Machine Learning Classification and Regression Models for Predicting Directional Changes Trend Reversal in FX Markets

Article *in* Expert Systems with Applications · January 2021

DOI: 10.1016/j.eswa.2021.114645

2 authors:

Adesola Adegboye
University of Kent

Michael Kampouridis
University of Kent

Machine Learning Classification and Regression Models for Predicting Directional Changes Trend Reversal in FX Markets

Adesola Adegboye[a], Michael Kampouridis[b,*]

[a]School of Computing, University of Kent, Medway, ME4 4AG, UK
Email: atna3@kent.ac.uk
[b]Centre for Computational Finance and Economic Agents,
School of Computer Science and Electronic Engineering,
University of Essex, Wivenhoe Park, CO4 3SQ, UK
Email: mkampo@essex.ac.uk

Abstract

Most forecasting algorithms in financial markets use physical time for studying price movements, making the flow of time discontinuous. The use of physical time scale can make traders oblivious to significant activities in the market, which poses a risk. Directional changes (DC) is an alternative approach that uses event-based time to sample data. In this work, we propose a novel DC-based framework, which uses machine learning algorithms to predict when a trend will reverse. This allows traders to be in a position to take an action before this happens and thus increase their profitability. We combine our approach with a novel DC-based trading strategy and perform an in-depth investigation, by applying it to 10-minute data from 20 foreign exchange markets over a 10-month period. The total number of tested datasets is 1,000, which allows us to argue that our results can be generalised and are widely applicable. We compare our results to six benchmarks (both DC and non-DC based, such as technical analysis and buy-and-hold). Our findings show that our proposed approach is able to return a significantly higher profit, as well as reduced risk, and statistically outperform the other trading strategies in a number of different performance metrics.

Keywords: Directional changes, Regression, Classification, Genetic programming, Forex/FX, Machine learning

1. Introduction

Financial forecasting is a major and very common activity in financial markets [1]. A major challenge faced by financial market traders is the ability to identify trends in the market and predict peaks and troughs of these trends to maximise trading returns with minimal associated risk. A traditional approach used by traders is technical analysis. In this approach, traders use mathematical calculations in identifying and predicting repeating trends in historic data sampled in predetermined physical-time interval [2, 3, 4, 5, 6]. An alternative approach to physical time data sampling is intrinsic time data sampling. In intrinsic time series, data is sampled when events considered to be significant

*Corresponding author
Email address: mkampo@essex.ac.uk (Michael Kampouridis)

occur in the market [7, 8, 9]. Mandelbrot and Taylor [7] were pioneer researchers in intrinsic time series sampling. The idea is, by focusing on important market activities, noise is obfuscated enabling traders build trading strategies around important trends. Since their initial work, different intrinsic time sampling techniques have emerged, such as perceptual important points [10, 11] , turning point [12], zigzag [13, 14], and directional changes (DC) [15, 16, 17]. In this work, we use the DC data sampling approach.

Directional Changes is based on the idea that an event-based system can capture significant points in price movements that the traditional physical time methods ignore. Hence, instead of looking at the market from an interval-based perspective, DC record the key events in the market (e.g., changes in the stock price by a pre-specified percentage) and summarise the data based on these events, moving away from a physical-time view to an event-based-time view. Under this new paradigm, a threshold θ is defined, usually expressed by a percentage of the price. The market is then fragmented and summarised into upward and downward trends. Each of these trends are further dismembered into a directional change (DC) event and an overshoot (OS) event. Different thresholds produce different price summaries. Thus, the directional changes paradigm focuses on the size of price change, while time is the varying factor; whereas in the physical-time paradigm, time was fixed (e.g. daily closing prices).

In a previous work [18], we used a genetic programming (GP) algorithm to undertake symbolic regression and evolve equations that express linear and non-linear relationships between the length of DC and OS events in a given dataset. The advantage of that approach was that it allowed us to predict when a trend will reverse, and thus increase trading profitability. We used this approach as part of a DC-based trading strategy and tested it over 5 different Forex currency pairs for a total of 250 monthly datasets (5 DC thresholds, over 5 Forex pairs, over 10 months). Our findings showed that our proposed approach was able to outperform other state-of-the-art trading approaches.

This work poses an important step forward for more accurate trend reversal prediction by building on the authors' previous work [18] in the following ways. (i) We do not assume that a DC event is always followed by an OS event, as is often done in the literature. Instead, we create a new step, where we use a series of classification algorithms to predict whereas a DC event is going to be followed by an OS event. As part of this step, we propose new DC-based classification attributes. In the end, when a DC event is classified having a corresponding OS event, only then we perform the symbolic regression step from [18], (ii) We propose a new DC-based trading strategy that uses the combined classification and regression steps, (iii) We do not use the same set of fixed thresholds θ across all datasets. Instead, we use a pool of thresholds and then the best thresholds (in terms of RMSE) are selected for each dataset. Thus, the thresholds we use are tailored to the datasets. (iv) We use a wider range of datasets from 20 Forex currency pairs. In total, our experiments are run over 1,000 different directional changes datasets, making our results much more significant and generalisable. (v) We add two new benchmarks and one more performance metric to enhance our results analysis. (vi) We present samples of the best equations returned by the symbolic regression and discuss if we can have a generalised equation for predicting trend reversals across different datasets.

Our goals can thus be summarised as follows: (i) Demonstrate that the combination of classification and regression leads to error reduction when compared to other trend reversal algorithms, and (ii) Demonstrate that our proposed DC-based trading strategy, which utilises our proposed trend reversal approach, is able to be profitable and outperform other trading strategies, both DC and non-DC-based, including from physical time, such as technical analysis and buy-and-hold.

The rest of this paper is organized as follows: Section 2 provides an overview of the DC approach, as well as a discussion on the relevant literature. Section 3 presents all steps of our methodology, namely classification, regression, and trading strategy. Section 4 presents the experimental setup, and Section 5 presents and discusses our findings. Finally, Section 6 concludes the paper and discusses directions for future work.

2. DC Background

2.1. Overview

The directional change (DC) approach is an alternative approach for summarising market price movements. A DC event is identified by a change in the price of a given financial instrument. This change is defined by a threshold value, which was in advance decided by the trader. Such an event can be either an upturn or a downturn event. After the confirmation of a DC event, an overshoot (OS) event usually follows. This OS event finishes once an opposite DC event takes place. The combination of a downturn event and a downward overshoot event represents a downward trend and, the combination of an upturn event and an upturn overshoot event represents an upturn trend. In other words, a downward trend is a period between a downturn event and the next upturn event and an upturn trend is a period between an upturn event and the next downturn event.

Figure 1 presents an example of how a physical-time price curve is transformed to the so-called *intrinsic time* [19] and dissected into DC and OS events. As we can observe, two different thresholds are used, and each threshold generates a different event series. Thus, each threshold produces a unique series of events. The idea behind the different thresholds is that each trader might consider different thresholds (price percentage changes) as significant. A smaller threshold captures a higher number of directional changes events, while a higher thresholds captures fewer directional changes events.

Looking at the events generated by a threshold of $\theta = 0.01\%$ (events connected via solid and dashed lines), we can observe that any price change less than this threshold is not considered a trend. On the other hand, when the price changes above that threshold, then the market is divided accordingly, to uptrends and downtrends. DC events are in solid lines, and OS events are in dashed lines. So an downturn DC event starts at Point A and lasts until Point B, when the downturn OS events starts. The downturn OS lasts until Point C, when there is a reverse in the trend, and an uptrend starts, which lasts until Point D. From Point D to E we are in an upturn OS event, and so on. The end/beginning price point of a new DC trend is called *DC Extreme point (DCE)*; these are Points A, C, E, and $E^{'}$.

As we mentioned, different thresholds generate different event series. Looking at $theta = 0.018\%$ (events connected via dotted and dot-dashed lines), we can observe that the events generated are different: a downward trend starts from A and lasts until B′, and the downward OS is from Point B′ until C. Then, from Point C until Point E there

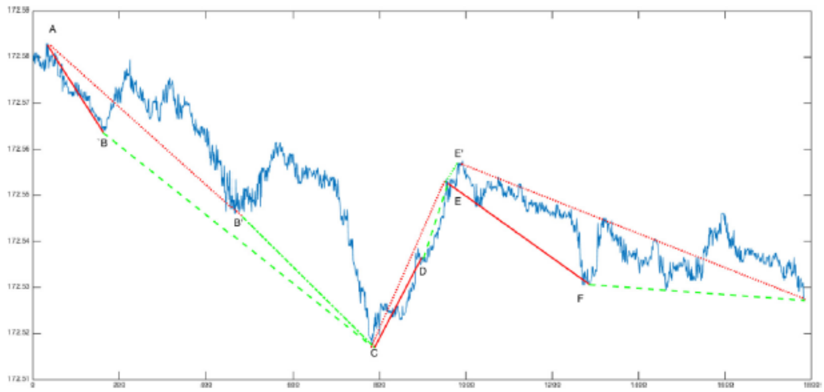


Figure 1: Directional changes for GBP/JPY currency pair. The solid and dashed lines denote a set of events defined by a threshold $\theta = 0.01\%$, while the dotted and dot-dashed lines refer to events defined by a threshold $\theta = 0.018\%$. The solid and the dotted lines indicate the DC events, and the dashed and dot-dashed indicate the OS events. Under $\theta = 0.01\%$, the data is summarised as follows: Point $A \mapsto B$ (Downward directional change), Point $B \mapsto C$ (Downward overshoot event), Point $C \mapsto D$ (Upward directional change), Point $D \mapsto E$ (Upward overshoot event), Point $E \mapsto F$ (Downward directional change). Under $\theta = 0.018\%$, the data is summarised as follows: Point $A \mapsto B^{'}$ (Downward directional change), Point $B^{'} \mapsto C$ (Downward overshoot event), Point $C \mapsto E$ (Upward directional change), Point $E \mapsto E^{'}$ (Upward overshoot event). Points A, C, E, and $E^{'}$ are DCE points (DC Extreme). Points B, $B^{'}$, D, E, and F are called DCC points (DC Confirmation).

is an upward DC trend, and from E to E′ there's an upward OS trend. Algorithm 1 presents the high-level pseudocode for generating directional changes events.

It is important to note here that the confirmation of a change of a trend can only be confirmed retrospectively, i.e. only after the price has changed by the pre-specified DC threshold value θ. For example, under $\theta = 0.01\%$ we can only confirm that we are in a upward trend from Point D onwards. Point D is thus called a *DC Confirmation point (DCC)*. Before Point D, the directional change had not been confirmed (i.e. the market price had not changed by the pre-specified threshold value), thus a trader summarising the data by the DC paradigm would continue believing we are in a downward trend, which started from Point A. Similarly, a trader using $\theta = 0.01\%$ would continue considering being in a upward trend from Point D until the price has reversed by $\theta = 0.01\%$, which only takes place at the next confirmation point, i.e., Point F. So what becomes important here is to be able to anticipate the change of the trend as early as possible, i.e. before Points C and E have been reached. In addition, since different thresholds generate different event series, we hypothesise that the combined information from these series would lead to profitable trading strategies.

