Improving the Criteria of the Investment on Stock Market Using Data Mining Techniques: The Case of S&P500 Index

Carlos Montenegro and Marco Molina

Abstract—**The stock market data, as S&P500 Index, is massive, complex, non-linear and noised. Thus, the investment criteria using this information have been a challenge. This study proposes the following short-term step by step strategy: to combine two information sources that the investors can analyse to make a decision. First, the index data constitutes the input for a Deep Learning Neural Network training, for representing and forecasting next day stock value. Second, this research identifies the most representative enterprises, included on Index, which represent the Index behavioural tendency, using Feature Selection Analysis. Finally, the outputs are complemented and corroborated; the process shows promising results to improve the investor's decision. Thus, the academics can revise a new experience in data analysis; for the practitioners, the research contributes to an approach for supporting investment decisions in the stock market.**

Index Terms—**Deep learning, feature selection, S&P500 index, stock market.**

I. Introduction

Predicting stock indices has been regarded as one of the most challenging tasks in econometrics. Measuring market risk requires quantitative techniques to analyse individual financial instruments and a portfolio of assets. This quantitative measure or model captures trends and behaviours in data which are then used to deduce future values [1].

The predictability of the stock market has been long a research topic. According to Fang [2], the overall stock indices are generally easier to work with than individual stocks. Zheng and Chen [3] and Olden [4] suggest that although the researchers cannot agree on whether the stock markets are predictable or not, studying whether one can predict them is an exciting theme.

Stock prediction can be made by using a statistical approach, which treats the stock data as a time series and uses no other information. Examples include Exponential Smoothing Models (ESM), ARIMA models, ARCH and GARCH models, among others [1]. The financial models are based on statistical proprieties and assumptions in the underlying data. In some instances, unrealistic assumptions are made to simplify the problem or to allow the mathematical derivation of the model. Given the complex behaviour of financial markets, these models can potentially misrepresent or fail to represent critical features of underlying data [1].

On the other hand, feature-based machine learning approaches take advantage of economic data, as well as historical stock data. Examples include Support Vector Machines, Genetic Algorithms, and Artificial Neural Network (ANN). ANN has been one of the most successful applications [5]-[8].

There has been a great interest in deep network structures more recently. Deep Neural Networks (DNN) has various successful reports in machine learning [2]. There are desirable features of neural networks, which make them a suitable tool for market risk modelling. According to Mostafa et al. [1] and Qian [5], the results achieved show the superiority of neural networks over statistical models. Also, they are naturally suitable to model nonlinearities in data.

The described scenario facilitates exploring alternatives to improve the criteria to stock market investments, using the most promise techniques of machine learning, as is posed in this study, and considering the case of S&P500 Index data.

II. Background and Related Work

A. S&P 500 Index

The Standard & Poor's 500 is an American stock market index based on the market capitalizations of 486 large companies having common stock listed on the New York Stock Exchange (NYSE) or Nasdaq Stock Market (NASDAQ). Each enterprise name is represented using an acronym; for example, INTC for Intel.

S&P Dow Jones Indices determine the S&P 500 index components and their weightings; it is considered one of the best representations of the U.S. stock market.

Table I shows an extract of the daily data available for the S&P500 Index for each Enterprise. Open, High, Low and Close refer to stock value; Volume refers to the number of shares of the stock market. The mentioned data can be obtained from URL: http://www.financeyahoo.com.

B. Stock Forecasting Using Neural Networks for Deep Learning

Many artificial intelligence methods have been employed to predict stock market prices [6], [7]. ANN remains as a

Manuscript received May 23, 2019; revised February 5, 2020. This work was supported in part by Escuela Politécnica Nacional (EPN), Ecuador.

Carlos Montenegro and Marco Molina are with Department of Computer Sciences and Informatics of Escuela Politécnica Nacional (EPN), Ecuador (e-mail: carlos.montenegro@epn.edu.ec).

doi: 10.18178/ijmlc.2020.10.2.936

popular choice for this task and is widely studied and have been shown to exhibit excellent performance [8], and recent literature suggests that researchers are attempting to use deep learning for stock prediction using DNN [8], [9].

DNN is a machine-learning paradigm for modelling complex nonlinear mappings between input and output, in which the internal parameters are updated iteratively to make the given inputs fit with target outputs [10], [11].

Deep Learning approaches consist in adding multiple repeatable layers to a neural network. Discussing the matter, deepest learning strategies rely at least on the following five types of architectures [12]-[15]: (i) Convolutional Neural Networks, (ii) Recurrent and Recursive Neural Networks; (iii) Multi-Layer Perceptron; (iv) Back Propagation Neural Network (BPNN); and, (v) Standard DNNs, which are a combination of layers of different types without any particular order.

