Knowledge and Information Systems (2019) 61:397–429
https://doi.org/10.1007/s10115-018-1303-x

REGULAR PAPER

# Short-term trend prediction in financial time series data

**Mustafa Onur Özorhan**[1] · **İsmail Hakkı Toroslu**[2] · **Onur Tolga Şehitoğlu**[2]

Received: 12 April 2017 / Revised: 6 November 2018 / Accepted: 24 November 2018 /
Published online: 13 December 2018


**Abstract**
This paper presents a method to predict short-term trends in financial time series data found in the foreign exchange market. Trends in the Forex market appear with similar chart patterns. We approach the chart patterns in the financial markets from a discovery of motifs in a time series perspective. Our method uses a modified Zigzag technical indicator to segment the data and discover motifs, expectation maximization to cluster the motifs and support vector machines to classify the motifs and predict accurate trading parameters for the identified motifs. The available input data are adapted to each trading time frame with a sliding window. The accuracy of the prediction models is tested across several different currency pairs, spanning 5 years of historical data from 2010 to 2015. The experimental results suggest that using the Zigzag technical indicator to discover motifs that identify short-term trends in financial data results in a high prediction accuracy and trade profits.

**Keywords** Short-term trend prediction · Forex forecasting · Support vector machines · Expectation maximization · Zigzag technical indicator · Motifs

## 1 Introduction

An exchange rate is the value of a country's currency expressed in terms of other countries' currencies. Many currencies are freely floating since the 1971 Smithsonian Agreement [43] and do not have a fixed value that is pegged by the central bank of a country. The value of currencies is determined in the foreign exchange market (Forex). Forex market is an extensive trading ground for traders across the world. It is available for trade 24 h a day, 5 days a week. The trade volume per day is in excess of 4 trillion USD. Many different bilateral currency pairs are traded in the Forex market, the most popular being: USD, EUR, GBP, JPY and CHF. 90.95% of all the trades taking place in the Forex market include the aforementioned currencies [28]. In the Forex market, a trader can profit from predicting the direction and magnitude of price fluctuations of a currency pair. Using a leverage value, it is possible to multiply wins and losses.

✉ Mustafa Onur Özorhan
onur.ozorhan@gmail.com

1 Central Bank of Turkey, Ankara, Turkey

2 Department of Computer Engineering, METU, Ankara, Turkey

In this paper, we address the problem of predicting the short-term trend for financial time series using time series data mining and machine learning techniques. The predictions should be applicable in real world, since trading concepts such as spreads, swap commissions and leverages are taken into account.

The short-term trends for a financial instrument are detected by exploiting lagging technical indicators in a motif discovery approach. A modified version of the Zigzag lagging technical indicator is used in our algorithm. The details of the indicator are discussed later in Sect. 5. In this work, novel motifs are discovered and compared with legacy chart patterns, such as the "Head and Shoulders Top" and "Head and Shoulders Bottom (Inverse Head and Shoulders)."

The contributions of this paper can be summarized in three categories: (1) a new method for discovering, clustering, classifying and segmenting subsequences of financial time series, (2) a modified version of the Zigzag technical indicator that performs better than the original indicator in a chart pattern detection scenario, (3) a real-world trading-focused learning approach for using the detected motifs as chart patterns.

Thus, the main contribution of this paper is exploiting and extending Zigzag technical indicator in a novel way to generate new patterns and new signals from it. Our results show the success of this idea through detailed experiments.

In terms of discovering the trends, current methodology is the observation of charts via chartists. In our work, raw price data are used in combination with Zigzag technical indicator signals and these data are used to discover motifs to predict future time series. In terms of real-world trading-focused learning approach, the study's contribution is employing a stop-loss, take-profit[1] and trade-window-based trading in the Forex market rather than simply trying to forecast the direction of the trend. Previous approaches try to forecast the direction of movement in the exchange rate but not the magnitude of the desired or undesired movement. In our approach, we are assessing the possible direction and magnitude of movements in the currency's future in both directions to determine trading parameters such as stop loss, take profit, time stop and lot size. (The amount of the value invested in the trade and standard lot corresponds to 100.000 units of the base currency, and various fractions of 1 lot can also be used in the trade.)

The rest of the paper is organized as follows: In Sect. 2, notable previous works are mentioned, in Sect. 3 necessary background information regarding time series is given, and in Sect. 4 necessary background information regarding the Forex market is given. In Sect. 5, overview of the system and the algorithm used for forecasting are presented. In Sect. 6, the performance of the system is discussed. Section 8 concludes the paper including possible future work.

## 2 Previous work

Various techniques are employed to analyze financial time series. Most of these techniques focus on the predicting the future direction of movement of a financial instrument. If predicted direction and actual direction of the financial instrument are the same, then the prediction is considered to be directionally symmetric.

Kamruzzaman and Sarker [18] use ANNs to forecast AUD against six major currencies (USD, GBP, JPY, SGD, NZD and CHF). The study experiments with three different types of ANNs: Standard Back Propagation, Scaled Conjugate Gradient and Bayesian Regularization.

[1] Please refer to the "Appendix A.1" for the definitions of Forex terms take profit and stop loss.

A number of different models are trained for each type of ANN, with varying initial weights, hidden unit layers and iteration counts. Varying intervals of Moving Average values and last week's closing values are fed to the networks as inputs. Results show that all three models outperform standalone ARIMA model and Scaled Conjugate Gradient is better than Standard Back Propagation. At all Forex categories, Scaled Conjugate Gradient and Standard Back Propagation models achieve a directional symmetry of at least 77%.

Lu et al. [27] use a two stage modeling approach for forecasting. In the first stage, independent component analysis of forecasting variables is made and noisy components are eliminated. In the second stage, remaining components are used for training an SVM. A remarkable directional symmetry of 86% is achieved in the tests for forecasting the Nikkei 225 index. However, the same approach performs at a 60.15% directional symmetry for the TAIEX index. The results show that model may not be applicable to different types of financial time series.

Korol [22] uses fuzzy logic and a combination of fundamental and technical analysis to forecast exchange rates. Fundamental economic factors such as relative interest rate between countries, GDP growth rates and credit ratings are considered. The model's mean absolute error rates are lower than models solely based on technical indicators.

Stella and Villa [46] use a Bayesian Network Classifier and use a continuous history window of data to train the classifier. The data are integrated from different Forex data providers, and moving averages are used to train the classifier. The one-year-long experiment achieves accuracy values ranging from 0.5468 to 0.8479 in a variety of currency pairs.

Yao and Tan [47] use an ANN to model USD's direction against four other major currencies (JPY, GBP, CHF and AUD). The prediction is made based on weekly closing prices. This method achieves directional symmetry values of up to 55%. It is also stated that a trained model can continue its successful performance for half a year in a weekly forecasting scheme. Longer periods of forecasting require retraining of the model.

Deep belief networks and the conjugate gradient method are to forecast exchange rates by Shen et al. [42]. British Pound/US Dollar (GBP/USD), Brazilian Real/US Dollar (BRL/USD) and Indian Rupee/US Dollar (INR/USD) are forecasted in weekly intervals. Different data intervals ranging from 1976 to 2004 are used for different currencies. A directional accuracy of 63.62% is achieved for GBP/USD on 885 training and 52 testing data points. The maximum accuracies achieved for INR/USD and BRL/USD are 58.73% and 45.09%, respectively.

Moosa and Burns [29] use conventional macroeconomic models to forecast exchange rates in the long term. The method tries to outperform random walk in out-of-sample forecast, in terms of prediction's directional symmetry. Canadian Dollar/US Dollar (CAD/USD), British Pound/Canadian Dollar (GBP/CAD), GBP/USD, Japanese Yen/Canadian Dollar (JPY/CAD) and Japanese Yen/British Pound (JPY/GBP) currency pairs are used in the experiments. Monthly data covering 1997–2013 are used for all currencies. Highest recorded accuracy is 72%, when forecasting is made for every six months and 12 data points are used for GBP/USD. For the same currency, quarterly forecasts have an accuracy of 56% and monthly forecasts have an accuracy of 48%. Overall lowest recorded accuracy is 38% when forecasting is made for every six months and 12 data points are used for JPY/USD.

Zhang et.al.'s work (State Frequency Memory or SFM in short) [49] uses deep learning approach for stock market prediction. SFM is a variant of well-known Long Short-Term Memory (LSTM) which was originally proposed in [15]. This work has been applied to predict the price of the 50 stocks with top market capitalization values. Experiments show that for 1, 3 and 5 steps (days) predictions, SFM reduces the error significantly compared to LSTM.

Ozorhan et al. [34] use SVMs and genetic algorithms to determine relative strength of a currency against a pool of currencies with the help of technical indicator signals. Forecasts are made in the daily interval using the predicted strengths of currencies. The accuracy of the prediction models is tested across several different sets of technical indicators and currency pair sets, spanning 5 years of historical data from 2010 to 2015. In a currency pool of EUR/GBP/USD and CHF, accuracy of forecasts is as high as 78%. Highest accuracy for single currency pair trading is achieved in USD/CHF pair and is 64%.

The following table summarizes the cited work and their methods.

| Work | Preprocess | Learning method | Outcome |
|---|---|---|---|
| [18] | | ANN | Forecast |
| [27] | Component analysis | SVM | Directional symmetry |
| [22] | | Fuzzy logic | Forecast |
| [46] | History window | Bayesian network | Directional symmetry |
| [47] | | ANN | Directional symmetry |
| [42] | | Deep belief network | Forecast |
| [29] | | Conventional macroeconomic models | forecast |
| [49] | | LSTM | Forecast |
| [34] | | SVM + GA | Currency strength |
| ZZMOP | Zigzag extraction | Clustering + SVM | Trading parameters |

In addition, there are huge amount of works on time series analysis in general. Some of these, such as clustering, segmentation, have also been utilized in this paper.

There are several time series clustering approaches; however, most clustering techniques require parameter optimization based on individual series data and are incompatible with multivariate time series. Denton et al. [10] propose a pattern-based time series subsequence clustering approach which uses radial distribution functions. Rakthanmanon et al. [40] propose an approach which includes both single and multivariate clustering based on minimum description length. In our approach, we use a pattern-based approach which segments and transforms a multivariate time series with expectation maximization.

There are also stochastic methods applied to various data domains of time series analysis like temporal autocorrelation [26,50]. Autocorrelation in time series corresponds to re-occurrence of similar signal pattern after some time interval. This approach has also been applied to financial data such as stocks and Forex [5,6,14,25]. Since our paper focuses on learning chart patterns of financial data rather than analyzing statistical properties for trend estimation, we have used a more popular tool called motifs.

A motif [30] is a subsequence of a longer time series which appears recurrently. Several motifs can exist within a single series, and motifs can be of varying lengths and might overlap. Exhaustively determining motifs in a time series requires subsequences to be compared against other subsequences using a similarity measure, to assure recurrent behavior.

Subsequence clustering rarely produces meaningful results. Thus, motif discovery is used to address time series problems such as anomaly detection and time series forecasting. There are several motif discovery approaches for streaming time series. Anh and Son [44] use R* tree algorithm and dimensionality reduction to discover motifs. Nguyen et al. [33] propose an algorithm based on a nearest neighbor classifier to discover motifs in a single scan of the time series. In our approach, we use a Zigzag-based algorithm to discover motifs of different lengths in the streaming time series.

Segmentation creates an approximation of the time series by reducing the dimensionality of the data. The reduction should accurately approximate the series by retaining the essential features. Segmentation should minimize the reconstruction error between the reduced representation and the original time series. There are sliding window-based approaches, top-down approaches and bottom-up approaches to segmentation of time series [19].

## 3 Time series preliminaries

A time series is the collection of values that are obtained from sequential measurements over a specific period of time. Time series analysis tries to visualize the characteristics of data. The mining, classification and forecasting of time series face numerous difficulties. Most frequently, these difficulties arise from the high dimensionality and the large volume of the data.

