Investment portfolio trading based on Markov chain and fuzzy logic

Riemann Ruiz-Cruz and Armando D. Diaz-Gonzalez

***Abstract*—In the present paper, a trading strategy is proposed for a portfolio composed of shares in the stock exchange. The proposed strategy is based mainly on three blocks: 1) a K-means clustering algorithm is used to determine and learn the internal hidden patterns in the time series of stock market prices, 2) a pattern predictor is performed based on a simple Markov chain, and 3) a fuzzy inference system take the decision to trade based on the estimation. The fuzzy inference system is composed by the rules provided by a expert trader. The performance of the trading algorithm is validated through simulations using real prices of the Mexican stock exchange.**

I. Introduction

The stock market is constantly evolving because the technology has made it possible for trading operations to be carried out faster and take advantage of market opportunities. Algorithmic trading is the process which use computers programmed to define and execute a set of specific instructions for placing a trade order to generate profits at a speed and frequency that is impossible for a human trader. The defined rules set are based on timing, price quantity or mathematical model [1], [2], [3].

The common trading strategies used in algorithmic trading are the Trend Following Strategies, Arbitrage Opportunities, Index Fund Re-balancing, Trading Range (Mean Reversion), Volume Weighted Average Price (VWAP), Time Weighted Average Price (TWAP), Percentage of Volume (POV), Implementation Shortfall and Mathematical Model Based Strategies [3][4]. Due to the growth of the use of this type of algorithms and the relevant importance in the large securities markets, recently have been proposed regulations for this type of operations in the stock market [5], [6].

Strategies based on mathematical models, the machine learning and artificial intelligence have served as the basis for trading algorithm design since 15 years ago. In [7], a genetic algorithm based on a fuzzy neural network is proposed as knowledge base to measure the qualitative effect on the stock market. The qualitative effect estimated is used by the neural network to take the trading decision. The complexity in the implementation of these algorithms makes it necessary a high computing capacity to perform on-line trading. In [8], a trading system where fuzzy logic is applied for defining the trading rules and managing the capital to invest. The fuzzy trading system is tested in the two markets (NASDAQ100 and EUROSTOXX) where the main conclusion suggest that the use of fuzzy logic is a good tool for the capital management. In [9], a gated Bayesian network model is used to learn the buy and sell decisions in a trading systems; and the trading algorithm performance is compared to the benchmark investment strategy buy-and-hold.

Some strategies focus on making the behavior estimation of the instrument price through propose a mathematical model and, based on that estimation make the trading decision. In [10], the empirical performance comparison between a threat-based, reservation price average price and buy-and-hold algorithms is presented. In that, the performance of the threat-based algorithm is better than all investigated algorithms. In [11], an algorithm to price the current value of an option assuming that there are no arbitrage opportunities, is proposed.

In this paper, a trading strategy is proposed for a portfolio composed of shares in the stock exchange. The proposed strategy is based on the K-means clustering algorithm to determine and learn the internal hidden patterns in the time series of stock market prices. Additionally, the trade decision is based on a Markov chain and fuzzy inference system. The Markov chain is used to estimate the pattern in a future time, and based on that estimation the fuzzy inference system take the decision to trade. The fuzzy inference system is composed by the rules provided by a expert trader. The performance of the trading algorithm is validated through simulations using real prices of the Mexican stock exchange. This paper is organized as follows: in section II, the mathematical preliminaries are included. In section III, the trading algorithm is obtained using a clustering algorithm, Markov chain and a fuzzy inference system. The simulation results of the proposed algorithm are shown in section IV. Finally, section V presents the conclusions of the paper.

II. Mathematical preliminaries

In this section the mathematical preliminaries are included, which are used to develop the proposed trading algorithm.

A. K-Means

The clustering algorithm K-Means is a method widely used to automatically find clusters by means the data set partition into K groups[12]. The algorithm consists of grouping the elements of a data set based on their proximity to a centroid that is initially chosen randomly.

Given a data set with n observations as m-dimensional real vectors, the K-means clustering groups the n observations in K partitions to minimize the within-cluster sum of square

Riemann Ruiz-Cruz and Armando D. Diaz-Gonzalez are with Research Laboratory on Optimal Design, Devices and Advanced Materials -OPTIMA-, Department of Mathematics and Physics, ITESO, Periférico Sur Manuel Gómez Morín 8585 C.P. 45604, Tlaquepaque, Jalisco, México (e-mail: riemannruiz@iteso.mx, armando.diaz.glez@gmail.com)

distances. Then, the objective is to find

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{K} \sum_{x \in S_i} \|x - \mu_i\|^2 \tag{1}$$

where, S_i is a partition of the K partitions ($\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$), μ_i is the centroid of the observations in S_i[13].

