

A Conceptual Nigeria Stock Exchange Prediction: Implementation Using Support Vector Machines-SMO Model

Abubakar S. Magaji

Department of Mathematical Sciences, Faculty of Science, Kaduna State University, Kaduna-Nigeria.

Audu Isah

Department of Mathematics & Statistics, SSSE, Federal University of Technology Minna Nigeria.

Victor Onomza Waziri

Department of Cyber Security, School of ICT, Federal University of Technology Minna-Nigeria.

Adeboye K.R.

Department of Mathematics & Statistics, SSSE, Federal University of Technology Minna Nigeria.

Abstract—This paper is a continuation of our research work on the Nigerian Stock Exchange (NSE) market uncertainties, In our first paper (Magaji et al, 2013) we presented the Naive Bayes algorithm as a tool for predicting the Nigerian Stock Exchange Market; subsequently we used the same transformed data of the NSE and explored the implementation of the Support Vector Machine algorithm on the WEKA platform, and results obtained, made us to also conclude that the Support Vector Machine-SOM is another algorithm that provides an avenue for predicting the Nigerian Stock Exchange.

Keywords- Nigerian Stock Market; Prediction; Data Mining; Machine Learning; Support Vector Machine.

I. INTRODUCTION

It is a common tradition to note that a vast amount of capital is traded through the Stock Markets in each country and around the World. The performance of each country at the stock exchange Market gives an insight to the economic growth of that country; generally known as the GDP.

Nigerian Stock Exchange (NSE) is definitely an avenue for trading shares and bonds, but it also has a great influence on the Nigerian Economy. It serves as a platform where industries and commercial ventures converges to raise the public capital that paves way for them to expand, and in the process they create new jobs, products, services and opportunities.

As observed during the recent Economic melt-down, one of the specific characteristics that all Stock Markets have in common is the uncertainty, which is related with their short and long-term future state. This feature is undesirable for the investor but it is also unavoidable whenever the Stock Market is selected as the investment tool. The best that one can do is to try in reducing this uncertainty, or if possible wipe it out completely. Stock Market Prediction (or Forecasting) is one of the instruments in the process of achieving this dream.

There is no doubt that the majority of the people related to stock markets are trying to achieve profit. Profit comes by investing in stocks that have a good future (short or long term

future) [1]. Thus what they are trying to accomplish one way or the other is to predict the future of the market. But what determines this future?

Predictions based on different models include some of the followings: Autoregressive Moving Average (ARMA), Random Walk (RW), Neural Network (NN), are being exploited now; all in an effort to improve the predictions, and thus, make reasonable indicial pronouncements that can guides the economic growth of a country. Other models that are useful in the forecasting of the economic indices are Naive Bayes (NB) and Support Vector Machines (SVM) which are useful in classifications or prediction of GDP.

The focus of our paper is time series forecasting, we intend to carry out prediction of the Nigerian Stock Market using Support Vector Machines. Our study of the Stock Exchange Market is limited to Nigerian reference frame of Nigerian Stock Exchange Market. In view of this all our training data set shall be acquired within the Nigerian Stock Exchange reference context. Our studies border models shall also be limited to the Support Vector Machines model.

II. RELATED WORKS

In the literature the data that are related to the stock markets are divided in three major categories [2]:

(1) Technical data: are all the data that are referred to stocks only. Technical data include:

- The price at the end of the day.
- The highest and the lowest price of a trading day.
- The volume of shares traded per day.

(2) Fundamental data: are data related to the intrinsic value of a company or category of companies as well as data related to the general economy. Fundamental data include: Inflation, Interest Rates, Trade Balance, Indexes of industries (e.g. heavy industry), Prices of related commodities (e.g. oil, metals, and currencies), Net profit margin of a firm, and Prognoses of future profits of a firm.

(3) Derived data: this type of data can be produced by transforming and combining technical and/or fundamental data.

Many experts in the stock markets have employed the technical analysis for better prediction for a long time. Generally speaking, the technical analysis derives the stock movement from the stock's own historical value. The historical data can be used directly to form the support level and the resistance or they can be plugged into many technical indicators for further investigation. Conventional research addressing this research problem have generally employed the time series analysis techniques (i.e. mixed Auto Regression Moving Average (ARMA)) [3] as well as multiple regression models. Considerable evidence exists and shows that stock market price is to some extent predictable [4].