The observations in a time series are collected from measurements taken at uniformly spaced time *instants* which results in a fixed *sampling rate*. The time series can be *univariate* or it can be *multivariate*. A *multivariate* time series spans multiple dimensions of data within the same time range.

Time series may have a fixed length or they might be streaming, in which case time instants continuously feed and grow the series. These types of time series are referred to as semi-infinite time series. Semi-infinite time series can be processed in a streaming manner or subsequences of it can be considered.

Time series mining algorithms try to represent the similarity between two time series with similarity measures. Similarity between time series is usually represented from a distance perspective.

Financial time series are multivariate and semi-infinite. Similarity is usually measured between subsequences extracted from a single or multiple time series using a similarity measure such as the distance measure.

Clustering finds groups or *clusters* in a given dataset. Clustering tries to create clusters containing data that are homogeneous, while clusters themselves are as distinct as possible from each other. Clustering minimizes intracluster variance and maximizes intercluster variance.

There are different types of time series clustering approaches. In financial time series, subsequence clustering is generally applied. In this approach, clusters are created by extracting subsequences from a single or multiple time series.

**Definition 3.1** (*Time series subsequence clustering*)—Given a time series $\mathbf{T} = \langle o_1, o_2, \ldots, o_n \rangle$ of length $n$, and a similarity measure $D(\mathbf{T_1}, \mathbf{T_2})$, *subsequence clustering* finds $C$, the set of clusters $c_i = \left\{ \mathbf{T'_j} \mid \mathbf{T'_j} \in \mathbf{S^m_T} \right\}$, where $\mathbf{S^m_T}$ represents all subsequences of $\mathbf{T}$ with length $\mathbf{m}$ and $c_i$ is a set of subsequences that maximizes intercluster variance and intracluster cohesion.

There are several time series clustering approaches; however, most clustering techniques require parameter optimization based on individual series data and are incompatible with multivariate time series. Denton et al. [10] propose a pattern-based time series subsequence clustering approach which uses radial distribution functions. Rakthanmanon et al. [40] propose an approach which includes both single and multivariate clustering based on minimum description length. In our approach, we use a pattern-based approach which segments and transforms a multivariate time series with expectation maximization.

Segmentation creates an approximation of the time series by reducing the dimensionality of the data. The reduction should accurately approximate the series by retaining the essential features.

**Definition 3.2** (*Time Series Segmentation*)—Given a time series $\mathbf{T} = \langle o_1, o_2, \ldots, o_n \rangle$ of length $n$, segmentation constructs a model **T'** such that dimensionality of **T'** is less than the dimensionality of **T** such that $d(\mathbf{T'}) \leq d(\mathbf{T})$ and **T'** approximates **T** with an error threshold $e$ for a reconstruction function $R$ where $D(R(\mathbf{T'}), \mathbf{T}) < e$.

Segmentation should minimize the reconstruction error between the reduced representation and the original time series. There are sliding window-based approaches, top-down approaches and bottom-up approaches to segmentation of time series.

A motif [30] is a subsequence of a longer time series which appears recurrently. Several motifs can exist within a single series, and motifs can be of varying lengths and might overlap. Exhaustively determining motifs in a time series requires subsequences to be compared against other subsequences using a similarity measure, to assure recurrent behavior. To the best of our knowledge, the motif concept was first defined for the time series data in [38]. Below, we make a very similar formal definition of motif.

**Definition 3.3** (*Motif*)—Given a time series $\mathbf{T} = \langle o_1, o_2, \ldots, o_n \rangle$ of length $n$, a motif **M** is a set of time series subsequences of **T** of length $m$, $\mathbf{M} = \left\{\mathbf{TS_i} \mid \mathbf{TS_i} \in \mathbf{S_T^m}\right\}$, and $\forall \mathbf{TS_i}, \mathbf{TS_j}$ : $D(\mathbf{TS_j}, \mathbf{TS_i}) < e \wedge i \neq j$ holds true for a predefined error $e$ where $D(\mathbf{TS_i}, \mathbf{TS_j})$ is the similarity measure between two time.

Subsequence clustering rarely produces meaningful results. Thus, motif discovery is used to address time series problems such as anomaly detection and time series forecasting. There are several motif discovery approaches for streaming time series. Anh and Son [44] use R* tree algorithm and dimensionality reduction to discover motifs. Nguyen et al. [33] propose an algorithm based on a nearest neighbor classifier to discover motifs in a single scan of the time series. In our approach, we define a segmentation method to convert time series data into *Zigzag indicator-based* sequences, then cluster and discover motifs of different lengths in these Zigzag sequences.

## 4 Forex preliminaries

### 4.1 Forex

Forex is a global and decentralized market for trading currencies. It is continuously operational except weekends. The value of exchange rates is determined based on market supply and demand. Supply and demand are further determined by political conditions, market psychology and a variety of fundamental economic factors.

New data are generated with every transaction in the Forex market. A transaction in the Forex market can take place with the meeting of the buyer and the seller at the same price point. This meeting is a nonzero-sum game [32], due to the presence of a market maker which provides the Forex service under transaction costs such as spreads and swaps. With the collection of the transactions in a given period of time, a summarizing set of data—which is called a bar—is produced. A bar in a time frame contains opening, closing, highest and lowest prices for the given time interval. These are the prices that traders value the most in an interval and are also called as raw price data.

With the use of leverage, profits and losses can be multiplied. Leverages generally result in borrowing costs (swaps). The margin between asking and bidding price is called the spread and is generally very small to allow traders to create high-frequency applications with lower profit margins. For a trading algorithm to be profitable, the sum of profits collected by the algorithm should be higher than sum of losses, borrowing costs and transaction costs.

In Forex terminology, 1 pip corresponds to 4th significant digit after decimal point and 1 pipette corresponds to 5th significant digit in the currencies, and all calculations are done with pipette level.

Please find other commonly used Forex terms that are also used in our system in "Appendix A.1."

## 4.2 Technical indicators, signals, chart patterns and trends

Technical indicators [4] are statistical metrics whose values are calculated from price history of financial instrument. There are two types of technical indicators: lagging indicators and leading indicators. Lagging technical indicators are generated to represent the past behavior of the price. Leading indicators try to predict future behaviors of the price.

Technical indicators capture certain properties of price movements but are available to everyone in a simple, numeric form. In their original form, they are numbers computed from raw price data without any meaning. With experience from the history, traders keep track of what kinds of values generated by the technical indicators can be used to generate successful buy and sell signals and trade accordingly. The rules used in this process are called technical indicator signals. A static set of technical indicators or signals cannot reflect the price changes of a financial instrument indefinitely. Therefore, both technical indicators and technical indicator signals are updated continuously.

Chart patterns are two-dimensional formations that appear on a financial instrument's price chart. Chartists and traders use these chart patterns to identify trends for the instrument to trigger buy and sell signals. There are various categories of chart patterns, such as reversal chart patterns and continuation chart patterns. Reversal chart patterns appear at the end of previous trends and are followed by opposite price action, and continuation chart patterns are intermediate consolidation areas in existing trends. A sample of reversal chart pattern and a continuation chart pattern is shown in Fig. 1a, b, respectively.

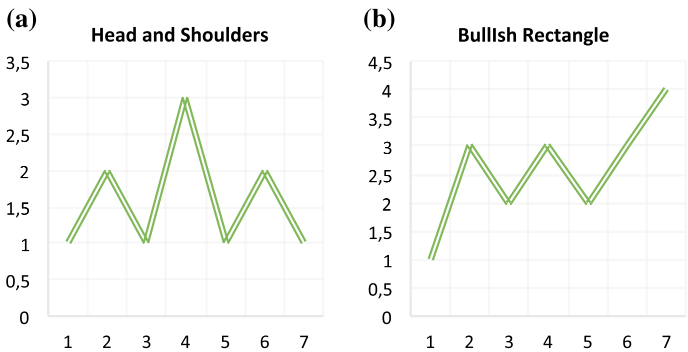


**Fig. 1** **a** Head and shoulders—reversal pattern, **b** Bullish rectangle—continuation pattern

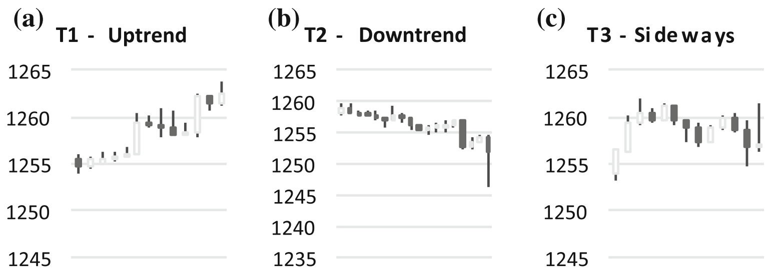


**Fig. 2** **a** Uptrend, **b** Downtrend and **c** Sideways trend at commodity XAU/USD

Financial markets, including the Forex market, have three states: uptrends, downtrends and sideways trends [35]. A sample for each of these states is shown in Fig. 2 for a sample Forex traded commodity (i.e., XAU/USD).

As specified in [16,23], predicting the trend of a financial instrument is not only more important but also easier than predicting the price at each time interval.

### 4.3 Zigzag indicator

Zigzag indicator is a lagging technical indicator for financial time series data. It does not make predictions regarding the future values of a financial instrument. It is used to highlight the significant highs and lows of the instrument's historic values and eliminates the noise in the data. From a time series perspective, Zigzag performs time series segmentation as described in Definition 3.2. In our work, we use the Zigzag technical indicator to detect legacy and novel motifs and create technical indicator signals that determine the trends in the Forex market.

**Definition 4.1** (*Zigzag Indicator*)—Given time series $\mathbf{T} = \langle t_1, t_2, \ldots, t_n \rangle$ $(t_i \in N^+)$ Zigzag $\mathbf{Z}$ is defined as follows:

1. $\mathbf{Z} = \langle z_1, z_2, \ldots, z_n \rangle$
2. $z_i = t_i$ if point is selected as *Zigzag point*
3. $z_i = *$ is the linear interpolation of $z_{i-}$ and $z_{i+}$, the preceding and succeeding *Zigzag points* of $z_i$, if point is not selected as a *Zigzag point*
4. $\forall$ *Zigzag point* $z_i$ of $\mathbf{Z}$ : $(z_i > z_{i-} \wedge z_i > z_{i+})$ or $(z_i < z_{i-} \wedge z_i < z_{i+})$

Depth and deviation are two important user-defined parameters to the Zigzag indicator, which are used to determine how much data will be filtered and how frequently the indicator will make adjustments to its previous values.

**Definition 4.2** (*Zigzag depth*)—The depth is a user-specified value that determines the number of bars in which a bar has to be the extremum bar to be qualified as a *Zigzag point*. *Zigzag points* satisfy the following for the depth parameter:

1. $\mathbf{T} = \langle t_1, t_2, \ldots, t_n \rangle$ $(t_i \in N^+)$
2. $\mathbf{Z} = \langle z_1, z_2, \ldots, z_n \rangle$
3. $\forall$ *Zigzag point* $z_i \in \mathbf{Z}$ : $z_i = t_j \rightarrow (\forall t_x \in \{t_{j-depth}..t_{j+depth}\}\ ,\ z_i \leq t_x)$ or $(\forall t_x \in \{t_{j-depth}..t_{j+depth}\}\ ,\ z_i \geq t_x)$

**Definition 4.3** (*Zigzag deviation*)—The deviation is a user-defined value of the Zigzag indicator, which represents the number of pip points that are required to establish a new low after a high *Zigzag point*, or a new high after a low *Zigzag point*. The new bar's value should deviate from the previous high or low value by at least the deviation amount. *Zigzag points* satisfy the following for the deviation parameter:

1. $\mathbf{T} = \langle t_1, t_2, \ldots, t_n \rangle$ $(t_i \in N^+)$
2. $\mathbf{Z} = \langle z_1, z_2, \ldots, z_n \rangle$
3. $\forall$ *Zigzag point* $z_i$ of $\mathbf{Z}$ : $(z_{i-1} + deviation \leq z_i \wedge z_i \geq z_{i+1} + deviation)$ or $(z_{i-1} \leq z_i + deviation \wedge z_i + deviation \geq z_{i+1})$

Zigzag depth determines the minimum required number of bars for a new top or bottom to be formed. This is an integer number denoting the number of time series data points to examine. When this number is small potentially more frequent, top/bottom points can be obtained as long as the deviation requirement is also met. Zigzag deviation, on the other hand, enforces the minimum required difference between a new bottom/top and an old bottom/top for the new bottom/top to be formed. Therefore, it describes the height between top and bottom points. In our work, these parameters are determined through experiments as it is explained in the next section.

