УДК 621.316.13

INTELLIGENT TIME SERIES DATA ANALYSIS OF CRYPTOCURRENCY MARKET DYNAMICS BASED ON OHLCV DATASET AND MOMENTUM TECHNICAL INDICATORS

I. S. Uzun[1], M. V. Lobachev[1]
[1] *Odesa Polytechnic National University*

Abstract. *This article presents an analysis of the historical dynamics of the cryptocurrency market based on time series data using the OHLCV dataset. The study presents the results of calculations of the main cryptocurrency market momentum technical indicators. Using intelligent computational methods, the paper assesses patterns and trends in the data of major cryptocurrencies. The study emphasizes the importance of technical analysis in understanding the volatile landscape of digital currencies.*

Key words: *time-series, data analysis, cryptocurrency market, momentum indicators, technical analysis indicators, OHLCV.*

Introduction

In the domain of computer science, the study of cryptocurrencies represents a dynamic intersection of cryptography, distributed systems, and data analytics. These digital assets, secured through cryptographic methods, operate on decentralized networks facilitated by blockchain technology, offering a novel approach to secure and transparent transactions. This setting presents a unique challenge for data analysts and computer scientists: deciphering the complex behavior of cryptocurrency markets through intelligent data analysis techniques.

Cryptocurrency market dynamics are characterized by rapid fluctuations and intricate patterns, making them an ideal subject for advanced computational analysis. The focus here shifts from traditional economic analysis to the application of time-series data analysis, a method critical in processing sequential data points collected over time intervals. This approach is fundamental in computer science for extracting meaningful patterns from data that evolves over time, such as cryptocurrency price movements and trading volumes.

Intelligent time-series analysis employs machine learning and statistical modeling to forecast future trends based on historical data. This technique is paramount in navigating the volatile cryptocurrency market, as it enables the identification of patterns and potential market movements with a high degree of precision. By applying algorithms that learn from data, researchers can predict changes in market conditions, offering valuable insights for automated trading systems and financial decision-making.

Technical momentum indicators are specialized algorithms used within this analytical framework to gauge the strength and direction of market trends. These indicators analyze past price and volume data to predict future market movements. In the realm of computer science, creating these indicators involves sophisticated algorithmic design and the application of statistical analysis techniques. They are crucial for quantifying market sentiment and momentum, thereby aiding in the development of predictive models and decision support systems for trading in cryptocurrency markets.

This article addresses the complex challenge of analyzing cryptocurrency market dynamics through a computer science lens, focusing on the use of OHLCV (Open, High, Low, Close, Volume) datasets and technical momentum indicators. It contributes to the existing body of research by applying intelligent time-series analysis methods to uncover and interpret market behaviors. The study aims to provide a comprehensive understanding of these dynamics, offering insights that can enhance algorithmic trading strategies and inform future research directions in the continuously evolving field of cryptocurrency analysis.

1. Overview of research tools

In the exploration of cryptocurrency market dynamics, the choice of research tools significantly influences the depth and efficiency of analysis. Python, a versatile and powerful programming language, stands at the forefront of this study due to its wide acceptance in the data science and machine learning communities. Its simplicity, readability, and extensive ecosystem of libraries make it an ideal choice for conducting sophisticated data analysis and algorithmic processing in the realm of financial markets.

Among the myriad of Python libraries available, several are particularly instrumental for this research. Firstly, pandas, a library offering high-performance, easy-to-use data structures, and data analysis tools, is crucial for handling and manipulating the OHLCV datasets. It enables the efficient processing of time-series data, facilitating the exploration of intricate market dynamics through a computational lens.

NumPy, another essential library, provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these data structures. Its capabilities are vital for performing numerical computations required in the calculation of technical momentum indicators and other statistical analyses.

For the specific task of calculating and analyzing technical momentum indicators, the TA-Lib library is of paramount importance. TA-Lib, short for Technical Analysis Library, is a specialized Python library that provides tools for developing trading and market analysis algorithms, including functions to compute over 150 technical indicators such as moving averages, RSI (Relative Strength Index), MACD (Moving Average Convergence Divergence), and many others discussed in this study.

Moreover, matplotlib and seaborn are indispensable for visualizing the data and the results of the analysis. These libraries offer a wide range of plotting functions, making it possible to create clear, informative graphs that illustrate the cryptocurrency market trends and the outcomes of technical indicator analyses.

2. Overview of the input datasets

The foundation of our analysis in the cryptocurrency market dynamics is based on the examination of OHLCV datasets. These datasets, representing Open, High, Low, Close, and Volume data, provide a granular view of market behavior over specified intervals. The datasets are structured as follows:

– «Date» column marks the specific day for the dataset entry, establishing a chronological order essential for time-series analysis;

– «Open» column stands for the opening price of the cryptocurrency at the start of the trading day;

– «High» column stands for the highest price point reached by the cryptocurrency during the trading day;

– «Low» column stands for the price point of the cryptocurrency within the same period;

– «Close» column stands for the closing price of the cryptocurrency at the end of the trading day, indicating the final market sentiment;

– «Volume» column reflects the total trading volume of the cryptocurrency for the day, offering insights into market activity.

For this study, we focus on daily interval datasets for three major cryptocurrencies:

– ETH and BTC: From April 2018 to 2024, capturing a broad spectrum of market behaviors and trends;

– LTC: From June 2018 to 2024, providing a comprehensive view of its market dynamics;

All datasets utilized in this study are open source, sourced from the CryptoDataDownload platform. This platform provides a valuable repository of historical cryptocurrency data, enabling researchers and analysts to conduct extensive market analysis.

This combination of detailed, interval-specific datasets and sophisticated analytical tools lays the groundwork for our comprehensive study of cryptocurrency market dynamics, aiming to uncover patterns, trends, and insights that could inform trading strategies and contribute to the broader field of financial technology research.

Figures 1-3 provide a visual overview of the datasets.