TABLE I: S&P500 INDEX DATA (EXTRACT)

Date	Open	High	Low	Close	Volume
08/05/2018	2670,26	2676,34	2655,20	2671,92	3717570000
09/05/2018	2678,12	2701,27	2674,14	2697,79	3909500000
10/05/2018	2705,02	2726,11	2704,54	2723,07	3333050000
11/05/2018	2722,70	2732,86	2717,45	2727,72	2862700000
14/05/2018	2738,47	2742,10	2725,47	2730,13	2972660000
15/05/2018	2718,59	2718,59	2701,91	2711,45	3290680000
16/05/2018	2712,62	2727,76	2712,17	2722,46	3202670000
17/05/2018	2719,71	2731,96	2711,36	2720,13	3475400000
18/05/2018	2717,35	2719,50	2709,18	2712,97	3368690000

Standard DNN provides a universal framework for modelling complex and high-dimensional data. An especially attractive feature of DNN approach is the inherent capability of covering all stages of data-driven modelling (features selection, data transformation, and classification/regression) within a single framework, i.e., ideally, the practitioner can start with raw data in the domain of interest and get ready-to-use solution; besides, this deep feature hierarchy enables DNNs to achieve good performance in many tasks [16], [17].

There is little research regard to the recommended use of Deep Learning Architectures. Nevertheless, the classical architecture for Deep Learning (Fig. 1) has the layers in each fully connected stack with fewer nodes than the preceding [9], [18]-[20]. The output from the final stack produces a prediction of the target variable.

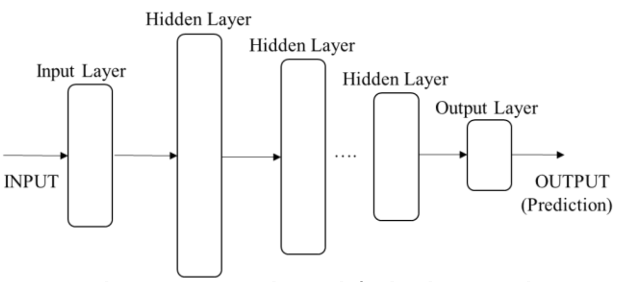

Fig. 1. Back propagation neural network for deep learning architecture.

C. Feature Selection Techniques

A feature is an individual measurable property of a process being observed, represented by a variable. Feature Selection (variable elimination) helps in understanding data, reducing computation requirement, reducing the effect of dimensionality and improving the predictor performance. Therefore, the focus of feature selection is to select a subset of input variables that can describe data, reducing effects from noise, or irrelevant variables and still provide better predictive results [21]-[23].

Regarding availability of label information, feature selection technique can be roughly classified into three families: supervised methods, semi-supervised methods, and unsupervised methods [24]-[26]. The availability of label information allows supervised feature selection algorithms to efficiently select discriminative and relevant features, to distinguish samples from different classes.

Based on different strategies of searching, feature selection can also be classified into three methods, i.e., filter, wrapper and embedded methods [22]-[24]. Wrapper methods involve a learning algorithm as a black box and consist of using its prediction performance to assess the relative usefulness of subsets of variables. In other words, the feature selection algorithm uses a learning method (Classifier) as a subroutine with the computational burden that comes from calling a learning algorithm to evaluate each subset of features [27], [28] (See Fig. 2).

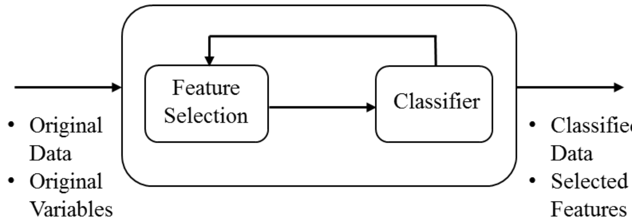

Fig. 2. Wrapper method configuration [26].

Options to Feature selection are Wrapper Subset Evaluator, Correlation-based Feature Subset Selection (CFS), Principal Components Analysis, among others [29], [30], [31]. On the other hand, options for Search Methods (Classifiers) are Greedy Stepwise, Evolutionary Search or Best First [22], [32], [33].

In a preliminary test round, using the Merit, or Pearson's correlation coefficient [34], [35] as a performance measure, the best results were shown by the CFS as a classifier and Greedy Stepwise forward chaining for feature selection.

CFS is a simple multivariate filter algorithm that ranks feature subsets according to a correlation-based heuristic evaluation function. The measure used is:

$$Q_{zc} = (m\, Q_{zi}) \,/\, (\, m + m(m-1)\, Q_{ii})^{1/2}$$

where Q_{zc} is the correlation between the individual features and the output class, m is the number of features, Q_{zi} is the measure of correlation between each feature and the output variable, and Q_{ii} is the average intercorrelation among features. So, the measure used assigns high values to the subsets that are highly correlated with the output while being weakly correlated with each other. Irrelevant features should be ignored because they will have a low correlation with the class. Redundant features should be screened out as they will be highly correlated with one or more of the remaining features. The acceptance of a feature will depend on the extent to which it predicts classes in areas of the instance space not already predicted by other features [28].