The advantage of this new way of summarising data is that it provides traders with new perspectives to price movements, and allows them to focus on those key points that an important event took place, blurring out other price details which could be considered irrelevant or even noise. Furthermore, DC have enabled researchers to discover new regularities in markets, which are not captured by the interval-based summaries [19]. Therefore, these new regularities give rise to new opportunities for traders, and also open a whole new area for research.

Next, we present a review of DC literature.

Algorithm 1 Pseudocode for generating directional changes events given threshold Δx_{dc}.

Require: Initialise variables (event is Upturn event, $p^h = p^l = \mathrm{p}(t_0)$, $\Delta x_{dc}(Fixed) \geq 0, t_0^{dc} = t_1^{dc} = t_0^{os} = t_1^{os} = t_0$)

if event is Upturn Event **then**
    if $p(t) \leq p^h \times (1 - \Delta x_{dc})$ **then**
        $event \leftarrow Downturn\ Event$
        $P^l \leftarrow p(t)$ //Price at end time for a Downturn Event
        $t_1^{dc} \leftarrow t$ //End time for a Downturn Event
        $t_0^{os} \leftarrow t + 1$ //Start time for a Downward Overshoot Event
    else
        if $p^h < p(t)$ **then**
            $p^h \leftarrow p(t)$ //Price at start of Downturn event
            $t_0^{dc} \leftarrow t$ //Start time for Downturn event
            $t_1^{os} \leftarrow t - 1$ //End time for a Upturn Overshoot Event
        end if
    end if
else
    if $p(t) \geq p^l \times (1 + \Delta x_{dc})$ **then**
        $event \leftarrow Upturn\ Event$
        $P^h \leftarrow p(t)$ //Price at end time for upturn event
        $t_1^{dc} \leftarrow t$ //End time for a Upturn Event
        $t_0^{os} \leftarrow t + 1$ //Start time for a Upturn Overshoot Event
    else
        if $p^l > p(t)$ **then**
            $p^l \leftarrow p(t)$ //Price at start time for upturn event
            $t_0^{dc} \leftarrow t$ //Start time for a Upturn Event
            $t_1^{os} \leftarrow t - 1$ //End time for a Downturn Overshoot Event
        end if
    end if
end if

2.2. Review of DC literature

One could divide the DC literature into two main categories: (i) scaling laws discovery, and (ii) trend reversal estimation and trading. Below we present the relevant literature for these two categories.

2.2.1. *Scaling laws discovery*

According to [20], scaling laws is an economics term used in referring to empirical findings accepted as truth due to their consistency. The first work that attempted to find scaling laws using DC technique was [19], which discovered that "the size of mean return is scale-invariant to the interval time of occurrence". Furthermore, [21] discovered 17 new scaling laws. In another work, [15] uncovered 12 additional empirical scaling laws through empirical observations in Forex price data. Moreover, [22] described how the DC approach can be used as a tool to improve our understanding of the dynamic behaviour of the financial market. They observed that events of different magnitude are independent of physical-time changes. Their experimental result indicated that summarizing price movements using intrinsic time gives a different insight into how price changes. In [23], 4 additional scaling laws were discovered, which were successfully applied to investigate the impact of different strategies on trading activities in high-frequency Forex market. Further work by [24] contributed 5 additional scaling laws. [17] introduced 4 indicators and showed how these indicators helped profile the markets. In a subsequent work, [25] introduced a new indicator for detecting regime changes. This work was able to detect regimes changes undetectable under physical time series. In a follow up paper in [26] it was observed that normal regime change in different markets have statistical similarities. Based on these statistical similarity findings this work was able to differentiate between normal and abnormal regime changes irrespective of the market. [27] validated some of the scaling laws discovered in Forex market data in Stock market data. It concluded that these scaling laws hold in stock market data as well. Scale of Market Quakes (SMQ), a way of sizing the impact of economic or political development and other major breaking news on price movement within the Forex market was proposed by [28]. Their work on SMQ aimed to set a foundation for creating a metric that can be used to measure price change when major world events occurs. Finally, the DC approach has also been used in observing volatility seasonality in cryptocurrencies [29, 30]. These two articles successfully observed some of the aforementioned scaling laws in cryptocurrency and demonstrated the robustness of DC approach in virtual currency market.

2.2.2. *DC Trend reversal estimations and trading*

Trend is the perceived tendency of financial markets to move in a particular direction over time. Trend reversal is a change in the direction from either upwards to downwards or vice-versa. A successful trading strategy is expected to identify trading opportunities in trends that minimises risks and maximises profits. To achieve this fit, a trend reversal estimation algorithm is a crucial component a trading strategy most have. [31] developed agents that model trader's behaviour in Forex market. Their work focused on establishing scaling laws regarding how traders react and adapt to changes in Forex market. Their agents used strategies known as ZI-DC0 developed by combining DC approach with trend following and contrary trading technical indicators. [32] further proposed a new trading strategy called ZI-DC1 as an improvement to the study in [31], Comparison results between ZI-DC0 and ZI-DC1 showed that ZI-DC1 was more profitable. [33] introduced an automated trading strategy (DCT2) that can perceive changes in market conditions and adapt dynamically to remain profitable. [34] proposed a DCbased trading strategy named 'DBA'. DBA opened a position within OS event region if the magnitude of price change is greater than a certain threshold and closed the position at the next DC trend's DCC point. They embedded their trend reversal estimation technique in a contrarian trading strategy and tested using 3 currency pairs. Their results showed a mean return of around 14%. [35] proposed an improvement to their 'DBA' based trading strategy to intelligently determine the quantity to trade and evaluate the risk profile of potential trade. They concluded that the new approach was more profitable and less risky than DBA. [36] developed a neuro-fuzzy logic based trading strategy that captures volatility of DC trends. They embedded a future price estimation algorithm in their trading strategy to predict the future price of an asset based on the current price and the immediate past 3 consecutive observations in the market. Their trading strategy's returns outclassed the physical-time scale trading strategies they compared with. [37] attempted to solve a DC trend forecasting problem using classification. Their goal was to establish the estimative power in directional changes approach. To this end, they proposed 3 new directional changes indicators derived from a technical indicator. These indicators were used in forecasting price value at OS extreme point. [38] proposed a forecasting model using two DC summaries from distinct thresholds. They used data summaries of smaller thresholds in exploiting trends in data summary of larger threshold. They embedded this forecast model in a trading strategy called TSFDC. They reported that TSFDC performed considerably well against other DC based strategies compared. [39] was the first work to use a genetic programming algorithm to generate DC-based trading strategies. Results showed that the new algorithm had the potential to outperform its competitors. [40] developed a decision tree based trading strategy that used dynamic threshold to summarise historic market data. Their idea was to adjust DC threshold according daily price evolution. They reported that their approach of summarising data was more profitable than using fixed threshold. They applied the same idea in detecting significant event in data stream [41] and concluded that detecting events using dynamically calculated threshold was more effective than using fixed thresholds. This conclusion was reached after verifying the validity of the DC events with important news in the same period when the data was sampled. [42] developed 4 strategies for closing trade positions opened at DCC point. They concluded that estimating trend reversal using trailing loss was a good techniques for guaranteeing profit. However, they were not able to show the suitability of combining DC with technical analysis as a trading strategy. [43] estimated Value-at-Risk in DC using an FGARCH model and proposed a DC-based trading strategy. They estimated trend reversal to be the sum of DC event and average OS event length. Comparison result with moving window (MW) strategy demonstrated that their trading strategy approach outperformed MW without an increase in market risk.

2.3. DC-OS event length relationships

One of the scaling law in DC summaries is the relationship between the DC event length and the OS event length [15], where the OS length can be expressed as a function of the DC length. This insight can therefore be leveraged by traders for estimating DC trend reversal. The reversal point is estimated by summing up the DC event length and the OS event length. Thus, by using this law, traders are able to devise new trading strategies to maximise return and simultaneously minimise risk in their investments decisions. This law has been utilised in different forms for DC trading purposes [15, 44, 18, 43]. More specifically, [15] proposed Equation 1 based on the empirical observation that on average OS event length is twice the length of DC event length. [44, 45] made similar observations as [15]. Nonetheless, to improve trend reversal estimation, it proposed Equation 2, which treated the OS and DC length relationships as a linear function with a C constant, where C is the average DC length for the given datasets. Because algorithms proposed in [15] and [44] focused only on linear relationships, in our previous work [18], we proposed Equation 3, which used a genetic programming (GP) algorithm to perform symbolic regression and generate both linear and non-linear relationships of DC and OS lengths. The advantage of this approach was that there were no assumptions for the relationship between DC and OS lengths, and it was up to the GP to uncover the function that describes this relationship.

$$OS_l \approx 2 \times DC_l \tag{1}$$

$$OS_l = C \times DC_l; C > 0 \tag{2}$$

$$OS_l = f(DC_l) \tag{3}$$

An interesting observation made in [18], was that it is possible for a DC trend not to have a corresponding OS event. In fact, we have observed that on certain datasets there can be as little as 14.77% of DC trends with a corresponding OS event. This is, of course, threshold-dependent. Nevertheless, the fact remains that one cannot assume that a DC event will always be followed by an OS event; it can be the case that a DC event is followed by another DC event from the opposite direction. This was an important finding because is means that none of the Equations 1-3 are taking this into account.

Thus, in this work, we propose further improvements to our previous work [18], by introducing a classification task, which is going to predict whereas a DC event will have a corresponding OS event. Only when this is true, then we will be applying a GP algorithm to perform the symbolic regression task and derive new formulas, based on Equation 3. The next section presents the classification step, along with the other two steps of our methodology, namely regression and trading strategy.