Within the context of Machine Learning, researchers have dwelt into a number of areas for prediction purposes; these includes Applied Mathematics (Yue et al.,2008, Berwald et al., 2011, Khudabukhsh et al.,2012), Business and Finance (Pompe et al., 1997, Holmes et al., 1998, Huang et al., 2005, Shah,2007, Kinlay et al.,2008, Fletcher et al,2008) , Computer Science (Boetticher,1994, Murray et al.,2005, Singh et al., 2006, Gammerman et al.,2007, Andrzejak et al., 2008, Alonso et al., 2010, Kumar et al., 2012, Haffey, 2012, Malhotra et al., 2012), Life Science (Muggleton et al., 1992, Calder et al., 1996, Demser et al., 2005, Cruz et al., 2006, Garzon et al., 2006, Cheng et al., 2008, Nugent, 2010, Kruppa et al., 2012, Qataweh et al., 2012, Miller,2012), Energy (Gross et al., 2005, Arnold et al., 2006, Sharma et al., 2011), Sports (Lyle,2005, Joseph et al.,2006, Warner, 2010, Davis et al,2012), etc.

A closer look at the previous works done in predicting stock markets reveals Shah,2007 [5] whose paper discusses the application of Support Vector Machines, Linear Regression, Prediction using Decision Stumps, Expert Weighting and Online Learning in detail along with the benefits and pitfalls of each method. The main goal of the project was to study and apply as many Machine Learning Algorithms as possible on a dataset involving a particular domain, namely the Stock Market, as opposed to coming up with a newer (and/or better) algorithm that is more efficient in predicting the price of a stock.

Kinlay et al., 2008 [6] they applied SVM techniques to forecast market direction in the S&P 500 index, and also used a competitive model framework provided by the 11Ants modeling system to select the best performing combinations of

non-linear models employing a variety of non-linear classification techniques. Fletcher et al,2008 [7] used the following algorithms, Artificial Neural Networks (ANN), Support Vector Machines (SVM) and Relevance Vector Machines (RVM) to predict daily returns for an FX carry basket.

As for the most recent works that featured to predict Nigerian Stock Markets, three out of five conducted researches (2009 -2011) used the ANN models. Akinnwale et al (2009) [8] used error back propagation algorithm and regression analysis to analyze and predict untraslated and translated Nigerian Stock Market Price (NSMP). Based on the findings of the study, translated NSMP prediction approach was more accurate than untranslated NSMP using either regression analysis or error back propagation algorithm.

Olabode et al [9] presented the use of a neural network simulation tool for stock market price, where various neural models like Multi-Layered Perceptron (MLP), Radial Basis Function (RBF), Generalized Regression Neural Networks (GRNN), Generalized Feed Forward Neural Networks (GFFNN) and Time Lagged Recurrent Networks (TLRN) were tested. The TLRN network architecture with one hidden layer and five processing elements was able to model the problem, as it came out to be the best model with good generalization capability.

Bello et al (2011) utilized ANNs model to predict closing price of AshakaCem Security in Nigerian Stock Market price index. They employed Feed Forward Artificial Neural Network (FFANN) Architecture and obtained results, which were evaluated on four performance indicators [Mean Square Error (MSE), Correlation Coefficient (r), Normalize Mean Square Error (NMSE) and Mean Absolute Error (MAE)] [10].

Whereas Agwuegbo et al (2010) urged that "The daily behaviour of the market prices revealed that the future stock prices cannot be predicted based on past movements" [11]. Though the result from the study provided evidence that the Nigerian stock exchange is not efficient even in weak form and that NSE follow the random walk model; thus concluded that Martingale defines the fairness or unfairness of the investment and no investor can alter the stock price as defined by expectation. The other work found in the literature, that did not make use of the ANN is the one presented by Emenike K.O (2010) [12], the Autoregressive Integrated Moving Average (p,d,q) model [ARIMA] was used to models and forecasts stock prices of the Nigerian Stock Exchange. The predictions failed to match market performance between certain periods of time, thus the adequacy of ARIMA (1.1.1) model to forecast the NSE index was questioned. The researcher concluded that the deviations found between forecast and actual values indicate that the global economies crisis destroyed the correlation relationship existing between the NSE index and its past.

