International Journal of Advanced Engineering Research and Science (IJAERS)
Peer-Reviewed Journal
ISSN: 2349-6495(P) | 2456-1908(O)
Vol-11, Issue-3; Mar, 2024
Journal Home Page Available: https://ijaers.com/
Article DOI: https://dx.doi.org/10.22161/ijaers.114.2

Machine Learning Model for Attenuating Outliers in Stock Data

[1]Njideka Nkemdilim Mbeledogu and [2]Kaodilichukwu Chidi Mbeledogu

[1]Department of Computer Science, Nnamdi Azikiwe University, Awka, Nigeria
[2]Department of Mechanical Engineering, University of Ottawa, Canada

Received: 07 Feb 2024,

Receive in revised form: 12 Mar 2024,

Accepted: 20 Mar 2024,

Available online: 30 Mar 2024

Keywords— ***Outliers, Uncertainty, Artificial Neural Network, Fuzzy Artificial Neural Network, Root Mean Square Error***

Abstract— *The presence of outliers has deleterious effects on the stock value because the unreliable information may discourage investors from investing in the stock. This associated problem paved way for the importance of intelligent prediction paradigm. However, existing stock forecasting models like Artificial neural network (ANN) performed better than traditional statistical models in handling the problem of non-linearity and complexity in forecasting stock price, it still lacked the capacity to handle outliers which are inherent in the stock market. Based on this, the researchers were motivated to forecast stock market based on the observation from literature that researchers do not report checking, prediction and proper management of outliers of any sort in stock market price forecasting. This paper addressed outliers' deleterious effects on the stock value by proposing a hybrid Fuzzy Artificial Neural Network (FANN) Model that attenuates outliers in stock forecasting accurately. The proposed model was simulated using MATLAB. The historical Nigerian stock quantitative datasets from Nigeria Stock Exchange (NSE) for 2008-2011 were used to test the simulated model. The Gaussian Membership function, due to its ability to handle minimum uncertainty principle, was used for the fuzzification of the extracted stock features to capture the stock dynamism. The proposed model's predictive performance was calculated using Root Mean Square Error (RMSE). The outlier detection analysis of the actual historical stock data, ANN and the proposed FANN Model predictions was calculated using Z-score. The proposed model had a RMSE value of about 3.83 which shows that it is a reliable stock forecasting model. The Z-score value of the proposed model was calculated to be about 0.78 which shows that it significantly attenuates outliers in stock forecasting. In overall, the results proved that the proposed FANN model can handle outliers in stock forecasting.*

I. INTRODUCTION

When dealing with real-world issues, one can rarely avoid uncertainty. At the experiential echelon, it cannot be avoided from any calculation, resulting from a combination of unavoidable experimental errors and limits of the calibrating instruments. At the mental rank, it is presented as a result from the vagueness and ambiguity that are fundamental in natural language. At the social stratum, it is originated and sustained by individuals for various needs (privacy, secrecy, propriety) while at economic level, taking uncertainty in stock can be caused by financial data characterized by erroneous data, outliers (Jatinder, 2012), geopolitical events, market sentiment and news. Whatever the level is, uncertainty affects the nation's economy by affecting individual companies and the economy as a whole respectively.

Uncertainty is an unclear multi-faceted characterization about data or predictions made from data that may include error, accuracy, validity, quality, noise and confidence and readability (Dungan et al., 2002). In any analyses, it can feature due to insufficient data, indistinct, conflicting, incoherent, not fully reliable, deficient in some way, erroneous, outliers , transnational events and information deficiency in market views and thus, results in different types of uncertainty.

Mendel (2001) stated that three types of uncertainty exists - *fuzziness* (or vagueness), which results from the imprecise boundaries of fuzzy sets; *nonspecificity* (or imprecision), which is connected with sizes (cardinalities) of relevant sets of alternatives; and *strife* (discord, outlier), which expresses conflicts among the various sets of alternatives. The researcher went further to group these into two major classes. They are *fuzziness* and *ambiguity*, where ambiguity (one-to-many relationships) encompasses *nonspecificity* and *strif*e.

II. LITERATURE REVIEW

Outliers are observations in the dataset that appear to be unusual and discordant (Yanfang, 2014). In Statistics, it is an observation that is significantly distant from the rest of the data. Hawkins (1980) defined it as an observation which deviates so much from the other observations as to spur suspicions that it was generated by a different mechanism. Aggarwal (2005) referred to them as abnormalities, discordant, deviants or anomalies in the data mining and statistics.

In many analyses, a relatively small size of outliers can disrupt even simple analysis because they can shoot up the error variance, thereby, lessening the power of statistical tests, normality in a non-randomly distributed situation by altering the propability of producing Type I and Type II errors and can greatly affect estimates that may be of valuable interest (Osborne and Overbay, 2004).

Based on research, many researchers gave different views on how best to handle outliers observed in a data. Barnett and Lewis (1994) proposed its removal when not in accordance with the other valid data. This notion was generally accepted by many researchers even in the situations when they are licit or has vague purpose. However, some researchers such as Orr *et al*. (1991) and Osborne and Overbay (2004) felt otherwise but suggested it as a triggering factor for investigation amidst its erroneous value that might contain valuable information in a more global sense (Osborne and Overbay, 2004). Based on this, Osborne (2002) opted that it is needful to use a transformation technique to keep the individual observation in the dataset as well as reduce its disruption in the statistical inference.

