficial tool for the analysis of solid catalyzed gas-phase reactions. The entire study confirmed the relevance utilization of ANN model in prediction of hydrogenolysis and hydrogenation kinetics for improvement, design and optimization of catalytic schemes in a more effective and prompt means. The work confirmed an appropriate method which may function as a esteemed tool to optimize and design innovative catalytic systems in a more proficient and prompt way.

ANN can be utilized to model properly the kinetics of diverse catalytic reactions under diverse reaction and catalyst conditions, being an alternative mechanism-based kinetics models. A more robust predictive tool can be formulated integrating larger data sets for different types of reactions, with the definitive goal of providing

predictions for any type of catalyst-substrate combination. The neural network model can be advanced by including surface structure effects and catalyst preparation and pre-treatment procedures.

The turnover reaction rate was sensitive to the metal catalyst types and combinations. The ANN model developed can be useful in utilization to screen different alkane catalyt combinations before the actual reaction is perfomed.

6.2 Case Study 2 (Batch process)

An optimal ANN architecture was derived from five different multi-layer feed-forward ANN architectures that were designed and trained using data collected from the plant. The agreement between the comparison of predicted and targeted values was greatly adequate thus indicating the capabilities of an optimum ANN. Artificial neural networks definitely work.

The ANN model present an ideal black box model that simple relate the effect of input variables to the output one, with a tolerable and reasonable degree of error. Based on the results predicted ANN model the artificial neural network architecture had been designed with desired input and output information. ANN and the data collected can be used to construct a beneficial tool for the analysis of batch process. A more robust predictive tool can be formulated integrating larger data sets for different types of variants, with the definitive goal of providing predictions for the process outputs. The neural network model can be advanced by including ingredients of the process.

The entire study confirmed the relevance utilization of ANN model in prediction of batch process outputs for improvement, design and optimization in a more effective and prompt means. There is an in-depth study opportunity in future and perhaps with more different variants and a lot of data set for each variant for a quite good study.

From the operability point of view total cycle time was to highly sensitive to changes sonolation time shown in Figure C.5 to C10. The redundancy of sonolator would

eliminate any batch delay or loss, in way that if there were issues with one the other can be utilized. All the above observation were obtained by the use of an ANN.

Lastly the work inveterate the appropriately of present ANN method which may ultimately function as a very valued tool to optimize the process. The modelling of batch process by ANN has been confirmed and the effect of experimental scattering in fitting data has been also predictable, allowing reasonable degrees of error.

From the operability point of view total cycle time was to highly sensitive to changes sonolation tim. The redundancy of sonolator would eliminate any batch delay or loss, in way that if there were issues with one the other can be utilized. All the above observation were obtained by the use of an ANN.

REFERENCES

Agirre-Basurko, E., Ibarra-Berastegi, G., Madariaga, I., (2006). Regression and Multilayer Perceptron-based Models to Forecast hourly O3 and NO2 Levels in the Area, Environmental Modelling & Software, Volume 21, Issue 4, pp 430-446.Last Accessed:04/08/2017
http://shodhganga.inflibnet.ac.in/bitstream/10603/48/6/chaper%204_c%20b%20bangal.pdf

Aguirre, L. A. (2007). An Introduction to System Identification – Linear and Nonlinear Techniques Applied to Real Systems, 3rd ed, UFMG, Minas Gerais, Brazil.

Alhamud, A. A., (2004). The Use of a Neural Network to Recognize Placental Insufficiency from Blood Flow Velocity Waveforms in the Umbilical Cord, Biomedical Engineering, University of Cape Town, Thesis.
http://open.uct.ac.za/handle/11427/7909

Alheeti, K. M., (2011). Intrusion Detection System and Artificial Intelligent, Alanbar University, Iraq. Last Accessed: 04/08/2017
http://cdn.intechopen.com/pdfs-wm/14360.pdf

Anderson, D., McNeill, G., (1990), Artificial Neural Networks Technology, Kaman Sciences Corporation, New York. Last Accessed: 04/08/2017
http://andrei.clubcisco.ro/cursuri/f/f-sym/5master/aac-nnga/AI_neural_nets.pdf

Arce-Medina, E., Paz-Paredes, J. I., (2009). Artificial Neural Network Modeling Technique Applied to the Hydrodesulfurization Porcess, Mathematical and Computer Modelling, Volume 49, Issues1-2, pp 207-214. Last Accessed: 04/08/2017
http://www.sciencedirect.com/science/article/pii/S0895717708001398?via%3Dihub

Arvindbhai, P. U., Optimization of Process Parameter in Mig Welding Process on Dissimillar Material by using Artificial Neural Network. http://gnu.inflibnet.ac.in/bitstream/123456789/2580/1/Ujjvalkumar%20A.%20Patel.pdf

Asgari, H., Chen X., and Sainudiin R. (2013). Analysis of ANN-Based Modelling Approach for Industrial Systems. International Journal of Innovation, Management and Technology, Vol. 4, No.1. http://ijimt.org/papers/383-K2003.pdf

Asgari, H., (2014). Modelling, Simulation and Control of Gas Turbines Using Artificial Neural Networks, Mechanical Engineering, University of Canterbury Christchurch, New Zealand, Thesis. Last Accessed: 04/08/2017 http://ir.canterbury.ac.nz/bitstream/handle/10092/9355/thesis_fulltext.pdf?isAllowed=y&sequence=1

Baghat, P., (1990). An introduction to neural nets, Chem. Eng. Prog. 55-61.

Baughman, F. (1995) Neural Networks in Bioprocessing and Chemical Engineering. Dissertation. 20-40.

Bourikas, L., (2016). Microclimate adapted localized weather data generation Implications for urban Modelling and Energy Consumption of buildings, Faculty of Engineering and the Environment Energy and Climate Change Division. Last Accessed: 04/08/2017 https://eprints.soton.ac.uk/412948/1/FINAL_e_thesis_for_e_prints_BOURIKAS_24500267.pdf

Chen, S., Cowan, C., and Grant, P. (1991). Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks, IEEE Transactions on Neural Networks, 2(2), 302-309.

Hernandez-Pichard, L. M., Macias-Salinas, R., (2017)Modeling the n-Hexane Isomerization over Iron Promoted Pt/WO_x-ZrO_2 Catalysts Using Artificial Neural Networks. Last Accessed: 04/08/2017
http://pubs.acs.org/doi/10.1021/acs.iecr.6b01821

Himmelblau, D.M., (2000). Applications of Artificial Neural Networks in Chemical Engineering, Korean Journal of Chemical Engineering, Volume 17, Issue 4, pp 373-392. Last Accessed: 04/08/2017
http://link.springer.com/article/10.1007%2FBF02706848

Hogan, W. D., (1974), Solid waste disposal in rural Arizona: Application of a least-cost model, The University of Arizona, Thesis. Last Accessed: 04/08/2017
http://arizona.openrepository.com/arizona/handle/10150/566408

Hu, S., (2014). Developing Domain-specific Simulation Objects for Modeling Clinical Laboratory Operations, Industrial and Systems Engineering, University of Washington, Thesis.

Ike, D. U., Anthony, A. U., (2013), Back-Propagation Artificial Neural Network Techniques for Optical Character Recognition- Survey, International Journal Computersand Distributed System. Covenant University, Ota. Last Accessed: 04/08/2017
http://eprints.covenantuniversity.edu.ng/4178/1/Adoghe%202.pdf

Ismail, D. B., Boyac, H., (2007). Modeling and Optimization II: Comparison of Estimation Capabilities of Response Surface Methodology with Artificial Neural Networks in a Biochemical Reaction, Journal of Food Engineering, Volume 78, Issue 3, pp 846-854.

Ioannis, Kampianakis, (2011). Credit Risk Modeling with use of Neural Networks and Genetic Algorithms. Derpatment of Production Engineering & Management, Technical University of Crete, Dissertation. Last Accessed: 04/08/2017
http://artemis.library.tuc.gr/MT2012-0062/MT2012-0062.pdf

Jain, A. K., Mao, Mohiuddin, K. M. (1996). Artificial Neural Networks: A Tutorial. IEEE Computer, 29, 31-44.

Kasar, M., Bhattacharyya, D., (2016). Face Recognition Using Neural Network: A Review, International Journal of Security and Its Applications, Vol. 10, No.3, pages 81-100. Last Accessed: 04/08/2017
www.sersc.org/journals/IJSIA/vol10_no3_2016/8.pdf

Kelly, M., (2004), Intelligent Manufacturing, United States. Last Accessed: 04/08/2017
https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5263/1/A-self-learning-machine-vision-system/10.1117/12.518547.short?SSO=1

Kriesel, David, (2005). A Brief Introduction to Neural Networks, University of Benin, Germany.
http://www.dkriesel.com/_media/science/neuronalenetze-en-zeta2-2col-dkrieselcom.pdf

Mohd, k., Idris A., Siong L.J., Wong H. M. and Morad, N.A. (2003). Artificial Neural Network Modelling of Steady State Chemical Engineering Systems. Artificial Intelligence Applications in Industry. Kuala Lumpur.

Olniyan, A. M., Owalabi, A. B., Applying Artificial Nueral Network Modeling for Predicting Postharvest Loss in sme Common Agrifood Commodities., Department of Agricultural and Bioresouces Engineering, Nigeria. Last Accessed: 04/08/2017
http://www.msc-les.org/proceedings/foodops/2016/FOODOPS2016_11.pdf

Ozkan, O., (2010). Group Technology and Cellular Manufacturing with artificial Neural networks, Graduate School of Natural and Applied Sciences, Dokuz Eylul University.

Ryan, J.D, (2015) Investigation of Fluid Dynamics and Emulsification in Sonolator Liquid Whistles, School of Chemical Engineering, University of Birmingham, Thesis.

Santos, R. C., Santos, B. F., Fileti, M. F., da Silva, F. V., Zemp, R. J., (2013). Application of Artificial Neural Networks in an Experimental Batch Reactor of Styrene Polymerization for Predictive Model Development. Last Accessed: 04/08/2017
http://www.aidic.it/cet/13/32/234.pdf

Sapuan, S.M., Mujtaba, I.M., (2009), Composite Materials Technology, First Edition, pp 368, , Boca Raton. Last Accessed: 04/08/2017
https://www.taylorfrancis.com/books/9781420093339

Serra, J. M., Corma, A., Valero, E., Botti, S.,(2003), Applied Catalysis A: General, Volume 254, Issue 1, pp N1-N4. Last Accessed: 04/08/2017
http://www.sciencedirect.com/science/article/pii/S0926860X03002837?via%3Dihub

Serra, Jose M, Valero, S., and Botti, V., (2003). Neural Networks for Modelling of Kinetic Reaction Data Applicable to Catalyst Scale Up and Process Control and Optimization in the Frame of Combinatorial Catalysis, Applied Catalysis A General, Polytechnic University of Valencia.

Somorjai, Gabor A., (1994). Introduction to Surface Chemistry and Catalysis, 2nd Edition, Department of Chemistry, University of California, Berkeley, California.

Srinivas G., Tarakalyani, S., (2013). Optimal Control to Sensor less Vector Control of Induction Motor using AI Techniques, International Journal of Engineering Research & Technology (IJERT), Vol. 2, Issue 9. Last Accessed: 04/08/2017
www.ijert.org

Suzuki, K., (2011). Artificial Neural Networks Methodological Advances and Bio Medical Applications, InTech, Croatia. Last Accessed: 04/08/2017
https://www.scribd.com/document/85153627/Artificial-Neural-Networks-Methodological-Advances-and-Bio-Medical-Applications-by-Kenji-Suzuki

Siddique, N., Adeli, H., (2013), Neural Networks, Computational Intelligence: Synergies of Fuzzy Logic, Neural and Evolutionary Computing. First Edition. Last Accessed: 04/08/2017
http://onlinelibrary.wiley.com/doi/10.1002/9781118534823.ch4/summary

Sumathi, S., Paneerselvan, S., (2010) Comutational Intelligence Pagadigms, Theory & Applications Using Matlab, First Edition, pp 851, Boca Raton. Last Accessed: 04/08/2017
https://www.taylorfrancis.com/books/9781439809037

Suzuki, K., (2013). Numerical Analysis and Scientific Computing, InTech, Croatia. Last Accessed: 04/08/2017
https://www.intechopen.com/books/artificial-neural-networks-architectures-and-applications/applications-of-artificial-neural-networks-in-chemical-problems

Tao, F., Schneider, F. S., Kamat P., V. (2015). Heterogeneous Catalysis at Nanoscale for Energy Applications, Canada

Tavakolmoghadam, M., Safavi, M., (2012). An Optimized Neural Network Model of Desalination by Vacuum Mebrane Distillation Using Genetic Algorithm, Prcedia Engineering, ScienceDirect Voume 42, pp 106-112. Last Accessed: 04/08/2017
http://www.sciencedirect.com/science/article/pii/S1877705812028007?via%3Dihub

Tymvios, F.S., Jacovides, C.P., Michaelides, S.C., and Scouteli, C. (2005) Comparative study of Ångström's and artificial neural networks' methodologies in estimating global solar radiation'. Sol Energy, 78, 752–762

Valero, S., Argente, E., Botti, V., Serra, J. M., Corma Avelino, (2009). Soft Computing Techniques Applied to Catalytuc Reactions, Artificial Intelligence, pp 536-545. Last Accessed: 04/08/2017
https://link.springer.com/chapter/10.1007%2F978-3-540-25945-9_53

