UNIVERSITY OF NAIROBI

An Artificial Neural Network Model for Predicting Retail Maize Prices In Kenya

By

Mayabi, Timothy Wamalwa

P52/85998/2016

A research project submitted for the partial fulfillment for the requirements for the award of Degree of

Masters in Computational Intelligence at the

School of Computing and Informatics

Of University of Nairobi

August 2019

Declaration

I hereby declare that this thesis is my own work and has, to the best of my knowledge, not been submitted to any other institution of higher learning.

Signature: ………………………………… Date: ………………………………

Name: Mayabi Timothy Wamalwa **Registration Number:** P52/85998/2016

This research project has been submitted for examination towards fulfilment for the award of degree of Masters in Computational Intelligence with my approval as the supervisor.

Name: Dr Lawrence Muchemi
School of Computing and Informatics
University Of Nairobi

Signature: ………………………………… Date: ………………………………

DEDICATION.

This research project is dedicated to my late parents Mr. and Mrs Luvisia Wamalwa.

ACKNOWLEDGEMENT

I would like to acknowledge God for giving me the opportunity to pursue my post graduate studies and mostly for provision throughout since the demise of my parents. I would also wish to thank my classmate Emmanuel Ahishakiye for the help. Most importantly patience and guidance from my supervisor Dr. Lawrence Muchemi and the whole panel of Drs and professors who took up the initiative to ensure that I do a proper research played a key role during research.

ABSTRACT

The intention of food price forecasting is to achieve reliability and usefulness for agricultural, non-policymakers, policy makers and agricultural related business. In the current globalization era, food security management in developing countries like Kenya that consider agriculture as a dominant economic activity require efficient and reliable food price forecasting models more than ever. Due to rare data availability and data time lag in developing agricultural dominated economies, normally needs reliance on time series forecasting models. Artificial Neural Network (ANN) modelling methodology gives a possible potential price forecasting method in developing countries based on available data. This study demonstrated the superiority of ANN over linear model methodology based on Root Mean Squared Error (RMSE) and Mean Absolute Deviation (MAD) performance metrics using monthly retail price (real prices) series of maize in three major counties. I.e. Kisumu, Nairobi and Eldoret. This study also portrayed the superiority of the ANN model in its univariate form over its multivariate form based on MAD and RMSE performance metrics. Lower comparative RMSE value would imply a better prediction while results with lower MAD were more close to actual values. Based on empirical study, showed that an ANN model is able to capture adequate number of directions of monthly price fluctuations as compared to models that employ the linear approach. It has also been observed that feeding the model with lagged observation of the same variable (univariate form) leads to more accurate forecasts than its performance in its multivariate form (Feeding it with different variables). Most models reviewed during this study, showed little effort in development of research tools, therefore this study has purposed to develop a user- friendly ANN prototype based on the proposed model.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

AI	Artificial Intelligence, Artificial Intelligence
ANN	Artificial Neural Network
ARIMA	Auto-regressive Integrated Moving Average
CRISP	Cross Industry Standard Process
DM	Data Mining
FAO	Food and Agriculture Organization
GB	Gigabytes
GDP	Gross Domestic Produc
IBM	International Business Machines
IDE	Integrated Development Environment
JSF	Java Server Faces
KEBS	Kenya Bureau of Standards
MAD	Mean Absolute Deviation
MAPE	Mean Absolute Percentage Error
NCPB	National Cereals and Produce Board
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
UI	User Interface
US$	United States Dollars
VCS	Version Control System

CHAPTER 1

INTRODUCTION

1.1. Background

Previous research in the area of agriculture and food security in Kenya, came up with consumption figures of maize in that provided computational foundation of food balance sheets and formulated estimations in regards to import and export prices of cereals. (T.S, Stephen, & Joshua, 2010), reported that maize accounts for up to about 29 percent of total farm yield from the small-scale farming part. This is because of its importance in Kenya's economy based on agricultural production trends.

Policy makers in Kenya have been faced by the typical predicament in relation to food prices. There are circumstances whereby policy makers are under duress to ensure that enough motivation is given to maize farmers to produce and market the crop. In another situation, there is need to maintain maize prices at affordable levels in both rural and urban areas (James K. Nyoro). Many years have seen policy makers attempting to achieve an equilibrium between these two competing goals – how to ensure enough returns for domestic maize production while keeping prices as low as possible for consumers. Various organizations and agricultural research institutions such as 'Tegemeo Letu' often have discussions over the role of government in shaping policies around maize production. They have also had other debates touching on trade barriers and how it affects returns of various cash crops. The government has often conducted its business (pricing and income transfer policies) around maize through (a) purchase and sell prices set by the administration at the National Cereals and Produce Board (NCPB) and (b) implementing import tariff barriers on external maize trade. Existing scrutiny as reported by previous researchers on the effects of the NCPB's activities, and government maize trade policy and how it affects maize prices and variations in maize prices indicate that there is controversy and lack of transparency in the information shared by institutions mandated by the government to implement policies for the benefit of maize producers and consumers in Kenya. As reported by (Joshua, T.S, & Stephen, 2010), Kenya has the highest prices of maize in the Eastern and Southern Africa region.

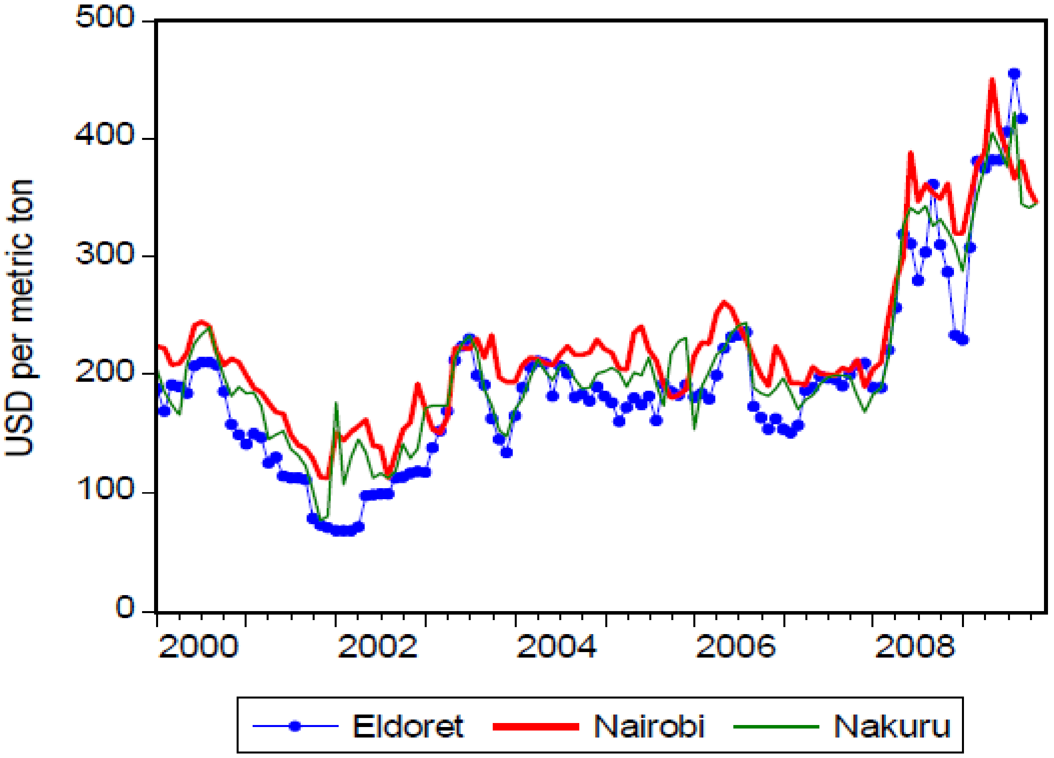


Figure 1 - Wholesale maize prices, nominal USD per metric ton, 2000-2009

Typical wisdom in Kenya blames NCPB for the high maize prices, mainly for benefitting the large and politically well-connected farmers but there is little rigorous analysis to support this. Maize is therefore termed as a political crop by most Kenyans. Previous research by (Jayne, Robert, & James, 2005) reported that maize is an important commodity in the Kenyan market, therefore they used empirical research on the historical effects on NCPB's activities in shaping the wholesale price movements of maize. Therefore this study will also make use of empirical research with an aim of answering the identified research questions. This study proposes a review of trends in price levels and forces that shape the trends as a good starting point in formulating a meaningful discussion about alternative food price policy. To further give more insight during an alternative policy formulation, the study proposes an inclusion of forecasted trends in formulating policies as opposed to the use of past and current trends only, with an aim of reducing the ramifications experienced as a result of volatile maize prices for both policy and non-policy makers. In summary price fluctuations have ramifications for every individual across the country. Non policy makers also need to be empowered to enable them make prior adjustments involving maize and related maize products expenditure based on the predicted prices. The current popular methods of technical analysis (using trends for guessing future maize prices) that are used by non–policy makers have no pointers to the exact wholesale price for maize. A system that can guide on the most likely price for the following year and hence assist in policy formulation (for policy makers) and expenditure adjustments (for non-policy makers) is therefore missing and necessary. It is for this reason that there is need to develop an artificial intelligence (AI) model that can be developed into a tool that can be used by both maize consumers and maize policy makers. Such a tool should not only show the price trends but also the most probable wholesale maize price for the following month. The AI system can be able to provide long term expenditure plans on maize by providing price information for future years. Such an AI system can be based on neural networks, which are computer algorithms formulated using specific AI rules to learn from data and then be used for tasks such as prediction. Using trending as a technical analysis, for basis of prediction is a tool that has already been employed globally through the FAO (Food and Agriculture Organization) website. The website provides information for different agricultural products for trending future prices. Research has also been done on other markets where prediction on an agricultural commodity has been attempted. (Li et al., 2010) developed a short-term price forecasting ANN model for Agro – products in China achieving an average accuracy of 95% (Mir & Yaser, 2015) developed an ANN model for predicting price of milk in Iran and concluded that neural networks have a higher level of prediction than ARIMA (Auto-regressive Integrated Moving Average) models.

1.2. Problem statement

The general perception as reported by (T.S, Stephen, & Joshua, 2010) is that policy makers and other sectors of the society about food security in Kenya, is similar to maize security. Therefore maize security is used as the significant parameter in determining the food situation in the country. Previous research by (T.S, Stephen, & Joshua, 2010) indicated that maize is the most common crop grown by rural poor households for food. Their research also concluded that the importance attached to maize by policy-makers in Kenya can be deduced from the emphasis laid on maize in current and past national food policies. This means that volatility of food prices has been a problem in Kenya and such fluctuations have ramifications for every individual in Kenya. Currently non-policy makers rely on their experience, technical analysis and fundamental analysis when buying and selling maize. These methods are subjective and misleading since they are not backed by actual figures. This study did not come across any artificially intelligent predictive tool for future maize prices in Kenya. There are some tools that provide information on maize price trends, however they do not have predictive mechanisms for future maize prices in Kenya. These tools provide information that point to the use of fundamental and technical analysis methods as being their basis of prediction of future wholesale maize prices. The tools neither show trends in future wholesale maize prices nor shows the actual figures of the most probable wholesale future maize prices. It is therefore desirable to have a tool that does not just point to the direction of price movement but also provides the most likely price value of whole sale maize itself. Non – policy makers that include the end consumers are the appropriate targets for such a tool since they are directly affected by volatile fluctuations in maize prices. AI methods that can actually analyze maize prices over time and gain intelligence then use this intelligence in prediction can be used to model such a tool. The predictive model shall provide information that will be a basis for consumers in making important decisions with regards to their expenditure on maize and maize related products.

1.3. Significance of study

Currently Kenyan consumers, mostly non-policy makers, use non-AI tools which may not be effective. Additionally these current methods do not have predictive abilities. The methods mainly provide trends but not the most likely price of maize for the following month. Due to the lack of AI tools, Kenyan farmers and consumers have no option but to use their own intuition in determining the most likely prices of maize and therefore make necessary preparations in response to the anticipated outcome. By using AI models to develop tools that can also help food policy researchers' advice food policy makers in Kenya, consumers and farmers will be able to make decisions from an informed point based on actual data provided to them.

1.4. Research objective

The research project had the following four specific objectives:

1. Investigate artificial intelligence models that are capable of food price prediction as a basis of designing a model for the Kenyan market.
2. Design an AI based model for use by both Kenyan non-policy makers and policy makers in predicting prices of maize.
3. Develop a prototype based on the design of the model.
4. Test and evaluate the performance of the model in predicting maize prices based on obtained data from various sources compiled.

1.5. Research questions

1. Are there artificial intelligent models that are capable of food price prediction that can be used as a basis for designing a model for the Kenyan market?
2. Will the proposed model be reliable for use by both Kenyan non-policy and policy makers?

1.6. Assumptions

The research made some assumptions as the basis of the study. The first assumption was that, historical monthly maize price data acquired for conducting the research has already been shaped by political activities at that time. This is because politics usually play a major role in Kenya's inflation rate. The second assumption was that maize data used in this research has already been altered by the informal, unrecorded cross-border agricultural trade activities between Kenya and her neighbors. The third assumption was that despite the maize sector's liberalization that enabled the private entities to participate in the maize market, the ministry of agriculture through NCPB remains to be the main player in the buying and selling of maize. The fourth assumption was that maize price data has already been shaped by purchase and usage of agricultural facilities such as fertilizer, pesticides and irrigation schemes. The final assumption was that the predicted outcome is independent of possible natural calamities like floods and drought.

1.7. Limitations.

Maize prices in Kenya vary depending on the region where it is being purchased and sold. The data from various regions could be used in this study. However, to enable data consistency, the study was limited to Nairobi, Eldoret and Kisumu counties.

1.8. Structure of the project

The first chapter introduced the topic of the Kenyan maize market. The chapter also gave the significance for the use of artificial intelligence models to build tools that both policy and non- policy makers can use as a basis of making important decisions in terms of maize production and maize trade in Kenya.

The second chapter provides a detailed review of what is happening in the sectors of maize production, trade, and consumption with an aim of linking the field of AI and agriculture. It also provides insights into research done in agricultural products prediction, the gaps identified and how predictive models can be constructed.

The third chapter provides details on the methodology that was used to design the model, develop the prototype and the performance metrics used to evaluate the model through experimentation. Data collection sources are also presented.

The fourth chapter provides the results and then reviews, interprets and analyzes these results. The final chapter summarizes the research findings and suggests areas of further study.

CHAPTER 2

LITERATURE REVIEW

2.1. Introduction.

Humans have always been and will always be curious about the future. A good example which is familiar especially in Kenya is prediction of a soccer game outcome through betting platforms like sport-pesa, m-cheza etc. The good advantage about this curiosity is the fact that there are possibilities of huge rewards in terms of money. Other examples include; predicting an upcoming election, and more complex scenarios like what the fate of our planet and the universe will be in the next couple of billion years. (Forecasting, n.d.) defines forecasting as the process of making predictions based on previous and current data and by analyzing data trends patterns. A famous example of a successful prediction is by the German astronomer Johann Gottfried Galle, who predicted the existence of the previously unknown planet Neptune by calculations based on Sir Isaac Newton's law of gravity. Other predictions have become famous because of their extreme lack of accuracy. IBM chairman, Thomas watson made a prediction in 1943 there would be a world market for approximately five computers. The last example highlights the value of predictions in economic situations. If a company always knew everything about the coming market trends and the company products, it would of course be easier to adapt to the market and optimize the company's strategy. Even if only partial information were available, it would be of high value. The basis for the purchase and sale of products largely depends on price forecasting. Previous research by (Girish & Kanchan, 2013) reported that assessable validity while taking into consideration the minor errors involved, together with decisive prediction power is key when assessing prediction models largely due to the variations of food production and prices as a result of unanticipated eventualities like floods, drought, and diseases. This leads to a substantial risk and precariousness during price modelling and forecasting process. Based on a report obtained from KEBS website, food is at the top of basic human need, Food prices play an important role in consumers' access to food as it has a direct impact to low income earners who will most likely spend a big percentage of their income on food.

Broad Commodity Group	CPI Weight	Weight % Change on last month (April 2017/ March 2017)	% Change on same month of previous year (April 2017/ April 2016)
Food & Non-Alcoholic Beverages	36.04	3.55	20.98
Alcoholic Beverages, Tobacco & Narcotics	2.06	0.04	3.26
Clothing & Footwear	7.43	0.04	4.01
Furnishings, Household Equipment and Routine Household Maintenance	18.3	0.63	2.94
Health	3.13	-0.19	3.05
Transport	8.66	-0.22	5.11
Communication	3.82	0.00	0.10
Recreation & Culture	2.25	0.03	1.99
Education	3.14	0.00	2.85
Restaurant & Hotels	4.48	1.10	5.65
Miscellaneous Goods & Services	4.52	0.11	3.55
Total	**100**	**1.79**	**11.48**

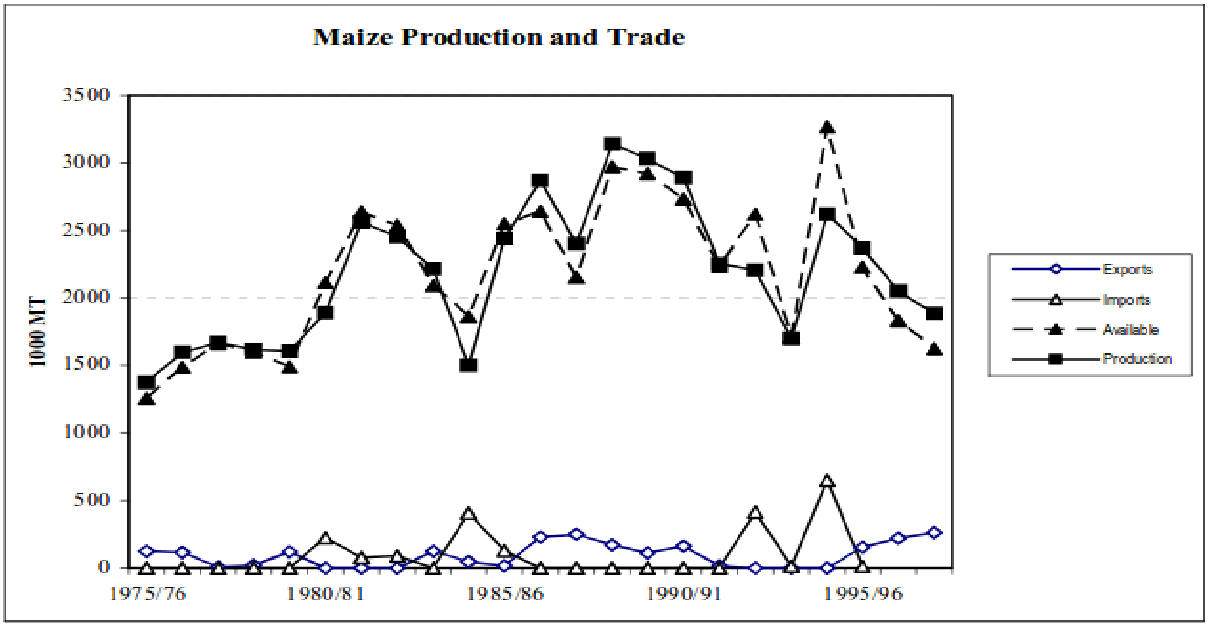
Table 1 Commodity prices and CPI weight

A good forecast model is thus defined by its reliability, ease of use, having an output that is meaningful, compatibility with other systems, timeliness of the forecast and reliable accuracy (Arienda & Asana, 2015).