Given an initial set of K means or centroids $\{\mu_1, \mu_2, \ldots, \mu_K\}$, the algorithm proceeds iteratively in two steps[14]:

- Assignment step: Assign each observation to the cluster whose mean has the least squared Euclidean distance, this is intuitively the "nearest" mean. $S_i^{(t)} = \Big\{x_p : \|x_p - \mu_i^{(t)}\|^2 \leq \|x_p - \mu_j^{(t)}\|^2 \forall j, 1 \leq j \leq k\Big\}$, where each x_p is assigned to exactly one $S^{(t)}$, even if it could be assigned to two or more of them.
- Update step:Calculate the new centroids of the observations in the new clusters. $\mu_i = \bar{x}, x \in S_i$.

B. Markov chains

Markov chains [15], [16], are structured mathematical models based on linear algebra. This is a good tool for the study of the processes behavior that evolve non-deterministically. The Markov chains are used to describe experiments that are carried out several times in the same way and in which the result of each of the next test depends solely on the test immediately preceding.

The Markov chain is a aleatory process without memory, where the possible values from a aleatory variable are named states. This is, a variable X is in the state i_k, if at the instant $k-1$ it is in the state i_{k-1}, then:

$$\begin{aligned} P(X_k &= i_k | X_0 = i_0, X_1 = i_1, \ldots, X_{k-1} = i_{k-1}) \\ &= P(X_k = i_k | X_{k-1} = i_{k-1}) = p_{i_{k-1}, i_k}. \end{aligned} \tag{2}$$

The Markov chain can be describe as the matrix form as [16]. Given $p_{1,k}, p_{2,k}, \ldots, p_{n,k}$ as the probabilities of the n states at the time k and $p_{ij,k}$ as the probability to move from state i to state j at the time k, with $1 \leq i, j \leq n$, then the Markov chain matrix form can be expressed as:

$$\begin{bmatrix} p_{1,k+1} \\ \vdots \\ p_{n,k+1} \end{bmatrix} = \begin{bmatrix} p_{11,k} & \cdots & p_{1n,k} \\ \vdots & \ddots & \vdots \\ p_{n1,k} & \cdots & p_{nn,k} \end{bmatrix} \begin{bmatrix} p_{1,k} \\ \vdots \\ p_{n,k} \end{bmatrix}. \tag{3}$$

The probabilities matrix $M = \begin{bmatrix} p_{11,k} & \cdots & p_{1n,k} \\ \vdots & \ddots & \vdots \\ p_{n1,k} & \cdots & p_{nn,k} \end{bmatrix}$ is namely the transition matrix and the sum of each column is 1. If the matrix M is independent of the time k, the Markov chain is stationary and the states at the time m can be obtained as:

$$\begin{bmatrix} p_{1,k+m} \\ \vdots \\ p_{n,k+m} \end{bmatrix} = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix}^m \begin{bmatrix} p_{1,k} \\ \vdots \\ p_{n,k} \end{bmatrix}. \tag{4}$$

C. Fuzzy Logic

Fuzzy inference systems also known as fuzzy rule-based systems or fuzzy models are schematically shown in Fig. 1. They are composed of the following blocks [17]:

- a rule base, which contains a number of fuzzy if-then rules,
- a database which defines the membership functions of the fuzzy sets used in the fuzzy rules,
- a decision-making unit which performs the inference operations on the rules,
- a fuzzification interface which transform the inputs into degrees of match with linguistic values,
- a defuzzification interface which transform the fuzzy results of the inference into a output.