Vargas, R., (2015). Applying Neural Networks and Analogous Estimating to Determine the Project Budget. Last Accessed: 04/08/2017
http://www.ricardo-vargas.com/downloads/download-file/396/

Xiaohu, Z., (2006). Research on Fabrication of microstructure by using U-LIGA, International Technology and Innovation Confrence
http://mr.crossref.org/iPage?doi=10.1049%2Fcp%3A20060943

Zahed,I., G., Fgaier, H., Jahanmiri, A. & Al-Enezi, G. (2006) Artificial Neural Network Identification and Evaluation of Hydrotreater Plant, PetroleumScience

Zahed, I., G., (2007). Artificial Neural Network Identification/ Evaluation of Hydrotreater Plant. Last Accessed: 04/08/2017
http://www.informaworld.com/openurl?genre=article&doi=10.1081/LFT-200056789&magic=crossref%7C%7CD404A21C5BB053405B1A640AFFD44AE3

Appendix A

Sample Calculations

Targeted	Predicted	Error	|Error|	Error^2	SSE
0.5720	0.5700	0.001957	0.001957	0.000003829	0.00003968
0.7950	0.7995	-0.004476	0.004476	0.000002004	**RMSE**
0.7980	0.7947	0.003256	0.003256	0.000001060	0.002.817
0.5510	0.5523	-0.001268	0.001268	0.000001608	**MAD**
0.8460	0.8479	-0.001898	0.001898	0.000003601	0.002571

$$Error = Targeted\ value - Predicted\ value$$

$$n = Number\ of\ sample$$

$$SSE = \sum Error^2$$

$$SSE = 0.00003829 + 0.00002004 + 0.0000106 + 0.00000360E - 6$$

$$SSE = 0.00003968$$

$$RMSE = \sqrt[2]{\frac{SSE}{n}}$$

$$= \sqrt[2]{\frac{0.00003968}{5}}$$

$$RMSE = 0.002817$$

$$MAD = \frac{\sum |Error|}{n}$$

$$MAD = \frac{0.001957 + 0.004476 + 0.003256 + 0.001268 + 0.001898}{5}$$

$$= 0.002571$$

Appendix B

Table B.1.1 Hydrogenolysis normalized raw data

Compound	Co	Ni	Cu	Ru	Rh	Pd	Re	Os	Ir	Pt	Fe	CH3	CH2	SiO2	Al2O3	Temp.	HC Conc.	H2 Conc.	Log10Rate
Ethane	**0.05**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**	0.867	0.951	0.955	0.769
	0	0.05	0	0	0	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.867
	0	0	0	0.01	0	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.919
	0	0	0	0.05	0	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.877
	0	0	0	0	0.001	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.831
	0	0	0	0	0.003	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.831
	0	0	0	0	0.01	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.859
	0	0	0	0	0.05	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.849
	0	0	0	0	0.1	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.849
	0	0	0	0	0	0.05	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.552
	0	0	0	0	0	0	0.05	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.790
	0	0	0	0	0	0	0	0.05	0	0	0	1	0	1	0	0.867	0.951	0.955	0.968
	0	0	0	0	0	0	0	0	0.05	0	0	1	0	1	0	0.867	0.951	0.955	0.813
	0	0	0	0	0	0	0	0	0	0.05	0	1	0	1	0	0.867	0.951	0.955	0.552
	0	0	0.0013	0	0.01	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.859
	0	0	0.0032	0.01	0	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.831
	0	0	0.0063	0.01	0	0	0	0	0	0	0	1	0	1	0	0.867	0.951	0.955	0.730
	0	0	0.001	0	0	0	0	0.01	0	0	0	1	0	1	0	0.867	0.951	0.955	0.831
	0	0	0.0017	0	0	0	0	0.01	0	0	0	1	0	1	0	0.867	0.951	0.955	0.790
Propane	**0**	**0**	**0**	**0.005**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0.66667**	**0.3333**	**0**	**1**	0.701	0.999	0.987	0.800
	0.063	0	0	0	0	0	0	0	0	0	0	0.66667	0.3333	1	0	0.867	1.000	1.000	0.813
	0.07	0	0	0	0	0	0	0	0	0	0	0.66667	0.3333	1	0	0.867	1.000	1.000	0.859
	0	0.041	0	0	0	0	0	0	0	0	0	0.66667	0.3333	1	0	0.867	1.000	1.000	0.741
	0	0	0	0.005	0	0	0	0	0	0	0	0.66667	0.3333	0	1	0.867	1.000	1.000	1.000
	0	0	0	0	0	0	0	0	0	0.02	0	0.66667	0.3333	0	1	0.950	0.979	0.989	0.712
n-pentane	0	0	0	0	0.005	0	0	0	0	0	0	0.4	0.6	0	1	0.619	0.932	0.946	0.681
	0	0	0	0	0.019	0	0	0	0	0	0	0.4	0.6	0	1	0.619	0.932	0.946	0.671
	0	0	0	0	0.087	0	0	0	0	0	0	0.4	0.6	0	1	0.619	0.932	0.946	0.650
	0	0	0	0	0.124	0	0	0	0	0	0	0.4	0.6	0	1	0.619	0.932	0.946	0.644
	0	0	0	0	0	0	0.01	0	0	0	0	0.4	0.6	0	1	0.760	0.977	0.989	0.748
	0	0	0	0	0	0	0	0	0.006	0	0	0.4	0.6	0	1	0.760	0.977	0.989	0.712
	0	0	0	0	0	0	0.01	0	7E-04	0	0	0.4	0.6	0	1	0.760	0.977	0.989	0.759
	0	0	0	0	0	0	0	0	0.003	0	0	0.4	0.6	0	1	0.760	0.977	0.989	0.766
	0	0	0	0	0	0	0	0	0.005	0	0	0.4	0.6	0	1	0.760	0.977	0.989	0.748
	0	0	0	0	0	0	0	0	0.003	0	0	0.4	0.6	0	1	0.867	0.977	0.989	0.867
	0	0	0.0005	0	0	0	0	0	0.002	0	0	0.4	0.6	0	1	0.867	0.977	0.989	0.689
	0	0	0.00125	0	0	0	0	0	0.001	0	0	0.4	0.6	0	1	0.867	0.977	0.989	0.741
	0	0	0.002	0	0	0	0	0	5E-04	0	0	0.4	0.6	0	1	0.867	0.977	0.989	0.689
	0	0	0	0.0033	0	0	0	0	0	0	0	0.4	0.6	0	1	0.867	0.990	0.987	0.822
	0	0	0	0.0086	0	0	0	0	0	0	0	0.4	0.6	0	1	0.867	0.990	0.987	0.831

Appendix B

Table B.1.2 Hydrogenolysis raw data

Compound	Co	Ni	Cu	Ru	Rh	Pd	Re	Os	Ir	Pt	Fe	CH3	CH2	SiO2	Al2O3	Temp.	HC Conc.	H2 Conc.	Log10Rate
n-hexane	**0**	**0.09**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	0.33333	0.6667	1	0	1.000	0.995	0.986	0.576
	0	0	0	0	0	0	0	0	0	0	0	0.33333	0.6667	0	1	0.950	0.974	0.989	0.769
	0	0	0	0	0	0	0	0	0	0.01	0	0.33333	0.6667	0	1	0.950	0.974	0.989	0.763
n-butane	0	0	0	0	0	0	0	0	0.002	0	0	0.5	0.5	0	1	1.000	1.000	1.000	0.867

Appendix B

Table B.2.1 Hydrogenolysis raw data

Compound	Co	Ni	Cu	Ru	Rh	Pd	Re	Os	Ir	Pt	Fe	Au	Sn	Tc	CH3	CH2	CH	SiO2	Al2O3	Temp	HC Conc.	H2 Conc.	Log10Rate
Ethylene	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	0	0.758	0.941	0.928	0.910
	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	1	0.758	0.941	0.928	0.918
	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	0	0.758	0.941	0.928	0.932
Ethylene	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	1	0.949	0.941	0.928	0.959
Ethylene	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	0	0.949	0.941	0.928	0.882
Ethylene	0	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	1	0	0.796	0.941	0.928	0.775
	0	0	0	0	0	0	0	0	0	0.005	0	0	0	0	0	1	0	0	0	0.481	0.941	0.928	0.793
	0	0	0	0	0	0	0	0	0	0.005	0	0	0	0	0	1	0	1	0	0.481	0.941	0.928	0.765
	0	0	0	0	0	0	0	0	0	0.012	0	0	0	0	0	1	0	1	0	0.491	0.937	0.962	0.724
	0	0	0	0	0	0	0	0	0	0.12	0	0	0	0	0	1	0	1	0	0.491	0.937	0.962	0.759
C3H6	0	0	0	0	0	0	0	0	0	0.07	0	0	0	0	0.333	0.333	0.333	1	0	0.550	0.953	0.962	0.810
C3H6	0	0	0	0	0	0	0	0	0	0.81	0	0	0	0	0.333	0.333	0.333	1	0	0.550	0.953	0.962	0.834
C5H10	0	0	0	0	0	0	0	0	0	0	0	0.001	0	0	0	1	0	1	0	1.000	0.875	0.963	0.682
C5H10	0	0	0	0	0	0	0	0	0	0	0	0.007	0	0	0	1	0	1	0	1.000	0.875	0.963	0.641
C5H10	0	0	0	0	0	0	0	0	0	0	0	0.003	0	0	0	1	0	1	0	1.000	0.875	0.963	0.606
C5H10	0	0	0	0	0	0	0	0	0	0	0	0.01	0	0	0	1	0	1	0	1.000	0.875	0.963	0.524
C5H10	0	0	0	0	0	0	0	0	0	0	0	0.01	0	0	0	1	0	0	1	1.000	0.875	0.963	0.506
Benzene	0	0.043	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.758	0.978	0.959	0.682
	0	0.79	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.758	0.978	0.959	0.703
	0	0.043	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.758	0.967	0.958	0.765
Benzene	0	0.52	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.758	0.967	0.958	0.778
Benzene	0	0.75	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.758	0.920	0.924	0.675
	0	0	0	0	0	0.006	0	0	0	0	0	0	0	0	0	1	0	0	1	0.758	0.936	0.962	0.765
	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0.758	0.936	0.962	0.882
	0	0	0	0	0	0	0	0	0.006	0	0	0	0	0	0	1	0	0	1	0.758	0.936	0.962	0.800
	0	0	0	0	0	0	0.0006	0	0.005	0	0	0	0	0	0	1	0	1	1	0.758	0.936	0.962	0.810
Benzene	0	0	0	0	0	0	0.0027	0	0.003	0	0	0	0	0	0	1	0	1	1	0.758	0.936	0.962	0.810
Benzene	0	0	0	0	0	0	0.0052	0	8E-04	0	0	0	0	0	0	1	0	1	1	0.758	0.936	0.962	0.800
	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	1	0	0	1	0.758	0.962	0.961	0.818

Appendix B

Table B.2.2 Hydrogenation raw data

Compound	Co	Ni	Cu	Ru	Rh	Pd	Re	Os	Ir	Pt	Au	Sn	Tc	CH3	CH2	CH	SiO2	Al2O3	Temp	HC Conc.	H2 Conc.	Log10Rate
	0	0	0	0	0	0	0	0	0	0.028	0	0	0	0	1	0	1	0	0.758	0.941	0.962	0.806
	0	0	0	0	0	0	0	0	0	0.02	0	0	0	0	1	0	1	0	0.758	0.941	0.962	0.688
	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.947	0.948	0.767
	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.947	0.948	0.775
	0	0	0	0	0	0	0	0	0	0	0	0	0.001	0	1	0	1	0	0.949	0.885	0.973	0.703
	0	0	0	0	0	0	0	0	0	0	0	0	0.001	0	1	0	0	1	0.949	0.885	0.973	0.747
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.885	0.973	0.824
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.885	0.973	0.824
	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0.949	1.000	1.000	0.800
	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.800
	0	0	0	0	0	0.025	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.930	0.962	0.724
Benzene	0	0	0	0	0	0.025	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.930	0.962	0.756
Benzene	0	0	0	0	0	0.001	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.885	0.973	0.741
	0	0	0	0	0	0.001	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.885	0.973	0.724
	0	0	0	0	0	0	0.001	0	0	0	0	0	0	0	1	0	1	0	0.949	0.885	0.973	0.675
	0	0	0	0	0	0	0.001	0	0	0	0	0	0	0	1	0	0	1	0.949	0.885	0.973	0.647
	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.837
	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.877
	0	0	0	0	0	0	0	0	0.003	0	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.865
	0	0	0	0	0	0	0	0	5E-04	0	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.852
	0	0	0	0	0	0	0	0	0.001	0	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.824
	0	0	0	0	0	0	0	0	0.002	0	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.806
	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0.949	1.000	1.000	0.873
	0	0	0	0	0	0	0	0	0	0.005	0	0	0	0	1	0	0	1	0.949	0.962	0.955	0.841
	0	0	0	0	0	0	0	0	0	0.004	0	0.005	0	0	1	0	0	1	0.949	0.962	0.955	0.782
Benzene	0	0	0	0	0	0	0	0	0	0.004	0	0.007	0	0	1	0	0	1	0.949	0.962	0.955	0.834
	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.865
	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.806

