Rosillo, R.; Giner, J. & and De la Fuente, D. (2014): "Stock Market simulation using support vector machines", ***Journal of Forecasting***, vol. 33, nº 6, (488–500).

Stock Market Simulation using Support Vector Machines

Abstract:

The aim of this research is to analyse the different results that can be achieved using Support Vector Machines (SVM) to forecast the weekly change movement of the different simulated markets. The different simulated markets are developed by a GARCH model based on the S&P 500. These simulated markets are grouped by a main parameter: high volatility, bearish trend, bullish trend and low volatility. The inputs retained of the SVM are traditional technical trading rules used in quantitative analysis such as Relative Strength Index (RSI) and Moving Average Convergence Divergence (MACD) decision rules. The outputs of the SVM are the degree of set membership and the market movement (bullish or bearish). The design of the SVM algorithm has been developed by Matlab and SVM-KM. The configuration for the SVM shows that the best results are achieved in simulated markets with high volatility; also results are good in trend simulated markets.

Key words: Support Vector Machines, Stock Market Simulation, RSI, MACD.

[1]***Rafael Rosillo*** *is a Professor at the University of Oviedo in the Business Management department.*
[2]***Javier Giner*** *is a Professor at the University of La Laguna in the Finances and Economics department.*
[3]***David de la Fuente*** *is a Full Professor at the University of Oviedo in the Business Management department.*
****Corresponding with author*** – rosillo@uniovi.es

1. Introduction

Quantitative decision making in financial markets is a topic of constant innovation. Artificial Intelligence is helping investors in this decision making. In order to manage the money, trading systems are using Neural Networks (NN), Genetic Algorithm (GA), Fuzzy Logic and more recently Support Vector Machines (SVMs). In this article, a trading system based on Support Vector Machines is developed.

The goal of this study is to understand where the trading system model is working better. In order to know this, different simulated markets are presented. The first simulated market is characterised by a high volatility, the second simulated market is characterised by low volatility, the third simulated market is characterised by a bullish trend and the last one is characterised by a bearish trend. The different simulated markets are developed by a GARCH model based on the S&P 500. A trading system for the prediction of the directional weekly movement of each kind of market has been developed in order to achieve the aim of the paper.

SVMs are a supervised learning technique used for data analysis and pattern recognition mainly in classification problems with an increasing number of real-world applications including finance. The parameters of the SVM such as kernel function and C parameter are changed in order to achieve better results.

Technical analysis is widely used by investors (Taylor and Allen, 1992) to make decisions. Due to this importance, two of the main indicators of this analysis have been considered like inputs of this SVM, such as, Relative Strength Index (RSI) and Moving Average Convergence Divergence (MACD).

In this paper, we propose an intelligent stock trading system based on Support Vector Machines using Technical Analysis (RSI and MACD). The results demonstrate that this algorithm obtains better profits than Buy and Hold (B&H) and Naïve strategy, especially, in the simulated markets with high volatility.

The rest of the paper is structured as follows. In Section 2, the literature review relevant to the SVM and Technical Analysis is presented. Section 3 explains the simulation markets procedure. Section 4 explains the SVM trading algorithm created. Section 5 shows the empirical results of the trading system. Finally, Section 6 provides some concluding remarks.

2. Literature Review

In this section, the literature review relevant to the SVM and Technical Analysis is presented.

2.1 Support Vector Machines

A basic theory of the Support Vector Machine Classifier model is presented. SVMs are specific learning algorithms characterised by the capacity control of the decision function and the use of kernel functions (Vapnik, 1999; Cristianini and Taylor, 2000). The correct selection of the kernel function is very important.

SVMs were originally developed by Vapnik (1998). For a detailed introduction to the subject, Burges (1998) and Evgeniou et al. (2000) are recommended.

The methods based on kernel functions suggest that instead of attaching an algebraic correspondence to each element of the input domain represented by

$$\Phi : X \rightarrow F \qquad [1]$$

a kernel function

$$K : X \times X \rightarrow R \qquad [2]$$

is used to calculate the similarity of each pair of objects in the input set. An example is illustrated in Figure 1 (Huang and Sun, 2001).