In the stock exchange, deviation is a percentage, in the Forex market deviation is an amount of pips. Higher depth and deviation values would result in a lower number of Zigzag points; hence, more noise would be filtered. When selecting parameters for the Zigzag indicator, depth and deviation should be high enough to ensure noise is filtered but should be low enough to detect significant movements in instrument's price.

In the Forex environment, the values generated by the Zigzag indicator can be used in conjunction with different trading techniques such as Elliott waves [3], Fibonacci retracements [13] and chart patterns. In this work, we use the Zigzag indicator to determine similarities in historic financial time series data in the form of motifs. Algorithm 1 describes the Zigzag algorithm.

**Algorithm 1** Zigzag $(S, E, c)$

**Require:** $S \leftarrow$ start date, $E \leftarrow$ end date, $c \leftarrow$currency time series, $zz_{deviation} \leftarrow$Zigzag deviation
**Ensure:** Output the list of Zigzag values $l_{zz}$ between $S$ and $E$

$zz_{high} \leftarrow S$, $zz_{low} \leftarrow S$
**for all** $d \in \langle S...E \rangle$ **do**
  {case: previous Zigzag point is a high point:}
  **if** $c[d] > c[zz_{high}]$ **then**
    {The last Zigzag point is a trailing point. When higher values are encountered after a Zigzag high point, the Zigzag high point is updated.}
    $zz_{high} \leftarrow d$
    {When there is enough deviation between the previous Zigzag high and low points a new Zigzag point is generated.}
    **if** $c[zz_{high}] - c[zz_{low}] > zz_{deviation}$ **then**
      $l_{zz}.append(zz_{low})$
  {case: previous Zigzag point is a low point:}
  **if** $c[d] < c[zz_{low}]$ **then**
    {The last Zigzag point is a trailing point. When lower values are encountered after a Zigzag low point, the Zigzag low point is updated.}
    $zz_{low} \leftarrow d$
    {When there is enough deviation between the previous Zigzag high and low points a new Zigzag point is generated.}
    **if** $c[zz_{high}] - c[zz_{low}] > zz_{deviation}$ **then**
      $l_{zz}.append(zz_{high})$
**return** $l_{zz}$

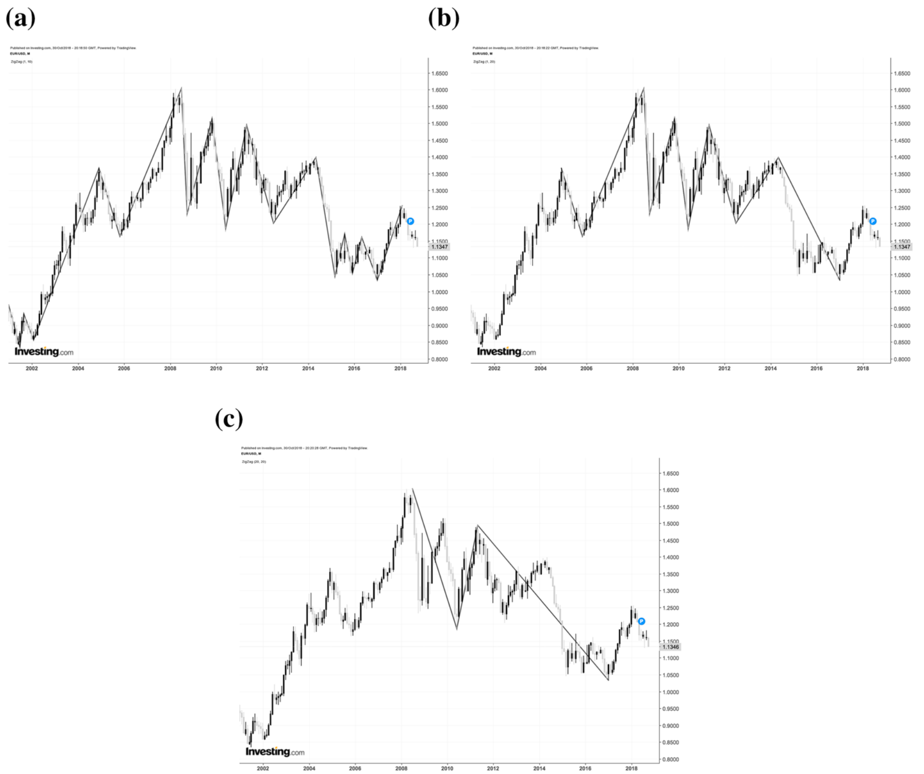


**Fig. 3** Zigzag with **a** depth = 10, deviation = 1%, **b** depth = 20, deviation = 1%, **c** depth = 20, deviation = 20%

Figure 3a–c show the results of a variety of different Zigzag parameters being applied to the same financial time series data.

## 5 Short-term trend prediction in financial time series data with motifs using Zigzag

Our short-term trend prediction method (Zigzag Motif Predictor or ZZMOP in short) has two phases. The first phase, which is called as ZZMOP Model Construction Algorithm, uses a modified (thick) Zigzag indicator to segment historical time series to possible motifs, clusters the motifs and defines reward/risk ratio criteria for the clusters, classifies the motifs in clusters to learn trading rules for the created clusters. The flow of these operations is shown in Fig. 4.

Once ZZMOP Model Constructor constructs models required for trading, in the second phase, our trading algorithm, called as ZZMOP Trading Algorithm, uses these models to


**Fig. 4** Flowchart illustrating algorithmic flow of ZZMOP model construction

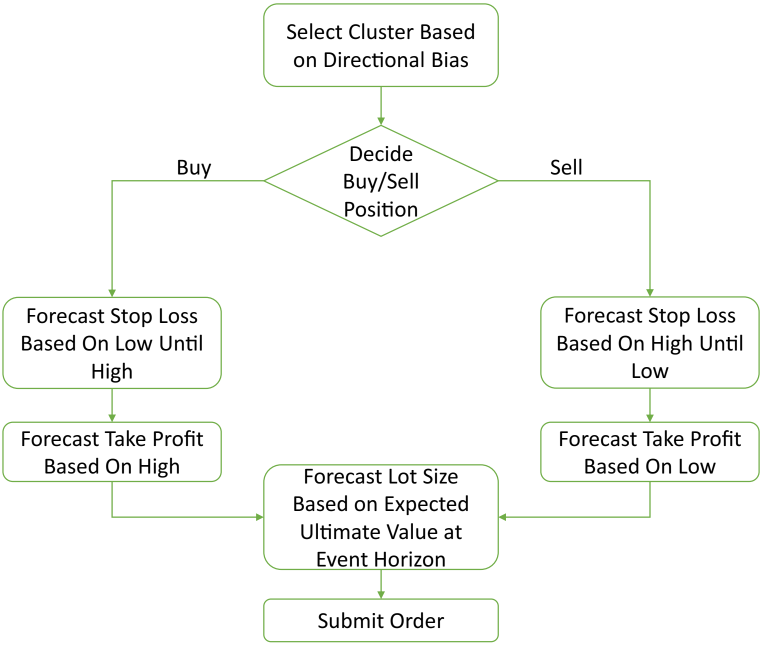


**Fig. 5** Flowchart illustrating ZZMOP trading algorithm

learn trading parameters and submit trade orders. The flow of these operations is shown in Fig. 5.

The trades made by our machine are not simply buying or selling a single financial instrument, rather it is taking a position with a calculated risk and reward. The amount of risk or reward is also multiplied or divided based on the position's future value. Whenever the predicted change in the value is extreme, the lot size of the order is increased accordingly and vice versa.

The motifs recognized during model construction by the system are clustered using the clusterer in the model construction layer. Once clustering is done, SVM's are used to create trading parameters optimal for the associated cluster. Based on classification results, trade orders are submitted to the trade engine. Trades are implicitly closed once they hit stop-loss or take-profit values. If a trade does not hit a stop loss or take profit until motif's event horizon, trade is closed explicitly by the trading layer. In our system, we accept the stop-loss and take-profit values that are forecasted based on SVM classifications. The order sizes are also scaled based on SVM's forecasts.

**Definition 5.1** (*Motif Event Horizon*)—Given a streaming time series $\mathbf{T} = \langle o_1, o_2, \ldots \rangle$, a motif $\mathbf{M}$ of length $n$ with the signal point $o_i$ (please refer to "Appendix A.1." for the definition of signal point), event horizon of $\mathbf{M}$ , $\mathbf{E_M}$ is a subsequence of $\mathbf{T}$ length $n$ consisting of contiguous time instants from $\mathbf{T}$ such as $\mathbf{E_M} = \langle o_{i+1}, o_{i+2}, \ldots, o_{i+n} \rangle$. A motif is characterized by the definitive Zigzag points that it contains; however, due to Zigzag's nature motifs with same number of Zigzag points can be of varying lengths. The length of the event horizon for a motif is equal to the number of data points in the motif.

### 5.1 ZZMOP model construction algorithm

The system we propose has a layered architecture: The first one is the *data collection* layer, second layer is the *calibration layer*, and third layer is the *model construction and trading layer*. The architecture is similar to the systems created in [17,20,39]. The first layer is the data collection layer, and it collects both tick-by-tick and 15 min interval data for currencies of interest. The second layer is the calibration layer. In this layer, indicator parameters, clustering parameters and data sanitization parameters are calibrated with historical data. Five-year historic data are retrieved from data collection layer and separated into disjoint sets per currency pair to discover new motifs and cluster the motifs. In the third layer, historical data are used to create motifs and motif clusters, and then the clusters are used for each interval to train SVM models using the given data and previously acquired calibration results. Motifs might result in differing price movements based on their underlying characteristics. Our program represents the Zigzag technical indicator data as a set of inputs to train for the SVM to capture these characteristics.

ZZMOP Model Construction Algorithm, which is also given in Algorithm 2, has the following steps:

- Create a new motif with the Zigzag indicator. This includes normalization of the motif data based on the primary data point in the motif. Then, store the new motif in the list of motifs to cluster (lines 3 and 4).
- Cluster the current set of motifs based on their similarities (line 5).
- Calculate reward risk ratio based on available future of motifs (lines 6 and 7). The cluster reward risk ratio is the average of reward risk ratios of each motif in that cluster.
- Train SVMs for forecasting the price at the end of the event horizon, stop-loss and take-profit values (lines 8, 9, and 10).

The following subsections describe the steps of the ZZMOP model construction algorithm whose flow is depicted in Fig. 5.

