Hybrid System Combining Artificial Neural Networks and Support Vector Machines for Trading the Forex Market Intraday

Vasco Amaral, *Instituto Superior Técnico*
Lisboa, Portugal

***Abstract*—This thesis proposes an intradaily Forex trading system combining a Support Vector Machine capable of identifying and classifying different types of markets, namely bullish, bearish and sideways and several Artificial Neural Networks, one for each market type, capable of finding intradaily price patterns and predict price movements from *12:00pm* till *4:30pm*. An incremental window moving average is applied on a price transformation of logarithmic return rates, creating the transformed data that is fed to the ANN as the features and target value, each sample representing a trading day. The SVM uses price sequence windows, of approximately three months, as features. Several strategies are proposed based on the forecasting done by the ANNs, in order to optimize the final performance of the trading system. The strategies proposed focus on the trading rules to enter the market, and in which direction, long or short, and also the level of confidence, which is represented in the trading size applied for each trade. The training was done with data from 2004 until 2018 and tested for the year of 2019. The final optimized system achieved a return on investment of 87.5% over the testing year and a maximum drawdown of 13%, largely overperforming the comparison methods of *Buy & Hold* and *Sell & Hold*.**

***Index Terms*—Artificial Neural Networks, Data Processing, Forex Market, Intraday Trading, Support Vector Machine**

I. Introduction

The foreign exchange market is a global market to trade currencies, shared by governments, banks, hedge funds and all kind of traders. It is, by far, the most traded market, as it's volume goes up to 6 trillion dollars per day [13], and respects everyone that has some kind of money in the world. One does not need to be in the financial sector to make use of this market. Either for working purposes, receiving the salary in a different country, for international holidays purpose, to trade cash for the countries currency or even to invest in stocks from a different country, one will need to trade its current money for a different currency, and therefore access the Forex market. The only rule in any financial market is that if the search is bigger than the offer the price will rise and vice-versa, so every time someone accesses a market to trade in it, is actually influencing the course of it.

The other subject on this thesis is Artificial Intelligence, particularly Artificial Neural Networks, which are algorithms that resemble a computational network inspired in biological neural systems, as Walczak and Cerpa [19] said, these models simulate the electrical activity of the brain and nervous system. This are very powerful tools, capable of recognizing and learning patterns from big amounts of data, that a human being is simply not capable of, as we have limited memory and processing power, when compared to a super computer.

So, in my opinion, it is rather amazing to be making use of computers that simulate the human brain, in order to make predictions on a market shared and influenced by every single person and entity on this planet.

A. Objectives

The main objective of this work is to implement a complex trading system for the Forex pair *EUR/USD*, capable of forecasting price movements on a specific time horizon and intradaily trade the market in a profitable way.

The system will be based on Artificial Intelligence. A Support Vector Machine will be trained to identify different market conditions and an Artificial Neural Network will be trained, for each set of market conditions, in order to identify price movement patterns and forecast these.

Data regarding the Forex pair *EUR/USD* from 2004 until 2018 will be used for this purpose and the final system will be implemented as if it was built and validated before 2019, and tested during this year on a real time simulation, so the results presented would actually be the ones obtained if one trusted the system to invest its money in the beginning of 2019.

B. Contributions

Following are the main contributions presented by this work:

- The identification of the best time windows to trade the Forex pair *EUR/USD*, intradaily.
- A Support Vector Machine capable of identifying and classifying different types of markets, regarding the behaviour of these.
- The implementation of three Artificial Neural Networks capable of finding intradaily patterns and predict the price movement based on these patterns, until the time of the prediction, and the current type of market.

II. Related Work

The literature review on AI solutions applied to Forex, revealed that a technical approach is usually the preferred choice, either analyzing price sequences or technical indicators. Fundamentals, which are very useful for dealing with other financial markets, such as stocks, are usually left aside when dealing with currency rates.

On a recent study from 2019, Chihab et al. [4] proposed a system to trade the Forex market intraweekly, based on a combination of some of the most common technical indicators. The system was implemented combining a Probit Regression model and a Random Forest algorithm, and achieved a ROI of 78% over 17 weeks, on the *GBP/USD* currency pair. A year before, Carapuço et al. [2] proposed a model of reinforcement learning, based on a neural network that uses the *Q-learning* algorithm to train three layers of ReLU neurons, this network trains itself over the testing period with past test data, making it possible to always be in the market and up to date. The proposed system achieved a ROI of 114% from 2010 till 2017 in the *EUR/USD* market, making an average yearly ROI of 16.3%.

Ealier in 2007 Yao et al. [21] had suggested a different approach than the traditional forex trading. Supported on the assumption that "the key in Forex trading is to pick the right currency to trade at the right time", this study proposes a system capable of managing a Forex portfolio of different currencies, instead of the usual trading systems based on single currency movements. The objective is achieved by implementing a fuzzy neural network (FNN) capable of forecasting price movements over a range of currencies. Yao et al. [21] claims that the "experimental results on real world Forex market data show that the proposed mechanism yields significantly higher profits against various popular benchmarks".

The research done over this topic indicates that a lot of different ML algorithms can be used to make accurate predictions on Forex time series, however, Petropoulos et al. [14] states that "in real-world trading settings, no single machine learning model can consistently outperform the alternatives" and proposes a voting system, in 2017, that combines the predictions of several algorithms, namely SVMs, random forests, ANNs and naive Bayes classifiers.

It is suggested on different studies that hybrid ML models combining different algorithms, usually outperform single algorithm systems. The most common case is the use of an algorithm to optimize the other, which is actually doing the forecasting. For example, Yu and Wang [22] proposed an online learning algorithm in 2007, capable of optimizing the learning stage of a neural network, outperforming the classical approaches of batch learning and the Levenberg-Marquard optimizer.

Later in 2012 Maknickiene and Maknickas [12] suggested an alternate version of a recurrent neural network (RNN) based on a genetic algorithm (GA), which uses a Long short-term memory network (LSTM), called *EVOLINO*. In order to use this model to trade the *EUR/USD* FX pair, Maknickiene and Maknickas [12] creates a system that uses several of this networks and a Delfi method, which works as a voting system depending on the individual outputs of each network, to ultimately generate the trading signal.