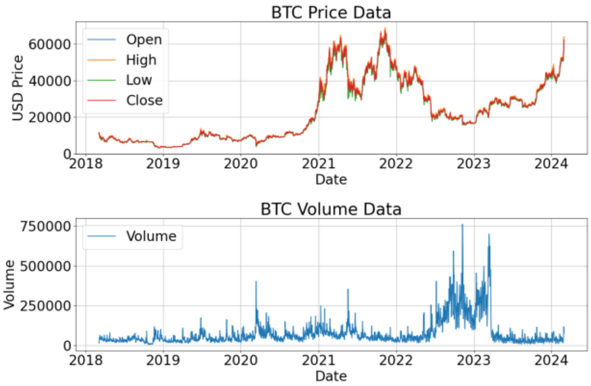

Fig. 1. Bitcoin prices and volume data

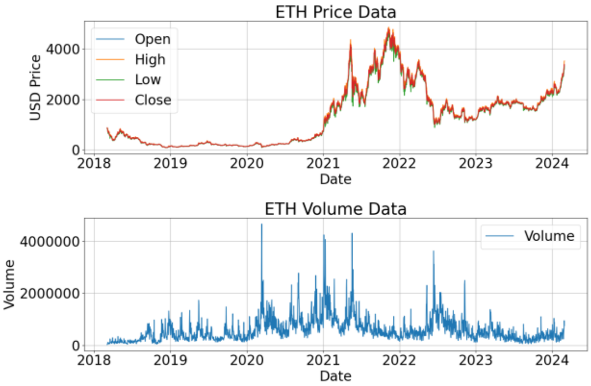

Fig. 2. Ethereum prices and volume data

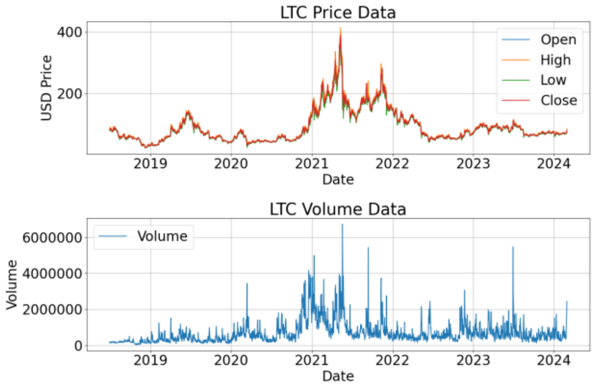

Fig. 3. Litecoin prices and volume data

3. Momentum technical indicators calculation

In the core section of our study, we delve into the intricate process of computing momentum technical indicators, a critical component in the analysis of cryptocurrency market dynamics. These indicators are pivotal for understanding market trends, gauging volatility, and identifying potential entry and exit points in trading strategies. Each indicator employs mathematical computations based on the OHLCV dataset to provide insights into the market's momentum and direction.

Our analysis will cover a comprehensive array of momentum technical indicators, which are elaborated upon below. It's imperative to note that our approach is deeply rooted in computer science methodologies, particularly in the application of data analysis, signal processing, and algorithmic design. The indicators to be calculated and discussed include:

– ADX (Average Directional Movement Index);
– ADXR (Average Directional Movement Index Rating);
– APO (Absolute Price Oscillator);
– AROON Oscillator;
– BOP (Balance Of Power);
– CCI (Commodity Channel Index);
– CMO (Chande Momentum Oscillator);
– MFI (Money Flow Index);
– MOM (Momentum);
– PPO (Percentage Price Oscillator);
– ROC (Rate of Change);
– RSI (Relative Strength Index);
– TRIX (Triple Exponential Average).

Through meticulous computation and analysis of these indicators, we aim to uncover underlying patterns and trends within the cryptocurrency markets. This exploration not only furthers our understanding of market dynamics but also showcases the application of advanced computational techniques in the realm of financial data analysis.

3.1. ADX (Average Directional Movement Index)

The Average Directional Movement Index (ADX) is a technical momentum indicator that measures the strength of a trend regardless of its direction. Developed by J. Welles Wilder, it is widely used to identify whether a market is trending or ranging, thus helping traders to decide if they should employ trend-following strategies or not. The ADX is a component of the Directional Movement System, which also includes the +DI (Positive Directional Indicator) and -DI (Negative Directional Indicator) [1].

The ADX is calculated as a moving average of the Directional Movement Index (DX), which itself is derived from the +DI and -DI values. The formula for ADX is:

$$\text{ADX} = \frac{\text{Smoothed Moving Average of DX}}{\text{Period}} \tag{1}$$

Where the DX is calculated as:

$$\text{DX} = \frac{|+DI - -DI|}{|+DI + -DI|} \times 100 \tag{2}$$

And +DI and -DI are calculated based on the differences between consecutive highs and lows.

The ADX formula involves several steps:

– +DI and -DI calculation: These indicators represent the positive and negative directional movements respectively, calculated based on the differences between the current high (or low) and the previous high (or low);

– DX calculation: The DX measures the directional movement by comparing the difference between +DI and -DI to their sum, normalized to a scale of 0 to 100;

– ADX calculation: The ADX is the smoothed moving average of the DX over a specified period, typically 14 days.

In a computer science context, implementing the ADX involves data manipulation and calculation of moving averages, which can be efficiently performed using Python's data analysis libraries, such as Pandas. The computational aspect focuses on iterating over the dataset, calculating the differences between consecutive highs and lows for the +DI and -DI, followed by the DX, and finally applying a smoothing function to obtain the ADX. This process highlights the application of algorithmic thinking and numerical analysis within the realm of financial data analysis.

Figure 4 presents the results of ADX calculations for BTC, ETH and LTC.

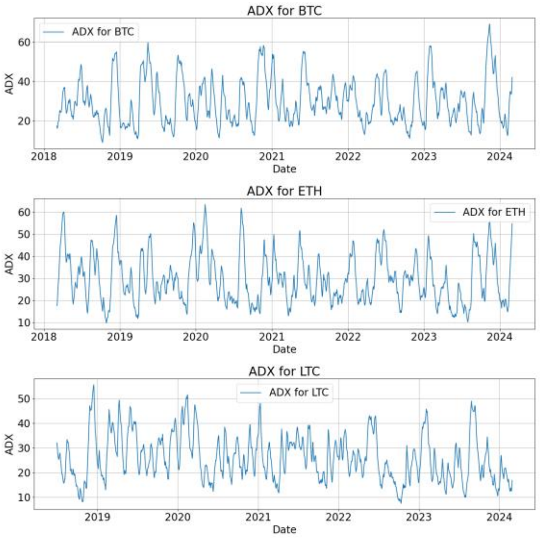

Fig. 4. ADX calculation results

3.2. ADXR (Average Directional Movement Index Rating)

The Average Directional Movement Index Rating (ADXR) is a technical indicator derived from the Average Directional Movement Index (ADX). It aims to measure the strength of a trend over time, providing a rating that helps investors to assess the robustness of a trend. The ADXR is calculated by taking the average of the current ADX value and the ADX value from a selected period in the past, offering a smoothed measure of the trend strength [1].

