Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 11, No. 1, February 2020), Pages: 1-17
www.jacr.iausari.ac.ir

Stock Trading Signal Prediction Using a Combination of K-Means Clustering and Colored Petri Nets (Case Study: Tehran Stock Exchange)

Ali Ghorbani[1], Mahmood Yahyazadehfar[2]✉, Seyed Ali Nabavi Chashmi[1]
1) Department of Financial Management, Babol Branch, Islamic Azad University, Babol, Iran
2) Faculty of Economics and Administrative Science, University of Mazandaran, Babolsar, Iran
ghorbani_letter@yahoo.com; m.yahyazadeh@umz.ac.ir; anabavichashmi2003@gmail.com

Received: 2019/09/06; Accepted: 2020/09/22

Abstract

Stock markets are attractive in nature for investors to gain profit. However decision making about suitable points of trading is a challenging issue, due to various properties of stocks, unstable values and data frequencies. Predicting stock price movements and discovering turning points using technical indicators, for the sake of data frequency reduction in short-term, is a preferred choice in comparison with price forecasting which commonly uses fundamental analysis. In this ambit, this paper proposes a Colored Petri Net model combined with k-means clustering decision making rules to predict stock trading signal, namely buy, sell, and hold, enhanced by a strength coefficient in a 7-step process. The paper focuses on Tehran stock exchange as case study in a two-year time interval. Simulation results implies superiority of proposed model against other state-of-the-art approaches, i.e. artificial neural networks, decision tree, and linear regression, with the accuracy rate of 88% in term of correctly classifying.

Keywords: *Colored Petri Nets, K-Means Clustering, Technical Analysis, Stock Trading Signal*

1. Introduction

Investment in stock market is one of the common ways to gain profit for investors. Effort to achieve more profit and avoid losses is the chief among concerns in this market. Two main strategies in this ambit have been emerged in previous researches: forecasting stock price and predicting the direction and trend of stock [1]. These two strategies' difference is due to difference of fundamental analysis and technical analysis. Fundamental analysis deals with assessment of an individual company share price by considering company's last prices conditions (e.g. earning, overall economy, ...).

In fundamental analysis if a stock is undervalued and company fundamentals are strong, then buy signal will be sent to investors. In contrast if the stock is overvalued and stock fundamentals are weak, sell signal will going to be sent to investors [2].

The technical analysis involves many statistical data of market (e.g. stock open price, close price, volume of stock, ...). Technical analysis focuses on finding investors sentiments to predict the overall trend of stock [3, 4].

Price prediction has motivated many researchers to study in this area. But price changes in short-term is highly challenging under frequency of data characteristics. Whereas in the other major direction, technical analysis, the turning points (trading signal) commonly have longer periods in short-term, so the high frequency characteristics of data is accordingly reduced, and the results are generally promising in comparison with price prediction [1, 5, 6]. Investigations indicate that some certain combinations of technical indicators facilitate forecasting of financial market exchanges with a high precision. Therefor more benefit is gained comparing to pure trading [7]. In financial analysis, paying attention to the peak or valley of stock's price variations is beneficial. Specially, in financial time series, minimal and maximal points, and the events occurrence between aforementioned points in the form of technical indicators could contain some valuable information which guides investors to make decision if to buy/hold/sell a stock with the goal of maximizing profit and minimizing loss. Discovery of repetitive patterns in certain time intervals in stock market history with the goal of predicting future behavior of the market, such as what happens in time series, is a challenging issue that has motivated scientists to solve this problem by employing Artificial Intelligence (AI) techniques such as meta-heuristic algorithms [8, 9] and data mining [10-21]. The researches' outcomes indicate that AI techniques outperformed traditional statistical approaches [22].

Data mining approaches have pervasively been used for prediction in different areas [23-25]. However in data mining approaches, there is a high dependency between results and input data that decreases precision of outcomes in case of existing incorrect or faulty data. In the other hand, Meta-heuristic-based approaches are fundamentally established to work on random generated data and produce approximation solutions, so they do not guarantee that global optimized solution will be finally achieved [26-28].

Regarding to mentioned shortcomings, this paper proposes a combination of Colored Petri Net (CPN) model and some if-then rules resulted from the past historical patterns of stock situation in the market extracted by k-means clustering technique, to predict stock trading signals. CPN is an extension of classic Petri Nets (PN) which is able to model and simulate different states and behaviors of a system or process statically or dynamically without essentially need to massive volume of input data. Therefor dependency to huge amount of data is eliminated. CPN is a mathematical graph-based formal method empowered by Markup Language (ML) programming [29, 32]. So in case of feeding the CPN model with some verified data, the results are significantly precise compared to approaches based on random generated data and approximation solutions. Accordingly, the main contributions of this paper are as follow:

- Employing efficient technical indicators in stock trend variations and relevant historical data.
- Designing CPN model of signal trading system, and required mathematical functions considering if-then rules extracted by k-means clustering from historical data.
- Setting the data of technical indicators as CPN model's input data.
- Evaluating precision of prediction resulted from simulating stock trading signal system using CPN model compared to some other state-of-the-art approaches.

The rest of the paper is arranged as follows: section 2 has a review on related works. We have a brief overview on literature in section 3. Proposed model is described in section 4. Section 5 is dedicated to simulation and evaluation. Conclusion and future work directions is presented in section 6.

2. Related Works

Petri net is a powerful formal modeling and simulation language which has recently appeared in new researches to predict stock trading signal.

Authors of [34] have presented a time series petri nets which combines the control flow point view of Petri nets with time series prediction. Outcomes showed the enhancement of model by relying on time series forecasting power.

In [35] authors presented an approach to forecast stock prices in Taiwan. Firstly they simulated the trend of trading approximately on the data sets gathered daily. Then they analyzed technical indicators by using trained model, to extract likely trend pattern and make a forecasting. At last, the business behavior of financial investment systems has been modeled using a high-level fuzzy Petri net (HLFPN), aim to making a more precise decision in comparison with other candidate approaches.

Authors of [36] presented a hybrid model based on Genetic Algorithm (GA) and Fuzzy Logic Enhanced Time Petri Net (FLETPN) to explain the forecast behavior and reflections that influence the market. Specialist's rules were set by GA and historical data was employed to enhance fuzzy rules. Outcomes were near to real statistics comparing to daily forecasting.