**Algorithm 2** ZZMOP Model Construction Algorithm

**Require:** $CL \leftarrow$ currency list, $RPH \leftarrow$ raw price history with fixed time frame
**Ensure:** SVMs trained from clusters containing historic motif clusters
1: **for all** $c \in CL$ **do**
2:     **for all** $timeframe \in RPH$ **do**
3:         $motif \leftarrow Zigzag[RPH[c, timeframe]]$
4:         $motifList.append(motif)$
5:     $clusterer[c] \leftarrow buildCluster(motifList)$
6:     **for all** $motif \in motifList$ **do**
7:         $clusterer[c].updateClusterRewardRiskRatio(motif)$
8: $lastForecastingSVM \leftarrow train[cluster]$
9: $lowUntilForecastingSVM \leftarrow train[cluster]$
10: $highForecastingSVM \leftarrow train[cluster]$
11: **return** SVMs

#### 5.1.1 Motif construction

**Modified Zigzag indicator with thickness**

Zigzag indicator is used to highlight the significant highs and lows of the instrument's historic values and eliminates the noise in the data; however, it does not provide a method to indicate

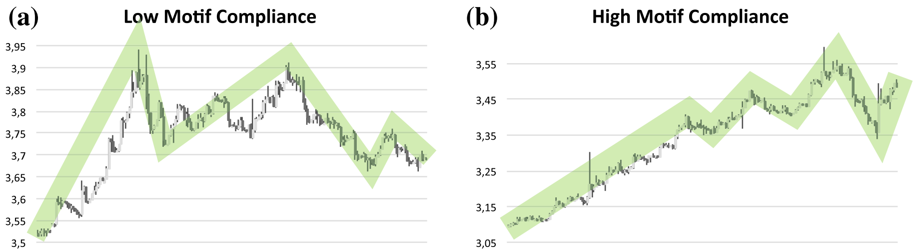


**Fig. 6** **a** A thick Zigzag line with low motif compliance and **b** a thick Zigzag line with high motif compliance

the compliance of the time series data with Zigzag lines. We add a thickness variable to the Zigzag indicator to represent the compliance of the time series data with the $Zigzag_{thick}$ lines.

**Definition 5.2** (*Zigzag thickness*)—Given time series $\mathbf{T} = \langle t_1, t_2, \ldots, t_n \rangle$ $(t_i \in N^+)$ and thickness $k$ Zigzag, $\mathbf{Z_{thick}}$ satisfies the following:

1. $\mathbf{Z_{thick}} = \langle z_1, z_2, \ldots, z_n \rangle \subset \mathbf{T}$
2. $\forall z_i \; z_i \neq z_{i+1}$
3. $\forall$ Zigzag point $z_i \in \mathbf{Z_{thick}} = \langle z_1, \ldots, z_n \rangle \rightarrow ((z_i > z_{i-} \wedge z_i > z_{i+}) \vee ((z_i < z_{i-} \wedge z_i < z_{i+})$
4. $\forall t_i \in T, \forall z_i \in \mathbf{Z_{thick}} \rightarrow z_i - k \leq t_i \leq z_i + k$

Using thickness as an indication of motif compliance requires the consideration of two different variables: first one being the value of the thickness variable itself and the second one being the amount of coverage, which represents the amount of bars that are required to be contained in the thick Zigzag line.

**Definition 5.3** (*Thick Zigzag coverage*)—Given a multivariate time series $\mathbf{MT} = \langle \langle low, high \rangle_1, \langle low, high \rangle_2, \ldots, \langle low, high \rangle_n \rangle$ where $o_m \in \mathbb{R}$, $\mathbf{Z_{thick}} = \langle z_1, z_2, \ldots, z_n \rangle \subset \mathbf{T}$ with thickness $k$, and coverage $C$, $C\%$ of the elements in $\mathbf{MT}$ satisfies the following: $\forall t_i \in \mathbf{MT}$, $\forall$ Zigzag point $z_i \in \mathbf{Z_{thick}} \rightarrow z_i - k \leq t_i \leq z_i + k$

There could be scenarios where requiring full coverage which would require a very high thickness is less optimal than a partial coverage which would require a lower thickness. In each scenario, a high bar percentage value and a low thickness value would indicate a higher motif compliance. Similarly, a low bar percentage and high thickness would indicate a low motif compliance. The Zigzag indicator with thickness effectively creates an error function to describe a covering motif [30] which can be used to select thinnest motifs available. Two sample motifs with low and high compliances to the thick Zigzag lines are provided in Fig. 6a, b, respectively.

*Detection of the signal point in a Motif*
A financial time series in the Forex environment is a multivariate time series consisting of observations made at equal intervals. Each observation is a vector $\mathbf{O} = \langle open, close, low, high \rangle$. Even though most distance measures and mining algorithms are invariant to the start time and sampling interval of the time series [12,36], the case for the Forex market is different [8]. Therefore, the detection of a signal point is crucial for the success of forecasting. Detection of the signal point in a motif problem is similar to change point detection in time series. A survey of change point detection methods can be found

in [2]. Our approach includes time series transformations to the original time series and is supervised by its nature.

For a given financial time series $\mathbf{T}$, Zigzag indicator creates a list of values $\mathbf{V} = \langle zz_1, zz_2, ..., zz_n\rangle$ such that $(\langle zz_1, zz_3, \ldots, zz_{2x+1}\rangle \in \mathbf{ZZ_{low}} \wedge \langle zz_2, zz_4, \ldots, zz_{2x}\rangle \in \mathbf{ZZ_{high}}) \vee (\langle zz_1, zz_3, \ldots, zz_{2x+1}\rangle \in \mathbf{ZZ_{high}} \wedge \langle zz_2, zz_4, \ldots, zz_{2x}\rangle \in \mathbf{ZZ_{low}})$ where $\mathbf{ZZ_{low}} = \{zz_{low} : zz_{low} \in \mathbf{T}\} \Leftrightarrow \forall o_x \in \{o_{i-depth}..o_{i+depth}\} : o_i \leq o_x \vee \mathbf{ZZ_{high}} = \{zz_{high} : zz_{high} \in \mathbf{T}\} \Leftrightarrow \forall o_x \in \{o_{i-depth}..o_{i+depth}\} : o_i \geq o_x$.

In other words, the values generated by the Zigzag indicator are Zigzag high and Zigzag low values in alternating order. Also, the Zigzag values are members of the original time series; however, for a specific interval a Zigzag can be valued at any of the four possible values of the observation vector.

Zigzag creates high and low values for specific time intervals; however, for the remaining intervals no values are generated. Ordering all of the values generated by Zigzag in a set $T_V = \{(t_i, v_i) \mid t_i < t_{i+1}, i = 1 \ldots n-1\}$ would result in a sparse time series. A sparse time series has many empty valued observations as opposed to nonzero observations. The ratio between the length of the time series and the number of nonzero observations is defined as the *sparsity factor* $s$ of the time series. In our application, the sparsity factor $s$ is dependent on Zigzag depth $d$. A higher Zigzag depth would result in more empty valued observations. For the encoding of the Zigzag time series $\mathbf{T_V}$, we use a length-encoding method [31]. In the length-encoded series, $\mathbf{T_{Ve}}$ we replace the contiguous $k$ zeroes in $\mathbf{T_V}$ with a $(k)$. For instance, for the sparse Zigzag time series $\mathbf{T_V} = \langle 1.2569, *, *, *, *, 1.2437, *, *, *, 1.2701\rangle$, the length-encoded series is $\mathbf{T_{Ve}} = \langle 1.2569, (4), 1.2437, (3), 1.2701\rangle$. In this example, 1.2569 and 1.2701 are high Zigzag points and 1.2437 is the low Zigzag point. All three points are *Zigzag points*, whereas the remaining 7 points are insignificant data points.

**Definition 5.4** (*Signal point*)—For a motif $\mathbf{T_V} = \langle v_1, v_2, \ldots, v_n\rangle$, the $n$th point in the original time series is a signal point if the motif belongs to a cluster $C$ where $Reward/RiskRatio(C) > e$, and $e$ is a predefined error. The signal price for a motif, $p_{zz}$ at point $n$ in $\mathbf{T_V} = \langle v_1, v_2, \ldots, v_n\rangle$ is equal to the closing price $close_i$ of the $n$th point in the original time series $\mathbf{T} = \langle o_1, o_2, \ldots, o_n\rangle$ where $\mathbf{o_i} = \langle open_i, close_i, low_i, high_i\rangle$.

### 5.1.2 Clustering motifs

The clustering and classification of motifs in different time frames require time series transformations such as amplitude shifting (adding a constant value), uniform amplification (multiplying with a constant values) and uniform time scaling (uniform change of time). Distance or similarity between the Zigzag motifs in the clusters is defined based on these transformations.

The Zigzag motifs detected by our system are clustered via the expectation maximization (EM) clusterer [9]. The EM algorithm operates iteratively to find the maximum likelihood estimates of parameters in a statistical model. The EM model depends on unobserved latent variables. EM iterations alternate between performing the expectation step and the maximization step. The expectation step creates the function for expectation of the log likelihood with the current estimate for the parameters. The maximization step computes the parameters maximizing the expected log likelihood found on the expectation step. The estimates are used iteratively to determine the distribution of the latent variables in the next expectation step.

### 5.1.3 Motif reward/risk ratio

Trades in the financial markets come with rewards and risk. The ratio of these two numbers is called the reward/risk ratio [41] and is an indicator of the profitability of the trade position.

**Definition 5.5** (*Reward/risk ratio*)—Reward/risk ratio is the ratio of possible profits to possible losses. Reward is the movement of price in the expected direction, which would result in profits, and the risk is the movement of price in the unexpected direction, which would result in losses.

Every detected motif and signal associated with the motif should not result in an open trade position. The reward associated with the motif should be high enough to bear the risk. In our system, the reward/risk ratio of motif clusters is continuously monitored. Once a trade signal is received from a motif, the trade position is opened if and only if the motif ends up in one of the most rewarding clusters.

Reward/Risk ratio is essentially used to classify a cluster of motifs using a feedback loop where future data in the time series affect the predefined class label of the cluster.

**Definition 5.6** (*High until low*)—High until low is the difference between the signal point of a short motif and the highest point in the event horizon of the motif that is recorded before the lowest point in the event horizon of the motif. It represents the risks associated with a short motif in points.

**Definition 5.7** (*Low until high*)—Low until high is the difference between the signal point of a long motif and the lowest point in the event horizon of the motif that is recorded before the highest point in the event horizon of the motif. It represents the risks associated with a long motif in points.

ZZMOP predicts future movement after the signal point of the motif. Despite a motif being a short motif a “high until low” value (i.e., the risk) is taken into account to allow the future decrease in price to be achieved. For a short motif, the “low” value forecasted represents the reward. A sample short motif having a reward/risk ratio of 3:1 means that when the increase after the decisive point is proportional to 10 pips, the decrease after the maximum drawdown point is proportional to 40 pips which would place the absolute difference between decisive point and maximum profit point proportional to 30 pips. In this scenario, possible losses at maximum drawdown point are valued at 10 pips where the possible rewards at maximum profit point are 30 pips. Hence, the reward/risk ratio is 3:1.

### 5.1.4 Classifying motifs in the clusters

An SVM is a machine that constructs a set of hyper-planes in a multi-dimensional space for classifying and regression of data [7]. An empirical analysis of the systems used in time series forecasting shows that SVMs perform much better than the remaining models in terms of directional symmetry [1]. Even though Gaussian Process and multilayer perceptron is better than SVMs in terms of mean squared error and mean absolute deviation, research shows that in the Forex market directional symmetry is much more important [24,29,48] than distance-based error and deviation performance; therefore, in our system we use SVMs to train on selected technical indicators.

SVM’s main objective is identifying the maximum margin hyper-plane so that separation between classes of samples can be maximized. Mapping of input vectors to the high dimension feature space is performed by a kernel function. Polynomial function and radial basis function kernels are commonly used in trading systems. Due to superior performance recorded by polynomial function kernels in related work [20,37], we use a polynomial function kernel in our system. Degree of polynomial function ($d$) and regularization parameter ($c$) are the parameters of the SVM. To determine these parameters efficiently, several different settings are experimented and the selection is made based on training and validation accuracy. Final results obtained with our algorithm using the optimal parameters are shown in Sect. 6.

In our algorithm, SVMs provide estimates for the captured motif's future behavior such as the maximum amount of decrease/increase to be expected in the event horizon, the maximum amount of inverse price action before the expected movement and the last price at the event horizon. All these SVMs categorical values are discussed in "Experiments" section. Also notice that one of the SVMs is used to forecast the price in terms of the number of pips increase or decrease at the end of the event horizon. Its result is used to decide the lot size of the trade.

