DOI 10.33111/nfmte.2023.175

IDENTIFYING MOMENTS OF DECISION MAKING ON TRADE IN FINANCIAL TIME SERIES USING FUZZY CLUSTER ANALYSIS

Vladyslav Kabachii

Vinnytsia National Technical University
95 Khmelnytske Hwy., Vinnytsia, 21021, Ukraine
ORCID: 0009-0001-3158-2889, E-mail: vkabachiy@gmail.com

Roman Maslii

Vinnytsia National Technical University
95 Khmelnytske Hwy., Vinnytsia, 21021, Ukraine
ORCID: 0000-0003-3021-4328, E-mail: romas4580@gmail.com

Serhii Kozlovskyi

Vasyl' Stus Donetsk National University
21 600-richchia Str., Vinnytsia, 21021, Ukraine
ORCID: 0000-0003-0707-4996, E-mail: s.kozlovskyy@donnu.edu.ua

Oleksandr Dronchack

IntelSoft Technologies LLC
58 Yaroslavska Str., Kyiv, 04071, Ukraine
ORCID: 0009-0006-2536-9906, E-mail: vinmetiz@gmail.com

The article investigates the problem of identifying trading decision points in financial time series using the Fuzzy C-Means (FCM) method. The authors argue that classical forecasting methods have limited effectiveness for decision-making in trading, as they do not take into account market structure and nonlinear patterns. The proposed methodology involves analysing time series using additional features derived from technical indicators (MACD, Stochastic) and further clustering based on FCM, which allows identifying market entry and exit points. In contrast to traditional approaches based on the assessment of forecasting accuracy (e.g. MAE, RMSE, MAPE), this study focuses on financially oriented metrics such as Net Profit, Max Drawdown, Win Rate and Profit Factor, which more accurately reflect the effectiveness of trading strategies in real market conditions. Experiments on the currency pairs EUR/USD, AUD/USD, USD/JPY, USD/CAD on daily and four-hour timeframes have demonstrated that the use of the proposed approach can improve the efficiency of trading strategies. The simulation results showed fairly high stable profitability results with low risks (drawdown). The proposed approach can be useful in developing automated trading systems and further research in the field of financial analytics.

Keywords: *financial time series, trading, cluster analysis, fuzzy c-means, technical analysis, financial performance metrics, trend prediction*

JEL Classification: C53, C38, C63, G10

Introduction

The pursuit of stable, long-term profits remains the main goal of trading. Achieving success in trading financial instruments largely depends on a trader's ability to quickly and accurately interpret market fluctuations, especially changes in trends on financial time series. Effective analysis can provide a significant competitive advantage by making profits more predictable. Therefore, combining different methods of market analysis is crucial, emphasising the need to study a wide range of indicators to make informed decisions.

Time series forecasting methods, such as simple moving average, ARMA, ARIMA, etc. [1, 2], which are commonly used in many works on financial time series forecasting, usually make a forecast one step ahead. They can be useful in certain situations (for example, in macroeconomic forecasting), but in trading they often turn out to be inappropriate for several important reasons. Trading is not just about accurately predicting future prices, but also about effective risk management and decision-making regarding the moments of market entry and exit. The moment of entry into the market is key to making a profit, and the correct determination of the moment of exit allows you to preserve the achieved profits or limit losses, while the price between the entry and exit points can change significantly in different directions. One-step ahead forecasting is not capable of providing such strategic decisions, as it focuses only on predicting the price at the next moment, rather than analysing market conditions and determining the optimal points for transactions.

When predicting financial time series, an investor seeks to maximise profit rather than minimise the standard deviation, as in the case of function approximation. Article [3] describes the results of experiments on forecasting financial time series using a wide range of econometric approaches, spectral analysis and neural networks and shows that when optimising models based on the least squares method, the accuracy of the predicted directions of change in the price of a financial instrument is very low – just over 50%. To improve the accuracy of the forecast and, accordingly, the efficiency of the trading system, the mentioned article proposes to predict the direction of price change, rather than the exact value of the financial instrument, for which a model was built based on fuzzy logic, the rule base of which was formed using Elliott wave theory.

However, there is an alternative approach, when the profitability of a trading system is ensured by identifying entry and exit points, as these moments are critical for effective trading. For this purpose, technical analysis is used [4, 5], which is based on various indicators and strategies to identify the optimal moments for opening and closing trades. This allows traders to take into account not only the price, but also other important factors, such as volatility, trend changes, support and resistance levels, etc. The use of such method and similar characteristics allows not only to increase the probability of making a profit, but also to reduce potential risks.

We propose to call such a task – determining the moments of market entry and exit – the problem of identifying trading decision points in financial time series. In a similar formulation, the task was solved, for example, in [6], and in [7] it was about trend identification and pattern recognition in financial time series, which is essentially similar.

In general, there are two types of transactions: buying with the intention to sell at a higher price or selling with the intention to buy at a lower price. Fig. 1 shows such actions (Buy and Sell) on the hourly timeframe (H1) for the EUR/USD currency pair. This means that each count (or bar) on the chart reflects the price change for 1 hour of trading. Since the price representation on this chart is dotted, it means that each dot on the chart represents the price for 1 hour of trading. As a rule, this is the Close price, unless otherwise noted, although each hour in the database is usually represented by four prices: Open, High, Low, and Close, which respectively define: the price at the beginning of the hour, the maximum and minimum price during this hour, and the price at the end of the hour. You can read about different representations of price charts in [5]. It is worth noting here that further in the article we will use the D (daily) and H4 (four-hour) timeframes.

Fig. 1. Moments of rational actions (Sell and Buy) on financial time series

The task of analysing a financial time series in trading is to identify the described moments on the chart of a certain timeframe as accurately as possible, which is the goal of effective decision-making in financial markets. The profitability of a strategy and the efficiency of the system depend on the timeliness of decision-making. However, as noted in [4], classical approaches of technical analysis, although widely used in trading, provide a small advantage, rarely exceeding 55-60% of guessing the direction of price change, which is a low value for building effective trading systems. Therefore, traders try to combine different approaches, even within the framework of technical analysis.

In particular, although one time dimension is usually used for decision-making on financial time series, to increase the efficiency of applying technical analysis, it is possible to simultaneously take into account different timeframes, when a larger one, for example, a 4-hour (H4) acts as a filter for decisions made on a smaller 15-minute (M15), as was done in the article [8]. That is, only those signals that coincide with the direction of the trend on a larger timeframe are accepted. This allows to get rid of unprofitable transactions, which on M15 are directed against the dominant trend.

On the other hand, modern intelligent technologies, such as neural networks, genetic algorithms, fuzzy logic, etc., which, compared to traditional methods, can more accurately describe complex and variable data relationships, have more advantages in solving nonlinear problems with a complex structure and many factors, such as price forecasting in financial markets. An important point in their application is not only the preparation such data as time series, but also the understanding of the domain to which they are applied [9, 10]. Therefore, in order to effective use the latest technologies for financial markets, it is necessary to have the skills of a data scientist and preferably have the mindset of a trader who operations on the market for performs profit.

Analysis of recent research and publications

As shown above, it makes sense to forecast the development of financial time series not by predicting future exact values, but by identifying typical patterns in order to determine the future trend. For example, as stated in [8], this involves identifying certain moments in the time series when a decision to sell or buy is made (Fig. 1). In [8], these moments are determined on the basis of linear models, clear rules of behaviour of certain indicators at the current and previous moments. Built on the authors' practical experience in the real market as traders, with empirically obtained model parameters, such a decision-making system has demonstrated profitability both when tested on historical data and in real time for more than a year of practical application. Moreover, through a series of experiments and optimisation, the results were improved by another 15-25%.