Forecasting and Circuit Breakers in Stock Market

A forecast is a prediction of some future event or events. It is a needful aspect of life that spreads across many fields such as business and industry, government, economics, environmental sciences, medicine, social science, politics, and finance. It is often categorized as short-term, medium-term, and long-term. Historical data are used in forecasting because they display inactiveness to change fast. Securities are traded in the Stock Market. The last price at which it is traded in a day gives the most up-to-date valuation of that security until the next trading day begins (Investopedia, 2013). This last price is termed Closing Price. Close Price is needful in stock market because it shows a valuable benchmark for investors to check differences in stock prices over time such that measuring the market opinion for a given security over a trading day, the closing price of one day can be compared to the preceding day. For investors to predict the closing stock price, daily historical stock data are collected.

Companies normally forecast sales and trends in production for the investors to follow with the believe that the market conditions are normal but the observed variables (predicting/attributes) usually contain outliers. If a stock technical indicator contains outliers, the incorrect evaluation from contaminated observations may be highly misrepresented, thus leading to unreliable results (Yanfang, 2014). Based on this, there is need to identify the outliers using the circuit breakers that is within $\pm 10\%$ by the Security and Exchange Commission (SEC) (Ohuche and Ikoku, 2015) and minimize their discordance. Therefore, the key part of this stock data analysis is the detection and proper handling of outliers.

Artificial Neural Network (ANN)

ANNs provide a way to emulate biological neurons to solve complex problems in the same manner as the human brain. It is an enormously parallel distributed processor that has a natural capacity for storing experiential knowledge and making it available for use (Haykin, 1998). It is capable of learning because it is modeled after the human brain (Haag *et al*., 1998). Though it is a mathematical model of information processing, ANN is relatively different from turing machines with stored programs (Fausett, 1996). Its information processing system (Figure 1) is developed based on the idea of mathematical models of the brain's cognition on the belief that the capability of processing information occurs at its simple elements called neurons.

The signals are transmitted amidst these neurons through the connection links. Every connection link has an associated weight that multiplies the transmitted signal. The output signal is determined by an activation function (usually nonlinear) of each neuron upon the sum of the weighted input signals (net input).

ANN offers so many benefits. Some of such benefits are nonlinearity, input-output Mapping, adaptation, evidential response, contextual information, fault tolerance, VLSI Implementation, uniformity of analysis and design, and neurobiological analogy.

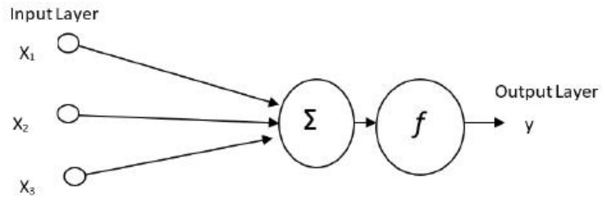

Fig.1: Single layer feed forward network

The mapping of single unit perceptron can be expressed as:

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \quad \text{Eq. 1}$$

Where w_i are the individual weights, x_i are the inputs and b is the bias.

REVIEW OF RELATED WORKS

Moghar and Hamiche (2020) predicted the future stock market values using Long-Short Term Memory (LSTM) and Recurrent Neural Network. Also, they determined which precision a machine learning algorithm can predict and how much epochs can improve their model.

Noel (2023) employed dynamic neural networks to predict the future closing stock price using two prevailing theories that sought to qualify market behavior. The two theories were the Efficient Market Hypothesis (EMH) and Chaos.

Thakkar and Chaudhari (2021) delved into a comprehensive survey on deep neural networks for stock market. The authors presented the need, challenges, future directions and the applicability of deep neural network variations to the temporal stock market data and NN meta-heuristics approaches were discussed.

III. RESEARCH METHODOLOGY

Data and Sources of data

The daily stock data of three randomly selected companies from the 27 blue chip companies in Nigeria were collected from the Nigeria Stock Exchange from 2008-2011. The companies are Dangote Sugar Refinery, GlaxoSmith Kline Consumer Plc and Julius Berger Construction Company.

The inputs into the system were four stock variables (Open price, High price, Low price and Close price) that are independent in nature.

Outlier Detection

When the sample data is drawn from a non-normally distributed population and the sample size is large enough (that is, ≥ 30), a standard normal test can be employed. This was empowered by the Central Limit Theorem (CLT) which states that if random variable S_n is defined as the sum of n independent and identically distributed (i.i.d) random variables, $X_1, X_2, \dots, X_{n;}$ with mean,μ and standard deviation, σ. Then, for large enough n (typically $n \geq 30$), S_n is approximately normally distributed with parameters : $\mu_{sn} = n\mu$ and $\sigma_{sn} = \sqrt{n\sigma}$. This result holds regardless of the shape of the X distribution (that is, X do not have to be normally distributed (Eze *et al*., 2005; Filmus, 2010).

Therefore, so long as n is large and are independent variables, by CLT, the Z-score (Z-test) is distributed as standard normal and was adopted.

Z-score

The basic idea of this rule is that if X follows a normal distribution, N (μ, σ^2), then Z follows a standard normal distribution, N (0, 1), and Z-scores that exceed 3 in absolute value are generally considered as outliers. This method is simple and it is the same formula as the 3 standard deviation method when the criterion of an outlier is an absolute value of a Z-score of at least 3.