### 5.2 ZZMOP trading algorithm

The ZZMOP trading algorithm's flow is described in Fig. 5. Basically, there are 3 steps: selecting the cluster based on the directional bias, determining 3 main parameters of the trade (stop loss, take profit and the lot size), and finally executing the trade. The following two subsections describe the details of the first two steps of this process. The results of the final step are collected to be discussed in "Experiments" section.

ZZMOP Trading Algorithm, whose pseudocode is given in Algorithm 3, has the following steps:

- Determine whether the motif reward/risk ratio is high enough to buy or sell (lines 5 and 10).
- Determine the number of lots to use with the trade order based on the expected increase/decrease at the end of motif event horizon (lines 6 and 11).
- Determine the stop loss point to use with the trade order based on the expected minimum low/maximum high value until the high value / low value is encountered in the motif event horizon (lines 7 and 12).
- Determine the take profit point to use with the trade order based on the expected maximum value/minimum value in the motif event horizon (lines 8 and 13).
- After determining the parameters to the trade with the forecasts, submit the trade order to the underlying trade platform (lines 9 and 14).

#### 5.2.1 Motif directional bias

Prices in the financial markets fluctuate in both directions in a specific time frame but with different magnitudes. One can benefit from a motif in both directions, as long as take profit and stop loss locations are determined properly. Since the event horizon of a motif contains movements in both directions, a reward/risk ratio can be computed for both directions, too.

For a motif to be used for a sell position, the reward/risk ratio of the motif in the sell direction should be higher than the reward/risk ratio in the buy direction. For a motif to carry a meaningful information with respect to the direction of trade, the reward/risk ratios between sell and buy positions should have a significant difference. Otherwise, the motif can be used in both directions and hence does not point to a certain type of trade.

**Algorithm 3** ZZMOP Trading Algorithm

**Require:** $CL$ ←currency list, $clusterer$ ←clusterers, $RPF$ ← raw prices in the future with fixed time frame
**Ensure:** Trade orders with trading parameters

**for all** $c \in CL$ **do**
  **for all** $timeframe \in RPF$ **do**
    $motif \leftarrow Zigzag[RPF[c, timeframe]]$
    **for all** $motif \in clusterer[c]$ **do**
      **if** $buyRewardRiskRatioHigh(cluster)$ **then**
        $lotSize \leftarrow lastForecastingSVM$
        $stopLoss \leftarrow lowUntilForecastingSVM$
        $takeProfit \leftarrow highForecastingSVM$
        **return** $order = submitBuyOrder(c, lotSize, stopLoss, takeProfit)$
      **else if** $sellRewardRiskRatioHigh(cluster)$ **then**
        $lotSize \leftarrow lastForecastingSVM$
        $stopLoss \leftarrow highUntilLowForecastingSVM$
        $takeProfit \leftarrow lowForecastingSVM$
        **return** $order = submitSellOrder(c, lotSize, stopLoss, takeProfit)$

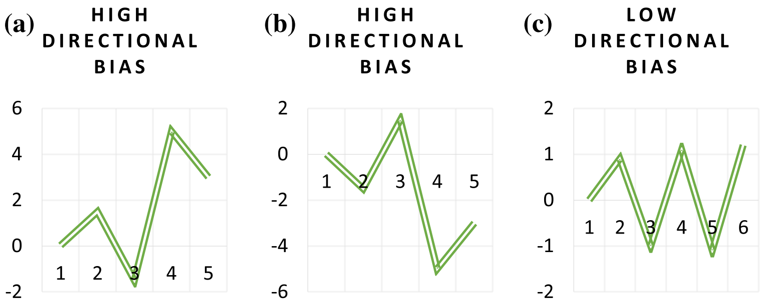


**Fig. 7** Price changes in the event horizon of a **a** buy motif with high directional bias **b** sell motif with high directional bias **c** motif with low directional bias

**Definition 5.8** (*Motif directional bias*)—Given a motif M, directional bias of M can be calculated as the absolute difference of short (or going short on a currency pair means selling the base currency in the pair against the quote currency) and long (or going long on a currency pair means buying the base currency in the pair against the quote currency) risk reward ratios: $db_M = |RewardRiskRatio_{short} - RewardRiskRatio_{long}|$.

For a given motif, the reward/risk ratio can be computed once the prices materialize in the event horizon of the motif. We provide sample price action in the event horizon of different motifs in Fig. 7. Figure 7a shows the price action of a buy motif with high directional bias, (b) a sell motif with high directional bias, (c) a motif with low directional bias.

### 5.2.2 Predicting trade parameters: lot size, stop loss and take profit locations

Before entering a trade with a discovered motif, our algorithm determines the lot size for the order, the stop loss and take profit locations. This is performed using the previously recorded motifs in the cluster. For each type of order, a separate set of values are forecasted to determine these values.

For sell orders, a drop in the price is expected in the future. The expected amount of drop is supposed to be the take profit location. However, there might be a slight increase in the price before a drop occurs. Our system should be able to overlook this much of a loss at any time to reach the forecasted drop and take profit. Another possibility is that the increase in the price continues and price does not drop in the event horizon; in which case, the system should stop the losses. Hence, the stop loss location should account for these two possible price movements. The stop loss location for a sell order is called "high until low," and the take profit location is called "low."

For buy orders, a hike in the price is expected in the future, which will be the take profit location. A decrease in price prior to the hike should be allowed to reach the hike forecasted. A higher decrease should be evaluated as a false signal, and losses shall be suffered. The stop loss location for a buy order is called "low until high," and the take profit location is called "high." Our system trains with previous data in a motif's cluster to determine the appropriate take profit and stop losses to maximize gains for each type of motif.

For both types of orders, the expected ultimate value in the event horizon of the motif reflects the reliability of the signal. If the expected ultimate value is very high for a motif, the likelihood of a profit for the motif is higher. In that case, the lot size of the order can be increased accordingly to maximize profits. The same logic applies to sell orders.

## 6 Experiments

In our algorithm, we are making use of the Zigzag technical indicator, expectation maximization clustering algorithm and SVMs. All three of these components have certain parameters that adapt them to the problem at hand. We discuss the characteristics of our dataset and how we adapt parameters for the aforementioned components to our data.

### 6.1 Characteristics of our dataset

The transactions and data in the Forex market are determined in real time by an electronic network of Forex brokers, liquidity providers, banks and other financial institutions. Historical data are available through several different channels, and the data used in our work are obtained from TrueFX [45], which is also used in related work. Two types of historical data are collected by our system. First is price data with 15 min intervals which summarizes the opening, closing, low and high values for the given interval. Fifteen-minute interval data values are used for training purposes. Second data are the real-time price data which contain all the price changes that have happened in the currencies. Real-time price data are used for testing and simulation purposes. Both data are collected for currency pairs including USD, EUR, GBP, and CHF. Summary statistics of the data can be found in Table 1. Total data points are the number of 15 min bars, minimum and maximum values represent the highest and lowest values recorded, average value is the average of 15 min closes, and increases and decreases in value show a 15 min increase or a 15 min decrease in the closing price of the currency pair. For training, 15 min values are used.

Since our system makes use of real-life trading parameters such as stop loss and take profit, for testing and model simulation, real-time price data are used. Statistics for these values are found in Table 2.

The set of parameters stored for each financial instrument are listed below in Table 3.

**Table 1** 15-minute exchange rate (training data) statistics

| Instrument | Total data points | Maximum value | Minimum value | Average value | Increases in value | Decreases in value |
|---|---|---|---|---|---|---|
| EUR/CHF | 121,440 | 1.488 | 1.007 | 1.253 | 55776 | 65664 |
| EUR/GBP | 121,536 | 0.913 | 0.776 | 0.839 | 60960 | 60576 |
| EUR/USD | 121,536 | 1.493 | 1.187 | 1.334 | 62208 | 59328 |
| GBP/CHF | 121,440 | 1.711 | 1.146 | 1.493 | 58944 | 62496 |
| GBP/USD | 121,536 | 1.718 | 1.423 | 1.589 | 60960 | 60576 |
| USD/CHF | 121,536 | 1.172 | 0.706 | 0.941 | 59424 | 62112 |

**Table 2** Real-time exchange rate (testing data) statistics

| Instrument | Total data points | Maximum value | Minimum value | Average value | Increases in value (%) | Decreases in value (%) |
|---|---|---|---|---|---|---|
| EUR/CHF | 59,948,502 | 1.488 | 1.007 | 1.248 | 45.88 | 54.12 |
| EUR/GBP | 67,221,512 | 0.913 | 0.776 | 0.826 | 50.12 | 49.88 |
| EUR/USD | 62,915,513 | 1.493 | 1.187 | 1.316 | 51.12 | 48.88 |
| GBP/CHF | 76,187,806 | 1.711 | 1.146 | 1.498 | 47.99 | 52.01 |
| GBP/USD | 65,708,389 | 1.718 | 1.423 | 1.591 | 50.22 | 49.78 |
| USD/CHF | 59,461,873 | 1.172 | 0.706 | 0.939 | 48.84 | 51.16 |

**Table 3** Symbol parameters

| Parameter name | Parameter definition |
|---|---|
| Symbol name | The name of the financial instrument (i.e., EUR/USD, GBP/CHF) |
| Open | The opening value of the financial instrument for the given trade day |
| Close | The closing value of the financial instrument for the given trade day |
| Low | The lowest value the financial instrument is traded for in the given trade day |
| High | The highest value the financial instrument is traded for in the given day |

## 6.2 Zigzag-related parameters

Zigzag can detect any number of points given a large enough history window. Our system can then use the Zigzag discovered price points to discover motifs. The size of the window, the minimum required depth of Zigzag points and number of Zigzag points to use are Zigzag-related parameters of our algorithm. These are detailed in Table 4.

The number of points to take into account when clustering and classifying possible motifs is particularly important for the success of the system. We are detailing the selection of 7 Zigzag points for our motifs based on the experiments with different motif lengths to determine the optimal length of a motif in Table 5.

**Table 4** Zigzag-related parameters

| Parameter name | Values experimented | Value used | Explanation |
|---|---|---|---|
| History window size in bars | 50, 100, 150, 200 | 150 | Our algorithm will run the Zigzag indicator in a history window of price bars. This number determines how large that window will be, and hence how many price bars it will contain |
| Zigzag depth | 8, 12, 16, 20 | 16 | Zigzag will look for tops and bottoms for a specific period of bars. This number determines the required minimum depth for a top or bottom to be formed |
| Zigzag deviation | 5, 10, 15, 20 | 5 | The deviation value of the Zigzag indicator represents the number of pip points that are required to establish a new low after a high, or a new high after a low |
| Zigzag pattern length | 5, 6, 7, 8, 9, 10 | 7 | Zigzag will look for this many Zigzag points to form a motif. A very short motif would be too common to be meaningful, and a very long motif would be too scarce |
| Zigzag thickness | 5, 10, 30, 50 | 10 | Zigzag thickness is the amount of pips Zigzag line will extend in both directions to cover prices occurring in the instruments historic data |
| Thick Zigzag coverage | 0.5, 0.6, 0.7, 0.8, 0.9, 1 | 0.8 | Thick Zigzag coverage is the percent of price action that has to be covered by the thick Zigzag line |

**Table 5** Performance of different Motif lengths

| Motif length | Average short reward/risk ratio | Average long reward/risk ratio | Top short reward/risk ratio | Top long reward/risk ratio | Average reward/risk ratio difference |
|---|---|---|---|---|---|
| 5 | 2.0880 | 2.0511 | 3.2293 | 2.6919 | 0.7630 |
| 6 | 2.0969 | 2.1037 | 3.0239 | 2.5817 | 0.6615 |
| 7 | 2.2250 | 2.1827 | 3.3267 | 3.0474 | 0.6763 |
| 8 | 2.2036 | 2.1296 | 3.0884 | 2.9385 | 0.5712 |
| 9 | 2.2180 | 2.1887 | 3.2179 | 2.8030 | 0.5849 |
| 11 | 2.2791 | 2.1648 | 2.9968 | 2.7557 | 0.4844 |
| 13 | 2.2960 | 2.2408 | 3.1513 | 2.9476 | 0.6079 |

The different motif lengths resulted in motif clusters with different characteristics and although top efficiencies were achieved by motifs with length 7, there was no clear winner since some other motif lengths resulted in higher average reward/risk ratio and reward/risk ratio differences. To make a selection, we have awarded each first, second and third place three, two and one points, respectively. Based on this criteria, motifs with length 7 were selected since they have acquired the highest score.