On the other hand, Greedy Stepwise performs a greedy forward or backward search through the space of attribute subsets. The process stops when the addition/deletion of any remaining attributes results in an evaluation decrease [33]. The process solves the following model:

$$\max_{S \subset P} R^2\,(G, S)$$
$$\text{s. t. } |S| = k$$

where:

k number of data sources to choose

P data sources

G target data

α_i are the regression coefficients from fitting G using de P_i's

$$R^2\,(G, S) = \frac{Var(G) - Var\,(G - \sum_{i \in S} \alpha i\ \mathrm{Pi})}{Var\,(G)}$$

Var corresponds to the Variance.

III. Methods

The stock market data, as S&P500, are massive, complex, non-linear and noised. Thus, the investment criteria have been a challenge. This study proposes the following strategy: generate combined information that the investors can analyse. First, index value forecasting can be made using a supervised learning approach. Additionally, the most influent enterprises with volume values that represent the Index behaviour, are identified. So, the methods to be used are the following:

1) A forecasting model based on deep learning neural networks is defined and trained.

The forecasting process requires a model to represent the phenomenon. So, the model must extrapolate using the new input data, due to the explicit or implicit presence of time as a variable.

The ANNs can exhibit abnormal behaviour in extrapolation cases, as some classical studies suggest [36], [37] as well as more recently investigations [38], [39]. In this study, an initial explorative work shows behavioural anomalies in extrapolation forecasting. For solving the problem, Barnard and Wessels [36] suggest simulating the ANN with values around the trained data.

The forecasting model must represent data adequately, produce quantitative future index values and suggest the qualitative characteristic of the forecasted value: UP for a value increase, DOWN for a value decrease, and STABLE for an invariant value.

According to the above considerations, using data of the previous step, the next value in time is calculated. The sliding window technique is showed in Fig. 3.

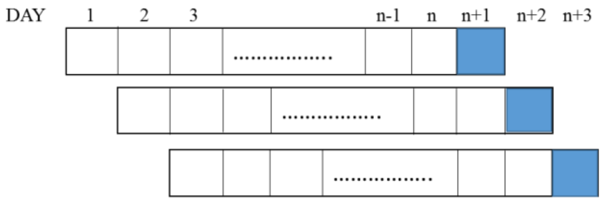

Fig. 3. Sliding window technique.

Data of n days is used to forecast the value of the day (*n*+1). At the next step, the real data of the day (*n*+1) is added, and the data on the first day is deleted. Then, the value of the day (*n*+2) is forecasted and so on.

Besides, as is showed later, new variables must be defined to obtain an implicit representation time and transform the extrapolation into an interpolation problem.

2) Feature selection: A data mining technique, is used to diminish the complexity and noise of data. So, the investors can identify the most influents enterprises in monitoring the index behaviour. In this case, the forecasting model must represent the data adequately and produce the future quantitative values of variable Open and suggest the qualitative characteristic of it (UP, DOWN, STABLE). The enterprise's data are used to corroborate and complement the forecasting model results.

Thus, the investment decision has two complementary sources. The proposed approach is shown in Fig. 4.

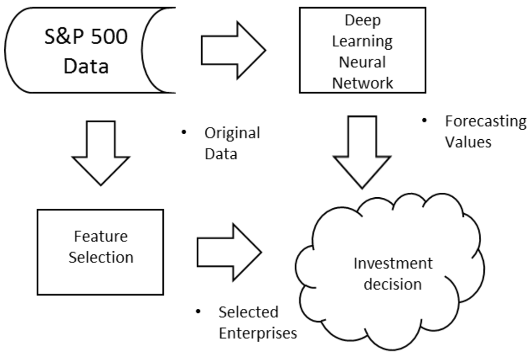

Fig. 4. Investment decision process.

The decision reliability scenarios can be revisited in Table II. The green, yellow and red options represent great, medium and low reliability respectively.

TABLE II: Decision Scenarios

		The behavior of representative Enterprises		
		UP	DOWN	STABLE
Deep Learning forecasting	UP			
	DOWN			
	STABLE			

IV. Results

A. Data

The S&P 500 Index data correspond to market activity days, from June 7, 2013, to June 6, 2018. For initial data processing, enterprises constitute 486 variables and 1259 examples of Open values for each one. The missing values were replaced with mean values.

Fig. 5 shows an extract of the most interesting descriptive statistical results: nonlinearly distributed variables (In the diagonal of the matrix), and the proportionality relationships between them. The relations between variables have high dispersion producing clouds images, as shown in the figure.

Alike, Table III shows the Correlation Matrix of data for the first eight enterprises (Vi). There is a high correlation between some variables, which suggest the data duplication presence.