Özorhan et al. [24] implemented an hybrid system in 2017, based on a SVM to predict the "direction and magnitude of movement of currency pairs in the foreign exchange market". The SVM used several technical indicators as features adapted for each trading day by a genetic algorithm. The results obtained "suggest that using trend deterministic technical indicator signals mixed with raw data improves overall performance and dynamically adapting the input data to each trading period results in increased profits" [24].

A. Research on Artificial Neural Networks applied to Financial Markets

Artificial Neural Networks showed to be a widely used solution for forecasting time series in financial markets, became very popular in the 90s when more studies on the area started to arise, usually outperforming traditional trading strategies.

"Technical analysis is not designed to deal with non-uniform periodic, and discontinuous functions", such as forex prices time series, was claimed by Chan and Foo Kean Teong [3], and in 1995 they proposed a model composed of a single neural network capable of optimizing new technical indicators in order to open trades before the common, and widely used, technical indicators generate trading signals, followed by most of the crowd. Later in 1997, Yao and Tan [20] reported "empirical evidence that a neural network model is applicable to the prediction of foreign exchange rates". Their work consisted in the use of technical indicators and price sequences as features for the model proposed, composed by five neural networks, each forecasting a different major currency in relation to the USD. The study showed that "significant paper profits can be achieved for out-of-sample data with simple technical indicators", without an extensive training dataset.

As the evolution in the field of computational intelligence progressed, more complex solutions for the financial area, regarding newly optimized Artificial Neural Networks, started to be study.

In 2009 Butler and Daniyal [1] attempt to predict the movement of the stock market using an evolutionary artificial neural network (EANN). The results showed that the optimal solution was achieved by updating the EANN's weights through genetic operations, such as crossover and mutation, and that the traditional backpropagation method tended to overfit the data. A second conclusion taken from this study is that a training approach with multi-objective optimization (MOO) produces better results, represented in a bigger return on investment (ROI), than a traditional single objective optimization. A MOO consists of optimizing the neural network at each iteration, considering more metrics than just the prediction error in the cost function, for example the final ROI or the accuracy.

Evans et al. [6], in 2013, proposed an intradaily trading system for 3 related FX pairs, *EUR/USD*, *GBP/USD* and *EUR/GBP*, which trades the market that provides the bigger level of confidence, each day, according to the other two, for example if the 3 forecasted movements are in accordance. The forecasting is implemented by an Artificial Neural Network, which has its parameters and topology tuned by a Genetic Algorithm, in order to achieve optimal performance. The optimal proposed model achieved a prediction accuracy of 72.5% and an annualized net return of 23.3%. Furthermore, an important statistical test, of this study, "confirmed with a significance of more than 95% that the daily FX currency rates time series are not randomly distributed". Later in 2016 Galeshchuk [7] used a model based on ANNs to predict exchange rates in 3 different markets, *EUR/USD*, *GBP/USD* and *USD/JPY*, comparing the results and predictability of each. The results obtained suggested that *EUR/USD* was the most predictable market in daily steps, with an average relative prediction error of 0.2%, while *USD/JPY*, with a value of 0.3% for the same error in monthly steps, was the market to perform better under such circumstances. Lastly, *GBP/USD*, with an average error of 1.9% was the best performance for quarterly steps, with the *USD/JPY* performing at 3.5% relative error, showing a huge rise in unpredictability for bigger time ranges.

Neural networks are still widely used nowadays, because of their great adaptability to the features provided and also a low computational cost when compared to new ML algorithms that were invented over the years. On a very recent study from 2020, Zafeiriou and Kalles [23] implement an intraday forecasting system based on an ANN which is fed data at the *tick* of the currency pair *EUR/USD*. This system "aims to simulate the judgment and decision making of the human expert, responding in a timely manner to changes in market conditions, thus facilitating the optimization of ultra-short-term transactions" [23]. The neural network generates a "trend forecasting signal" and positions are opened in the market when a good opportunity arises, for example the price is shorter the the predicted trend. The final system was tested for more than 2 million data points, corresponding to October of 2018 and reached an accuracy of 78%.

B. Research on Support Vector Machines applied to Financial Markets

Support Vector Machines are also widely used in the financial time series prediction area, some prefer it over ANNs for they reduced computational cost and easier understandability of what is happening during the training stage. As Kyoung-jae Kim [11] claimed, "SVMs are promising methods for the prediction of financial time-series because they use a risk function consisting of the empirical error and a regularized term which is derived from the structural risk minimization principle". In 2003, he applied a simple support vector machine to predict the stock market prices and compared the performance with a backpropagation neural network and a case-based reasoning, achieving an accuracy of 57.8% against 54.7% and 51.9% by the NN and CBR, respectively.

"The SVM has been applied to the problem of bankruptcy prediction, and proved to be superior to competing methods such as the neural network, the linear multiple discriminant approaches and logistic regression" [8]. In 2007 Hua et al. [8] proposed a system combining support vector machines with a logistic regression, in order to improve accuracy, capable of predicting financial distresses in companies, based on fundamental data, showing promising results since it "outperformed the conventional SVM".

More recent studies keep proving the reliability of the SVM, both in forecasting, classification and optimization of more complex ML systems.

"The trend of currency rates can be predicted with supporting from supervised machine learning in the transaction systems such as support vector machine" [17]. In 2018 Thu and Xuan [17] achieved a return on investment, trading the Forex pair *EUR/USD* over 2016, of 33.8% using a single SVM to predict the direction of the market. A simple strategy was implemented to trade accordingly with the direction forecasted. In the same year, Jubert de Almeida et al. [9] implemented an hybrid system between a Support Vector Machine and a Genetic Algorithm. The second implements a classic approach of optimizing trading rules based on different technical indicators, the first (SVM), however, presents a unique approach of pre-classifying the type of market at the current moment and consequently using one of three distinctly trained GAs, according to said type of market. The best proposed system achieved a ROI, over a period slightly bigger than a year, of 83%.

III. Implementation

This work's architecture is composed of four main layers. Each one of those has its own specific goal, and functionality in order to accomplish it. These layers are connected to each other as shown in figure 1 and together form a complex system for trading, intradaily, the *Forex* pair *EUR/USD*, in a profitable way.