Authors of [37] and [38] presented CPN and rough petri net (RPN) models respectively to find out the relationships among a variety of technical indicators. They both have extracted the trading rules of trading signals hidden in historical data. Their outcomes showed an acceptable return on investment. The other common way to solve stock trading signal problem is to applying Artificial Intelligence (AI) such as meta-heuristic algorithms. Meta-heuristics are generally based on a random generated data and produce approximation solutions.

In [8] a GA was used to maximize the profit in stock market. The result in term of performances achieved from trading strategies produced by using the GA were compared with the performance of the Buy-and-Hold (B&H) Strategy on Pakistan Stock Exchange (PSX) data. The strategies established by GA outperformed the B&H strategies on candidate stocks.

Authors of [9] have proposed a Genetic Network Programming (GNP) with rule association to create stock trading signals regarding upward and downward movements and events replications of specific buy/sell timing. A great amount of buy/sell rules were discovered by the individuals deducted in the training intervals. Then, a unique mechanism for classifying was used to determine whether to buy/sell stocks in an appropriate manner and using the extracted rules. In the testing step, the stock trading was extracted out using the achieved rules and it was proved that the rule-based trading model generates higher profits than the conventional individual-based trading model. The other famous AI trend to solve stock trading signal problem is to employ data-mining-based techniques.

Authors of [10] have proposed a Recurrent Neural Network (RNN) to find out specifications from the data of multi-variance market in a one-minute frequency and true market was subsequently forecasted to find trading signals. Outcomes showed the strength of deep recurrent architecture to discover the relationship between the historical behavior and specification movement for high frequency data instances. The deep RNN was then compared with SVM, random forest, logistic regression, using CSI300 one-minute data on the test duration. The outcome implied that the capability of the deep

RNN trading signals created by extreme movement forecasting improved the profitability.

In the study [11] a Neural Network (NN)-based stock price forecasting and trading system was proposed which performed by employing technical analysis indicators. The model firstly converted the financial time series data into a sequences of buy-sell-hold signals based on the preferred technical indicators. In second step, a Multilayer Perceptron (MLP) artificial NN (ANN) model was generated and trained by learning data achieved from the daily stock prices. Then the framework of Apache Spark big data was employed in the training step. The outcomes implied that when the most appropriate technical indicators are chosen, the NN model can totally gain considerable outputs against the B&H strategy. In addition, fine deducing the technical indicators and/or optimization strategy can improve the overall trading performance.

Authors of [12] have proposed a stock trading model utilizing feed-forward deep NN to predict index price belongs to Singapore stock market by employing the FTSE Straits Time Index (STI) in t days ahead. They tested the model through simulations on historical daily prices in market. The algorithm of the training step was stochastic gradient descent with back-propagation that was enhanced using a multi-core processing. The trading system simulations showed considerable outcome with a rate of 18.67 profit, 70.83% profitable trades and ratio of 5.34 Sharpe.

In study [13], a method was proposed which associated principal component analysis (PCA) into weighted support vector machine (WSVM) to forecast trading spots in the stocks. In first step, the stock trading signals forecasting was modeled as a weighted 4-class classification problem. Then, PCA was employed for cleaning the main data set and re-arrangement to another new data structure. In third step, WSVM was used over the newly structured data set to predict the turning points of the stock. At the last step simulation experiments were accomplished beside PCA-WSVM, WSVM, PCA-ANN and B&H strategies. The results showed the effectiveness of PCA-WSVM compared to others.

Authors in [14] have studied main portions of a quantitative trading system by Machine learning, in which many trading strategies were implemented adapted for real-time market. Linear Regression (LR) and SVR models were applied to forecast stock trading directions. As well, different optimization methods were used to enhance the return and reduce the risk of trading as possible. Outcomes showed that proposed strategy yielded an acceptable return than the S&P 500 ETF-SPY.

Authors of [15] presented a biclustering algorithm to discover hidden patterns in the trading data. Then the achieved patterns were utilized to predict the market direction based on the Naive Bayesian algorithm. At last, the Adaboost algorithm was employed to enhance the precision of the predictions. Considering the outcome of experiments, proposed algorithm outperformed the other four candidate approaches.

The method proposed by [16] was consist of two objectives: the first objective was selecting and identifying the technical indicator specifications by applying Boruta feature selection technique. The second objective was to creating a forecasting model for stocks prediction using ANN and Regression. The results of experiments showed decreasing the rate of error in the forecasting to 12%.

Research of [17] developed an approach based on the Piecewise LR and ANNs for analyzing the nonlinear relationships between the stock closed price and different technical indexes, and uncovering the knowledge achieved from the trading signals in historical data. The results of research showed superiority of proposed model compared to [1] and other benchmark researches.

Authors of [18] proposed an ensemble method of ANN and SVR to forecast the stock prices. Firstly, historical stock prices were divided to some subseries. Secondly, each subseries was predicted by applying machine learning algorithms. Finally, predictions of individual subseries were considered to gain the final prediction. The results of research showed that trading rules based on ensemble model gives higher return on investment in comparison with B&H strategy.

Authors of [19] have proposed an approach based on feature ranking and feature selection, combined with weighted kernel least squares support vector machines (LS-SVMs). They used the analytic hierarchy process (AHP) into the stock market. The weights of feature achieved using AHP method were used to select and rank feature, and used combined with the LS-SVMs through a weighted kernel. The experiments outcomes showed that the model outperformed other benchmark models.

In research of [20] authors proposed an approach to find styles of suitable stock investment in terms of investment time. In first stage, biclustering was employed on a matrix that was derived from technical indicators in each trading day to find out trading patterns considered as trading rules. In next stage a k-nearest neighbor (KNN) algorithm was applied to make a transformation between trading rules and trading actions (buy/sell/hold signals). In final stage, a min-max and quantization strategy was created to determine the temporal investment style in stock data. The proposed approach was applied on 30 stocks from US bull, flat and bear markets, and results implied usefulness of method.

Authors of [21] have proposed an approach to simplify financial series which were filled with noisy data. To do so sequences were reconstructed by leveraging motifs (frequent patterns), and then a convolutional NN was employed to capture spatial structure of time series. The results of simulations showed the superiority of the method in feature learning. Also it has been shown that proposed method outperformed traditional signal process methods with 4%_7% precision improvement. Some of the researches have focused on other formal methods.