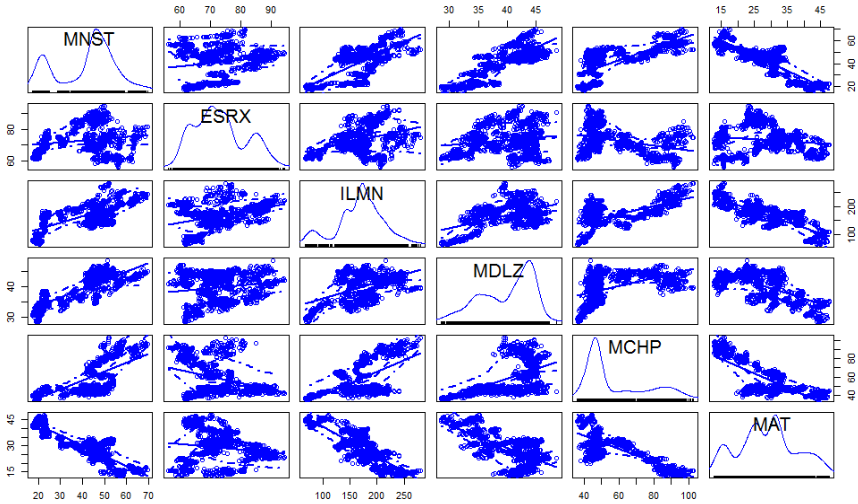

Fig. 5. Scatter Plot Matrix for some variables.

TABLE III: CORRELATION MATRIX FOR SOME VARIABLES

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8
V_1	1.0	0.1	0.5	0.2	0.2	0.6	-0.4	0.0
V_2	0.1	1.0	0.7	0.3	0.3	0.6	-0.7	0.3
V_3	0.5	0.7	1.0	0.3	0.5	0.9	-0.8	0.5
V_4	0.2	0.3	0.3	1.0	-0.1	0.5	-0.3	-0.4
V_5	0.2	0.3	0.5	-0.1	1.0	0.3	-0.2	0.6
V_6	0.6	0.6	0.9	0.5	0.3	1.0	-0.8	0.1
V_7	-0.4	-0.7	-0.8	-0.3	-0.2	-0.8	1.0	-0.2
V_8	0.0	0.3	0.5	-0.4	0.6	0.1	-0.2	1.0

Besides, the Covariance matrix (Table IV) shows the different proportionality between variables, which corroborate the scatter plot matrix results.

TABLE IV: COVARIANCE MATRIX FOR SOME VARIABLES

	V1	V2	V3	V4	V5	V6	V7	V8
V1	10.6	1.7	12.2	1.4	3.5	18.7	-7.0	0.4
V2	1.7	38.9	30.8	3.3	7.3	37.8	-21.5	22.9
V3	12.2	30.8	47.8	3.6	14.6	56.7	-25.4	38.0
V4	1.4	3.3	3.6	4.0	-0.8	8.8	-3.1	-9.2
V5	3.5	7.3	14.6	-0.8	19.6	13.5	-4.9	28.5
V6	18.7	37.8	56.7	8.8	13.5	88.2	-38.4	10.8
V7	-7.0	-21.5	-25.4	-3.1	-4.9	-38.4	23.4	-8.9
V8	0.4	22.9	38.0	-9.2	28.5	10.8	-8.9	126.6

The significance of the foregoing facts can be summarized as: i) The frequency of high values on linear correlations suggest that some variables can introduce duplications in data, a way of noise, as a recognized characteristic of stock market behavior; ii) The distribution of variables are nonlinear; therefore, possible models for treating data must support nonlinear data.

B. Stock Forecasting Using DNN

For constructing, training, and testing the model, MATLAB software [40] is used. The performance of DNN is evaluated using the correlation value, *R*, of the modelled output. *R* is determined as follows:

$$R^2 = 1 - \frac{\sum(y_{exp} - y_{pred})^2}{\sum(y_{exp} - \bar{y})^2}$$

where y_{exp} is the experimental (real) value, y_{pred} is the predicted value, and $\bar{y}$ is the mean value.

The following eleven variables are derived and used in the forecasting process. These variables definition convert the initial extrapolation forecasting into an interpolation forecasting. Note that the data of year is eliminated. As a notation, S_i and S_{i+1} are successive days.

Input Variables:

- Month: it refers to the month to which a given record belongs.
- MonthDay: day of the month to which a given record belongs.
- WeekDay: refers to the day of the week corresponding to a given stock record.
- LowDiff: For two consecutive slots S_1 and S_2. If L_1 and L_2 refer to the Low values for S_1 and S_2 respectively, then LowDiff for S_2 is computed as (L_2 - L_1).
- HighDiff: the difference between the High values of two successive slots. The computation is identical to LowDiff.
- CloseDiff: If two successive slots S_1 and S_2 have close values C_1 and C_2 respectively, then CloseDiff for S_2 is calculated as (C_2 - C_1).
- VolDiff: For two consecutive slots S_1 and S_2, if the mean values of Volume for both the slots are V_1 and V_2 respectively, the VolDiff for S_2 is (V_2 - V_1).
- RangeDiff: For two consecutive slots S_1 and S_2, suppose the High and Low values are H_1, H_2, L_1 and L_2 respectively. Hence, the Range value for S_1 is $R_1 = (H_1 - L_1)$ and for S_2 is $R_2 = (H_2 - L_2)$. The RangeDiff for the slot S_2 is (R_2 - R_1).
- OpenClose: Suppose two consecutive slots: S_1 and S_2. Let the Open price of S_2 be X_2, and for the Close price of S_1 be X_1. The OpenClose for the slot S_2 is (X_2 - X_1).