The actual close stock price data was used because the scope of the prediction was the next day's closing price.

The Z-score can be computed as

$$Z_i = \frac{ABS\ (X_i - \mu)}{\sigma} \quad \text{Eq. 2}$$

Where X_i is the value, μ is the mean and σ is the standard deviation.

Outlier Analysis of the Actual Stock Dataset

Three different stock datasets: Dangote Sugar [2008-2011 (938 data values)], GlaxoSmith Kline Consumer Plc [2012 (240 data values)] and Julius Berger [2012 (237 data values)] were used. Table 1 shows sample of the stock dataset and the detected outliers using Z-score (Z_i) based on the parameter analysis stated in Table 2. Figure 2 depicts the polar plot of Dangote Sugar.

Table 1: Outlier detection of the sample stock dataset

Dangote Sugar	Z_i **DS**	**GlaxoSmith Kline Cons. Plc**	Z_i **GS**	**Julius Berger**	Z_i **JB**
15	0.4237941	23	0.563775983	30.99	0.558068152
15.11	0.406774133	23	0.563775983	31.06	0.577889533
15.25	0.385112356	23	0.563775983	31.06	0.577889533
15.2	0.392848705	23	0.563775983	31.06	0.577889533
15.2	0.392848705	23	0.563775983	31.06	0.577889533
15.01	0.422246831	21.85	0.6901164	31.06	0.577889533
115.3	**15.09532188**	21.85	0.6901164	32.61	1.016791541
15.55	0.338694262	22.9	0.574762106	34.24	1.478346557
15.49	0.347977881	23	0.563775983	**0**	**8.217140398**
15.46	0.35261969	23	0.563775983	34	1.410387536
15.99	0.270614391	23	0.563775983	34	1.410387536
16	0.269067122	23	0.563775983	34	1.410387536
16	0.269067122	**0**	**3.090584321**	33.01	1.130056576
16.31	0.221101758	22.7	0.596734353	30.61	0.450466369

Where DS is Dangote Sugar, GS is GlaxoSmith and JB is Julius Berger.

Table 2: Outlier Parameter Analysis

Stock	μ	σ
Dangote Sugar	17.73898	6.462997
GlaxoSmith Kline Consumer Plc	28.13171	9.102392
Julius Berger	29.01916	3.53154

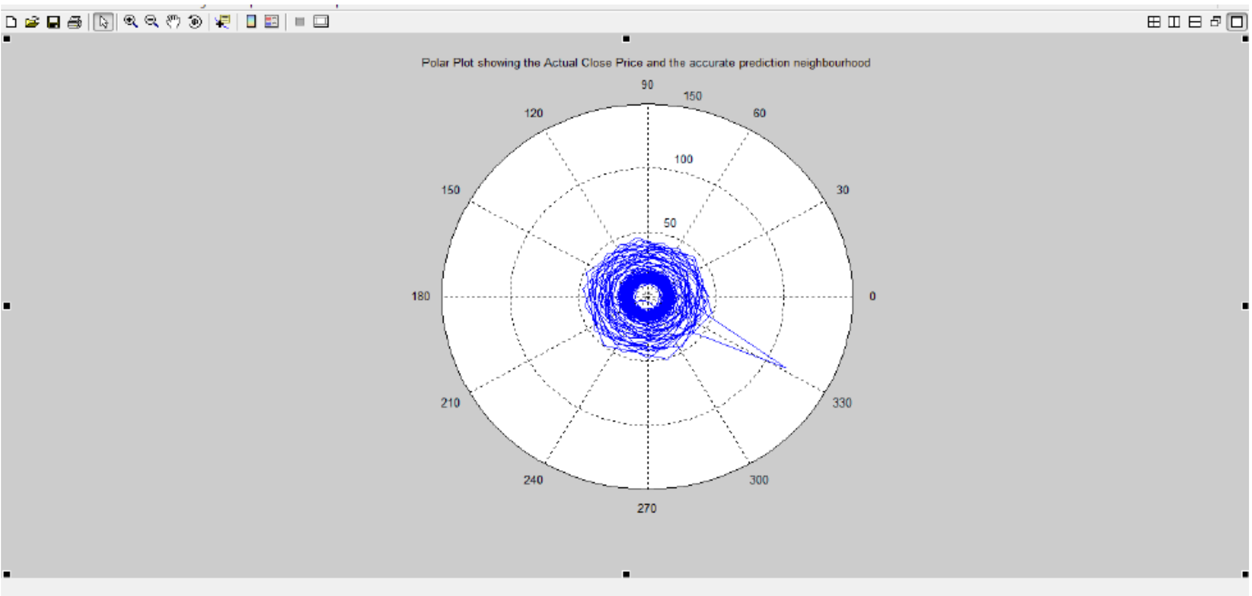

Fig.2: Polar plot showing the Actual close price for Dangote Sugar and accurate neighbourhood of prediction

From Table 1, it will be observed that an outlier was detected from each of the three data sets. For Dangote Sugar, the stock data value 115.30 with Z_i of 15.09532188, stock data 0 of GlaxoSmith Kline Consumer Plc with Z_i of 3.090584321 and stock data 0 of Julius Berger with Z_i of 8.217140398. Figure 1 depicts the Polar plot showing the Actual Close Price for Dangote Sugar. Polar plot gives the position of a point in a 2D surface (Tsishchanks, 2010). This shows how the data are clustered thereby showing the accurate neighbourhood of prediction and the outlier.