In [39] authors proposed a framework based on tensor information to forecast stock directions. The proposed model has managed investor information environment with tensors. Simulations were accomplished on data of China Securities Index stocks and implied that proposed tensor-based framework outperformed the Top-N trading strategy and two state-of-the-art media-aware trading approaches.

Authors in [40] have transformed trading signals into a Fourier series and deduced financial information from the amplitude and phase of Fourier series' components. The proposed approach was implemented to US stock-indices, and results were acceptable compared to other candidate works.

In research of [41] a method was proposed to discover candlestick patterns in a stock trading system by taking advantage of fuzzy logic. The method was formed of a fuzzy trading system that employed three candlestick patterns. The results of experiments against its crisp counterpart and the classical B&H trading strategies showed that fuzzy candlestick-based trading system improved the recognition of pattern compared to crisp version, and also less risk and more stable behavior were reported against the other rival trading systems.

3. Literature Review

3.1 Colored Petri Nets

CPN is a powerful formal extension of classic PNs which supports mathematics as well as ML programming. CPNs are capable to model and illustrate different processes and systems containing hierarchy, time, and data type (in form of color) concepts. A CPN model consists of four main elements: *place*, *transition*, *arc* and *token* as shown in Figure 1. Places show the state of the system. Events are shown by transitions. When a transition fires, the system state changes according to expression of arc and expression of *guard* which appears close to transition in the model. Data are generally shown by tokens. Tokens in a CPN model could be defined of different standard (int, bool, string, real, ...) or user-defined data types in form of color sets. In addition, by using multi-sets and complicated sets (i.e. *product*, *record*) definition, a token in CPN model can carry many relevant properties with itself, such as value and name, across the model [29-33].

Formal definition of a non-hierarchical CPN is a tuple $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ which satisfies following requirements [29, 32]:

- Σ: a finite set of non-empty types (color sets).
- P: a finite set of places.
- T: a finite set of transitions.
- A: a finite set of arcs ($P \cap T = P \cap A = T \cap A = \varnothing$).
- N: a node function, defined from A into $P \times T \cup T \times P$.
- C: a color function, defined from P into Σ.
- G: a guard function, defined from T into expressions.
- E: an arc expression function, defined from A into expressions.
- I: an initializing function, defined from P into closed expressions.

In this paper we take benefit of multi-sets and product color sets to define properties of stock market input data into CPN model. Also we use arc expressions to define if-then rules in the model.

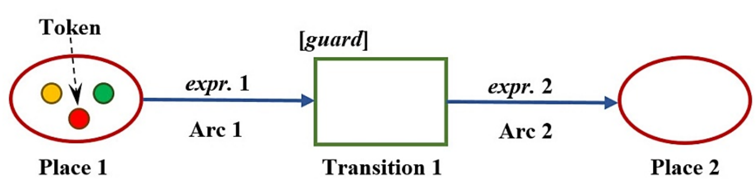


Figure 1. A CPN model elements

3.2 Data mining and k-means clustering

Data mining is the process of discovering interesting patterns and knowledge from large amounts of data. Data may be consist of database, data warehouse, web information repositories or a combination of aforementioned resources. The gathered data before being used in a data mining process should be preprocessed and cleaned to reach an acceptable quality and obtain compatibility. The most important steps of preprocess are: removing outliers and noises, missing values management, and data transformation and dimension reduction. One of the unsupervised techniques of data mining is clustering which is referred to as the process of partitioning a set of data objects into subsets, called clusters, so that objects in a cluster are similar to each other yet dissimilar to objects in other clusters. The partitioning process is performed by a clustering algorithm, and is useful to discover hidden patterns within previously existing data. k-means is the basic clustering algorithm which uses numeral distance, such as Euclidean distance, to measure similarity and dissimilarity between objects, so that dissimilarity increases as well as distance increases and vice versa. Euclidean distance between two objects, p and q, with n components is measured by equation (1):

$$Euc\_Dist(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \tag{1}$$

The quality of cluster C with central point *centroid* can be measured by *sum of squared error* between all objects in C and the *centroid* point, as equation (2), which obviously is desired to be minimized [42].

$$Error = \sum_{p \in C} Euc\_Dist(p, centroid)^2 \tag{2}$$

3.3 Stock trading signal

In finance literature, two frequently used approaches for stock occurrences forecasting are fundamental analysis and technical analysis. Fundamental analysis uses fundamental indicators of the company such as Price to Equity (P/E ratio), Return on Equity (ROE) and Earnings per Share (EPS). In technical analysis, patterns of historical stock prices are used for forecasting the future values in market [18]. Technical analysis is the process of analyzing historical stock price to forecast trend of stock prices in future. In technical analysis, assumption is that some historically formed regularities in the stock exchange will be repeated in future trading belief that “history repeats itself”. Technical analysis in combination with transaction amount produces high precision moment for stock trading. So a group of technical rules based on technical indicators have been developed [17]. A technical indicator is a mathematical calculation based on historic price, volume, or open interest information that aims to forecast financial market direction [43]. The goal of each technical rule is to generate a stock trading signal. Stock trading signal is a challenging task which helps investors to make decision and determine when to buy, sell or hold a stock. Stock trading signal normally uses technical indicators to monitor the stock price and assist investor in setting up trading rules for decision making. Stock trading signal commonly focuses on turning points in historical market data. So that, a buy signal is generated when a bull market is anticipated or a sell signal is generated when a bear market is expected, and otherwise a hold signal is generated [17].

4. Proposed Model

4.1 Overall workflow

In this paper we propose a formal model to generate stock trading signal based on Tehran stock exchange data. To do so we follow an overall roadmap, consists of seven steps. Firstly we select important fundamental indicators affecting a stock trading in the market according to opinion of domain expert. Secondly we gather values of selected indicators for each stock in a certain continuous time interval containing 3 sequential periods. In third step, we refine gathered data to observe and calculate variations of fundamental indicator values between periods 1-2, and periods 2-3. In fourth step we try to find most efficient indicators and pattern of their variations' impact on EPS of each stock, using k-means clustering technique, as technical indicators. In fifth step we extract trading rules in form of if-then rules resulted from outcome of clustering. In sixth step we construct CPN model for generating stock trading signal, define mathematical functions to calculate indicator variations, and train the model using if-then rules achieved from previous step. Finally in seventh step we feed the constructed CPN model by some test stock data records in form of tokens, to make decision for generating buy/sell/hold signal along with a strength coefficient. A brief workflow of this paper is illustrated in Figure 2.