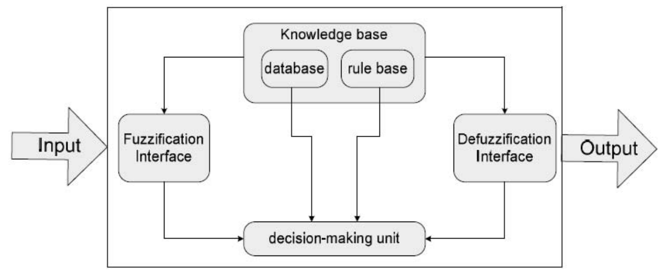

Fig. 1. Fuzzy inference system

Generally, the inference mechanism is a set of Mandani rules with the following structure:

$$If\ A_{i1}(x_1), A_{i2}(x_2), \ldots, A_{im}(x_m)\ then\ Y\ is\ B_i,$$

where, $x_1, x_2, \ldots, x_m$ are the input variables, $A_{ij}(x_j)$, $(j = 1, 2, \ldots, m)$ is a fussy set in x_j, Y is an output variable, B_i is a fuzzy set on Y. $A(x_i)$ is a semantic function which gives a "meaning" (interpretation) of a linguistic value in terms of the quantitative elements of X, i.e.,

$$A(X) : LX_i \rightarrow \ \mu_{LX}(x), \tag{5}$$

where $A(X)$ is a denotation for a fuzzy set or a membership function defined over X. Considering that the fuzzy processing was realized by the Mandani method, the inputs of the inference engine are fuzzy sets and the output is also a fuzzy set [18].

The Sugeno method is proposed to change the consequent part of the Mandani rules as a mathematical function of the input variables. The format of the rule is now:

$$If\ A_{i1}(x_1), \ldots, A_{im}(x_m)\ then\ Y = f(x_1, \ldots, x_m)$$

The antecedent part is similar to the Mandani method, and the function f in a consequent is usually a simple mathematical function, which generally is proposed as linear or quadratic function.

$$f = a_0 + a_1 x_1 + a_2 x_2 + \ldots + a_m x_m \tag{6}$$

The antecedent part in this case is processed in exactly the same way as the Mandani method, and then an obtained degree of applicability is assigned to the value of Y calculated as the function of real inputs [18].

D. Portfolio modeling

A portfolio is a grouping of financial assets such as stocks, bonds and cash equivalents, as well as their funds counterparts, including mutual, exchange-traded and closed funds. Portfolios are held directly by investors and/or managed by financial professionals [19]. Particularly for a shares portfolio, the portfolio value can be modeled in discrete-time as [2]:

$$\begin{aligned} v_{p,k} &= p_k n_{a,k} + m_k, \\ m_{k+1} &= m_k - p_k u_k - r_{com} p_k \|u_k\|, \\ n_{a,k+1} &= n_{a,k} + u_k, \end{aligned} \tag{7}$$

where, $v_{p,k}$ is the total portfolio value, m_k is the amount available to invest, $n_{a,k}$ is a shares number, r_{com} is the percentage of the intermediary charges for the operations, u_k is the shares number that will be buy or sale and p_k is the stock market price of the corresponding asset. Considering the limitations of the investment, the control signal u_k is bounded as follows:

$$u_{min} \leq u_k \leq u_{max}, \tag{8}$$

where, the control signal limits are defined as $u_{max} = floor\left(\frac{m_k}{(1+r_{com})p_k}\right)$ and $u_{min} = -n_{a,k}$.

III. Trading algorithm design

In the stock market, a share price is established by the price of offer and demand.The share prices can be considered as a stochastic time series, where its estimation turns out to be a great challenge that is still trying to solve because it is useful information to make decisions. A price time series from the stock market of Mexico are shown in Fig. 2. In this figure, it can be seen that the shares of different companies have a different price level and the time behavior is different too.

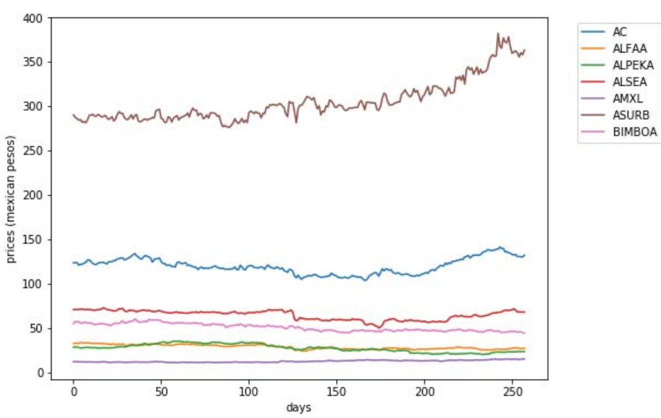

Fig. 2. Share prices from Mexican stock market

Frequently, stock traders use analytic methods as linear or exponential regressions, moving averages and economic indicators that can give information to make the decision to buy or sell shares. This type of analysis is commonly called technical analysis. Some of these methods are based on the ability of the trader to determine patterns in the price time series and based on these take the decisions, which could be predetermined for those patterns. Knowing this, in this paper an unsupervised clustering algorithm is used to determine patterns in the price time series of 32 shares of the stock market of Mexico.