2.2. Maize market in Kenya

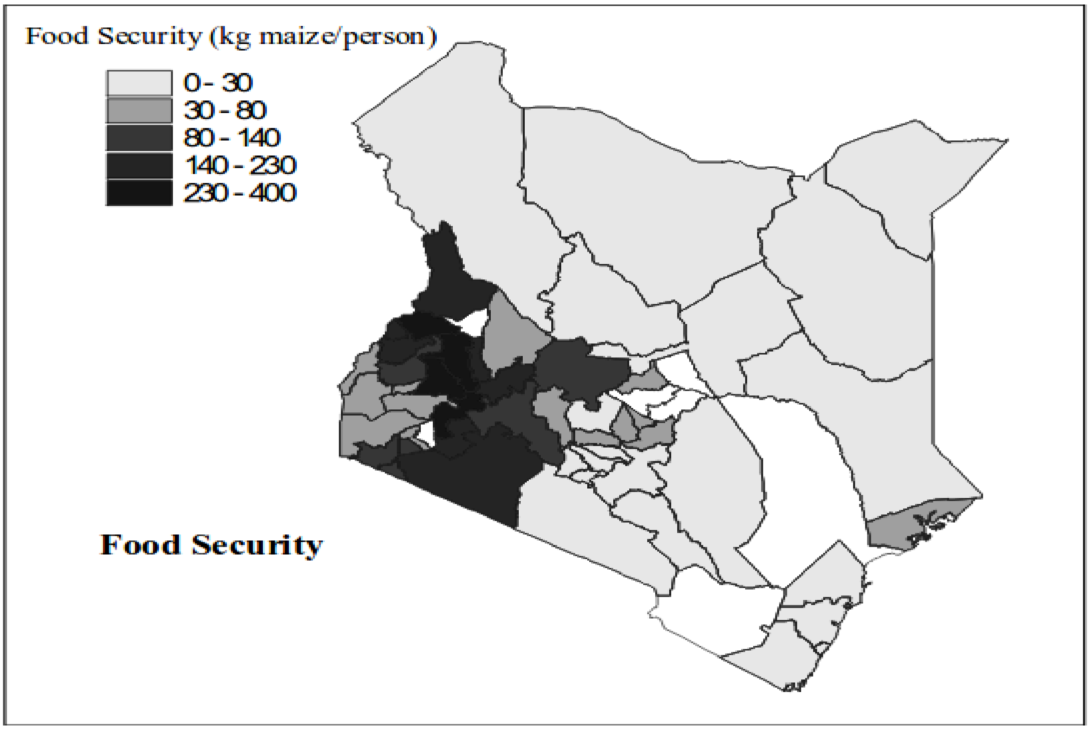
Maize is the most consumed crop in Kenya which results to the high demand in both rural and urban areas. This means that maize is at the core position in the diet of Kenyan people. Previous research by agricultural organizations reported that the variations in maize prices have been a cause for concern because it adversely affects both the producers and consumers. Weather induced volatility has also impacted the production patterns of maize. (Shawiza, 2016) ,reported that maize is the key food crop in Kenya, constituting 3 percent of Kenya's Gross Domestic Product (GDP), 12 percent of the agricultural GDP and 21 percent of the total value of primary agricultural commodities. Researchers in the agricultural field have classified maize into both subsistence and commercial crop, it is estimated that the crop occupies about 1.4 million hectares and it is grown by both large-scale and small scale farmers. Figure 2 shows a summary of maize production activities between the years 1988 and 1998. Production was 2.3 million metric tons varying from 1.7 million metric tons during the period 1993-1994 to 3.14 million tons between the periods 1988-1989 ((Kenya, 1998); (Argwings, 1998)). Previous researchers have estimated that, about 40% of maize produced in Kenya is marketed while the rest is used for subsistence. Figure 3 shows the main maize surplus and deficit districts of Kenya. A research report by (Wilson, 2016) indicated that the main maize production areas are in, Nakuru, Nandi, Kericho, Uasin-Gishu and Trans Nzoia. These areas are a major contributor to the activities of maize market in Kenya. The same report also showed that most counties involved in maize production and marketing were found in Western, Nyanza and parts of Eastern Provinces. Figure 3 highlights the maize production areas in Kenya as of 1998. Prior to liberalization the Kenyan government was actively involved in controlling all aspects of maize marketing. This liberalization was as a result of a major policy shift from 1986 to 1995. Countries in the Eastern and Southern of African that had strictly regulated the marketing of maize were largely affected by policy shift. Therefore, corporations in these states that controlled the marketing of maize were reduced to traders.

Figure 2 - Maize production and trade in Kenya 1976-1996



Source: Government of Kenya, Statistical Abstracts 1976-1998.

Figure 3 - Maize Surplus and Deficit Districts of Kenya, 1998



Source: Authors' design using data from Government of Kenya, Statistical Abstracts 1998.

2.3. Artificial intelligence

Artificial intelligence in the 21st century has made research prevalent in many fields such as engineering, education, health, mining, stock market, law and economics, among others. A lot of work has been done in the field of AI to the extent that one is not able to keep track of each and every AI related innovation in whichever area is being adopted. Therefore this has brought about the clustering of AI into various areas with each area contributing to various fields of knowledge. Examples of AI clusters include; Machine learning, Image processing, Data mining and Natural language processing (NLP). These AI constituents have become very important in almost every field in our day to day tasks. Automotive industries have actively used machine learning especially in the areas of vehicle maneuvering and safety characteristics. Daimler benz have made use of the sensotronic braking system (SBC), the system has the ability to learn the behavior of a driver's braking patterns based on the force applied on the brake pedal. The pedal is associated with several sensors which transmit important information such as road traction, speed of the vehicle, tyre pressure etc to the SBC. These parameters serve as input variables which are used by the system to learn and eventually make various decisions based on how it was trained by the manufacturer. Similarly other vehicle manufacturers like Nissan, Jeep, Toyota and manufacturers for high performance sports vehicles have made use of machine learning in the areas of passenger safety, fuel economy, comfort systems and vehicle stability. In the business and financial trading world, machine learning has extensively been used for classification and forecasting purposes. A typical classification scenario is where by a business is segmenting clients based on various parameters that form patterns that in turn help in decision making. This is a good foundation that will enable a business to come up with a decision support system with machine learning capabilities. Such a system will assist the business to allocate resources from an informed point of view based on the identified focus areas. Therefore there is a reduced rate of blind investments which often results in unnecessary loss of time and money. As for financial trading, companies have made the use of statistical models that make use of past occurrences (trends) to forecast the performance of stock in various markets. Governments are also making use of these models for the purposes of planning and resource allocation. The recent rise of cryptocurrencies such a s bitcoin, etherium, litcoin etc, have necessitated the use of statistical models with machine learning abilities. For example, bitcoin miners have made the use of such models in determining where, when and which coin to mine in order to save on electricity power and computational resources. Image recognition has mostly been used in areas/industries that offer security services. Face and thumb recognition has largely been adopted for purposes of access control. A famous example is on our mobile phone devices whereby fingerprint detection is used for access to one’s mobile phone device. Other examples include, Facebook's face recognition feature that suggests to you names of people you should tag when you upload a photo with several images. Google’s tensor flow deep learning framework provides a good development platform with various algorithms that make it easy for one to develop or implement an image recognition feature in an application. Other deep learning frameworks include; PyTorch, Neuroph studio, Sonnet, Keras, MXNET, Gluon,Swift, Chainer, DL4J and ONNX. Need for identification of patterns and relationships through data mining has also led to extensive research in this area. Financial companies with huge data are extensively researching with the aim of maximizing profits using these data sets. Health awareness is also being driven by data collected over a long period of time. With these, researchers in the health industry are able to identify patterns and provide recommendations based on their findings. Therefore data is being equated to as ‘Gold’ because of its importance in virtually all areas of human life. Research in Natural Language Processing (NLP) has really assisted in globalization of many areas as a result of market integration as well as cultural/societal integration. People around the world can comfortably communicate using devices with NLP capabilities that are able to do various language translations. Security issues such as terrorism has immensely reduced due to the extensive research done on sentiment analysis. Things to do with hate speech and abuse on online platforms can easily be flagged and necessary actions taken based on the laid down policies and procedures as far as the two are concerned. However many researchers have acknowledged that a lot of work still needs to be done in the area of sentiment analysis due to the diverse nature of languages and the context in which certain statements are applied. In almost every language there is a slang occurrence and the context in which certain slang words are applied is different depending on the area where it is being spoken. Therefore slang is not standard and the changes are very frequent, this means that more work needs to be done to accommodate such changes. Text analytics has also made it possible for people to do financial transactions through chat bots such as telegram and whatsapp. Nowadays, a developer can train a bot to learn things based on the domain in which it is being applied. The health industry also makes use of AI system with text analytics that assist in diagnosis and decision making.

2.4. Agricultural prediction and time series modelling

A lot of research has been done in Kenya in the area of Agriculture. However much of it has been about the effects and impacts of various parameters on Agriculture. Not much work has been done in the area of price prediction in this field. In other countries, research in the area of agricultural price prediction have resulted to various conclusions in relation to market prices of agricultural products. Among the conclusions is that agricultural products portray certain features that are totally different from non-agricultural products. These features are derived from both natural and market variations. They include; seasonality of production, product demand and product prices. Therefore, statistical modelling of agricultural products is different from non-agricultural products. In another conclusion, some researchers argued that information as a result of market hypotheses should lay the foundation for reliable forecasts. Any other information would make the forecast unreliable. Another one argued that the use of future prices is not a partial predictor to the existing prices. The following are various conclusions by researchers as far as prediction is concerned.

Gardner (1976) proposed a reflection of the market's estimate of subsequent period cash prices using a future market prices. Rauser & Just (1981) described that forecasts were generally conflicting to the related future market prices (Khalid, Sultana, & Zaidi, 2014).

Garcia, Hudson, and Waller (1988) suggested a having a mixture of parameters based on evidence regarding the pricing efficiency of agricultural forecasting models and if they can improve the forecast performance. (Khalid, Sultana, & Zaidi, 2014). Previous work by researchers portrayed the preference of the use of models when it comes to prediction of future market prices. (Bessler & Brandt, 1992) used a vector auto regression model and reported that future prices of cattle resulted to inefficient forecasts compared to actual cash prices. Other researchers like Irwin, Gerlaw and Liu (1992) reported differently by saying that they did not find many significant variations between the US department of agriculture and the forecast accuracy of live cattle futures prices and live hog. Therefore, based on literature review it is possible to make use of forecasting models in the field of Agriculture. However, most researchers who have developed forecasting models and applied them in the field of agriculture, really emphasize on data availability and consistency. Some researchers like Zulauf and Irwin (1997) suggest that data obtained from an efficient market is more likely to be consistent with individual-generated forecasts.

In their study (Girish & Kanchan, 2013) identified structural and time series models as the main ways of doing forecasts. The fundamental principle of structural models is that of consumer and producer theory with an objective of identifying the relationship between prices and the underlying demand and supply functions. They also reported that structural modelling methods provided informative insights about factors affecting movements of prices. They therefore concluded that developing countries don't have the ability to adopt structural models due to the routine nature of data in these countries. This means that researchers in these countries will have to rely on sparing price related data for use in their forecasting models. Previous research has shown that time series models are able to make efficient use of less data for efficient predictions. Therefore, time series is the analysis of past observations of the same variable with an aim of understanding the underlying relationships. The use of the box jenkins methodology has made Autoregressive Integrated Moving Average (ARIMA) a famous time series model.

In recent times the use of Artificial Neural Network (ANN) modelling has become popular as it adds to the many existing choices of forecasting techniques. As reported by (Girish & Kanchan, 2013) the universal functional approximator as the main advantage of ANN. Therefore with ANN, there is no need to manipulate data in order to conform to a particular model occurrence for a given data set. Therefore this study established that time series models are only limited to one type of data (univariate) that exhibits linearity for purposes of making predictions. On the other hand ANN models are able to handle both linear and non-linear data whether in univariate or multivariate forms.

2.5. Artificial neural network

The original works of Artificial neural networks (ANN) attempted to make machines work like the human brain. ANN is capable of acquiring information, store this information and develop knowledge using this information. The knowledge is as a result of experience gained in the domain within which the ANN system operates in. Modifications to the synaptic connections between the neurons is a fundamental process that takes place during learning in human beings.

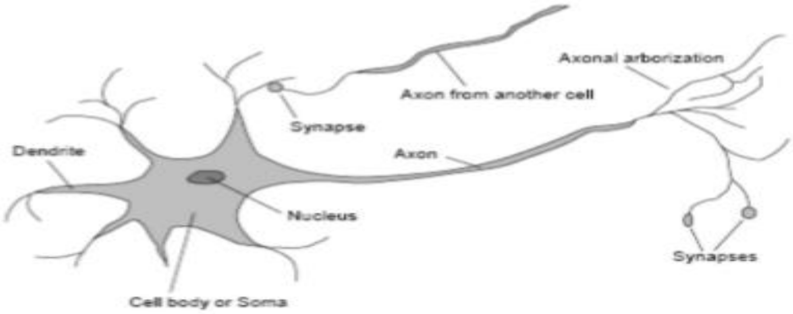


Figure 4 - Representation of human neuron (Source: Bethard (2008)).

2.5.1. A single neuron

The neuron is the fundamental unit of computation in an artificial neural network, it is often referred to as a node or unit. The output is as a result of computation based on inputs from other nodes. Each input is associated with a weight (w), which is randomly assigned. The node applies a function *f* to the weighted sum of its inputs as shown in Figure 5 (ujjwalkarn, 2016)

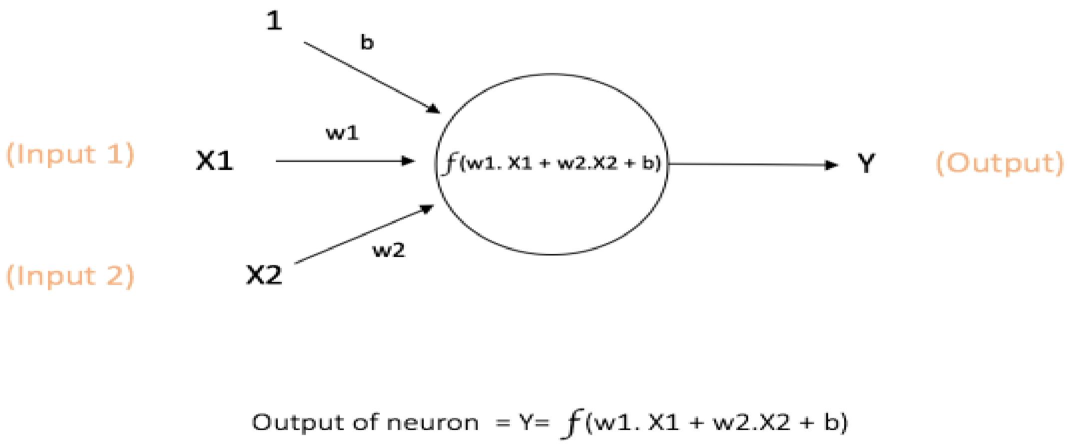


Figure 5 - Representation of a single neuron (Source: Author).

There is also another input 1 with weight b (called the **Bias**) connected with it.

Figure 1 shows the result (Output Y) based on the computation from the neuron. The function *f* is non-linear and is referred to as the activation function. The resultant non-linearity attribute on the output of a neuron is due to the workings of the activation function. This is important because most real world data is nonlinear and it is important that neurons are able to process these nonlinear representations.

Figure 6 shows the workings of the activation function.

Some of the activation functions include:

a) **Sigmoid:** takes a real-valued input and transforms it it to a range between 0 and 1

$$\alpha(x) = 1/(1 + \exp(\text{-}x))$$

Equation 1 Sigmoid function

b) **tanh:** takes a real-valued input and transforms it to the range [-1, 1]

$$tanh(x) = 2\sigma(2x) - 1$$

Equation 2 Tanh function

c) **ReLU**: ReLU stands for Rectified Linear Unit. It takes a real-valued input and replaces negative values with zeros

$$f(x) = \max(0, x)$$

Equation 3 Rehlu function

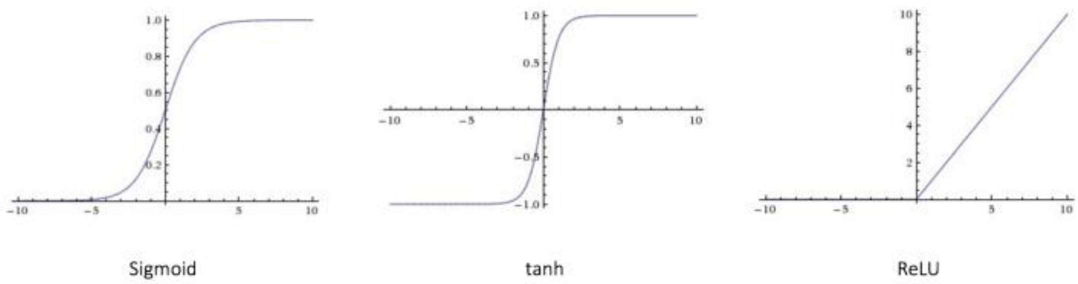


Figure 6 - Activation functions (Source: Author).