Appendix B

Table 2.3 Hydrogenation raw data

Compound	Co	Ni	Cu	Ru	Rh	Pd	Re	Os	Ir	Pt	Au	Sn	Tc	CH3	CH2	CH	SiO2	Al2O3	Temp	HC Conc.	H2 Conc.	Log10Rate
	0	0	0	0	0	0	0.2	0	0	0.8	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.810
	0	0	0	0	0	0	0.5	0	0	0.5	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.793
Benzene	0	0	0	0	0	0	0.6	0	0	0.4	0	0	0	0	1	0	0	1	0.949	0.953	0.962	0.814
	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	1	0	1	0	0.949	0.947	0.948	0.775
	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0.949	0.936	0.962	0.865
	0	0	0	0	0	0	0	0	0	0.01	0	0	0	0	1	0	1	0	0.949	0.885	0.973	0.824
	0	0	0	0	0	0	0	0	0	0.01	0	0	0	0	1	0	0	1	0.949	0.885	0.973	0.800
	0	0.05	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.956	0.962	0.836
Benzene	0	0.05	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0.949	0.956	0.962	0.771
	0	0.08	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0.949	0.953	0.961	0.810
	0	0	0	0	0	0.08	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.962	0.961	0.832
	0	**0**	**0**	**0**	**0**	**0.013**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	0	**1**	0	**1**	**0**	**0.949**	**0.962**	**0.961**	**0.810**
Benzene	0	0	0	0	0	0.01	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.962	0.961	0.759
Benzene	0	0	0	0	0	0.022	0	0	0	0	0	0	0	0	1	0	0	1	0.949	0.940	0.963	0.923
	0	0	0	0	0	0	0	0	0.001	0	0	0	0	0	1	0	0	1	0.949	0.994	0.955	0.824
	0	0	0	0	0	0	0	0	0	0.007	0	0	0	0	1	0	1	1	0.949	0.971	0.960	0.918
Benzene	0	0	0	0	0	0	0	0	0	0.007	0	0	0	0	1	0	0	1	0.949	0.971	0.960	0.923
	0	0	0	0	0	0	0	0	0	0.01	0	0	0	0	1	0	0	1	0.949	0.962	0.955	0.806
	0	0	0	0	0	0.032	0	0	0	0.008	0	0	0	0	1	0	0	1	0.949	0.962	0.955	0.775
	0	0	0	0	0	0	0	0	0	0.039	0	0	0	0	1	0	0	1	0.949	0.959	0.943	0.793
H2C=CH2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.959	0.919	0.882
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.959	0.919	0.910
	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.959	0.919	0.882
H2C=CH2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.959	0.919	0.959
H2C=CH2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0.949	0.959	0.919	1.000

Appendix B

Table B.3.1

Variant	Date	Type Variant	Total aq Phase Time (min)	Addition of liquid ingredients (min)	Addition od Dry ingredients (min)	Addition of Perfume (min)	Transfer of Premix (min)
VSL Dry Skin	17-01-20	0	1.000	0.632	0.261	0.500	0.333
VICl Adv Repair	17-01-23	0	0.754	1.000	0.217	0.500	1.000
Dawn	17-01-26	1	0.444	0.789	0.478	0.500	0.385
Dawn Active Men	17-01-30	1	0.373	0.632	0.304	0.500	0.205
Dawn Men refresh cooling	17-02-02	1	0.468	0.737	0.739	0.500	0.359
Dawn Special musk	17-01-20	1	0.262	0.553	0.217	1.000	0.128
VICL Coco Glow	17-01-25	0	0.278	0.526	0.304	0.500	0.154
Dawn Active Men	17-01-30	1	0.317	0.763	0.130	0.500	0.179
VSL W Even Tone	17-02-02	0	0.508	0.737	1.000	0.500	0.308

Table B.3.2

Variant	Date	Type Variant	Oil Phase (min)	Addition of PJ (min)	Addition of Liquid ingredients (min)	Addition of Dry ingredients (min)	Time to raise temperature (min)
VSL Dry Skin	17-01-20	0	1.000	1.000	0.300	0.000	1.000
VICl Adv Repair	17-01-23	0	0.363	0.145	0.225	0.000	0.474
Dawn	17-01-26	1	0.363	0.145	0.225	0.000	0.474
Dawn Active Men	17-01-30	1	0.357	0.053	0.250	0.254	0.605
Dawn Men refresh cooling	17-02-02	1	0.350	0.079	0.325	0.433	0.158
Dawn Special musk	17-01-20	1	0.427	0.342	0.100	0.164	0.658
VICL Coco Glow	17-01-25	0	0.363	0.145	0.225	0.000	0.474
Dawn Active Men	17-01-30	1	0.312	0.105	0.250	0.254	0.474
VSL W Even Tone	17-02-02	0	0.771	0.132	1.000	1.000	0.132

Appendix B

Table B.3.3

Variant	Date	Type Variant	Premix 1 (min)	Addition of Dry ingredients (min)	Addition of Liquid (min)	Flushing (min)	Total Time before sonolation (min)	Sonolation Time (min)
VSL Dry Skin	17-01-20	0	1.000	0.357	0.500	0.375	1.000	0.765
VICl Adv Repair	17-01-23	0	0.400	0.500	0.125	0.250	0.369	0.976
Dawn	17-01-26	1	0.360	0.357	0.250	0.250	0.369	1.000
Dawn Active Men	17-01-30	1	0.800	1.000	0.500	0.250	0.357	1.000
Dawn Men refresh cooling	17-02-02	1	0.600	0.929	0.125	0.125	0.376	1.000
Dawn Special musk	17-01-20	1	0.800	0.357	0.500	1.000	0.427	0.988
VICL Coco Glow	17-01-25	0	0.520	0.214	1.000	0.250	0.369	0.847
Dawn Active Men	17-01-30	1	0.760	1.000	0.375	0.250	0.312	1.000
VSL W Even Tone	17-02-02	0	0.200	0.143	0.250	0.250	0.771	0.659

Table B.3.4

Variant	Date	Type Variant	Aqueous Phase Inlet (kg)	Oil Phase Inlet (kg)	Water Phase Inlet (kg)	Water Phase Inlet (kg/hr)	Aqueous Phase Outlet (kg)	Batch Size (kg)
VSL Dry Skin	17-01-20	0	0.786	1.000	0.453	0.544	0.884	0.733
VICl Adv Repair	17-01-23	0	0.938	0.854	0.980	0.922	0.543	1.000
Dawn	17-01-26	1	1.000	0.494	0.988	0.908	0.084	1.000
Dawn Active Men	17-01-30	1	1.000	0.488	1.000	0.919	0.769	1.000
Dawn Men refresh cooling	17-02-02	1	0.997	0.047	0.997	0.917	0.147	1.000
Dawn Special musk	17-01-20	1	0.992	0.491	0.984	0.916	0.630	1.000
VICL Coco Glow	17-01-25	0	0.713	0.948	0.921	1.000	0.639	0.833
Dawn Active Men	17-01-30	1	0.769	0.491	0.997	0.916	1.000	1.000
VSL W Even Tone	17-02-02	0	0.649	0.482	0.628	0.876	0.665	0.667

Appendix B

Table B.3.5

Variant	Date	Type Variant	Total Cycle Time (min)	pH	Specific Gravity	Viscosity (cP)	Material Use Efficiency
VSL Dry Skin	17-01-20	0	0.822	0.928	1.000	0.453	0.959
VICl Adv Repair	17-01-23	0	1.000	1.000	0.971	1.000	0.980
Dawn	17-01-26	1	0.596	0.950	0.970	0.863	1.000
Dawn Active Men	17-01-30	1	0.642	0.962	0.964	0.930	0.973
Dawn Men refresh cooling	17-02-02	1	0.572	0.966	0.960	0.930	0.997
Dawn Special musk	17-01-20	1	0.795	0.894	0.980	0.670	0.978
VICL Coco Glow	17-01-25	0	0.798	0.997	0.980	0.919	0.968
Dawn Active Men	17-01-30	1	0.551	0.962	0.963	0.930	0.952
VSL W Even Tone	17-02-02	0	0.846	0.871	0.976	0.504	0.963