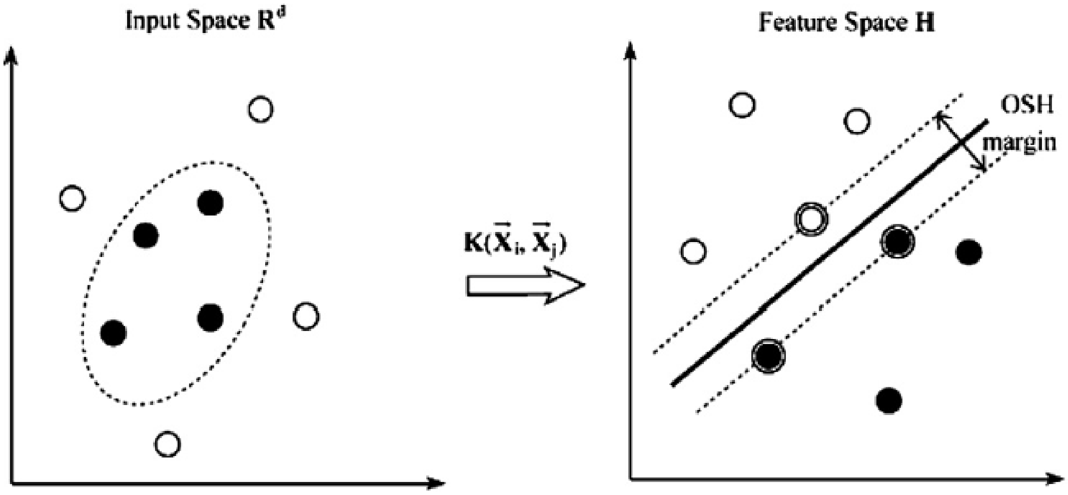

Figure 1. An example of how a kernel function works

The biggest difference between SVMs and other traditional methods of learning is that SVMs do not focus on an optimisation protocol that makes minimal errors as with other techniques. Traditionally, most learning algorithms have focused on minimising errors generated by their models. They are based on what is called the principle of Empirical Risk Minimization (ERM). The goal of SVM is different. It does not seek to reduce the empirical risk of making just a few mistakes, but intends to build reliable models. This principle is called Structural Risk Minimization. The SVM searches a structural model that has little risk of making mistakes with future data.

The main idea of SVMs is to construct a hyperplane as the decision surface so that the margin of separation between positive and negative examples is maximised (Xu et al., 2009); it is called the Optimum Separation Hyperplane (OSH), as shown in Figure 1.

Given a training set of instance-label pairs $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, m$ where $\mathbf{x}_i \in \Re^n$ and $y_i \in \{1, -1\}$, indicating y_i the class to which the point $\mathbf{x}_i$ belongs, the SVMs require the solution of the following problem:

$$\min_{\mathbf{w},\xi,b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \right\} \qquad [3]$$

subject to:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1 - \xi_i$$

$$\xi_i \geq 0 \qquad [4]$$

where ξ_i are the slack variables introduced by the method which measure the degree of misclassification of the data $\mathbf{x}_i$; $\mathbf{w}$ is the normal vector to the hyperplane; b is the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$; and $C > 0$ is the penalty parameter of the error term. Different values of parameter C are tested in order to achieve the best results to forecast the movement.

SVMs can be used in two different ways: for classification or regression. Recently there have been reports of the use of SVMs to solve financial forecasting problems.

Two applications on SVM financial time series forecasting were developed in 2003: in Cao and Tay (2003), SVM are applied to the problem of forecasting several futures contracts from the Chicago Mercantile Market, showing the superiority of SVMs over back-propagation and regularised Radial Basis Function Neural Networks; in Kim (2003), SVMs are used to predict the direction of change in the daily Korean composite stock index and they are benchmarked against back-propagation neural networks and Case Base Reasoning. The experimental results show that SVMs outperform the other methods and that they should be considered as a promising methodology for financial time-series forecasting. In Huang et al. (2005), a Support Vector Machines Classifier is used to predict the directional movement of the Nikkei225 index with extremely promising results. Also Ince and Trafalis (2006) try to solve portfolio problems optimisation using SVM.