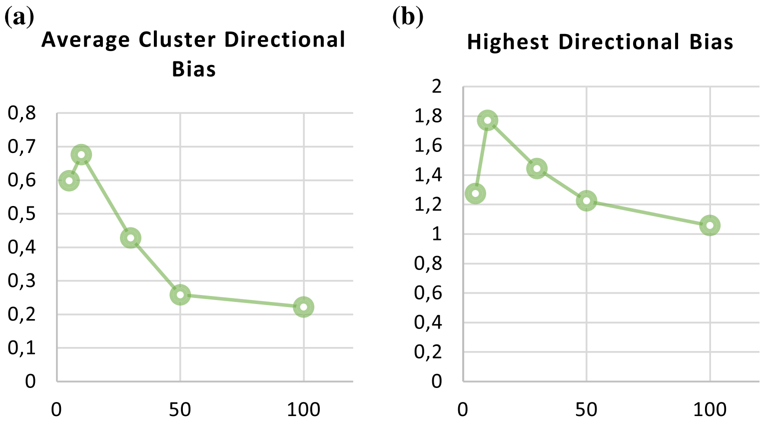


**Fig. 8** **a** Average cluster directional bias with respect to Zigzag thickness, **b** highest cluster directional bias with respect to Zigzag thickness with fixed thick Zigzag coverage of 0.8

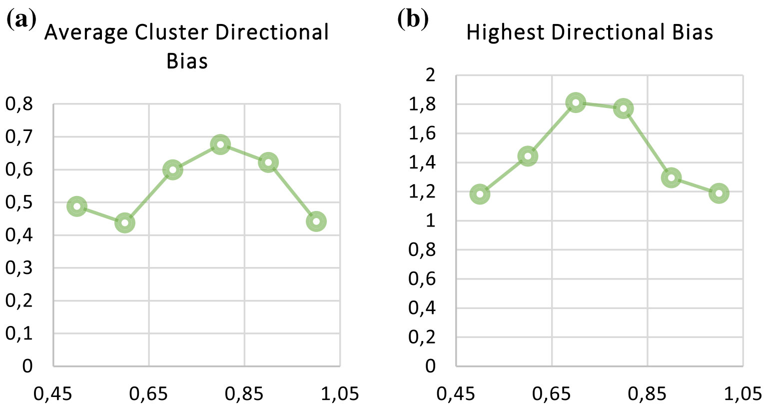


**Fig. 9** **a** Average cluster directional bias with respect to thick Zigzag coverage, **b** highest cluster directional bias with respect to thick Zigzag coverage with fixed Zigzag thickness of 10

The thickness and coverage properties of the modified Zigzag indicator contribute significantly to the directional bias of the discovered motifs and hence the success of our algorithm. Fig. 8 depicts the (a) variation of average cluster directional bias and (b) directional bias of the cluster with highest directional bias with respect to Zigzag thickness with a fixed thick Zigzag coverage value of 0.8. Figure 9 depicts the variation aforementioned variables with respect to Thick Zigzag coverage with a fixed Zigzag thickness of 10.

### 6.3 Clustering-related parameters

The EM clusterer uses certain parameters to operate, which we have exhaustively searched with 432 trials to find the optimal. The used parameters are explained in Table 6

With the parameters specified in Table 4 and 6, EM clusterer creates 23 clusters for the interval 2010–2015 with a 15 min operational time frame. Clusters with top three highest

**Table 6** EM Parameters

| Parameter name | Values experimented | Value used | Explanation |
|---|---|---|---|
| Iterations | 100, 500, 1000 | 500 | The maximum number of iterations to perform. A low number would result in a solution that is not converged. A high number would result in unnecessary computation |
| Folds | 10, 50, 100 | 10 | Number of folds to use when cross-validating to find the best number of clusters |
| Minimum log likelihood improvement | 0.00001, 0.0001, 0.001, 0.01 | 0.0001 | The minimum improvement in cross-validated log likelihood used to consider increasing the number of clusters when cross-validating to find the best number of clusters |
| Minimum standard deviation | 0.00001, 0.0001, 0.001, 0.01 | 0.00001 | The minimum value for standard deviation when calculating normal density |
| Simple K-means runs | 10, 50, 100 | 50 | Number of runs of k-means to perform |

short and long position efficiencies are highlighted below. The average statistics regarding the clusters are specified below in Table 7 and displayed in Figs. 10, 11.

### 6.4 SVM-related parameters

An SVM requires certain parameters which directly affect its performance. We have empirically experimented with several values for these parameters that have been mentioned in related research and selected the best-performing ones. To determine the best-performing parameters for our SVM, all possible combinations of the parameters have been exhaustively searched resulting in 540 experiments per each training interval. The parameters that we have used are outlined in Table 8.

### 6.5 Trading parameters

A realistic trading system requires certain parameters such as trading volume, stop loss or take profit. In our system, we have empirically experimented with several values for these parameters that are common in the industry and selected the best-performing ones. To determine the best-performing parameters for our system, all possible combinations of the parameters have been exhaustively searched resulting in 30,720 experiments per each forecasting period of 15 minutes for an interval of 5 years. These parameters are outlined in Table 9.

### 6.6 Sample chart patterns detected by ZZMOP

As specified in Algorithms 1 and 2 for each currency, at the completion of each bar, the Zigzag indicator is applied to the historic bars of the currency to create Zigzag points. Zigzag points

**Table 7** Clusters created by EM for 2010–2015 interval

| Cluster number | Pattern count | Average pattern length | Short position reward /risk ratio | Long position reward /risk ratio | Reward/risk ratio difference |
|---|---|---|---|---|---|
| 0 | 126 | 142.09 | 2.8512 | 1.7218 | 1.1294 |
| 1 | 534 | 103.01 | 2.3607 | 1.8544 | 0.5063 |
| 2 | 166 | 139.53 | 1.9180 | 2.4896 | 0.5716 |
| 3 | 267 | 132.80 | 1.8988 | 2.1156 | 0.2168 |
| 4 | 310 | 133.48 | 2.0464 | 2.2881 | 0.2416 |
| 5 | 109 | 154.15 | 1.6734 | 2.5769 | 0.9035 |
| 6 | 196 | 130.73 | 2.2928 | 2.0434 | 0.2494 |
| 7 | 195 | 122.04 | 2.8373 | 1.6496 | 1.1877 |
| 8 | 478 | 108.51 | 2.5654 | 1.8966 | 0.6687 |
| 9 | 105 | 151.11 | 2.9010 | 1.7912 | 1.1098 |
| 10 | 374 | 116.91 | 2.0763 | 1.9058 | 0.1704 |
| 11 | 216 | 140.06 | 3.2293 | 1.4582 | 1.7711 |
| 12 | 151 | 176.78 | 2.1294 | 1.8646 | 0.2647 |
| 13 | 216 | 133.43 | 1.6528 | 2.9476 | 1.2947 |
| 14 | 341 | 128.65 | 2.1239 | 1.9483 | 0.1756 |
| 15 | 459 | 102.88 | 2.0389 | 2.6718 | 0.6328 |
| 16 | 152 | 146.03 | 2.4670 | 1.5849 | 0.8821 |
| 17 | 518 | 121.86 | 1.7670 | 2.7252 | 0.9581 |
| 18 | 392 | 118.46 | 1.7768 | 2.9415 | 1.1646 |
| 19 | 254 | 129.70 | 2.3306 | 1.8516 | 0.4789 |
| 20 | 532 | 107.69 | 1.8644 | 2.4398 | 0.5754 |
| 21 | 290 | 140.25 | 2.1870 | 1.9863 | 0.2006 |
| 22 | 224 | 126.80 | 2.0254 | 2.2271 | 0.2016 |

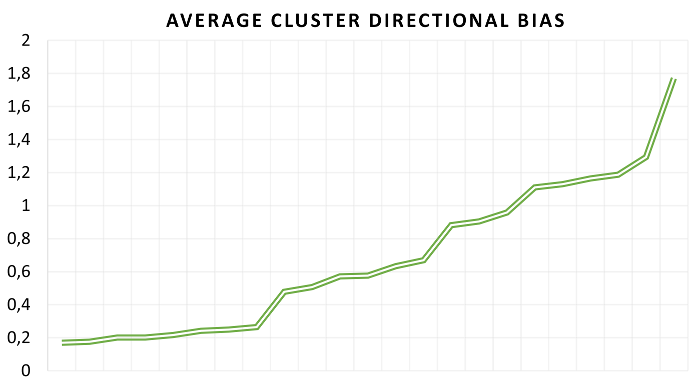


**Fig. 10** Average directional bias of motifs in different clusters

are discontinuous and are alternating between highs and lows, as is the case with legacy chart patterns. In this section, we will show two legacy chart patterns and two new motifs detected by our system.

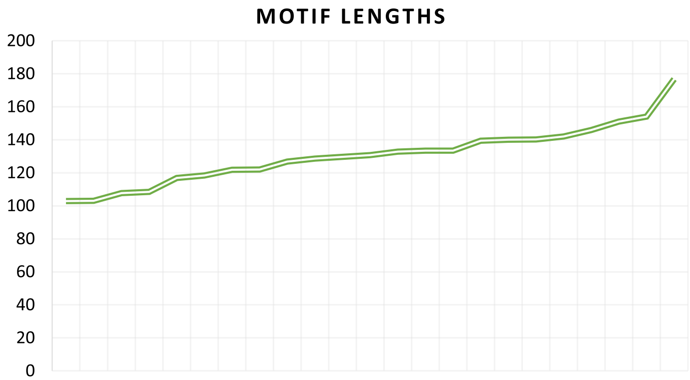


**Fig. 11** Distribution of average lengths of motifs in different clusters

**Table 8** SVM parameters

| Parameter name | Values experimented | Value used | Explanation |
|---|---|---|---|
| Kernel type | Polynomial, radial basis | Polynomial | Polynomial function kernels achieved superior performance in related work [20,21]. Thus, we use the polynomial function kernel in our system |
| Function degree (d) | 3, 4, 5 | 4 | Degree of polynomial function characterizes how well the model fits to the data. A very high degree could result in overfitting |
| Regularization parameter (nu) | 0.00001, 0.0001, 0.01, 0.1, 0.25, 0.5 | 0.0001 | Regularization parameter controls margin and misclassification error |
| Attribute normalization | Yes/no | Yes | Input attributes are normalized or left with assigned bucket values across the training set |
| Training interval (years) | 1, 2, 3, 4, 5 | 5 | A model in our system is trained with the given number of years before making decisions in validation or actual datasets |

### 6.6.1 Legacy chart patterns

Two famous chart patterns are Head and Shoulders (H&S) and Inverse Head and Shoulders (IH&S) chart patterns. For H&S and IH&S chart patterns to be formed, at least 7 points are required on a chart. Sample H&S and IH&S chart patterns found by ZZMOP are shown in Fig. 12.