Note that these patterns worked, although they represented rules clearly defined by the expert. However, they can be specified as rules in fuzzy logic models [3, 11], which allows fine-tuning their significance and parameters on historical data without expert judgments. They can also be defined as fuzzy time series based on a sequence of fuzzy sets [12]. As argued in [3], methods based on fuzzy sets have an advantage in terms of interpretability when identifying patterns in time series development.

At the same time, to identify patterns in the development of financial time series, various clustering methods can be no less effective tools. It should be noted that a wide range of different clustering algorithms have been used to solve this problem. In particular, in [13] the authors propose a recommender system that uses regression trees for dimensionality reduction and clustering based on self-organising maps (SOM). The system is designed to help stock market investors identify potential profit opportunities and better understand how to extract relevant information from stock price data.

In [14], the authors use the Kohonen mapping tool to identify different groups of cryptocurrencies for investment based on a number of indicators that characterise their risk and return. In [15], this toolkit is used to solve a similar problem, but in terms of analysing the rationality of traders' behaviour and segmenting them into groups that characterise the efficiency of their trading activities. The article [16] considers the problem of predicting the spread of financial crises by identifying groups of countries by the types of reaction to crisis phenomena using self-organising maps. A perceptron-type neural network is then used to predict the effects of crisis transfers for each individual cluster.

In [17], Encke et al. proposed a three-stage system combining multiple regression, type-2 fuzzy clustering, and neural networks for stock price forecasting. While this model effectively accounts for nonlinearities and complex interactions in financial data, it lacks a direct mechanism for identifying specific entry/exit points, as it focuses on general price trend forecasting rather than practical trading decisions. Mehmanpazir and Asadi [18] developed the evolutionary fuzzy expert system, which incorporates K-means clustering along with Mamdani-type fuzzy rule-based system and genetic algorithms for rule filtering and model parameter adjusting. Although K-means method reduces data complexity, its inability to handle overlapping clusters limits the accuracy of detecting nuances that are crucial to correctly identifying entry/exit points in volatile markets.

Shi et al. [19] analysed intraday financial time series using K-means, Chameleon, and self-organising maps. Their findings highlighted the stability and accuracy of SOM. However, the study primarily assesses the algorithmic performance without translating the results into clear entry/exit strategies, leaving a gap between clustering results and practical application.

In [20], Bandara et al. clustered time series using interpretable features and applied RNN models to each cluster for forecasting. Despite the method's reliability in improving forecasting accuracy, its focus on long-term trends rather than short-term signals reduces its usefulness for traders looking for precise identification of entry/exit points. Similarly, Long et al. [21] clustered investors using the desensitized transaction records and public market information to reduce the dimensions of the trading feature matrices, and then the matrices are fed into the convolutional neural network to predict price movements, and Wang et al. [22] introduced morphological similarity-based clustering of stock groups to learn patterns from similar stocks and make predictions based on deep learning models. Both approaches succeeded in segmenting trends, but do not provide mechanisms for directly correlating these trends with existing trading rules.

In [23], clustering was used to identify anomalies in cryptocurrency trading, demonstrating an increased return on investment by eliminating them. However, this approach is more suitable for filtering noise than for systematically identifying optimal entry/exit points. The paper [24] proposes K-means clustering in combination with ARIMA and LSTM for stock forecasting, but does not consider trading timeframes or how clustering can directly support entry/exit decisions. Finally, D'Urso et al. [25] proposed robust fuzzy clustering methods to manage outliers in data, but focused on data stability rather than practical implications for trading.

As can be seen, clustering as a fundamental method of unsupervised learning has undergone significant development and practical application in recent years. Since the introduction of the K-means algorithm [26], clustering has become important in time series analysis [19, 20, 22, 24]. The simplicity and efficiency of K-means have made it one of the most popular clustering algorithms, especially in scenarios where the goal is to divide data into separate clusters based on similarity. However, K-means is not without its drawbacks. One of the main disadvantages is its reliance on the assumption that clusters are spherical and of equal size, which often does not correspond to real data. To solve this problem, several extensions and variations of K-means have been developed, but the real breakthrough was the advent of the Fuzzy C-Means (FCM) algorithm [27], which is a fuzzy variant of the K-means partitional algorithm and is free of these drawbacks [28].

Compared to traditional methods, where an object belongs entirely to one cluster, FCM provides a more flexible approach to modelling uncertainty in data. The main idea of the method is that each object belongs to each cluster, but with a different degree of membership. FCM is well suited for tasks where objects may have ambiguous or overlapping cluster membership principles. Moreover, in the real world, it is not easy to define a clear boundary between clusters, so the fuzzy approach seems more adequate than the deterministic one. Finally, fuzzy sets has proven to be very useful for clustering time series with special properties, such as financial asset prices [12, 17, 25], including identifying dependencies in them.

Thus, the objective of the study is to implement a fuzzy clustering method for analysing a financial time series using technical analysis indicators, with further calculation of the moments for opening and closing transactions depending on the degree of belonging of the current market development pattern to one or another cluster. The obtained signals should be suitable for building expert decision-making systems for conducting purchase and sale transactions in the financial markets in automatic mode.

Description of the methodological approach to applying FCM in trading systems in financial markets

To achieve the stated goal, it is necessary to select financial time data (for which financial instruments, time period and timeframe) that will be used in this task. For research using FCM, the data must be prepared in a certain way and additionally processed to obtain the necessary signals. Thus, the modelling process will consist of the following stages (see Fig. 2), which are the basis of the proposed approach:

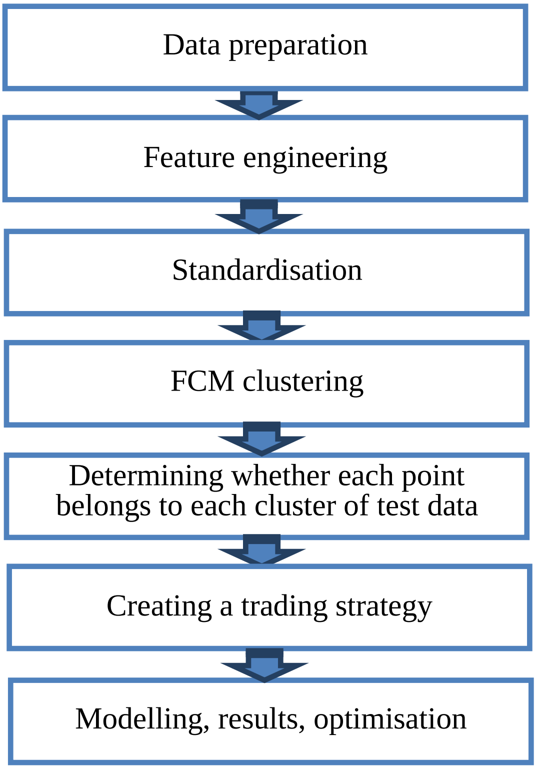


Fig. 2. Stages of the proposed methodological approach

1. Data preparation, which includes: collecting sufficient information on financial instruments for each timeframe (opening, closing prices, etc.).
2. Feature engineering – selection of the most informative features of the financial time series and calculating technical indicators. Here, new predictors are calculated and added to the main data set (price values over time) for the purpose of joint analysis to obtain a predictive model.
3. Standardisation of data before clustering to ensure correct operation of the algorithm. This procedure normalises each feature in the dataset by converting its values to z-scores:

$$X_{scaled} = \frac{X - \bar{X}}{\sigma}, \tag{1}$$

where X is the initial value; $\bar{X}$ is the mean value of the feature; σ is the standard deviation of the feature.

Also at this stage, the data is divided into training and test samples for each financial instrument to verify the model's robustness to new data and prevent overfitting.