FANN Modeling

The FANN model was simulated using MATLAB. Figure 3 shows the model diagram of FANN that hybridized Fuzzy and ANN. Four stock inputs (Open price, High price, Low price and Close price) were used.

Fuzzy Modeling in FANN

The Gaussian membership function was used to transform the crisp stock data into fuzzy values (fuzzification). The choice of membership was with the aim of capturing the stock dynamism by the function's smoothness at the edges.

The Gaussian Membership Function is calculated as follows:

$$f(x_i, \sigma, m) = \exp\left\{-\frac{(x_i - m)^2}{2\sigma^2}\right\} \qquad \text{Eq. 3}$$

Where m represents the mean value, σ represents the standard deviation for a given membership function and x_i represents the raw stock training data.

$$net_i = x_i \qquad \text{Eq. 4}$$

$$out_j = f(net_i, \sigma_{ij}, m_{ij}) \qquad \text{Eq. 5}$$

Where out_j represents the output corresponding to the j^{th} membership function that corresponds to the input x_i.

Rule Generation

The Multiple input multiple output (MIMO) form of representing the expert knowledge was used such that:

Fact: μ_1 is A_1^i and μ_2 is A_2^i and … and μ_n is A_n^i

Rule R_{ij}: If μ_1 is A_1^i and μ_2 is A_2^i and … and μ_n is A_n^i then T_1 is C_1^j, T_2 is C_2^j, …, T_m is C_m^j, W_{ij}

Result: T_1 is C_1^j, T_2 is C_2^j, …, T_m is C_m^j

Where $u_1, \dots, u_n$ are the stock input linguistic variables (process state variables) and $T_1, \dots, T_m$ are the stock control Linguistic variables, A_1^i, …, A_n^i and C_1^j, …, C_m^j are the stock linguistic values of the linguistic variables $u_1, \dots, u_n$ and $T_1, \dots, T_m$ in the stock universe of discourse X and Y. The rules are weighted such that the degree of confidence with which the stock input fuzzy set A_1^i, …, A_n^i (which is composed of fuzzy intersection (AND) of several univariate stock fuzzy sets) is related to the stock output fuzzy set C_1^j, …, C_m^j is given by $W_{ij} \in [0,1]$. When W_{ij} is zero, the rule is inactive and does not contribute to the output. Otherwise, it partially fires whenever its antecedent is activated to a degree greater than zero. R_{ij} represents the stock rule number.

Defuzzification

Center of Gravity (CoG) which is also called Center of Area (CoA) defuzzification method was employed to transform the fuzzified data into its equivalent crisp form. The centroid of the area bounded by the controller output MF is determined and its abscissa taken as the crisp controlling value. Its computation is given as:

$$CoA\ [C(Z)] = \frac{\sum_{i=1}^{q} Z_i C(Z_i)}{\sum_{i=1}^{q} C(Z_i)} \qquad \text{Eq. 6}$$

where q is the number of sample values of the stock dataset and Z_i is the value of the control output at the sample value. Because they are four inputs of the stock control output MF, it resulted to four groups of CoA defuzzification groups.

ANN modeling in FANN

A Multi-layer perceptron model (MLP) was used because it can be trained to approximate most functions arbitrarily well while Single-Layer networks cannot. The MLP comprised of four layers – input layer, two hidden layers and the output layer. The output from the fuzzy formed the inputs into the ANN. The financial time series under consideration is highly non-linear and requires a sufficiently non-linear function to represent all the properties of the series. Log-sigmoid and Purelin activation functions were deployed within the layers because they can be trained to approximate most functions arbitrarily well (Hagan *et al.*, 1996).