The ADXR is calculated as the average of the current ADX value and the ADX value «n» periods ago:

$$ADXR = \frac{ADX_{current} + ADX_{n\ periods\ ago}}{2} \qquad (3)$$

The formula for ADXR emphasizes the importance of historical trend strength in assessing the current market condition. By averaging the current ADX with a past ADX value, the ADXR smoothens out the volatility in the ADX itself, providing a more stable indicator of trend strength over time. The choice of «n» periods ago for comparison is typically the same as the period used for calculating the ADX, often 14 days.

Implementing the ADXR in a computer science context involves accessing historical data points to calculate the ADX values at two different times (current and «n» periods ago) and then computing their average. This process highlights the use of time-series data manipulation and underscores the significance of historical data analysis in financial computing. Utilizing Python for this calculation, one would leverage data analysis libraries such as Pandas to handle the dataset and perform the necessary calculations efficiently, showcasing the application of programming skills in financial data analysis.

Figure 5 presents the results of ADXR calculations for BTC, ETH and LTC.

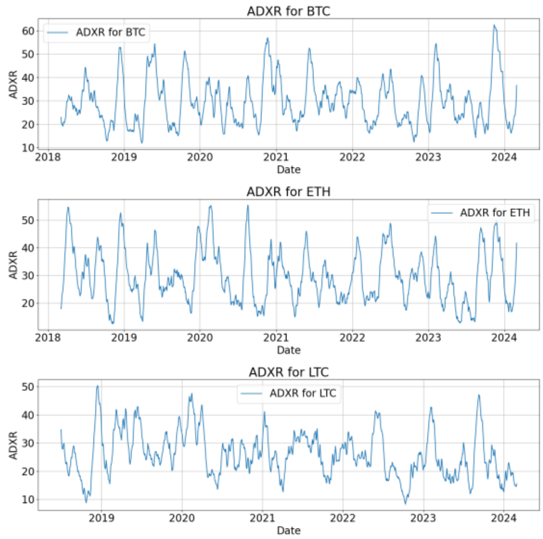

Fig. 5. ADXR calculation results

3.3. APO (Absolute Price Oscillator)

The Absolute Price Oscillator (APO) is a momentum indicator that measures the difference between two moving averages of a security's price. Essentially, it is a specific case of the Moving Average Convergence Divergence (MACD) indicator but focuses solely on the absolute difference between moving averages, ignoring the signal line used in MACD. The APO helps in identifying momentum shifts by signaling when a short-term average moves above or below a long-term average, indicating potential buying or selling opportunities.

The formula for calculating the APO is given by:

$$APO = EMA_{fast} - EMA_{slow} \qquad (4)$$

Where EMA (fast) is the Exponential Moving Average over a shorter period (fastperiod) and EMA (slow) is the Exponential Moving Average over a longer period (slowperiod).

From a computational perspective, implementing the APO in Python involves the use of libraries such as Pandas for handling time series data efficiently. The calculation process includes: computing the EMA for both the fast and slow periods using the respective time frames; subtracting the slow period EMA from the fast period EMA to obtain the APO value.

This process showcases the application of exponential smoothing techniques and highlights the importance of algorithmic efficiency in handling large datasets for financial analysis. The APO, with its straightforward calculation method, serves as an excellent example of applying data processing and numerical analysis techniques in the field of cryptocurrency market analysis.

Figure 6 presents the results of APO calculations for BTC, ETH and LTC.

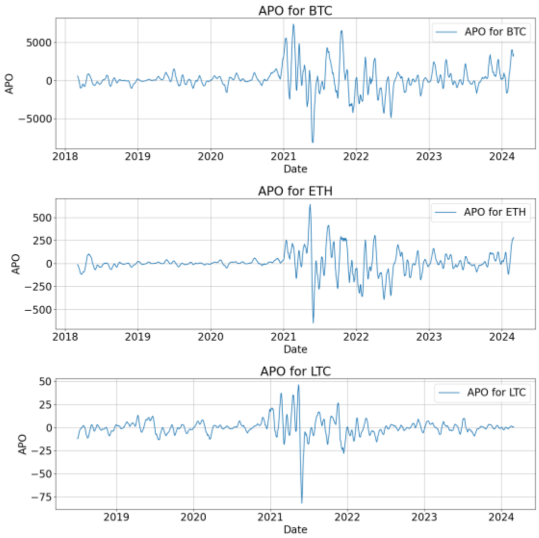

Fig. 6. APO calculation results

3.4. AROON Oscillator

The Aroon Oscillator formula is:

$$\text{Aroon Oscillator} = \text{AroonUp} - \text{AroonDown} \quad (5)$$

Where:

$$\text{AroonUp} = \left(\frac{25-\text{ Periods since 25-period high}}{25}\right) \times 100 \quad (6)$$

$$\text{AroonDown} = \left(\frac{25\text{-Periods since 25-period low}}{25}\right) \times 100 \quad (7)$$

The Aroon Oscillator is derived from two main components, Aroon-Up and Aroon-Down. Aroon-Up measures the time elapsed since the highest price was reached within a given window (typically 25 periods), while Aroon-Down measures the time since the lowest price was recorded. Both are expressed as a percentage of the total period. The oscillator value itself represents the difference between these two measurements, providing a single metric that indicates the direction and strength of a trend. Positive values indicate a bullish trend, while negative values suggest a bearish trend.

Implementing the Aroon Oscillator in a computer science context involves iterating over the dataset to identify the periods since a 25-period high or low, then calculating the Aroon-Up and Aroon-Down values before finally determining the oscillator value. This task can be efficiently accomplished using Python, particularly with libraries such as NumPy for numerical operations and Pandas for data manipulation. The process highlights the integration of computational techniques in analyzing financial time series data, focusing on algorithm development and the application of statistical methods.

Figure 7 presents the results of AROON calculations for BTC, ETH and LTC.