Output Variable:

Two possible Output variables are defined:

- OpenPerc: Suppose two consecutive slots: S_1 and S_2. Let the Open price of the stock for the record of S_1 be X_1, and that for S_2 be X_2, the OpenPerc for the slot S_2 is computed as $(X_2 - X_1)/X_1*100$.
- AveragePerc: Let Average = (Open + Close)/2. Suppose two consecutive slots: S_1 and S_2. Let the Average price of the stock for the record of S_1 be X_1, and that for S_2 be X_2, AveragePerc for slot S_2 is computed as $(X_2 - X_1)/X_1 \times 100$.

OpenPerc gives early daily information about stock price and can be used for an "instantaneous" early investment decision. AveragePerc is a good option for investment, that takes into account a daily period. One, or both of them, can be utilized depending on the investment strategy.

According to the current theory and practice, Fig. 6 shows the used architecture: four feedforward backpropagation fully connected layers, with sixty-six neurons.

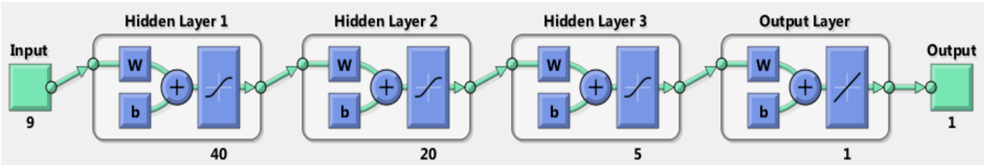

Fig. 6. DNN architecture for this case study.

TABLE V: DATA FOR OPENPERC VARIABLE FORECASTING

Case	Month	Month Day	Week Day	Low Diff	High Diff	Close Diff	Vol Diff	Range Diff	Open Close	Open Perc
1	6	10	2	13.99	4.29	-0.57	-393260000	-9.70	1.29	1.19
2	6	11	3	-16.34	-8.56	-16.68	456980000	7.78	-4.17	-0.37
3	6	12	4	-12.00	-2.42	-13.61	-233160000	9.58	3.81	-0.53
….	….	….	….	….	….	….	….	….	…...	….
1258	6	6	4	8.95	19.78	23.55	133850000	10.83	4.45	0.17
1259	6	7	5	8.95	19.78	23.55	133850000	10.83	4.45	***0.24***

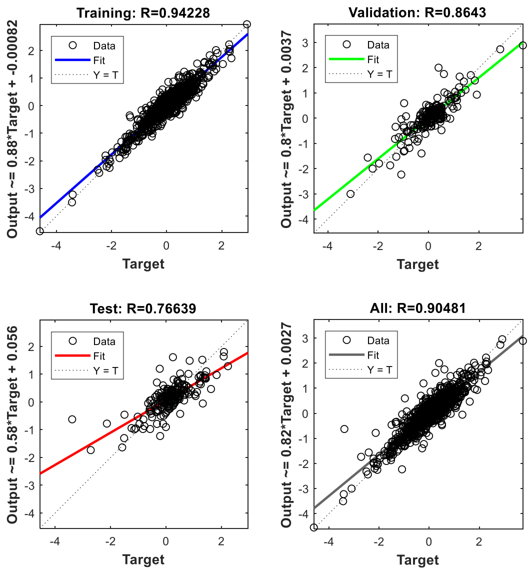

Fig. 7. Data adjustment.

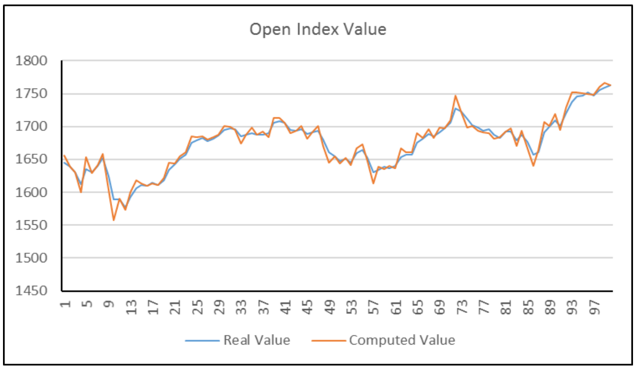

Fig. 8. Real and computed open values.

The first hidden layer attempt to produce weights from the activations, to be used for regression. The second and third hidden layer act as pooling layers that perform down sampling along the spatial dimensionality of the given input, further reducing the number of parameters of the phenomena. The linear transfer function on the output layer acts as a regression layer.

Adjust to the training, validation, testing, and complete data are presented in Fig. 7, including R values. The training process uses 70% of the data, the validation uses 15% of the data, and 15 % of the data is taken into account on test calculations. The results show a great capability of DNN to represent the phenomena, using the OpenPerc Output variable.