A trainable non-changing value is obtained at each node due to the workings of the bias. This is also in addition to the normal inputs that the node receives. (GeeksForGeeks, n.d.)

2.5.2. Feed Forward Neural Network

The feed forward neural network ensures that information moves in the forward direction. Figure 7. This network consists of layers that has a number of nodes which have connections with nodes from the adjacent layer.

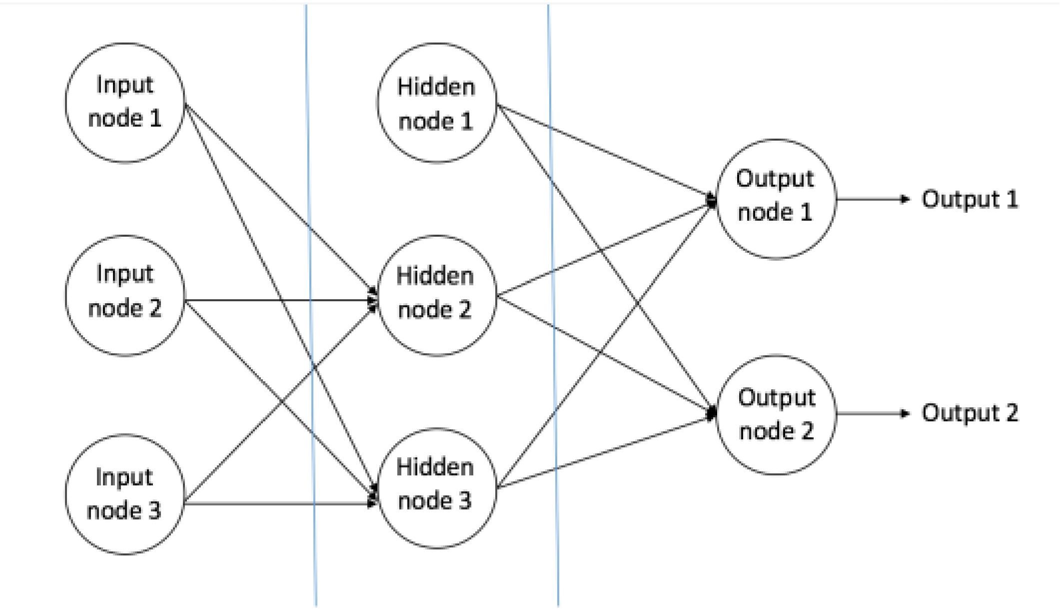


Figure 7 - Feed forward neural network.

Constituents of a feed forward include:

1. **Input Nodes –** This constitutes to what is referred to as the Input layer as they are responsible for providing information from the outside world to the network..
2. **Hidden Nodes –** The Hidden nodes are abstracted from the outside world. They function to pass information from the input neurons to the output neurons. They constitute to what is referred to as the hidden layer. A feed forward network can have a one input and output layer, it can also have zero or many hidden Layers.
3. **Output Nodes –** The Output nodes constitute to what is are referred to as the output layer and function to transfer information from the hidden layers to the eternal environment.

Examples include:

1. **Single Layer Perceptron** – A feed forward neural network with no hidden layer.
2. **Multi-Layer Perceptron** – Has zero to multiple hidden layers.

2.5.3. The multilayer perceptron

The MLP (Multilayer perceptron) includes an input layer (responsible for receiving input values), an output layer and at least one hidden layer. Each layer consists of a set of nodes where by inputs of the hidden layer are gotten from come from units of previous adjacent layer. The output is sent inputs of the next layer. Learning algorithm is implemented during the training process. Flow of information is represented by the input and output layer. Back propagation is the learning algorithm used by the MLP, which is basically a gradient technique. Certain variants of the algorithm have also been implemented to deal with the problem of slow convergence. Upon trained process, the network weights are halted and can be used to compute output values for new input samples.

The network learning is a process in which the weights, ω, are adapted by a continuous interaction (k) with the environment.

$$w_{nj}(k+1) = w_{nj}(k) + \Delta w_{nj}(k)$$

Equation 4 Network learning process with weight adjustments

$\omega(k)$ = previous value of the weight vector.

$\omega(k+1)$ = updated result from the previous weight vector.

$w_{nj}(k)$ = determined by a set of rules to solve the learning problem.

Error correction is also an important algorithm to consider. See the n-th neuron in the iteration.

y_n = neuron response.

$w_{nj}(k)$ = environment stimuli vector.

$\{x(k), k_n(k)\}$ = training pair.

The resultant error signal will be:

$$e_n(k) = d_n(k) - y_n(k)$$

Equation 5 Signal error

Minimizing the cost function is the goal, therefore this error will be considered. Once the criteria is selected, the error correction scenario develops into an optimization problem. A differentiation function of a weight vector e.g. function$\epsilon(\omega)$. The changes the elements from w to real numbers. Therefore it is necessary to find an optimal solution ω^* that satisfies such a condition:

$$\epsilon(\omega *) < \epsilon(\omega)$$

Solving an optimization problem becomes possible without limitations such as: the cost function minimization e(ω) with respect to the weight vector. This is given by:

$$\epsilon(\omega *) < \epsilon(\omega)$$

Equation 6 Cost function minimization with respect to weigh

∇ = gradient operator.

Gradient descent forms the basis of an optimization algorithm without limitations. The first condition,$\omega(0)$ generates a sequence$\omega(1)$, (2) , . . ., causing the cost function $\epsilon(\omega)$to reduce at every descent. The desired convergence of the algorithm to get to an optimal solution will strive to meet the following equation 7.

$$\epsilon(\omega(k+1)) < \epsilon(\omega(k))$$

Equation 7 Cost function optimal solution

The weight vector is subjected to back-to-back adjustments in the direction of the gradient descent. This is represented as:

$$\mathrm{g} = \Delta\epsilon(\omega)$$

Equation 8 summarizes the gradient descent algorithm:

$$w(k+1) = w(k) - \eta g(k)$$

Equation 8 Gradient descent

η = learning rate.

$g(k)$ = gradient vector evaluated at$\omega(k)$.

Equation 9 summarizes the correlation applied to the weight vector.

$$\Delta w(k) = w(k+1) - w(k) = -\eta g(k)$$

Equation 9 Weight correction on gradient vector

The optimal solution w $*$ is achieved by having the method on a slow convergence. It is also important to note that the learning rate has adversely affects the process of convergence. The path of$\omega(k)$, w becomes smooth when η is small. The path of $\omega(k)$over w is oscillatory when η is large whereas the path $\omega(k)$ over the plane w becomes unstable as η goes beyond a certain critical value. Therefore back propagation algorithm is an approach taken to implement the method of gradient descent in a weight space for a network. The fundamental idea is to efficiently calculate the partial derivatives of an approximate function of the behavior by the neural network with respect to all elements of the adjustable vector of parameters ω for a given value of the input vector x.

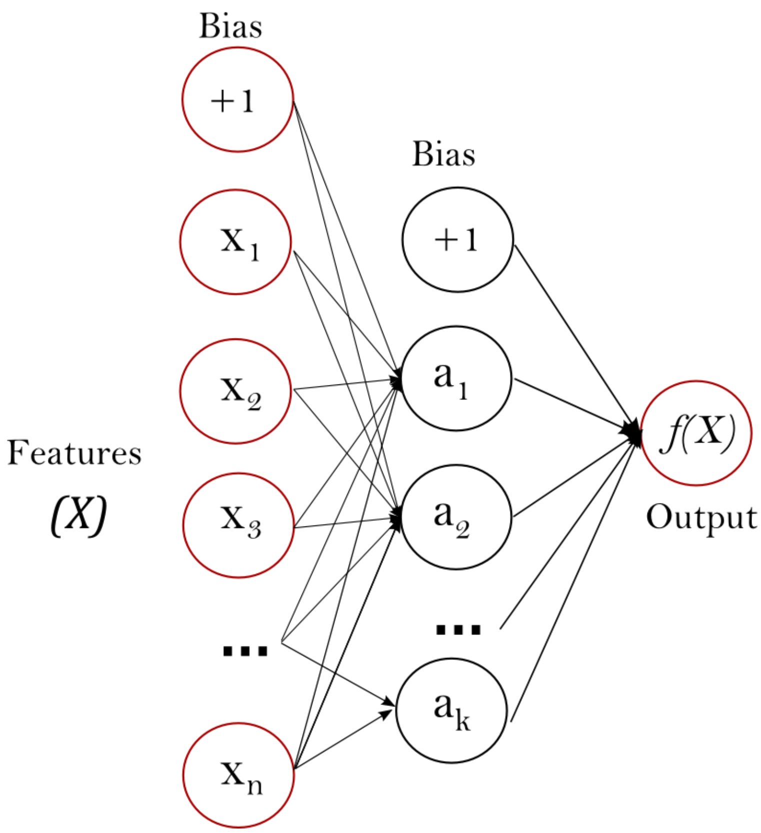


Figure 8 - Multilayer perceptron

2.6. Conceptual model

Review of the literature enabled the research to develop the conceptual model shown in figure 9.

This research is optic to predict the prices of retail maize prices using ANN. The aim of this paper is not to prove that ANN methods of prediction are best, but it is to show that they are applicable to daily life problems. This research also seeks to provide answers to the research questions raised in chapter one.

Transformation of time series data is possible if ANN is developed to have dynamic properties. Therefore functional representation of time is indirectly provided to the ANN model so as to model this data. Further literature review suggested that a neural network can be made dynamic by having it make use of memory in either short-term or long-term basis, depending on the preservation time, into the structure of a static network. Developing a short-term memory into the structure of a neural network is a good approach towards ensuring that a neural network has a time delay functionality. Previous research by (Girish & Kanchan, 2013) made use of Time-Delay Neural Network (TDNN).

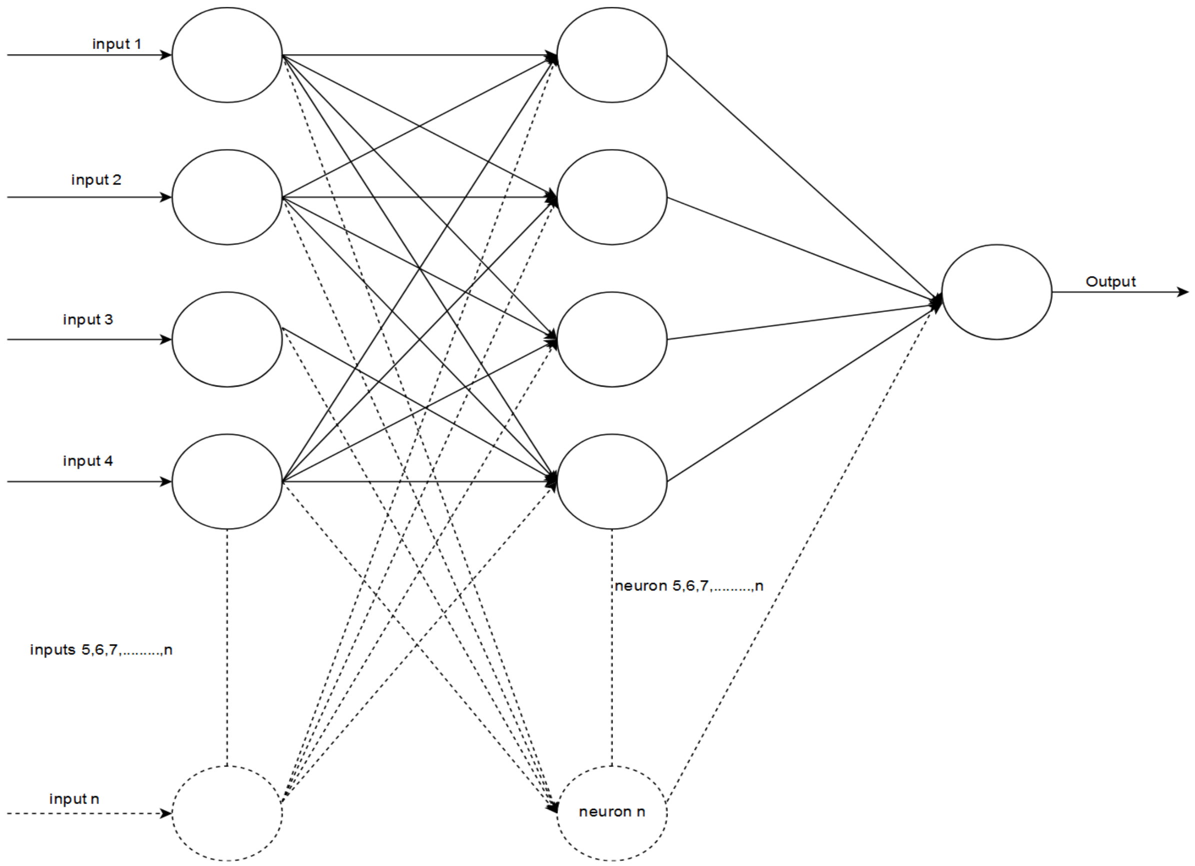


Figure 9 - Conceptual model (Source: Author)

2.7. Chapter summary

Chapter provides a detailed review of how forecasting has been previously used in other fields and what is happening in the sectors of maize production, trade, and consumption with an aim of linking the field of AI and agriculture. It also provides insights into research done in agricultural products prediction, the gaps identified and how predictive models can be constructed.

CHAPTER 3

RESEARCH METHODOLOGY.

3.1. Introduction.

Research is defined as a fact finding activity that involves a scientific investigation or thorough study of a given subject matter of particular interest. A research can be exploratory, descriptive or diagnostic in nature and thus qualitative or quantitative approaches are applied as per the research design. Research has been proved to be a vital tool that provides the basis for economic decision making by government institutions and policy makers (Mackey & M, 2013).

This project used both exploratory and applied research to create a specific artificial intelligence tool based on a model and tested its performance on a practical problem. The exploratory part of the research was attempting to identify new insights into the possibility of incorporating more diverse predictors for maize price forecasting. The predictors used are variables which were identified to be in the staple food category from the literature review. Chapter 2 indicated the need for such a tool that can be used by both policy and non-policy makers in the field of agriculture in Kenya. Therefore this research applied a particular underlying AI method to a practical field, in this case, the maize market prices in Nairobi, Eldoret and Kisumu counties. Applied research aims at solving a 'societal or business' problem (Kothari, 2004).

3.2. Research Design.

Research design is a work plan detailing how the research was be undertaken, type of data to be collected, tools and techniques that was be employed to obtain data and the method of data analysis to be used (Wyk, 2012). The research design adopted for the study was an exploratory- quantitative research approach (Mackey & M, 2013).

Exploratory research seeks to help identify new hidden factors that might be found in data and their significance to the study under research. This type of research approach creates a foundation for further future in-depth research from a scientific point of view with the intent of finding new insights and ideas about the topic under study. The exploratory design was adopted to investigate the performance of the neural network model when supplied with a lagged input of the same variable (historical monthly maize prices) over time compared to supplying it with different variables (prices of other staple foods, precipitation, inflation rate and maize production data) when predicting maize price of the next month. An AI model was designed based on the artificial neural network algorithm which was then developed into a working prototype for the purposes of testing. The model employed the use of a feed forward ANN with multilayer perceptron using back propagation and trained using supervised learning. The programming language environment was Java 8. Other java components included the java server faces (JSF) framework, neuroph library and primefaces library with ajax components to provide some user interface (UI) capabilities.

3.2.1. Research Data.

For purposes of designing and evaluating the model, the research needed historical monthly maize price data (data from Nairobi,Kisumu and Eldoret counties), country wide yearly price data for agricultural products identified as staple foods in this study and yearly maize production data.

3.2.2. Data Sources.

Data collection method was not experimental and included collection of secondary data from relevant publications and the internet. The Food and Agriculture Organization (FAO), a specialized agency of the United Nations that leads international efforts to defeat hunger provided monthly maize price data from Nairobi, kisumu and Eldoret counties through their website. Country wide yearly price and production was obtained from the knoema website, a free to use public and open data platform for users with interests in statistics and data analysis. Data was later imported into mysql database for ease of retrieval during model development and testing.

3.2.3. Data preparation.

This phase covers all activities that will be involved in constructing the final dataset (data that will be fed into the model) from the initial raw data. The step also involved deciding on the best method of data retrieval for the model and data partitioning for both univariate and multivariate scenarios of the model. Data partitioning involved segmenting the sample data into two sets for purposes of training and testing the model. The training set constituted the largest portion of the sample data and was used by the ANN model to learn the patterns present in data. 70% of the data was used as the training set while the remaining 20% was used as the testing set. Thus the sample data was partitioned as follows:

1. Training set = 70% of total sample size.
2. Testing set = 30% of total sample size.

The volume of data implied a period of twelve years for the monthly maize price data and twenty four years for the yearly price data of other staple foods. In other studies of agricultural products prediction, other researchers considered periods that varied from 10 years to 30 years. Girish and Kanchan(2013) did a test over a thirty year period while Shahriary and Mir(2015 did a test over 14 year period. Due to these considerations, the project decided on a twelve years for the monthly maize price data and twenty four years for the yearly price data of other staple foods.

3.2.4. Data evaluation.

Evaluation of data first involved the transformation of input and output variables to reduce noise. Plotting the data was the next step in order to understand the underlying relationships. Data transformation is crucial in ANNs for achieving a good prediction performance by removing the bias and correlations between the inputs and making them statistically independent (Oancea & Ciuncu, 2014). A study by (Nayak, Misra, & Behera, 2014) concluded that data transformation through normalization speeded up training time. They started by the training process for each label within the same scale. This is highly regarded when modelling an application where the inputs are on totally different scales. Decimal scaling was the preferred normalization approach which involved moving the decimal point of values of attribute A. A value v of A is normalized to v' by computing: v' = (v / 10 power j) where j is the smallest integer such that Max(|v'|)<1.