The points defining the chart patterns are numbered 1–7 from left to right. The sample chart patterns are less than perfect, for instance 1 and 3 are not equal to each other, and they are also not equal to 5 and 7 which is by definition the case for H&S and IH&S chart patterns. The local tops 2 and 6 are also not equal to each other. However, it is very hard to find a perfect chart pattern in the real world. In the presence of the defining characteristics of the H&S, a chart pattern should be accepted as an H&S chart pattern—same holds true

**Table 9** Trading system parameters learned by the SVM

| Parameter name | Estimated categories | Explanation |
|---|---|---|
| Difference with last value (Pips) | 0, 10, 20, 50, 100 | The forecasts for currency pairs are based on motifs. Each motif impacts a possible future which we call the event horizon. Our model forecasts the last value that occurs at the end of the event horizon. This forecast is used to determine the *Trading Volume in lots*. Estimated pip difference values are mapped into lot sizes as 0.01, 0.1, 0.25, 0.5, 1. In our experiments, using fixed lot sizes proved less profitable, and therefore the lot size is dynamically selected based on trade profitability analysis |
| Stop loss (Pips) | 10, 20, 50, 100 | When a currency pair moves in the opposite direction of an open trade, stop loss amount determines how much loss is accepted before closing position. In our experiments, using fixed stop losses proved less profitable, and therefore the stop loss is dynamically selected based on reverse price-action estimate |
| Take profit (Pips) | 10, 20, 50, 100 | A trade position cannot be kept open forever even though it is in profit, since markets fluctuate. Take profit determines how much profit a single open trade can make. In our experiments, using fixed take profits proved less profitable, and therefore the take profit is dynamically selected based on price-action estimate |

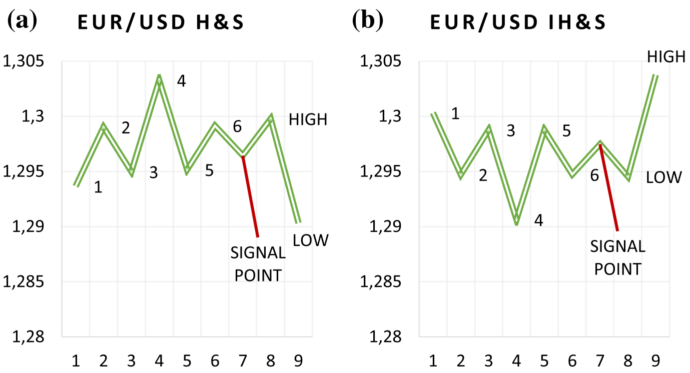


**Fig. 12** A Sample **a** H&S pattern, **b** IH&S Pattern discovered by ZZMOP

for IH&S. These characteristics are the presence of three hills where the first and third (i.e., the shoulders) hills are smaller than the second hill (i.e., the head). Therefore, a preliminary elimination in our algorithm looks for at least 7 Zigzag points in the history window of a bar which have these characteristics of an H&S chart pattern.

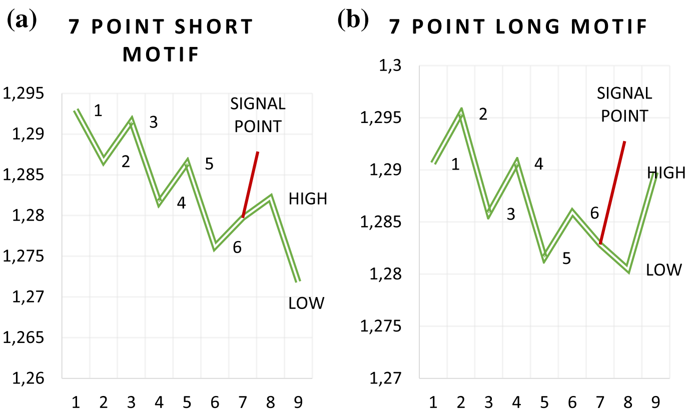


**Fig. 13** Sample novel **a** Short pattern, **b** Long pattern discovered by ZZMOP

### 6.6.2 Novel motifs

Our system does not only detect previously established chart patterns, and it also discovers new motifs and groups them into clusters with similar future price-action characteristics. Two sample motifs that are detected by our algorithm are presented in Fig. 13. The 7 points in the motifs are the signal points and marked with "Signal label." The future price action for the motifs show an average of the future behavior of motif instances.

It can be observed when averaged, the behavior of the future prices result in an amount of inverse price. In cases where the inverse price movement is equal to the desired price movement, the motifs are inefficient due to commissions. For these motifs, the time series are $\mathbf{T_{short}} = \langle 1.29299, 1.28672, 1.29156, 1.28164, 1.28644, 1.27619, 1.27972, 1.28215, 1.27186\rangle$ and $\mathbf{T_{long}} = \langle 1.29061, 1.29542, 1.28582, 1.29064, 1.28158, 1.28593, 1.28283, 1.28047, 1.28979\rangle$. The reward/risk ratio of these motifs are $R_{short} = 3.2345$ and $R_{long} = 2.9491$.

### 6.6.3 Experiments with random dataset

In order to verify that our solution does not overfit the test data, we have also made experiments with random data that we have generated. Our experiments with real data resulted in average motif reward risk ratios around 2 and more in some clusters. As expected, in random data, most of the reward risk ratios were around 1, meaning in the event horizon of the motif there's roughly same upside and downside potential. This goes for both types of operations, both short and long operations. Hence, average reward risk ratio is almost 0 in all the motif lengths. This means that the clusterer cannot actually differentiate meaningfully between any motif, since motifs do not really have intrinsic meanings. Thus, our system normally do not make any transaction with such a result, which is probably the best policy under random behavior.

All implementations and data can be accessed from https://github.com/mozorhan/ZZMOP.

**Table 10** ZZMOP algorithm performance

| Traded currencies | Precision | Recall | Accuracy | F-measure |
|---|---|---|---|---|
| EUR/CHF | 0.6953 | 0.6121 | 0.6577 | 0.6510 |
| EUR/GBP | 0.6283 | 0.6363 | 0.6334 | 0.6323 |
| EUR/USD | 0.6358 | 0.6721 | 0.6548 | 0.6534 |
| GBP/CHF | 0.6221 | 0.6044 | 0.6189 | 0.6131 |
| GBP/USD | 0.6881 | 0.6702 | 0.6737 | 0.6790 |
| USD/CHF | 0.6445 | 0.6425 | 0.6508 | 0.6435 |

## 7 Performance of the system

Different trading strategies and input compositions were used with the system, and different results in terms of performance were obtained. Obtained results are presented below. For exchange ratios, buy and sell decisions are materialized based on the closing value of the decision date, since the market is open for 24 h. The system can generate successive buy signals, successive sell signals or alternating buy and sell signals. At any time, different orders expecting market movement in both directions could be open. This is due to the fact that signals generated by our system can have different reward/risk ratios and may result in different trade parameters such as take profit, stop loss and lot size.

Our forecasting SVMs forecast possible termination points for a trade for each motif. The termination can be performed by a take profit, stop loss or a time stop at the event horizon. Performance of the system is measured with accuracy and f-measure criteria based on the termination of the motifs as is the case in related work [4, 26, 34]. Computation of these are made with precision and recall which are calculated from true positive (TP), false positive (FP), true negative (TN) and false negative (FN) decision counts (Table 10).

### 7.1 Comparison of performance

The state-of-the-art models do not try to determine the magnitude of future movement in both directions to increase their trade accuracy, rather they predict a single value and measure the direction of the movement with their prediction direction. For the comparison, we have implemented Kamruzzaman and Sarker's [18] algorithm SCG-ANN, Stella and Villa's [46] algorithm CTBNC, Shen et al. [42] algorithm DBN-CRBM, Moosa and Burns' approach [29] TVP, Zhang et. al.'s SFM algorithm [49]. We are also comparing ZZMOP with our previous algorithm SCPT-DAP from [34]. All algorithms use the same dataset obtained from TrueFX. Same spreads and commissions defined in our system are applied. For each algorithm, paper's origin country time-zone is used. The results are presented in the below table.

In their original work, Kamruzzaman and Sarker make [18] weekly forecasts for AUD against five major forex currencies. The highest directional accuracy recorded in the given work by SCG-ANN algorithm is 0.7714 for the AUD/GBP exchange rate. The weekly data used are from years 1991 to 2002. There are 65 testing data points which result in the given performance. In our experiments, this model's best performances were recorded for GBP pairs (i.e., GBPCHF, GBPUSD and EURGBP). The highest accuracy was achieved at GBPUSD pair and is 0.6398.

**Table 11** Performance summary of currency pair trading systems based on accuracy

| Currency pair | ZZMOP | SCPT DAP | TVP | DBN CRBM | CTBNC | SCG ANN | SFM |
|---|---|---|---|---|---|---|---|
| EUR/CHF | **0.6577** | 0.6297 | 0.5842 | 0.6221 | *0.6340* | 0.5162 | 0.5227 |
| EUR/GBP | **0.6334** | 0.6068 | 0.5466 | 0.6130 | 0.5940 | *0.6145* | 0.4610 |
| EUR/USD | **0.6548** | *0.5962* | 0.5506 | 0.5782 | 0.5529 | 0.5632 | 0.4675 |
| GBP/CHF | 0.6189 | **0.6320** | 0.4988 | 0.6032 | *0.6190* | 0.5826 | 0.5130 |
| GBP/USD | **0.6737** | 0.6382 | 0.5103 | 0.6272 | 0.6240 | *0.6398* | 0.5649 |
| USD/CHF | **0.6508** | *0.6400* | 0.5324 | 0.5782 | 0.6193 | 0.5237 | 0.5357 |
| Average | **0.6482** | *0.6238* | 0.5372 | 0.6037 | 0.6072 | 0.5733 | 0.5108 |

Best accuracy values are shown in bold

Stella and Villa [46] have used a continuous time Bayesian network classifier for predicting intraday values of foreign exchange rates. The predictions have been made in EUR/USD, GBP/USD and EUR/CHF exchange rates. The work uses three different datasets (i.e., TrueFX, Dukascopy and GainCapital), and different directional accuracies have been recorded in different datasets. In our experiments, highest recorded performance of the CTBNC algorithm is 0.6340 for EUR/CHF.

Shen et al. work [42] achieve their highest accuracies in GBP/USD exchange rate. The achieved accuracy is 0.6362. Forecasts are performed weekly, and there are only 52 testing data points. In our experiments, the best performance recorded is again on the GBP/USD with an accuracy of 0.6272 for the DBN-CRBM algorithm.

Mossa and Burns' work [29] uses three different intervals—monthly, quarterly and every six months—to predict the exchange rates of CAD, GBP, JPY and USD pairs. The highest directional accuracy is once again achieved for the GBP/USD exchange rate and is 0.72. This is better than our top GBP/USD forecast of 0.6382; however, this performance achieved with 12 data points and predictions are made in six month intervals as opposed to our daily forecast approach. When forecasts are made for the same currency quarterly, the accuracy falls to 0.56 and for monthly forecasts the accuracy is 0.48. In our experiments, the best accuracy achieved by TVP is at EUR/CHF pair with an accuracy of 0.5842.

Zhang et.al.'s work (State Frequency Memory or SFM in short) [49] uses deep learning approach for stock market prediction. SFM is a variant of well-known Long Short-Term Memory (LSTM) which was originally proposed in [15]. We have applied SFM for forex directly. In our experiments, the error rate of the predictions was also very small and the overall performance results were similar to the ones presented in the paper. On the other hand, as mentioned by the reviewer, for forex trading, rather than the amount of the error of the predicted value, the direction of the prediction is more important. Since their method has been designed to reduce the error, in general they produce very small error, but, unfortunately, they fail to predict the trend direction correctly.

The comparison of performances of our system and related work is presented in Table 11. With seven algorithms and six currency pairs present, ZZMOP performs as the best model in five of the currency pairs, and takes the third place in the GBP/CHF pair. SCPT-DAP and CTBNC outperform ZZMOP in GBP/CHF. In all the remaining instances, ZZMOP performs better than the competition. GBP/CHF pair is the most "exotic" pair among our pairs. Indeed, all of the currencies we have used are "major" currencies which make more than 90% of all FOREX trading in the world. Among these, GBP/CHF is the one with the smallest percentage. Major pairs are more stable; therefore, their expected behaviors are also more predictable.