Appendix C Sensitivity Graphs

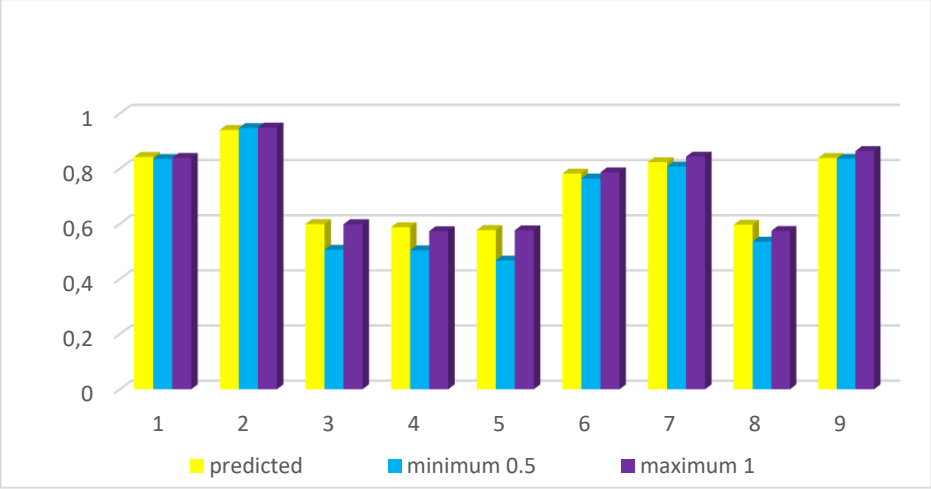


Figure C.1 Sensitivity of total cycle time on aqueous phase inlet

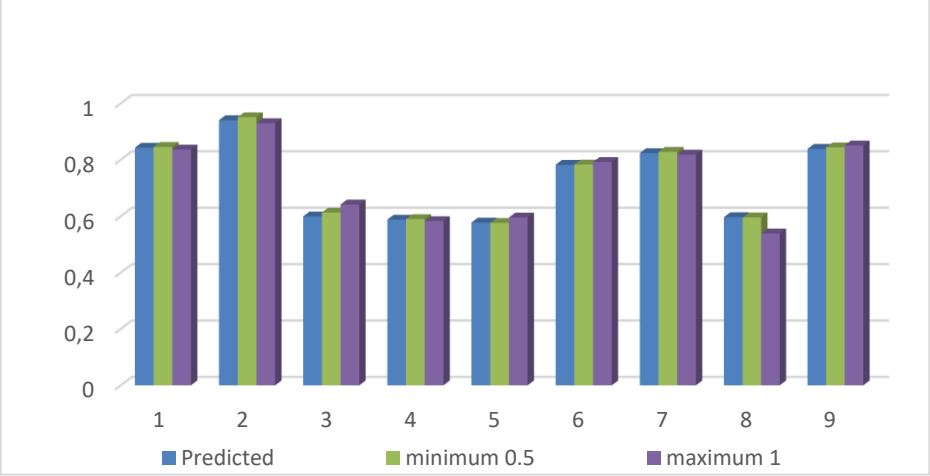


Figure C.2 Sensitivity of total cycle time on aqueous phase outlet

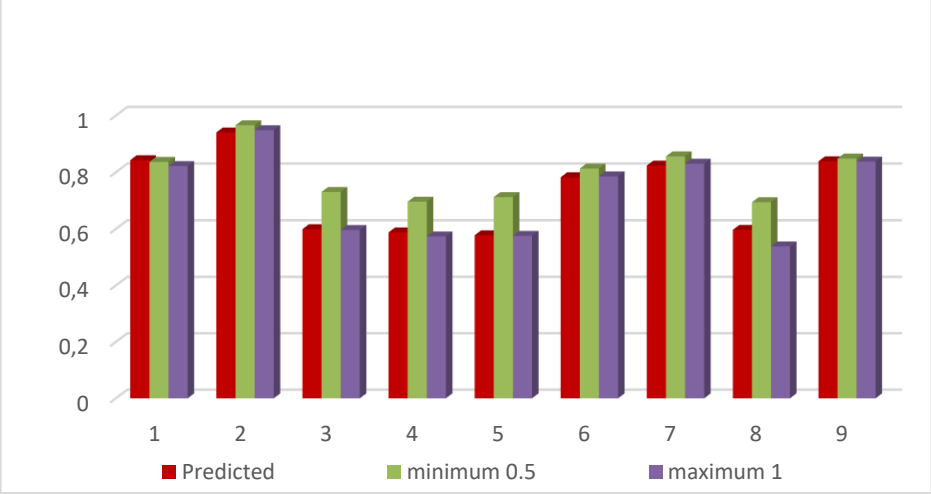


Figure C.3 sensitivity of total cycle time on water phase inlet

Appendix C

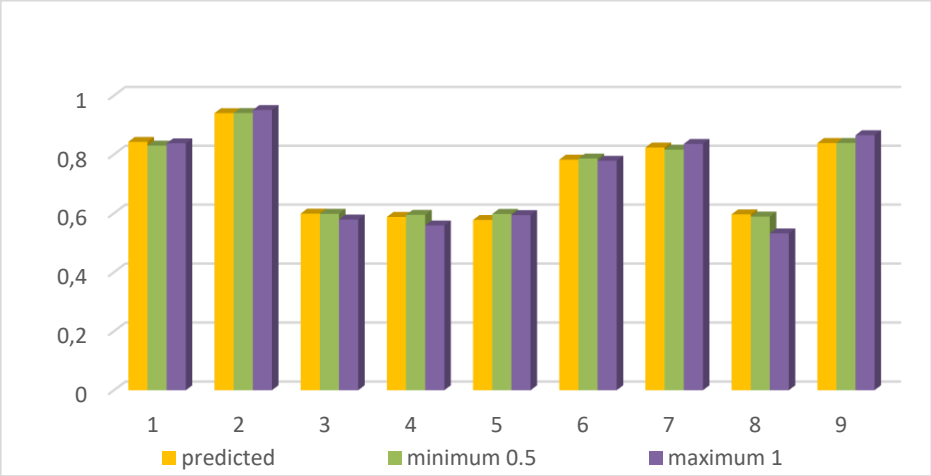


Figure C.4 Sensitivity of total cycle time on oil phase inlet

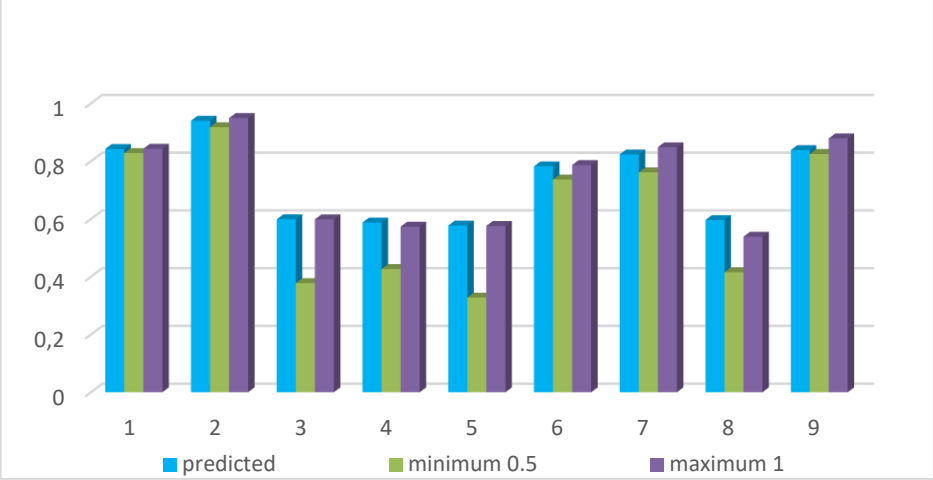


Figure C.5 Sensitivity of total cycle time on sonolation time

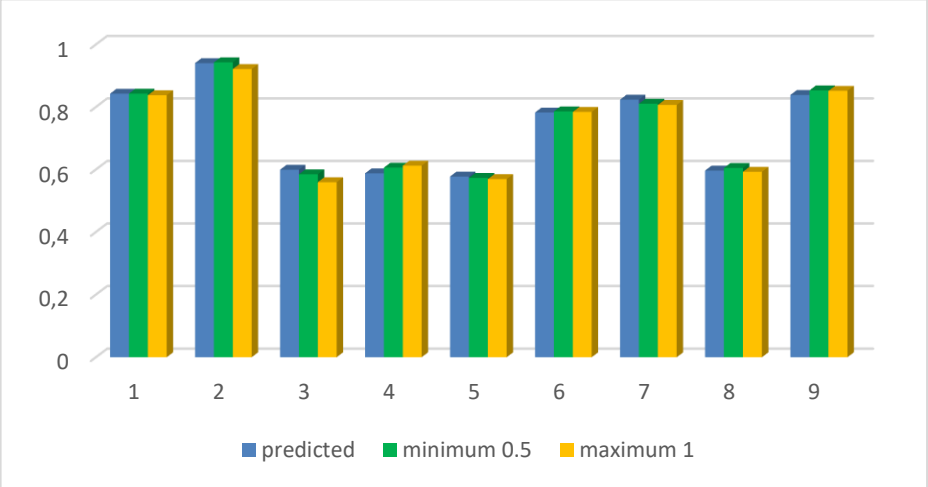


Figure C.6 Sensitivity of total cycle time on sonolation time

Appendix C

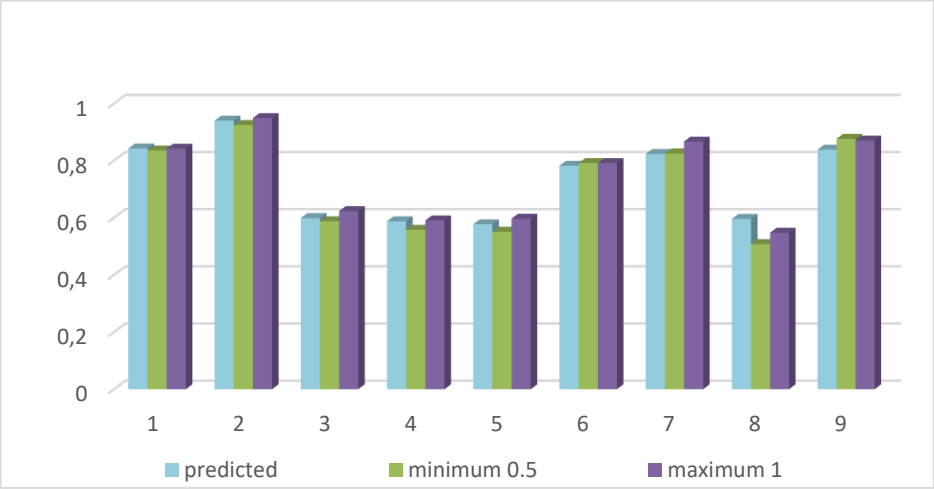


Figure C.7 Sensitivity of total cycle time on addition of liquid ingredients

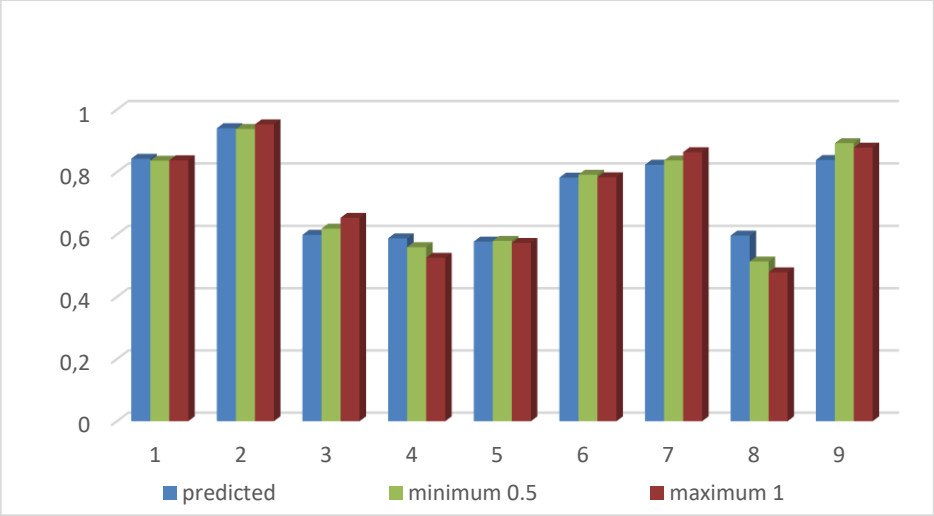


Figure C.8 Sensitivity of total cycle time on total aqueous phase time

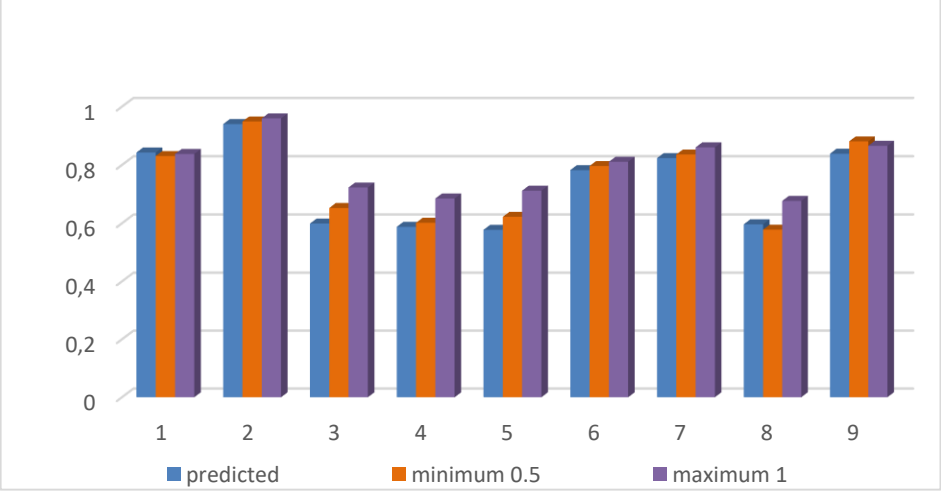


Figure C.9 Sensitivity of total cycle time on oil phase time

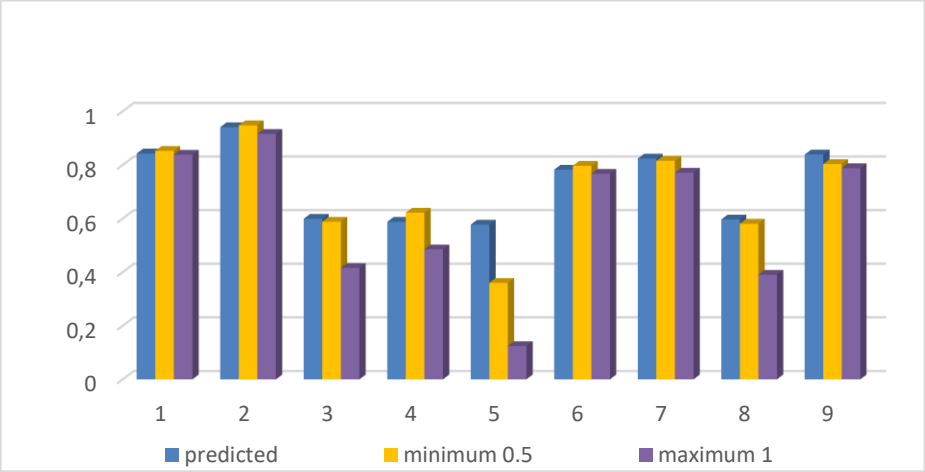


Figure C.10 Sensitivity of total cycle time on time to raise temperature

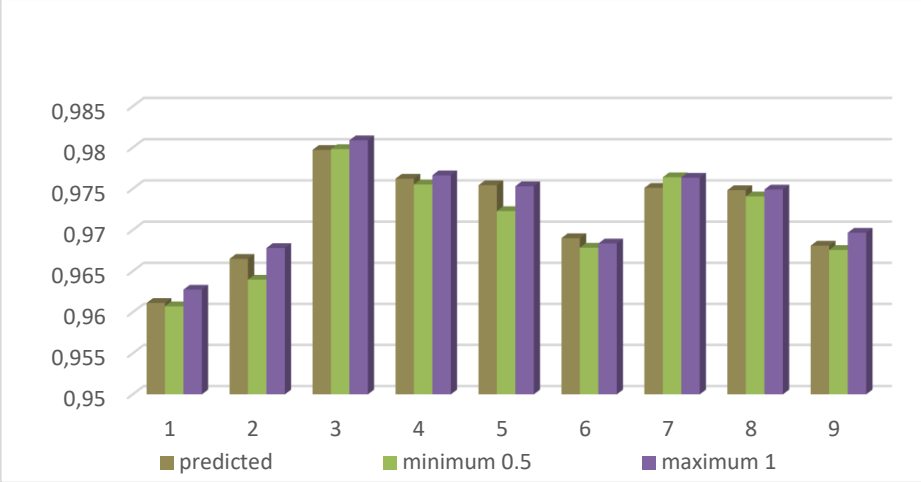