A. Determination of patterns in market prices

The objective is to be able to determine if there are common patterns within the 35 prices time series available. The patterns that interest us to determine mainly are: 1) the movements of change of direction to the upside, 2) change of direction to the fall, 3) upward movements and 4) downward movements. In the Fig. 3, the patterns previously mentioned are shown over the prices time series.

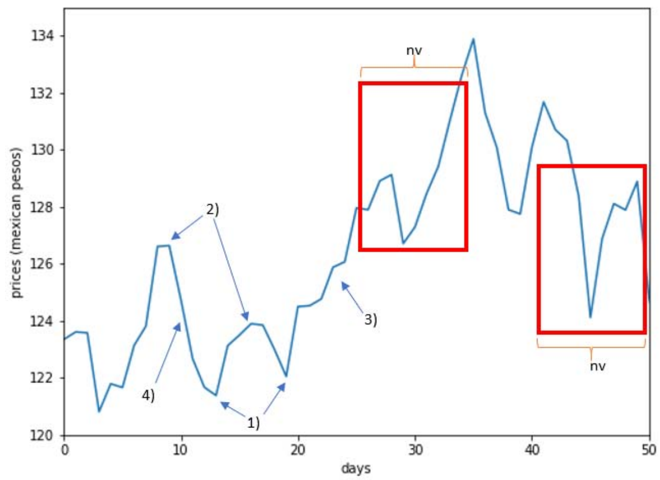

Fig. 3. Important patterns within a prices time series

In order to determine these patterns we must consider that the time series are composed of time subseries, which have different patterns among them. In Figure 3, it can be seen that two subsections of length nv are highlighted within the time series, it can be easily seen that they have different behaviors between them, then these time subseries can be grouped by means of a clustering algorithm. Then, the 32 time series of stock market shares of length n days will be converted into $n - nv$ time subseries of length nv days. Then, the nv value correspond to the time's window selected to analyze the patterns, for our case, the nv value is chosen as 5 (the time window is 5 because a week operational for the stock exchange have 5 days).

The sub-time series created will be classified by means of the K-means clustering algorithm, which is described in subsection II-A. The K-means algorithm use the euclidean distance as similarity index to group the data in k clusters. Because of this, it is important to consider that the time subseries comes from different shares, then the prices level can vary considerably and it would be a problem for the clustering algorithm to classify them correctly. That is, an upward pattern of an action with a price between $\$4 - \$5$ can not be directly compared with an upward pattern of an action with a price between $\$150 - \$160$, because based on the similarity index they would be classified as different patterns.

In order to remove the dependence of the different price levels and that the patterns are comparable to each other, all time subseries are standardized under the following criteria:

$$x'_i = \frac{x_i - \mu_{x_i}}{\sigma_{x_i}},\ (i = 1, 2, \ldots, n - nv) \tag{9}$$

where x_i is the data vector corresponding a one time subseries, μ_{x_i} is the mean of the data vector x_i, σ_{x_i} is the variance of the data vector x_i and x'_i correspond to the new vector with the standardized values.

Then, all the standardized time subseries are included in a one dataset as follows:

$$X' = \begin{bmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_{n-nv} \end{bmatrix} \tag{10}$$

The K-means algorithm made a data classification into $K = 4$ groups, of which the centroid of each group can be interpreted as the average pattern of the group. The patterns determined by the algorithm are shown in Figure 4.

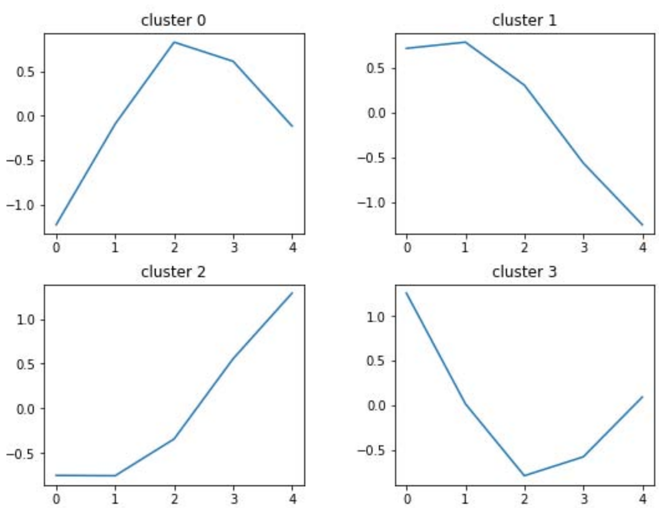

Fig. 4. K-means patterns classification

Now the clustering algorithm is trained to be able to classify time subseries of length nv, and if it is execute on-line, the new data will be classified in some of 4 patterns determined by the K-means. These classifications can be interpreted by the trader as signals of buy and sale of shares.