3. Methodology

As explained in the previous section, there can be a high percentage of DC events that are not followed by an OS event. Therefore, creating a symbolic GP algorithm to predict the length of an OS event as a function f of a DC

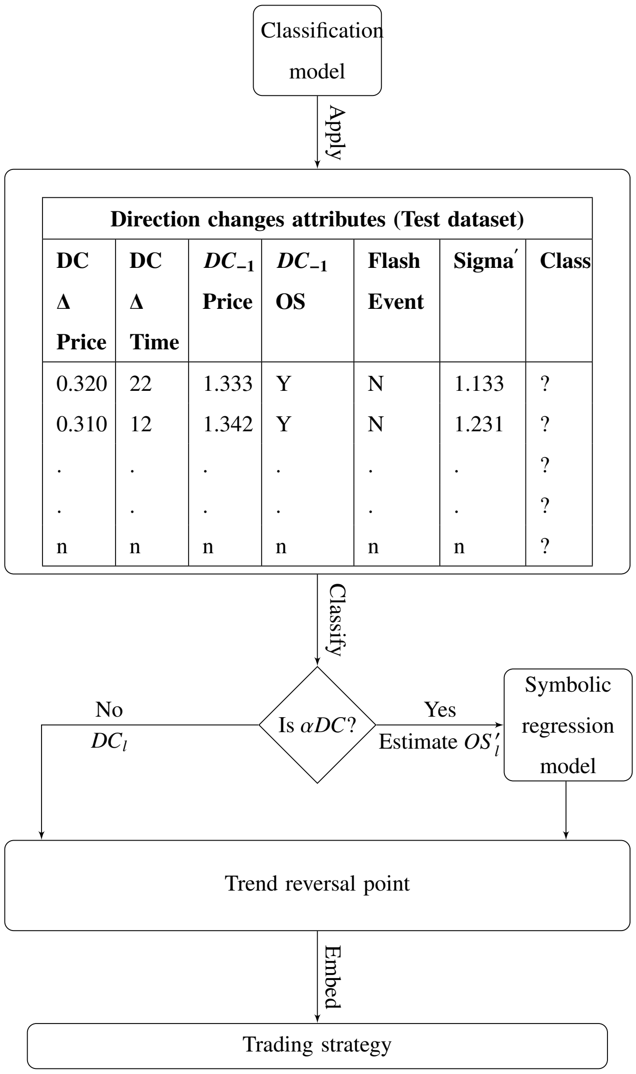


Figure 2: Our proposed framework for predicting trend reversal in DC. A DC trend classified to compose of only DC event is expected to reverse at DCC, while DC trend classified to compose of DC and OS events is expected to reverse at estimated DCE which is the sum of DC event length at DCC and GP evolved OS event length.

event (and thus predicting the end of the current trend) over the whole series of DC and OS events is an approach with a major drawback: the resulted function f that describes this relationship does not take into account that many DC events are not followed by an OS event. Hence a function in the form of $\text{OS}_l = 2\times\text{DC}_l$ would have been learnt by using inaccurate data.

What we propose in our current work is to clearly separate the cases where a DC event is followed by an OS event (we call this αDC), from those cases that a DC event is followed by another DC event of the opposite direction (e.g. a downwards DC trend is directly followed by an upwards DC trend – we call this βDC). In order to separate these cases, we will be performing classification to predict whether a DC event is followed by an OS event. If the classifier predicts that there will not be an OS event following the DC event, then there is no further action to be taken and the trend reversal point will be the end of the DC event. On the other hand, if DC is followed by OS, then we will use a symbolic regression model to represent the relationship between DC and OS lengths, and thus predict the trend reversal point, which will be at the end of the OS event. This whole process is summarised in Figure 2. Once this process has been completed, we then embed this as part of a trading strategy to evaluate its profitability.

In order to better present the above processes, we have broken down the remainder of this section as follows: Section 3.1 explains how we build a GP to perform the symbolic regression task, Section 3.2 presents the classification task, and finally, Section 3.3 presents the trading strategy.

3.1. GP - Symbolic Regression

Before going into the details of the GP algorithm, it is important to clarify that even though the GP symbolic regression model is inputed into our framework after the classification step (see Figure 2 above), in terms of implementation, it actually happens first. This is because if we perform the classification task first, then any classification errors are going to affect the effectiveness of the GP models. More specifically, let us assume that we have a dataset of 10 DC events, and that the first 8 DC events are followed by an OS event, whereas the last 2 DC events are not followed by an OS event. Let us also assume that a classifier incorrectly predicts that all 10 DC events are followed by an OS event. In this case, when we apply the GP to perform the symbolic regression task, it will be incorrectly using information (data) from all 10 events in order to construct its models. But what would have been more accurate would be applying the GP only to the DC events that have a corresponding OS event.

For this reason, we perform the GP regression task first under perfect foresight on the training dataset. So we identify those DC events with a corresponding OS event and apply the GP to that data only. The advantage of this approach is that we do not need to deal with the classification task and its corresponding classification errors. Instead, we train the GP only on data that matters, instead of also including noise (i.e. DC events that are not followed with an OS event). Furthermore, the fact that we only do this process on the training dataset avoids having any bias when we eventually apply the selected GP model to the (unseen) test data.

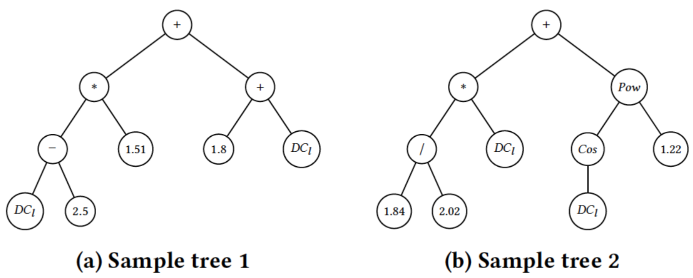


Figure 3: Sample GP individual trees: internal nodes are represented by arithmetic functions. The leaf nodes are represent by numeric constants and the DC length, denoted as DC_l . Given a DC event length the tree estimates the corresponding OS length.

3.1.1. *Model representation*

We represent our evolved GP individuals using tree structure. The inner nodes of our trees are composed of linear and non-linear functions. We utilise 2-arity functions: {addition, subtraction, division, multiplication, power} and 1-arity functions: {sine, cosine, power, log, exponential}. Division, log, exp and power functions are protected. The terminal nodes consist of an ephemeral random constant (ERC) and an external input which represented DC event length. We have selected an ERC with a probability Pr, alternatively the DC event length is selected with probability $1 - Pr$. All of our functions and terminals are presented in Table 1. To initialise our population we used ramped half-and-half technique. Figure 3 shows two sample trees from our GP. The first tree represent the equation that calculates OS length as $((DC_l-2.5)\times1.51)+(1.8+DC_l)$ and the second trees represents the equation $((\frac{1.84}{2.02})\times DC_l)+(\cos(DC_l)^{1.22})$, where DC_l in both equations is the length of DC event.

Table 1: Configuration of the proposed GP algorithm

Configuration	Value
Function set	addition, subtraction, division, multiplication, sine, cosine, power, log and exponential.
Terminal set	input variable (i.e DC events length) and ephemeral random constant.
Genetic operation	subtree mutation and subtree crossover.

3.1.2. *Model evaluation*

It is important to highlight here that the confirmation of a change from upward DC trend to downward DC trend and vice-versa can only be confirmed retrospectively after the price has changed by the pre-specified DC threshold value. Once a directional change is confirmed and a DC trend is classified to be compose of both DC event and OS event, traders are better informed on the potential point in time when DC trend is expected to reverse if OS event length can be adequately estimated. This potential point in time will be the sum of DC event length known at DCC (DC Confirmation point) and OS event length of the αDC estimated using our GP model. To evaluate our GP model we measure the error between actual OS length (OS_l) and estimated OS length ($\hat{OS}_l$). To describe our model performance, we measure the error ε using RMSE shown in Equation 4.

$$\varepsilon = \sqrt{\frac{\sum_{i=1}^{N}(OS_l - \hat{OS}_l)^2}{n}} \tag{4}$$

where n is the sample size.

During evolution, we penalise trees that have only constants as terminal nodes, trees that estimate a negative value and trees that evaluate fitness to NaN or infinity. We perform tournament selection and select parents based on fitness level. We also consider tree depth as a secondary selection criteria in cases where the lowest RMSE is attained by more than one tree. We give preference to the tree with a shorter depth.

3.1.3. Operators and other parameters

We use elitism, subtree mutation and subtree crossover (see Table 1). To control growth, we use hard limits on the depth of offspring programs generated. $Maximum\_depth$ is used for controlling mutation operation.

We have introduced a wrapper to replace incorrect predictions of OS_l with 0. This value was necessary because it was possible for GP to predict negative, NaN or infinite OS length value. We chose the value of 0 because after empirical observations, we realised that there were cases where a DC event is directly followed by another DC event of the opposite direction, hence the OS length of the preceding DC event was 0.

3.1.4. GP outputs

So far we have discussed using the GP to perform the regression task on DC-based data. However, a problem we faced was which DC threshold to use. As we have already mentioned, different DC thresholds produce completely different different DC event series. For example, as we showed in Figure 1, a 0.01% threshold produces different event series to a 0.018% threshold.

In order to decide which thresholds to use during our experiments, we create a pool of DC thresholds. Each threshold then generates a different DC event series. Then, we apply the GP to each one of these event series. As a result we obtain a GP symbolic regression model for each DC event series. Lastly, we rank all GP models in terms of RMSE. This allows us to obtain two outputs: the best GP model and its corresponding DC event series (threshold). This process is summarised in Figure 4.

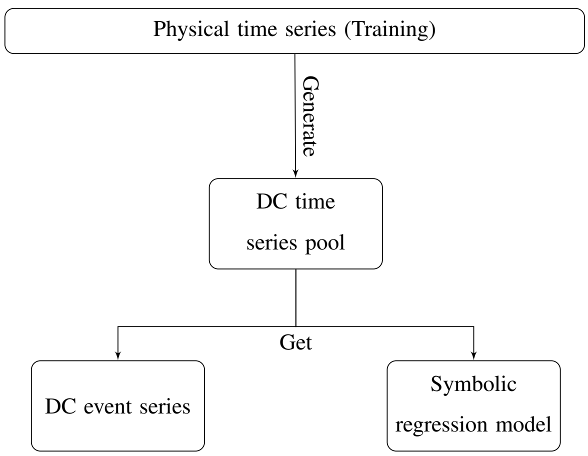


Figure 4: Our proposed framework for evolving symbolic regression model and selecting DC event with high DC:OS event ratio.

3.2. Classification

The next step in our framework is to classify whereas a DC trend is composed by a DC event and an OS event (αDC), or if a DC trend is solely composed of a DC event (βDC). As there are numerous classification algorithms that can be used for this task, we decided to use Auto-Weka [46], which considers 39 different classification algorithms, as well as their hyperparameters. We apply Auto-Weka to each dataset, thus each dataset can end up using a different classification algorithm. The advantage of this is that we have a tailored classification algorithm and tailored hyperparameters for each dataset. To avoid any bias, Auto-Weka is only applied to the validation dataset and is validated using 10 k-fold cross-validation. In order to decide which classification model to use for each dataset, we run Auto-Weka 10 independent times per dataset, and in the end we select the classification model with the best f-measure.

The attributes used for the classification task are all DC-related and are presented in Table 2. As we can observe, we use 6 different attributes, which are related to DC and OS events price and time, as well as the speed of price changing. Attributes X1, X2, were first derived and presented in [15]. Furthermore, attributes X3, X4, X5 and X6 were created for the purposes of this paper, after experimenting with a set of different attributes and identifying the ones with the best classification performance on a training dataset.