4. FCM clustering on training data to partition them, which maximises the similarity between data within each cluster and minimises the similarity between data of different clusters [29]:

$$\sum_{j=1}^{N} \sum_{i=1}^{C} \left\| x_j - v_i \right\|^2 \mu_{ij}^m \to min, \tag{2}$$

where μ_{ij} is the degree of membership to cluster i of a point x_j, which is a data vector of observed object j; v_i is the centre of cluster i, which is a vector of the same dimension as the input data; C is the number of clusters; N is the total number of points in the sample; m is the degree of fuzziness, $m > 1$; $\left\| x_j - v_i \right\|$ is the distance between point x_j and the centre of cluster v_i; and the following condition must be met:

$$\sum_{i=1}^{C} \mu_{ij} = 1. \tag{3}$$

The procedure of this algorithm is as follows:

a) Determine the initial values of C, m and $v_i, i = \overline{1, C}$ (initial centres for each cluster).

b) Calculate the membership value of each data point (point on the price chart described by the feature vector) to each cluster using the following formula:

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\left\| x_j - v_i \right\|}{\left\| x_j - v_k \right\|} \right)^{2/(m-1)}}, \qquad i = \overline{1, C}, j = \overline{1, N}. \tag{4}$$

c) Calculate new centres for each cluster using the formula:

$$v_i = \frac{\sum_{j=1}^{N} x_j \cdot \mu_{ij}^m}{\sum_{j=1}^{N} \mu_{ij}^m}, \qquad i = \overline{1, C}. \tag{5}$$

d) Repeat steps b and c until the difference between the values of objective function (2) at the new and previous stage becomes less than the threshold value.

5. Investigation of the optimal number of clusters, which is necessary to ensure high-quality and reliable grouping of time series. This will improve clustering accuracy, model generalisability, analysis efficiency, and optimisation of computing resources, which is important for further use of the results in forecasting and decision-making. To do this, it is advisable to use one of the key metrics for assessing the quality of clustering in the FCM approach – Fuzzy Partition Coefficient (FPC). It allows determining the optimal number of clusters by assessing the clarity of the distribution of points between clusters. FPC is defined as:

$$FPC = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C} \mu_{ij}^2. \tag{6}$$

The *FPC* values range from 0 to 1. If $FPC \approx 1$, it means that the clustering is clear, if $FPC \approx 0$, then the clustering is blurred, which may indicate insufficient resolution between clusters. It is important to analyse the change in *FPC* as *C* increases by repeating the calculations in step 4.

6. Assessment of the membership of each point in the test dataset to the clusters obtained during training.

7. Creating a trading strategy – defining the rules for using the membership levels of each point to different clusters to make decisions about opening or closing positions in trading on financial markets. The following approach is proposed to determine the potential moments of market entry and exit for opening long (Buy) or short (Sell) positions when the degree of membership of the point to the corresponding cluster exceeds a certain value.

8. Modelling the trading strategy on training and test data for selected financial instruments on different timeframes. Obtaining statistical data based on the modelling results, drawing conclusions and optimising the parameters of the resulting decision-making system.

Selecting metrics for evaluating model performance

To assess the accuracy of time series forecasting models, in particular for predicting price values, such indicators as mean absolute error (MAE), root mean square error (RMSE) and mean absolute percentage error (MAPE) are widely used [12]. However, as noted above, they are not sufficiently informative for analysing the efficiency of trading systems in financial markets. The main reason for this is that these metrics only assess the deviation of predicted values from actual ones, without taking into account the financial consequences of these deviations.

In trading, the main criterion is the ability of s trading system to correctly determine the moments for opening and closing trades, and not the accuracy of the price forecast at each time step. For example, a system can have a low MAE value, predicting the price with high accuracy, but not taking into account critical signals for entering or exiting the market. As a result, such a system can generate losses even if its forecasts are generally accurate. This is unacceptable for trading, as the general direction of the market (up or down) is often more important than the accuracy of short-term forecasts.

Another problem is that these metrics do not take into account the dynamic nature of financial markets. Trading involves complex relationships between risk and reward that are not reflected in statistical metrics of accuracy. Therefore, MAE, RMSE and MAPE metrics are not suitable for evaluating trading systems because they ignore key aspects of trading: financial performance, risk management, and the ability to make effective decisions in real-world market conditions.

To evaluate trading systems, it is more appropriate to use metrics that focus on financial results and take into account the specifics of market operations. One of the key ones is Net Profit, which reflects the total income from all transactions after deducting losses. This metric allows us to assess the system's ability to generate stable income over a long period of time. For example, even if the accuracy of the system's forecasts is low, but its decisions allow it to achieve significant profits, such a system is considered effective.

Another metric is Drawdown, which characterises the fall of the trader's capital from its maximum to a subsequent minimum before a new increase. Max Drawdown, which is determined accordingly over the entire test period, is an important indicator for risk assessment, as a large drawdown may indicate the risk of losing a significant part of the capital. In combination with the profitability indicator, drawdown allows you to assess the stability of the trading system.

Additionally, the Win Rate metric is used to estimate the share of profitable transactions in the total volume of transactions. Although this metric does not take into account the amount of profits and losses, it gives an idea of the overall stability of the system. At the same time, the Profit Factor reflects the ratio of the volume of profitable trades to the volume of unprofitable trades. Combined with other metrics, they provide a comprehensive picture of the system's performance.

Thus, to evaluate the results of the trading system built on the basis of the proposed methodological approach, we will use some financially oriented metrics (Net Profit, Max Drawdown, Win Rate, Profit Factor) that provide a more accurate and, most importantly, practical assessment of the effectiveness of trading systems.

Data preparation and feature engineering

We have chosen currency pairs as statistical material for modelling the development of financial time series, since historical data on them can be taken from various sources. We will use daily (D) and four-hour (H4) chart data of the major currency pairs such as EUR/USD, AUD/USD, USD/CAD, USD/JPY. Taking into account the availability of data and the need to split it into training and a test samples, we will take 800 data points for each of them for each currency pair.

For the D timeframe, this will be data from 31.03.2017 to 01.05.2020 for training sample and from 04.05.2020 to 31.05.2023 for the test one. For the H4 timeframe, we use data from 24.06.2022 to

27.12.2022 for the training sample and from 27.12.2022 to 31.05.2023 for the test sample. That is, for daily data, each sample takes a period of 3 years and one month (taking into account breaks in trading), and for H4 – about 6 months.

To collect historical data in the required format, we will apply the MetaTrader 4 platform [30]. Using it, we download data in a csv file separately for each instrument. To process the data within the framework of the described approach, we chose Jupyter Notebook – an interactive programming environment that allows combining code, text descriptions and graphs in one document, supporting the Python language.

To perform feature engineering, extract characteristic features from the original financial time series and identify patterns in the development of price curves, we will use the tools of technical analysis of market.

Technical analysis is a method of forecasting prices by examining charts of market movements (financial time series) for previous periods of time. All technical analysis techniques were created separately from each other and only in the 1970s were united into a single theory with a common philosophy, axioms and basic principles [31]. The classification of technical analysis methods can be found, for example, in [4, 5, 31]. In this paper, we will use mathematical technical analysis, based on indicators and/or oscillators, additionally built on a price chart. They represent specific mathematical functions, which can help interpret the probability of further development of time series in the financial market.

Indicators help to identify trends and their reversals, allow a deeper assessment of the balance of power between buyers and sellers, but they have a drawback: they often contradict each other. Some are better at catching trends, others work better in horizontal trading corridors. Some are good at signalling reversals, while others are better at tracking trend directions.