III. THE SUPPORT VECTOR MACHINE MODEL

The Support Vector Machines are techniques, which are often categorized under the Machine Learning Methods. Machine Learning includes a number of advanced statistical methods for handling regression and classification tasks with

multiple dependent and independent variables. These methods include Support Vector Machines (SVM) for regression and classification, Naive Bayes for classification, and k-Nearest Neighbours (KNN) for regression and classification.

In a nutshell, the whole idea behind Support Vector Machines (SVMs) is to make use of a non-linear mapping function Φ that transforms data in input space to data in feature space in such a way as to make a problem linearly separable. The SVM then automatically discovers the optimal separating hyper-plane, which when mapped back into input space via Φ^{-1} , turns into a complex decision surface.

Basically the Support Vector Machines (SVM) performs regression and classification tasks by constructing nonlinear decision boundaries. Because of the nature of the feature space in which these boundaries are found, Support Vector Machines can exhibit a large degree of flexibility in handling classification and regression tasks of varied complexities. There are several types of Support Vector models including linear, polynomial, RBF, and sigmoid.

3.1 SVM Linear Classifiers

A key concept that is considered here is the dot product (inner product or scalar product) between two vectors, defined as $W^T X = \sum_i w_i x_i$. A linear classifier is based on a linear discriminant function of the form

$$f(X) = W^T X + b \tag{3.1}$$

where W is the weight vector and b is called the bias.

Consider the case $b = 0$. The set of points X , such that $W^T X = 0$ are all points that are perpendicular to W and passes through the origin, which is a line in 2-dimensions and a plane in 3-dimensions or more generally, a hyper-plane. Note that the bias b translates the hyper-plane away from the origin. The hyper-plane

$$X: f(X) = W^T X + b = 0 \tag{3.2}$$

Divides the space into two: The sign of the discriminant function (X), denotes which side of the hyper-plane a point is located. The boundary between regions classified as positive ($f(X) > 0$) and negative ($f(X) < 0$), is called the decision boundary of the classifier. In this case the decision boundary defined by the hyper-plane is said to be linear. A classifier with a linear decision boundary is called a linear classifier. Conversely, when the decision boundary of a classifier depends on the data in a non-linear form, then we say the classifier is non-linear [13]. The naive method of creating a Non-linear classifier out of a linear classifier is to map our data from the input space \mathcal{X} to a feature space \mathcal{F} using a non-linear function $\Phi: \mathcal{X} \rightarrow \mathcal{F}$. In the space \mathcal{F} the discriminant function is

$$f(X) = W^T \Phi(X) + b \tag{3.2}$$

3.2 Regression SVM

In a regression SVM, you have to estimate the functional dependence of the dependent variable y on a set of independent variables x . It assumes, like other regression problems, that the relationship between the independent and dependent variables

is given by a deterministic function f plus the addition of some additive noise:

$$y = f(x) + noise \tag{3.3}$$

The task is then to find a functional form for f that can correctly predict new cases that the SVM has not been presented with before. This can be achieved by training the SVM model on a sample set, i.e., training set, a process that involves the sequential optimization of an error function. Depending on the definition of this error function, two types of SVM models can be recognized; these are (i) Regression SVM type1, also known as the epsilon-SVM regression and (ii) Regression SVM type 2, also known as nu-SVM regression.

> Regression SVM Type 1

For this type of SVM the error function is:

$$\frac{1}{2} W^T W + C \sum_{i=1}^N \xi_i + C \sum_{i=1}^N \xi^*_i \tag{3.4}$$

which we minimize subject to:

$$\begin{aligned} W^T \phi(x_i) + b - y_i &\leq \varepsilon + \xi^*_i \\ y_i - W^T \phi(x_i) - b &\leq \varepsilon + \xi_i \\ \xi_i, \xi^*_i &\geq 0, i = 1, \dots, N \end{aligned}$$