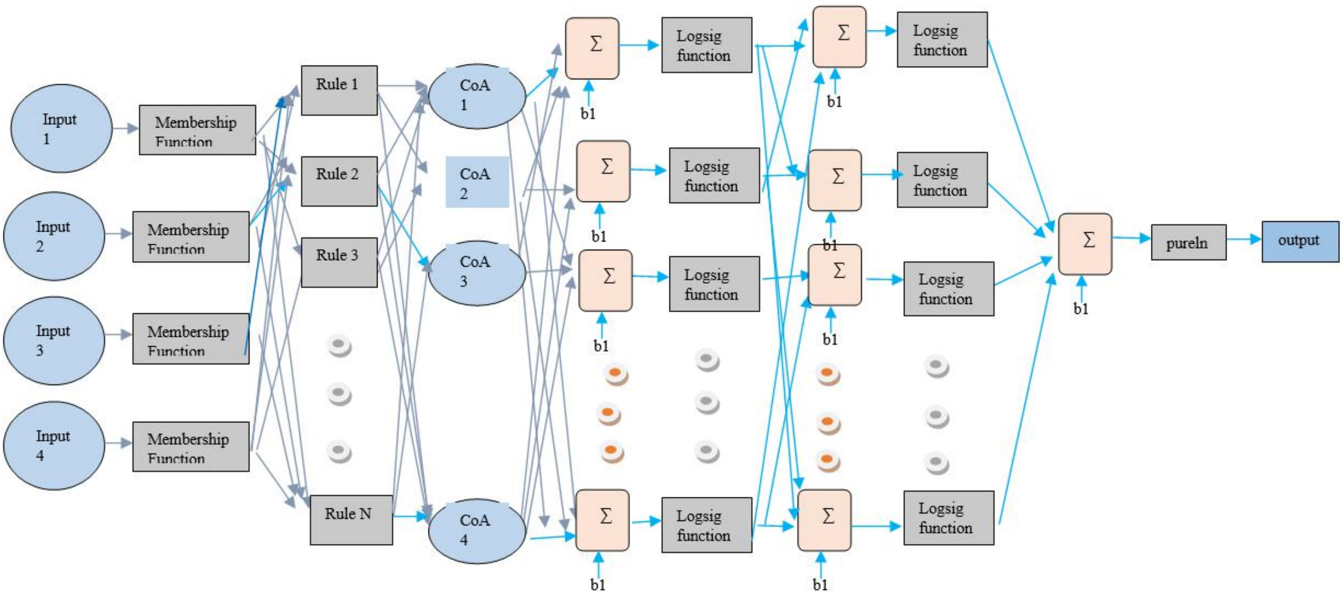

Fig.3: High Level Model of FANN Architecture

Algorithm

Training and learning functions are mathematical procedures used to automatically adjust the network's weights and biases. The training function dictates a global algorithm that affects all the weights and biases of a given network while the learning function can be applied to individual weights and biases within a network (Mathworks, 2014).

The system deployed a supervised paradigm by adopting Levenberg Marquardt (LM) back propagation for training and Gradient Descent (GD) for learning.

Step 1: The defuzzified data are the inputs to the neural network. Propagate the input forward through the network by selecting random weights and biases:

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}) \quad \text{for} \quad m = 0,1,\ldots,M-1 \qquad \text{Eq. 7}$$

Step 2: Calculate the errors

$$e_q = t_q - a_q^M \qquad \text{Eq. 8}$$

Where e_q is the error, t_q is the target value and a_q^M is the output.

Step 3: Compute the sum of squared errors over all inputs, $F(x)$:

$$F(x) = \sum_{q=1}^{Q}(t_q - a_q)^T (t_q - a_q) \qquad \text{Eq. 9}$$

$$= \sum_{q=1}^{Q} e_q^T e_q \qquad \text{Eq. 10}$$

$$= \sum_{q=1}^{Q}\sum_{j=1}^{s^m}(e_{j,q})^2 \qquad \text{Eq. 11}$$

$$= \sum_{i=1}^{N}(v_i)^2 \qquad \text{Eq. 12}$$

Where $e_{j,q}$ is the jth element of the error for the qth input/target pair and v is the error vector.

$$v^T = [v_1\ v_2\ \ldots\ v_N] \qquad \text{Eq. 13}$$

$$= [e_{1,1}\ e_{2,1}\ \ldots\ e_{s^m,1}\ e_{1,2}\ \ldots e_{s^m,Q}] \qquad \text{Eq. 14}$$

Step 4: Compute the Jacobian matrix J(x):

$$J(x) = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_{1,1}^1} & \cdots & \frac{\partial e_{1,1}}{\partial b_1^1} \\ \vdots & \ddots & \vdots \\ \frac{\partial e_{s^m,1}}{\partial w_{1,1}^1} & \cdots & \frac{\partial e_{s^m,1}}{\partial b_1^1} \end{bmatrix} \qquad \text{Eq. 15}$$

Step 5: Calculate the sensitivity with the recurrence relations

Initialize the back propagation with

$$S_q^{-m} = -F^m(n_q^m) \qquad \text{Eq. 16}$$

Step 6: Propagate each column of the matrix S_q^{-m}

$$S_q^{-m} = F^m(n_q^m)(W^{m+1})^T S_q^{-m+1} \qquad \text{Eq. 17}$$

Step 7: Augment the individual matrices into Marquardt sensitivities:

$$S^{-m} = \langle S_1^{-m} | \ldots | S_Q^{-m} \rangle \qquad \text{Eq. 18}$$

Note: For each input presented to the network, the sensitivity vectors are propagated back. This is because the derivatives of each individual error is computed and not the derivative of the sum of squares of the errors. For every input applied to the network there will be S^m errors (one for each element of the network output). For each error, there will be one row of the Jacobian matrix.

Step 8: Compute the elements of the Jacobian matrix:

$$[J]_{h,l} = \frac{\partial v_h}{\partial x_l} = \frac{\partial e_{k,q}}{\partial w_{i,j}^m} = \frac{\partial e_{k,q}}{\partial n_{i,q}^m} x \frac{\partial n_{i,q}^m}{\partial w_{i,j}^m} = s_{i,h}^{-m} x \frac{\partial n_{i,q}^m}{\partial w_{i,j}^m} = s_{i,h}^{-m} x a_{j,q}^{m-1} \quad \text{Eq. 19}$$

$$[J]_{h,l} = \frac{\partial v_h}{\partial x_l} = \frac{\partial e_{k,q}}{\partial b_i^m} = \frac{\partial e_{k,q}}{\partial n_{i,q}^m} x \frac{\partial n_{i,q}^m}{\partial b_i^m} = s_{i,h}^{-m} x \frac{\partial n_{i,q}^m}{\partial b_i^m} = s_{i,h}^{-m} \quad \text{Eq. 20}$$

if x_l is a bias

Step 9: Solve to obtain Δx_k

$$\Delta x_k = -[J^T(x_k)J(x_k) + \mu_k I]^{-1} J^T(x_k)v(x_k) \quad \text{Eq. 21}$$

Step 10: Re-compute the sum of squares errors using $x_k + \Delta x_k$. If this new sum of squares is smaller than the value computed in step 1, then divide μ by v, let

$$x_{k+1} = x_k + \Delta x_k \quad \text{Eq. 22}$$

and go back to step 1. If the sum of squares is not reduced, then multiply μ by v and go back to step 9.