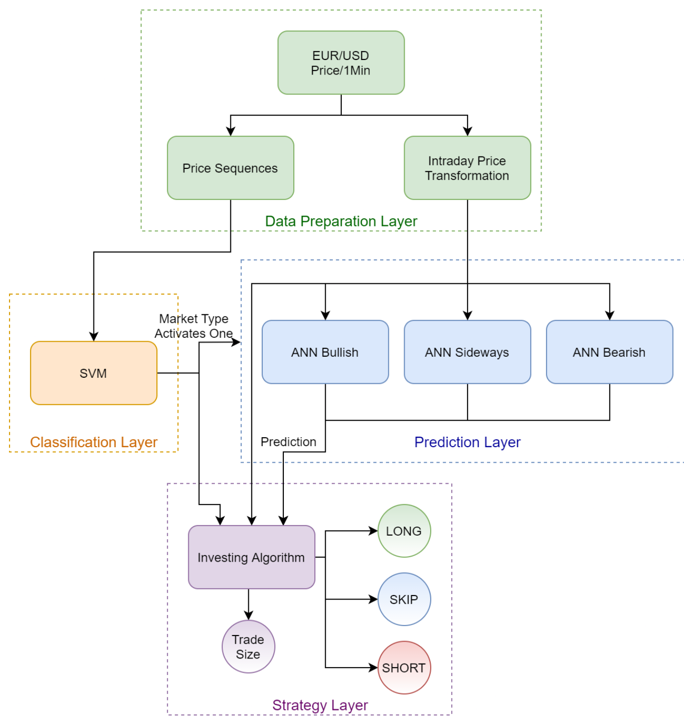

Fig. 1. System overall architecture

This system works on a daily basis, it analyzes price data from midnight till noon, and makes a prediction on a 4 and a half hour horizon - i.e., tries to predict, with a certain level of confidence, if the price of this pair will rise or fall until 4:30pm. Based on that prediction, and the level of confidence, the system will eventually take one of 3 decisions, either go long, short or skip that trade. If the decision is one of the first two and decides to open a trade, it will also compute the volume of that trade, based on the amount of money available to invest and the said level of confidence. As the prediction has a specific time horizon, the trade will be closed at 4:30pm no matter the price, it is also important to note that the volume traded is only possible with *margin trading*, which consists in the use of leverage. Another perk of intraday trades is that no *Swap* needs to be payed, since it is charged overnight on open positions.

Following is a description of each layer's function and how these interact with each other in order to ultimately take said decision.

1) The **Data Preparation Layer** is responsible for preparing the data to feed as an input to every other layer. From the raw prices, ticked at the minute, of the *FX* pair *EUR/USD*, on one side arranges a three month window of price sequences to feed the classification layer while on the other prepares daily samples of a scaled transformation to feed the prediction and strategy layer, respectively, as seen in figure 1. All the raw data comes from *.csv* files, covering prices at the minute from 2004 to 2019, downloaded from *dukascopy.com*.
2) The **Classification Layer** is based on Jubert de Almeida et al. [9] work and consists of a single *Support Vector Machine* to accomplish its role in the system. The goal is to add a layer, before the actual prediction, that has the ability to classify the market in three different groups, either a *bullish* market, which has an upgoing trend, a *sideways* market or a *bearish* market, which has a downgoing trend. For this, it will have access to a completely different timeframe than the next layer, while the prediction layer is intradaily based - i.e., each sample only has information about one specific day, this layer has access to time windows of three months. Any *ML* algorithm is subject to *overfitting*, or *underfitting* logically, but *Artificial Neural Networks*, specifically, are very prone to overfit a solution. The idea behind this classification layer, in order to mitigate this issue, is based on the popular saying "if you cannot beat them, join them". If it is hard to avoid *overfitting* without generalizing the problem too much, one can use it to its own advantage. With the data clustered in three different groups, based on the type of market at the time, one can train three different neural networks that will only see data belonging to a specific type of market, each, and therefore be *overfitted* to that kind of data, which is not a problem as long as the classification layer accomplishes its goal, accurately, during real time predictions. To summarize, data on the last three months will be given to the *SVM*, which will classify it into one of three types and activate one, and one only, of the neural networks on the next layer, accordingly.
3) The **Prediction Layer** is the core of this work, its goal is to predict price movements on a four and a half hours horizon, fundamented in neural networks. It can be looked as if this is the main layer and the others work for this one, each doing a different optimization, but if the prediction is inaccurate, the entire system will fail. This layer will receive a sample per day, which consists of a transformation of price sequences from midnight until noon, and output its prediction, on that transformation, at *4:30pm*. As it is a supervised learning system, it will also receive the accurate value of this transformation at *4:30pm*, during the training stage - i.e., the sample's *label*. It is composed of three, distinctly trained, *Artificial Neural Networks* activated one a time depending on the type of market being traded at the moment, previously classified by the last layer. Its output will be fed to the strategy layer, alongside the initial data transformation and other meaningful variables, so that it can decide what to do at *12:00pm* based on what it predicts will happen until *4:30pm*.
4) The **Strategy Layer**, similarly to the data preparation one, has no machine learning behind. It is a simple algorithm that joins the outputs from all other layers to ultimately take a decision between skipping a trade or take a long or short position, and, for the last two, also the size, or volume, of the said trade. Different strategies will be tried and placed against each other, in order to evaluate how these behave in different conditions and which one brings more profit.

A. *Features for Prediction*

For the prediction, a system of intradaily samples was chosen, with one sample corresponding to the trading prices of a single day from midnight until noon, with the target value being that same price at *4:30pm*. According to Evans et al. [6], Forex intraday price rates are noisy, chaotic and present non-stationary behaviour, so to make a prediction on those there is the need to apply some kind of transformation to deal with these issues.

The first issue that needs to be dealt with, is the sampling frequency used to represent an entire day. On one side, high frequency samples means too much information, "useless and sometimes disorienting" [6], while low frequency might mean that crucial information to identify patterns can be missing. In his book, Refenes [15] stated that a sampling period between 5 and 60 minutes is ideal for the forex market depending on the currency pair. For this work a sampling period of 30 minutes was chosen, which corresponds to a vector with a size of 25 features, to be fed as an input to the artificial neural networks. However, this layer has to treat the data until *4:30pm*, as this is the time corresponding to the target value, used for training. Lastly, in order to mitigate the noise present in price variations during the day, an average of 30 minute windows was taken for each point. An example of a few different samples, after this smoothness transformation is applied, representing price variations during the day, is represented in figure 2. This prices correspond to the year 2019 of the forex pair *EUR/USD* and each line represents a different day, as shown in the figure's label.