B. Patterns prediction by Markov chain

The classifier developed in subsection III-A has the characteristic that it can classify the time subseries as long as it already happened. This is because, it is necessary to know the all data in the corresponding time window in order to be classified.

Then, a Markov chain as (3) is proposed to estimate the change from one pattern to another in the prices time series. The states for the Markov chain are the each cluster determined by the K-means. The transition matrix of the Markov chain proposed is defined as follows:

$$M = \begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \\ p_{30} & p_{31} & p_{32} & p_{33} \end{bmatrix}, \tag{11}$$

where, each element p_{ij} in the matrix M is the empirical probability of passing a cluster j to a cluster i. These empirical probabilities are obtained as results of using the previously designed classifier in the data of the original 32 time series. Then, the transition matrix obtained is

$$M = \begin{bmatrix} 0.30793 & 0.00408 & 0.39850 & 0.10207 \\ 0.43001 & 0.57405 & 0.02838 & 0.16789 \\ 0.15785 & 0.02400 & 0.56668 & 0.41420 \\ 0.10419 & 0.39785 & 0.00642 & 0.31582 \end{bmatrix}. \tag{12}$$

The Markov chain is used to estimate the probability to change a cluster knowing a priori the cluster in which the current pattern is. The prediction is calculated as (3), where the states vector si defined as:

$$P_k = \begin{bmatrix} p_{1,k} \\ \vdots \\ p_{n,k} \end{bmatrix}, \tag{13}$$

where, $p_{i,k}$ is the probability of being in the current pattern.

C. Trading decision based on fuzzy logic

For making decisions to buy and sell shares, a fuzzy inference mechanism is proposed, which is based on rules type Mandani. The Mandani rules are proposed by a trader, which are based on the current pattern and the expected next pattern. The decision or consequent is to propose the value that u should have in the model (7). This value is the amount of shares that will be sold, bought or none of the above. Considering The Mandani rules are classified according to the current pattern as follows:

If the current pattern is cluster 0

1) $If \quad p_{0,k} = 1, p_{0,k+1} = 1 \quad then \quad u_{0,k} = 0$
2) $If \quad p_{0,k} = 1, p_{1,k+1} = 1 \quad then \quad u_{0,k} = u_{min}$
3) $If \quad p_{0,k} = 1, p_{2,k+1} = 1 \quad then \quad u_{0,k} = u_{max}$
4) $If \quad p_{0,k} = 1, p_{3,k+1} = 1 \quad then \quad u_{0,k} = u_{max}$

If the current pattern is cluster 1

1) $If \quad p_{1,k} = 1, p_{0,k+1} = 1 \quad then \quad u_{1,k} = u_{min}$
2) $If \quad p_{1,k} = 1, p_{1,k+1} = 1 \quad then \quad u_{1,k} = u_{min}$
3) $If \quad p_{1,k} = 1, p_{2,k+1} = 1 \quad then \quad u_{1,k} = u_{max}$
4) $If \quad p_{1,k} = 1, p_{3,k+1} = 1 \quad then \quad u_{1,k} = u_{max}$

If the current pattern is cluster 2

1) $If \quad p_{2,k} = 1, p_{0,k+1} = 1 \quad then \quad u_{2,k} = u_{min}$
2) $If \quad p_{2,k} = 1, p_{1,k+1} = 1 \quad then \quad u_{2,k} = u_{min}$
3) $If \quad p_{2,k} = 1, p_{2,k+1} = 1 \quad then \quad u_{2,k} = 0$
4) $If \quad p_{2,k} = 1, p_{3,k+1} = 1 \quad then \quad u_{2,k} = u_{max}$