The algorithm is assumed to have converged when the norm of the gradient is less than some predetermined value, or when the sum of squares has been reduced to some error goal.

$$\nabla F(x) = 2J^T(x)v(x) \quad \text{Eq. 23}$$

Learning Algorithm

Gradient descent learning algorithm with learning rate and momentum coefficient were used for the learning. It was implemented using the Learngdm Function.

Training

Functional tests and comparisons presented in Tables 3, 5 and 7 are the actual close price of the stock data, the predictions of ANN and FANN, and the Z-score for the actual stock price, Dangote Sugar, GlaxoSmith Kline and Julius Berger datasets respectively based on their outlier parameter analysis presented on Tables 4, 6 and 8. Figures 4 and 5 show the graphical representation of FANN predictions and the Polar plot showing the Actual close price and FANN Model for Dangote Sugar respectively.

Table 3: Actual close price with ANN and FANN and their z-scores for Dangote Sugar

Days	Actual Close Price	Z_i of Actual Close Price	ANN Predictions	Z_i of ANN	FANN prediction	Z_i of FANN
518	15.20	0.385112356	15.25	0.422214959	15.05	0.544055695
519	15.20	0.392848705	15.20	0.427652245	15.00	0.550129038
520	15.01	0.392848705	15.20	0.427652187	14.99	0.55141744
521	115.30	0.422246831	15.01	0.448312847	**14.85**	0.570113596
522	15.55	**15.09532188**	**115.30**	**10.45726241**	25.00	**0.780386738**
523	15.49	0.338694262	15.55	0.389593024	25.00	0.780386738
524	15.46	0.347977881	15.49	0.396117468	15.29	0.511555605
525	15.99	0.35261969	15.46	0.399379704	15.21	0.522823472
526	16.00	0.270614391	15.99	0.341747185	15.67	0.461188991

Table 4 : Outlier Parameter Analysis of Dangote Sugar

System	μ	σ
Actual Close Price	17.73898	6.462997
ANN	19.13278063	9.196213649
FANN Model	19.13458208	7.515948831

Table 5: Actual close price with ANN and FANN predictions of GlaxoSmith Kline Plc

Days	**Actual Close Price**	**Zi of Actual Close Price**	**ANN Predictions**	Z_i **of ANN**	**FANN Predictions**	**Zi of FANN**
6	21.85	0.6901164	22.99999	0.555166773	22.44521	0.678771729
7	21.85	0.6901164	21.85	0.682216357	22.44519	0.678774061
8	22.9	0.574762106	21.84999	0.682217461	23.37474	0.553814392
9	23	0.563775983	22.9	0.566213554	23.45614	0.542871598
10	23	0.563775983	23	0.555165668	23.45614	0.542871598
11	23	0.563775983	23	0.555165668	23.45614	0.542871598
12	23	0.563775983	23	0.555165668	23.45614	0.542871598
13	0	3.090584321	23	0.555165668	13.35331	1.901003085
14	22.7	0.596734353	0.183169	3.075943133	23.20793	0.576238447
15	22.7	0.596734353	22.70001	0.588308221	23.20793	0.576238447
16	22.7	0.596734353	22.70001	0.588308221	23.20793	0.576238447
17	22.7	0.596734353	22.70001	0.588308221	23.20793	0.576238447

Table 6: Outlier Parameter Analysis for GlaxoSmith Plc

System	μ	σ
Actual Close Price	28.13171	9.102392
ANN	28.02508508	9.051505466
FANN Model	27.49443484	7.438769163

Table 7: Actual close price with ANN and FANN predictions for Julius Berger.

Days	Actual Close Price	Zi of Actual Close Price	ANN Predictions	Z_i of ANN	FANN Predictions	Zi of FANN
87	31.06	0.577889533	31.06012	-1.417360474	30.89801	0.642466247
88	31.06	0.577889533	31.06012	-1.417360474	30.89801	0.642466247
89	31.06	0.577889533	31.06012	-1.417360474	30.89801	0.642466247
90	31.06	0.577889533	31.06012	-1.417360474	30.89801	0.642466247
91	31.06	0.577889533	31.06027	-1.417210474	30.89825	0.642542611
92	32.61	1.016791541	32.61015	0.132669526	32.8368	1.259353294
93	34.24	1.478346557	34.24013	1.762649526	**34.50837**	1.79121589
94	**0**	**8.217140398**	**2.078105**	**23.48781559**	**13.38482**	**4.929906245**
95	34	1.410387536	34.00015	1.522669526	34.32237	1.732034137
96	34	1.410387536	34.00015	1.522669526	34.32237	1.732034137
97	34	1.410387536	33.99997	1.522489526	34.09875	1.660882397
98	33.01	1.130056576	33.01014	0.532659526	33.29709	1.405809042