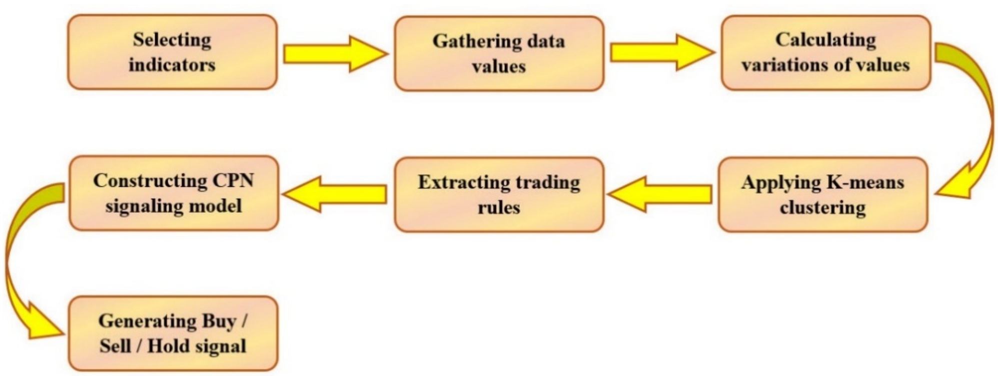


Figure 2. Overall workflow of proposed approach

4.2 Technical indicators and rules

In stock market, there is a non-linear relation between some indicators and the behavior of stock price diagram. Generally two main categorized approaches have been presented in this ambit: approaches based on fundamental indicators that mostly focus on price prediction, and approaches based on technical indicators that aim to price trend prediction using patterns of trading in historical data. In this research we select 14 financial and macroeconomic indicators according to opinion of domain expert as step 1. Then we investigate the variations of selected indicators' values in 2 sequential time durations to achieve frequent trading rules. In fact we take benefit of combining fundamental analysis and technical analysis to construct the proposed signaling model. Selected indicators are listed in Table 1.

In step 2, values of indicators shown in Table 1 have been gathered weekly from 162 companies in Tehran stock exchange in a time duration between April-2015 and March-2017.

In step 3, for each stock data record we calculate the average of variations of each indicator in two sequential periods: from April-2015 till March 2016 as delta 1 ($\Delta 1$), and from April 2016 till March 2017 as delta 2 ($\Delta 2$). Therefor totally we have 324 data records. Then in each stock data record and for each indicator we add a field containing the sign of achieved delta value (e.g. positive or negative).

Table 1. Selected indicators

Financial indicators	Macroeconomics indicators
Size of firm	Inflation Rate
Debt to Equity	Exchange Rate
Return on Asset (RoA)	Money Supply
Return on Equity (RoE)	Industry Growth Rate
Sales Changes	
Operational Benefit	
Price to Earnings ratio (P/E)	
Volume of Average Daily Trading	
Beta Risk Factor	
Earnings Per Share (EPS)	

If the delta value of a given field shows a same sign in all records, that delta value is omitted for 4th step. The reason is clearly because of the same impact of fixed sign value on all of the records. The process of calculating delta values for each stock data record in data set is depicted in Table 2.

Table 2. Calculation of delta values

	Δ (Size)	**Δ (RoA)**	**...**	**Δ (EPS)**
Stock A	**Δ1 (Size)** *= Size(2016) – Size(2015)* *= (positive / negative)*	**Δ1 (RoA)** *= RoA(2016) – RoA(2015)* *= (positive / negative)*	...	**Δ1 (EPS)** *= EPS(2016) – EPS(2015)* *= (positive / negative)*
	Δ2 (Size) *= Size(2017) – Size(2016)* *= (positive / negative)*	**Δ2 (RoA)** *= RoA(2017) – RoA(2016)* *= (positive / negative)*	...	**Δ2 (EPS)** *= EPS(2017) – EPS(2016)* *= (positive / negative)*

When all delta values are calculated and added to data set, in step 4, we apply k-means clustering technique to discover patterns of delta signs of indicators on EPS variations of each stock. We partitioned all 324 data records belonging to 162 companies into 15 clusters according to opinion of domain expert and based on the highest number of total records placed in correct EPS category. Procedure of k-means clustering is summarized in Algorithm 1.

Algorithm 1: *k*-means

Input: *k*: the number of clusters, *ds*: data set containing n data instances
Output: *k* clusters, each centroid is represented by the mean value of intra-cluster instances

```
for (i = 1 ; i <= k ; i ++)
centroid_i = an arbitrary instance from ds;
While (centroids no change)
{
   for (j = 1 ; j <= k ; j ++)
   {
     Assign a data instance to cluster_j so that distance (centroid_j , instance) to be minimized;
     Centroid_j = (Σ intra-cluster_j instances' values) / number of intra-cluster_j instances;
   } // end for
} // end while
return k clusters;
```

Statistically, 159 of 324 records have positive Δ (EPS), and 165 records have negative Δ (EPS). By applying k-means clustering using 15 cluster numbers, 152 instances fall in positive Δ (EPS) category and 172 instances fall in negative Δ (EPS) category. So totally 310 instances placed in correct EPS cluster and 14 instances placed in incorrect EPS cluster. Therefor 95.68% of all instances are placed in correct cluster. Result of applying k-means clustering on data instances is shown in Figure 3.