The classification process is summarised in Figure 5. As we can see we use the 'Best DC event series' as input. This is essentially one of the outputs of the GP process presented in Figure 4. We thus use this event series to create DC attributes for our classification task. Auto-Weka is then applied to these attributes and at the end we obtain the best classification model for each dataset. A reminder that this whole process takes place for the training data.

Once the classification process is also completed, we are then ready to predict the end of a trend in the (unseen) test set by combining the outputs of the classification and the regression steps. What we therefore do is apply the classification model obtained from the training data to the test dataset and classify whether a DC trend is composed by both a DC and OS event (αDC), or not. If the answer is no, then we can predict that the end of the trend will be at the end of the current DC event (DC_l). On the other hand, if a DC event is followed by an OS event, then we can use the symbolic regression model obtained in Section 3.1 and predict the trend reversal point, which is the sum of the DC and OS lengths. This process was also illustrated earlier in Figure 2, which was presented earlier at the beginning of Section 3.

3.3. Trading Strategy

The first two steps presented in the previous two sections allow us to predict the end of a trend in DC event series. To understand how effective this prediction is, we need to use it as part of a trading strategy. In order to do this, we embed our trend reversal prediction process into a trading strategy. For the remainder of this section, we present the trading strategy we use in our experiments.

Attributes	Name	Description
X1	DCevent price	This is the price difference between the upturn/downturn point and the directional change confirmation point.
X2	DCevent time	This is the time difference between the upturn/downturn point and the directional change confirmation point.
X3	$Sigma'$	This is the speed at which price change from the start of a trend to directional changes confirmation point.
X4	$DCevent_{-1}$ price	This is the market price at the previous confirmation point.
X5	$DCevent_{-1}$ OS	This is a Boolean variable (Yes/No). Indicates whether the immediate previous DC trend has an OS event.
X6	Flash event	This is a Boolean variable (Yes/No). Indicates whether DC event start time and end time are equa.l

Table 2: Classification attributes - A brief description of independent variables used for classifying whether a DC trend has OS event or not.

3.3.1. Trading strategy overview

In order to decide how to trade, we differentiate between opening and closing a position. Opening a position means we sell the base currency and buy the quoted currency. Closing a position means we buy the base currency and sell the quoted currency. We open a position at upward DC trends, provided there is not an existing open position and return is positive after deducting transaction costs. We close a position if there is an existing open position and return is positive after deducting transaction costs. In all other cases we adopt a hold trading strategy. All transactions are done using our entire capital. The transaction cost is 0.025% per transaction. The Opening and Closing strategies are summarised in Algorithms 2 and 3.

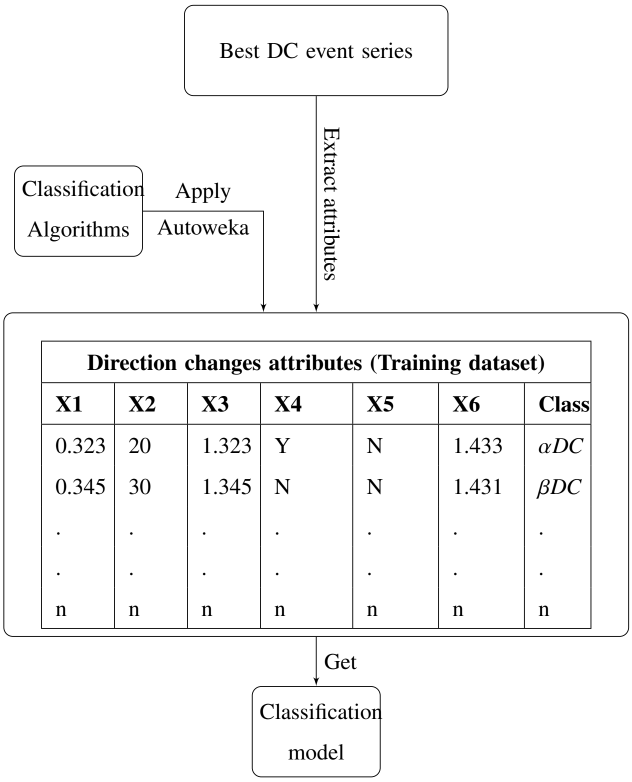


Figure 5: Our proposed framework for creating classification model. The classification model classifies DC trends into αDC and βDC

Algorithm 2 Trading rules used for selling base currency

Require: Sell rule
  if DC trend is upward **then**
    if There is no open position **then**
      if Is βDC && Return is not negative **then** Open a position at DCC point
      else if Is αDC && DC trend does not reverse before estimated DCE point && Return is not negative **then** Open a position at estimated DCE point
      else Hold
      end if
    end if
  end if

3.3.2. Trading strategy evaluation

To evaluate our trading strategy, we measure profitability and risk. We report return, mean maximum drawdown (MDD) and sharp ratio. Return, shown in Equation 5 is the accumulated profit or loss during our trading period and it is calculated by deducting transaction cost from quantity of Forex available for trading and multiplying the result with the exchange rate. Transaction cost shown in Equation 6 is the expense incurred for effecting a trade transaction, in

Algorithm 3 Trading rules used for buying base currency

Require: Buy rule

if DC trend is downward **then**

  if There is an open position **then**

    if Is βDC && Return is not negative **then** Close position at DCC point

    else if Is αDC && DC trend does not reverse before estimated DCE point && Return is not negative **then** Close position at estimated DCE point

    else Hold

    end if

  end if

end if

our case the sale or purchase of a foreign currency and it is calculated as 0.025% of the quantity of Forex available for trading. MDD shown in Equation 7 is the downside risk of our strategy and it is measured by calculating the maximum observed loss from a peak price to a trough before a new peak is reached. To measure our excess return, above the risk-free rate, we use Sharpe ratio shown in Equation 8. It measure the amount of risk involved in obtaining our returns. It is calculated by deduction risk-free rate from mean return and dividing the result by the standard deviation. In this work, we assign 0 as our risk-free rate.

$$R = (Q - TC) \times FXrate \tag{5}$$

$$TC = Q \times \frac{0.025}{100} \tag{6}$$

$$MDD = \frac{P_{trough} - P_{peak}}{P_{peak}} \tag{7}$$

$$SharpeRatio = \frac{R - RFR}{\sigma_{return}} \tag{8}$$

where R is the return, Q is the quantity, TC the transaction cost, $FRrate$ the FX rate of the relevant currency pair, MDD is the Maximum Drawdown, P_{trough} the trough of the price, P_{peak} the peak of the price, RFR the risk free rate, and σ_R is the standard deviation of the return.

4. Experimental setup

This section is divided into the following parts: Section 4.1 presents the data we are using for the experiments, Section 4.2 presents the tuning configurations for the classification and regression tasks of our framework, and lastly, Section 4.3 presents the setup of the trading experiments.

4.1. Data

We used 10-minute interval high frequency data from March 2016 to February 2017 of the following currency pairs: AUD/JPY (Australian Dollar and Japanese Yen), AUD/NZD (Australian Dollar and New Zealand Dollar), AUD/USD (Australian Dollar and US Dollar), CAD/JPY (CAD Dollar and Japanese Yen), EUR/AUD (Euro and Australian Dollar), EUR/CAD (Euro and Canadian Dollar), EUR/CSK (Euro and Czechoslovak koruna), EUR/NOK (Euro and NOK), GBP/AUD (British Pound and Australian Dollar), NZD/USD (New Zealand Dollar and US Dollar), USD/CAD (US Dollar and Canadian Dollar), USD/NOK (US Dollar and Norwegian Krona), USD/JPY (US Dollar and Japanese Yen), USD/SGD (US Dollar and Singaporean Dollar), USD/ZAR (US Dollar and South African Rand), EUR/GBP (Euro and British Pound). We also used 10-minute interval data from June 2013 to May 2014 of the following currency pairs: EUR/USD (Euro and US dollar), EUR/JPY (Euro and Japanese Yen), GBP/CHF (British Pound and Swiss Franc), and GBP/USD (British Pound and US dollar). All data was purchased from OLSENDATA.com. We considered each month in the period as a separate physical-time dataset. In our tuning phase we used 200 DC datasets for tuning (i.e. 5 thresholds $\times$ 20 currency pairs $\times$ first 2 months of our physical-time data). For the rest of our experiment we use 1000 DC dataset (i.e. 5 DC thresholds $\times$ 20 currency pairs $\times$ remaining 10 months of our physical time datasets). Our tuning and non-tuning DC dataset were split in 70:30 ratio as our training and testing sets.

As different DC thresholds produce different DC event series, we have chosen to evaluate 5 different thresholds for all tuning and non-tuning DC datasets. These thresholds are the best 5 thresholds that are dynamically selected during the GP regression step presented in Section 3.1. In the results section, we will be reporting the average performance of each algorithm, over these 5 DC thresholds.

4.2. Regression and classification algorithms' tuning

The only parameter of Auto-Weka that required tuning was its execution time. This is because Auto-WEKA requires to be given enough time to search its algorithms and hyperparameter space for a classification model that is best in predicting our two class labels (αDC, βDC). We experimented with different runtime configurations namely 15 minutes, 30 minutes, 45 minutes 60 minutes, 75 minutes. We chose a runtime of 60 minutes based on average f-measure, which we observed to diminish at a runtime of 75 minutes. Depending on the number of CPU cores available, it is possible to execute Auto-Weka in serial or parallel mode. For our experiment we executed Auto-Weka in serial mode, using 1 CPU core.

We tuned GP population size, number of generations, tournament size, crossover probability and maximum depth parameters using the I/F-Race package[47]. I/F-Race package is based on an iterated racing procedure, which is an extension of the Iterated F-race procedure. It implements racing methods for the selection of the best configuration for an optimisation algorithm by empirically selecting the most appropriate settings from a set of instances of an optimisation problem[48]. Table 3 presents our GP parameter configuration determined using I/F-Race.

Parameter	
Population	500
Generation	37
Tournament size	3
Crossover probability	0.98
Mutation probability	0.02
Maximum depth	3
Elitism	0.10

Table 3: Regression GP experimental parameters for detecting DC-OS relationship, determined using I/F-Race.

4.3. Trading experimental setup

As we have already explained, after completing the classification and regression tasks, our framework can make a prediction on when a DC trend ends. We can the n embed this as part of the trading strategy we presented in Section 3.3. This is our proposed algorithm and is called it C+GP+TS.

In order to evaluate the efficiency of our proposed algorithm, we will be comparing it with several other benchmarks. Below we present in detail the different algorithms that we use to benchmark our approach. These benchmarks can be separated into two categories: DC-related algorithms, and non-DC-related algorithms.

4.3.1. DC-related benchmarks

O+TS. This is a DC trend reversal approach originally presented in [15], where it was observed that on average OS event length is twice the DC event length. In this trading strategy, instead of embedding our classification and regression steps, we embed Equation 1. Thus, the trend reversal point becomes the point where the OS event length is twice the DC length.