Lastly, Lee (2009) explains a prediction model based on SVM with a hybrid feature selection to predict the trend of stock markets. It is shown that SVM outperforms Back Propagation Neural Network in the problem of stock trend prediction. Dunis et al. (2013a) show that it is possible to forecast some periods of IBEX-35 index under some chosen risk-aversion parameters using SVM Classifier. In Dunis et al. (2013b), a genetic algorithm was used to optimize the inputs selection procedure and the parameters of a SVM model. This methodology was applied to the one day ahead forecasting and trading problem using the FTSE100 and ASE20 indexes. A new financial oriented fitness function plus confirmation filters and leveraging techniques were applied to improve the performance of the overall methodology. Experimental results indicated that this method outperformed more classical techniques such as MACD, ARMA models, Bayesian predictors and Higher Order Neural Networks.

2.2 Technical Analysis

The main literature review on technical analysis is Menkhoff and Taylor (2007). Four arguments are analysed: technical analysis may exploit the influence of central bank interventions, the foreign exchange markets may be characterised by not-fully-rational behaviour, technical analysis may be an efficient form of information processing and it may provide information on the non-fundamental influences on foreign exchange movements. This study will focus on the last two arguments.

Almost all foreign exchange professional traders use technical analysis as a tool in decision making at least to some degree and the relative weight given to technical analysis as opposed to fundamental analysis rises as the trading or forecast horizon declines, as shown by Menkhoff and Taylor (2007). Technical analysis is used more than fundamental analysis; according to Taylor and Allen (1992), 90% of polled investors use it. Allen and Taylor (1990) and Taylor and Allen (1992) document systematically for the first time that technical analysis is, indeed, an important tool in decision making in the foreign exchange market.

There are many more recent studies which recommend the use of technical analysis for trading rules. Brock et al. (1992) prove that the use of moving averages and the use of supports and resistances as trading tools for the technical analysis of companies of the Dow Jones index from 1897 to 1986 generates better profitability than the Buy and Hold strategy for the same Index. Mills (1997) shows a similar result to the one considered in the previous article, but for the Financial Times Institute of Actuaries 30 (FT30 Index).

Kwon and Kish (2002) document that technical trading rules achieve better profitability than the Buy and Hold strategy in the NYSE while Chong and Ng (2008) recommend the use of technical trading rules using the RSI and MACD indicators for the FT30 index and they show that the use of both oscillators generates a greater profitability than the Buy and Hold strategy. Rosillo et al. (2013) recommend the use of technical trading rules using the RSI indicator for blue chips and Momentum indicator for small caps and they show that the use of both oscillators generates a greater profitability.

Finally, Rodriguez-Gonzalez et al. (2011) develop systems trading with Neural Networks based on RSI financial indicator.

As it has been described, there are studies that support the validity of Technical Analysis and stochastic indicators in order to forecast stock markets, and this is the main motivation why RSI and MACD have been used as inputs of the SVM.

3. Simulation Procedure

Financial returns series are mainly characterised by being zero mean, exhibiting high kurtosis and little, if any, correlation. The squares of these returns often present high correlation and persistence, which makes ARCH-type models suitable for characterising the conditional volatility of such processes; see Engle (1982) for the seminal work, Bollerslev *et al.* (1994) for a survey on market volatility models and Engle and Patton (2001) for several extensions.

3.1. GARCH(1,1) Model

The GARCH(1,1) model provides a simple representation of the main statistical characteristics of return series for a wide range of assets and, consequently, it is extensively used to model real financial time series. It serves as a natural benchmark for the forecast performance of heterocedastic models based on ARCH. In the simplest set up, if R_t follows a GARCH(1,1) model, then

$$R_t = \mu + \sigma_t \varepsilon_t$$
$$\sigma_t^2 = \omega + \alpha R_{t-1}^2 + \beta \sigma_{t-1}^2 \qquad [5]$$

where ε_t is an uncorrelated process with zero mean and unit variance. Following the definition in (3.1), the conditional variance σ_t^2 is a stochastic process assumed to be a constant plus a weighted average of last period's forecast, σ_{t-1}^2, and last period's squared observation, R_{t-1}^2. The parameters ω, α and β must satisfy that $\omega > 0, \alpha, \beta \geq 0$ to ensure that the conditional variance is positive. The process R_t is stationary if $\alpha + \beta < 1$.