The next step is to have a meaningful data representation for the *ANNs*, Vanstone and Finnie [18] claimed that it is crucial that *ANNs* do not have visibility of the raw market prices, otherwise identical patterns happening at different price zones will be treated as distinct ones, making the generalization close

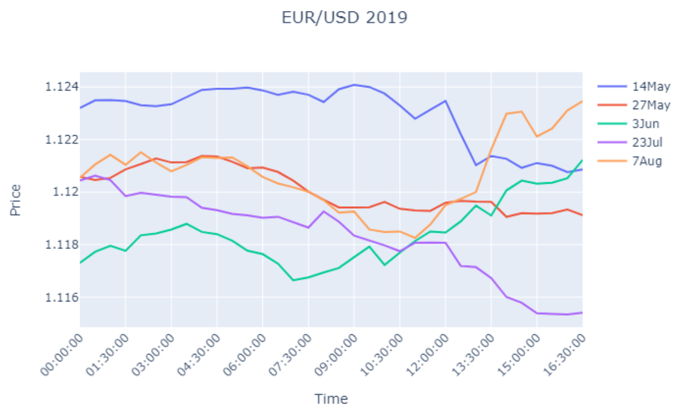

Fig. 2. Price during the day of different *EUR/USD* samples

to impossible. So this data needs to be scaled to a specific range where identical patterns will be treated as so. Evans et al. [6] suggests an approach based on return rates between the current point and the last one, accomplished by a log difference transformation as shown in equation 1, where P_t represents the price at time t and P_{t-1} the one before, at $t-1$.

$$r = ln(\frac{P_t}{P_{t-1}}) \tag{1}$$

In figure 3 this transformation is shown on the same examples used before in figure 2.

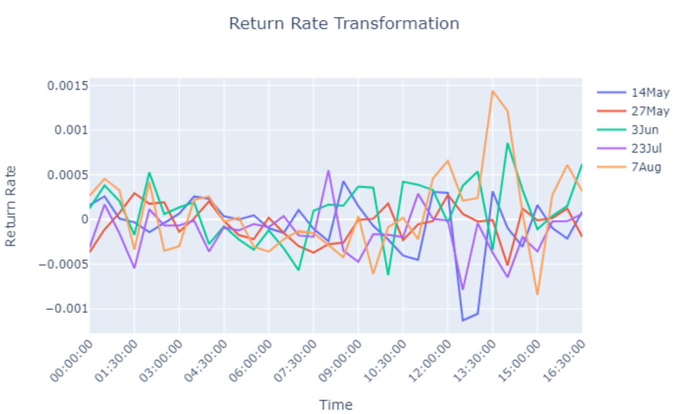

Fig. 3. Return rate transformation of different samples

Now that the generalization issue has been resolved, a new one rises, which is convergence, as this transformation makes each point depend solely on itself and the last one. In this conditions it is hard for an *ANN* to make a prediction, as the data would be considered random. A possible way to overcome said issue, is with the use of a technical indicator that takes all this return rates in consideration, which actually represent price movements. For example, if one can average every return rate, this would give information about how much the price moved, in average, during the day. As each sample starts at midnight and has no information behind that point, a moving average with a window sized n is not possible. The workaround is the use of the Incremental Window Moving Average. This indicator takes in consideration the entire window that is behind a certain point and for each new point that window will grow, as can be seen in equation 2, which is ideal for the intradaily samples used in this work.

$$y_{(i,j)} = \frac{\sum_{l=1}^{k} x_{(l,j)}}{k} \forall k = 2, 3, 4, ...n \tag{2}$$

The outcome of the IWMA transformation is the actual data that will be fed to the prediction layer, containing the input features needed for the *ANNs* and also the target value to be predicted. In figure 4 is shown this transformation on the same examples presented before in both figures 2 and 3.

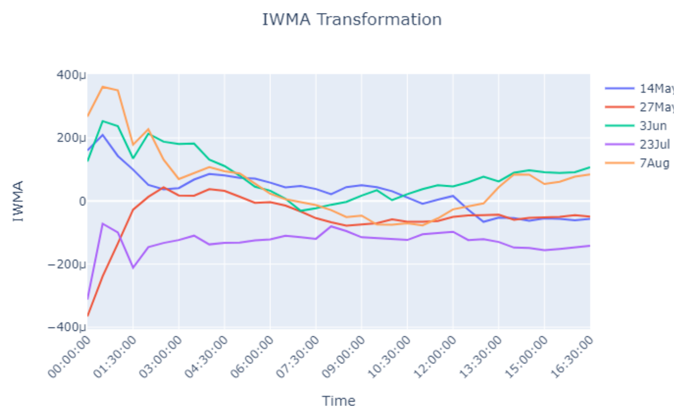

Fig. 4. Incremental window moving average transformation of different samples

B. Classification Layer

This layer's function is very simple but crucial on the entire system functionality, its job is to classify the current market as belonging to one of 3 types, bearish, bullish or sideways, relying on a single *SVM*. This classification will be used on the next layer, the prediction one, as explained before, to decide which neural network to use. It will also be useful for the strategy layer (strategy D) to adjust decision rules of when to enter the market.

The input features chosen are 3 month price sequence windows and the labels were obtained by classifying the market manually. This sections explains how this layer is implemented, how the best hyperparameters were searched for and the results obtained.

C. Hyperparameters

The first hyperparameter to tune is the type of kernel used, with 3 options available. The simplest one is the linear kernel, it is very fast to train, however is limited for problems where data is linearly separable. Both the radial basis function and the polynomial kernel can approach non linear problems, with the polynomial one being more versatile but taking much longer times during training phase, specially for bigger dimensions - i.e., higher n values.

The *C* parameter is the one regarding the cost function, the higher it is, the bigger the influence of each individual support vector, getting bigger penalizations for wrongly classified samples, which in the limit is extremely dangerous regarding overfitting, specially for less representative training datasets.

Lastly, the *Gamma* parameter, non-existent for the linear kernel, controls the curvature of the decision boundary, with higher values meaning a higher curvature. Again, it depends on the data to find the perfect values for this parameter, however, bigger values are usually prone to overfit as decision boundaries will shrink around samples.

D. Grid search Validation

In order to evaluate the best parameters, according to the metrics described, a grid search algorithm was applied. The grid search algorithm will train and test the *SVM* multiple times with every combination of *C* and *Gamma* values for the 3 kernels, according to table I. The best solutions will be evaluated and ultimately one set of hyperparameters will be chosen.