M+TS. This is a DC trend reversal approach originally presented in [44], where a constant was used to describe the linear relationship between DC and OS length. This constant was tailored to each dataset and separate ratios were calculated for upward trends and downward trends. In this trading strategy, instead of embedding our classification and regression steps, we embed Equation 2. Thus, the trend reversal point is tailored to each dataset.

GP+TS. This is a DC trend reversal approach presented in [18], where symbolic regression GP is used to evolve an equation which represents the ratio between DC event length and OS event length in a dataset. This is the predecessor of our proposed approach, as while it includes the regression step, it does not have the classification step. In this trading strategy, we embed Equation 3.

4.3.2. Non-DC benchmarks

Technical analysis trading strategy. Technical analysis trading strategies is a very popular approach in trading. It uses technical indicators, for insight into when to make trading decisions. We experiment with two trading strategies that utilise exponential movement average indicator (EMA) and Moving Average Convergence Divergence (MACD), respectively.

Buy and hold. Buy and hold is a well-known benchmark for trading algorithms. Under this trading strategy we bought the quoted currency in the first month of our non-tuning data, then sold it in exchange for the base currency after the 10 month period.

5. Result and analysis

This section presents results for our experiments. It is divided into three main sections: Section 5.1, which presents the classification results, Section 5.2, which presents the regression results, and Section 5.3, which presents the trading results.

We would like to remind the reader that the goal of our work is twofold: (i) Demonstrate that the combination of classification and regression leads to error reduction when compared to other trend reversal algorithms, and (ii) Demonstrate that our proposed DC-based trading strategy, which utilises our proposed trend reversal approach, is able to be profitable and outperform other trading strategies, both DC and non-DC-based, including from physical time, such as technical analysis and buy-and-hold. The first goal will be addressed in Sections 5.1 and 5.2, while the second goal will be address in Section 5.3.

5.1. Classification result

We measure the performance of our classification models according to accuracy, precision and recall. Table 4 presents the average results over the 5 DC thresholds and over the 10 months of data, per currency pair. The total average accuracy, precision and recall across the 20 currency pairs we experimented are 0.817, 0.842 and 0.822 respectively. The least average accuracy record currency pair was 0.780 (USD/SGD). The least average precision recorded per currency pair was 0.620 (EUR/CSK), however the remaining 19 average results were all above 0.820. The least average recall recorded per currency pair was 0.419 (EUR/CSK), however the remaining currency pair averages were above 0.713. These results are very important, because they will allow the GP, the next step in our proposed framework, to perform regression only on DC trends that are classified to have a DC and a corresponding OS event (αDC). Therefore, the fact that we have such high values of accuracy, precision and recall for most currency pairs will allow us to obtain better results when predicting the end of a trend. This will become evident in the next section, where the addition of the classification step has led to a much reduced regression error.

Data	Accuracy	Precision	Recall
AUD/JPY	0.851	0.867	0.839
AUD/NZD	0.805	0.827	0.817
AUD/USD	0.829	0.851	0.832
CAD/JPY	0.820	0.825	0.828
EUR/AUD	0.821	0.833	0.866
EUR/CAD	0.839	0.850	0.890
EUR/CSK	0.557	0.620	0.419
EUR/GBP	0.825	0.857	0.857
EUR/JPY	0.821	0.858	0.874
EUR/NOK	0.818	0.841	0.811
EUR/USD	0.806	0.847	0.880
GBP/AUD	0.837	0.896	0.804
GBP/CHF	0.831	0.860	0.861
GBP/USD	0.851	0.858	0.897
NZD/USD	0.848	0.862	0.874
USD/CAD	0.797	0.841	0.829
USD/JPY	0.850	0.863	0.877
USD/NOK	0.887	0.889	0.851
USD/SGD	0.780	0.820	0.816
USD/ZAR	0.877	0.869	0.713
Average	0.817	0.842	0.822

Table 4: Average accuracy, precision and recall results. 1000 datasets consisting of 5 different dynamically generated thresholds tailored to each DC dataset, 20 currency pairs, and 10 months of 10-minute interval data for each currency pair.

5.2. Regression result

Table 5 presents the average RMSE result of the regression step over the 5 DC thresholds and over the 10 months of data, per currency pair. We predict OS event length in DC trends classified as αDC in the classification step. The table also presents currency pair average RMSE results of other OS event length estimation techniques, which we described with Equations 1, 2, and 3. From the table we see that our framework that uses the classification and GP steps (C+GP) consistently outperforms other trend reversal estimators in 13 of the 20 currency pairs. It also ranks second in five cases, behind Equation 3 (the predecessor of our proposed C+GP presented in [18], which evolved GP symbolic regression models, assuming all DC trends have a corresponding OS event). In addition, C+GP has the lowest average RMSE across all datasets, which is 18.617. This positive result confirms the strength of GP in itself to finding an equation that best represent the relationship between DC and OS event lengths. The introduction of the classification step into the GP step has been proven a successful addition, based on the average RMSE results. To support our findings, we applied Friedman's non-parametric statistical test. The null hypothesis is that the algorithms come from the same continuous distribution. For each algorithm/equation, the table shows the average rank according to the Friedman test (first column), and the adjusted p-value of the statistical test when that equation's average rank is compared to the average rank of the algorithm with the best rank (control algorithm) according to the Hommel post-hoc test (second column). As we can observe, our prosed approach of C+GP ranks first and statistically outperforms at the $\alpha = 0.05$ level Equations 1 and 2. It is also worth noting that Equation 3, which as we have mentioned is the predecessor to our approach, ranks second.

To sup up the findings so far, we can make two important observations: (i) the addition of the classification step (C+GP) to our existing algorithm that was only using regression to predict the trend reversal has significantly reduced the predictive error, and (ii) our proposed C+GP algorithm ranks first and significantly outperforms two out of the three other trend reversal equations.

Our interest now shifts to using this C+GP as the trend reversal estimation algorithm of a DC-based trading strategy, to investigate whether estimating trend reversal in this manner can lead to an increase trading profit margins.

5.3. Trading result

5.3.1. Comparison against DC-based and technical analysis algorithms

Table 7 presents currency pair summary returns of the trading strategies that we detailed in Section 4.3. We would like to draw attention of reader to cases where 0.00 is reported as return in the table, this indicates that for a given currency pair, a hold action was taken by the trading strategy in the 10 months period we experimented.

We see in the table that our C+GP+TS algorithm ranks first in 13 out of 18 currency pairs (for the remaining 2 pairs, both algorithms took a hold action, so there was no trading and no returns to report). It is also worth noting that with the exception of two currency pairs (EUR/USD, GBP/USD), in all other cases where trading took place, C+GP+TS showed positive returns over the 10-month period of our experiments. To support our findings, we applied again Friedman's non-parametric statistical test. The null hypothesis is that the algorithms come from the same continuous distribution. The result of the statistical test presented in Table 8 shows C+GP+TS ranking the highest, and that the difference in ranking is statistically significant when it is compared to all other algorithms at the 5% level.

In Table 9 we report the maximum return per currency pair (i.e. 10 months $\times$ 5 thresholds) of the different trading strategies compared. C+GP+TS has the highest maximum return obtained on average over the 20 currency pairs. It also has the maximum returns in 9 currency pairs. In similar manner, Table 10 presents minimum return per currency

Algorithms	C+GP	Equation 1	Equation 2	Equation 3
AUD/JPY	**15.567**	22.269	25.527	15.627
AUD/NZD	27.368	41.592	51.242	**24.332**
AUD/USD	**11.580**	14.060	16.095	12.814
CAD/JPY	**11.843**	27.251	39.970	18.785
EUR/AUD	21.171	**19.749**	25.728	20.569
EUR/CAD	**16.205**	24.867	23.183	17.719
EUR/CSK	**41.990**	83.845	188.608	52.949
EUR/GBP	24.173	**18.790**	31.430	22.635
EUR/JPY	**19.965**	25.204	28.162	21.117
EUR/NOK	**13.717**	22.499	27.201	13.762
EUR/USD	**28.260**	30.038	38.532	31.061
GBP/AUD	15.138	17.910	21.670	**14.719**
GBP/CHF	**15.961**	23.669	19.358	17.204
GBP/USD	**19.204**	27.778	21.223	24.889
NZD/USD	**10.230**	15.896	14.731	10.588
USD/CAD	26.934	29.315	34.654	**26.818**
USD/JPY	**13.704**	18.326	17.998	14.543
USD/NOK	7.718	10.764	14.128	**7.357**
USD/SGD	**26.932**	34.360	41.712	34.148
USD/ZAR	5.440	7.720	7.796	**4.796**
Average	**18.617**	25.795	34.477	20.265

Table 5: Average RMSE values for each OS length estimator algorithm. 1000 datasets consisting of 5 different dynamically generated thresholds tailored to each DC dataset, 20 currency pairs, and 10 months of 10-minute interval data for each currency pair. Best result per currency pair presented in boldface.

Algorithm	Average Rank	$Adjust_{pHomm}$
C+GP (c)	1.450	-
Equation 3	**1.850**	0.327
Equation 1	**3.000**	2.932E-4
Equation 2	**3.700**	1.068E-7

Table 6: Statistical test results of OS length estimation according to the non-parametric Friedman test with the Hommel post-hoc test. Significant differences at the $\alpha = 0.05$ level are shown in boldface.

pair. Results show that C+GP+TS recorded the least negative return amongst DC based trading strategies over the 20 currency pairs we experimented with. However, amongst all trading strategies, EMA recorded the least negative return; it had the least negative return in 10 of the 20 currency pairs.