We can define the unconditional, or long-run average, variance σ^2, to be:

$$\sigma^2 = E[\sigma_t^2] = \omega + \alpha E[R_{t-1}^2] + \beta E[\sigma_{t-1}^2] = \omega + \alpha \sigma^2 + \beta \sigma^2$$

so $\sigma^2 = \omega/(1-\alpha-\beta)$ [6]

If we assume $\omega = \sigma^2(1-\alpha-\beta)$ we get the desirable property that the GARCH model implicitly relies on σ^2. Substituting it in [5], variance equation, we get:

$$\sigma_t^2 = \sigma^2(1-\alpha-\beta) + \alpha R_{t-1}^2 + \beta\sigma_{t-1}^2 = \sigma^2 + \alpha(R_{t-1}^2 - \sigma^2) + \beta(\sigma_{t-1}^2 - \sigma^2) \qquad [7]$$

So we can see the GARCH model as a weighted average of the long-run variance. The contemporaneous variance is the long run average variance with one term added (substracted) if *t*-1's squared return is above (below) its long-run average, and other term added (substracted) if *t*-1's variance is above (below) its long-run average.

The sum $\alpha + \beta$ is named the persistence of the model. A high persistence, $\alpha + \beta$ close to 1, implies that socks that push variance away from its long-run average will persist for a long time.

The GARCH parameters are usually estimated using Maximum Likelihood (ML) procedures that are optimal when the data has been drawn from a Gaussian distribution. This model is usually estimated using the (conditionally) Gaussian log-likelihood function and maximizing it through an iterative algorithm such as BHHH (Berndt *et al.*, 1974), because the functional to be maximized is non-linear in its arguments. The estimates are called maximum likelihood (ML) when the Gaussian distribution is the underlying probability density function the data has been sampled from, or quasi-ML, otherwise. Bollerslev and Wooldridge (1992) have shown the consistency of these estimates in this case, which does not ensure that for a finite sample set it is the best estimate.

3.2 The S&P 500 Model

The simulated markets will be designed following the main parameters of the S&P 500 index market, in order to reflect as much as possible the real market situations. The daily data of S&P500 index are taken from 2001 to 2010, and its GARCH parameters are estimated. Then, some model parameters will be modified to create new simulated markets, trying the simulated markets involve real situations.

The empirical unconditional daily return variance is $\sigma_d^2 =$ 0.00018965, the annual variance $\sigma_a^2 = 250\,\sigma_d^2 =$ 0.047413, and the annual volatility $\sigma_a =$=0.21775 (21.775%). The studied period comprise years with high volatility (40% in 2008) and other years with low volatility (10% in 2005 or 2006).

Using maximum likelihood (ML) we estimate the equations [5] GARCH model parameters. The parameter ω is calculated with the constrained condition [6], so the model matches well the unconditional variance. The S&P 500 GARCH (1,1) model in 2001-2011 period is:

$$\begin{aligned} R_t &= \sigma_t \,\varepsilon_t \\ \sigma_t^2 &= 3.0\,10^{-7} + 0.081 R_{t-1}^2 + 0.913\sigma_{t-1}^2 \end{aligned} \qquad [8]$$

The mean of the daily financial return can be neglected without significantly degrading the performance of the proposed model. In Figure 2, the original S&P 500 index and the corresponding GARCH simulated series are shown. The innovations used in the model are the corresponding ones to the original market data.