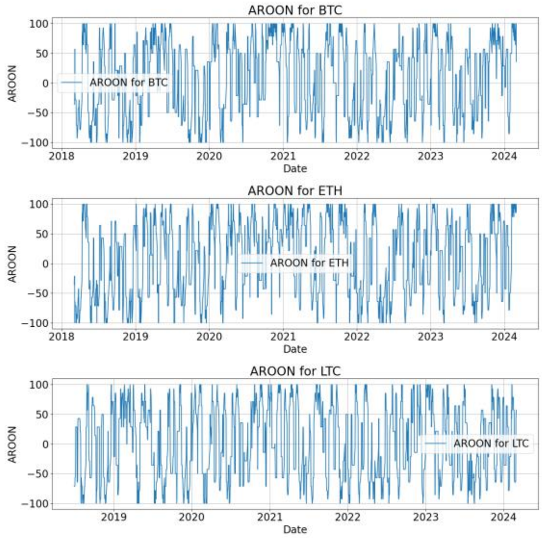

Fig. 7. AROON calculation results

3.5. BOP (Balance Of Power)

The Balance Of Power (BOP) is a momentum indicator that measures the strength of buying against selling pressure in a market. It does so by comparing the extent to which buyers are willing to push prices up versus the extent to which sellers are forcing them down. The BOP oscillates around a zero line, where values above zero indicate buying (bullish) dominance, and values below zero suggest selling (bearish) pressure. This indicator is particularly useful for confirming trend strength and potential reversal points by observing divergences between price movements and BOP values.

The BOP is calculated using the following formula:

$$\text{BOP} = \frac{\text{Close}-\text{Open}}{High-Low} \quad (8)$$

The formula for BOP takes the difference between the closing and opening prices of a security (numerator) and divides it by the range between the highest and lowest prices during the same period (denominator).

A value of BOP near 1 indicates strong buying pressure (as the close is near the high), while a value near -1 indicates strong selling pressure (as the close is near the low).

In terms of computational implementation, calculating the BOP for a dataset requires accessing the open, high, low, and close prices for each period, which can be efficiently done using Python's Pandas library. This involves element-wise operations on the dataset's columns, showcasing the application of vectorized operations for financial data analysis. The BOP's simplicity in calculation, yet profound ability to reveal market sentiment, makes it a valuable tool in the arsenal of technical indicators for analyzing cryptocurrency markets from a computer science perspective.

Figure 8 presents the results of BOP calculations for BTC, ETH and LTC.

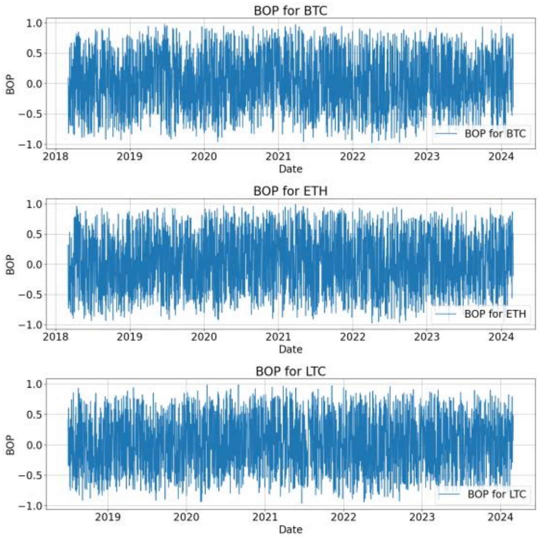

Fig. 8. BOP calculation results

3.6. CCI (Commodity Channel Index)

The Commodity Channel Index (CCI) is a versatile momentum indicator that can be used to identify new trends or warn of extreme conditions. Donald Lambert introduced the CCI to measure the variation of a security's price from its statistical mean. High values show that prices are unusually high compared to the average price, and low values indicate that prices are unusually low. This information is used to assess overbought or oversold levels in the price of a commodity, making the CCI particularly useful for identifying cyclical trends in cryptocurrencies [2].

The CCI is calculated using the following formula:

$$CCI = \frac{1}{0.015} \times \left(\frac{\text{Typical Price - SMA of Typical Price}}{\text{Mean Deviation}} \right) \quad (9)$$

Where Typical Price for each period is:

$$\text{Typical Price} = \frac{\text{High+Low+Close}}{3} \quad (10)$$

SMA of Typical Price is the Simple Moving Average of the Typical Price over «n» periods and Mean Deviation is the average of the absolute deviations of each period's typical price from the SMA of the Typical Price.

The CCI compares the current typical price of a cryptocurrency to the average of typical prices over a specific time frame (usually 20 periods). The result is then normalized by the mean deviation. The constant 0.015 is used to ensure that approximately 70% to 80% of CCI values would fall between -100 and +100 under normal conditions. This normalization process helps to standardize the indicator, making it easier to compare different securities or markets.

Implementing the CCI in a computer science context involves applying statistical functions to the time-series data of cryptocurrencies, which can be adeptly handled using Python libraries like Pandas and NumPy.

Figure 9 presents the results of CCI calculations.

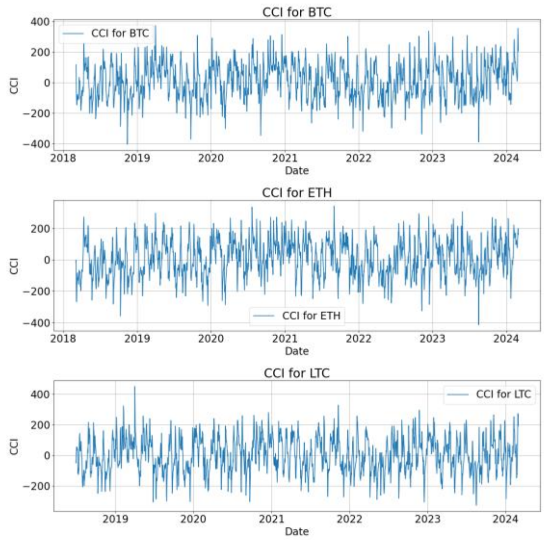

Fig. 9. CCI calculation results

3.7. CMO (Chande Momentum Oscillator)

The Chande Momentum Oscillator (CMO) is designed to capture the momentum of a security's price movement within a bounded range, typically between -100 and +100. Developed by Tushar Chande, the CMO aims to identify overbought and oversold conditions by measuring the strength and direction of trends. Unlike other momentum indicators, the CMO compares the sum of up days to the sum of down days over a given period, providing a more direct view of market momentum.