TABLE I
HYPERPARAMETERS TO BE APPLIED ON GRID SEARCH FOR *SVM*

Kernel		C	Gamma
Linear		[1; 10; 100; 1000]	Non-Applicable
Polynomial,	n=2,3,4	[1; 10; 100; 1000]	$[10^{-1}; 10^{-2}; 10^{-3}]$
RBF		[1; 10; 100; 1000]	$[10^{-1}; 10^{-2}; 10^{-3}; 10^{-4}]$

Below in table II the best parameters found for each kernel, as well as the corresponding performances and cross validations, can be seen.

This table provides the values for precision and recall per classification group, from left to right, being the first the bearish markets, then sideways and then the bullish ones. Looking at table II it is obvious that the best overall solution is the polynomial kernel, even though its computational time is considerably higher, it is the most balanced solution. The linear solution appears to be overfitted just by looking at the parameter *C* which takes the biggest value of all, 1000. Also has a 60% value both for precision and recall, even though it is in different classification groups. The last "red-flag" is a 49.5% value for one of the 5-Fold cross validations, which is very distinct from all others, once again showing a biased solution. The radial basis function kernel solution also presents signs of overfitting, again one of the cross validation values is less than 50% which is very bad. It is also obvious from looking at precision and recall values, as the precision for the bearish and bullish markets is slightly better, and then the recall for the sideways markets is way bigger, it means that the system classifies most of the markets as sideways, therefore having the perfect recall but not so good precision and vice-versa for the other classes.

The solution chosen to apply on this layer is the polynomial one presented with a degree of 3, a C value of 100 and a Gamma of 0.01, as all values of precision and recall look very stable. The accuracy is also very good at almost 86% and the lowest cross validation value is at 78.1% with a maximum deviation of 9.7%.

E. Prediction Layer

This layer is composed by three distinctly trained neural networks and its job is to predict intradaily price movements on a four and a half hours horizon. The final system solution uses the previous layer to classify the market type and then only the correspondent ANN is activated. Each ANN is trained with the corresponding data - i.e., the bullish one will only be trained with samples classified as belonging to a bullish market.

F. Activation Function

As the values of the IWMA, which represent both features and target values for the ANN, are in the range [-1;1], as explained in section III-A, the activation function chosen is the hyperbolic tangent, shown in equation 3, one the of the most common activation functions for MLP networks and which range is the same as the IWMA, [-1;1].

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3}$$

G. Loss Function and Performance Metrics

For this work, the loss function chosen was the Total Sum of Squares (TSS), which is defined in equation 4, even though the RMSE was kept as a performance metric.

$$TSS = \sum_{i=1}^{n} (\hat{Y}_i - Y_i)^2 \tag{4}$$

Another performance metric used in the next evaluations is the accuracy, which at this point is considered to be the relation between prediction and target value in reference to the point at *12:00pm*, instead of the actual trade accuracy, as a trade strategy is not in place yet, and is represented in table III has *Accuracy**. A sample is considered rightly predicted, for this specific accuracy computation, when it is in the same direction as the target value in relation to the initial value, at midday - i.e., if both (target and prediction) are higher or both lower than the initial value, even if distant from each other, than it is an accurate prediction. This will be proven not to be always true when comparing the transformation to the actual price movement, so the total loss of the network is still very important to keep the predictions and the target values as close as possible to each other.

H. Network's Topology

One of the most important things in a neural network is its topology - i.e., the number of layers and nodes per layer. The input and output layer is given by the problem itself, in this case, the input layer has 25 neurons, the number of features,

TABLE II
BEST PARAMETERS AND RESULTS FOR THE SVM

Parameters		**Precision** [%]	**Recall** [%]	**Accuracy**	**Cross-Validation** [%]
		Average			
Linear	C = 1000	[81 ; 86 ; 60] 76%	[60 ; 89 ; 80] 76%	82.35%	[80.9, 49.5, 75.5, 63.1, 78.8]
Polynomial (n=3)	**C = 100 Gamma = 0.01**	**[83 ; 89 ; 68] 80%**	**[73 ; 91 ; 78] 81%**	**85.93%**	**[87.8, 78.1, 84.3, 83.6, 86.3]**
RBF	C = 100 Gamma = 0.01	[99 ; 79 ; 83] 87%	[51 ; 100 ; 4] 51%	80.97%	[60.3, 49.1, 77.1, 70.4, 76.2]

and the output layer has 1, the single prediction the network is supposed to output. So the margin to optimize an ANN to a specific problem is in the hidden layers and nodes. There is not a rule on how to build an ANN, any topology is valid as long as it suits the problem. There are, however, a few guidelines that fit the majority of problems. For example, the number of neurons on a given layer should be in between the number of nodes from the last and the following layer [10], having more neurons than inputs would mean redundant neurons learning the same functions as each other.

In order to find the optimal solution, for the topology of this work's ANNs, a series of tests were made on different topology configurations, with more or less layers and neurons. After having a validation of which kind of configuration has better performance, a test on a narrower range of similar topologies is applied. On table III the results of the tests applied on different kinds of topologies can be analyzed.

TABLE III
EVALUATION METRICS FOR DIFFERENT ANN TOPOLOGIES (AVERAGE OF 3 TESTS PER CONFIGURATION)

Topology	Total Loss (TSS)	RMSE	Accuracy*
[12]	4.024×10^{-9}	4.742×10^{-5}	62.1%
[20]	3.926×10^{-9}	4.695×10^{-5}	**62.7%**
[12 - 8]	4.17×10^{-9}	4.904×10^{-5}	56.2%
[20 - 16]	4.278×10^{-9}	4.969×10^{-5}	57.6%
[12 - 10 - 8 - 6]	4.223×10^{-9}	4.86×10^{-5}	58.6%

The tests provided the expected results, given the unpredictability and noise of the forex market it was expected that wider networks would behave better. As stated, fewer layers prevent overfitting, and even though the error metrics are not as different, the accuracy regarding the direction of the movement on the IWMA proves that a single layer with more neurons is the better approach. The next search is done on a narrower range of similar configurations, changing only the number of neurons. An additional grid search, according to table IV, for each of the topologies was done, in order to discover the best learning rate and batch size for the network to use. The results of this entire validation set for the best parameters are provided in table V.