0	1	2	3	4	5	6	7
(41)	(36)	(27)	(9)	(36)	(25)	(26)	(19)
size_pos	size_pos	size_pos	size_pos	size_pos	size_neg	size_pos	size_pos
debt_pos	debt_pos	debt_pos	debt_pos	debt_neg	debt_pos	debt_pos	debt_pos
roa_neg	roa_neg	roa_neg	roa_neg	roa_pos	roa_neg	roa_pos	roa_pos
roe_neg	roe_neg	roe_neg	roe_neg	roe_pos	roe_neg	roe_pos	roe_pos
sale_pos	sale_pos	sale_neg	sale_neg	sale_pos	sale_neg	sale_pos	sale_pos
benefit_pos	benefit_neg	benefit_neg	benefit_neg	benefit_pos	benefit_neg	benefit_pos	benefit_pos
pe_neg	pe_neg	pe_pos	pe_neg	pe_pos	pe_neg	pe_pos	pe_neg
vol_neg	vol_pos	vol_pos	vol_pos	vol_neg	vol_neg	vol_pos	vol_pos
infl_neg	infl_pos	infl_neg	infl_neg	infl_pos	infl_pos	infl_neg	infl_pos
zero_one	zero_one	neg1_0	two_three	neg1_0	zero_one	zero_one	neg1_0
eps_neg	eps_neg	eps_neg	eps_neg	eps_pos	eps_neg	eps_pos	eps_pos

8	9	10	11	12	13	14
(20)	(18)	(17)	(4)	(17)	(19)	(10)
size_neg	size_pos	size_neg	size_pos	size_pos	size_pos	size_pos
debt_neg	debt_neg	debt_pos	debt_pos	debt_neg	debt_pos	debt_pos
roa_neg	roa_pos	roa_pos	roa_neg	roa_pos	roa_pos	roa_neg
roe_neg	roe_pos	roe_pos	roe_neg	roe_pos	roe_pos	roe_neg
sale_pos	sale_pos	sale_pos	sale_pos	sale_neg	sale_pos	sale_pos
benefit_neg	benefit_pos	benefit_pos	benefit_pos	benefit_pos	benefit_pos	benefit_neg
pe_pos	pe_neg	pe_pos	pe_neg	pe_neg	pe_pos	pe_pos
vol_pos	vol_pos	vol_pos	vol_pos	vol_pos	vol_neg	vol_neg
infl_neg	infl_pos	infl_neg	infl_neg	infl_neg	infl_neg	infl_neg
zero_one	neg1_0	one_two	zero_one	zero_one	zero_one	zero_one
eps_neg	eps_pos	eps_pos	eps_neg	eps_pos	eps_pos	eps_neg

Figure 3. Result of k-means clustering on data set

In step 5, we extract trading rules based on results of step 4 (k-means clustering). Regarding to Figure 3, the trading rules of each cluster can be achieved in *if-then* form. For example, for cluster #0 we have a trading rule as below:

IF ((Size of Company > 0) ***AND*** (Debt to Equity > 0) ***AND*** (RoA < 0) ***AND***
(RoE < 0) ***AND*** (Sale Changes > 0) ***AND*** (Operational Benefit > 0) ***AND***
(P/E < 0) ***AND*** (Volume of Average Daily Trading < 0) ***AND***
(Inflation Rate < 0) ***AND*** (0 < Beta Risk Factor < = 1))
THEN (EPS < 0)

As such, 15 trading rules can be extracted from patterns of indicators relevant to data records placed in 15 cluster, to be used in 6th step, constructing our CPN signaling model in next sub-section.

4.3 Proposed CPN model

In this section we design our proposed CPN signaling model as step 6, and describe how it performs. A conceptual design of the model is represented in Figure 4.

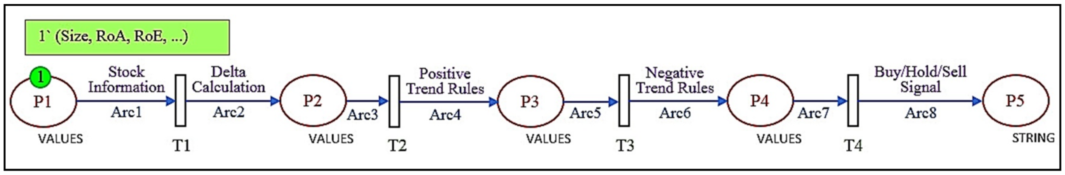


Figure 4. Conceptual design of proposed CPN signaling model

Stock data records are defined in form of tokens in place P1. Each token carries its related indicators' values such as company size, RoA, RoE, ... To do so, a *product* color set is defined as below to cover required properties:

*Color_set VALUE = product REAL * REAL * ... * STRING * INT * REAL;*

Each couple of REAL values are representing an indicator's value in beginning and ending of the time interval respectively. These are used to calculate delta values of indicators after firing transition T1. The STRING type carries the name of the company. The INT type carries stock's historical trend, that is if given stock's trend was positive, negative or neutral in its last history. We explain about this property in following. The final REAL type carries the value of beta risk factor.

As mentioned above, by firing transition T1, function *delta* is executed to calculate delta values of indicators according to descriptions of step 3 in sub-section 4.2 and stock token moves to place P2. Then, trading rules extracted from k-means clustering results (step 5) are defined in model to be applied on calculated delta values. The rules related to positive EPS values are defined as expression of Arc 4, and those related to negative EPS values are defined on Arc 6. The logic sequentially definition of positive and negative trading rules is as follows: when transition T2 fires, if a stock's values satisfy one of the if-then conditions of positive trend, then a positive infinity value will be assigned to the first property of the stock in token, otherwise token does not change. Then the token moves to place P3 and transition T3 fires unconditionally. With firing T3, the value of first property of stock in token will be checked. If it equals positive infinity, the token will pass without change. If it does not equal positive infinity, then the token will be checked if it is matched with any of conditions of negative trend. If it is matched, then a negative infinity value will be assigned to the first property of the stock in token. Otherwise the token does not change. Then the token enters to place P4 and transition T4 fires unconditionally. When T4 fires, value of the first property of token will be checked. If it contains positive infinity value, then it is evaluated as "*positive trend*". If it contains negative infinity value, then it is evaluated as "*negative trend*". Otherwise it is evaluated as "*neutral trend*". Normally, a positive trend should generate a "*BUY*" signal, a negative trend should generate a "*SELL*" signal, and a neutral trend should generate a "*HOLD*" signal. Here we pay attention to stock's last historical trend to generate final signal.

We consider a weight coefficient for historical trend and new trend according to their importance. The historical trend receives coefficient $1/3 \approx (33.\overline{33}\%)$, and new trend receives coefficient $2/3 \approx (66.\overline{66}\%)$. Neutral trend, historical or new, receives no coefficient. Trading signal will be generated based on sum of trends' coefficients. The sign of coefficients of negative trends is considered as (-) and for positive trends is considered as (+). Therefor by firing transition T4, a trading signal will be generated along with a strength coefficient summarized in Table 3.