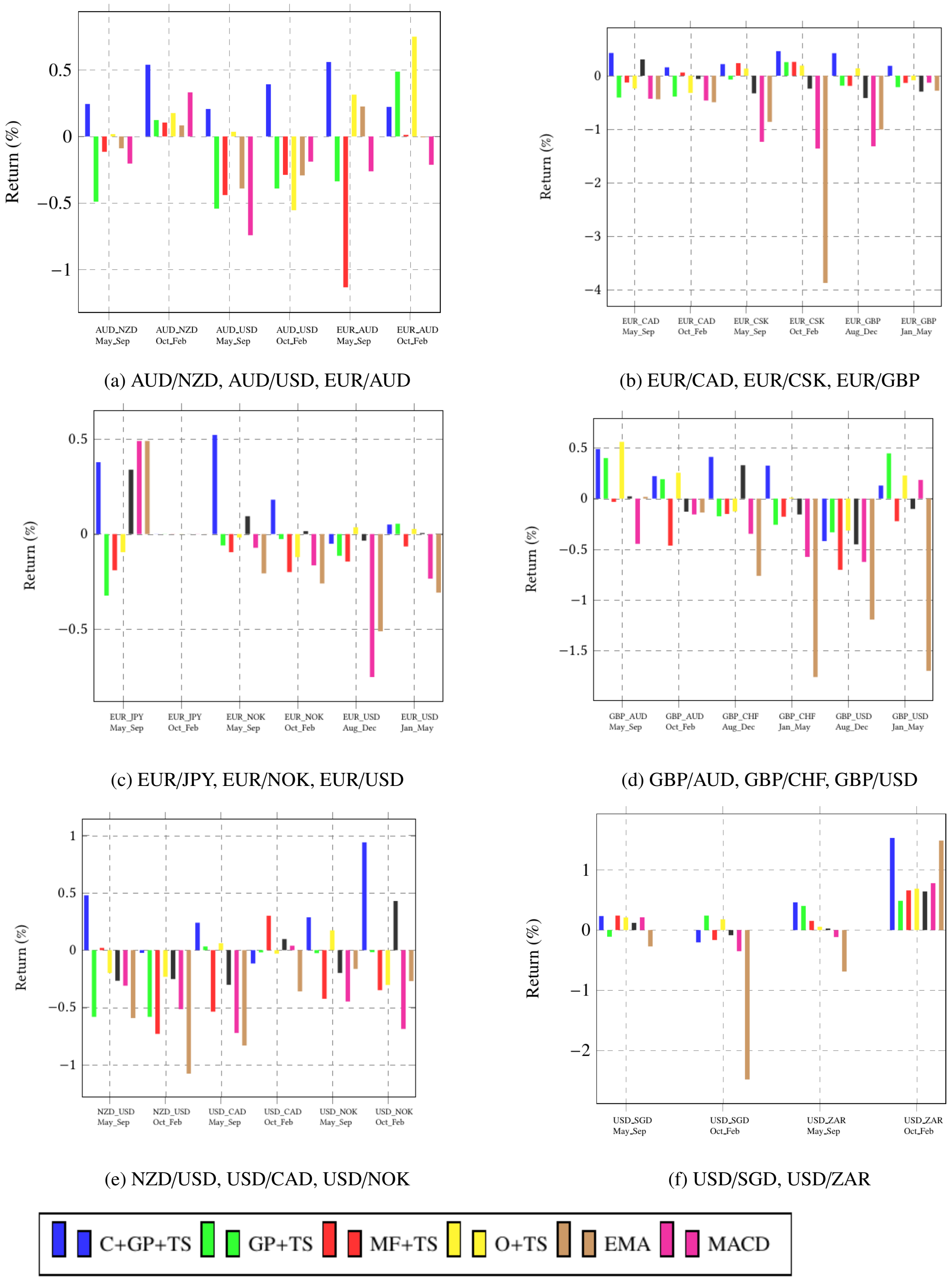
Even though C+GP+TS recorded higher return than the other trading strategies, it is important to measure the risk taken to achieve it. For this reason we also present results of our risk measures, namely MDD and Sharpe ratio. We did not record risk measures for currency pairs of AUD/JPY, CAD/JPY and USD/JPY, as no trading took place in these markets. Table 11 presents the MDD result. As we can observe, C+GP+TS has the best average MDD of 0.107, outperforming again all other algorithms. In terms of ranking, C+GP+TS ranks again first, and significantly outperforms M+TS and O+OS at the 5% significance level, as it can be seen in Table 12.

Furthermore, Figure 6 presents a chart that details the Sharpe ratio results of the trading strategies. The x-axis presents the time period covered for the relevant currency pair, and the y-axis presents average risk-adjusted return in percentages. The pairs where there are no values are the ones where no trading took place. As we can observe, C+GP+TS is the trading strategy that consistently has positive adjusted returns, whereas the other strategies have a lot of negative returns. Excluding 6 periods where no trading took place, there are 34 risk-adjust return summaries in total. Out of the 34, C+GP+TS had positive Sharpe ratio in 28. Meanwhile, GP+TS, MF+TS, O+TS, EMA and MACD had 11, 7, 17, 4 and 12 respectively. Of the 28 positive Sharpe ratio results, 6 where above 0.5, 18 were above 0.2 and the rest were below 0.2.[1] Table 13, which presents the Friedman test, confirms our findings, as C+GP+TS again ranks first and statistically outperforms the majority of the other trading strategies at the 5% level.

5.3.2. Comparison with Buy-and-hold

Since C+GP+TS has been shown to be the best algorithm across all other DC and technical analysis algorithms, we will now compare it with the well-known buy-and-hold (BandH) benchmark. The reason we are doing this comparison separately is because for BandH we do not have 10 monthly datasets, as we did with all other algorithms. Instead, we simply buy on the first day of the first month, and sell on the last day of the tenth month.

[1] A ratio of 0.2-0.3 is in line with the general market. A value of 0.5 is considered a market-beating performance if achieved over a long period, a ratio of 1 or better considered superb and difficult to achieve over long periods and a negative Sharpe ratio indicates negative returns.



(a) AUD/NZD, AUD/USD, EUR/AUD

(b) EUR/CAD, EUR/CSK, EUR/GBP

(c) EUR/JPY, EUR/NOK, EUR/USD

(d) GBP/AUD, GBP/CHF, GBP/USD

(e) NZD/USD, USD/CAD, USD/NOK

(f) USD/SGD, USD/ZAR

Figure 6: Average Sharpe ratio for all currency pairs.

Trading strategies	**C+GP+TS**	**GP+TS**	**M+TS**	**O+TS**	**EMA**	**MACD**
AUD/JPY	0.000	0.000	0.000	0.000	0.000	0.000
AUD/NZD	**0.260**	-0.089	0.00	0.068	0.002	0.005
AUD/USD	**0.273**	-0.464	-0.175	-0.173	-0.145	-0.147
CAD/JPY	0.000	0.000	0.000	0.000	0.000	0.000
EUR/AUD	0.186	-0.039	-0.031	**0.312**	0.057	-0.092
EUR/CAD	**0.192**	-0.243	0.02	-0.109	-0.226	-0.346
EUR/CSK	**0.034**	0.01	0.013	0.012	-0.233	-0.281
EUR/GBP	**0.104**	-0.087	-0.045	-0.008	-0.135	-0.240
EUR/JPY	**0.020**	-0.062	0.012	**0.020**	0.015	0.013
EUR/NOK	**0.351**	-0.043	0.041	-0.044	-0.118	-0.233
EUR/USD	-0.001	**0.020**	-0.07	-0.041	-0.492	-0.409
GBP/AUD	**0.354**	0.296	-0.157	0.286	-0.302	-0.061
GBP/CHF	**0.202**	-0.116	-0.065	-0.037	-0.268	-0.331
GBP/USD	-0.059	**-0.048**	-0.188	-0.18	-0.076	-0.361
NZD/USD	**0.280**	-0.478	-0.339	-0.341	-0.234	-0.366
USD/CAD	**0.044**	0.011	-0.145	0.03	-0.306	-0.571
USD/JPY	0.000	0.000	0.000	0.000	0.000	0.000
USD/NOK	**0.461**	-0.021	0.064	0.055	-0.075	0.154
USD/SGD	0.03	0.027	0.018	**0.088**	-0.044	-0.295
USD/ZAR	**1.762**	0.84	0.368	0.293	0.344	0.110
Average	**0.225**	-0.024	-0.034	0.012	-0.112	-0.188

Table 7: Average return result for trading strategies compared. 10-minute interval out-of-sample data. 20 different currency pairs and 10 calendar months each representing the physical dataset. 5 DC dataset were generated using 5 dynamically generated thresholds tailored to each DC dataset. Best result per currency pair presented in boldface.

Table 14 presents comparison trading result between C+GP+TS and BandH. C+GP+TS recorded positive mean annual return in 15 of 20 currency pairs. It is outperforming BandH in 12 currency pairs. C+GP+TS's average return across all currency pairs iss 0.225% and that of BandH was -0.121%. C+GP+TS reported a variance of 0.153 and BandH's reported a variance of 0.515. This result shows us that C+GP+TS is more profitable and less risky that

Trading strategies	Average Rank	$Adjust_{pHomm}$
C+GP+TS (c)	1.675	-
O+TS	**3.335**	0.028
M+TS	**4.15**	0.002
GP+TS	**4.350**	7.096E-4
EMA	**4.85**	7.888E-5
MACD	**5.85**	3.533E-8

Table 8: Statistical test results of returns according to the non-parametric Friedman test with the Hommel post-hoc test. 10-minute interval out-of-sample date. Significant differences at the $\alpha = 0.05$ level are shown in boldface.

BandH. Finally we performed the Kolmogorov-Smirnov statistical test to investigate whether there is a statistical significance in the results between C+GP+TS and BandH. The p-value of the test was 7.2529e-04, which confirms the statistical significance between the two results. Thus, the fact that C+GP+TS is more profitable and less risky, outperforming BandH in more markets makes a more attractive investment strategy.

5.3.3. A sample of best GP models

For completeness, we present some of the equations that our symbolic regression GP evolved below, in their equation format (and not in their tree format). OS_l is the OS length and DC_l is DC length. These four examples are four of our best trees in terms of profitability over all our datasets.

$$OS_l = \log(a + DC_l^b) \tag{9}$$
where a= 1609.55 and b = 5.023.

$$OS_l = \log((DC_l \times a)^b) \tag{10}$$
where a= 4.117 and b = 5.764.

$$OS_l = \cos(a \times \cos(DC_l)) + \frac{b}{exp(\cos(DC_l)} \tag{11}$$
where a = 292.160 and b= 4.569

$$OS_l = \exp(\exp(\sin(\sin(DC_l)))) + (a \times (b + \log(DC_l))) \tag{12}$$
where a = 1.750 and b = 1.957.

As we can observe, most equations have different structures. The first two are logarithmic equations, whereas the third has both the cosine and the exponential functions, and the fourth equation has exponential, sine and logarithmic

Trading strategies	C+GP+TS	GP+TS	M+TS	O+TS	EMA	MACD
AUD/JPY	0.000	0.000	0.000	0.000	0.000	0.000
AUD/NZD	**2.325**	1.857	0.910	1.690	0.598	0.641
AUD/USD	**3.551**	1.481	0.965	1.236	0.324	0.445
CAD/JPY	0.000	0.000	**0.304**	0.000	0.000	0.000
EUR/AUD	**1.763**	1.380	0.867	1.368	0.950	0.458
EUR/CAD	**2.119**	1.193	0.663	1.066	0.658	0.764
EUR/CSK	0.424	**0.449**	0.234	0.231	0.057	0.000
EUR/GBP	0.906	**1.044**	0.754	0.922	0.541	0.554
EUR/JPY	0.433	0.557	**1.575**	0.511	0.154	0.135
EUR/NOK	**2.534**	1.216	0.980	1.031	0.081	0.853
EUR/USD	1.004	0.926	**1.151**	0.393	0.000	0.415
GBP/AUD	2.764	**3.121**	1.391	3.106	0.255	1.413
GBP/CHF	**2.065**	1.124	1.042	1.516	0.643	0.228
GBP/USD	**1.577**	1.064	0.879	0.639	0.588	0.063
NZD/USD	**3.059**	1.896	1.167	2.153	0.489	1.017
USD/CAD	1.868	**2.104**	0.800	1.534	1.441	0.887
USD/JPY	0.000	0.000	**1.967**	0.000	0.000	0.000
USD/NOK	**3.273**	1.723	1.574	2.109	0.843	0.730
USD/SGD	1.336	1.213	**2.441**	1.662	0.460	0.513
USD/ZAR	7.129	5.045	3.160	3.680	2.603	2.323
Average Maximum	**1.907**	1.370	1.141	1.242	0.534	0.572

Table 9: % Maximum return result for trading strategies compared. 10-minute interval out-of-sample data. 20 different currency pairs and 10 calendar months each representing the physical dataset. 5 DC dataset were generated using 5 dynamically generated thresholds tailored to each DC dataset. Best result per currency pair is shown in boldface.

functions. This is particularly interesting and important, because it demonstrates that the relationship between the DC and OS lengths can be non-linear and also dependent on the dataset. Thus our work of using classification and regression to predict the OS length has allowed us to uncover this relationship for each dataset and increase the profitability of the trading strategies.