If the current pattern is cluster 3

1) $If \quad p_{3,k} = 1, p_{0,k+1} = 1 \quad then \quad u_{3,k} = u_{min}$
2) $If \quad p_{3,k} = 1, p_{1,k+1} = 1 \quad then \quad u_{3,k} = u_{min}$
3) $If \quad p_{3,k} = 1, p_{2,k+1} = 1 \quad then \quad u_{3,k} = u_{max}$
4) $If \quad p_{3,k} = 1, p_{3,k+1} = 1 \quad then \quad u_{3,k} = 0$

The control limits u_{min} and u_{max} of the model (7) are defined as (8). Due to the proposed rules only consider the options when the probabilities are 1, that is, that we are sure of knowing the next pattern, the real u_k value is obtained by the Sugeno method (6). Then the control signal is defined as:

$$u_k = p_{0,k+1}u_{0,k}+p_{1,k+1}u_{1,k}+p_{2,k+1}u_{2,k}+p_{0,k+1}u_{3,k}, \quad (14)$$

where, $p_{i,k+1}, \ (i = 1, 2, 3, 4)$ are the probabilities obtained from the Markov chain and $u_{i,k} \ (i = 1, 2, 3, 4)$ are the consequents recommended for the expert trader. In general, the decision made for the portfolio is obtained by

$$u_k = \sum_{i=1}^{K} p_{i,k+1}u_{i,k}, \quad (15)$$

where K are the clusters number used in the clustering algortihm.

IV. Simulation results

In order to test the behavior of the proposed trading algortihm, simulations with real stock shares were implemented. Software used for simulation was Python and the stock prices were obtained from Yahoo Finance[1]. Simulation conditions for all tests scenarios are the following:

- Amount to invest: $m_0 = 50000$ MXN (Mexican pesos)
- Operation commission: $r_{com} = 0.29\%$
- Simulation period: 51 days (03/07/2017 − 05/23/2017)
- Shares to test: GRUMAB, AMXL and BOLSAA from Mexican stock exchange, which are chosen for comparison to [2].
- For the simulations it is considered that the buying and selling operations are executed instantly.

Fig. 5-7 will show: 1) stock value variations during the given period, 2) portfolio value calculation based on the control algorithm, 3) control signal behavior.

In Fig 5, simulation results using GRUMAB stock prices are shown. In top graph, the GRUMAB prices behavior in the given period is displayed. The portfolio value behavior and the control signal are presented in middle and down graphs respectively. It can be seen that, the portfolio value behaves similarly to the behavior of the share price, so the portfolio has a negative performance because the action also has a negative behavior. In the period of the simulation, the share price is depreciated 10.83% of the initial value, and the portfolio only had a loss of 6.52%. Then, the trading algorithm attenuates the losses although a commission (r_{com}) percentage is charged for each operation.

In Fig. 6, results with the AMXL stock prices are presented. Similar to the previous behavior, the portfolio value has positive performance mainly because the share price in the simulated period has a positive behavior. In this case the earnings of the share price was 11.79% and the portfolio had a return of 4.05%. In the same way, for the BOLSAA stock prices, the earnings of the share price was 10.78% and the portfolio had a return of 2.41%, which can be seen in the Fig. 7. From the simulation results, we can say that this algorithm has a good behavior to limit the possible losses.

[1]https://finance.yahoo.com/

In Table I, a resume of the results obtained with trading algorithm proposed is compared with the results obtained with the technique to "Buy-and-Hold" that is very used by investors who want long-term returns.

TABLE I
Results comparison.

Algorithm	GRUMAB	AMXL	BOLSAA
Buy-and-Hold	-11.35%	11.14%	10.14%
Trading algorithm proposed	-6.52%	4.05%	2.41%