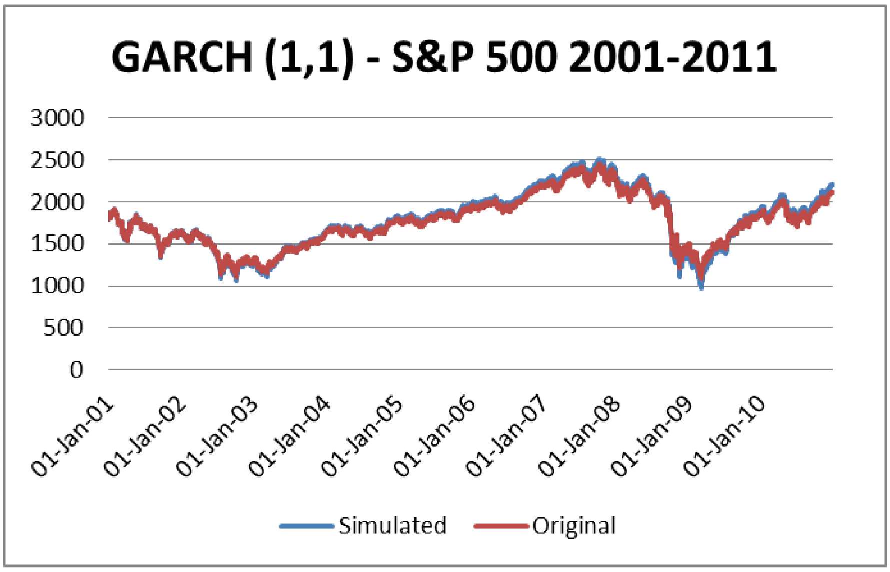

Figure 2. S&P 500 index, original and GARCH(1,1) model simulated, 2001-2011 period.

3.3 Simulated Markets

The simulation process is based on the previous GARCH model, generating new random innovations ε_t that will determine the particular results for each stochastic process. Different kind of simulated markets are generated by the changing of some specific parameters model. This is the procedure we are using to generate the different simulated markets.

The simulated markets can be classified in this paper by four kind of markets: bullish trend (Figure 3), bearish trend (Figure 4), high volatility (Figure 5) and low volatility (Figure 6).

Bullish Trend

Market is characterised by S&P 500 parameters, but we introduced an annual drift $\mu_a = 4\%$ so the model follows a light bullish trend.

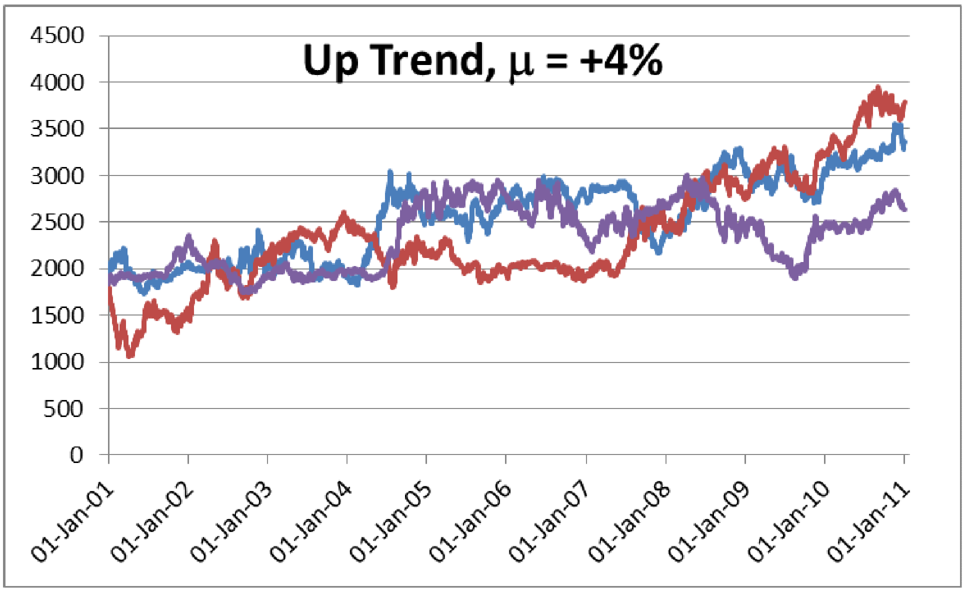

Figure 3. Bullish trend.

Bearish Trend

Market is characterised by S&P 500 parameters, but we introduced an annual drift $\mu_a = -4\%$ so the model follows a light bearish trend.