Trading strategies	**C+GP+TS**	**GP+TS**	**M+TS**	**O+TS**	**EMA**	**MACD**
AUD/JPY	0.000	0.000	0.000	0.000	0.000	0.000
AUD/NZD	-1.189	-1.559	-0.840	-1.191	**-0.482**	-1.040
AUD/USD	-1.918	-3.646	-1.374	-1.332	**-1.022**	-1.170
CAD/JPY	0.000	0.000	-0.303	0.000	0.000	0.000
EUR/AUD	-0.632	-2.024	-1.065	-0.896	**-0.605**	-0.722
EUR/CAD	-1.299	-1.801	-1.230	**-1.203**	-1.285	-2.068
EUR/CSK	-0.168	-0.202	-0.481	**-0.025**	-0.670	-0.626
EUR/GBP	-0.619	-1.206	-1.460	-1.276	**-0.503**	-0.992
EUR/JPY	0.000	-1.344	-1.300	-0.576	0.000	0.000
EUR/NOK	-0.863	-1.490	-1.172	-1.338	**-0.499**	-1.182
EUR/USD	-1.189	-0.744	-1.222	**-0.700**	-1.181	-1.102
GBP/AUD	-1.506	-2.736	-2.175	-2.055	-1.250	**-0.755**
GBP/CHF	**-0.811**	-1.542	-1.178	-1.235	-1.546	-0.994
GBP/USD	-0.938	-2.396	-2.121	-2.665	**-0.658**	-0.774
NZD/USD	-1.758	-2.556	-2.478	-2.612	-1.659	**-1.028**
USD/CAD	-1.870	-2.625	-1.363	**-1.157**	-2.010	-2.501
USD/JPY	**0.000**	**0.000**	-2.638	**0.000**	**0.000**	**0.000**
USD/NOK	-3.141	-1.644	-1.162	-1.514	**-1.136**	-1.669
USD/SGD	-0.724	-0.647	-1.921	**-0.501**	-0.567	-0.762
USD/ZAR	-3.105	-4.979	**-1.425**	-5.292	-2.164	-2.679
Average	-1.087	-1.657	-1.345	-1.278	**-0.862**	-1.003

Table 10: % Minimum return result for trading strategies compared. 10-minute interval out-of-sample data. 20 different currency pairs and 10 calendar months each representing the physical dataset. 5 DC dataset were generated using 5 dynamically generated thresholds tailored to each DC dataset. Best result per currency pair shown in boldface.

5.3.4. Computational times

Table 15 presents the average computational times for all algorithms. We should note that M+TS, O+TS, EMA, MACD, and BandH are deterministic algorithms and are thus very fast in executing (around 5 seconds). Because C+GP+TS and GP+TS are non-deterministic algorithms, their execution times vary between 5 and 70 minutes. The

Trading strategies	C+GP+TS	GP+TS	M+TS	O+TS	EMA	MACD
AUD/NZD	**0.123**	0.151	0.211	0.302	0.17	0.159
AUD/USD	0.160	0.312	0.425	0.451	**0.118**	0.17
EUR/AUD	**0.106**	0.155	0.192	0.255	0.120	0.173
EUR/CAD	0.135	0.258	0.148	0.289	**0.123**	0.148
EUR/CSK	**0.006**	0.008	0.007	0.003	0.036	0.039
EUR/GBP	0.100	**0.078**	0.182	0.203	0.242	0.196
EUR/JPY	0.011	0.038	0	0.053	0.004	**0.001**
EUR/NOK	0.133	0.148	0.202	0.247	0.114	**0.101**
EUR/USD	0.155	**0.069**	0.169	0.112	0.224	0.223
GBP/AUD	**0.191**	0.239	0.419	0.453	0.208	0.216
GBP/CHF	**0.096**	0.106	0.254	0.289	0.163	0.200
GBP/USD	0.132	0.180	0.318	0.324	**0.122**	0.270
NZD/USD	0.289	0.324	0.53	0.643	0.27	**0.263**
USD/CAD	0.168	**0.162**	0.338	0.187	0.323	0.167
USD/NOK	0.141	0.175	0.459	0.352	0.235	**0.099**
USD/SGD	0.077	**0.074**	0.105	0.088	0.127	0.079
USD/ZAR	**0.117**	0.145	0.552	0.597	0.357	0.709
Average MDD	**0.107**	0.131	0.225	0.242	0.148	0.161

Table 11: %Average maximum drawdown results for 10-minute interval out-of-sample data. 20 different currency pairs and 10 calendar months each representing the physical dataset. 5 DC dataset were generated using 5 dynamically generated thresholds tailored to each DC dataset. Best result per currency pair shown in boldface.

higher computation times for C+GP+TS were expected. This is because we had allowed AutoWeka to run for 60 minutes in order to find the best classification model per dataset, and also optimise its hyperparameters.[2] The regression task of C+GP+TS took 5.45 minutes on average, which comparable to its predecessor's 6.2 minutes. Since GP is involved, it is not surprising that it would need some time to evolve a good solution, since multiple individuals and

[2]The time taken in the classification phase of C+GP+TS went above the allotted time of 60 minutes due to CPU time slice as other processes were running on the hardware simultaneously. With the availability a dedicated hardware with sufficient CPU cores, a large speed up could be obtained by switching the classification phase from serial mode to parallel mode and also reducing the execution time. For example using a 60 core hardware and reducing executing time to around 2 minute.

Trading strategies	Average Rank	$Adjust_{pHomm}$
C+GP+TS (c)	2.059	
GP+TS	2.882	0.199
MACD	3.265	0.163
EMA	3.294	0.163
M+TS	**4.441**	8.205E-4
O+TS	**5.059**	1.469E-5

Table 12: : Statistical test results of maximum drawdown of DC based trading strategies according to the non-parametric Friedman test with the Hommel post-hoc test. 10-minute interval out-of-sample data. Significant differences at the $\alpha = 0.05$ level are shown in boldface.

Trading strategies	Average Rank	$Adjust_{pHomm}$
C+GP+TS (c)	2.412	-
GP+TS	3.012	0.213
O+TS	3.088	0.213
M+TS	**3.486**	0.031
EMA	**4.199**	7.716E-5
MACD	**4.800**	5.742E-8

Table 13: Statistical test results of the Sharpe ratio results according to the non-parametric Friedman test with the Hommel post-hoc test. 10-minute interval out-of-sample date. Significant differences at the $\alpha = 0.05$ level

generations are involved. Lastly, the trading task has the same duration across all algorithms: less than 1 second.

It is important to note here that for trading we would normally do the learning processes on the training data off-line, and then simply apply the best model to the test data. Thus, the fact that classification and regression last above 70 minutes is not a problem, since they happen off-line. On the other hand, applying the best model for trading takes less than 1 second. Therefore, we believe that given the significant improvements we have observed in returns and risk, this slower execution time is justified.

Lastly, the computation cost of C+GP+TS can be reduced by parallelising the evolutionary phase of the GP. It has actually been shown in [49] that parallelising the evolutionary tasks can reduce computational cost by up to 21 folds. Thus, C+GP+TS can have relatively fast execution times.

5.4. Summary

Based on our experimental results, we can reach the following conclusions.

Using classification algorithms and GP for symbolic regression is an effective way of predicting the trend reversal in DC summaries. As we observed in Tables 4 - 6, the very positive classification results have led to significantly

Trading strategies	**C+GP+TS**	**Buy-and-hold**
AUD/JPY	**0.000**	-6.278
AUD/NZD	**0.260**	-0.516
AUD/USD	**0.273**	-5.728
CAD/JPY	**0.000**	-4.109
EUR/AUD	**0.186**	-2.672
EUR/CAD	0.192	**18.555**
EUR/CSK	0.034	**7.770**
EUR/GBP	**0.104**	-0.292
EUR/JPY	**0.020**	-6.211
EUR/NOK	0.351	**2.046**
EUR/USD	-0.001	**8.801**
GBP/AUD	0.354	**3.936**
GBP/CHF	**0.202**	-2.395
GBP/USD	-0.059	**8.464**
NZD/USD	**0.280**	-6.443
USD/CAD	0.044	**2.345**
USD/JPY	**0.000**	-9.430
USD/NOK	**0.461**	-6.102
USD/SGD	0.030	**0.207**
USD/ZAR	**1.762**	-4.505
Mean	**0.225**	-0.128

Table 14: % Mean trading result of C+GP+TS vs Buy-and-hold trading strategies per currency pair. 10-minute interval out-of-sample data. Results show RMSE value. They are averaged over 5 different dynamically generated thresholds tailored to each DC dataset and 20 currency pairs.

reduced RMSE, ranking our proposed C+GP first against all other DC-based trend reversal algorithms.

Utilising the above trend reversal algorithm as part of a trading strategy can lead to profitable results. In fact, C+GP+TS significantly outperformed all other trading strategies in a variety of metrics, such as mean, maximum and minimum returns.

Our proposed trading strategy is one of the least risky strategies. As we saw from both MDD and Sharpe ratio

Trading strategies	C+GP+TS	GP+TS	M+TS	O+TS	EMA	MACD	
Classification	~ 65 mins	–	–	–	–	–	–
Regression	~ 5.45 mins	~ 6.2 mins	~ 30 secs	~ 20 secs	~ 3 secs	~ 3 secs	
Trading	~ 3 sec	~ 3 sec	~ 3 sec	~ 3 sec	~ 3 sec	~ 3 sec	

Table 15: Average computational times per run for C+GP+TS, GP+TS , M+TS , O+TS, EMA, MACD, and BH are deterministic algorithms and only take 1 second to execute.

results, C+GP+TS ranked first in both cases, and statistically outperformed 2 (MDD) and 3 (Sharpe ratio) other algorithms, respectively. The above thus lead us to conclude that C+GP+TS is a trading strategy with very low risk when compared to all other strategies presented in this work.