> Regression SVM Type 2

For this SVM model, the error function is given by:

$$\frac{1}{2} W^T W - C \left(v\varepsilon + \frac{1}{N} \sum_{i=1}^N (\xi_i + \xi^*_i) \right) \tag{3.5}$$

which we minimize subject to:

$$\begin{aligned} (W^T \phi(x_i) + b) - y_i &\leq \varepsilon + \xi_i \\ y_i - (W^T \phi(x_i) + b_i) &\leq \varepsilon + \xi^*_i \\ \xi_i, \xi^*_i &\geq 0, i = 1, \dots, N, \varepsilon \geq 0 \end{aligned}$$

There are number of kernels that can be used in Support Vector Machines models. These include Linear, Polynomial, Gaussian and Sigmoid:

$$\phi = \left\{ \begin{array}{ll} x_i * x_i & \text{Linear} \\ (\gamma x_i x_j + coefficient)^{degree} & \text{Polynomial} \\ \exp(-\gamma |x_i - x_j|^2) & \text{Gaussian} \\ \tanh(\gamma x_i x_j + coefficient) & \text{Sigmoid} \end{array} \right\}$$

The Gaussian is by far the most popular choice of kernel types used in Support Vector Machines. This is mainly because of their localized and finite responses across the entire range of the real x-axis.

3.3 Methodology

The objectives of this work is to illustrate that Support Vector Machines can effectively be used to predict the Nigerian Stock Exchange Market (NSE) index values using previous day's index values, and previous day's NGN/USD exchange rate.

In this study the following input variables would be considered to ultimately affect the stock exchange market index value.

- ♦ NSE All Share index (according to closing price) (NSE_ASI)
- ♦ NGN/USD exchange rate (NGN_USD)
- ♦ NSE Market Capitalization (according to closing price) (NSE_MCAP)

- Volume (VOL_CLOSING)
- Value (VAL_CLOSING)

Considering the input variables, the following system model was considered for the prediction stock exchange market index value:

$$VAL_{cls} = f(NSE_{ASX}, NGN_s, NSE_{Mcap}, VOL_{cls}) \quad (3.5)$$

Experimental data were downloaded from the websites of three prominent/registered Nigerian stock brokers these are Cowry, CashCraft and BGL. The data collected is for a period of 570 days starting from January 4, 2010 to April 30, 2012 excluding weekends and public holidays.

While pre-processing our data the mean of each the five attributes ($NSE_{ASX}, NGN_s, NSE_{Mcap}, VOL_{cls}$ and VAL_{cls}) were used to further transformed data into nominal values of “small” and “large” for the first four attributes, while nominal values of “low” and “high” were used for the fifth attribute. We implemented the Support Vector Machines-SMO algorithm using the WEKA software and results were obtained as presented in section 4 below.

3.3.1 The Sequential minimal optimization (SMO) algorithm

belongs to the SVMs learning algorithm that uses an analytic quadratic programming step (QP-Step) instead of numerical quadratic programming (QP).

Sequential minimal optimization (SMO) is an algorithm for efficiently solving the optimization problem which arises during the training of support vector machines. It was invented by John Platt in 1998 at Microsoft Research. SMO is widely used for training support vector machines and is implemented by the popular LIBSVM tool. The publication of the SMO algorithm in 1998 has generated a lot of excitement in the SVM community, as previously available methods for SVM training were much more complex and required expensive third-party QP solvers [14].

Optimization problem: Consider a binary classification problem with a dataset $(x_1, y_1), \dots, (x_n, y_n)$, where x_i is an input vector and $y_i \in \{-1, +1\}$ is a binary label corresponding to it. A soft-margin support vector machine is trained by solving a quadratic programming problem, which is expressed in the dual form as follows:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j K(x_i, x_j) \alpha_i \alpha_j, \quad (3.6)$$

subject to:

$$0 \leq \alpha_i \leq C, \text{ for } i = 1, 2, \dots, n,$$

$$\sum_{i=1}^n y_i \alpha_i = 0$$

where C is an SVM hyper-parameter and $K(x_i, x_j)$ is the kernel function, both supplied by the user; and the variables α_i are Lagrange multipliers.

3.3.2 SMO Algorithm

SMO is an iterative algorithm for solving the optimization problem described above. SMO breaks this problem into a series of smallest possible sub-problems, which are then solved analytically. Because of the linear equality constraint involving

the Lagrange multipliers α_i , the smallest possible problem involves two such multipliers. Then, for any two multipliers α_1 and α_2 , the constraints are reduced to:

$$0 \leq \alpha_1, \alpha_2 \leq C,$$

$$y_1 \alpha_1 + y_2 \alpha_2 = k$$

and this reduced problem can be solved analytically.