Most beginners look for a single indicator to guide them through the stock market intricacies [5]. In fact, a good result is obtained by thoughtful use of several indicators together. Thus, professionals divide indicators into three groups: trend-following indicators, oscillators, and sentiment indicators. Trend-following indicators (also called trend indicators) are excellent for analysing a market moving up or down, but when the market is stagnant, their signals are unreliable and often wrong. Oscillators are excellent at catching changes in stagnant markets, but once a trend is established, they provide premature and even false signals [5]. Sentiment indicators are obtained by information quantification of news sentiment and are significant auxiliary precursors in detecting changes in primarily global trends or sharp spikes in financial markets [32].

Relying on our own experience of professional trading, three indicators MA, MACD and Stochastic [4, 5, 31] were used as a basis for this study. As a result of a number of experiments, it was found that the best clustering results are obtained by using one of the trend-following indicators – MACD, consisting of two lines MACD1, MACD2, and one oscillator, also consisting of two lines S%K, S%D (see Figs. 3 and 4).

	Date	Close	macd1	macd2	S%K	S%D
0	2017.03.31	1.06541	0.003044	0.003866	36.291459	58.358209
1	2017.04.03	1.06690	0.002304	0.003554	17.852860	37.119216
2	2017.04.04	1.06722	0.001724	0.003188	11.755911	21.966743
3	2017.04.05	1.06617	0.001166	0.002783	6.459825	12.022865
4	2017.04.06	1.06435	0.000570	0.002341	4.089101	7.434946
...	...	...	...	...	...	...
795	2020.04.27	1.08283	-0.003889	-0.003456	16.378885	15.730082
796	2020.04.28	1.08182	-0.003873	-0.003539	23.433646	17.712712
797	2020.04.29	1.08731	-0.003379	-0.003507	31.853314	23.888615
798	2020.04.30	1.09498	-0.002341	-0.003274	51.833580	35.706846
799	2020.05.01	1.09789	-0.001270	-0.002873	78.112153	53.933015

Fig. 3. Numerical representation of the training set of daily data for the EUR/USD currency pair

Note that the price of the financial instrument does not participate in clustering, but only these four indicators calculated based on it.

To calculate additional features and build models, we will use only the Close price, which is most often the basis for building line charts of financial markets. By removing unnecessary columns and adding new calculated indicators, we obtain a database, a fragment of which in Jupyter Notebook (Python) is shown in Fig. 3, for further normalization and application in FCM.

A visual representation of the time series corresponding to the training data for the EUR/USD pair on the daily timeframe for the period from 31.03.2017 to 01.05.2020, along with the MACD1, MACD2, S%K, S%D features, is shown in Fig. 4. It represents an upward trend in the asset price, followed by a prolonged decline, and then an upward movement again.

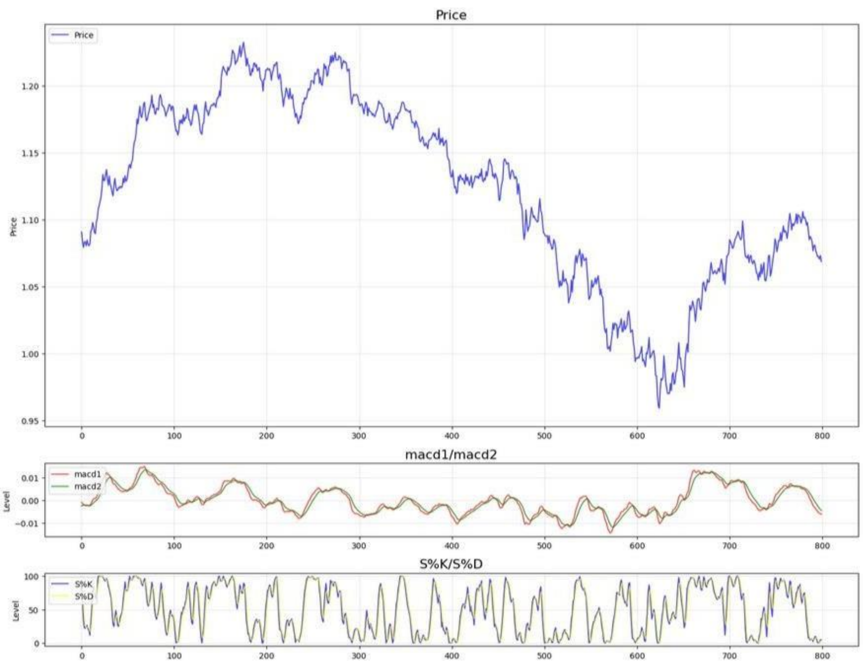


Fig. 4. Visualisation of time series of EUR/USD and MACD1, MACD2, S%K, S%D indicators for the test set of the daily chart

The results of the modelling

After the first three stages of our methodological approach – Data preparation, Feature engineering and Standardisation, we proceed to the fourth stage, where clustering in the space of features is performed. To build the FCM model, we used the fuzzy clustering function (fuzz.cluster.cmeans from the scikit-fuzzy library for the Python programming language, as seen in Fig. 5.a), which takes the following input parameters:

```
# Fuzzy C-Means clustering
cntr, u, u0, d, jm, p, fpc = fuzz.cluster.cmeans(
    data_scaled, n_clusters, m=m, error=0.005, maxiter=1000, init=None
)
```

(a)

```
u_new, _, _, _, _, _ = fuzz.cluster.cmeans_predict(
    data_scaled, cntr, m, error=0.005, maxiter=1000
)
```

(b)

Fig. 5. Fuzzy clustering functions of the scikit-fuzzy library:
for training data (a), for prediction on test data (b)

– *data_scaled* is an input data array of dimension (*n_samples*, *n_features*), where *n_samples* is the number of points (data vectors for all observations) in the sample, *n_features* is the number of measurements (features). In our case, *n_samples* is equal to 800, and *n_features* is equal to 4 (MACD1, MACD2, S%K, S%D);

– *n_clusters* is the number of clusters into which the data should be divided;

– *m* is the fuzziness coefficient, which determines the level of clustering fuzziness: when $m = 1$, the clustering becomes rigid (clear), where each point belongs to only one cluster (similar to K-means); when $m > 1$, the clustering becomes fuzzy, and points can simultaneously belong to several clusters with different levels of membership;

– *error* is the convergence threshold of the algorithm (usually a small value, for example, 0.005), which determines the termination of optimisation process if the change in the objective function (2) between iterations is less than a given value;

– *maxiter* is the maximum number of iterations of the algorithm after which execution is terminated.

It returns the following outputs:

– *cntr* is an array of dimension (*n_clusters*, *n_features*) containing the coordinates of the cluster centroids;

– *u* is a matrix (*n_clusters*, *n_samples*) containing the degrees of membership of each point to each of the clusters;

– *fpc* is the clustering quality index.

The fuzz.cluster.cmeans_predict function was used to predict the membership of new data to the obtained clusters (Fig. 5.b). It processes a test array *data_scaled* of normalised data of dimension (*n_samples_test*, *n_features*), where *n_samples_test* is the number of new objects (predicted points in the time series of the test sample). The function uses the defined *cntr* and the same parameters *m*, *error*, *maxiter*, as in training. The output of the function is a matrix of membership of test sample points to all clusters *u_new* of dimension (*n_clusters*, *n_samples_test*).

We conduct simulations with different numbers of clusters to obtain the FPC (6) for each variant. A visual representation of the modelling results is shown in Fig. 6.