3.3. Designing the proposed model

The system design approach for developing the model was the CRISP-DM methodology .It incorporates six design phases that comprehensively cover the model development process. As used by (Cortez, 2010), the methodology works well with Artificial Neural Network for predictive purposes. The Figure 10 illustrates the development cycle of the methodology.

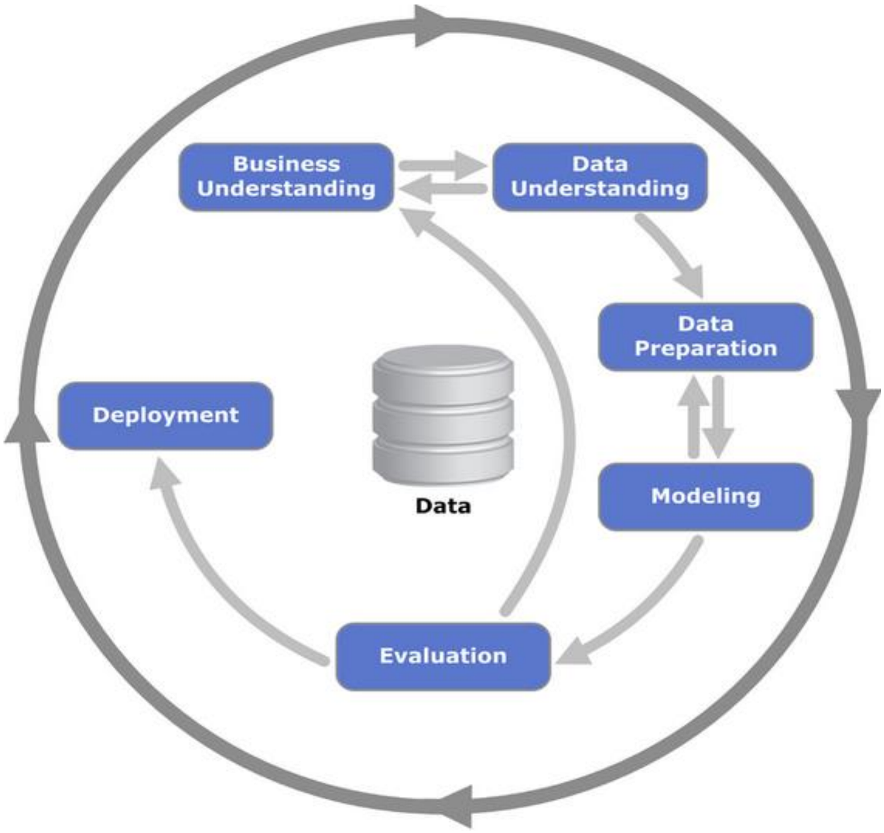


Figure 10 - CRISP DM (Source: Wirth & Hipp 2010)

The model design describes the organization of the ANN and defined the number of input neurons. The input neurons were used to capture each independent variable. Model design also involved determining the number of ANN hidden layers, the number of neurons in the hidden layer, number of output neurons and the ANN activation function. The first step was to be the formulation of a baseline model as per the proposals by other researchers. The third step would then be the development of the new proposed model based on the baseline model after change of parameters as determined by experimentation.

3.3.1. Formulating the baseline model

The model needed to be dynamic in a way that it can be trained in both univariate and multivariate modes. Therefore the model input varied depending on the training mode.
The two modes (univariate and multivariate) of the baseline model were developed for experimental purposes so as to evaluate the performance of the model and help in determining the best mode so as to come up with a new model. Based on other research,

the hidden neurons should be 1 layers (Girish & Kanchan, 2013). The first experiment was done on the model in its univariate mode using real prices of monthly maize data from 2006 to 2018. The baseline model, with a configuration of 4:8:1 was subjected to training using 70% data (Jan-2006 to Apr- 2018) and the balance for testing. The number of training repetitions was set at 50,000 maximum, after which the training was stopped. The results obtained for each of the monthly prices is shown in Appendix 7. The figure 4.2 below shows the graph of MAPE for the monthly price data. The second experiment was done on the model in its multivariate mode. The data used included yearly wheat price, rice price, rainfall, inflation rate and maize production data from 1992 to 2016. The baseline model, with a configuration of 5:8:1 was subjected to training using 70% data and the balance for testing. The number of training repetitions was set at 5,000 maximum, after which the training was stopped. The results obtained for each of the monthly prices is shown in Appendix 8. The figure 4.3 below shows the graph of MAPE for the monthly price data.

3.3.2. Univariate baseline model

In its univariate mode the model comprised of input of 4 monthly maize prices is used with the aim of predicting the 5th price in the series i.e. 4 inputs and 1 output.
The baseline model was therefore formulated with the following configuration:

a) Number of inputs = 4

b) Number of hidden layers = 1

c) Number of neurons per hidden layer shall = 8

d) Number of outputs = 1

e) Bias per layer = 1

f) Transfer function = sigmoid.

g) Maximum error = 0.01

h) Learning rate = 0.5

g) Maximum iterations = 50000

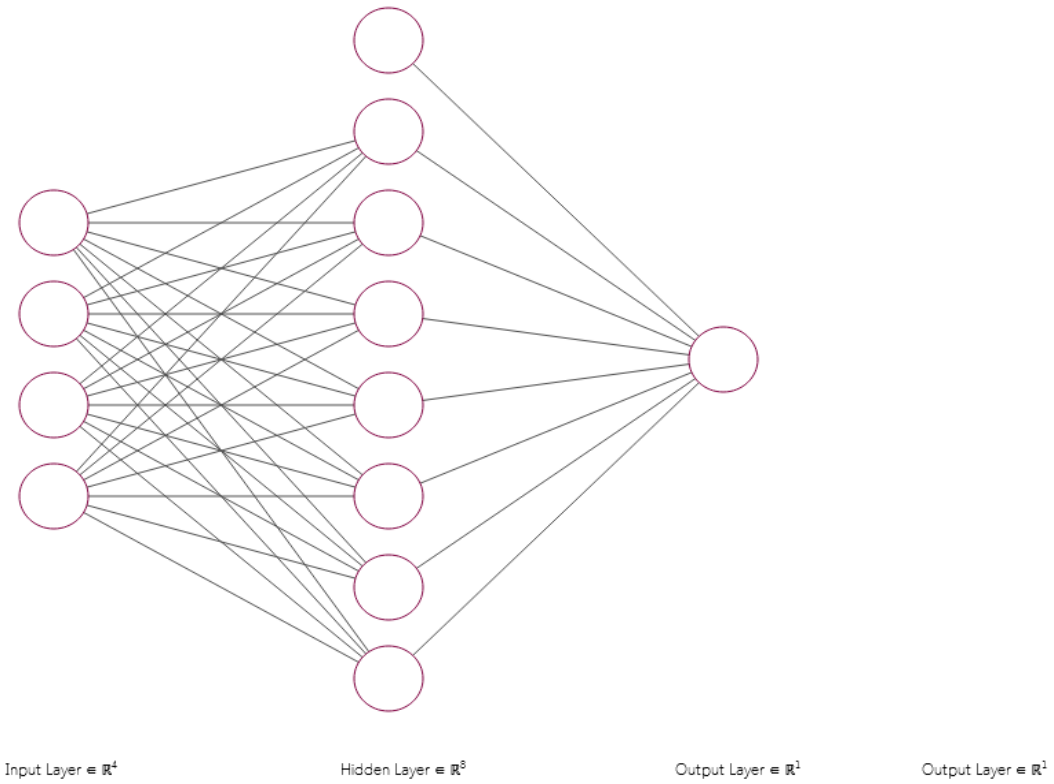


Figure 11- Univariate baseline model (Source: Author)

3.3.3. Multivariate baseline model.

In its multivariate mode the model comprised of input of 5 variables that included yearly rainfall amount (mm), wheat price, rice price, inflation rate and maize production (tons) with the aim of predicting maize price per ton for a particular year i.e. 5 inputs and 1 output.

The baseline model was therefore formulated with the following configuration:

a) Number of inputs = 5

b) Number of hidden layers = 1

c) Number of neurons per hidden layer shall = 8

d) Number of outputs = 1

e) Bias per layer = 1

f) Transfer function = sigmoid.

g) Maximum error = 0.01

h) Learning rate = 0.5

g) Maximum iterations = 50000

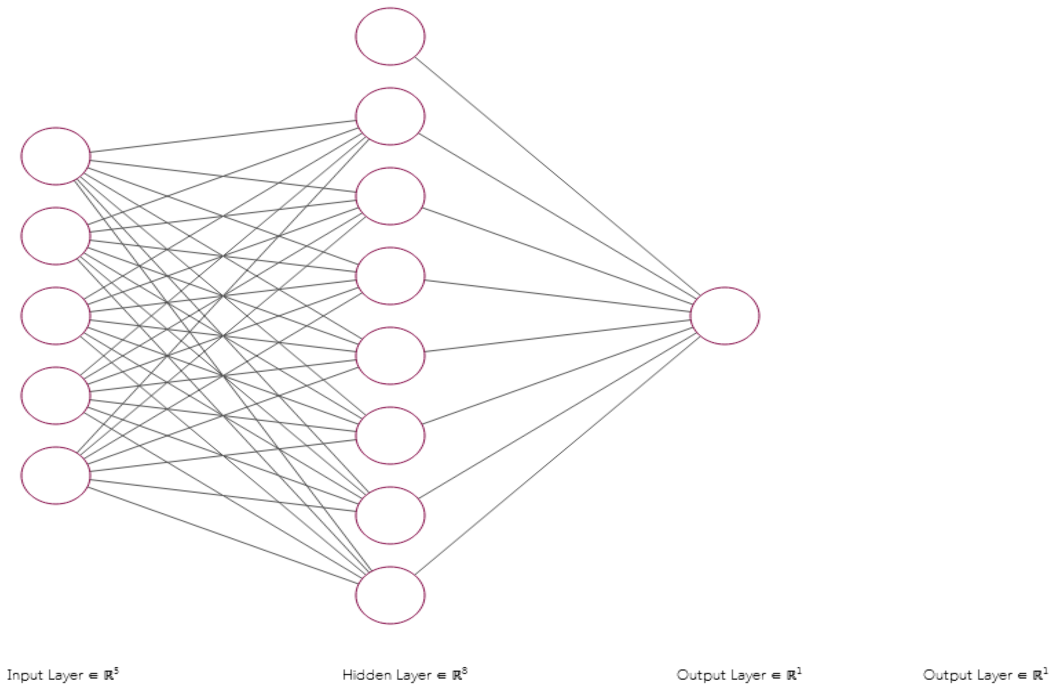


Figure 12 - Multivariate baseline model (Source: Author)

3.3.4. Baseline model testing and validation

The performance of the ANN baseline models was analyzed based on their accuracy in predicting maize prices for a continuous range of dates beyond the last date of their training. Training effectiveness was determined using the root mean square error (RMSE), over the range of training cycles. RMSE was also used to compare the performance of model, where a lower comparative RMSE value would imply a better prediction. However, the testing phase was measured on the basis of mean absolute deviation error (MAD), to determine exactly how far the actual and predicted values were. Results with lower MAD were more close to them actual values. In other research, (Girish & Kanchan, 2013) also based the measurement of the performance of their model on the selected crops (soybean and rapeseed-mustard) on the basis of MAD and RMSE while (Mir & Yaser, 2015) based the performance measurement of their model on livestock milk using MAD and RMSE The formula for RMS error (RMSE) is given by Equation 3.1 while that for MAD is given by Equation 7 below:

$$RMSE = \sqrt{\frac{\sum_{i=0}^{i}(y'_i - y_i)^2}{n}}$$

Equation 10Root mean squared error

Where:

n = number of observations, i= predicted value, yi= actual value

$$MAD = \frac{\sum_{i=0}^{i}(y'_i - y_i)}{n}$$

Equation 11Mean Absolute Deviation

Where:

n = number of observations, i= predicted value, yi= actual value

3.4 Proposed model parameters

3.4.1. Number of Hidden Layers and Neurons

The ANN structure for a particular problem in time series prediction includes determination of number of layers and total number of nodes in each layer. This study adopted the experimental approach in determining the number of nodes in each layer because we did not come across theoretical basis for determining these parameters. Previous research by (Girish & Kanchan, 2013) proved that approximations on non-linear functions is possible provided that an artificial neural networks has one hidden layer and sufficient number of nodes. This study used an ANN with a single layer. The auto correlation structure of a time series model is determined by the number of input nodes (Lagged observations of the same variables). In this study, one output node has been used. The univariate model setting was initiated with a baseline of settings 4:8:1 while the multivariate model setting had a baseline setting of 8:8:1, using 70% data for testing at 50,000 training cycles. This baseline model was then tuned, through experiment, to determine an optimum number of neurons on the hidden layer. The number of neurons on the hidden layer was progressively adjusted, followed by a series of training and testing phases. This test kept the number of hidden layers fixed as 1. The model in its univariate mode had its inputs and output fixed at 4 and 1 respectively while in its multivariate mode had its input fixed at 8 and its output fixed at 1. The inputs and output were also fixed at 4 and 1 respectively. The data volume was 70% for training (2006 to 2018) and 30% test data i.e. The results obtained for each setting are discussed in chapter four of this study based on raw data shown on Appendix 8. Based on the research questions and objectives of this study, we therefore ended up having two models to cater for the identified model scenarios, i.e. univariate and multivariate scenarios. Therefore the study preferred to have the model operate in both univariate and multivariate modes. The univariate mode gives monthly predications while multivariate mode gives yearly predictions based on user input.

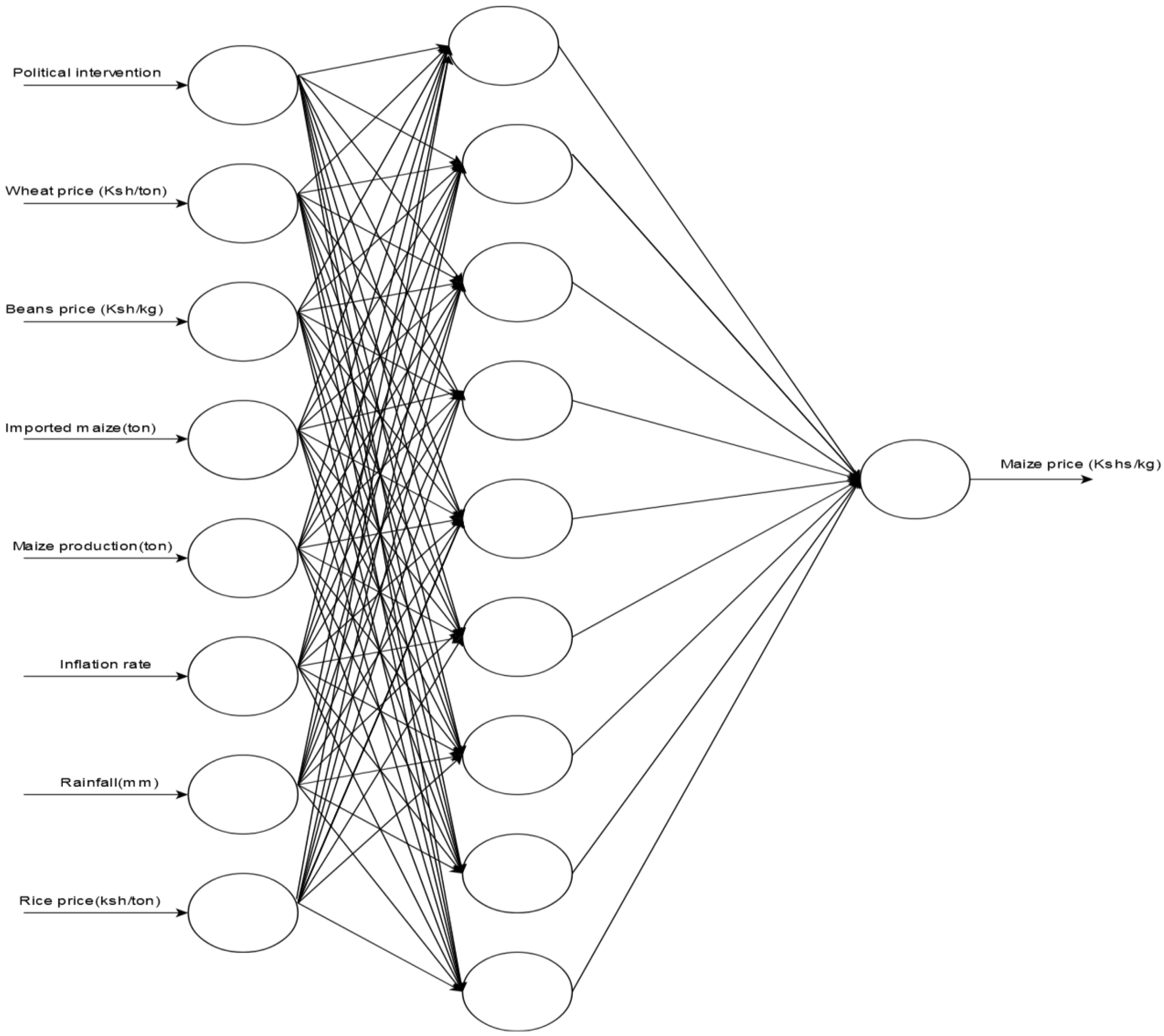


Figure 13 – Proposed multivariate model (Source: Author)

The proposed multivariate model is designed to handle both numerical and categorical inputs. An example of a categorical input would is inflation rate which this study decided to categorize it into either good or bad. This is because inflation rate and political situation are part of the assumptions that will be entirely based on human intuition.

The numerical inputs are actually parameters with measurable data.

Due to data complexity, there was need to apply the sigmoid activation function so as to represent non-linear complex arbitrary functional mappings between inputs and outputs.