TABLE IV
HYPERPARAMETERS TO BE APPLIED ON GRID SEARCH FOR *ANN*

Batch Size	[4; 8; 16; 32]
Learning Rate	$[10^{-3}; 10^{-4}; 10^{-5}; 10^{-6}]$

TABLE V
EVALUATION METRICS FOR OPTIMAL TOPOLOGIES (AVERAGE OF 3 TESTS PER CONFIGURATION)

Topology		Total Loss (TSS)	RMSE	Accuracy*
[18]	BS = 4 LR = 10^{-5}	3.937×10^{-9}	4.719×10^{-5}	60.25%
[20]	BS = 8 LR = 10^{-5}	3.926×10^{-9}	4.695×10^{-5}	62.7%
[22]	**BS = 8 LR = 10^{-5}**	$\mathbf{3.803\times10^{-9}}$	$\mathbf{4.597\times10^{-5}}$	**63.5%**
[24]	BS = 4 LR = 10^{-6}	4.362×10^{-9}	5.025×10^{-5}	57.44%

The network which performed better was the one with 22 perceptrons in one hidden layer, with an accuracy of 63.5% (over IWMA direction as mentioned) and a RMSE of 4.596×10^{-5}. The best learning rate was 10^{-5} and a batch size of 8 was used.

IV. SYSTEM EVALUATION

This section presents all the tests to which the final trading system was subject too, as well as specific case studies in order to evaluate the best possible approach. The measures used, for the purpose of evaluating the possible solutions were Return on Investment (ROI), maximum Drawdown and the Accuracy.

For the training of the ANNs, the data used is referent to the years of 2010 till 2018, while the training of the SVM is done with data from 2004 till 2018. All the case studies are tested and simulated on the year of 2019, which is never presented to the AI algorithms during the training stage. For every test presented below, the result of an average of three distinct iterations is presented, given the randomness factor of ANNs one test cannot supply enough confidence to make assumptions on what works better. It is also important to note that the only commission taken into account by the trading system is the *spread* for each trade, as swap and other commission do not apply. The *spread* considered is 2 pips per trade, which is slightly higher than the average *spread* at the time the trade is entered (1.7 pips) [16].

A. 1st Case Study - Prediction Time Window

The first case study compares two different time ranges for the prediction horizon which the trading system uses. It is important to note that the system at this point is composed of a single ANN and the strategy in place to calculate the ROI is the first and simplest strategy. The first time range is of four and a half hours, from *12:00pm*, time at which the system makes a prediction and opens a trade, until *4:30pm*, time of the prediction itself and at which the trade is closed. The second time range is of six hours, from *1:30pm* until *7:30pm*.

The average price variations and the performance of the ANN in both situations can be seen in table VI, alongside the results of the simulation for both time ranges.

TABLE VI
EVALUATION METRICS FOR DIFFERENT MARKET HOURS STRATEGY (*EUR/USD* 2019)

Measures		Entry at *12:00pm* Close at *4:30pm*	Entry at *1:30pm* Close at *7:30pm*
ROI		27.5%	-23.7%
Max Drawdown		20.6%	27.2%
Accuracy		54.6%	51.5%
Total Trades		260	260
	Longs	87	80
	Shorts	173	180
Max ROI		43.0%	0%
Min ROI		-20.6%	-33.7%
Accuracy* (IWMA)		63.5%	60.8%
Average Price Variation		17.6pips	19.7pips

Even though, in average, there is a smaller price variation between *12:00pm* and *4:30pm* (17.6 pips) than between *1:30pm* and *7:30pm* (19.7 pips), which means less margin to make a profit, the system showed to perform considerably better for the first case scenario, as can be seen in figure 5, with a ROI of 27.5% when compared to a negative ROI of -23.7%. The trade accuracy is also slightly better, 54.6% compared to 51.5%, which is too close to 50%, explaining the negative ROI. However, the IWMA accuracies of the networks are very similar, 63.5% and 60.8%, which indicates a good performance of both, so the problem is not in the prediction of the IWMA transformation, but in the relation of this to the actual price movement.

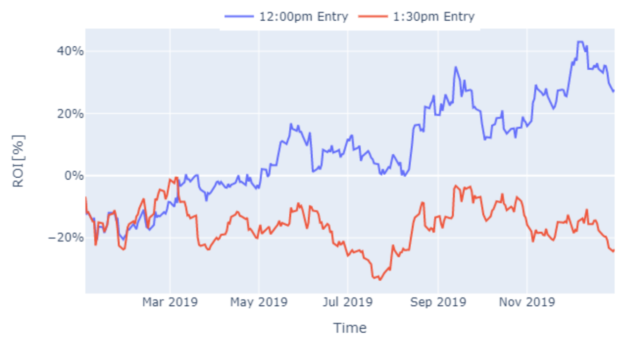

Fig. 5. ROI over time for different market hours strategy (*EUR/USD* 2019)

The fact that the relations, between IWMA and price movement are not as coherent for the second time range, as for the first, has several reasons. The first, and most logical, is the fact that the time window is bigger and, consequently, there is more time for unpredictable variations. Beyond, the pattern explained for the second scenario (strategy B), also happens more often. There is more time for the price to keep growing steady, for example, but with a lower return rate each timestep, and consequently the IWMA will be lower in the end, even though the price is bigger.

In addition to the size of the prediction horizon, the actual hours at which the trades are being opened and closed is also relevant. The *EUR/USD* market has the biggest trading volumes, usually, between *12pm* and *4pm*, as shown in figure 6, consequence of having both the European and American trading sessions running during this time. With more active traders on the market, movements can be more predictive and impulses bigger, corrections and unexpected movements usually happen after hours, as there is less volume in the market, so the price is subject to a bigger volatility and the will of who is actually in the market. A highest trading volume also means a narrower spread [6], which is the only commission this trading system is subject too, so it is crucial that it assumes the minimum possible value. Lastly, the daily exchange rate considered and used by most official sources, such as banks and exchange offices, in Europe, is taken between *4:00pm* and *4:30pm* [5], which means that the ”big players”, as banks and hedge funds are usually referred to, will be in the market making sure the price is in the desired zone by this time, once again, making this the most predictive time window to trade at.