Figure C.11 Sensitivity of material utilized efficiency on aqueous phase inlet

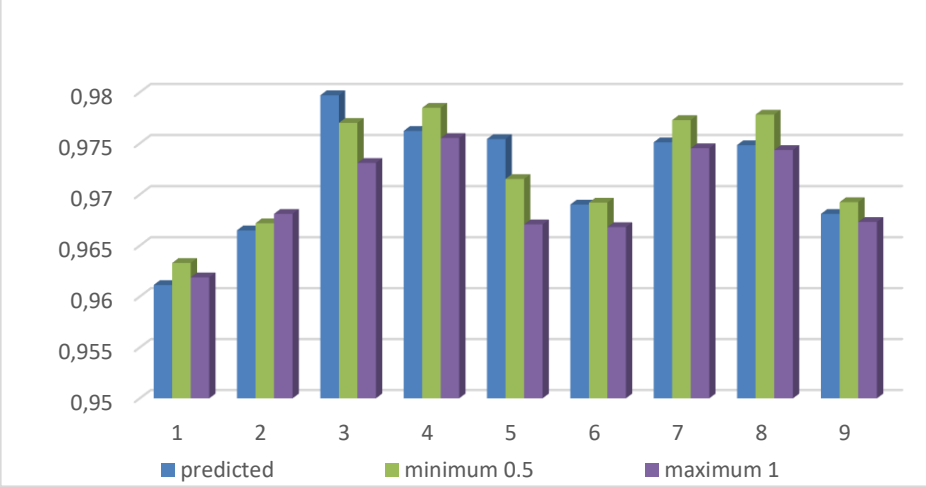


Figure C.12 Sensitivity of material utilized efficiency on aqueous phase outlet

Appendix C

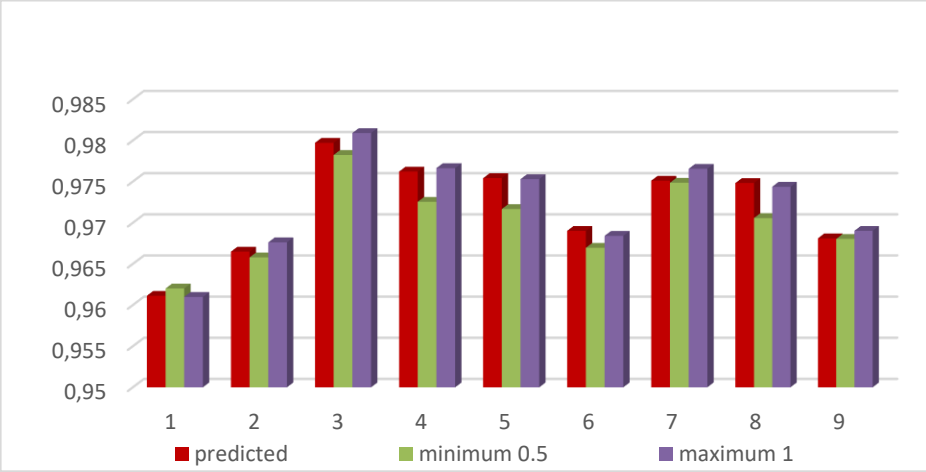


Figure C.13 Sensitivity of material utilized efficiency on water phase inlet

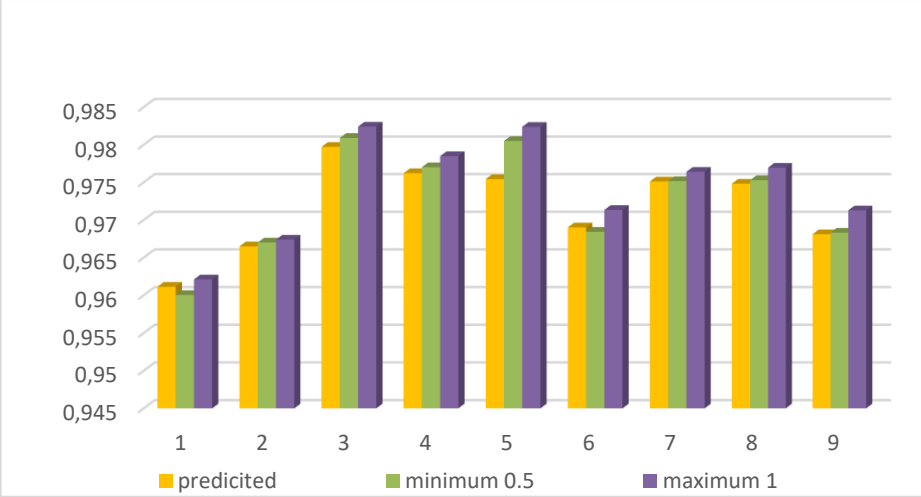


Figure C.14 Sensitivity of material utilized efficiency on oil phase inlet

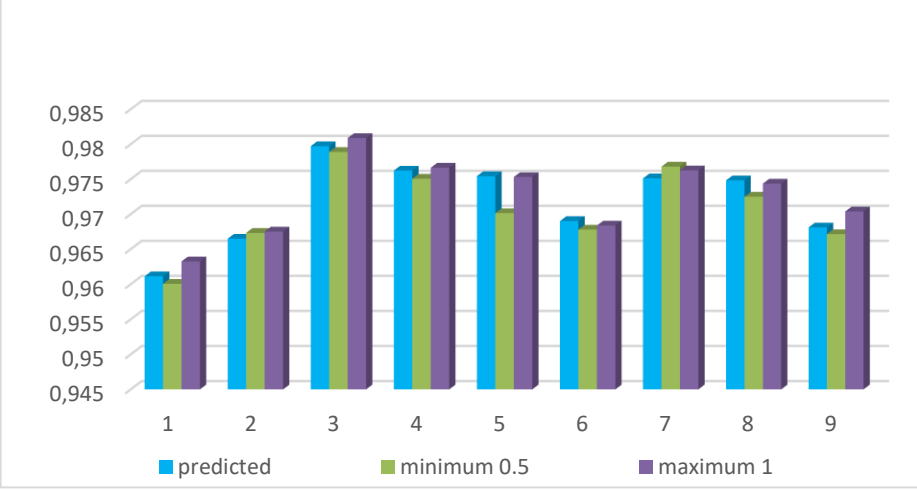


Figure C.15 Sensitivity of material utilized efficiency on sonolation time

Appendix C

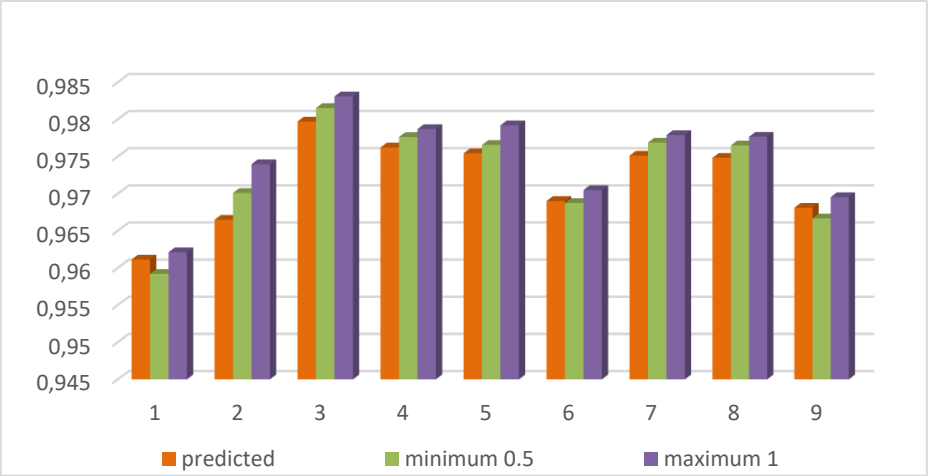


Figure C.16 Sensitivity of material utilized efficiency on time before sonolation

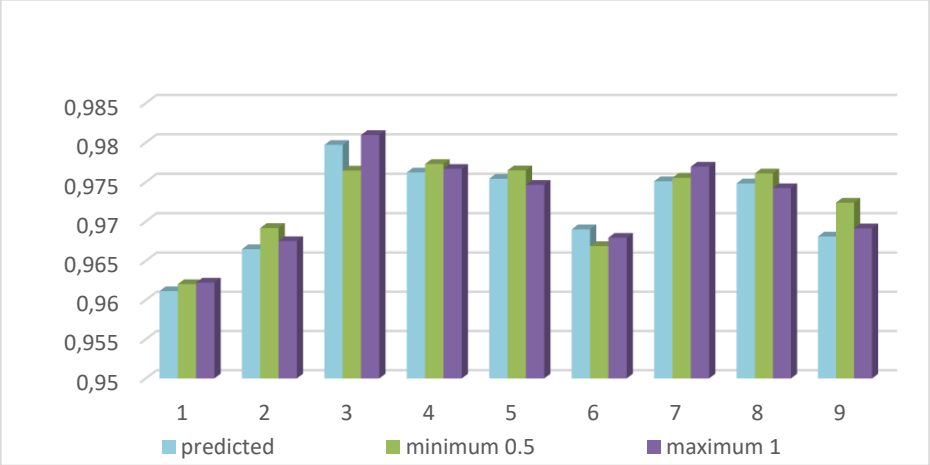


Figure C.17 Sensitivity of material utilized efficiency on addition of liquid ingredients

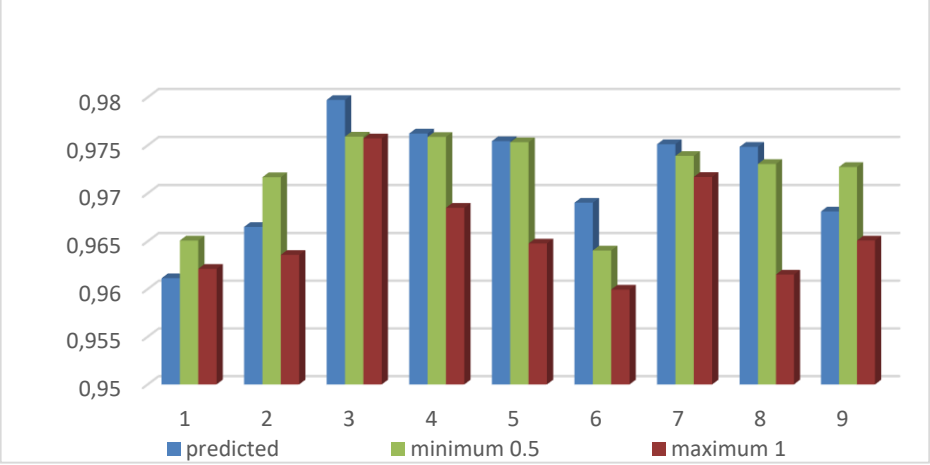


Figure C.18 Sensitivity of material utilized efficiency on total aqueous phase time

Appendix C

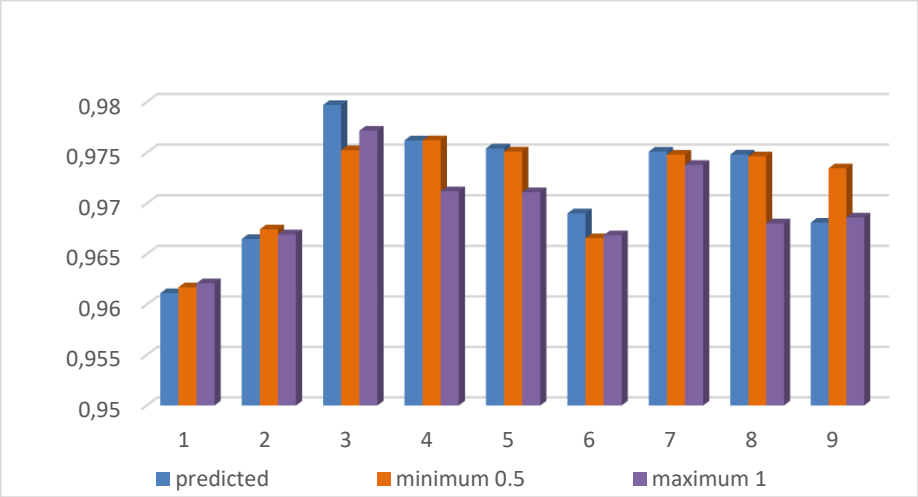


Figure C.19 Sensitivity of material utilized efficiency on oil phase time

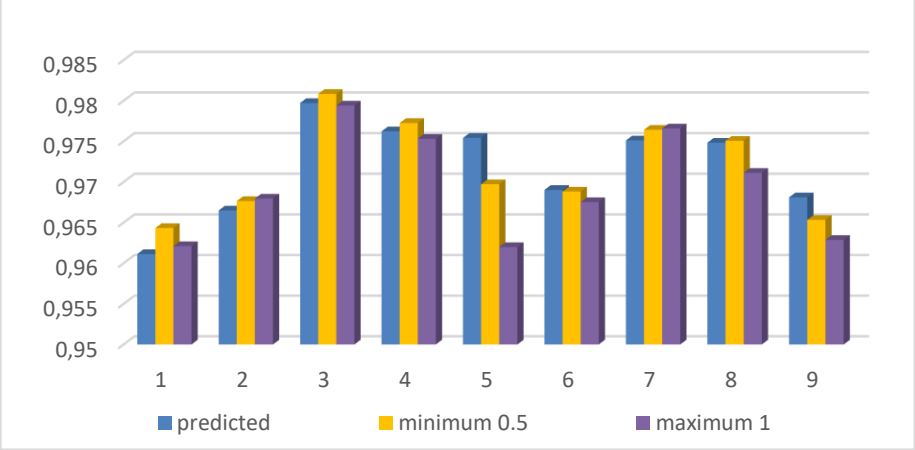


Figure C.20 Sensitivity of material utilized efficiency on time to raise temperature