There is no generalised formula for predicting trend reversal in DC-based summaries. This is perhaps the most important finding of our work. This is because it demonstrates that each dataset can have its own unique characteristics, and predicting trend reversal requires tailored solutions and not equations that are applied to all trends, irrespectively of their characteristics.

6. Conclusion

To conclude, this paper presented a new framework, where we used different machine learning algorithms for classification and regression in DC-based summaries, to predict end of trend. This then enabled us to develop profitable and low-risk trading strategies, which were able to outperform six benchmarks, including other DC-based trading strategies, technical analysis, and buy and hold. It is important to note here that we run extensive experiments over a total of 1,000 datasets from 20 different Forex currency pairs. This thus leads us to believe that our results are not only significant, but also widely applicable.

Future work will focus on combining multiple DC thresholds under a single trading strategy. As we have explained, each DC threshold creates a different summary. In our current work, we experimented with 5 different thresholds and presented their average results. We believe that it would be interesting to combine the 'knowledge' of multiple thresholds under an optimisation algorithm, and investigate how/if the trading strategies' profitability can further increase. Other research directions could also be to create a tailored classification algorithm (instead of using Auto-Weka's out-of-the-box algorithms) and further improve our GP algorithm, to reduce the error of predicting end of trend even more, and thus lead to more profitable trading strategies.

Project code: The source code for this project can be found in a GitHub repository, at the following address:

`https://github.com/adesolaadegboye/SymbolicRegression`

References

[1] A. Brabazon, M. Kampouridis, M. O'Neill, Applications of genetic programming to finance and economics: past, present, future, Genetic Programming and Evolvable Machines. 21 (2020) 33–53.

[2] M. Kampouridis, E. Tsang, Investment opportunities forecasting: Extending the grammar of a gp-based tool, International Journal of Computational Intelligence Systems 5 (3) (2012) 530–541.

[3] M. Kampouridis, A. Alsheddy, E. Tsang, On the investigation of hyper-heuristics on a financial forecasting problem, Annals of Mathematics and Artificial Intelligence 68 (2013) 225–246.

[4] M. Kampouridis, F. E. Otero, Heuristic procedures for improving the predictability of a genetic programming financial forecasting algorithm, Soft Computing (2015) 1–16.

[5] M. Kampouridis, S.-H. Chen, E. Tsang, Market fraction hypothesis: A proposed test, International Review of Financial Analysis 23 (2012) 41 – 54.

[6] M. Kampouridis, S.-H. Chen, E. Tsang, Microstructure dynamics and agent-based financial markets: can dinosaurs return?, Advances in Complex Systems 15 (supp02) (2012) 1250060.

[7] B. Mandelbrot, H. M. Taylor, On the distribution of stock price differences, Operations research 15 (6) (1967) 1057–1062.

[8] R. C. Cavalcante, R. C. Brasileiro, V. L. Souza, J. P. Nobrega, A. L. Oliveira, Computational intelligence and financial markets: A survey and future directions, Expert Systems with Applications 55 (2016) 194–211.

[9] Y. Wan, X. Gong, Y.-W. Si, Effect of segmentation on financial time series pattern matching, Applied Soft Computing 38 (2016) 346–359.

[10] T.-l. Chen, F.-y. Chen, An intelligent pattern recognition model for supporting investment decisions in stock market, Information Sciences 346 (2016) 261–274.

[11] F.-L. Chung, T. C. Fu, R. Luk, V. Ng, Flexible time series pattern matching based on perceptually important points, International Joint Conference on Artificial Intelligence Workshop on Learning from Temporal and Spatial Data (2001) 1–7.

[12] J. Yin, Y.-W. Si, Z. Gong, Financial time series segmentation based on turning points, in: Proceedings 2011 International Conference on System Science and Engineering, IEEE, 2011, pp. 394–399.

[13] A. Azzini, C. da Costa Pereira, A. G. Tettamanzi, Modeling turning points in financial markets with soft computing techniques, in: Natural Computing in Computational Finance, Springer, 2010, pp. 147–167.

[14] M. O. Özorhan, İ. H. Toroslu, O. T. Şehitoğlu, Short-term trend prediction in financial time series data, Knowledge and Information Systems (2018) 1–33.

[15] J. Glattfelder, A. Dupuis, R. Olsen, Patterns in high-frequency FX data: discovery of 12 empirical scaling laws, Quantitative Finance 11 (4) (2011) 599–614.

[16] E. Tsang, Directional changes, definitions, Working Paper WP050-10 Centre for Computational Finance and Economic Agents (CCFEA), University of Essex Revised 1, Tech. Rep.

[17] E. P. Tsang, R. Tao, A. Serguieva, S. Ma, Profiling high-frequency equity price movements in directional changes, Quantitative finance 17 (2) (2017) 217–225.

[18] A. Adegboye, M. Kampouridis, C. G. Johnson, Regression genetic programming for estimating trend end in foreign exchange market, in: Computational Intelligence (SSCI), 2017 IEEE Symposium Series on, IEEE, 2017, pp. 1–8.

[19] D. M. Guillaume, M. M. Dacorogna, R. R. Davé, U. A. Müller, R. B. Olsen, O. V. Pictet, From the bird's eye to the microscope: A survey of new stylized facts of the intra-daily foreign exchange markets, Finance and stochastics 1 (2) (1997) 95–129.

[20] M. Sewell, Characterization of financial time series, Research Note, RN/11/01, UCL Department of Computer Science (2011).

[21] J. Glattfelder, A. Dupuis, R. Olsen, An extensive set of scaling laws and the FX coastline, Centre for Computational Finance and Economics Agents, Working Paper Series WP025-08.

[22] M. Aloud, E. Tsang, R. B. Olsen, A. Dupuis, A directional-change events approach for studying financial time series, Economics Discussion Paper 2011-28.

[23] M. Aloud, M. Fasli, The impact of strategies on the stylized facts in the FX market, Technical Report, University of Essex, United Kingdom (2013).

[24] M. E. Aloud, Time series analysis indicators under directional changes: The case of Saudi stock market, International Journal of Economics and Financial Issues 6 (1).

[25] E. Tsang, J. Chen, Regime change detection using directional change indicators in the foreign exchange market to chart brexit, IEEE Transactions on Emerging Topics in Computational Intelligence 2 (3) (2018) 185–193.

[26] J. Chen, E. Tsang, Classification of normal and abnormal regimes in financial markets, Algorithms 11 (12) (2018) 202.

[27] H. Ao, A directional changes based study on stock market, Ph.D. thesis, University of Essex (2018).

[28] T. Bisig, A. Dupuis, V. Impagliazzo, R. Olsen, The scale of market quakes, Quantitative Finance 12 (4) (2012) 501–508.

[29] V. Petrov, A. Golubb, Volatility seasonality of bitcoin prices.

[30] V. Petrov, A. Golub, R. Olsen, Instantaneous volatility seasonality of high-frequency markets in directional-change intrinsic time, Journal of Risk and Financial Management 12 (2) (2019) 54.

[31] M. Aloud, E. Tsang, R. Olsen, Modeling the FX market traders behavior: An agent-based approach, Banking, Finance, and Accounting: Concepts, Methodologies, Tools, and Applications (2014) 350.

[32] M. Aloud, Directional-change event trading strategy: Profit-maximizing learning strategy, The Seventh International Conference on Advanced Cognitive Technology and Applications.

[33] M. E. Aloud, Profitability of directional change based trading strategies: The case of Saudi stock market, International Journal of Economics and Financial Issues 6 (1).

[34] A. Bakhach, E. Tsang, W. L. Ng, V. R. Chinthalapati, Backlash agent: A trading strategy based on directional change, in: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2016, pp. 1–9.

[35] A. Bakhach, V. Chinthalapati, E. Tsang, A. El Sayed, Intelligent dynamic backlash agent: A trading strategy based on the directional change framework, Algorithms 11 (11) (2018) 171.

[36] A. Kablan, W. L. Ng, Intraday high-frequency FX trading with adaptive neuro-fuzzy inference systems, International Journal of Financial Markets and Derivatives 2 (1-2) (2011) 68–87.

[37] A. Bakhach, E. Tsang, W. L. Ng, Forecasting directional changes in financial markets, Working Paper WP075-15 Centre for Computational Finance and Economic Agents (CCFEA), University of Essex, Tech. Rep.

[38] A. Bakhach, E. P. K. Tsang, H. Jalalian, Forecasting directional changes in FX markets, in: IEEE Symposium on Computational Intelligence for Financial Engineering & Economics (IEEE CIFEr'16), IEEE, Athens Greece, 2016, pp. 6–9.

[39] J. Gypteau, F. E. Otero, M. Kampouridis, Generating directional change based trading strategies with genetic programming, in: European Conference on the Applications of Evolutionary Computation, Springer, 2015, pp. 267–278.

[40] N. Alkhamees, M. Fasli, A directional change based trading strategy with dynamic thresholds, in: 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2017, pp. 283–292.

[41] N. Alkhamees, M. Fasli, Event detection from time-series streams using directional change and dynamic thresholds, in: 2017 IEEE International Conference on Big Data (Big Data), IEEE, 2017, pp. 1882–1891.

[42] A. Ye, V. R. Chinthalapati, A. Serguieva, E. Tsang, Developing sustainable trading strategies using directional changes with high frequency data, in: 2017 IEEE International Conference on Big Data (Big Data), IEEE, 2017, pp. 4265–4271.

[43] R. J. Almeida, N. Baştürk, R. Golan, Intraday value-at-risk estimation for directional change events and investment strategies, in: 2017 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2017, pp. 1–8.

[44] M. Kampouridis, F. E. Otero, Evolving trading strategies using directional changes, Expert Systems with Applications 73 (2017) 145–160.

[45] M. Kampouridis, A. Adegboye, C. Johnson, Evolving directional changes trading strategies with a new event-based indicator, in: Asia-Pacific Conference on Simulated Evolution and Learning, Springer, 2017, pp. 727–738.

[46] C. Thornton, F. Hutter, H. H. Hoos, K. Leyton-Brown, Auto-weka: Combined selection and hyperparameter optimization of classification algorithms, in: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2013, pp.

847–855.

[47] M. López-Ibánez, J. Dubois-Lacoste, T. Stützle, M. Birattari, The irace package, iterated race for automatic algorithm configuration, Tech. rep., Citeseer (2011).

[48] M. Birattari, Z. Yuan, P. Balaprakash, T. Stützle, F-race and iterated f-race: An overview, in: Experimental methods for the analysis of optimization algorithms, Springer, 2010, pp. 311–336.

[49] J. Brookhouse, F. E. Otero, M. Kampouridis, Working with opencl to speed up a genetic programming financial forecasting algorithm: initial results, in: Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation, 2014, pp. 1117–1124.