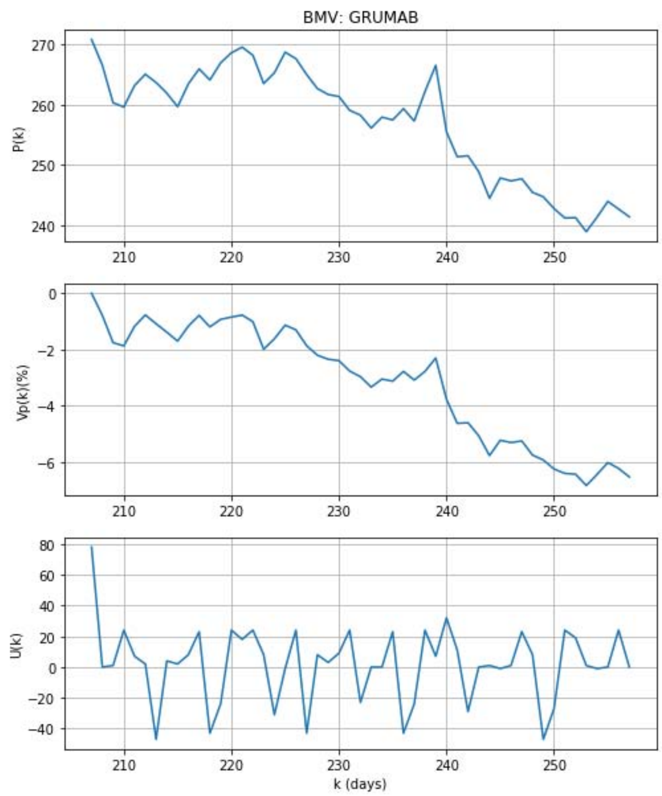

Fig. 5. Simulation results of the GRUMAB investment controlled with the algortihm proposed

V. Conclusions

In the present paper, based on the K-means clustering algorithm, a identification of the internal patterns of the time series of the price shares is obtained. The estimation of the pattern in the next time $(k + 1)$ is obtained by means a simple Markov chain. The matrix transition is calculated from the empirical probabilities once the clustering algorithm was applied to the original time series.

From the simulations results, it can be seen that the trading algorithm achieves a desired behavior when the stock price has a downward trend. This is because the algorithm proposed limits the losses of holding the shares during the duration of that trend.

The proposed algorithm gives us the idea that it is possible that the experience in technical analysis of an expert stock

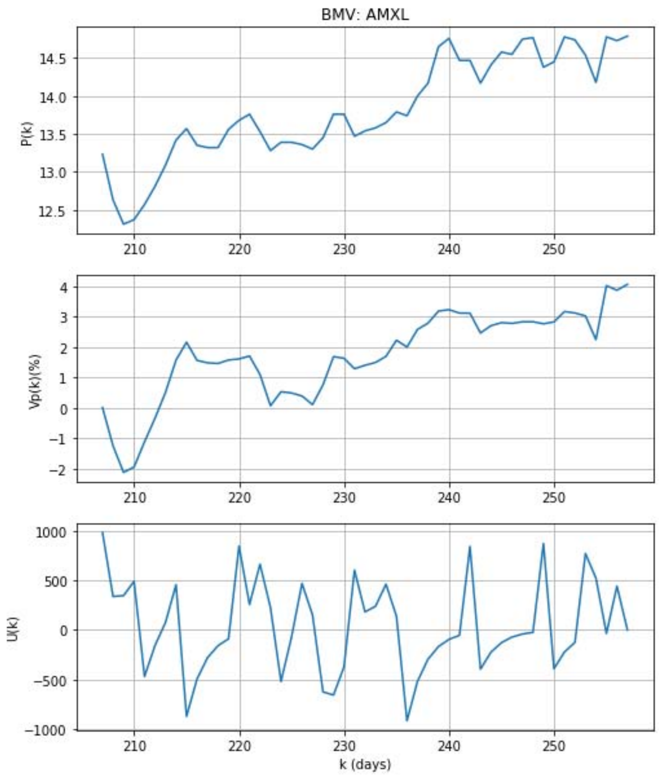

Fig. 6. Simulation results of the AMXL investment controlled with the algortihm proposed