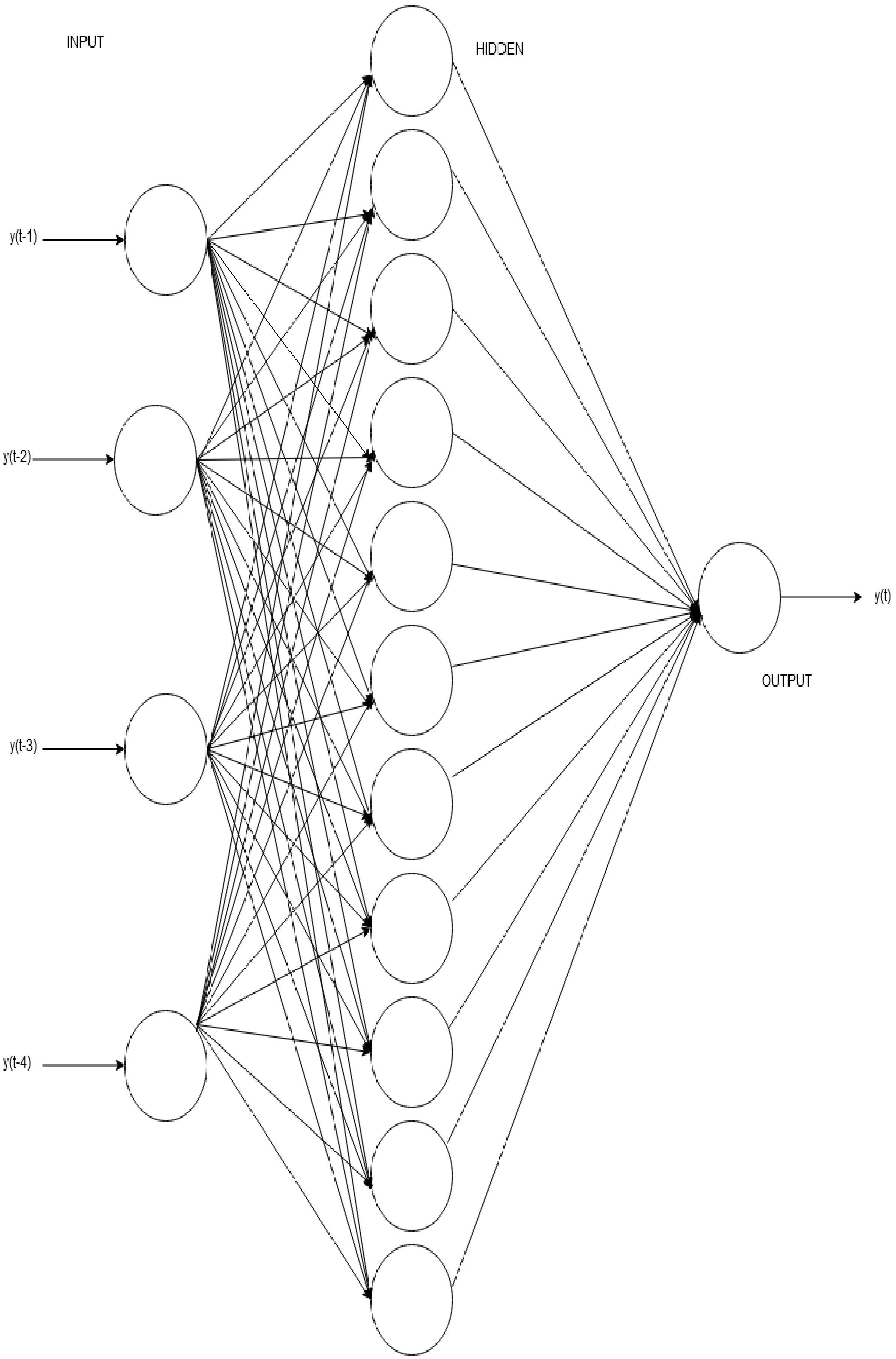


Figure 14 Proposed univariate model

The proposed univariate model setting implements a time delay approach whereby the output is determined by lagged observations of the last four months in the series. The universal expression for the final output value yt+1is given by equation 12

$$y_{t+1} = g[\sum_{j=0}^{q} \alpha_j \, f(\sum_{i=0}^{p} \beta_{ij} y_{t-i})]$$

Equation 12 Proposed univariate model equation

f = activation function at the hidden layer (Sigmoid function).

g = activation function at the output layers (Identity function).

p = number of input nodes.

q = the number of hidden nodes.

$\beta ij =$ weight attached to the connection between ith input node to the jth node of hidden layer.

$\alpha j =$ weight attached to the connection from the jth hidden node to the output node.

$yt{-}i = jth$ input (lag) of the model.

Every node of the hidden layer receives the weighted sum of all the inputs plus a bias variable for which the value of input variable will always take one as a value. The available activation (a non-linear sigmoid function) function f in each hidden node transforms the weighted sum of input variables. In the same way, the output node also receives the weighted sum of the output of all the hidden nodes and produces an output by transforming the weighted sum using its activation function g. For a univariate time series forecasting problem, the past observations of a given variable serves as input variables. The univariate model in this study will therefore attempt to conform to the following function.

$$y_{t-1} = f\left(y_t, y_{t-1}, \dots \dots . y_{t-1+p}, w\right) + \varepsilon_{t+1}$$

yt+1 = observation at time t+1.
p = number of lagged observation.
w = vector of network weights.
εt+1 = error-term at time t+1.

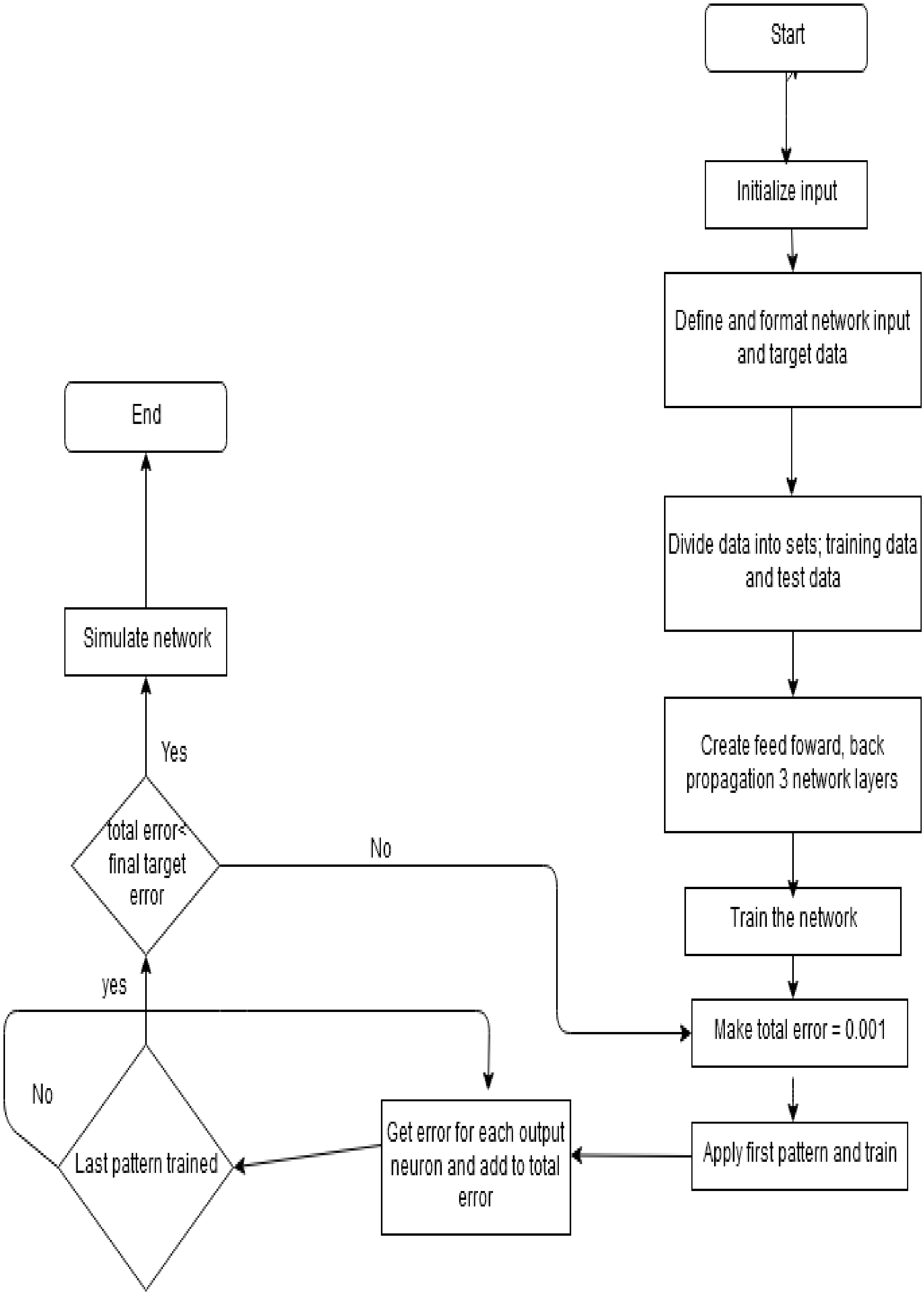


Figure 15 Univariate and multivariate model flow chat

3.5. Designing and developing the prototype

3.5.1. Prototype Design

The prototype was based on the model of configuration 8:9:1, using 70% of the available data (1992 to 2017) for training, with at least 150,000 repetitions. The ANN layout is multilayer perceptron (input layer, hidden layers, output layer, each with a number of neurons), the network connectivity is feed forward network (connectivity is between input neurons towards output neurons), weight adjustment is by error back propagation (resilient) and training is by supervised learning. The design tool that was used was a sequence diagram.

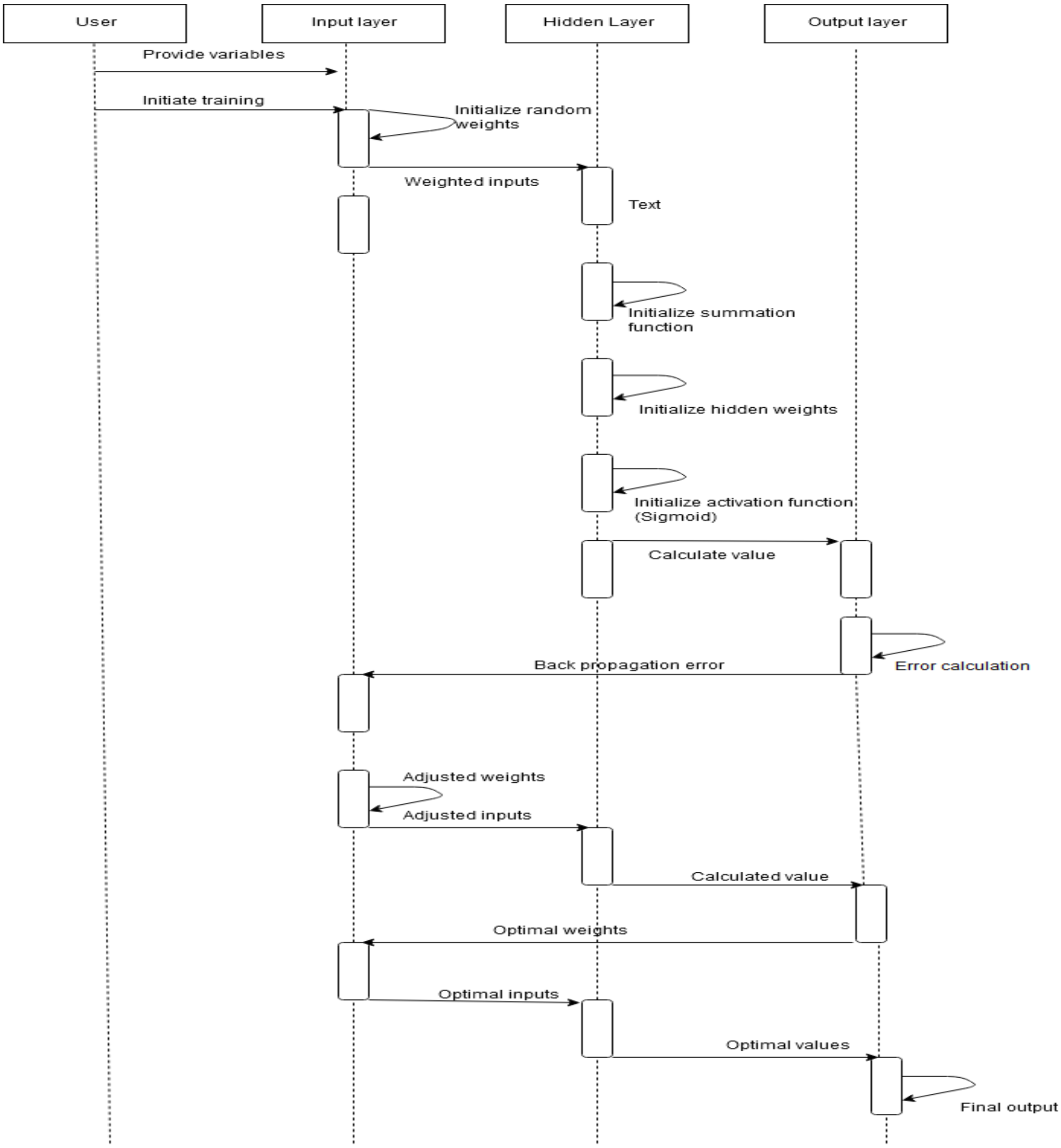


Figure 16 Prototype sequence diagram (Source:Author)

3.5.2. Prototype development

Requirements for prototype development included:

1. Laptop or desktop with at least 4GB RAM.
2. Linux or windows operating system.
3. Java 8.
4. Mysql server 5.3
5. Glassfish 4 application server.
6. Netbeans IDE.
7. Primefaces library (Has ajax components for user interface experience).
8. Neuroph library.
9. Github version control. (VCS).
10. Python numpy and scikit learn libraries.

Development of the prototype was done in java 8 programming environment. The web framework used was JSF (Java Server Faces). The user interface made use of bootstrap and primefaces library with ajax components that enabled ease of navigation for a non-technical user. Data was obtained from FAO and knoema websites in csv format and uploaded to a database created in mysql server. Source code snippet and version control repository link are provided in **Appendix 5**.

<u>Configurations page.</u>

This page contains the network configurations as shown in figure 4.9.6. The configurations include; maximum iterations, training and test data percentage, learning rate, allowable network training error and maximum normalization value.

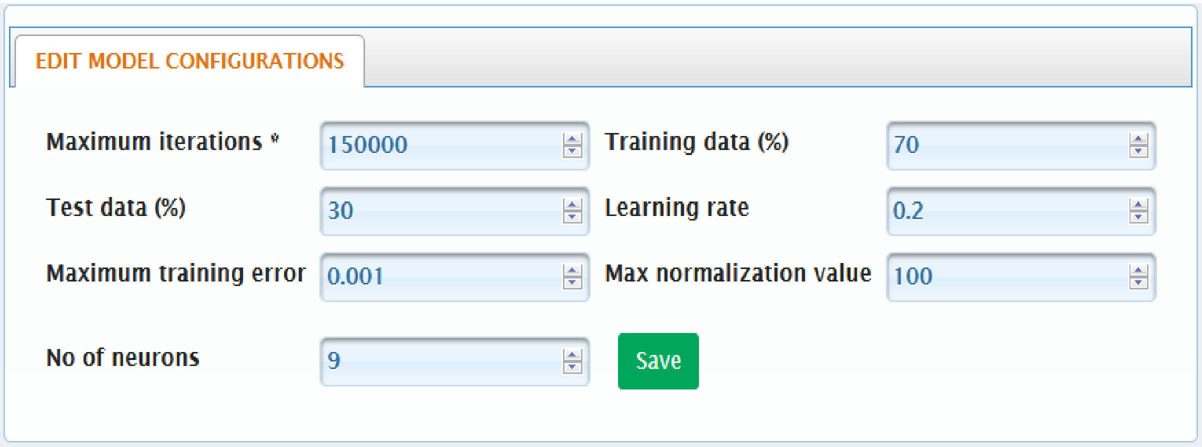


Figure 17 Model configurations

<u>Training page.</u>

The prototype has training pages for both univariate and multivariate scenarios. Time series data that was used in this study for univariate training covered Eldoret, Nairobi and Kisumu counties, therefore the univariate training page provides a drop down option menu for the user to select and train the model using data from a particular county.