Table 3. Generating stock trading signal along with strength coefficient

Historical trend	Current trend	Coefficient calculation			Generated signal
		Hist.	Curr.	Sum	
Positive	Positive	33.33 %	66.66 %	99.99 %	BUY (99.99 %)
Neutral	Positive	0 %	66.66 %	66.66 %	BUY (66.66 %)
Negative	Positive	-33.33 %	66.66 %	33.33 %	BUY (33.33 %)
Positive	Neutral	33.33 %	0 %	33.33 %	BUY (33.33 %)
Neutral	Neutral	0 %	0 %	0 %	HOLD
Negative	Neutral	-33.33 %	0 %	-33.33 %	SELL (33.33 %)
Positive	Negative	33.33 %	-66.66 %	-33.33 %	SELL (33.33 %)
Neutral	Negative	0 %	-66.66 %	-66.66 %	SELL (66.66 %)
Negative	Negative	-33.33 %	-66.66 %	-99.99 %	SELL (99.99 %)

4.4 Time complexity

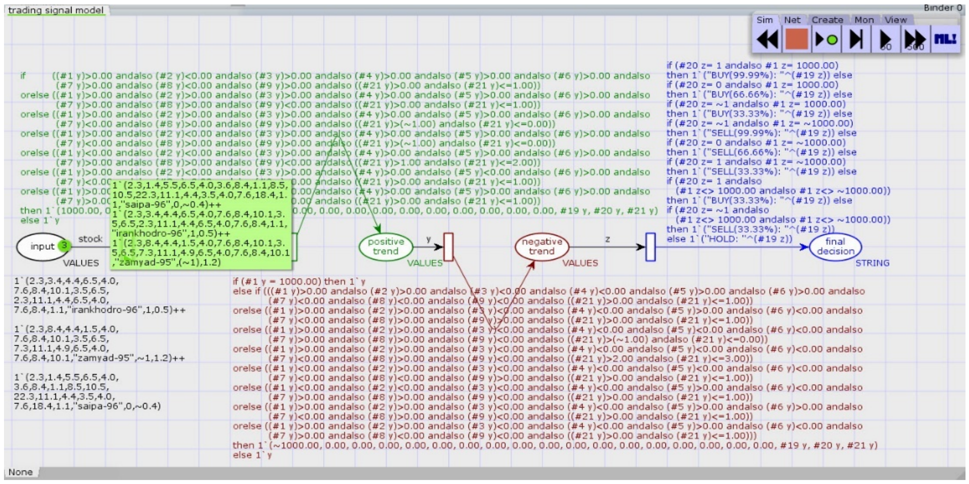
Time complexity of proposed approach is estimated in two sections: k-means clustering algorithm, and simulation with CPN model. Regarding to Algorithm 1, optimizing the intra-cluster variations is the most computationally challenging task. It has been shown that the problem is of NP-hard time complexity class in general Euclidean space even for k = 2, and also in 2-demension Euclidean space for an arbitrary number of k. If the number of k and dimensionality (d) of the space are fixed, for n data instances the problem belongs to $O(n^{dk+1} \log n)$ [42]. Simulation with CPN model sequentially is repeated for m data records in $O(m)$ execution time. So total process is done in $O(n^{dk+1} \log n + m)$.

5. Simulation and Evaluation

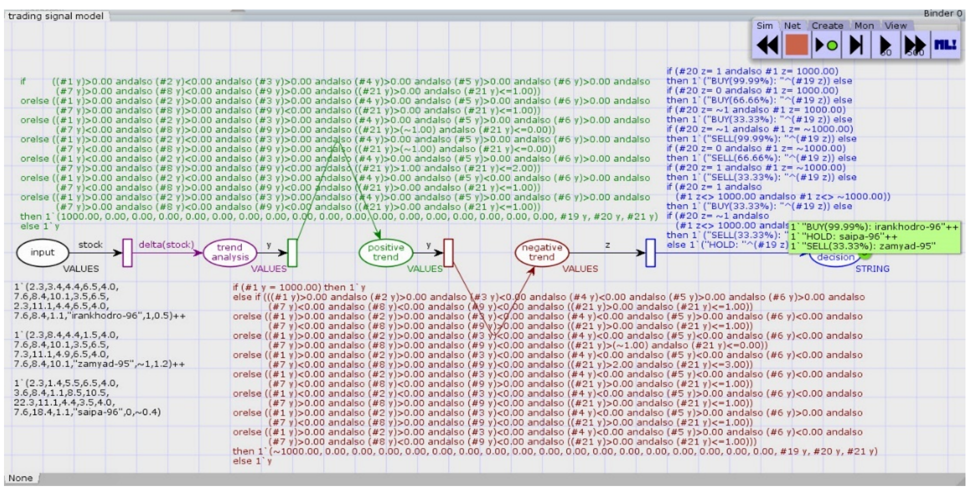
5.1 Simulation environment

The process of implementing k-means clustering technique has been accomplished in Weka environment, and the process of modeling and generating trading signal has been done in CPN Tools, a dedicated tool for modeling and simulation based on CPN.

Hardware configuration consists of Intel Core 2 Duo 2.20 GHz processor on Dell 1545 Inspiron laptop with Windows 10 x64 ultimate operating system. The result of k-means clustering has been shown in Figure 3. A view of our modeled system in CPNTools is shown in Figure 5(a), and the model after simulation and creating trading signals is shown in Figure 5(b). The type *VALUES* definition, which allows carrying values of stocks' indicators across the model along with tokens, and also *delta* function definition are depicted in Figure 6.



a. Trading signal model before simulating



b. Trading signal model after simulating

Figure 5. Stock trading signal prediction model in CPN Tools

```
▼colset VALUES= product
 REAL*REAL*REAL*REAL*REAL*REAL*
 REAL*REAL*REAL*REAL*REAL*REAL*
 REAL*REAL*REAL*REAL*REAL*REAL*
 STRING*INT*REAL;
▼var stock,y,z: VALUES;
▼var status: STRING;
▼fun delta(x:VALUES)=
 let
 val y= ((#2 x)-(#1 x),(#4 x)-(#3 x),
        (#6 x)-(#5 x),(#8 x)-(#7 x),
        (#10 x)-(#9 x),(#12 x)-(#11 x),
        (#14 x)-(#13 x),(#16 x)-(#15 x),
        (#18 x)-(#17 x),0.00,0.00,0.00,
        0.00,0.00,0.00,0.00,0.00,0.00,
        #19 x, #20 x,#21 x)
 in y
 end;
```

Figure 6. Definition of VALUES type and delta function

5.2 Evaluation

We fed our signaling model with 50 test data records, described in section 4.2. The results of simulations show that 46 records have been detected and classified correctly and corresponding appropriate trading signal have been generated. So precision of the model is estimated about 92% which should be multiplied by precision of k-means rules (95.68%) equals 88.0256% ≈ 88%. We have implemented other state-of-the-art approaches, i.e. artificial neural network, decision tree, and linear regression, using our data set. The results indicate the proposed model outperforms other approaches, as is shown in Table 4.