Fig. 8 shows a graphical representation of experimental data, using the values generated by the DNN for variable Open, against real Open values.

Table V shows the data used to predict the variable OpenPerc for the next slot, based on the historical behaviour of stock prices. In other words, if the current time slot is S_1, the technique will attempt to predict OpenPerc for the next slot S_2.

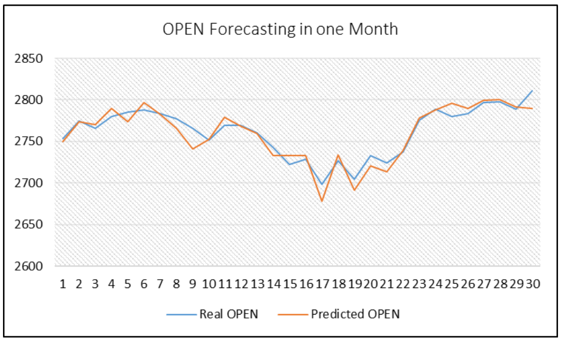

Fig. 9. Adjust of the Index open forecasted values.

A positive predicted OpenPerc, indicates that there is an expected rise in stock price of S_2 (UP option), while a negative OpenPerc indicates a fall in stock price for the next slot (DOWN option). An equal value, for both slots, constitutes a STABLE option.

At the last line of Table V (Case 1259) only the Weekday and the Month day values are modified (Time variables). Other values are the same as the previous day. Then, the value of OpenPerc is obtained simulating the trained DNN. In the showed case the predicted value of OpenPerc is 0.24, with a "UP" behaviour.

Thirty steps (from June 7, 2018, to July 19, 2018) are forecasted Using the previous procedure, as a proof of the adjustment quality. Fig. 9 shows a graph of forecasted and real values of the Open variable, with a correlation of R= 0.9322, that is acceptable.

TABLE VI: ESTIMATED AND REAL BEHAVIOUR OF STOCK OPEN VALUES TO THE NEXT DAY

	06/06/2018	06/07/2018			
Enterp	Open real value	Open real value	Open forecasting value	Est ímate behaviour	Real behaviour
MAR	137,97	141,66	136,31	DOWN	UP
INTC	56,53	56,92	55,52	DOWN	UP
GD	201,83	201,98	201,86	UP	UP
IPG	22,84	23,00	22,81	DOWN	UP
KEY	19,99	20.55	20,38	UP	UP
KO	43,06	43,22	42,77	DOWN	UP
APD	165,16	167,51	166,42	UP	UP
APH	90,00	90,5	92,99	UP	UP
BK	55,8	57,72	56,19	UP	UP
BRK-A	287680	291900	287678,55	UP	UP
UTF	22,68	22,7	22,76	UP	UP
CHK	4,33	4,49	4,36	UP	UP
FDX	253,93	257,3	256,31	UP	UP
F	11,87	11,97	11,93	UP	UP
ECL	144,98	146,53	145,12	UP	UP
PKI	76,73	78,41	77,39	UP	UP
ORCL	47,42	47,93	47,62	UP	UP
SPGI	203,85	206,03	205,78	UP	UP
SHW	390,51	395,37	389,59	UP	DOWN
SCHW	56,63	58,11	56,55	UP	DOWN
RSG	67,92	68,25	67,59	UP	DOWN
MMC	81,3	81,85	82,25	UP	UP
WY	38,16	38,06	38,13	DOWN	DOWN
USB	51,39	51,94	51,95	UP	UP
TXT	68,33	68,88	69,11	UP	UP
ABT	63,11	63,56	61,58	DOWN	UP
PBCT	18,93	19,14	19,03	UP	UP
CSCO	43,76	44,24	44,20	UP	UP
CMCSA	24,8	24,98	24,83	UP	UP
ADP	134,43	136,32	134,12	DOWN	UP

C. Selecting Features of S&P500 Index

The daily values of variable Open from 486 enterprises are processed, to identify the essential features.

With the Index value (Open variable) as label value, the relevant variables (features) are selected, using CFS, and Greedy Stepwise, which implementations and functionality can be revisited in [34], [35].

Thirty Enterprises (Attributes) are selected as relevant for the index data tendency: MAR, INTC, GD, IPG, KEY, KO, APD, APH, BK, BRK-A, UTF, CHK, FDX, F, ECL, PKI, ORCL, SPGI, SHW, SCHW, RSG, MMC, WY, USB, TXT, ABT, PBCT, CSCO, CMCSA, ADP. The Merit, or Pearson's correlation coefficient value, is 0.998.

The computed values of enterprises were obtained training the data of each representative enterprise, with the same model architecture used for the SP&500 Index forecasting. See the real and calculate values of variable Open of thirty representative enterprises for the Case 1259 (Table 6). The value of the Open variable of the day 06/07/2018 is forecasted using the values of variables for day 06/06/2018.