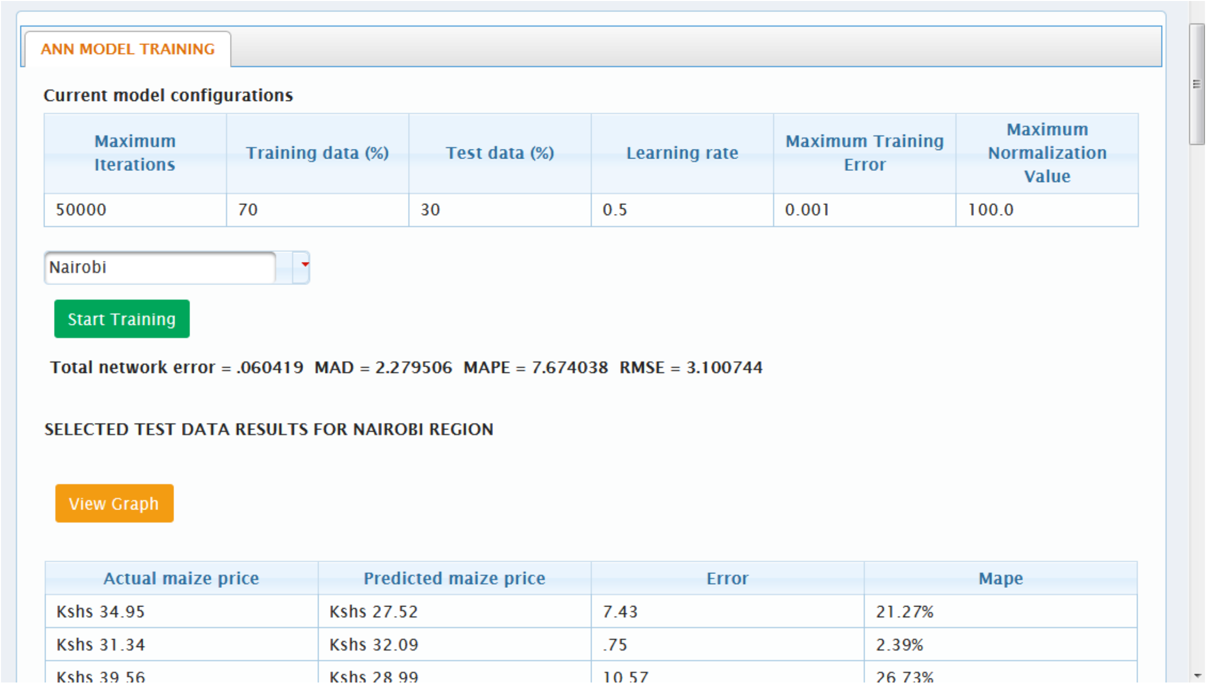


Figure 18 Model training (Univariate mode)

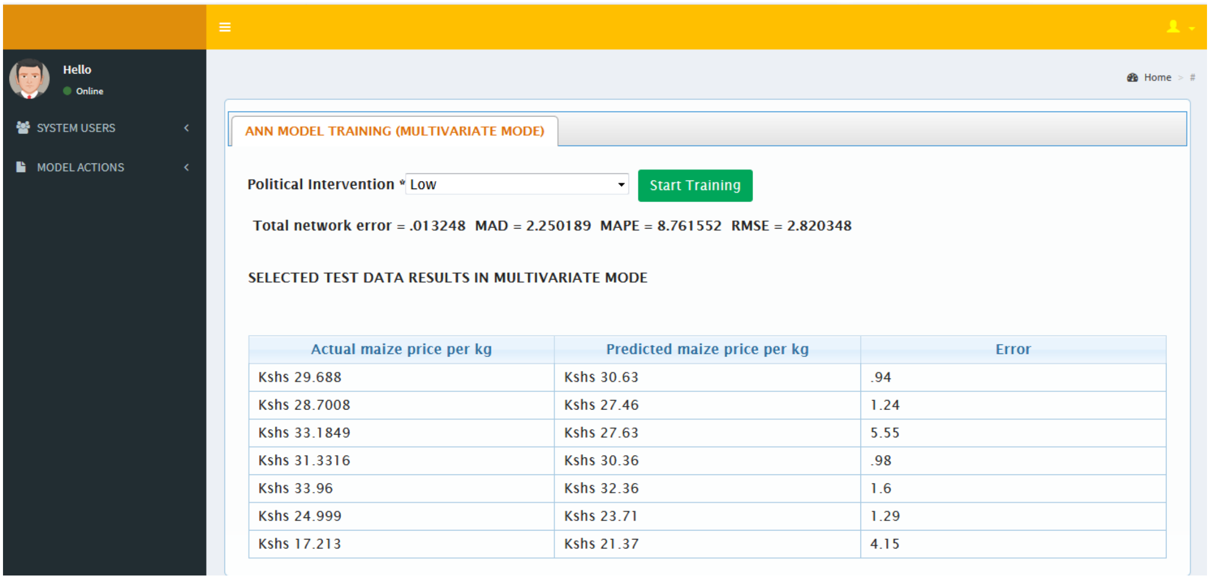


Figure 19 Model training (Multi variate mode)

<u>Dashboard.</u>

This is the landing page once a user logs in to the prototype. It gives the user the ability to predict maize price based on time series data (univariate) or a variety of inputs (multivariate) by the user.

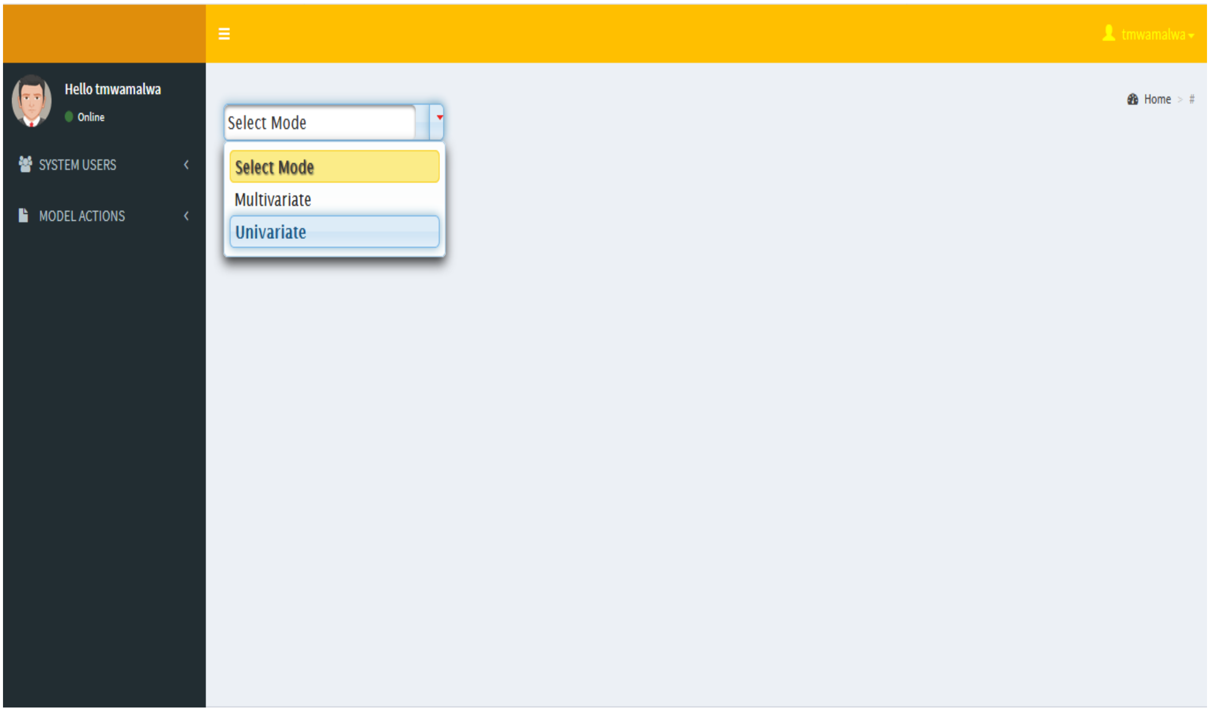


Figure 20 Dashboard

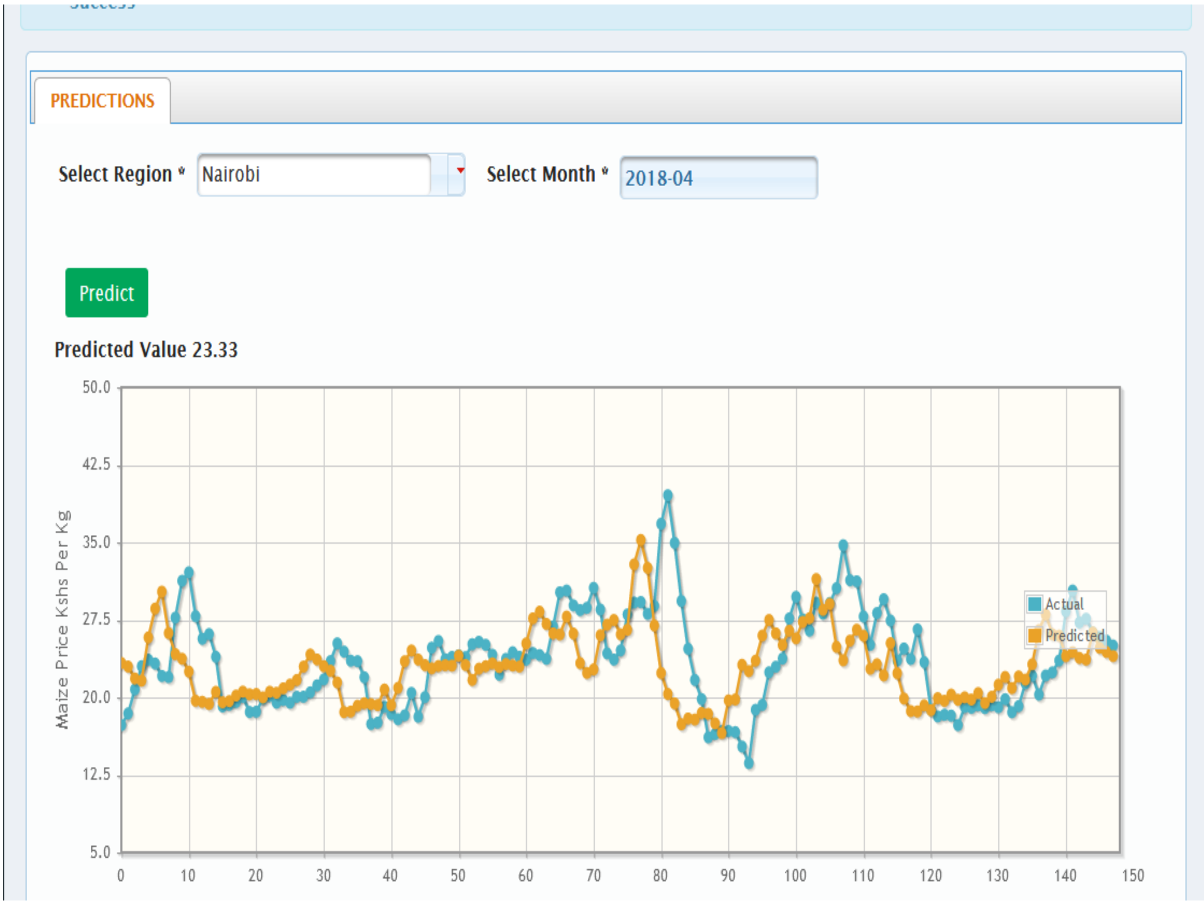


Figure 21 Time series dashboard

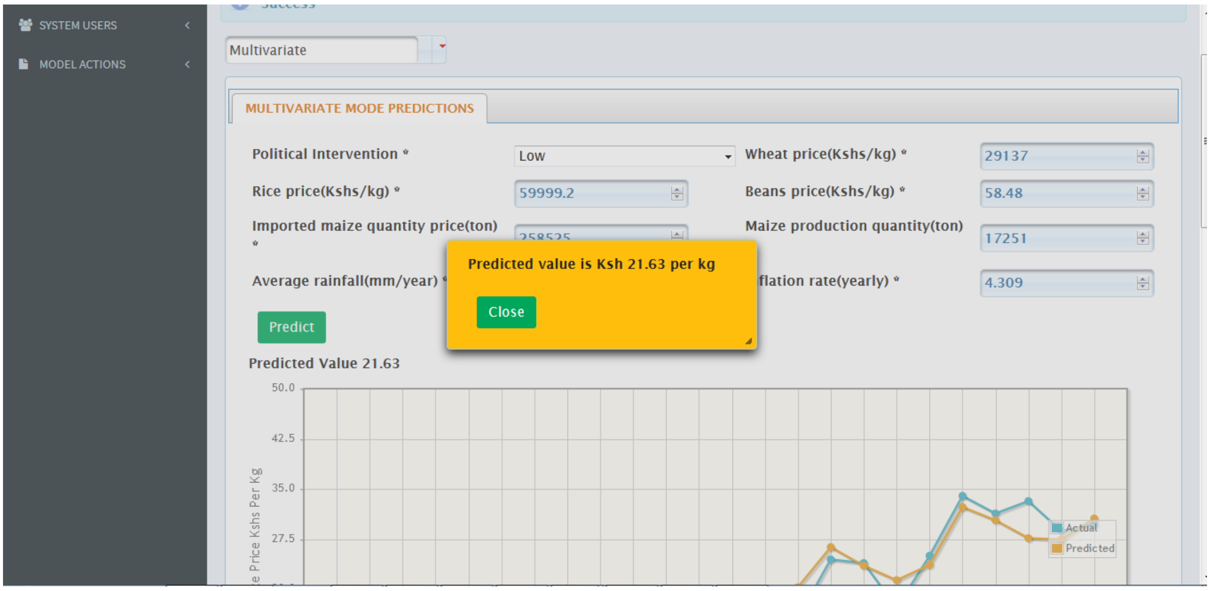


Figure 22 Multivariate dashboard

CHAPTER 4

RESULTS AND ANALYSIS

4.1. Introduction.

The process of model analysis encompasses evaluation of the model from its baseline to final proposed stage based on the parameters and provided data. This is captured by defining the model requirements analysis. It is a process that describes the process of studying and developing the business and the user needs to arrive at a definition of the problem domain and model requirements. It is the most critical aspect of the study and determines the goals and functions of the developed model (Dennis, Wixom, & Roth, 2012).

4.2. Baseline model analysis.

The main objective of conducting this evaluation was to obtain optimal model parameters that guided prototype development.

The results obtained for each setting are shown on Table 2 below, based on raw data shown on Appendix 8.

Table 2 Results for varying number of hidden neurons

Input Neurons	**Hidden Layer (No of neurons)**	**Output Layer(No of neurons)**	**MAD**	**MAPE(%)**	**RMSE**
UNIVARIATE MODE					
4	1	1	3.290801	10.907103	4.271512
4	2	1	2.92237	9.682827	3.938135
4	3	1	2.823173	9.301333	3.871598
4	4	1	2.651561	9.008536	3.38449
4	5	1	2.400144	8.130653	3.141505
4	6	1	2.263778	7.666079	3.019906
4	7	1	2.316327	7.845671	3.067682
4	8	1	2.242621	7.590439	3.002267
4	9	1	2.3149	7.840495	3.068427
4	10	1	2.241042	7.585289	3.00102

4	11	1	2.182236	7.445803	2.938315*
4	12	1	2.301587	7.829555	3.058776
4	13	1	2.293111	7.698813	3.052891
4	14	1	2.307742	8.026973	3.125744
4	15	1	2.399848	7.93225	3.190799
4	16	1	2.321472	8.187287	3.084805
4	17	1	2.330329	8.293067	3.323696
MULTIVARIATE MODE					
8	1	1	4.201957	16.251688	4.935287
8	2	1	2.521771	9.572593	3.305861
8	3	1	2.370213	8.902132	3.002879
8	4	1	2.418833	9.1465	3.050093
8	5	1	2.3916	9.110782	3.059091
8	6	1	2.387732	9.105044	3.093425
8	7	1	2.391613	8.98785	2.988435
8	8	1	2.996512	8.963761	2.996512
8	9	1	2.355522	8.834932	2.9716*
8	10	1	2.417444	9.023716	2.989619
8	11	1	2.387357	8.970819	2.991796
8	12	1	2.403007	8.982441	2.98932

*Lowest value

These results are also shown graphically on Figure 23 and Figure 24 below.