The algorithm proceeds as follows:

1. Find a Lagrange multiplier α_1 that violates the Karush–Kuhn–Tucker (KKT) conditions for the optimization problem.
2. Pick a second multiplier α_2 and optimize the pair (α_1, α_2) .
3. Repeat steps 1 and 2 until convergence.

When all the Lagrange multipliers satisfy the KKT conditions (within a user-defined tolerance), the problem has been solved. Although this algorithm is guaranteed to converge, heuristics are used to choose the pair of multipliers so as to accelerate the rate of convergence.

3.3.3 Analysis of the SMO model

The performance of the SMO model is evaluated according to their precision (P) and recall (R), leading to a trade-off curve similar to ROC (receiver operating characteristic) curve. In order to make connections between the five attributes, we focus on two-class classification problems, calling one class “Bad” and designating the other class as “Good” [15]. Results of a two class instance (daily stock) classification problem may be summarized by a confusion matrix:

$$\begin{matrix} & a & b & \leftarrow \text{classified as} \\ \begin{pmatrix} TP & FP \\ FN & TN \end{pmatrix} & \begin{matrix} |a=Bad \\ |b=Good \end{matrix} \end{matrix} \quad (3.7)$$

Here TP(true positives) are the instances correctly classified as belonging to the “Bad” class, FP(false positives) are the instances incorrectly classified as belonging to the “Good” class, FN(false negatives) are the instances belonging to the class but not classified as such and finally TN(true negatives) are instances correctly classified as not belonging to the class.

The precision (P) and recall (R) are used to evaluate the learning tasks and are calculated from the confusion matrix in (3.7) as

$$P = \frac{|TP|}{|TP|+|FP|} \text{ and } R = \frac{|TP|}{|TP|+|FN|} \quad (3.8)$$

The objective of ROC analysis are to return all the “Bad” and “Good” instances, but it does so by maximizing recall (called the true positive rate) and minimizing the false rate:

$$R = \frac{|FP|}{|FP|+|TN|} \quad (3.9)$$

R shows how many of all possible instances belonging the “Good” class that have been classified as “Bad”; R is equivalent to the fallout measure in the SOM algorithm. Furthermore, in order to overcome the problem for identifying which objective is outperforming another, the F-Measure comes handy and is defined as

$$F_{\beta} = \frac{PR}{(1-\beta)P+\beta R} \quad (3.10)$$

F-measure (3.10) is a composite measure that combines the precision and recall objectives into one objective. For example

if $\beta=0.5$ the F-measure is the harmonic mean of precision and recall.

IV. EXPERIMENT AND RESULTS

The results for the implementation of the Support Vector Machine-SMO Algorithm, on WEKA, are presented below:

TABLE1: MACHINE LINEAR SHOWING ATTRIBUTE WEIGHTS

Attribute	Weight
NSE-ASI	0.76500
NGN_USD	0.10440
NSE_MCAP	-0.43750
VOL_CLS	2.20079
VAL_CLS	10.7889

TABLE2: RESULTS OF STRATIFIED CROSS VALIDATION FOR THE 570 INSTANCES

Correctly Classified Instances	509 (89.2982%)
Incorrectly Classified Instances	61 (10.7018%)
Number of Kernel Evaluations	14,580 (70.068%)
Kappa statistic	0.7640
Mean absolute error	0.1070
Root mean squared error	0.3271
Relative absolute error	22.4981%
Root relative squared error	67.0850%

TABLE3: DETAILED ACCURACY BY CLASS

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC
Bad	0.994	0.266	0.854	0.994	0.919	0.864
Good	0.734	0.006	0.988	0.734	0.842	0.864

==== Confusion Matrix ====
 a b <-- classified as
 $\begin{pmatrix} 346 & 2 \\ 59 & 163 \end{pmatrix}$ | a=Bad
 | b=Good

V. DISCUSSION

The historical data of the NSE is a noisy one, thus before subjecting our data to the Support Vector Machine implementation and analysis, we employed robust statistical techniques to alleviate the problem of noise sensitivity [16]; this process had greatly enhanced the quality of our data.