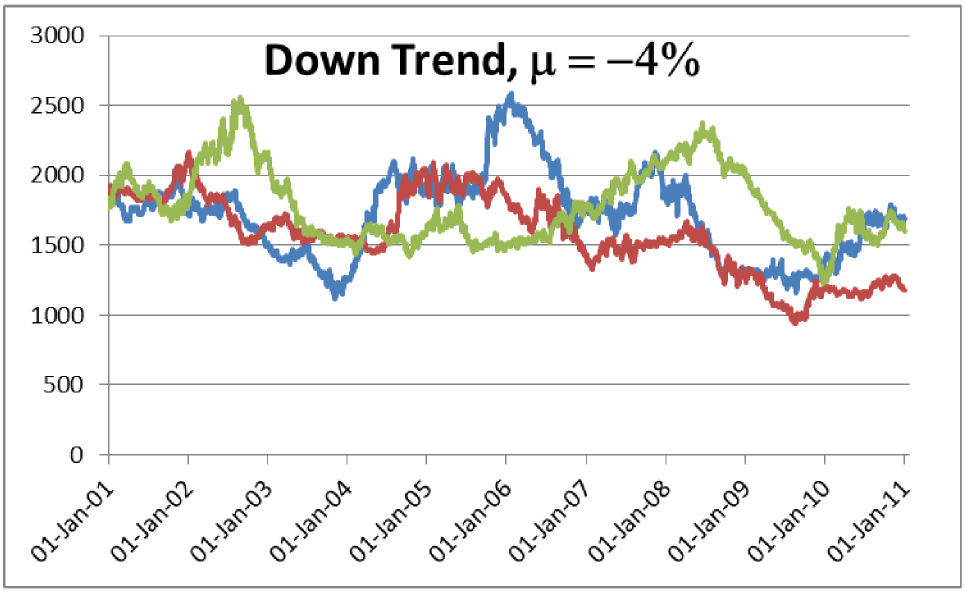

Figure 4. Bearish trend.

High Volatility

Market is characterised by S&P 500 parameters, but we introduced a double annual volatility $\sigma_a = 43\%$.

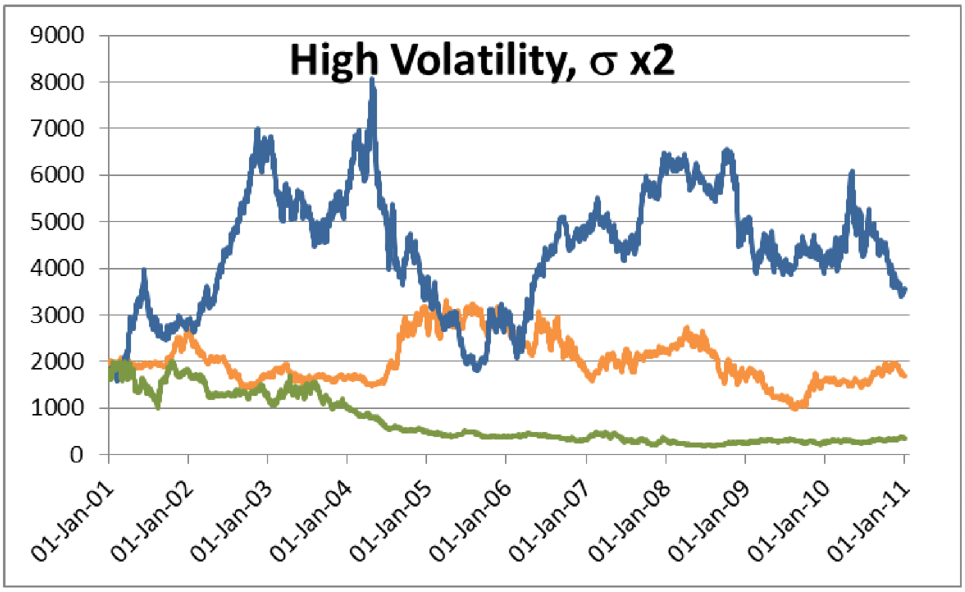

Figure 5. High volatility.

Low Volatility

Market is characterised by S&P 500 parameters, but we introduced the half of the annual volatility $\sigma_a = 10.9\%$.