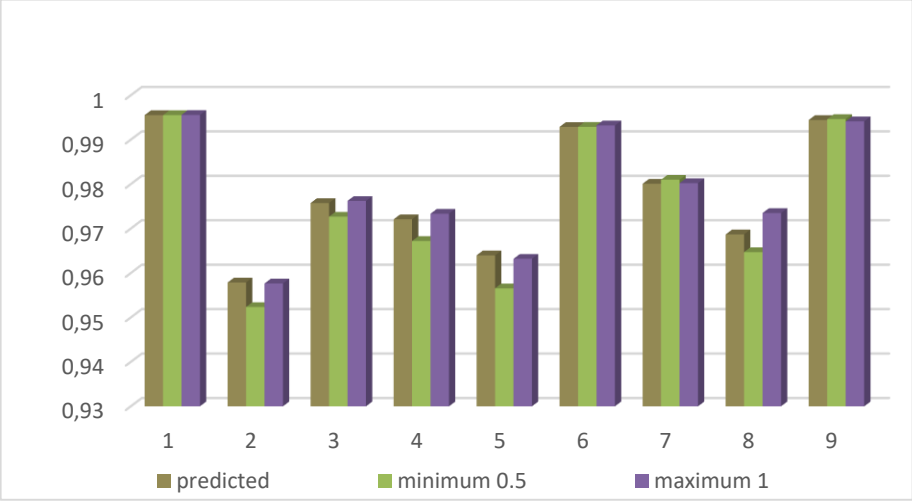


Figure C.21 Sensitivity of specific gravity on aqueous phase inlet

Appendix C

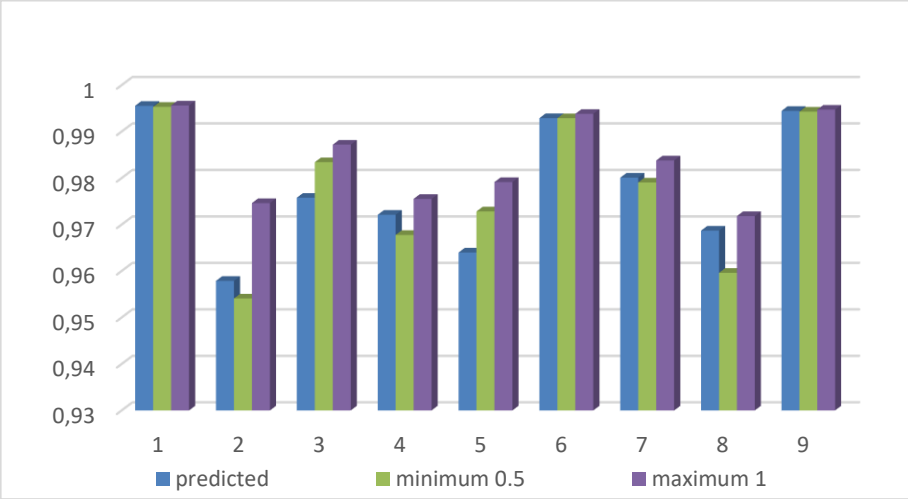


Figure C.22 Sensitivity of specific gravity aqueous phase outlet

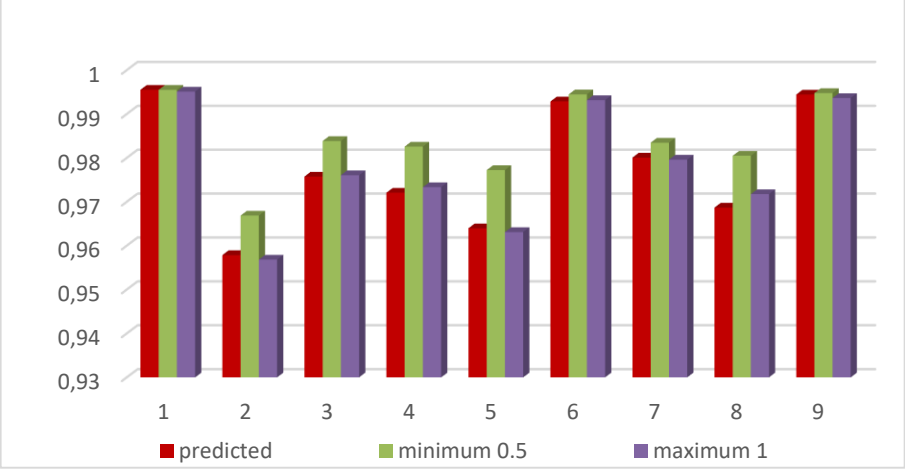


Figure C.23 Sensitivity of specific gravity on water phase inlet

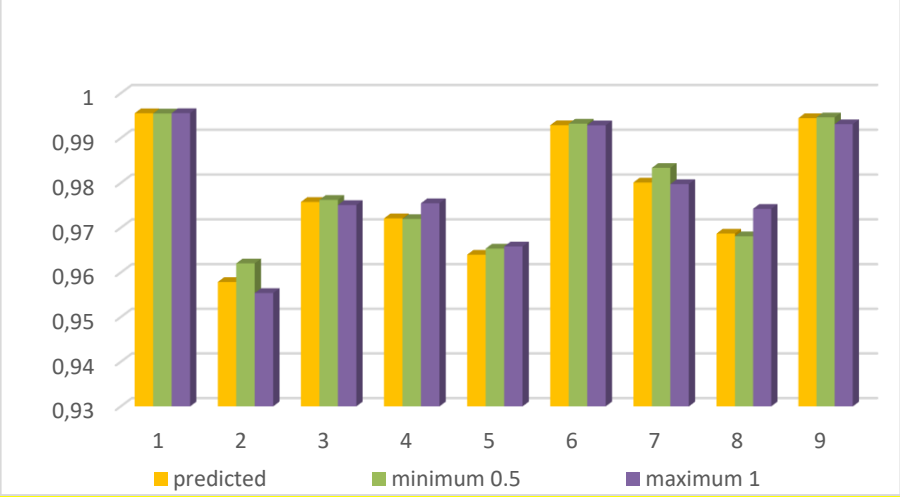


Figure C.24 Sensitivity of specific gravity on oil phase inlet

Appendix C

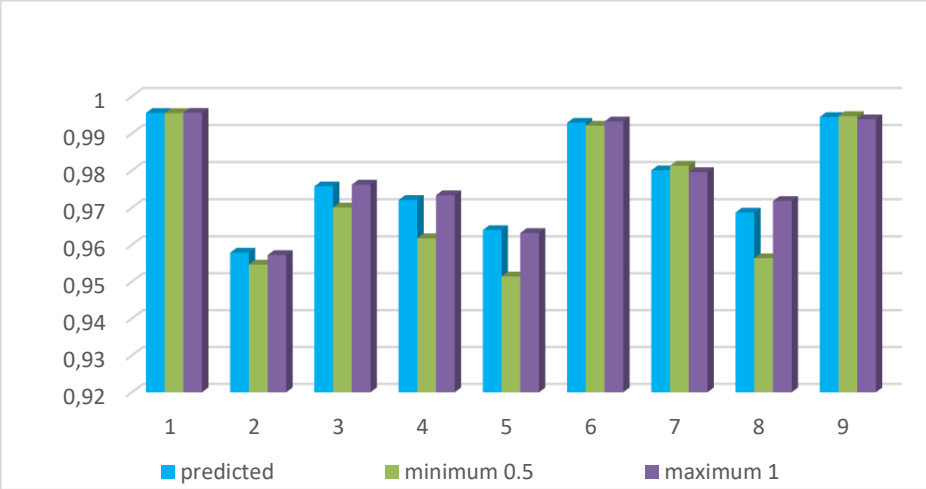


Figure C.25 Sensitivity of specific gravity on sonolation time

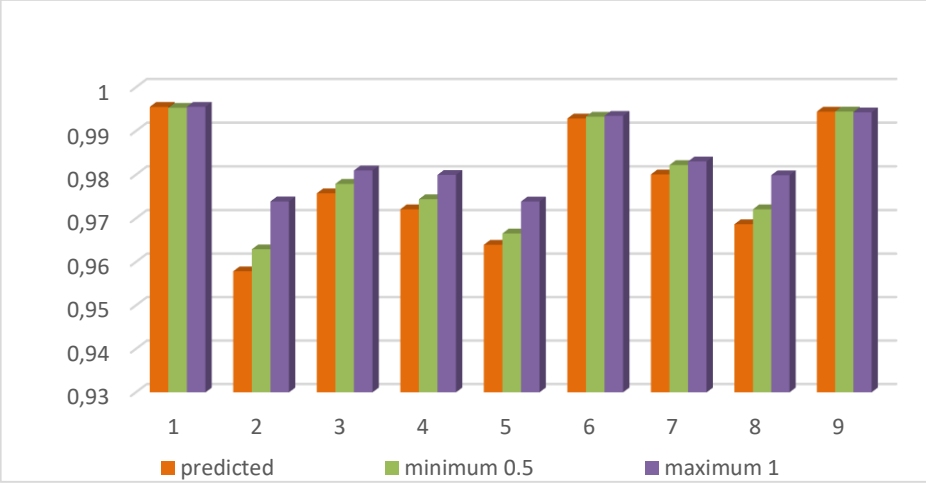


Figure C.26 Sensitivity of specific gravity on time before sonolation

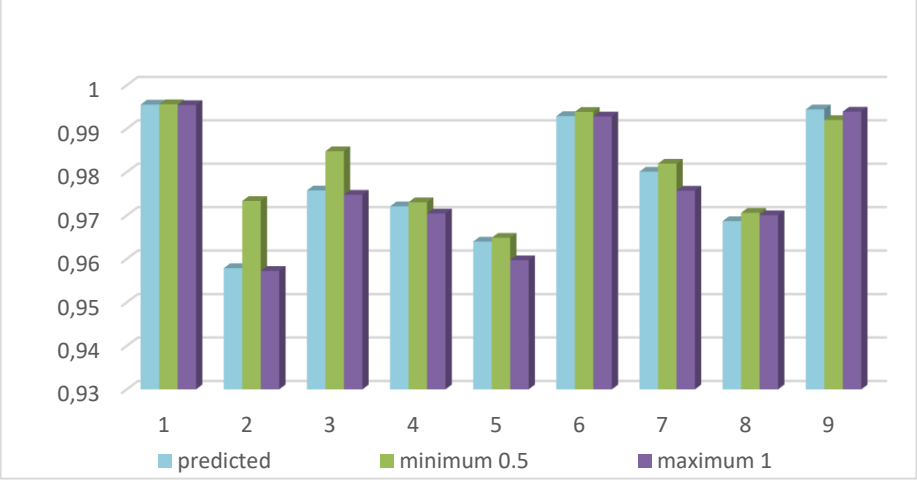


Figure C.27 Sensitivity of specific gravity on (time) addition of liquid ingredients