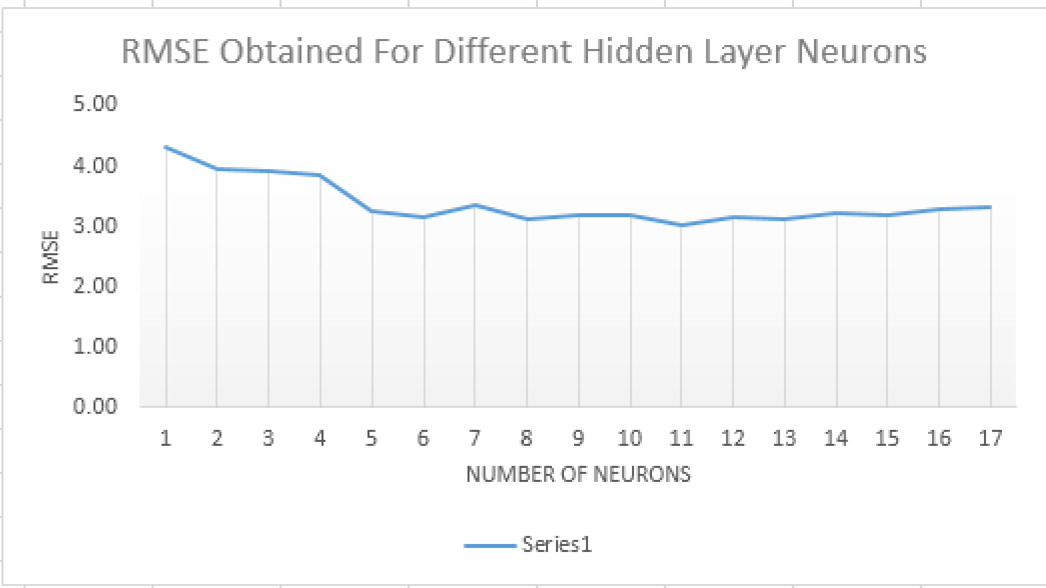


Figure 23 RMSE for neurons in the hidden layer – Univariate mode

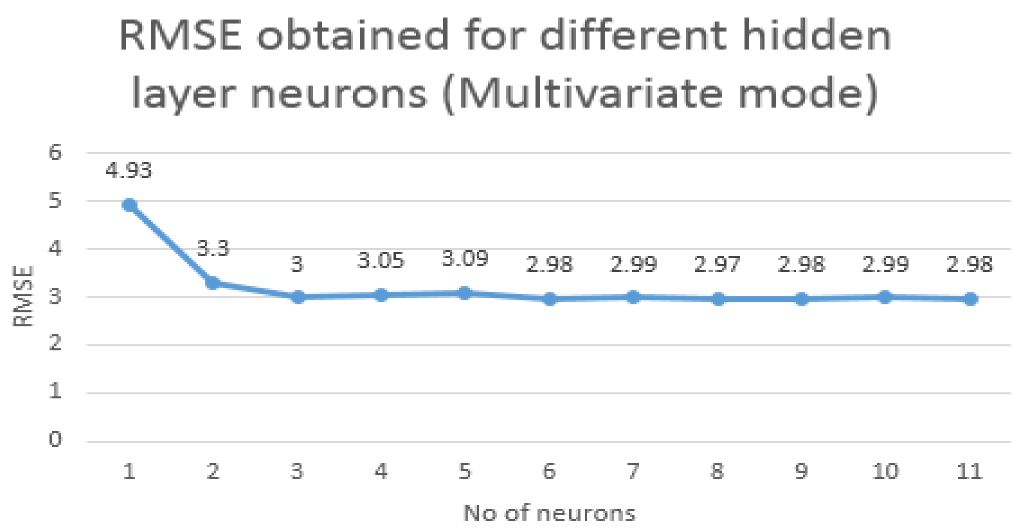


Figure 24 RMSE for neurons in the hidden layer - Multivariate mode

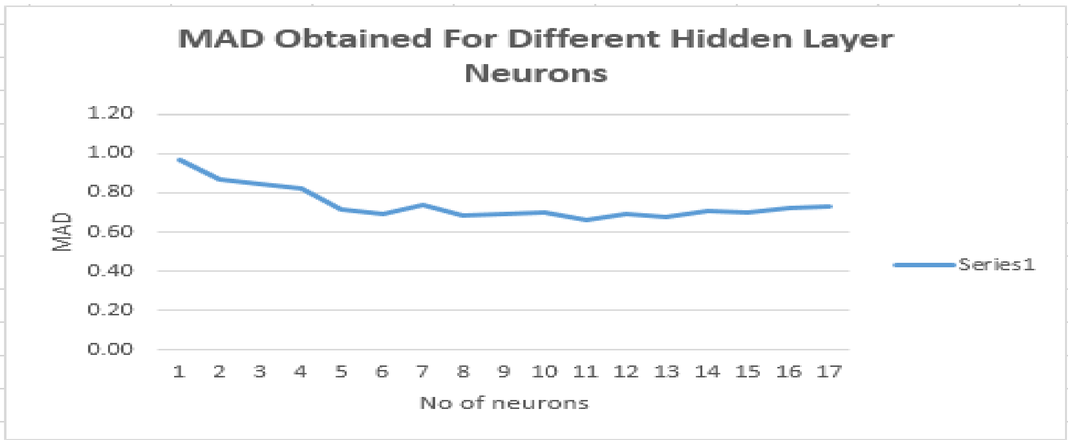


Figure 25 MAD obtained for neurons in the hidden layer – Univariate mode

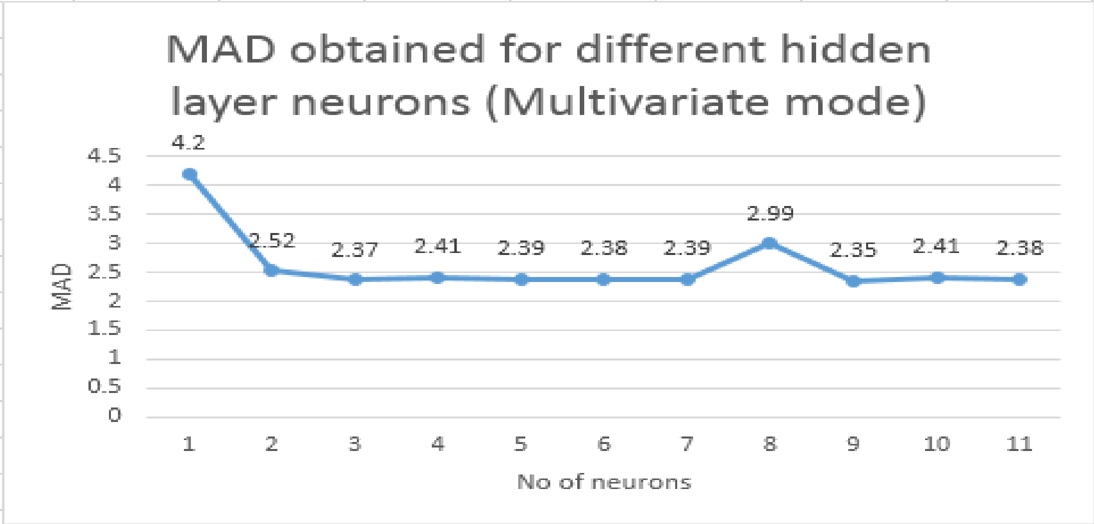


Figure 26 MAD obtained for neurons in the hidden layer – Multivariate mode

The following three observations were made – firstly, the low rates of RMSE was obtained with neurons per hidden layer of 8, 10 and 11 during testing the model in its univariate mode. Secondly, based on MAD, the best configuration was obtained from the models multivariate mode with 8 neurons for 9 neurons per hidden layer i.e. 8:9:1. Based on this determination, the configuration of 8:9:1 was therefore the optimal multivariate ANN configuration and 4:11: 1 for the time series configuration.

The new developed models and their flow chart diagram are shown in Figure 13, 14 and 15 respectively.

4.3. Proposed model analysis

Evaluation of the ANN model was done using experimental methods. The tests were done using the developed prototype to evaluate its univariate mode performance along various horizons that included; 1, 2, 3 and 4 month predictions. The model was also evaluated against an ARIMA model along the same periods. The results are shown in table 3 Table 4.3 shows test data results of the ANN's four month prediction while figure 4.9.5 is a graphical representation of the ARIMA model's four month prediction results. The model's multivariate mode performance was also evaluated based on different user inputs.

MODEL	1 month ahead		2 months ahead		3 months ahead		4 months ahead	
	RMSE	MAD	RMSE	MAD	RMSE	MAD	RMSE	MAD
ANN	2.8	1.9	3.06	2.15	3.10	2.18	3.10	2.2
ARIMA	4.691	4.879	4.807	5.103	4.918	5.141	4.931	5.159

Table 3 Forecasting results of ANN and ARIMA models.

Month-Year	Actual Price	Predicted Price	Error
Mar-12	Kshs 23.66	Kshs 23.21	0.45
Apr-12	Kshs 24.26	Kshs 22.53	1.73
May-12	Kshs 28.5	Kshs 27.	0.5
Jun-12	Kshs 30.6	Kshs 26.34	4.26
July-12	Kshs 28.67	Kshs 28.1	0.57
Aug-12	Kshs 28.45	Kshs 27.12	1.33
Sep-12	Kshs 28.91	Kshs 26.58	2.33
Oct-12	Kshs 30.34	Kshs 26.18	4.16
Nov-12	Kshs 30.17	Kshs 27.65	2.52
Dec-12	Kshs 26.87	Kshs 27.57	0.7
Jan-13	Kshs 23.73	Kshs 25.14	1.41
Feb-13	Kshs 24.1	Kshs 22.63	1.47
Mar-13	Kshs 24.3	Kshs 22.52	1.78
Apr-13	Kshs 23.65	Kshs 22.71	0.94
May-13	Kshs 23.93	Kshs 22.87	1.06
Jun-13	Kshs 24.37	Kshs 23.07	1.3
July-13	Kshs 23.73	Kshs 23.23	0.5
Aug-13	Kshs 22.2	Kshs 22.91	0.71
Sep-13	Kshs 24.13	Kshs 21.85	2.28

Table 4 Forecasting results of ANN models for 4 month's predictions.

Year	Actual maize price per kg	Predicted maize price per kg	Error
1992.0	Kshs 2.4	Kshs 5.85	Kshs 3.45
1993.0	Kshs 8.1	Kshs 7.06	Kshs -1.04
1994.0	Kshs 9.5	Kshs 9.42	Kshs -.08
1995.0	Kshs 8.0	Kshs 8.61	Kshs .61
1996.0	Kshs 9.0	Kshs 10.92	Kshs 1.92
1997.0	Kshs 13.73	Kshs 13.62	Kshs -.11
1998.0	Kshs 12.84	Kshs 11.7	Kshs -1.14
1999.0	Kshs 13.86	Kshs 12.5	Kshs -1.36
2000.0	Kshs 14.49	Kshs 11.62	Kshs -2.87
2001.0	Kshs 13.31	Kshs 12.4	Kshs -.91
2002.0	Kshs 11.14	Kshs 11.92	Kshs .78
2003.0	Kshs 11.96	Kshs 13.15	Kshs 1.19
2004.0	Kshs 15.34	Kshs 14.67	Kshs -.67
2005.0	Kshs 15.24	Kshs 12.53	Kshs -2.71
2006.0	Kshs 15.35	Kshs 13.33	Kshs -2.02
2007.0	Kshs 15.66	Kshs 20.29	Kshs 4.63
2008.0	Kshs 24.45	Kshs 26.3	Kshs 1.85
2009.0	Kshs 23.91	Kshs 23.52	Kshs -.39
2010.0	Kshs 17.21	Kshs 21.35	Kshs 4.14
2011.0	Kshs 25.0	Kshs 23.63	Kshs -1.37
2012.0	Kshs 33.96	Kshs 32.29	Kshs -1.67
2013.0	Kshs 31.33	Kshs 30.32	Kshs -1.01
2014.0	Kshs 33.18	Kshs 27.64	Kshs -5.54
2015.0	Kshs 28.7	Kshs 27.43	Kshs -1.27
2016.0	Kshs 29.69	Kshs 30.58	Kshs .89

Table 5 Forecasting results of ANN model (multivariate mode)

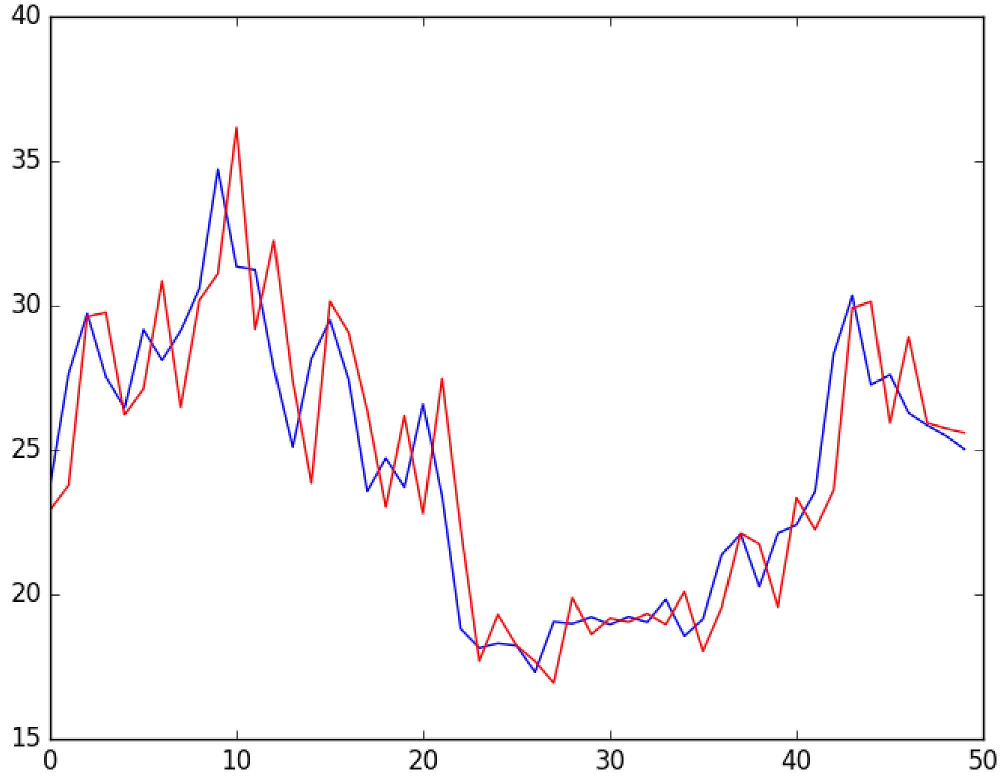


Figure 27 ARIMA prediction results from Jan-2006 to Apr -2018.

4.4. Chapter Summary

Chapter 4 provides the results and then reviews, interprets and analyzes these results. Results obtained from the model in its univariate and multivariate forms are compared. The model is also evaluated against other models that can do predictions (ARIMA).

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS.

5.1. Conclusions.

The main advantage of univariate time-series forecasting is that it needs data only of the time series that is being analyzed. This characteristic is useful if we are to forecast a huge amount of price series. Performance of model estimation largely depends on data and if this data has some missing values then restrictions over which the model can be estimated occurs. However agricultural commodity prices are shaped by various factors which need to be considered during forecasting, therefore the study identified average rainfall, alternative staple and cereal prices, maize production, inflation and political intervention as factors that can serve as model inputs for predicting maize prices.

This study has made a comparison between ARIMA and ANN models in terms of forecasting using monthly retail maize price data in three Kenyan counties, namely; Eldoret, Nairobi and Kisumu. Comparisons made between ANN and ARIMA models indicate that ANN provided a better forecast accuracy in terms of measures based on RMSE and MAD values. Nonlinearity of data determines the reliability on the forecasting accuracy of ARIMA and ANN models based on RMSE measures.

Globalization and interaction among various world markets have brought about the need for proper decision making. Therefore an effort towards designing intelligent systems is needed for purposes of integrating traditional statistical methods with emerging technologies like neural network, fuzzy logic, etc. to provide reliable forecasting information that can be used by farmers, retailers and policymakers so that they may make production, marketing and policy decisions well in advance.

Market regulatory conditions greatly affects the price of maize and maize products, therefore, farmers and producers should take future prices of this input into account.

Considering price of maize in the future, agricultural authorities can reduce price variations and consequently reduce the high risk present in maize and maize related products` market and finally can increase producers and consumers` interests. In fact, they can support maize farmers and maize product units in making the right decision by identifying and showing future price condition in this sector at different times.

5.2. Recommendation.

Price forecasting is a key factor, it is therefore important to have updated information by examining market condition in future researches at universities and research centers and also to make use of different prediction models such as artificial neural network in order to reduce maize production risk in Kenyan Agricultural Sector.

Future studied needs to explore the possibility of using a hybrid model (combined linear and non- linear model) for predicting maize prices in specific Kenyan counties based on aggregated price data from all counties.

Finally, further research is needed to determine how long a trained ANN system remains valid and effective in prediction before it is found to be in need of retraining.

REFERENCES

Argwings, K. (1998). *Policy Issues facing the Maize Sector in the North Rift.*

Arienda, A., & Asana, J. C. (2015, February). *Operations management forecasting.* Retrieved from http://www.slideshare.net/Euwinx/operations-management-forecasting

Bessler, & Brandt. (1992). An analysis of forecasts of livestock. *Journal of Economic Behavior and Organization* .

Chapoto, Antony, & Jayne, T. S. (2009). *Effects of Maize Marketing and Trade Policy on Price Unpredictability in Zambia.*

Forecasting. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Forecasting

GeeksForGeeks. (n.d.). Retrieved from https://www.geeksforgeeks.org/introduction-to-artificial-neutral-networks

Girish, K. J., & Kanchan, S. (2013). *Agricultural Price Forecasting Using Neural Network Model: An Innovative Information.*

Gupta, N. (2017). A Literature Survey on Artificial Intelligence. *ICPCN - 2017 Conference Proceedings.* https://www.ijert.org/research/a-literature-survey-on-artificial-intelligence-IJERTCONV5IS19015.pdf.

Indiana University. (n.d.). Retrieved from https://www.cs.indiana.edu/~predrag/classes/2005springi400/lecture_notes_4_1.pdf

James K, N., Lillian, K., & T.S, J. (n.d.). *COMPETITIVENESS OF KENYAN AND UGANDAN.*

James K. Nyoro, L. K. (n.d.). *COMPETITIVENESS OF KENYAN AND UGANDAN.* Nairobi.

Jayalakshmi, & Santhakumaran. (2011). Statistical normalization and back propagation for classification.

Jayne, T., Robert, & James. (2005). *The effects of NCPB marketing policies on maize market prices in Kenya.*

Jayne, T., Robert, J. M., & James, N. (2006). *The Effects of Government Maize Marketing and Trade.*

Joshua, A., T.S, J., & Stephen, N. (2010). *Staple food prices in Kenya* . Maputo.

Kenya, G. o. (1998). *Statistical Abstract. Nairobi: Central Bureau of Statistics.*

Khalid, Sultana, & Zaidi. (2014). Prediction of Agriculture Commodities Price Returns Using. *Journal of Natural Sciences Research* .

Kothari, C. (2004). *Research methodology.*

Linwood, H. (2004). *Using Futures Prices to Forecast the Season-Average U.S. Corn Price.*

Mackey, A., & M, S. (2013). Research Methods in Second Language Acquisition. *The Modern Language Journal*.

McLeod, A. a. (1983). Diagonostic checking ARMA time series models using squared residual autocorrelations. *Journal of Time Series Analysis, 4*, 269-273.

Mir, G. S., & Yaser. (2015). Application of Artificial Neural Network Model in Predicting Price of Milk in Iran.

Oancea, & Ciuncu. (2014). Time series forecasting using neural networks.

Shawiza, V. (2016, November 22). *Soko directory*. Retrieved from Soko directory: https://sokodirectory.com/2016/11/outlook-kenyan-maize-market/

society, A. (2017, August). *Artificial Intelligence And Electronics Society*. Retrieved from https://medium.com/@ariesiitr/an-artificial-neural-network-ann-is-a-computational-model-that-is-inspired-by-the-way-biological-c17b07166d4c

T.S, J., Stephen, N., & Joshua, A. (2010). *Staple food prices in Kenya.* Maputo.

T.S. Jayne, R. J. (2005). *The Effects of Government Maize Marketing and Trade.*

T.S. Jayne, R. J. (2007). The effects of NCPB marketing policies on maize market prices in Kenya. *Agricultural Economics*.

T.S. Jayne, R. J. (2007). THE EFFECTS OF NCPB MARKETING POLICIES ON MAIZE MARKET PRICES IN KENYA. *Agricultural Economics*.