The CMO is calculated using the following formula:

$$CMO = \left(\frac{\text{Sum of Upward Price Movements-Sum of Downward Price Movements}}{\text{Sum of Upward Price Movements+Sum of Downward Price Movements}} \right) \times 100 \quad (11)$$

To calculate the CMO, one first needs to determine the price movement (change) from one period to the next. An upward price movement is the difference in price if the current period's price is higher than the previous period's; conversely, a downward price movement is the difference if the current peri od's price is lower. The sums of these upward and downward movements are then calculated over a specified period (typically 14 days). The CMO formula subtracts the sum of downward movements from the sum of upward movements, divides this by the total movement (sum of both upward and downward movements), and then multiplies by 100 to scale the result.

Implementing the CMO in a computational context, such as with Python, involves iterating over the cryptocurrency price data to calculate daily price movements, aggregating these movements into sums of upward and downward changes, and then applying the formula to derive the oscillator values.

Figure 10 presents the results of CMO calculations for BTC, ETH and LTC.

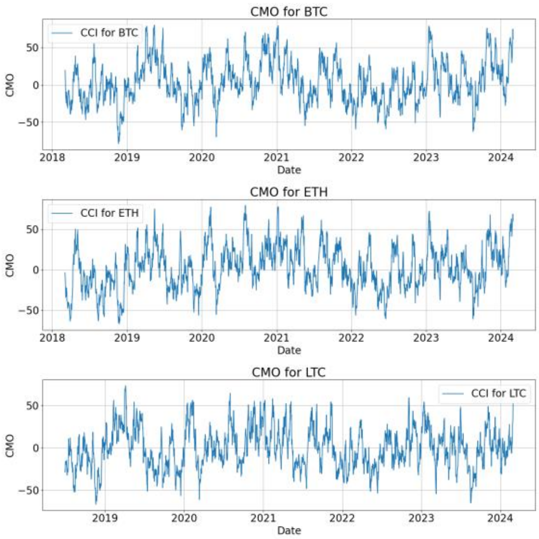

Fig. 10. CMO calculation results

3.8. MFI (Money Flow Index):

The Money Flow Index (MFI) is a volume-weighted momentum indicator that measures the flow of money into and out of a security over a specified period of time. It is similar to the Relative Strength Index (RSI) but incorporates volume, making it a valuable tool for identifying overbought or oversold conditions through the analysis of both price and volume data. The MFI can help indicate the strength of buying or selling pressure and potentially signal price reversals by comparing inflows and outflows of money.

The MFI is calculated as follows:

$$\text{MFI} = 100 - \frac{100}{1 + \text{Money Flow Ratio}} \quad (12)$$

Where the Money Flow Ratio is calculated by dividing the Positive Money Flow by the Negative Money Flow. Positive Money Flow is the sum of the Typical Prices for the days the price went up (multiplied by volume for those days), and Negative Money Flow is the sum of the Typical Prices for the days the price went down (multiplied by volume for those days). The Typical Price for each day is the average of high, low, and close prices and calculated by formula 10.

The MFI integrates price and volume information to gauge buying and selling pressure. It starts with the calculation of the Typical Price for each period, which is then multiplied by the volume to determine the money flow for that period. These money flows are classified as positive or negative based on whether the current period's typical price is higher or lower than the previous period's, respectively. The Money Flow Ratio is then calculated by dividing the sum of positive money flows by the sum of negative money flows over a specified period (usually 14 days). Finally, the MFI is derived by applying the ratio to the formula above, producing a value between 0 and 100 that can be used to identify overbought or oversold conditions.

Figure 11 presents the results of MFI calculations for BTC, ETH and LTC.

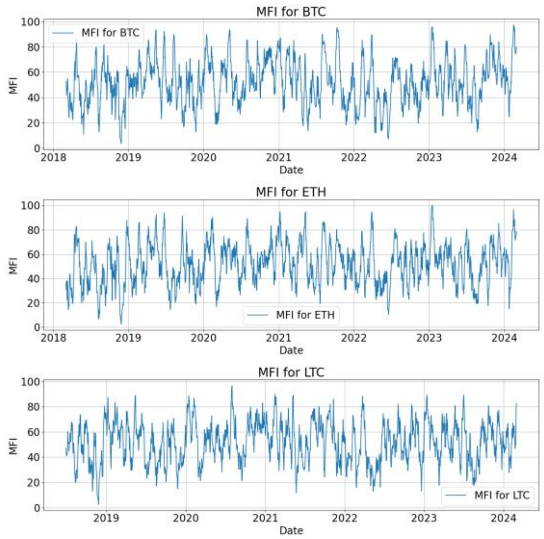

Fig. 11. MFI calculation results

3.9. MOM (Momentum)

The Momentum (MOM) indicator measures the rate of change or speed of price movement of a security. It compares the current price with the price from a specified number of periods ago. The indicator is used to identify the strength of a price movement; it can signal potential reversal points by indicating when a security is overbought or oversold. A positive momentum value indicates that prices are increasing, while a negative value suggests prices are decreasing. The MOM indicator is particularly useful in trending markets to identify when the trend strength is either increasing or waning [4].

The formula for calculating the Momentum indicator is:

$$MOM = Close_{current} - Close_{n\ periods\ ago} \quad (13)$$

This simple computation yields a value that represents the absolute change in the asset's price over the specified period. The period length can be adjusted based on the analysis needs, with shorter periods potentially offering insights into short-term momentum and longer periods providing a view of longer-term momentum trends.

From a computational perspective, calculating the Momentum indicator within a Python environment showcases the power of data manipulation libraries such as Pandas. By utilizing Pandas, one can easily access historical price data, apply the formula across the dataset using rolling or shift operations, and generate a series of momentum values. This process involves iterating over the dataset with a focus on efficient data access and manipulation, underscoring the application of computer science techniques in financial data analysis. The MOM indicator, with its emphasis on rate of change, serves as a fundamental tool for analyzing market dynamics and trend strength in the volatile cryptocurrency markets.

Figure 12 presents the results of MOM calculations for BTC, ETH and LTC.