Appendix C

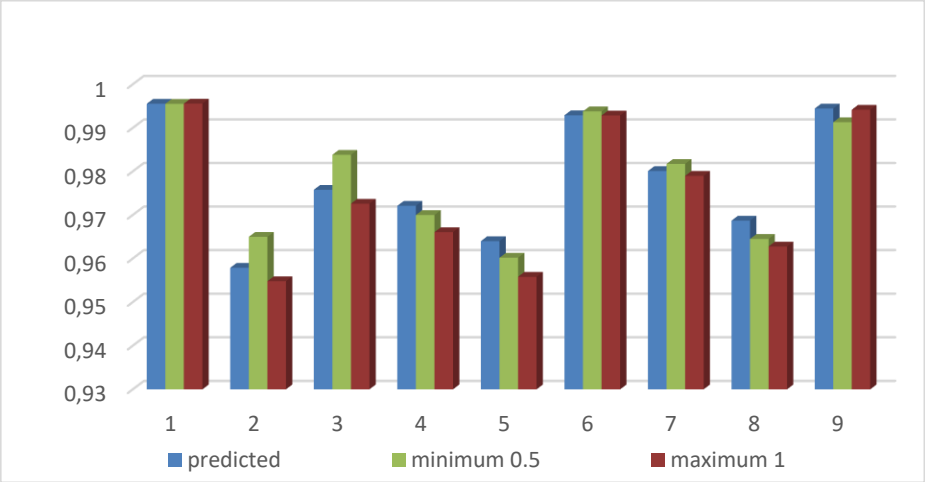


Figure C.28 Sensitivity of specific gravity on total aqueous phase time

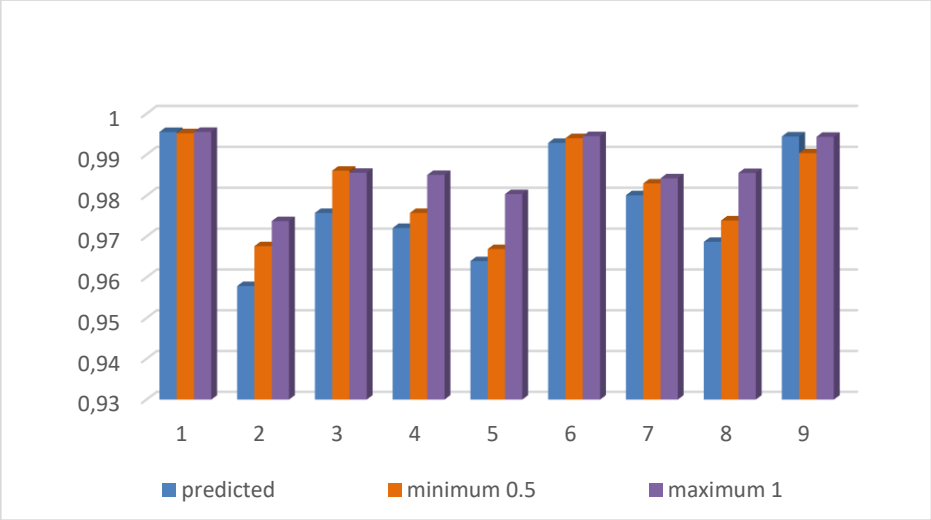


Figure C.29 Sensitivity of specific gravity on oil phase time

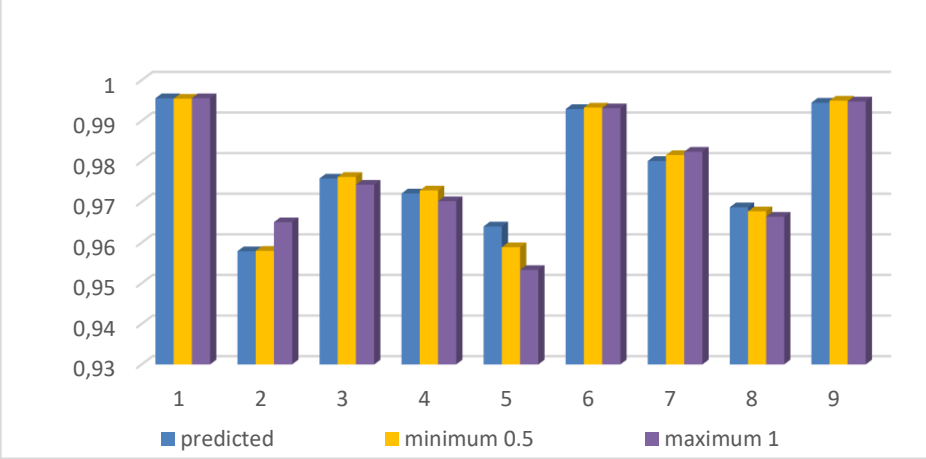


Figure C.30 Sensitivity of specific gravity of time to raise temperature

Appendix C

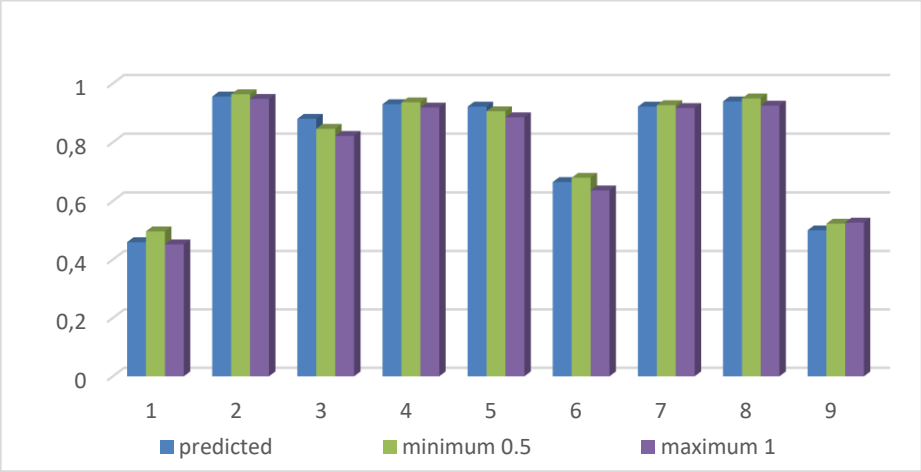


Figure C.31 Sensitivity of viscosity on aqueous phase outlet

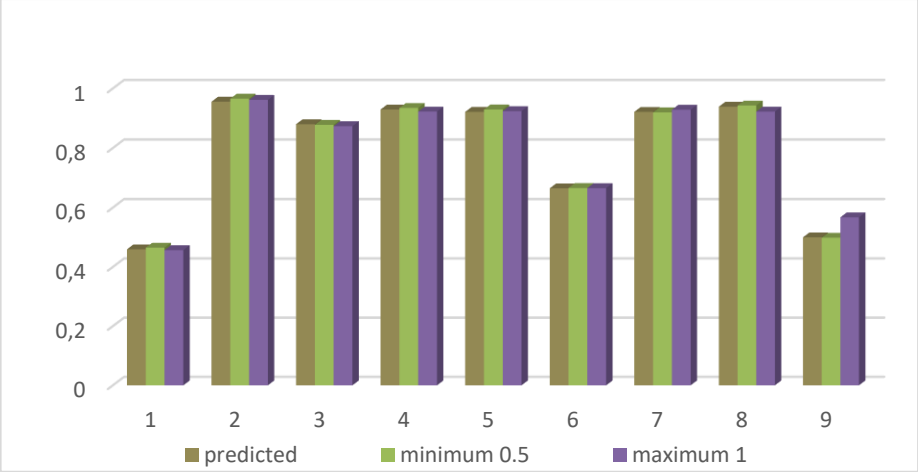


Figure C.32 Sensitivity of viscosity on aqueous phase inlet

Figure C.33 Sensitivity of viscosity on water phase inlet

Appendix C

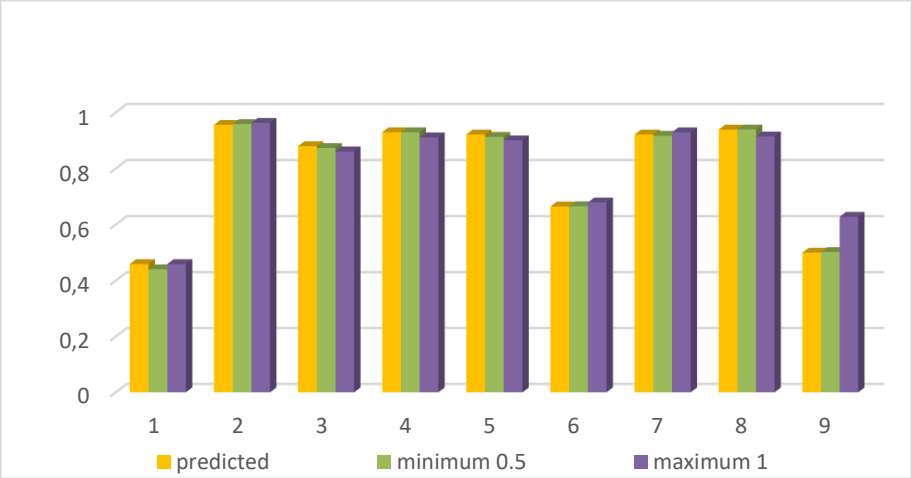


Figure C.34 Sensitivity of viscosity on oil phase inlet

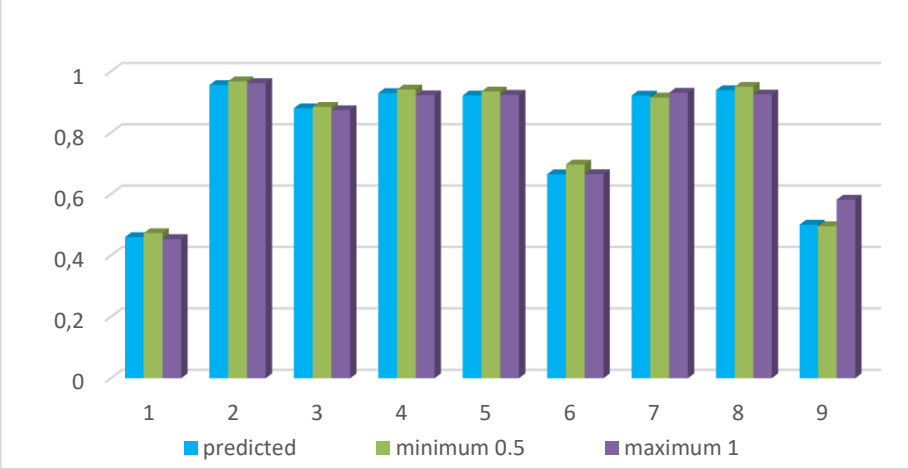


Figure C.35 Sensitivity of viscosity on sonolation time

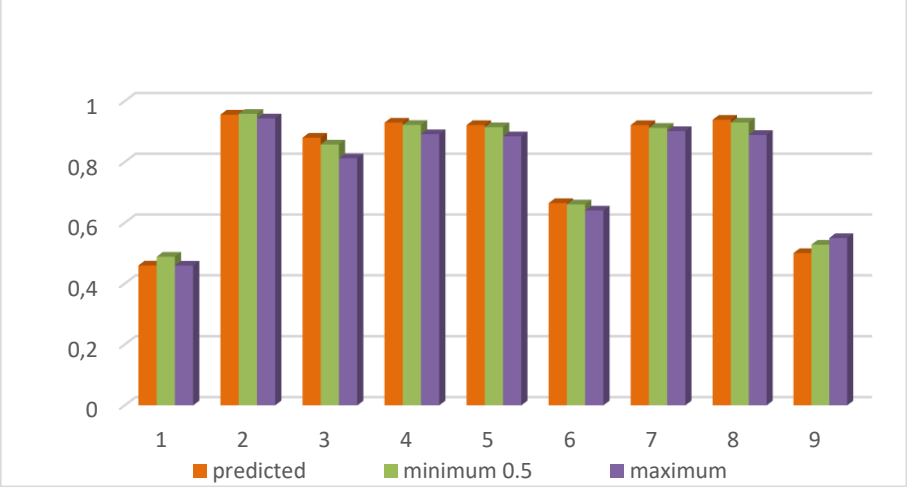


Figure C.36 Sensitivity of viscosity on time before sonolation

Appendix C

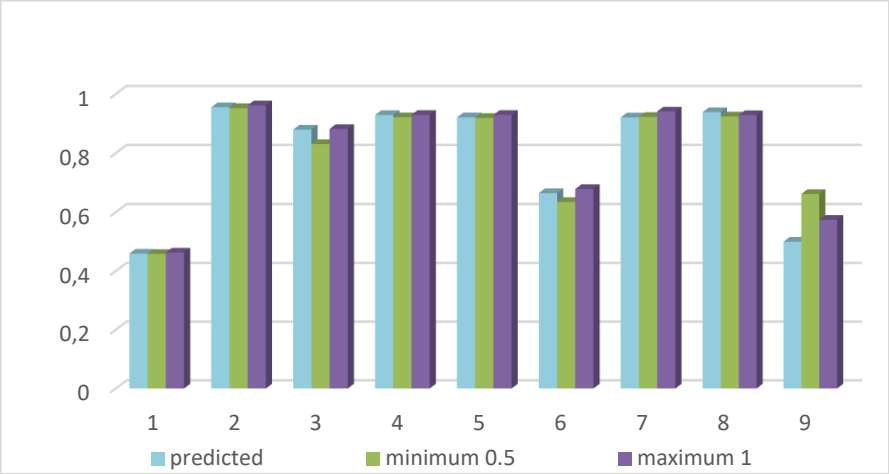


Figure C.37 Sensitivity of viscosity on addition of liquid ingredients

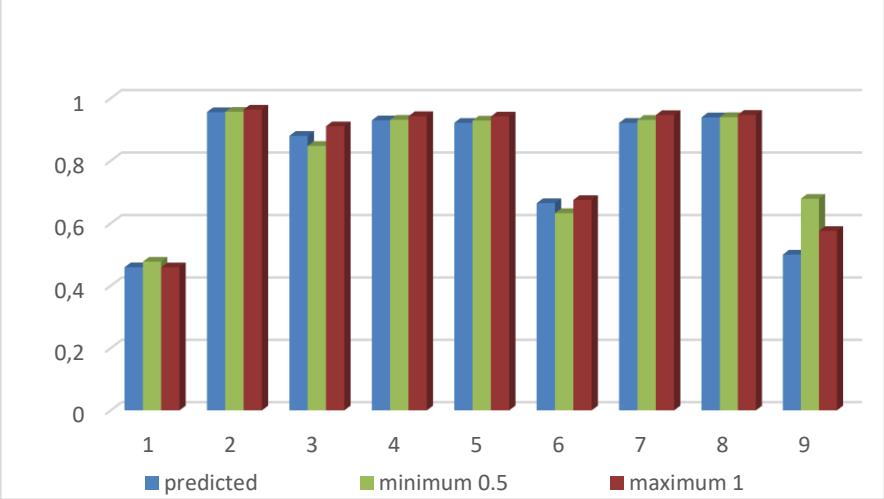


Figure C.38 Sensitivity of viscosity on total aqueous phase time

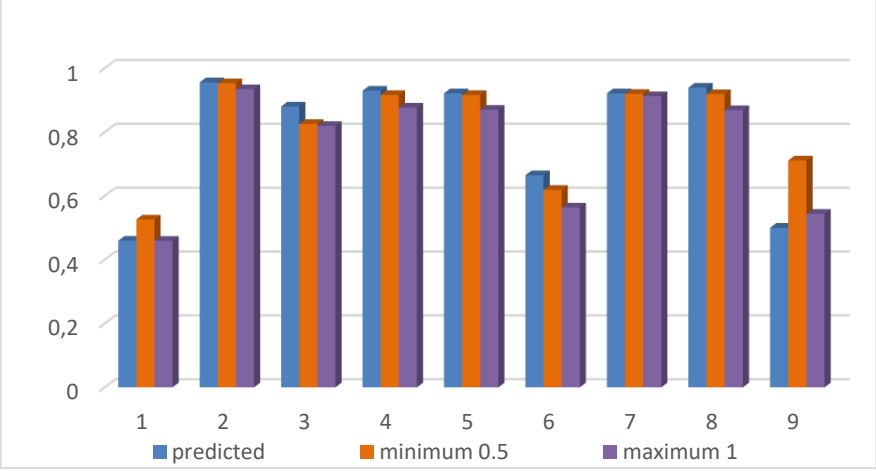


Figure C.39 Sensitivity of viscosity of material utilized efficiency on oil phase