Two conclusions can be taken from this case study, the longer the time window for a market prediction, the bigger the profit margin one can make, however, the unpredictability

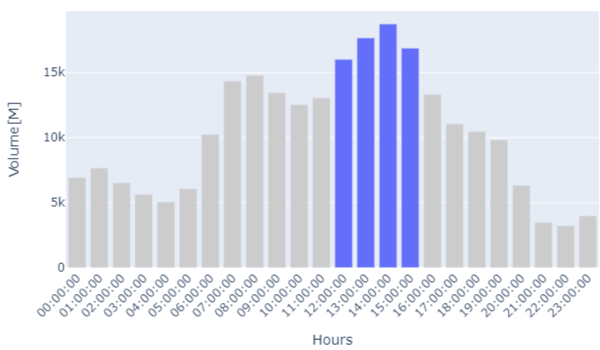

Fig. 6. Average Traded Volume per hour (*EUR/USD* 2017-2019)

of the market rises and so does the difficulty of solving the problem as suggested in this work. Secondly, the peak hours of the market - i.e., the time range with the biggest trading volume, represented mostly by banks and hedge funds from both Europe and the USA, presented to have the most predictable, and well defined, price movements.

B. 2nd Case Study - Trading Strategy

The second case study compares three different trading strategies regarding the decision logic behind the direction of a trade to take in accordance to the prediction of the IWMA, plus a fourth strategy regarding the size, or volume, of each trade. In addition to the strategies implemented by the system, the strategies *B&H* and *S&H*, for the same test period, are also implemented.

The results obtained for each strategy during the year 2019 are presented in table VII and a graphical representation on the evolution of the ROI for each strategy is shown in figure 7.

Fig. 7. ROI evolution for every strategy proposed (*EUR/USD* 2019)

Strategy A is the simplest of all strategies, never skips a trade, totaling 260 trades over the entire test period, making it the riskiest strategy of all, reflected in a maximum drawdown of 20.2% and a minimum ROI of -15%. The final ROI obtained, 31.4%, was enough to beat the *S&H* strategy which only obtained a ROI of 21.5%. However, the maximum drawdowns, 20.2% and 20.5%, are very close, which supports the claim that strategy A is still too dependent on the market, not providing a better risk than a fundamental analysis of staying short all year long. Looking at figure 7 is also possible to understand that strategy A movements on balance are very inconsistent, similar to the *S&H* ones.

Strategy B is the most conservative of all strategies, the one with less movements, corresponding to a total of 73 trades. Being the safest strategy in terms of price movement confidence, it is expected to present the smallest risk when taking a trade, which should be reflected on the drawdown, besides the fact of never having a negative ROI. Even though, strategy B presented a bigger maximum drawdown, 16.2%, than strategies C and D, by looking at figure 7, it is possible to understand that this strategy has the smoothest movements, and smallest falls. The bigger drawdown is a consequence of not availing the big movements up as the other strategies, never regaining the balance lost on a series of missed trades, so even though the drawdown is bigger, it happens over a larger period, making this the strategy with the least abrupt movements on balance. This strategy obtains a final ROI of 36.1%, beating strategy A's 31.4%, even with a slightly worse accuracy, 53.4% compared to 53.8%.

Strategy C is the most complete of the first three strategies, with a relatively high amount of trades entered, 176, and an accuracy of 57.4%. It is the strategy with the smallest maximum drawdown at only 11.9% and a final ROI of 73.4%, largely superior to strategies A and B. Observing figure 7 one can see the evolution of the ROI in strategy C, as expected, is less steady than strategy B, but does the same job avoiding the falls of strategy A without losing as many opportunities as B. As the trading size is directly related to the size of the current balance, the more the ROI grows, the bigger the impulses it can make, on accurate trades, and consequently bigger falls.

Strategy D, as expected, follows the evolution of strategy C, since the trading rules are practically the same. The objective was that this strategy would benefit greatlier from movements with a bigger level of confidence, and have smaller falls on uncertain trades, which was accomplished by an accuracy of 58.3% and a final ROI of 87.5%. This pattern can also be seen in figure 7, as the falls and impulses up happen at the same time, for both this strategies.

From this case study, can be concluded that strategy D outperforms all other strategies in every aspect except for the maximum drawdown, 13%, slightly higher than strategy C. However, this drawdowns happen at the same time and is a consequence of the event explained before, that the bigger the balance, the bigger the falls, since the trading size depends on the current balance.

V. Conclusion

The final system presented by this work, combines a single SVM with three distinct ANNs, in order to make intraday trades in the Forex market of the currency pair *EUR/USD*. The system was trained with data referent to the years of 2004 till 2018 and tested for the year of 2019, providing a great yearly return on investment of 87.5%.

TABLE VII
EVALUATION METRICS FOR STRATEGIES PROPOSED (*EUR/USD* 2019)

Measures	Strategy A	Strategy B	Strategy C	Strategy D	B&H	S&H
ROI	31.4%	36.1%	73.4%	87.5%	-21.5%	21.5%
Max Drawdown	20.2%	16.2%	11.9%	13.0%	56.3%	20.5%
Accuracy	53.8%	53.4%	57.4%	58.3%	0%	100%
Total Trades	260	73	176	156	1	1
Longs	95	3	18	17	1	0
Shorts	165	70	158	139	0	1
Max ROI	55.5%	53.4%	87.4%	108.1%	8.4%	52.8%
Min ROI	-15.0%	0%	0%	-2.6%	-52.8%	-8.4

The SVM receives price sequence windows, of approximately three months, to use as features, in order to classify the different market types, bullish, bearish or sideways. Depending on the classification, one of the ANNs, which was trained with the correspondent type of data, is activated, and performs an intraday forecasting, at *12:00pm*, for the price movement until *4:30pm*. The strategy layer, based on the prediction done by one of the ANNs, the type of market classified by the SVM and the current balance, takes one of 3 options, either to enter a long trade, a short or skip that day, it also decides the size of said trade for the first two options.

Support Vector Machines presented to be a reliable method for classification of financial time series, as long as the labels, for the classification purposes required, are provided and consistent, making it possible to define the patterns wanted to be identified and the SVM does the job of identifying it in new data.

Artificial Neural Networks, as expected from the literature review, presented very good forecasting results, as long as correctly optimized and trained with enough and relevant data. The use of different ANNs for different types of market, reducing the generalization of the ANN itself and therefore slightly overfitting the training data, showed to perform better than a single ANN in terms of forecasting the price movements, as long as the correct network is being used, so if the classification is missed, the probability of an unsuccessful forecast rises.

Lastly, the peak hours of the *EUR/USD* Forex market, from *12:00pm* till *4:30pm* - i.e., the time range with the biggest trading volume, represented mostly by banks and hedge funds from both Europe and the USA, presented to have the most predictable, and well defined, price movements.