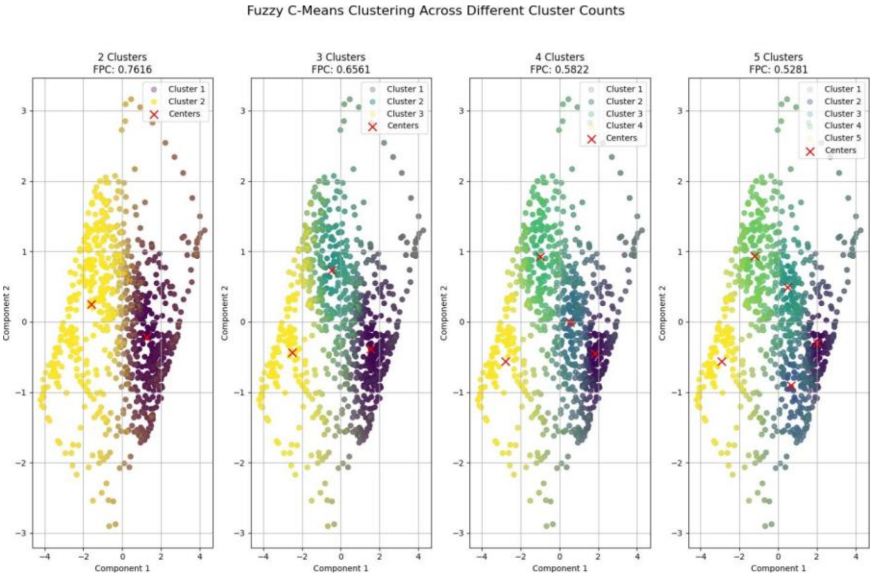


Fig. 6. Visualisation of the impact of the number of clusters in fuzzy clustering on FPC for a daily chart

To check the correctness of the clustering results, we use the principal component analysis (PCA) implemented in the Python scikit-learn library. This approach reduces the dimensionality of the space from four-dimensional (consisting of the features MACD1, MACD2, S%K, S%D) to two-dimensional (Component 1, Component 2). It is done by selecting two principal components that explain the largest variance in the data. Visualisation of the resulting clusters allows us to assess their quality and degree of separation. Since the clusters in Fig. 6 obtained different values of FPC, this allows us to study the impact of the choice of the number of clusters n_clusters and the level of fuzziness m on the data structure in a reduced dimensional space.

Studying the data to divide it into a number of clusters revealed a very important insight: having 4 features (MACD1, MACD2, S%K, S%D), it is most appropriate to identify two clusters, because for the task at hand, there are only two decisions in the financial market: go up (Buy) or go down (Sell).

Of course, we can also say that there is no explicit decision on the direction of price movement when a flat state is observed. But we take into consideration, that entry thresholds are used for each decision. And if the calculated values of membership to the corresponding clusters are below these entry thresholds, then it is a flat state, during which no decisions are made.

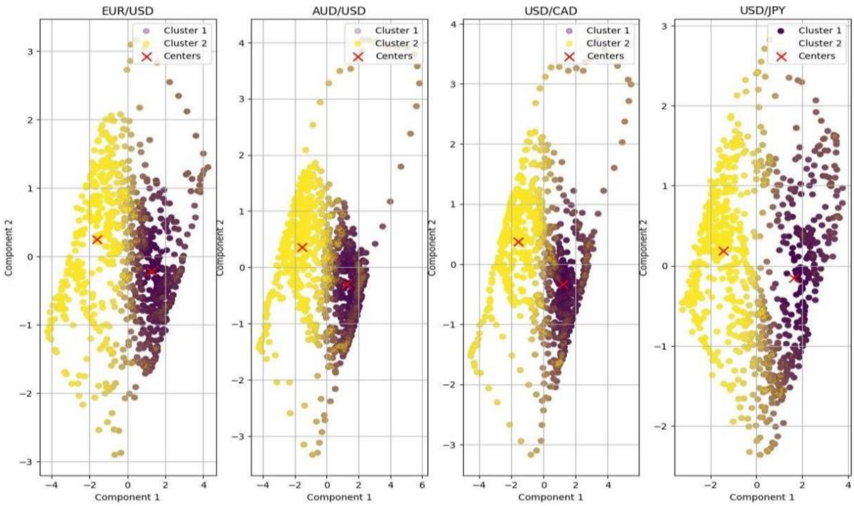
The experiments also showed that the best option for fuzziness rate is $m = 2$, although $m = 3$ was sometimes better for some datasets.

Similar experiments were done for other instruments on D and H4 timeframes, which confirmed the findings.

The result of applying the FCM algorithm for two-cluster division of the training sample of each currency pair on two selected timeframes is presented in Fig. 7.

As can be seen from Fig. 7, the two-component PCA representation using fuzzy clustering has similar characteristics for different currency pairs, indicating that the behavioural patterns (time series development dynamics) in these currency pairs are similar.

Afterwards, for each point on the chart (for each day on the D timeframe), the indicator of membership to two clusters (Buy and Sell) is determined. This indicator has a value in the range from 0 to 1.



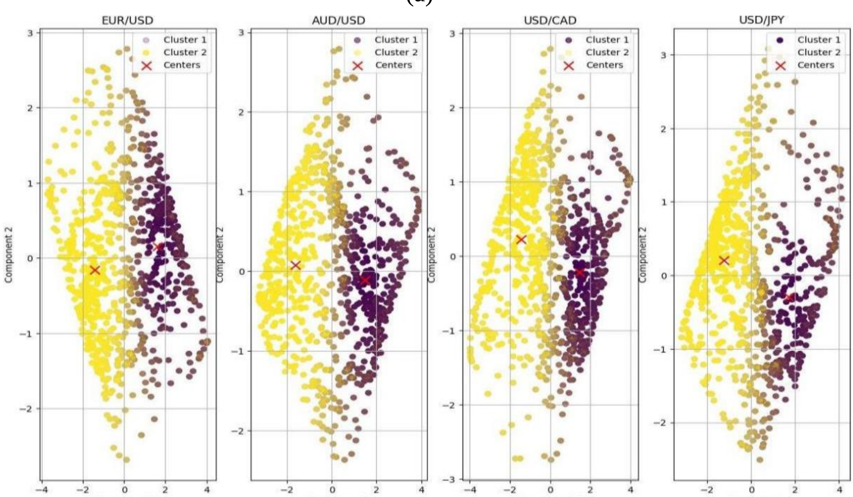
	macd1	macd2	S%K	S%D
Cluster 1	-0.58	-0.46	-0.73	-0.76
Cluster 2	0.75	0.6	0.91	0.93

macd1	macd2	S%K	S%D
-0.54	-0.4	-0.77	-0.79
0.68	0.5	0.95	0.98

macd1	macd2	S%K	S%D
-0.67	-0.51	-0.96	-0.98
0.49	0.36	0.76	0.77

macd1	macd2	S%K	S%D
-0.81	-0.64	-0.88	-0.92
0.68	0.54	0.8	0.83

(a)



	macd1	macd2	S%K	S%D
Cluster 1	-0.78	-0.65	-0.87	-0.9
Cluster 2	0.68	0.57	0.78	0.8

macd1	macd2	S%K	S%D
-0.75	-0.63	-0.79	-0.82
0.82	0.71	0.83	0.86

macd1	macd2	S%K	S%D
-0.67	-0.53	-0.82	-0.84
0.7	0.55	0.83	0.86

macd1	macd2	S%K	S%D
-0.78	-0.62	-0.99	-1.01
0.55	0.45	0.71	0.73

(b)

Fig. 7. Two-dimensional PCA representation of FCM clustering on training samples of four currency pairs for daily (a) and four-hour (b) timeframes

In order to obtain upward (Buy) and downward (Sell) signals, we set the following sensitivity thresholds for the two clusters (they are determined empirically): *open_buy* = 0.8, *close_buy* = 0.7, *open_sell* = 0.8, *close_sell* = 0.7. According to them, we will determine market entry points (opening positions) and market exit points (closing positions).