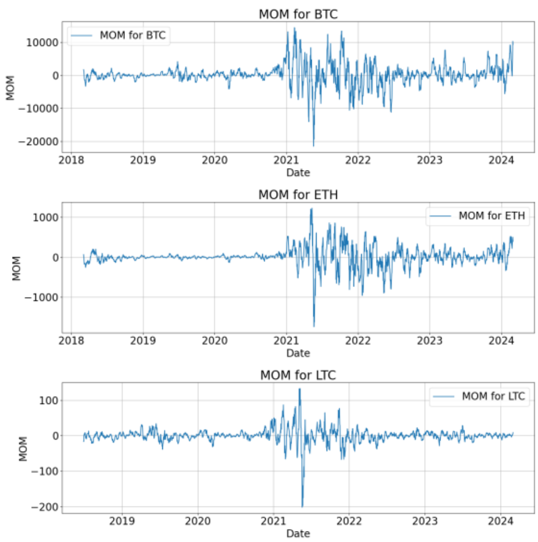

Fig. 12. MOM calculation results

3.10. PPO (Percentage Price Oscillator)

The Percentage Price Oscillator (PPO) is a momentum oscillator that measures the relative difference between two moving averages as a percentage of the slower moving average. Similar to the MACD (Moving Average Convergence Divergence), the PPO expresses this difference in percentage terms, providing a scale-free measure of the relationship between two moving averages. This allows for easier comparison between different securities regardless of their price levels. The PPO is useful for identifying potential trend reversals through the convergence and divergence of its moving averages, as well as through the crossing of its signal line.

The formula for calculating the PPO is:

$$PPO = \left(\frac{EMA_{fast} - EMA_{slow}}{EMA_{slow}}\right) \times 100 \quad (14)$$

The PPO calculation involves subtracting the fast EMA from the slow EMA, dividing the result by the slow EMA, and then multiplying by 100 to convert the result into a percentage. This process normalizes the difference between the two EMAs, making the PPO a relative measure that is independent of the security's price level. The resulting value indicates the percentage by which the fast EMA is above or below the slow EMA, offering insights into the momentum and direction of the price trend.

Implementing the PPO in a computer science context, particularly using Python, leverages the capabilities of libraries such as Pandas for efficient data handling and NumPy for numerical calculations. The process involves computing the EMAs for the specified periods, then applying the PPO formula across the dataset. The PPO's emphasis on percentage differences makes it a versatile tool for comparing momentum across different cryptocurrencies, facilitating a deeper understanding of market dynamics.

Figure 13 presents the results of PPO calculations for BTC, ETH and LTC.

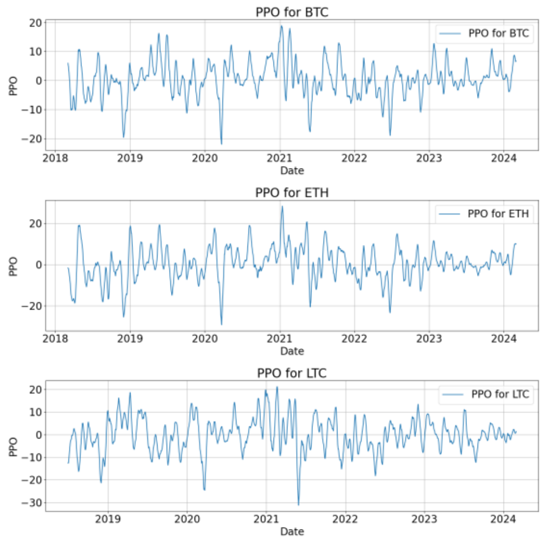

Fig. 13. PPO calculation results

3.11. ROC (Rate of Change)

The Rate of Change (ROC) indicator measures the percentage change in price between the current price and the price a certain number of periods ago. The ROC is a momentum oscillator that helps identify trend reversals, strength, and weaknesses within a trend. By comparing the current price to previous prices, the ROC reflects the speed at which prices are changing. A rising ROC indicates an acceleration in the upward trend (bullish momentum), while a falling ROC suggests a deceleration or a downward trend (bearish momentum) [5].

The formula for calculating the ROC is:

$$ROC = \left(\frac{\text{Close}_{\text{current}} - \text{Close}_{\text{n periods ago}}}{\text{Close}_{\text{n periods ago}}}\right) \times 100 \quad (15)$$

The ROC calculation involves finding the difference between the current closing price and the closing price from n periods ago, then dividing this difference by the closing price from n periods ago. Multiplying the result by 100 converts this ratio into a percentage, which represents the rate at which the price has changed over the specified period. This percentage change provides a straightforward measure of the momentum, with positive values indicating upward momentum and negative values signaling downward momentum.

From a computational perspective, implementing the ROC in Python involves using data manipulation libraries like Pandas, which can efficiently handle time-series data. The process includes accessing historical price data, applying the ROC formula using a combination of shift and rolling functions to calculate the price changes over the desired periods, and then scaling the results to percentage terms.

Figure 14 presents the results of ROC calculations for BTC, ETH and LTC.

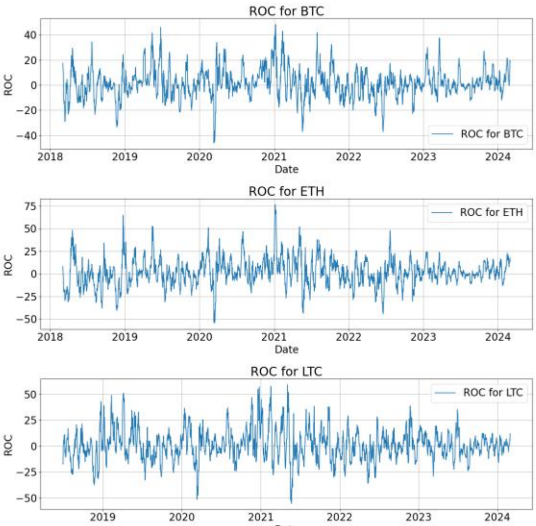

Fig. 14. ROC calculation results

3.12. RSI (Relative Strength Index)

The Relative Strength Index (RSI) is a momentum oscillator that measures the speed and change of price movements. It operates within a range of 0 to 100 and is most commonly used to identify overbought or oversold conditions in a market. The RSI compares the magnitude of recent gains to recent losses to assess whether a security is overvalued or undervalued. This indicator was developed by J. Welles Wilder and is widely used in technical analysis to predict the future direction of prices [6].

The formula for calculating the RSI is:

$$RSI = 100 - \left(\frac{100}{1+RS}\right) \quad (16)$$

Where RS (Relative Strength) is the average of «n» days' up closes divided by the average of «n» days' down closes.