Trading the currency pair *EUR/USD* intradaily, can bring huge profits when predictions are accurate and leverage is applied, however, the use of leverage comes with an increased risk. It was not the case during any of the tests applied on this system, but financial markets have proved over and over to be very unpredictable environments, so one needs to be conscious that the possibility of unplanned events during the time positions are open in the market, without any kind of control, can completely change the course of the price movement and have a huge negative impact in the overall ROI, from a single trade, especially when applying high leverages.

A. *Future Works*

Even though this work achieved considerably good results, there is always space for improvement. When analyzing the case studies presented, some issues arise, which can be addressed with a more robust and complete system. Possible improvements to this work should focus on a better risk management and mitigation, as well as improving the performance of both the ML algorithms. The following approaches should be considered:

- A secondary system to control market events and prices during the prediction horizon, when there are open positions, would prevent catastrophic losses from happening and therefore mitigate the risk the system is exposed to. Can be done with something as simple as the inclusion of stop losses, or a more robust system with some ML algorithm to perform anomalies detection in the patterns predicted.
- The ANNs would perform better if a genetic algorithm was used to control the hyperparameters and topologies of the networks, instead of the manual tuning of parameters which was done on this thesis. The SVM could also benefit from the inclusion of a GA to control not only the hyperparameters, but the size of the price sequences given as features.
- Instead of classifying three types of market, which had to be labeled manually in order to train the SVM. It would be interesting to explore some kind of unsupervised learning algorithm, capable of clustering different types of markets, according to the patterns found, which would for sure be different than what the human eye can see. This way, more than 3 types of markets could be explored and the system would be less reliable on the human side.
- Finally, AI should be introduced in the strategy layer in order to find optimal trading strategies, improving profits and reducing the losses. For example, a GA could be used, using the outputs of every other layer, and maybe even the inclusion of some technical indicators, to implement a more robust trading strategy capable of maximizing the ROI and mitigating the risk.

References

[1] Matthew Butler and Ali Daniyal. Multi-objective optimization with an evolutionary artificial neural network for financial forecasting. pages 1451–1458, 07 2009.

[2] João Carapuço, Rui Neves, and Nuno Horta. Reinforcement learning applied to forex trading. *Applied Soft Computing*, 73:783 – 794, 2018.

[3] K. C. C. Chan and Foo Kean Teong. Enhancing technical analysis in the forex market using neural networks. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 2, pages 1023–1027 vol.2, 1995.

[4] Younes Chihab, Zineb Bousbaa, Marouane Chihab, Omar Bencharef, and Soumia Ziti. Algo-trading strategy for intraweek foreign exchange speculation based on random forest and probit regression. *Applied Computational Intelligence and Soft Computing*, 2019, 2019.

[5] Banco de Portugal. Exchange rates. URL https://www.bportugal.pt/en/taxas-cambio.

[6] Cain Evans, Konstantinos Pappas, and Fatos Xhafa. Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation. *Mathematical and Computer Modelling*, 58(5):1249 – 1266, 2013.

[7] Svitlana Galeshchuk.

[8] Zhongsheng Hua, Yu Wang, Xiaoyan Xu, Bin Zhang, and Liang Liang. Predicting corporate financial distress based on integration of support vector machine and logistic regression. *Expert Systems with Applications*, 33(2):434 – 440, 2007.

[9] Bernardo Jubert de Almeida, Rui Ferreira Neves, and Nuno Horta. Combining support vector machine with genetic algorithms to optimize investments in forex markets with high leverage. *Applied Soft Computing*, 64:596 – 613, 2018.

[10] Iebeling Kaastra and Milton Boyd. Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, 10(3):215 – 236, 1996.

[11] Kyoung-jae Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1):307 – 319, 2003.

[12] Nijole Maknickiene and Algirdas Maknickas. Application of neural network for forecasting of exchange rates and forex trading. 05 2012.

[13] Lina Ni, Yujie Li, Xiao Wang, Jinquan Zhang, Jiguo Yu, and Chengming Qi. Forecasting of forex time series data based on deep learning. *Procedia Computer Science*, 147:647 – 652, 2019.

[14] Anastasios Petropoulos, Sotirios P. Chatzis, Vasilis Siakoulis, and Nikos Vlachogiannakis. A stacked generalization system for automated forex portfolio trading. *Expert Systems with Applications*, 90:290 – 302, 2017.

[15] Apostolos-Paul Refenes. *Neural Networks in the Capital Markets*. John Wiley Sons, Inc., 1994.

[16] Teletrade. Negotiation conditions. URL https://www.teletrade.eu/pt/trade/condition-mt4/forex.

[17] T. N. T. Thu and V. D. Xuan. Using support vector machine in forex predicting. In *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*, pages 1–5, 2018.

[18] Bruce Vanstone and Gavin Finnie. An empirical methodology for developing stockmarket trading systems using artificial neural networks. *Information Technology papers*, 36, 2009.

[19] Steven Walczak and Narciso Cerpa. Artificial neural networks. In Robert A. Meyers, editor, *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 631 – 645. Academic Press, New York, third edition edition, 2003.

[20] Jingtao Yao and Chew Lim Tan. A case study on using neural networks to perform technical forecasting of forex. *Neurocomputing*, 34(1):79 – 98, 2000.

[21] S. Yao, M. Pasquier, and C. Quek. A foreign exchange portfolio management mechanism based on fuzzy neural networks. In *2007 IEEE Congress on Evolutionary Computation*, pages 2576–2583, 2007.

[22] Lean Yu and Shouyang Wang. An online learning algorithm with adaptive forgetting factors for feedforward neural networks in financial time series forecasting. *Nonlinear Dynamics and Systems Theory*, 1, 03 2007.

[23] Theodoros Zafeiriou and Dimitris Kalles. Intraday ultra-short-term forecasting of foreign exchange rates using an ensemble of neural networks based on conventional technical indicators. In *11th Hellenic Conference on Artificial Intelligence*, page 224–231, New York, NY, USA, 2020. Association for Computing Machinery.

[24] Mustafa Onur Özorhan, İsmail Hakkı Toroslu, and Onur Tolga Şehitoğlu. A strength-biased prediction model for forecasting exchange rates using support vector machines and genetic algorithms. *Soft Computing*, 21: 6653 – 6671, 2017.