Appendix C

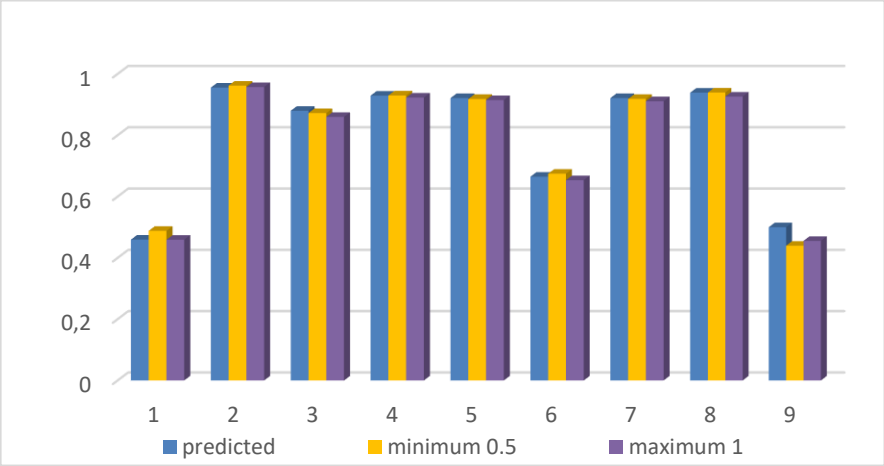


Figure C.40 Sensitivity of viscosity on time to raise temperature

Figure C.41 Sensitivity of pH on aqueous phase inlet

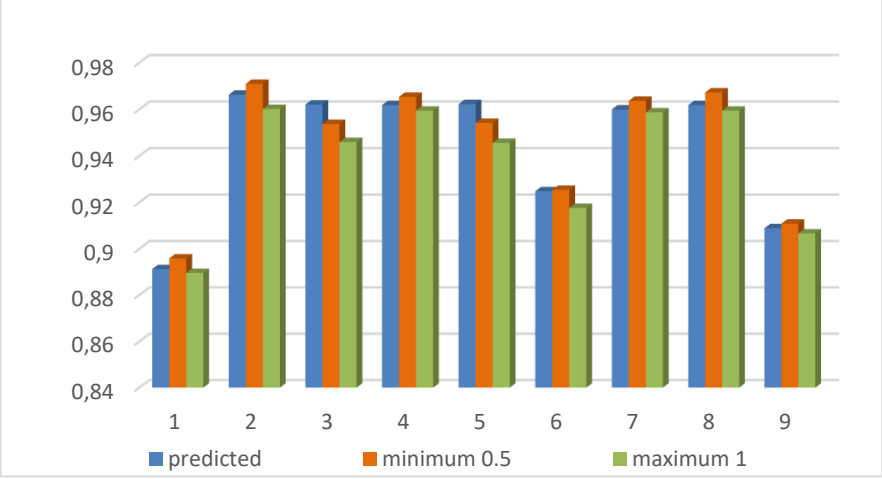


Figure C.42 Sensitivity of pH on aqueous phase oulet

Appendix C

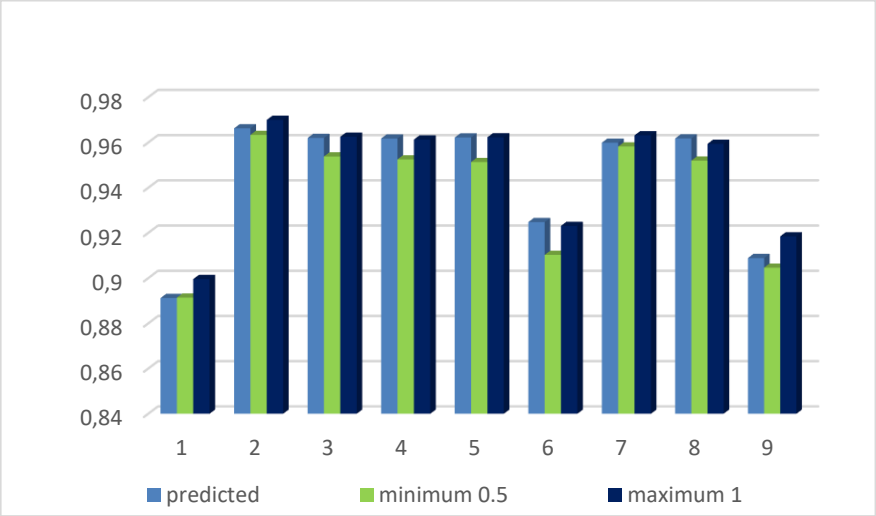


Figure C.43 Sensitivity of pH on water phase inlet

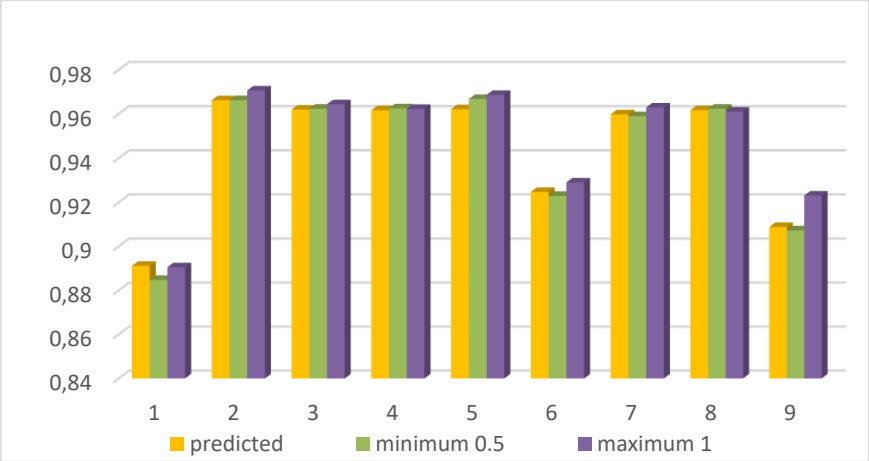


Figure C.44 Sensitivity of pH on oil phase inlet

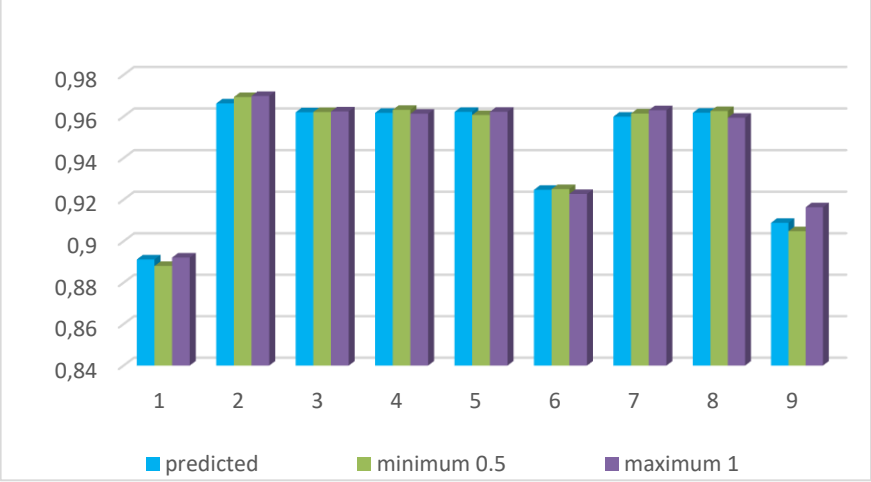


Figure C.45 Sensitivity of pH on sonolation time

Appendix C

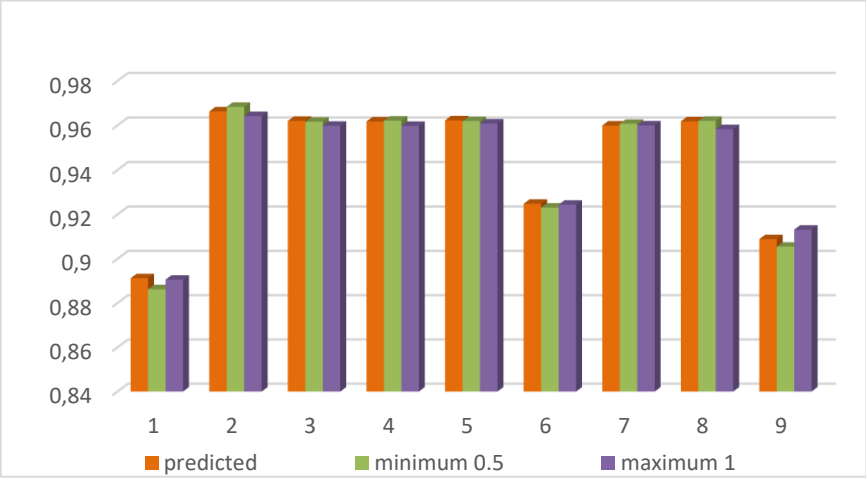


Figure C.46 Sensitivity of pH on time before sonolation

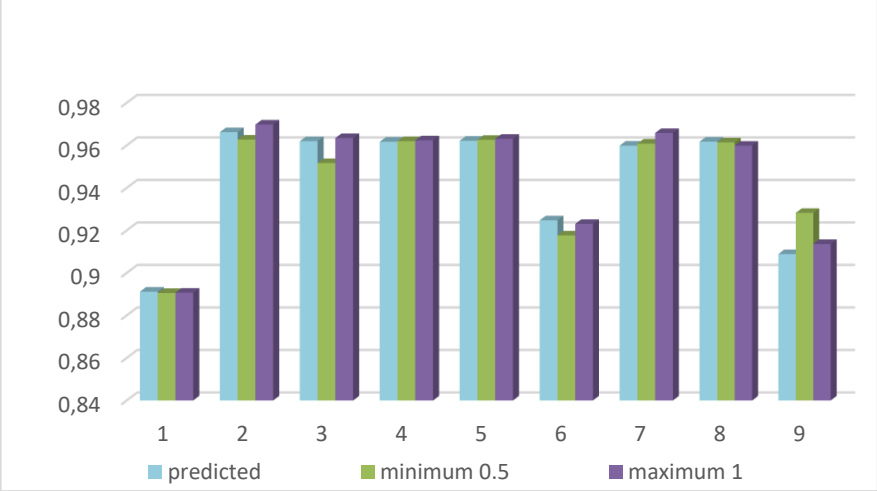


Figure C.47 Sensitivity of pH on addition of liquid ingredients

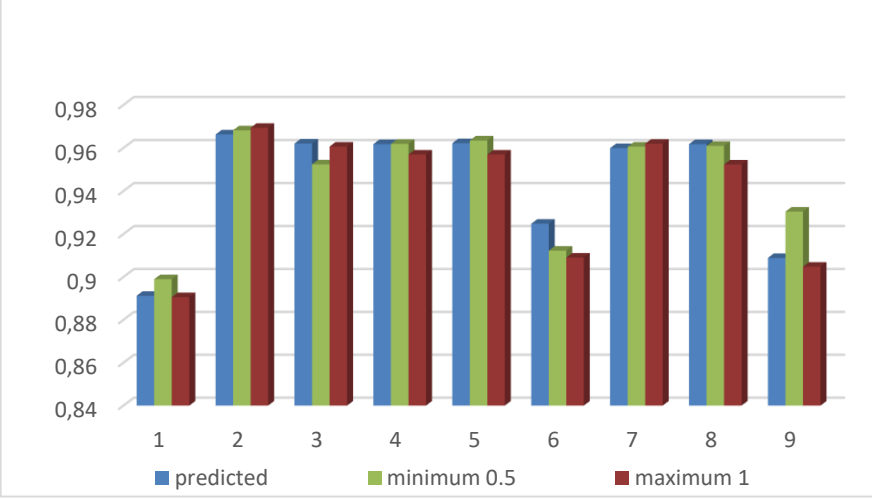


Figure C.48 Sensitivity of pH on total aqueous phase time

Appendix C

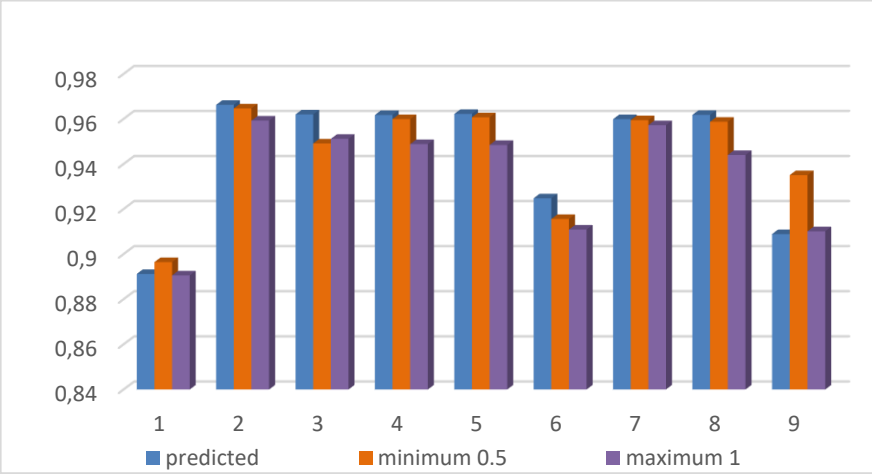


Figure C.49 Sensitivity of pH on oil phase time

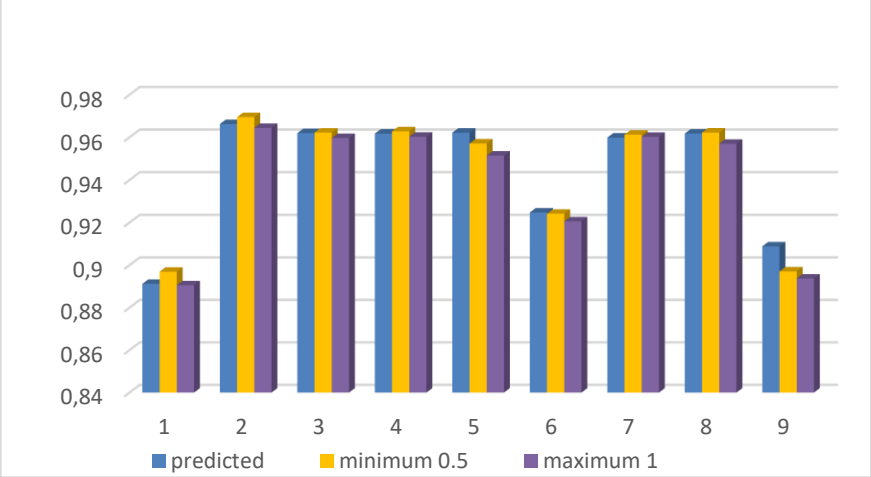


Figure C.50 Sensitivity of specific gravity on time to raise temperature