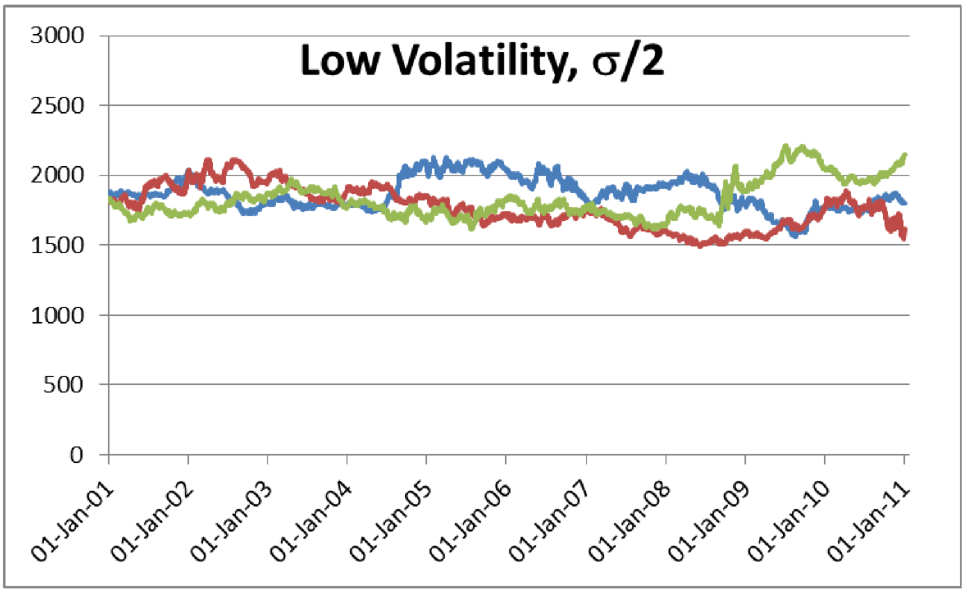

Figure 6. Low volatility.

4. SVM Trading Rule

The design of the experiment and the trading rule is presented in this section. The algorithm has been developed in Matlab[1]. An outline of the design of the trading rule is shown in Figure 7.