The RSI calculation involves several steps. First, one calculates the average gain and the average loss over a specific period, typically 14 days. The RS is then calculated by dividing the average gain by the average loss. The RSI is derived by subtracting the quotient of 100 divided by (1 + RS) from 100. The final value of the RSI oscillates between 0 and 100, where values above 70 indicate an overbought condition (potentially overvalued) and values below 30 suggest an oversold condition (potentially undervalued).

In terms of computational implementation, calculating the RSI in Python can be efficiently performed using data manipulation libraries such as Pandas. This involves computing the daily price changes, separating the gains and losses, calculating their averages over the chosen period, and applying the RSI formula.

Figure 15 presents the results of RSI calculations for BTC, ETH and LTC.

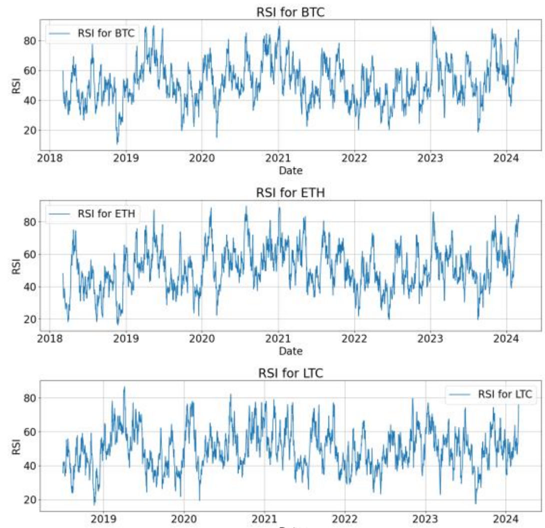

Fig. 15. RSI calculation results

3.13. TRIX (Triple Exponential Average)

The TRIX indicator, short for Triple Exponential Average, is a momentum oscillator that filters out price movements that are considered insignificant or minor. It does this by applying a triple smoothing of the closing prices, making it highly effective in identifying changes in trend direction. The TRIX oscillates around a zero line, where its primary use is to spot divergences, overbought, or oversold conditions, and to generate signals through its crossover with a signal line (often a moving average of the TRIX values) [7].

The TRIX is calculated through a three-step exponential smoothing process, which can be expressed as:

$$\text{TRIX} = 3^{rd}\ \text{EMA of (EMA of (EMA of Closing Price))} \quad (17)$$

The rate of change of the TRIX is often plotted to serve as the signal line:

$$\text{ROC of TRIX} = \frac{\text{Today's TRIX-Yesterday's TRIX}}{\text{Yesterday's TRIX}} \times 100 \quad (18)$$

The TRIX calculation begins with the exponential moving average (EMA) of the closing prices, followed by taking the EMA of that EMA, and then again taking the EMA of the result, effectively applying exponential smoothing three times. This triple smoothing is designed to filter out market noise and highlight significant trends and reversals. The final step often involves calculating the rate of change (ROC) of the TRIX to generate a signal line, which helps in identifying bullish or bearish momentum and potential entry or exit points.

In terms of computational implementation, the TRIX requires multiple iterations of EMA calculations, making efficient use of Python's data analysis libraries such as Pandas essential.

Figure 16 presents the results of TRIX calculations for BTC, ETH and LTC.

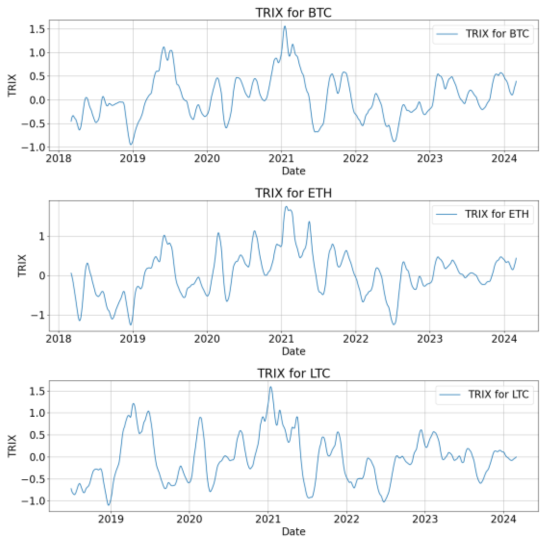

Fig. 16. TRIX calculation results

Conclusion

In the realm of cryptocurrency analysis, the intersection of computer science and financial market study presents a compelling avenue for exploring the volatile dynamics that define digital currencies. This article has ventured into an in-depth examination of the cryptocurrency market, utilizing a sophisticated computational approach to decipher the complex behaviors and trends that characterize this rapidly evolving landscape. Through the application of various technical momentum indicators, implemented via Python, we've demonstrated the profound impact of data analysis techniques in uncovering insights within the cryptocurrency domain.

The methodology adopted in this research underscores a significant shift towards a more technical and data-driven analysis of financial markets, particularly cryptocurrencies. By harnessing the capabilities of Python and its powerful libraries for data manipulation and statistical analysis, we have navigated through the intricacies of market data, extracting meaningful patterns and indicators of market sentiment and potential movements. This computational approach not only enhances the accuracy and efficiency of market analysis but also democratizes access to sophisticated financial analysis tools, enabling both researchers and traders to make informed decisions based on robust data analysis.

The core of our investigation centered around the intelligent analysis of time series data, specifically focusing on the OHLCV datasets for major cryptocurrencies. This choice of data representation is pivotal in financial market analysis, as it captures the essential elements required for a comprehensive evaluation of market dynamics. Through the application of a series of technical momentum indicators, we've delved into the quantitative aspects of market trends, momentum, and investor sentiment, providing a nuanced understanding of the factors that drive price movements in the cryptocurrency market.

One of the most significant outcomes of this research is the illustration of how computer science techniques can be effectively applied to financial data analysis, transcending traditional economic analysis methods. The focus on algorithmic calculations and the implementation of these indicators in Python showcases the versatility and power of programming in extracting and interpreting complex data sets. This approach not only facilitates a deeper understanding of market dynamics but also opens up new avenues for the development of automated trading systems and predictive models, leveraging the insights gained from the technical indicators.

Furthermore, this research highlights the importance of a computer science perspective in the field of financial analysis, where the ability to manipulate large datasets, apply statistical models, and develop algorithms becomes a crucial skill set. The application of computational methods in cryptocurrency analysis represents a fusion of disciplines, bringing together the rigor of computer science with the insights of financial theory to create a more comprehensive and nuanced understanding of market behaviors.