The Support Vector Machine algorithm learned all the 570 instances as categorized under the five attributes (Table 1), with un-biased estimates of the all instances but one (NSE_MCAP) as seen from the weight sum column. The results of the stratified cross validation of the instances (Table 2) came up with medium value of relative absolute error

(22.4981%) and an above average root relative squared error (67.0850%), however other indices most especially Kappa Statistics, mean absolute error and root mean squared error all depicts a positive results.; also (Table 3) the precision, recall, F-measure and ROC results seems to have a positive influence on the learning process [17]. As seen from the confusion matrix the SMO algorithm had almost near perfect learning/prediction capabilities; with TP=346, FP=2, FN=59 and TN=163.

The SMO randomly split into a 70% training set and a 30% test set, this portioning was performed 10 times to give 10 different folds to compare when evaluating the results; since the training and test sets are constant for all classifiers being compared, the precision/recall curves are as stable as ROC (0.864) curves.

VI. CONCLUSION/FURTHER WORKS

Based on our findings, it is clear that through the processes of data mining (semi-transformation of the data before analyzing it) the Support Vector Machines-Sequential Minimal Optimization algorithm had effectively predicted the NSE.

Further works will involve research work on possibilities of using other Support Vector Machines algorithms to predict the NSE.

REFERENCES

- Marwala L.R., 'Forecasting the Stock Market Index Using Artificial Intelligence Techniques'2007
- Hellstrom T. And Holmstrom K., Predicting the Stock Market, Technical Report Series, Malardalen Univeristy Sweden, 1998.
- S.M. Kendall and K.ord, Time Series, 3rd ed. (Oxford University Press, New York, 1990)
- A.W. Lo and A.C. Mackinlay, Stock market prices do not follow random walks: Evidence from a simple specification test, Review of Financial Studies 1 (1988) 41-66
- Shah H.V., 2007, Machine Learning Techniques for Stock Prediction, Foundations of Machine Learning | Spring 2007
- Kinlay J andRico D., Can Machine Learning Techniques be used to predict Market Directions?, Journal of Finance, 2008
- Fletcher T., Redpath F., and D'Alessadro J., Machine Learning in FX Carry Basket, 2007.
- Akinwale A.T, Arongundade O.T and Adekoya A.F. (2009), Translated Nigeria Stock Market Prices using Artificial Neural Network for Effective Prediction, Journal of Theoretical and Applied Information Technology, 36-43.
- Olabode O., Adeyemo A.B, and Coker C.O., Artificial Neural Network for Stock Market Forecasting, www.sciencenigeria.org/index2.php?option=com_docman...doc.
- Bello M.Y. and Chiroma H. (2011), Utilizing Artificial Neural Network for Prediction in the Nigerian Stock Market Price Index, Computer Science and Telecommunications 2011/No.1(30)
- Agwuegbo S.O.N., Adewole A.P., and Maduegbuna A.N. (2010), A Random Walk Model for Stock Market Prices, Journal of Mathematics and Statistics 6(3): 342-346
- Emenike K. O. (2010), Forecasting Nigerian Stock Exchange Returns: Evidence from Autoregressive Integrated Moving Average (ARIMA) Model, Social Science Research Network.
- Boser B.E.,Guyon I.M, and Vapnik V.N. (1992), A Training algorithm for optimal margin classifiers. 5th Annual ACM Workshop on COLT, pages 144-152, Pittsburg, PA, ACM Press.
- http://en.wikipedia.org/wiki/Sequential_minimal_optimization
- Fisher M.J., Fieldsend J.E., and Everson R.M. (2004), Precision and Recall Optimisation for Information Access Tasks, 1st International Workshop, ROCAI-2004, pages 45-54, Valencia, Spain

- [16] Chu F., Wang Y., and Zaniolo C., (2004), Mining Noisy Data Streams via Discriminative Model, Discover Science-Lecture Notes in Computer Science, Vol.3245,2004, pp 47-59
- [17] Mendes R.M.S. and Godinho M.A.B., Knowledge of Results Precision and Learning: A Review, Sousa 2003, pp 23.