D. The Investment Decision

The investment decision depends on the consideration of the forecasting of the Index value behaviour for the values of variable Open, and the behavior of the group of enterprises identified by feature selection. So, in Case 1259 the Index value has a positive variation (UP with OpenPerc value of 0,24% or 2759,86 for Open variable). Alike, the two-thirds of selected enterprises information show, as the Index, a current similar representative tendency, UP (Table VI).

Then, according to the criteria shown in Table II, to maintain the stocks of the Index, it is a good option as an investment decision, for the next day.

V. DISCUSSION AND CONCLUSIONS

The process of computing one forecasting value is laborious (in this case, the next day Open value). First, it is necessary to represent data, then forecast it. According to Prastyo *et al.* [41], the extrapolated forecasted values are successively less exact, as shown in their research. This fact corroborates that the Neural Nets can exhibit abnormal behaviour in extrapolation cases. Here, a heuristic is used to control the possible anomalies: to learn about a topic it is necessary to know it; then, in the learning process, only previous related data is used, and a sliding window technique is adopted. This technique updates the data for the forecasting process, in a step by step fashion (day by day). Besides, the input variables of the phenomenon procure to transform the extrapolation into an interpolation problem.

The values generated by the DNN model are significant, as is proved with the criterion of adjustment. As more detailed in time are data, more exact the forecasting will be, because more prediction variables can be defined; besides, the data synthesis enables other prediction periods (minute, hour, daily, weekly, etc.). Thus, the criteria used in this work are valid for short-term decision making, that is, step by step.

Additionally, the results of this research show, if compared with original variables, few relevant enterprises as representative of the S&P500 Index. That is, it is possible to diminish the complexity and noise of data and facilitate the data analysis and decision process. The information of enterprises complements the decision criteria. Nevertheless, it is necessary more empirical result analysis to improve the model architecture and the integration with enterprises data.

Regard to the analysed period; it is essential to show that it is conditioning the results validity; in this case, to the daily period of the S&P500 Index. Further, it is possible to analyse each enterprise or enterprise group, using several stock market indexes.

The model results (representation and forecast) can be considered as process improvement, possibly to be used as a component of a Decision Support System (DSS) for stock investments. This is a topic that requires more detailed theoretical and empirical studies.

Finally, this research describes a theoretical and a practical tool for academics and practitioners. The academics can revisit a new experience of using alternative data mining and learning processes. To the practitioners, it contributes to the expansion of the existing knowledge, concerning the criteria to assist the stock market investments.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Montenegro conceived the investigation. Montenegro and Molina processed and analysed the data, wrote the document and approved the final version.

References

[1] F. Mostafa, T. Dillon, and E. Chang, "Computational intelligence applications to option pricing, volatility forecasting and value at risk, Studies in Computational Intelligence," *Springer International Publishing*, vol. 697, 2017.

[2] Y. Fang, "Feature selection, deep neural network, and trend prediction," *Journal of Shanghai Jiaotong University (Science)*, vol. 23, no. 2, pp. 297–307, 2018.

[3] X. Zheng and B. Chen, "Stock market modeling and forecasting," *LNCIS*, Vol. 442, pp. 1–11, London: Springer-Verlag, 2013.

[4] M. Olden, "Predicting stocks with machine learning," Master's Thesis, 2016.

[5] X.-Y. Qian and S. Gao, "Financial series prediction: Comparison between precision of time series models and machine learning methods," *Mathematics, Computer Science, Economic*s, 2017.

[6] S. Banik and A. Khan, "Forecasting US NASDAQ stock index values using hybrid forecasting systems," in *Porc. 18th International Conference on Computer and Information Technology (ICCIT)*, 2015.

[7] Z. Cao, L. Wang, and G. Melo, "Multiple-weight recurrent neural networks," in *Proc. the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, 2017.

[8] R. Singh and S. Srivastava, "Stock prediction using deep learning," *Multimedia Tools Application*, vol. 76, pp. 18569–18584, 2017.

[9] B. Yong, M. Rahim, and A. Abdullah, "A stock market trading system using deep neural network," *AsiaSim 2017*, Part I, Singapore, 2017.

[10] B. Yang, Z. Gong, and W. Yang, "Stock market index prediction using deep neural network ensemble," in *Proc. the 36th Chinese Control Conference*, Dalian, 2017.

[11] J. Ma, M. Yu, S. Fong, K. Ono, E. Sage, B. Demchak, R. Sharan, and T. Ideker, "Using deep learning to model the hierarchical structure and function of a cell," *Nature Methods*, pp. 1-12, 2018.

[12] P. Addo, D. Guegan and B. Hassani, "Credit risk analysis using machine and deep learning models," *Documents de Travail du Centre d'Economie de la Sorbonne*, 2018.

[13] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks," *FAU Discussion Papers in Economics*, 2017.

[14] H. Abdou, "Prediction of financial strength ratings using machine learning and conventional techniques," *Investment Management and Financial Innovation*, vol. 14, no. 4, pp. 194-211, 2017.