Table 4. Evaluation and comparison

Method	Correctly classified instances	Precision
Artificial Neural Network	41 of 50	82 %
Decision Tree	43 of 50	86 %
Linear Regression	42 of 50	84 %
Proposed CPN Model	**46 of 50**	**88 %**

6. Conclusion

Stock market is one of the main interests for investors to gain more financial benefits and less losses. The decision support approaches that guide investors to make decision about when to buy/sell a stock are generally fall in two categories: fundamental analysis-based approaches that generally predict stock price, and technical analysis-based approaches that try to predict points of price changes and stock trend. Price changes in short-term is highly challenging under frequency of data characteristics. Whereas the turning points commonly have longer periods in short-term, and the high frequency characteristics of data is accordingly reduced. So in this paper we proposed a stock trading signal prediction model using CPN formal method to take benefit of mathematical nature and Markup Language of CPN simultaneously overcome to shortcomings of solution tools such as meta-heuristic algorithms and data mining techniques. We extracted some trading rules from patterns on 14 financial and macroeconomics indicators using k-means clustering technique and we defined them in form of if-then trading rules in proposed model. The results of our simulations in CPN Tools and implementation of other state-of-the-art approaches, i.e. artificial neural network, decision tree, and linear regression, indicated that our model outperformed others with 88% precision rate. As future direction, we planned to enhance precision of trading rules and prediction results by employing a combination of fuzzy and stochastic extensions of petri nets.

References

[1] Chang, P. C., Fan, C. Y., & Liu, C. H. (2008). Integrating a piecewise linear representation method and a neural network model for stock trading points prediction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *39*(1), 80-92. https://doi.org/10.1109/TSMCC.2008.2007255

[2] Chourmouziadis, K., & Chatzoglou, P. D. (2016). An intelligent short term stock trading fuzzy system for assisting investors in portfolio management. *Expert Systems with Applications*, *43*, 298-311. https://doi.org/10.1016/j.eswa.2015.07.063

[3] Kara, Y., Boyacioglu, M. A., & Baykan, Ö. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert systems with Applications*, *38*(5), 5311-5319. https://doi.org/10.1016/j.eswa.2010.10.027

[4] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, *42*(4), 2162-2172. https://doi.org/10.1016/j.eswa.2014.10.031

[5] Chang, P. C., Liao, T. W., Lin, J. J., & Fan, C. Y. (2011). A dynamic threshold decision system for stock trading signal detection. *Applied Soft Computing*, *11*(5), 3998-4010. https://doi.org/10.1016/j.asoc.2011.02.029

[6] Abu-Mostafa, Y. S., & Atiya, A. F. (1996). Introduction to financial forecasting. *Applied Intelligence*, *6*(3), 205-213. https://doi.org/10.1007/BF00126626

[7] Chang, P. C., Fan, C. Y., & Lin, J. L. (2011). Trend discovery in financial time series data using a case based fuzzy decision tree. *Expert Systems with Applications*, *38*(5), 6070-6080. https://doi.org/10.1016/j.eswa.2010.11.006

[8] Khan, B. T., Javed, N., Hanif, A., & Raja, M. A. (2017, October). Evolving technical trading strategies using genetic algorithms: A case about pakistan stock exchange. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 335-344). Springer, Cham. https://doi.org/10.1007/978-3-319-68935-7_37

[9] Mabu, S., Hirasawa, K., Obayashi, M., & Kuremoto, T. (2013). Enhanced decision making mechanism of rule-based genetic network programming for creating stock trading signals. *Expert Systems with Applications*, *40*(16), 6311-6320. https://doi.org/10.1016/j.eswa.2013.05.037

[10] Lu, Z., Long, W., & Guo, Y. (2018, June). Extreme Market Prediction for Trading Signal with Deep Recurrent Neural Network. In *International Conference on Computational Science* (pp. 410-418). Springer, Cham. https://doi.org/10.1007/978-3-319-93701-4_31

[11] Sezer, O. B., Ozbayoglu, A. M., & Dogdu, E. (2017, April). An artificial neural network-based stock trading system using technical analysis and big data framework. In *Proceedings of the SouthEast Conference* (pp. 223-226). ACM. https://doi.org/10.1145/3077286.3077294

[12] Yong, B. X., Rahim, M. R. A., & Abdullah, A. S. (2017, August). A stock market trading system using deep neural network. In *Asian Simulation Conference* (pp. 356-364). Springer, Singapore. https://doi.org/10.1007/978-981-10-6463-0_31

[13] Chen, Y., & Hao, Y. (2018). Integrating principle component analysis and weighted support vector machine for stock trading signals prediction. *Neurocomputing*, *321*, 381-402. https://doi.org/10.1016/j.neucom.2018.08.077

[14] Ta, V. D., Liu, C. M., & Addis, D. (2018, December). Prediction and Portfolio Optimization in Quantitative Trading Using Machine Learning Techniques. In *Proceedings of the Ninth International Symposium on Information and Communication Technology* (pp. 98-105). ACM. https://doi.org/10.1145/3287921.3287963

[15] Huang, Q., Kong, Z., Li, Y., Yang, J., & Li, X. (2018). Discovery of trading points based on Bayesian modeling of trading rules. *World Wide Web*, *21*(6), 1473-1490. https://doi.org/10.1007/s11280-018-0534-9

[16] Naik, N., & Mohan, B. R. (2019, February). Optimal Feature Selection of Technical Indicator and Stock Prediction Using Machine Learning Technique. In *International Conference on Emerging Technologies in Computer Engineering* (pp. 261-268). Springer, Singapore. https://doi.org/10.1007/978-981-13-8300-7_22