Fig. 8 shows a fragment of the graph with the levels of membership to the Buy and Sell clusters (orange and blue, respectively, in the lower part of the figure). In the upper left corner of the chart, there are marks corresponding to the "Buy", "Close Buy", "Sell" and "Close Sell" moments, which are determined according to the procedure described above. The green line on the central graph represents the cumulative profit-loss curve from the first to the last trade (if there is an open trade at the end of the analysed period, it is closed at the final price, generating the corresponding financial result).

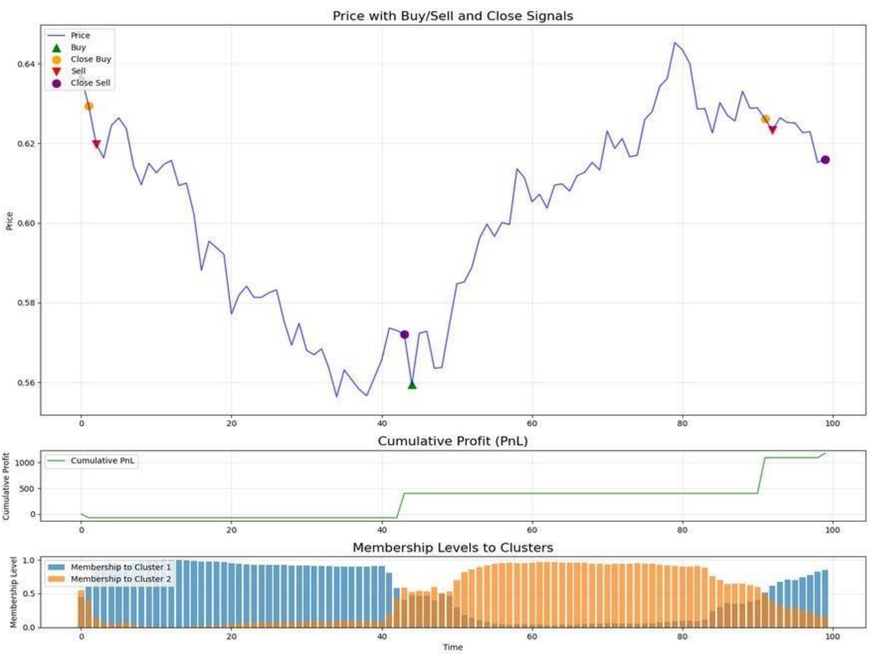


Fig. 8. Decision-making moments based on the fuzzy membership of points to one of two clusters (Buy and Sell)

Fig. 9 shows a visualisation of the decisions made for the EUR/USD pair over the entire test range for daily data.

Fig. 9. Decision moments to buy, sell or close a position depending on the membership to one of two clusters on test data for the daily timeframe

Fig. 10 shows the listing of all transactions for the same period as Fig. 9, with the financial result of each transaction. The resulting statistical metrics are shown in Fig. 11: the number of successful and unsuccessful trades with their overall financial result, the number of buy and sell trades with their characteristics, the profit factor, and the maximum drawdown, which makes it possible to assess the system's performance. During the test period 40 trades were executed and $2,049 in profit was made on the daily timeframe for the EUR/USD pair.

Trade ID	Date	Type	Open Price	Close Price	Profit
1	2020.05.07	Sell	1.07927	1.09775	-171.23
2	2020.05.21	Buy	1.09777	1.11769	178.35
3	2020.07.06	Buy	1.12447	1.18321	522.75
4	2020.08.31	Buy	1.18963	1.18113	-75.24
5	2020.09.22	Sell	1.17641	1.17626	1.33
6	2020.10.12	Buy	1.18087	1.17040	-92.71
7	2020.10.16	Sell	1.17047	1.18561	-134.05
8	2020.10.22	Buy	1.18553	1.17445	-95.60
9	2020.10.30	Sell	1.16716	1.18801	-181.03
10	2020.11.09	Buy	1.18873	1.22052	266.16
11	2021.01.19	Sell	1.20750	1.21168	-35.37
12	2021.02.15	Buy	1.21236	1.20477	-63.75
13	2021.03.02	Sell	1.20489	1.19106	116.16
14	2021.04.15	Buy	1.19787	1.20034	21.11
15	2021.05.10	Buy	1.21598	1.21662	5.40
16	2021.06.14	Sell	1.21063	1.18867	186.15
17	2021.08.06	Sell	1.18343	1.17509	73.63
18	2021.09.01	Buy	1.18084	1.18103	1.69
19	2021.09.16	Sell	1.18161	1.16320	163.96
20	2021.11.02	Sell	1.16041	1.13384	244.72
21	2021.12.17	Sell	1.13270	1.13239	2.99
22	2021.12.31	Buy	1.13179	1.12833	-33.43
23	2022.01.06	Sell	1.13099	1.13656	-53.69
24	2022.01.12	Buy	1.13640	1.13395	-23.39
25	2022.01.20	Sell	1.13390	1.14368	-93.37
26	2022.02.07	Buy	1.14625	1.13243	-129.39
27	2022.02.22	Sell	1.13073	1.10150	274.07
28	2022.04.05	Sell	1.09686	1.06883	277.94
29	2022.05.31	Buy	1.07750	1.06143	-166.36
30	2022.06.10	Sell	1.06146	1.02162	412.41
31	2022.08.16	Sell	1.01573	1.00463	124.58
32	2022.09.15	Sell	0.99758	0.98821	108.25
33	2022.10.10	Sell	0.97321	0.99641	-277.32
34	2022.10.26	Buy	0.99615	0.98729	-101.00
35	2022.11.03	Sell	0.98151	1.00196	-234.50
36	2022.11.08	Buy	1.00175	1.06006	641.47
37	2023.01.10	Buy	1.07273	1.07222	-5.54
38	2023.02.13	Sell	1.06795	1.07323	-57.63
39	2023.03.22	Buy	1.07661	1.09576	206.31
40	2023.05.12	Sell	1.09127	1.06866	244.58

Fig. 10. Transaction log of EUR/USD trading on test sample on the daily timeframe

Initial Deposit: $10,000.00

Total Trades: 40

Total Profit: $2,049.44

Successful Trades: 21 ($4,074.02)

Unsuccessful Trades: 19 ($-2,024.58)

Buy Trades: 18 (Success: 8, Fail: 10, Profit: $1,843.24, Loss: $-786.41)

Sell Trades: 22 (Success: 13, Fail: 9, Profit: $2,230.78, Loss: $-1,238.17)

Profit Factor: 2.01

Max Drawdown: 5.48%

Final Balance: $12,049.44

Fig. 11. General financial report for trading EUR/USD on daily timeframe

Given the overall positive result, as well as the fact that the number of losing and profitable trades is approximately the same, and the profit factor is equal to 2, we may conclude that the system loses money when trying to determine the moment of reversal of the price chart movement, and makes money (many times more than it loses on average) when it determines successfully the moment of reversal.

The calculations in Fig. 11 were obtained for each trade with a volume of 0.1 lot (10,000 units of the base currency). Under these conditions, the profit was approximately 20% over three years relative to the initial deposit of $10,000. The maximum drawdown of only 5.48% over the same period indicates the minimal risks of such a trading system.

If we use a leverage of 1:5 (note that FOREX brokers give a leverage of 1:50, which is the maximum allowed in the US, in other countries 1:100 and more), which is justified to keep the risk relatively small (within a 30% drawdown), than we will earn 100% profit in three years. This is a very good result even for intraday trading, and especially when working on daily timeframe.

For other currency pairs, the results of modelling on the D and H4 timeframes are summarised in Table 1. As can be seen from the table of daily charts (timeframe D), the results of trading on the AUD/USD and USD/JPY pairs are even better than for the EUR/USD pair. However, for the H4 timeframe, we can observe a result close to zero for all instruments except USD/JPY pair. The profitability of this pair is even 2 times higher than on the daily chart, taking into account that the financial result for H4 is given for six months, unlike the daily chart, where the trading period is about 3 years.