Table 8: Outlier Parameter Analysis for Julius Berger

System	μ	σ
Actual Close Price	29.01916	3.53154
ANN	29.02170053	3.455779944
FANN Model	28.87882814	3.142860608

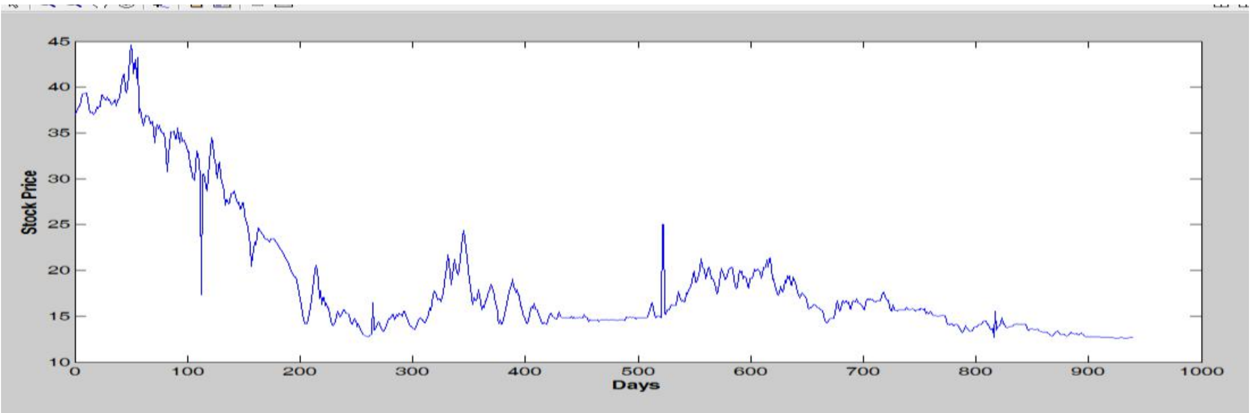

Fig.4: Graphical representation of FANN predictions for Dangote Sugar

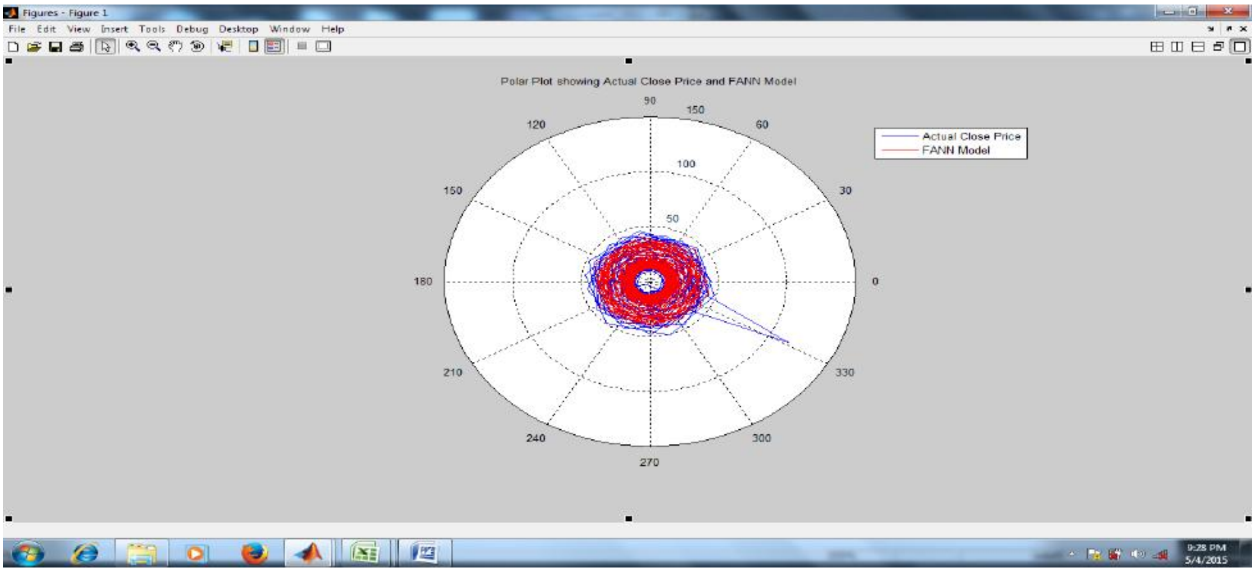

Fig.5: Polar plot showing Actual close price and FANN Model for Dangote Sugar

IV. RESULTS AND DISCUSSION

From Tables 3, 5 and 7, both systems can be used for predictions. Figures 3 and 4 show the graphical predictive capability and mitigating power of FANN respectively. The results analysis was based on the predictive performance using error analysis and its mitigation capability.

Error Analysis

The proposed model's predictive performance was evaluated using the Root Mean Square Error (RMSE) and compared with ANN as shown on Table 9.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs,i}-X_{model,i})^2}{n}} \qquad \text{Eq.24}$$

Where $X_{obs,i}$ is observed values and $X_{model,i}$ is modeled values at time.

Table 9: Performance Evaluation using RMSE for Dangote Sugar

Model	RMSE
ANN	4.7308
FANN	3.8321

The proposed model returned a RMSE value of about 3.83 which shows that it competes favourably with the existing systems and it is a reliable stock forecasting model.