[17] Chang, P. C., Liao, T. W., Lin, J. J., & Fan, C. Y. (2011). A dynamic threshold decision system for stock trading signal detection. *Applied Soft Computing*, *11*(5), 3998-4010. https://doi.org/10.1016/j.asoc.2011.02.029

[18] Jothimani, D., & Yadav, S. S. (2019). Stock trading decisions using ensemble-based forecasting models: a study of the Indian stock market. *Journal of Banking and Financial Technology*, 1-17. https://doi.org/10.1007/s42786-019-00009-7

[19] Marković, I., Stojanović, M., Stanković, J., & Stanković, M. (2017). Stock market trend prediction using AHP and weighted kernel LS-SVM. *Soft Computing*, *21*(18), 5387-5398. https://doi.org/10.1007/s00500-016-2123-0

[20] Sun, J., Huang, Q., & Li, X. (2019). Determination of Temporal Stock Investment Styles via Biclustering Trading Patterns. *Cognitive Computation*, 1-10. https://doi.org/10.1007/s12559-019-9626-9

[21] Wen, M., Li, P., Zhang, L., & Chen, Y. (2019). Stock Market Trend Prediction Using High-Order Information of Time Series. *IEEE Access*, *7*, 28299-28308. https://doi.org/10.1109/ACCESS.2019.2901842

[22] Lee, K. H., & Jo, G. S. (1999). Expert system for predicting stock market timing using a candlestick chart. *Expert systems with applications*, *16*(4), 357-364. https://doi.org/10.1016/S0957-4174(99)00011-1

[23] Ghorbani, A., & Farzai, S. (2018). Fraud detection in automobile insurance using a data mining based approach. *International Journal of Mechatronics, Elektrical and Computer Technology (IJMEC)*, *8*(27), 3764-3771.

[24] Farzai, S., Ghasemi, D., & Marzuni, S. S. M. Offenders Clustering Using FCM & K-Means.

[25] Bahrami, B., & Hosseini Shirvani, M. (2015). Prediction and Diagnosis of Heart Disease by Data Mining Techniques. *Journal of Multidisciplinary Engineering Science and Technology (JMEST)*, *2*(2), 164-168.

[26] Festa, P. (2014, July). A brief introduction to exact, approximation, and heuristic algorithms for solving hard combinatorial optimization problems. In *2014 16th International Conference on Transparent Optical Networks (ICTON)* (pp. 1-20). IEEE. https://doi.org/10.1109/ICTON.2014.6876285.

[27] Razavi, F., Zabihi, F., Hosseini Shirvani, M. (2016). Multi-layer Perceptron Neural Network Training Based on Improved of Stud GA. Journal of Advances in Computer Research, 7(3), 1-14.

[28] Hosseini, M. (2018). A new Shuffled Genetic-based Task Scheduling Algorithm in Heterogeneous Distributed Systems. *Journal of Advances in Computer Research*, 9(4), 19-36.

[29] Jensen, K. (2013). *Coloured Petri nets: basic concepts, analysis methods and practical use* (Vol. 1). Springer Science & Business Media. https://doi.org/10.1007/978-3-662-03241-1

[30] Motameni, H., MOHAMMAD, V., Shirazi, B., & Barzegar, B. (2013). Modeling and Evaluation of Web Services in Mobile Networks using Stochastic Colored Petri Nets.

[31] Barzegar, B., Motameni, H., & Bozorgi, H. (2012). Solving flexible job-shop scheduling problem using gravitational search algorithm and colored Petri net. *Journal of applied mathematics*, *2012.*

[32] Motameni, H. & Farzai, S. (2017). Petri nets, extensions and applications. Olum-Rayaneh publication, Babol. Iran. (In Persion).

[33] Mohsenian, N., Motameni, H., Jeddi Saravi, S. (2015). Evaluating Adaptability of SET Protocols Using Colored Petri Nets. *Journal of Advances in Computer Research*, 6(3), 39-49.

[34] Rogge-Solti, A., Vana, L., & Mendling, J. (2015, December). Time series petri net models-enrichment and prediction. CEUR Workshop Proceedings. http://ceur-ws.org/Vol-1527/paper8.pdf

[35] Shen, R. K., Yang, C. Y., Shen, V. R., Li, W. C., & Chen, T. S. (2018). A stock market prediction system based on high-level fuzzy Petri nets. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *26*(05), 771-808. https://www.worldscientific.com/doi/abs/10.1142/S0218488518500356

[36] Fat, R., Mic, L., Kilyen, A. O., Santa, M. M., & Letia, T. S. (2016, May). Model and method for the stock market forecast. In *2016 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)* (pp. 1-5). IEEE. https://doi.org/10.1109/AQTR.2016.7501345

[37] Shih, P. Y., Shih, D. H., & Liu, A. C. (2016, July). A colored Petri Net for stock trading signal detection. In *2016 IEEE Congress on Evolutionary Computation (CEC)* (pp. 4551-4554). IEEE. https://doi.org/10.1109/CEC.2016.7744369

[38] Shih, P. Y., Shih, D. H., Hung, S. Y., & Shih, M. H. (2016, July). A rough Petri nets model for stock trading signal detection. In *2016 IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)* (pp. 135-139). IEEE. https://doi.org/10.1109/ICCCBDA.2016.7529547

[39] Li, Q., Chen, Y., Jiang, L. L., Li, P., & Chen, H. (2016). A tensor-based information framework for predicting the stock market. *ACM Transactions on Information Systems (TOIS)*, *34*(2), 11. https://doi.org/10.1145/2838731

[40] Jun, D., Ahn, C., Kim, J., & Kim, G. (2019). Signal analysis of global financial crises using Fourier series. *Physica A: Statistical Mechanics and its Applications*, *526*, 121015. https://doi.org/10.1016/j.physa.2019.04.251

[41] Naranjo, R., Arroyo, J., & Santos, M. (2018). Fuzzy modeling of stock trading with fuzzy candlesticks. *Expert Systems with Applications*, *93*, 15-27. https://doi.org/10.1016/j.eswa.2017.10.002

[42] Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier. ISBN: 978-0-12-381479-1.

[43] Murphy, John J. (1999). Technical analysis of the financial markets: a comprehensive guide to trading methods and applications (2nd ed.). New York [u.a.]: New York Inst. of Finance. ISBN: 0735200661.