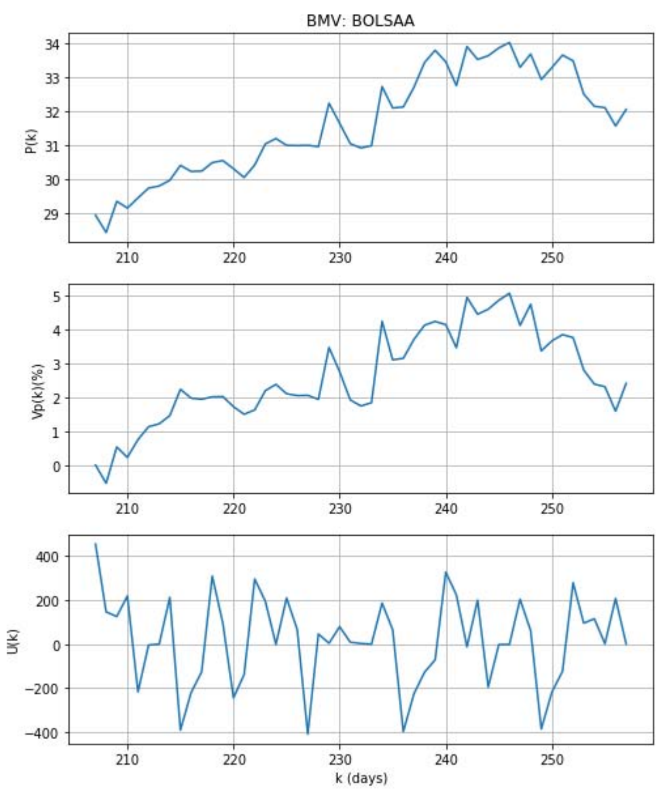

Fig. 7. Simulation results of the BOLSAA investment controlled with the algortihm proposed

trader, can largely be captured by a machine learning algorithm. Although it is proposed to have more information and parameters to train a more complex algorithm to have better results.

References

[1] S. Kirchner, "High frequency trading: Fact and fiction." *Policy*, vol. 31, no. 4, pp. 8 – 20, 2015.

[2] R. Ruiz-Cruz, "Portfolio modeling for an algorithmic trading based on control theory," in *Second Conference on Modelling, Identification and Control of Nonlinear Systems IFAC MICNON 2018*, Guadalajara, Jal, Mexico, Jul 2018, pp. 1–6.

[3] S. Seth. (2017, May) Basics of algorithmic trading: Concepts and examples. INVESTOPEDIA. [Online]. Available: http://www.investopedia.com/articles/active-trading/101014/basics-algorithmic-trading-concepts-and-examples.asp

[4] P. TRELEAVEN, M. GALAS, and V. LALCHAND, "Algorithmic trading review," *Communications of the ACM*, vol. 56, pp. 76–85, 2013.

[5] Y. Yadav, "Insider trading and market structure." *UCLA Law Review*, vol. 63, no. 4, pp. 968 – 1033, 2016.

[6] D. BUSCH, "MiFID II: regulating high frequency trading, other forms of algorithmic trading and direct electronic market access," *Law and Financial Markets Review*, vol. 10, pp. 72–82, 2016.

[7] C. C. R.J. Kuo and Y. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network," *Fuzzy Sets and Systems*, vol. 118, no. 1, pp. 21–45, February 2001.

[8] J. A. Rodrigo Naranjo, Albert Meco and M. Santos, "An intelligent trading system with fuzzy rules and fuzzy capital management," *International Journal of Intelligent Systems*, vol. 30, no. 8, pp. 963–983, April 2015.

[9] M. Bendtsen and J. M. Pena, "Gated bayesian networks for algorithmic trading." *International Journal of Approximate Reasoning*, vol. 69, pp. 58 – 80, 2016.

[10] E. M. Gunter Schmidt and M. Kersch, "Experimental analysis of an online trading algorithm," *Electronic Notes in Discrete Mathematics*, vol. 36, pp. 519–526, 2010.

[11] I. K. Peter DeMarzo and Y. Mansour, "Online trading algorithms and robust option pricing," in *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, Seattle, WA, USA, May 2006, pp. 477–486.

[12] J. MacQueen, "Some methods for classification and analysis of multivariate observations," vol. 1, 1967, pp. 281 – 297.

[13] E. S. Hans-Peter Kriegel and A. Zimek, "The (black) art of runtime evaluation: Are we comparing algorithms or implementations?" *Knowledge and Information Systems*, vol. 52, pp. 341 – 378, 2017.

[14] D. J. C. MacKay, *Information Theory, Inference adn Learning Algorithms*, C. U. Press, Ed. Cambridge University Press, 2003.

[15] E. Seneta, *Non-negative Matrices and Markov Chains*, S. Science, Ed. Springer Science, 1981.

[16] M. I. GarcÃa Planas and T. Gongadze, "Wind profile prediction using linear markov chains: A linear algebra approach," *IEEE Latin America transactions*, vol. 16, no. 2, pp. 536 – 541, 2018.

[17] K. Mehran, *Takagi-Sugeno Fuzzy Modeling for Process Control*, Newcastle University, 2008.

[18] L. REZNIK, *Fuzzy Controllers*, 1st ed. Newnes, 1997.

[19] INVESTOPEDIA. (2017) Portfolio. INVESTOPEDIA. [Online]. Available: http://www.investopedia.com/terms/p/portfolio.asp