Mitigation Capability

From Tables 3 and 5, the outliers in the actual stock price were significantly reduced to values that returned very low Z-scores of about 0.78 and 1.90 respectively, that is, to the accurate neighborhood of prediction that are not outliers. From Table 7, the Z-score of FANN Model prediction for Julius Berger was significantly reduced from 8.217140398 to 4.929906245. This shows that the proposed model significantly attenuates outliers in stock forecasting. Overall, the results proved that the proposed FANN Model can handle outliers in stock forecasting. Therefore, with this model, stock forecasting will be more accurate and hence, more reliable.

V. CONCLUSION

Stock forecasting has always been an active area of research because successful prediction of a stock's future price could yield significant profit, thereby affecting the entire nation's economy. In order to handle outliers that are inherent in stock data as well as predict stock price on normal condition, a new solution was proposed. The proposed solution combined approaches in soft computing - Fuzzy and ANN to automatically find patterns from trading data. The idea of their hybridization originated from the fact that fuzzy logic is appropriate for uncertainty and ANN with the aim of learning.

The experimental results indicated the combinational strength of fuzzy logic with artificial neural network in attenuating outliers. Therefore, a model now exists that significantly attenuates outliers in stock forecasting.

REFERENCES

[1] Aggarwal, C.C. (2005). On Abnormality Detection in Spuriously Populated Data Streams SIAM Conference on Data Mining, Kluwer Academic Publishers Boston London.

[2] Barnett, V. and Lewis, T. (1994). Outliers in Statistical data. John Wiley & Sons, 3rd edition, Kluwer Academic Publishers Boston London.

[3] Dungan, J.L., Gao, D and Pang, A.T (2002). Definitions of uncertainty. Retrieved from ftp://.cse.ucsc.edu/pub/reinas/papers/white.pdf

[4] Eze, J.I, Obiegbu, M.E and Jude-Eze, E.N. (2005). Statistics and Quantitative Methods for Construction and Business Managers, The Nigerian Institute of Building, pp. 1- 402.

[5] Fausett, L. (1996). Fundamentals of Neural network: Architectures, Algorithms and Applications, Prentice Hall, Upper Saddle River, New Jersey 07458, pp. 1-14.

[6] Filmus, Y. (2010). Two Proofs of the Central Limit Theorem. Retrieved from www.cs.toronto.Edu/~yuvalf/CLT.pdf.

[7] Haag, S., Cummings, M. and Dawkins, J. (1998). Management Information System for the information Age, Mc-Graw-Hill, USA, pp. 526.

[8] Hagan, M.T., Demth, H.B. and Beale, M. (1996). Neural network design, PWS Publishing company.

[9] Hawkins, D. (1980). Identification of Outliers, Chapman and Hall, Kluwer Academic Publishers Boston London.

[10] Haykin, S. (1998). Neural networks: A comprehensive Foundation, Macmillan College Publishing company, Inc. USA, pp. 1-41.

[11] Investopedia (2013). Stock Market. Retrieved from www.investopedia.com/terms/s/stock market.asp#axzz2IjJu5E13

[12] Jatinder, R. P. (2012). Root cause Analysis of IMRT QA outliers. Retrieved from www.aapm.org/meetings/amos2/pdf35-9828-49165-839.pdf

[13] Mathworks (2014). The mathworks. Help Guide. Mathworks [online] mathworks. Retrieved from www.mathworks.com/products/neural.network/features.html

[14] Mendel, J.M (2001). Uncertainty in fuzzy logic systems. Retrieved from www.inFormit.com/articles/article.aspx?=21313

[15] Moghar, .A. and Hamiche, .M. (2020). Stock Market Prediction using LSTM Recurrent Neural Networks, Procedia Computer Science 170, pp. 1168-1173

[16] Noel, .D.(2023). Stock Price Prediction using Dynamic Neural Networks, Computational Engineering, Finance and Science (cs.CE)

[17] Ohuche, F.k. and Ikoku, A.E. (2015). Financial Management Focus on Price Volatility and Circuit Breakers in the Nigerian Equity Market Implications for Monetary Policy. Journal of Financial Management and Analysis Vol. 27 No. 2.

[18] Orr, J.M., Sacket, P.R. and DuBois, C.L. (1991). Outlier detection and treatment in I/OPsychology: A Survey of researcher beliefs and an empirical illustration. Personnel Psychology, No. 44, pp. 73-486.

[19] Osborne, J.W. and Overbay, A. (2004). The Power of Outliers (and why researchers should Always check for them). Practical Assessment Research and Evaluation, Vol. 9, No. 6.

[20] Osborne, J.W. (2002). Notes on the use of data transformations. Practical Assessment Research and Evaluation, Vol. 8(6).

[21] Songwon, S.M.S. (2006). A Review and Comparison of Methods for detecting outliers in Univariate Data Sets. Retrieved from d-scholarship.pitt.edu/7948/1/seo.pdf.

[22] Thakkar, .A. and Chaudhari, .K. (2021). A Comprehensive Survey on Deep Neural Network for Stock Market: The need, Challenges and Future Directions, Expert System with Applications, Vol. 177
[23] Tsishchanks, K. (2010). Polar Coordinates. Retrieved from https://cims.nyu.edu/~kiryl /calculus/section_9.3--polar_coordinates/polar_coordinatinates.pdf
[24] Yanfang, L. (2014). Detection of Outliers in Panel Data of Intervention Effects Model Based on Variance of Remainder Disturbance, Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2015, Article 10902602.