Table 1

TOP-20 MOST EFFECTIVE CLUSTERING ALGORITHMS ON THE STUDY DATA

Timeframe	Metric	EUR/USD	AUD/USD	USD/JPY	USD/CAD
D	Net Profit	$2,049.44	$4,137.93	$3,189.54	$987.49
	Max Drawdown	5.48%	4.59%	4.15%	4.96%
	Win Rate	21/40	21/42	15/41	19/46
	Profit Factor	2.01	2.99	3.15	1.5
H4	Net Profit	$83.52	-$27.94	$978.99	-$87.38
	Max Drawdown	1.76%	7.4%	3.23%	4.54%
	Win Rate	17/41	17/46	18/45	17/44
	Profit Factor	1.12	0.98	1.86	0.87

Perhaps the overall lower profitability results (even losses) on the 4-hour timeframe can be explained by the transition to intraday dynamics of the time series, which may require adjustments to the underlying model (we will leave this for further research).

It should also be noted, that the entry and exit thresholds in the Table 1 were the same for all instruments and for both timeframes, as it is mentioned in the description for the EUR/USD pair. However, for EUR/USD, the thresholds were chosen experimentally for a specific dataset. We may assume, that they will be different for other datasets. Also, different thresholds may be preferable for Buy and Sell trades.

In this regard, we conducted an experiment for financial instruments on daily charts to study the impact of the threshold values for the Buy action (“Open Buy” and “Close Buy”) in the range from 0.5 to 0.95. The modelling results are shown in Fig. 12.

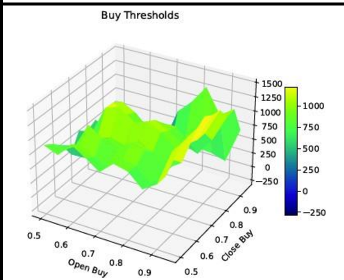
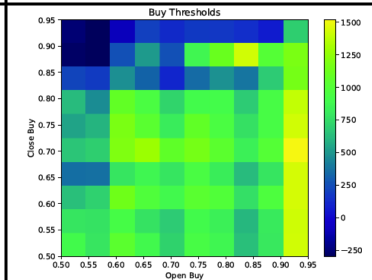
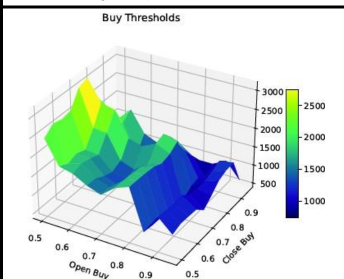
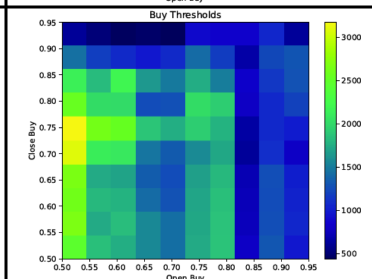
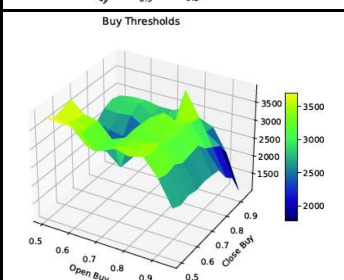
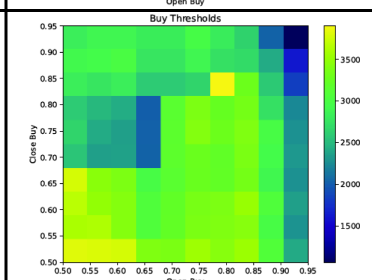
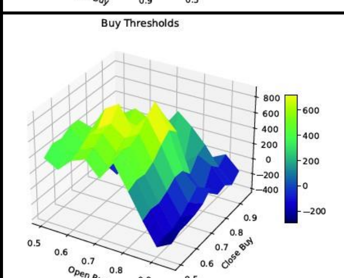
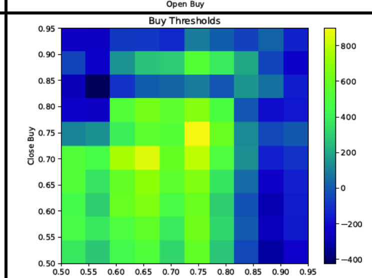
Currency pair	3D surface	Profit heatmap
EUR/USD		
AUD/USD		
USD/JPY		
USD/CAD		

Fig. 12. Study of entry and exit thresholds for Buy positions for four currency pairs on test data for the daily timeframe

The heatmaps in Fig. 12 show in colour the profit levels that the trading system would have demonstrated on the corresponding currency pair at certain intervals of threshold values for opening and closing a Buy position. Let us recall that the thresholds are set not for the price values of the financial instrument, but for the membership functions of the point on the chart, described by a number of features, to the Buy cluster.

For example, the brightest cell on the heatmap for the EUR/USD currency pair, which is located in the rightmost column in the middle and indicates the greatest profit potential when trading this financial instrument, suggests that this result would have been obtained with a threshold range for “Open Buy” from 0.90 to 0.95, and with a threshold for “Close Buy” from 0.68 to 0.73. At the same time, the profit from the system’s operation with such settings will be about $1,500, which can be seen from the values on the ruler to the right of the corresponding heatmap (in the upper right corner of Fig. 12 for the EUR/USD currency pair).

Recall that the profitability of our system exceeded $2,000, which can be seen in Table 1. It is worth noting that we set only the lower thresholds for the membership functions to their clusters, and therefore they cover much more options. Instead, Fig. 12 shows profits for narrow ranges of threshold values. Therefore, it is not a tool for making precise decisions on threshold optimisation, but rather an analytical tool for a general study of the system’s profitability depending on these metaparameters.

In addition, the profitability of the system depends on a number of other parameters – the thresholds for opening and closing a Sell position, as well as all the features. Therefore, the heatmaps show profit even if the threshold for “Close Buy” is higher than the threshold for “Open Buy”.

Comparing the heatmaps in Fig. 12, we can confirm the earlier conclusion that the trading profitability of each financial instrument varies greatly depending on the preset parameters for decision making. Thus, the model performance can be further improved by choosing the best pairs of entry and exit thresholds for each instrument and timeframe.

Conclusions

This study successfully applied the Fuzzy C-Means clustering method to analyse time series, particularly financial instruments. The decision-making system developed using this approach showed good results in identifying trend reversals, which contributed to more efficient buying and selling decisions.

The integration of cluster modelling with traditional technical analysis tools proved to be particularly useful when dealing with complex financial time series. The combination of these methods allowed us to more effectively identify non-linear patterns and process large data sets, which increased the accuracy of trend forecasting.

In particular, testing this approach on the AUD/USD currency pair for 3 years of trading on the daily timeframe demonstrated a profitability of 40%. At the same time, the article shows how to use leverage to further increase profits while remaining within the acceptable risk.

The ability of the models to handle overlapping clusters corresponding to different patterns in price dynamics and to cope with uncertainty about future development of the market makes them highly adaptable to various forecasting tasks in different market conditions. The approach proposed in this study is universal and can be applied to a wide range of financial instruments such as stocks, commodities, futures, etc., and to different timeframes. The importance of using financially-oriented metrics (Net Profit, Max Drawdown, Win Rate, Profit Factor) for assessing the effectiveness of the decision-making system and their advantages over traditional forecasting accuracy metrics are argued.

Although the methodology is self-sufficient for making decisions on financial time series, it is important to optimise the thresholds of membership to clusters for each financial instrument, corresponding to decisions on entering and exiting a position in rising and falling markets. There is also potential in optimising the calculation parameters of technical analysis indicators MACD1, MACD2, S%K, S%D and expanding their number.