[15] S. Edet. (July 12, 2017). Recurrent Neural Networks in Forecasting S&P 500 Index. [Online]. Available: https://ssrn.com/abstract=3001046or

[16] V. Gavrishchaka, Z. Yang, R. Miao, and O. Senyukova, "Advantages of hybrid deep learning frameworks in applications with limited Data," *International Journal of Machine Learning and Computing*, vol. 8, no. 6, pp. 549-558, 2018.

[17] V. Sze, Y. Chen, T. Yang, and J. Emer, "Efficient Processing of Deep Neural Networks: A Tutorial and Survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2296-2329, 2017.

[18] T. Cook and A. Hall, *Macroeconomic Indicator Forecasting with Deep Neural Networks*, 2017.

[19] M. Abe and H. Nakayama, *Deep Learning for Forecasting Stock Returns in the Cross-Section*, 2017.

[20] A. Moghaddama, M. Moghaddamb, and M. Esfandyaric, "Stock market index prediction using artificial neural network," *Journal of Economics, Finance and Administrative Science*, vol. 21, pp. 89–93, 2016.

[21] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157-1182, 2003.

[22] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, pp. 16–28, 2014

[23] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics Review*, vol. 23, no. 19, pp. 2507–2517, 2007.

[24] J. Miao and L. Niub, "A survey on feature selection," *Information Technology and Quantitative Managemen*, 2016.

[25] J. Dy and C. Brodley, "Feature selection for unsupervised learning," *Journal of Machine Learning Research*, vol. 5, pp. 845–889, 2004.

[26] M. Law, M. Figueiredo, and A. Jain, "Simultaneous feature selection and clustering using mixture models.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1154-1166, 2004.

[27] R. Kohavi and G. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, pp. 273-324, 1997.

[28] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, *Feature Selection for High-Dimensional Data*, Switzerland: Springer International Publishing, 2015.

[29] H. Abusamra, "A comparative study of feature selection and classification methods for gene expression data of glioma," *Procedia Computer Science*, vol. 23, pp. 5-14, 2013.

[30] R. Kaur, M. Sachdeva, and G. Kumar, "Study and comparison of feature selection approaches for intrusion detection," *International Journal of Computer Applications*, 2016.

[31] E. Mahsereci, S. Ayşe, and T. İbrikçib, "A comparative study on the effect of feature selection on classification accuracy," *Procedia Technology*, vol. 1, pp. 323-327, 2012.

[32] W. Puch, E. Goodman, M. Pei, L. Chia-Shun, P. Hovland and R. Enbody, "Further research on feature selection and classification using genetic algorithm," in *Porc. International Conference On Genetic Algorithm*, 1993.

[33] B. Arguello, "A survey of feature selection methods: Algorithms and software," *Austin*, 2015.

[34] The University of Waikato, *WEKA Manual for Version 3-7-8*, Hamilton: New Zealand, 2013

[35] I. Witten, F. Eibe and M. Hall, "Data Mining: Practical machine learning tools and techniques," *The Morgan Kaufmann Series in Data Management Systems*, 2011.

[36] E. Barnard and L.Wessels, "Extrapolation and interpolation in neural network classifiers," *IEEE Control Systems Magazine*, vol. 12, no. 5, pp. 50-53, 1992.

[37] P. Haley and D. Soloway, "Extrapolation limitations of multilayer feedforward neural networks," in *Proc. International Joint Conference on Neural Networks*, Baltimore, 1992.

[38] A. Pektas and H. Cigizoglu, "Investigating the extrapolation performance of neural network models in suspended sediment data," *Hydrological Sciences Journal*, vol. 62, no. 10, pp.1694-1703, 2017.

[39] P. Hettiarachchi, M. Hall and A. Minns, "The extrapolation of artificial neural networks for the modeling of rainfall-runoff relationships," *Journal of Hydroinformatics*, vol. 07, no. 4, pp. 291-296, 2005.

[40] MathWorks, *Neural Network Toolbox™. User's Guide. R2014a*, The MathWorks, Inc., 2014.

[41] A. Prastyo, D. Junaedi, and M. Sulistiyo, "Stock price forecasting using artificial neural network," in *Proc. Fifth International Conference on Information and Communication Technology (ICoICT)*, 2017.

Carlos Montenegro got the MSc. in informatics and computer sciences in 2001. Currently, he is a professor at Escuela Politécnica Nacional (EPN), Ecuador, and ICT consultant. Previously, he has been the dean of Faculty and CEO of the Department of Informatics and Computer Sciences. He has also been an expert witness in various national security incidents. His academic interest areas are machine learning, data mining, artificial intelligence, and ICT management. Montenegro is author and co-author of more than twenty publications in the areas of interest.

Marco Molina was born in Alausi, Ecuador. He received his electrical engineer in 1982; the master on computer science from Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil; the PhD on software and systems, Universidad Politécnica de Madrid, Spain in 2017. Marco's major field of study is data science. He was the director of the Ecuadorian Professional Training Center, and dean of the Computer Systems Faculty at Escuela Politécnica del Ejercito, Ecuador. Currently, he is a professor at Escuela Politécnica Nacional, Ecuador.