T.Thiraviyam. (2018). *Artificial Intelligence Marketing.*

ujjwalkarn. (2016, August). Retrieved from https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/

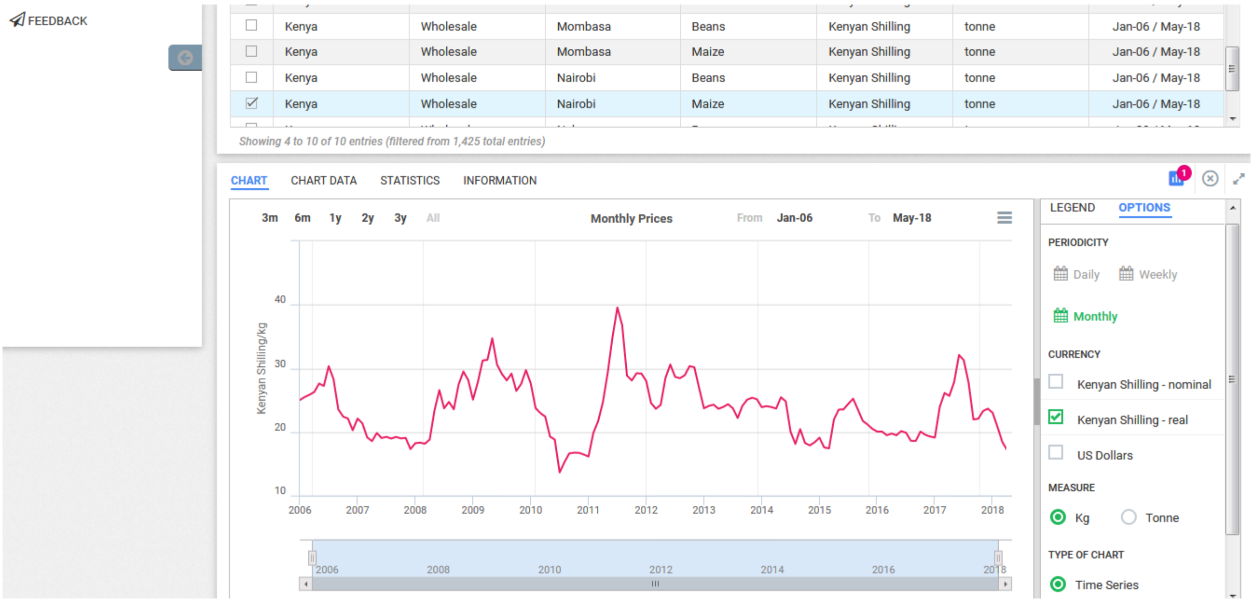
Wilson, E. (2016). *Supply Chain Analysis.* http://ffinetwork.org/about/calendar/2016/documents/Maize_Scoping.pdf. Retrieved from http://ffinetwork.org/about/calendar/2016/documents/Maize_Scoping.pdf

Wyk. (2012). Retrieved from https://s3.amazonaws.com/academia.edu.documents/43226041/Research_and_Design_I1.pdf?response-content-disposition=inline%3B%20filename%3DResearch_design_and_methods_Part_I.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F2019081

APPENDICES

APPENDIX 1 – Screen shot of FAO website.

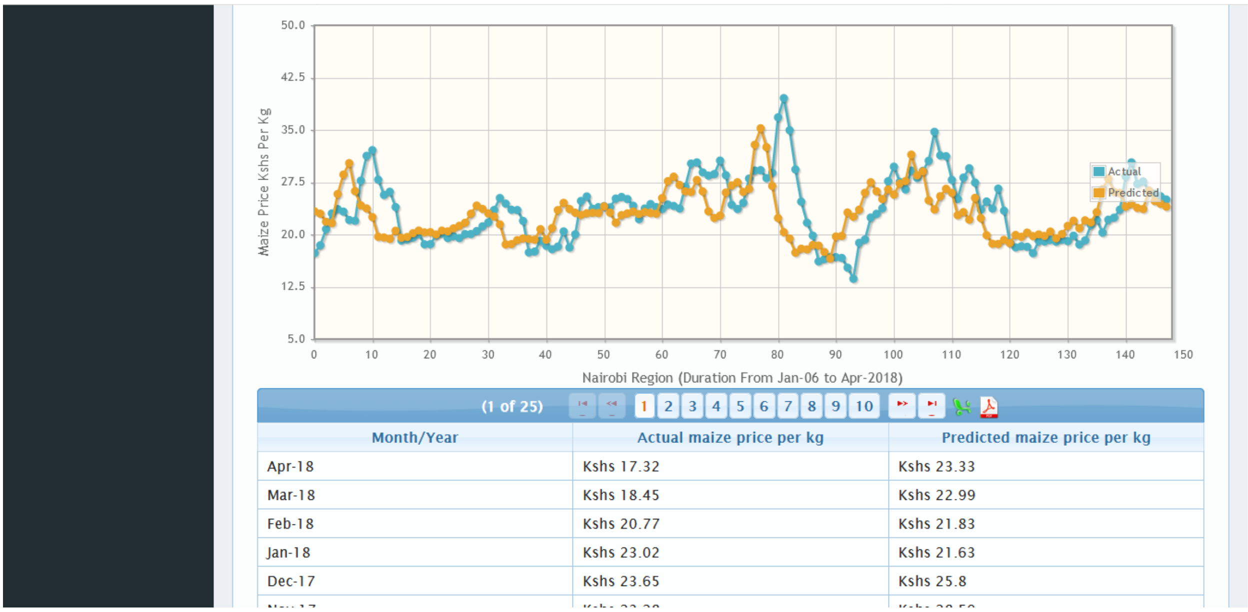
The screenshot below shows the FAO website indicating monthly maize prices from 2006 - 2018



Source: FAO website. (2018)

APPENDIX 2 – Screen shot of the prediction model.

The screenshot below shows both actual and predicted time series prices indicating monthly maize prices from 2006 - 2018

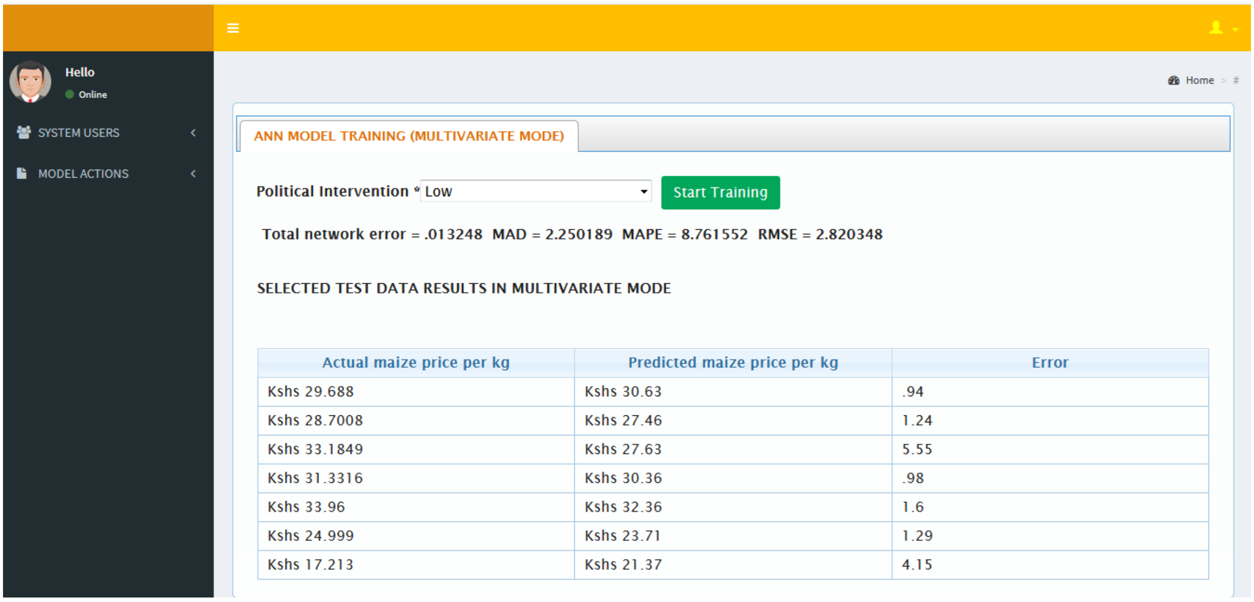


Month/Year	Actual maize price per kg	Predicted maize price per kg
Apr-18	Kshs 17.32	Kshs 23.33
Mar-18	Kshs 18.45	Kshs 22.99
Feb-18	Kshs 20.77	Kshs 21.83
Jan-18	Kshs 23.02	Kshs 21.63
Dec-17	Kshs 23.65	Kshs 25.8

Source: Author. (2018)

APPENDIX 3 – Screen shot of the prediction model in its multivariate mode.

The screenshot below shows test results of the model in its multivariate mode.



Actual maize price per kg	Predicted maize price per kg	Error
Kshs 29.688	Kshs 30.63	.94
Kshs 28.7008	Kshs 27.46	1.24
Kshs 33.1849	Kshs 27.63	5.55
Kshs 31.3316	Kshs 30.36	.98
Kshs 33.96	Kshs 32.36	1.6
Kshs 24.999	Kshs 23.71	1.29
Kshs 17.213	Kshs 21.37	4.15

NB: Price is in Ksh per ton.
Source: Author. (2018).

APPENDIX 4 – Multivariate data.

The data below was used during testing the model's multivariate capability.

Year	RicePrice	Wheatprice	maizeprice	maizeyield	inflation	precipitation	Maize import	Political Intevention	Bean prices
1992	1596	3857	2396	17271	27.332	630	415320	1	32.26
1993	1171	5650	8104	15549	45.979	630	80051	1	35.78
1994	5245	12000	9500	20400	28.814	630	650224	1	43.78
1995	5515	13000	8000	18759	1.554	630	40000	1	44.56
1996	12257.5	15630	9000	14506	8.864	630	6759	1	42.14
1997	19000	17700	13732	14713	12.096	630	1101105	1	39.68
1998	17623	16901	12844	16697	5.612	630	368761	1	47.5
1999	21087	18140	13859	14817	4.984	630	73520	1	48.18
2000	22844	16075	14494	14400	7.77	630	409416	1	43.4
2001	15745	18007	13308	17012	5.824	630	314381	1	50.54
2002	11292	17243	11144	15126	2.156	630	16348	1	55.32
2003	16153	19389	11959	16224	5.983	630	100132	1	48.04
2004	26000	22167	15342	19293	8.381	630	241757	1	46.34
2005	28611	18211	15237	16405	7.823	630	43652	1	55.21
2006	29947	19496	15354	17197	6.041	630	250352	1	53.23
2007	33977.8	28589	15664	18132	4.265	630	293073	1	47.65
2008	36533	31832	24453	13925	15.101	630	93473	1	63.16
2009	57970.2	29368	23913	12943	10.552	630	236000	1	60.12
2010	59999.2	29137	17213	17251	4.309	630	258525	1	58.48

2011	84354.4	30174	24999	15840	14.022	630	229596	1	64.82
2012	61462.5	36223.1	33960	17366	9.378	630	1508414	1	57.86
2013	47925.6	37448.5	31331.6	16922	5.717	630	243656	1	49.98
2014	47235.7	34953.4	33184.9	16602	6.878	630	113768	1	55.05
2015	57670	35616.2	28700.8	18230	6.582	630	147000	1	49.99
2016	55886.1	37184	29688	14284	6.318	630	94000	1	49.0

APPENDIX 5 – Code snippet and version control link.

The code snippet is a method responsible for training the data.

```
public String Train(){
        ctx = FacesContext.getCurrentInstance();
        rtx = RequestContext.getCurrentInstance();
       String page = "train.nnet";
       String networsavedstatename = "";

       try{

          if(selecteddata.equals("0")){
            rtx.execute("PF('dlg3').hide()");
           ctx.addMessage(null, new FacesMessage(FacesMessage.SEVERITY_WARN,
                   "Please select region", ""));
          }else{

             if(selecteddata.equals("NAIROBI")){
                 networsavedstatename = "nrbPerceptron.nnet";
             }else if(selecteddata.equals("ELDORET")){
                networsavedstatename = "eldPerceptron.nnet";
             }else{
                networsavedstatename = "ksmPerceptron.nnet";
             }
             loadModel = getModelDM().getRowData();
             System.out.println("This is the Max error "+ loadModel.getMaxiterations());
             double normolizer = loadModel.getNormalizer();
              DAO dao = new DAO();
              dao.readNRBData(selecteddata, normolizer,
                     loadModel.getTrainingdata());

               int maxIterations = loadModel.getMaxiterations();

                NeuralNetwork neuralNet = new
MultiLayerPerceptron(TransferFunctionType.SIGMOID,4, 11, 1);
      ((LMS) neuralNet.getLearningRule()).setMaxError(loadModel.getMaxerror());//0-1
      ((LMS)
neuralNet.getLearningRule()).setLearningRate(loadModel.getLearningrate());//0-1
      ((LMS) neuralNet.getLearningRule()).setMaxIterations(maxIterations);//0-1
      //TrainingSet trainingSet = new TrainingSet();
      TrainingSet trainingSet = dao.getTrainingSet();

       BackPropagation backPropagation = new BackPropagation();
             backPropagation.setMaxIterations(maxIterations);


        neuralNet.learnInSameThread(trainingSet);


      System.out.println("Neural Total network Error " +
```

```
((LMS)neuralNet.getLearningRule()).getTotalNetworkError());
        double networkError = ((LMS)neuralNet.getLearningRule()).getTotalNetworkError();
                 setTotalnetworkError(String.valueOf(df2.format(networkError)));
        //System.out.println("Neural Total network Error " + neuralNet.getLearningRule());
        // save trained neural network
          neuralNet.save(networsavedstatename);
          // load saved neural network
         NeuralNetwork loadedMlPerceptron = NeuralNetwork.load(networsavedstatename);
         HashMap hm = new HashMap();
         Statement stmt = null;

        PreparedStatement preparedstatement = null;
        String values="";
        String id = "";
        Connection connection = null;

        int counter = 0;

        String date = "";

         try {

             connection = GetDatabaseConnection.getMysqlConnection();
            stmt =  connection.createStatement();
            String query ="SELECT ID,DATE,DATA FROM "+selecteddata +" ORDER BY ID
DESC LIMIT";
             preparedstatement = (PreparedStatement) connection.prepareStatement(query);

           String name = "\"1-02\"";

            ResultSet result = preparedstatement.executeQuery();
            while(result.next()){

                values=result.getString("DATA");
                id = String.valueOf(result.getInt("ID"));
                date = result.getString("DATE");
                System.out.println("\""+date +"\","+ values);
                hm.put(id, values);
               counter = counter + 1;
            }

             connection.close();
        } catch (Exception ioe) {
            System.out.println("Oops- an IOException happened.");
            ioe.printStackTrace();
            System.exit(1);
        }
        int maxCount = (int)(loadModel.getTestingdata()*counter);
        System.out.println("full number of values = " + counter + " Percentage "+
maxCount/100);
```

```
setMaxCounter(maxCount/100);
Set s = hm.keySet();
Iterator i = s.iterator();
valuesRow = new String[this.getMaxCounter()];

int n = 0;
while (i.hasNext()) {
    String key = (String) i.next();
    String value = (String) hm.get(key);
    //System.out.println(key + "->" + value);
    n = n + 1;
    //configurable percentage of
    if (counter - n < this.getMaxCounter()) {
        valuesRow[counter - n] = value;
        //System.out.println("Get values "+ value);
      // System.out.println(valuesRow[counter - n] );
    }
}
System.out.println("valuesRow.length=" + valuesRow.length);
 if (valuesRow.length < 5) {
    System.out.println("valuesRow.length < 5");

}
 double error = 0.0;
 double rmse = 0.0;
 double sqrtrmse=0.0;
 double mape = 0.0;
 double mad =0.0;
 TrainingSet testSet = new TrainingSet();


 for (int j = 0; j + 4 < valuesRow.length; j++) {
        String s1 = valuesRow[j];
        String s2 = valuesRow[j + 1];
        String s3 = valuesRow[j + 2];
        String s4 = valuesRow[j + 3];
        String s5 = valuesRow[j + 4];
        double d1 = (Double.parseDouble(s1) - minlevel) / normolizer;
        //System.out.println("D1 "+d1 *normolizer + " S1 "+ s1);
        double d2 = (Double.parseDouble(s2) - minlevel) / normolizer;
        double d3 = (Double.parseDouble(s3) - minlevel) / normolizer;
        double d4 = (Double.parseDouble(s4) - minlevel) / normolizer;
        double d5 = (Double.parseDouble(s5) - minlevel) / normolizer;
       // System.out.print( "Actual"  + df2.format((d5*normolizer)));
         testSet.addElement(new TrainingElement(new double[]{d1}));
         loadedMlPerceptron.setInput(d1,d2,d3,d4);
          loadedMlPerceptron.calculate();

         //System.out.print(" Predicted "+
df2.format((loadedMlPerceptron.getOutput().firstElement())*normolizer));
```

```
                //error =((loadedMlPerceptron.getOutput().firstElement())-
(d5*normolizer)*normolizer);
                error = ((loadedMlPerceptron.getOutput().firstElement())*normolizer)-
(d5*normolizer);
               //System.out.print(" Error "+ df2.format(error));
               error = Double.parseDouble(df2.format(error));
               double actual = Double.parseDouble(df2.format(d5*normolizer));
               rmse+= (error*error);
               mad +=Math.abs(error);
                mape+=Math.abs(error/actual) *100;
                //System.out.println(" MAPE "+ Math.abs(error/actual) *100);
                 System.out.println("Actual"  + df2.format((d5*normolizer))+" Predicted "+
                       df2.format((loadedMlPerceptron.getOutput()
                               .firstElement())*normolizer)+" Error "+
                       df2.format(error)+" MAPE "+ Math.abs(error/actual) *100);
               // double mad2 = (error/actual)*100;
              // List output = null;
               //call function to add errors to list
               results(df2.format((d5*normolizer)),
                     df2.format((loadedMlPerceptron.getOutput()
                             .firstElement())*normolizer),
                     String.valueOf(error),String.valueOf((error/actual)*100));
           }
            setRenedered("true");
           sqrtrmse=sqrt((rmse/valuesRow.length));
            System.out.println(" Total RMSE  "+ df2.format(sqrtrmse));
               setRmse(String.valueOf(df2.format(sqrtrmse)));
            mad = (mad/valuesRow.length);
               setMadresult(String.valueOf(df2.format(mad)));
            mape = (mape/valuesRow.length);
            setMaperesult(String.valueOf(df2.format(mape)));
            System.out.println(" MAD  "+ df2.format(mad));
            System.out.println(" MAPE  "+ df2.format(mape));
             setPredsDM(output);

             System.out.println("Final list count == > "+ output.size());
           rtx.execute("PF('dlg3').hide()");
            ctx.addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO,
                     "Training was successful", ""));
           }

        }catch(Exception e){
            rtx.execute("PF('dlg3').hide()");
            ctx.addMessage(null, new FacesMessage(FacesMessage.SEVERITY_ERROR,
                     "An error occured, please try again", ""));
        }

        return page;
    }
```
Github link: https://github.com/timayabi2020/MaizePricePrediction