First of all, the authors plan to apply filtering of the signals obtained using indicators on higher timeframes, determining permissions for trading directions based on the golden cross method or author's approach [8], which should eliminate transactions against the dominant trend and improve the trading characteristics of the trading system.

References

1. Tsay, R. S. (2005). *Analysis of financial time series* (2nd ed.). John Wiley & Sons, Inc,
2. Brockwel, P. J., Davis, R. A. (2016). *Introduction to Time Series and Forecasting* (3rd ed.). Springer Cham. https://doi.org/10.1007/978-3-319-29854-2
3. Matviychuk, A. (2006). Fuzzy logic approach to identification and forecasting of financial time series using Elliott wave theory. *Fuzzy economic review, 11*(2), 51-68. https://doi.org/10.25102/fer.2006.02.04
4. Achelis, S. B. (2013). *Technical Analysis from A to Z* (2nd ed.). McGraw Hill.
5. Elder, A. (1993). *Trading for a Living: Psychology, Trading Tactics, Money Management*. Wiley.
6. Kuchansky, A., Nikolenko, V., & Rachenko, A. (2015). A method for identifying trends in financial time series based on trend models of forecasting. *Management of Development of Complex Systems,_24*, 84-89. https://urss.knuba.edu.ua/ua/zbirnyk-24/article-862 [in Ukrainian]
7. Wilson, S. J. (2016). *Trend Detection and Pattern Recognition in Financial Time Series* [Doctoral dissertation, George Mason University]. MARS. https://doi.org/10.13021/MARS/7317
8. Bakai, Ye. I., Kabachii, V. V., & Maslii, R. V. (2017). Decision-making Model on Financial Time Series Based on a Couple of Moving Averages Using the Assessment of Different Timeframes. *Visnyk of Vinnytsia Polytechnical Institute, 1*, 70-77. https://visnyk.vntu.edu.ua/index.php/visnyk/article/view/2014 [in Ukrainian]
9. Dash, T., Chitlangia, S., Ahuja, A., & Srinivasan, A. (2021). *Incorporating domain knowledge into deep neural networks*. arXiv. https://doi.org/10.48550/arXiv.2103.00180
10. Borghesi, A., Baldo, F., & Milano, M. (2020). *Improving Deep Learning Models via Constraint-Based Domain Knowledge: A Brief Survey*. arXiv. https://doi.org/10.48550/arXiv.2005.10691
11. Kozlovskyi, S., Mazur, H., Vdovenko, N., Shepel, T., & Kozlovskyi, V. (2018). Modeling and forecasting the level of state stimulation of agricultural production in Ukraine based on the theory of fuzzy logic. *Montenegrin Journal of Economics, 14*(3), 37-53. https://doi.org/10.14254/1800-5845/2018.14-3.3

12. Bielinskyi, A., Soloviev, V., Solovieva, V., & Velykoivanenko, H. (2022). Fuzzy time series forecasting using semantic artificial intelligence tools. *Neuro-Fuzzy Modeling Techniques in Economics, 11*, 157-198. http://doi.org/10.33111/nfmte.2022.157

13. Nair, B. B., Kumar, P. S., Sakthivel, N. R., & Vipin, U. (2017). Clustering stock price time series data to generate stock trading recommendations: An empirical study. *Expert Systems with Applications, 70*, 20-36. https://doi.org/10.1016/j.eswa.2016.11.002

14. Kaminskyi, A., Miroshnychenko, I., & Pysanets, K. (2019). Risk and return for cryptocurrencies as alternative investment: Kohonen maps clustering. *Neuro-Fuzzy Modeling Techniques in Economics, 8*, 175-193. http://doi.org/10.33111/nfmte.2019.175

15. Kononova, K., & Dek, A. (2018). Investigation of traders' behavioral characteristics: experimental economics methods and machine learning technologies. *Neuro-Fuzzy Modeling Techniques in Economics*, *7*, 148-167. http://doi.org/10.33111/nfmte.2018.148

16. Matviychuk, A., Strelchenko, I., Vashchaiev, S., & Velykoivanenko, H. (2019). Simulation of the Crisis Contagion Process between Countries with Different Levels of Socio-Economic Development. *CEUR Workshop Proceedings, 2393*, 485-496. http://ceur-ws.org/Vol-2393/paper_423.pdf

17. Enke, D., Grauer, M., & Mehdiyev, N. (2011). Stock market prediction with multiple regression, fuzzy type-2 clustering and neural networks. *Procedia Computer Science*, *6*, 201-206. https://doi.org/10.1016/j.procs.2011.08.038

18. Mehmanpazir, F., & Asadi, S. (2017). Development of an evolutionary fuzzy expert system for estimating future behavior of stock price. *Journal of Industrial Engineering International*, *13*, 29-46. https://doi.org/10.1007/s40092-016-0165-7

19. Shi, Y., Li, B., Du, G., & Dai, W. (2021). Clustering framework based on multi-scale analysis of intraday financial time series. *Physica A: Statistical Mechanics and its Applications*, *567*, Article 125728. https://doi.org/10.1016/j.physa.2020.125728

20. Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert systems with applications*, *140*, Article 112896. https://doi.org/10.1016/j.eswa.2019.112896

21. Long, J., Chen, Z., He, W., Wu, T., & Ren, J. (2020). An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in Chinese stock exchange market. *Applied Soft Computing*, *91*, Article 106205. https://doi.org/10.1016/j.asoc.2020.106205

22. Wang, X., Yang, K., & Liu, T. (2021). Stock Price Prediction Based on Morphological Similarity Clustering and Hierarchical Temporal Memory. *IEEE Access*, *9*, 67241-67248. https://doi.org/10.1109/ACCESS.2021.3077004

23. Ozer, F., & Okan Sakar, C. (2022). An automated cryptocurrency trading system based on the detection of unusual price movements with a time-series clustering-based approach. *Expert Systems with Applications*, *200*, Article 117017. https://doi.org/10.1016/j.eswa.2022.117017

24. Sáenz, J. V., Quiroga, F. M., & Bariviera, A. F. (2023). Data vs. information: Using clustering techniques to enhance stock returns forecasting. *International Review of Financial Analysis*, *88*, Article 102657. https://doi.org/10.1016/j.irfa.2023.102657

25. D'Urso, P., De Giovanni, L., & Massari, R. (2021). Trimmed fuzzy clustering of financial time series based on dynamic time warping. *Annals of operations research*, *299*, 1379-1395. https://doi.org/10.1007/s10479-019-03284-1

26. Hartigan, J.A., & Wong, M.A. (1979). A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C*, *28*(1), 100-108. https://doi.org/10.2307/2346830

27. Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, *3*(3), 32–57. https://doi.org/10.1080/01969727308546046

28. Goddard, J., de los Cobos Silva, S.G., & Gutiérrez Andrade, M.A. (2006). Comparison of k-means and fuzzy c-means using background knowledge. *Fuzzy economic review*, *11*(2), 3-16. https://doi.org/10.25102/fer.2006.02.01

29. Miyamoto, S., Ichihashi, H., & Honda, K. (2008). *Algorithms for Fuzzy Clustering. Methods in c-Means Clustering with Applications*. Springer. https://doi.org/10.1007/978-3-540-78737-2

30. MetaTrader 4. (2023, April 5). *MetaTrader 4 – the most popular Forex trading platform*. https://www.metatrader4.com/en

31. Murphy, J.J. (1999). *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance.

32. Haryono, A. T., Sarno, R., & Sungkono, K. R. (2023). Transformer-Gated Recurrent Unit Method for Predicting Stock Price Based on News Sentiments and Technical Indicators. *IEEE Access*, *11*, 77132-77146. https://doi.org/10.1109/ACCESS.2023.3298445

The article was submitted on 2023, August 28