## 8 Conclusion and future work

This paper presents a novel approach based on a modified version of the Zigzag technical indicator, expectation maximization and support vector machines for predicting short-term trends in financial time series found in the foreign exchange market. At the first stage, raw price data are processed with Zigzag to create Zigzag points. Thickness and coverage properties are introduced to Zigzag indicator to determine the compliance of prices to motifs. Then, the points are clustered each trading interval to better represent the future movement types. The upcoming motifs are also clustered to find similarities with previous motifs. Lastly, prediction is done for multiple parameters to determine which trading parameters would result in optimal profits.

Our work shows that technical indicator data can be used to discover motifs in conjunction with learning models such as support vector machines and clustering algorithms to forecast price changes in financial markets such as the Forex market. The success of the system depends on the selection of motif discovery algorithm, learning model and decision support mechanisms. The proposed strategy proves useful in terms of directional symmetry and profits. Thickness and coverage improvements to Zigzag indicator are found useful in terms of both average and maximal motif cluster directional bias.

Due to the nature of trade in the Forex market, directional symmetry performance of the system is more important than how closely prices are forecasted; therefore, a Forex trading system should focus more on directional symmetry.

Our system is automatic and requires no human intervention. This is practical in a real-time trading scenario, since the trades need to be instantaneous. Initial selection of technical indicator parameters is made using an exhaustive search on historical data. Another important point is our current algorithm approaches each time frame as a trading time frame, and hence it can make simplistic mistakes on low volatility intervals.

For forecasting, raw price data and technical indicator-generated motifs are used, but a rule-based financial instrument selection mechanism can also be implemented for further profits and directional symmetry.

### Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

**Ethical approval** This article does not contain any studies with human participants or animals performed by any of the authors.

## A Appendix

### A.1. Common forex terms

**Definition A.1** (*Take profit*)—*Take profit* is a point in the time series where the trading system locks the profits and closes an open position. It is defined as a number of points from the initial price point.

For example, if a long transaction is expected to reach to a level 1.2500 at the end of the motif event horizon, that transaction can be terminated earlier if a level 1.3000 has been reached before arriving to the end of the event horizon, which gives even a better profit than the expected one. Notice that waiting for the end of the event horizon may produce better or worse result. The user may set the take profit level in terms of the expected value.

**Definition A.2** (*Stop loss*)—*Stop loss* is a point in the time series where the trading system cuts the losses and closes an open position. Similar to the take profit, it is defined as a number of points from the initial price point.

This is the reverse of the take-profit value. Considering the same transaction, it can be terminated at level 1.2000 with a stop loss, accepting some loss before arriving to the event horizon. This decision can be made to prevent potentially higher losses. Notice that waiting for the event horizon may produce better or worse result. Also, the user sets the level in terms of the expected value.

**Definition A.3** (*Signal point*)—*Signal point* is a point in the time series where the trading system opens a new position. The position can be a sell or buy position, depending on the underlying signal.

In our system, signal points are the last and completing points of detected motifs, which are all definitive Zigzag points. However, due to Zigzag's backtracking mechanism, the last Zigzag point might be updated as long as it is not a signal point. The last Zigzag point in a motif becomes a signal point once the motif ends up in a cluster that has enough directional bias to be used for trading.

Consider a W-shaped motif where the 3rd peak is expected to be followed by an upward move. In real time, patterns are discovered in backward searches. So, at each time point, the data points must be searched, if this backward-searched sequence is similar to any of the motifs. For instance, if such a W-shaped motif has been found to be similar with the real data such that the current point corresponds to the last top point of W, this generates a signal for a long transaction to start (assuming 3rd peak is followed by and upward move).

## References


1. Ahmed NK, Atiya AF, Gayar NE, El-Shishiny H (2010) An empirical comparison of machine learning models for time series forecasting. Econom Rev 29(5–6):594–621
2. Aminikhanghahi S, Cook DJ (2017) A survey of methods for time series change point detection. Knowl Inf Syst 51(2):339–367
3. Bolton AH (1976) The Elliott wave principle: a critical appraisal. Monetary Research Limited, Hamilton
4. Brown DP, Jennings RH (1989) On technical analysis. Rev Financ Stud 2(4):527–551
5. Chakraborti A, Toke IM, Patriarca M, Abergel F (2011) Econophysics review: I. Empirical facts. Quant Finance 11(7):991–1012
6. Cont R (2005) Long range dependence in financial markets. In: Lutton E, Levy Vehel J (eds) Fractals in engineering. Springer, Berlin, pp 159–179
7. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
8. Danıelsson J, Payne R (2002) Real trading patterns and prices in spot foreign exchange markets. J Int Money Finance 21(2):203–222
9. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Series B (Methodol) 39(1):1–38
10. Denton AM, Besemann CA, Dorr DH (2009) Pattern-based time-series subsequence clustering using radial distribution functions. Knowl Inf Syst 18(1):1–27
11. Esling P, Agon C (2012) Time-series data mining. ACM Comput Surv (CSUR) 45(1):12
12. Faloutsos C, Ranganathan M, Manolopoulos Y (1994) Fast subsequence matching in time-series databases, vol 23. ACM, New York
13. Gaucan V et al (2011) How to use Fibonacci retracement to predict forex market. J Knowl Manag Econ Inf Technol Econ 2(2):1
14. Gopikrishnan P, Plerou V, Amaral LAN, Meyer M, Stanley HE (1999) Scaling of the distribution of fluctuations of financial market indices. Phys Rev E 60(5):5305

15. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
16. Huang C-J, Yang D-X, Chuang Y-T (2008) Application of wrapper approach and composite classifier to the stock trend prediction. Expert Syst Appl 34(4):2870–2878
17. Kampouridis M, Otero FE (2017) Heuristic procedures for improving the predictability of a genetic programming financial forecasting algorithm. Soft Comput 21(2):295–310
18. Kamruzzaman J, Sarker RA (2003) Forecasting of currency exchange rates using ANN: a case study. In: Proceedings of the 2003 international conference on neural networks and signal processing, 2003, vol 1. IEEE, pp 793–797
19. Keogh E, Chu S, Hart D, Pazzani M (2004) Segmenting time series: a survey and novel approach. In: Data mining in time series databases, World Scientific, pp 1–21
20. Khemchandani R, Chandra S et al (2009) Regularized least squares fuzzy support vector regression for financial time series forecasting. Expert Syst Appl 36(1):132–138
21. Kirkpatrick CD II, Dahlquist JA (2010) Technical analysis: the complete resource for financial market technicians. FT Press, Upper Saddle River
22. Korol T (2014) A fuzzy logic model for forecasting exchange rates. Knowl Based Syst 67:49–60
23. Lee M-C (2009) Using support vector machine with a hybrid feature selection method to the stock trend prediction. Expert Syst Appl 36(8):10896–10904
24. Leitch G, Tanner JE (1991) Economic forecast evaluation: profits versus the conventional error measures. Am Econ Rev 81:580–590
25. Lo AW (1989) Long-term memory in stock market prices, Technical report, National Bureau of Economic Research
26. Loglisci C, Malerba D (2017) Leveraging temporal autocorrelation of historical data for improving accuracy in network regression. Stat Anal Data Min ASA Data Sci J 10(1):40–53
27. Lu C-J, Lee T-S, Chiu C-C (2009) Financial time series forecasting using independent component analysis and support vector regression. Decis Supp Syst 47(2):115–125
28. Major Forex Currencies (n.d.). http://www.investopedia.com/university/forex-currencies/. Accessed 17 Aug 2015
29. Moosa I, Burns K (2014) The unbeatable random walk in exchange rate forecasting: Reality or myth? J Macroecon 40:69–81
30. Mueen A (2013) Enumeration of time series motifs of all lengths. In: 2013 IEEE 13th international conference on data mining (ICDM), IEEE, pp 547–556
31. Mueen A, Chavoshi N, Abu-El-Rub N, Hamooni H, Minnich A (2016) Awarp: fast warping distance for sparse time series. In: 2016 IEEE 16th international conference on data mining (ICDM), IEEE, pp 350–359
32. JV Neumann, Morgenstern O et al (1944) Theory of games and economic behavior. Princeton University Press, Princeton
33. Nguyen H-L, Ng W-K, Woon Y-K (2014) Closed motifs for streaming time series classification. Knowl Inf Syst 41(1):101–125
34. Özorhan MO, Toroslu İH, Şehitoğlu OT (2016) A strength-biased prediction model for forecasting exchange rates using support vector machines and genetic algorithms. Soft Comput, 1–19
35. Pang S, Song L, Kasabov N (2011) Correlation-aided support vector regression for forex time series prediction. Neural Comput Appl 20(8):1193–1203
36. Paparrizos J, Gravano L (2015) k-shape: efficient and accurate clustering of time series. In: Proceedings of the 2015 ACM SIGMOD international conference on management of data, ACM, pp 1855–1870
37. Patel J, Shah S, Thakkar P, Kotecha K (2015) Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. Expert Syst Appl 42(1):259–268
38. Patel Pranav KEJ, Leonardi S (2002) Mining motifs in massive time series databases. In: Proceedings of the ICDM 2002, ICDM, pp 370–377
39. Poh K-L (2000) An intelligent decision support system for investment analysis. Knowl Inf Syst 2(3):340–358
40. Rakthanmanon T, Keogh EJ, Lonardi S, Evans S (2012) MDL-based time series clustering. Knowl Inf Syst 33(2):371–399
41. Risk/reward ratio (n.d.). http://www.investopedia.com/terms/r/riskrewardratio.asp. Accessed 13 Feb 2017
42. Shen F, Chao J, Zhao J (2015) Forecasting exchange rate using deep belief networks and conjugate gradient method. Neurocomputing 167:243–253
43. Smithsonian Agreement (n.d.). http://www.federalreservehistory.org/Events/DetailView/34. Accessed 17 Aug 2015
44. Son NT, Anh DT (2016) Discovery of time series k-motifs based on multidimensional index. Knowl Inf Syst 46(1):59–86

45. Tick-by-Tick Real-Time and Historical Market Rates (n.d.). http://www.truefx.com/. Accessed 13 Feb 2017
46. Villa S, Stella F (2014) A continuous time bayesian network classifier for intraday FX prediction. Quant Finance 14(12):2079–2092
47. Yao J, Tan CL (2000) A case study on using neural networks to perform technical forecasting of forex. Neurocomputing 34(1):79–98
48. Yuan C (2011) The exchange rate and macroeconomic determinants: time-varying transitional dynamics. North Am J Econ Finance 22(2):197–220
49. Zhang Liheng CA, Qi G-J (2017) Stock price prediction via discovering multi-frequency trading patterns. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, ACM, pp 2141–2149
50. Zhou Y, Lin S-C, Wang J-L (2018) Local and global temporal correlations for longitudinal data. J Multivar Anal 167:1–14



**Dr. Mustafa Onur Özorhan** is with Amazon, since 2017. He has received his B.S. degree in computer science from Bilkent University, Ankara, in 2005. Dr. Ozorhan received his. M.S. and Ph.D. from METU, Ankara, in 2010 and 2017, respectively. His current research interests include data mining, text mining, and recommendation systems.

**Prof. İsmail Hakkı Toroslu** is with the Department of Computer Engineering, Middle East Technical University (METU) since 1993. He has received his B.S. and M.S. degrees in computer engineering from METU, Ankara, in 1987 and Bilkent University, Ankara, in 1989, respectively. Prof. Toroslu received his Ph.D. from the Department of Electrical Engineering and Computer Science at Northwestern University, IL, in 1993. He has been a visiting professor in the Department of Computer Science at University of Central Florida between 2000 and 2002. His current research interests include data mining, sentiment analysis and text mining, and recommendation systems. Prof. Toroslu has published more than 80 technical papers in variety of areas of computer science. Prof. Toroslu has also received IBM Faculty Award in 2010.

**Dr. Onur Tolga Şehitoğlu** is with the Department of Computer Engineering, Middle East Technical University (METU) since 2002. He has received his B.S., M.S. and Ph.D. degrees in computer engineering from METU, Ankara, in 1992, 1996, and 2002, respectively. His current research interests include evolutionary computation, computational design, and systems security.