In conclusion, the exploration of cryptocurrency market dynamics through the lens of computer science and technical momentum indicators has provided valuable insights into the underlying patterns and trends of this volatile market. The successful application of these indicators, facilitated by Python, underscores the potential of computational techniques in enhancing our understanding and analysis of financial markets. As the cryptocurrency market continues to mature and evolve, the methodologies and insights presented in this article will serve as a valuable resource for future research and analysis, contributing to the development of more sophisticated and informed trading and investment strategies. The convergence of computer science and financial analysis in this domain not only enriches our understanding of market dynamics but also paves the way for innovative approaches to financial market analysis in the digital age.

References

1. Anuradha, T., Formal, PSGAS., RamaDevi, J. (2024). Hybrid Model for Rainfall Prediction with Statistical and Technical Indicator Feature Set. Expert Systems with Applications. Elsevier. https://www.sciencedirect.com/science/article/pii/S0957417424001258
2. Gong, L., Gong, W., Luo, J., Yang, R. (2024). Trend-Enhanced Improved Bollinger Bands Trend-Following High-Frequency Trading Strategy for Futures Market. Highlights in Business, Economics, and Management. PDF: https://drpress.org/ojs/index.php/HBEM/article/download/16586/16103
3. Alfaro, A., Guerrero, LX., Albino, L., Raya, A. (2024). Perturbative analysis of the three-gluon vertex to one-loop. arXiv preprint arXiv:2402.05262. PDF: https://arxiv.org/pdf/2402.05262
4. Eichenlaub, M., Pleus, S., Freckmann, G. (2024). A Proposal for the Clinical Characterization of Continuous Glucose Monitoring Trend Arrow Accuracy. https://journals.sagepub.com/doi/abs/10.1177/19322968241232679
5. Garraux, H. (2024). The Measure of Potential Profitability in Intraday Trading Utilizing the Relative Strength Index (RSI) with Various Simple Moving Averages (SMA) on the S&P 500. Furman University. https://scholarexchange.furman.edu/scjas/2024/all/498/
6. Chen, J.H., Hussain, S., Chen, F.Y. (2023). The Relationship between VIX and Technical Indicator: The Analysis of Shared-Frailty Model. Journal of Applied Finance & Banking, 13(3). Scienpress Ltd. PDF: http://www.scienpress.com/Upload/JAFB/Vol%2013_3_5.pdf
7. Pasaribu, M.K.N. (2024). Technical Analysis of Stock Investment Decisions in Telecommunication Companies Listed on the Indonesian Stock Exchange. Indonesian Journal of Banking and Finance. PDF: https://journal.formosapublisher.org/index.php/fintech/article/download/7122/7486
8. Hackfort, S., Marquis, S., Bronson, K. (2024). Harvesting value: Corporate strategies of data assetization in agriculture and their socio-ecological implications. Big Data & Society. https://journals.sagepub.com/doi/abs/10.1177/20539517241234279
9. Ghulam, Y., Dhruva, K. (2024). Banking sector reforms in a challenging environment: An emerging financial market experience. International Review of Economics & Finance. https://www.sciencedirect.com/science/article/pii/S1059056024000972
10. Long, W., Gao, J., Bai, K., Lu, Z. (2024). A hybrid model for stock price prediction based on multi-view heterogeneous data. Financial Innovation. https://link.springer.com/article/10.1186/s40854-023-00519-w
11. Aliyon, K., Ritvanen, J. (2024). Deep Learning-Based Electricity Price Forecasting: Findings on Price Predictability and European Electricity Markets. SSRN Electronic Journal. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4738917
12. Diqi, M., Hiswati, M.E., Wijaya, N. (2024). Stacked Gated Recurrent Units and Indonesian Stock Predictions: A New Approach to Financial Forecasting. Jurnal Ilmu Komputer dan Informasi. https://ejournal.uhb.ac.id/index.php/IKOMTI/article/view/1106

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ІСТОРИЧНИХ ДАННИХ ЧАСОВИХ РЯДІВ ДИНАМІКИ КРИПТОВАЛЮТНОГО РИНКУ НА ОСНОВІ НАБОРУ ДАНИХ OHLCV ТА ТЕХНІЧНИХ ІНДИКАТОРІВ ІМПУЛЬСІВ

І. С. Узун[1], М.В. Лобачев[1]

[1] Національний університет «Одеська політехніка»

Анотація. *У цій статті представлено аналіз історичної динаміки ринку криптовалют на основі часових рядів даних з використанням набору даних OHLCV. У дослідженні представлені результати розрахунків основних технічних індикаторів імпульсів криптовалютного ринку. Використовуючи інтелектуальні обчислювальні методи, у роботі оцінюються закономірності та тенденції в даних основних криптовалют. Дослідження підкреслює важливість технічного аналізу в розумінні нестабільного ландшафту цифрових валют.*

Ключові слова: *часові ряди, аналіз даних, криптовалютний ринок, індикатори імпульсів, індикатори технічного аналізу, OHLCV.*

Отримано 15.02.2024

Узун Ілля Святославович, Національний університет «Одеська політехніка», аспірант Інституту Штучного Інтелекту та Робототехніки, старший викладач кафедри Штучного Інтелекту та Аналізу Даних. Ільфа і Петрова 29, кв. 127, Одесса, Украина, E-mail: uzun.illia.main@gmail.com, тел. +380679172040

Illia Uzun, Odesa Polytechnic National University, PhD student at the Institute of Artificial Intelligence and Robotics, Senior Lecturer at the Department of Artificial Intelligence and Data Analysis. Il'fa and Petrova str. 29, ap. 127, Odesa, Ukraine, E-mail: uzun.illia.main@gmail.com, tel. +380679172040

ORCID ID: 0000-0001-6619-4862

Лобачев Михайло Вікторович, Національний університет «Одеська політехніка», директор Інституту Штучного Інтелекту та Робототехніки, к.т.н., професор кафедри Штучного Інтелекту та Аналізу Даних. Просп. Шевченка, 1, Одеса, Україна, E-mail: lobachev@op.edu.ua, тел. +380952788602

Mykhaylo Lobachev, Odesa Polytechnic National University, The Director of the Artificial Intelligence and Robotics Institute, Ph.D, Professor of the Department of Artificial Intelligence and Data Analysis. Shevchenko av., 1, Odessa, E-mail: lobachev@op.edu.ua, tel. +380952788602

ORCID ID: 0000-0002-4859-304X