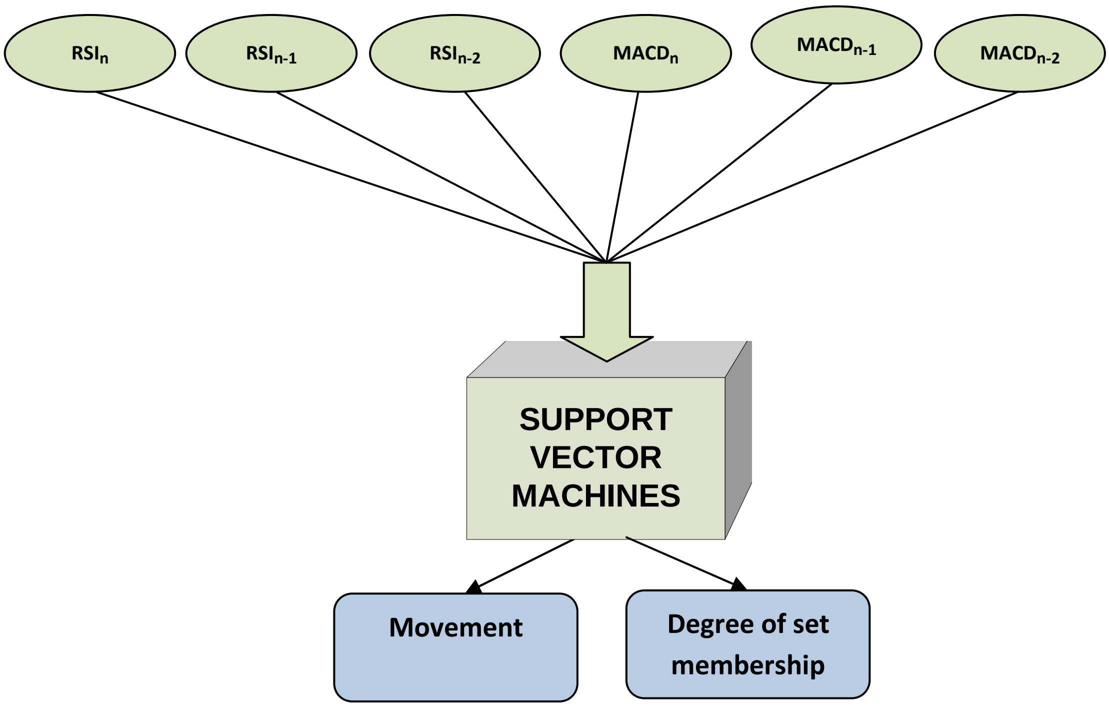

Figure 7. Design of the SVM trading rule

[1] The software used is MATLAB 7.8.0 (R2009a).

4.1. SVM Inputs

The inputs of the SVM are the quantitative analysis indicators RSI and MACD. In Rosillo et al. (2013) explain that RSI gets good profits in blue chips and Momentum indicator gets good profits in Small Caps, MACD and Stochastic indicators have been analysed over the Spanish Continuous Market too.

- Relative Strength Index (RSI)

It was designed by J. Welles Wilder Jr. (1978). A brief explanation of this indicator is shown below in equation [9]. If more details are needed it can be seen in J. Welles Wilder Jr. (1978).

The RSI is an oscillator that shows the strength or speed of the asset price by means of the comparison of the individual upward or downward movements of the consecutive closing prices.

For each day, an upward change (U_t) or downward change (D_t) is calculated. "Up days" are characterised by the daily close S_t being higher than the close of previous day S_{t-1}.

$$U_t = S_t - S_{t-1}$$

$$D_t = 0$$

"Down days" are characterised by the daily close being lower than the close of the previous day.

$$U_t = 0$$

$$D_t = S_{t-1} - S_t$$

The average U_t and D_t are calculated using an *n*-period exponential moving average (EMA_n).

Relative Strength Index at time *t* (RSI_t) is the following ratio between 0 and 100:

$$RSI_t = 100 \frac{EMA_n^U}{EMA_n^U + EMA_n^D} \qquad [9]$$

The 14-day RSI, a popular length of time utilised by traders, is also applied in this study. The RSI ranges from 0 to 100 however the range has been normalised between -1 and +1 in order to place it in the SVM.

- Moving Average Convergence Divergence (MACD)

The $MACD$ is designed mainly to identify trend changes. It is constructed based on moving averages and is calculated by subtracting a longer exponential moving average (EMA) from a shorter EMA. The $MACD$ is shown in equation [10]:

$$MACD(n) = EMA_k(i) - EMA_d(i) \qquad [10]$$

where:

$$EMA_n(i) = \alpha * S_i + (1-\alpha) * EMA_n(i-1)$$

$$\alpha = \frac{2}{1+n}$$

being n the number of days in the exponential average, and S_i is the asset price on i^{th} day.

In this article, k=12 and d=26 day EMA's are selected, which are commonly used time spans in order to calculate $MACD$ (Murphy, 1999).

The range of $MACD$ has been normalised between -1 and +1 in order to use it in the SVM.

4.2. The SVM Trading Rule

The SVM trading rule is explained in Rosillo et al. (2014). The best configuration of the SVM is getting by cross-validation algorithm. In this paper, we use the SVM trading rule from Rosillo et al. (2014) to analyse which is the best simulated market where it works. The only change that it has been made is in the inputs of the trading rule where VIX is avoided because we are working in this study with simulated markets.

An SVM Classifier has been chosen in order to make the quantitative decision. As it was explained in section 2, Support Vector Machines are helping investors in the decision making and many experiments demonstrate that SVMs generate better results than other Artificial Intelligence techniques.

The training period lasts 249 days and the following day (day 250) is tested by the SVM in order to know if the result is a good decision or not. Other periods such as 200 days, 300 days and 500 days have been tested as well but the best results are achieved with 250. So, the training period is 249 days and the testing period is 1 day. In total each experiment consists of 250 days, very similar to the length one of business year.

Although our dataset is daily, the trading strategy relies on a weekly prediction of the simulated market price move. A weekly forecast was selected as the expected price move, up or down, over a week is more significant.

The only problem that has been detected is in the situation when the SVM is being trained and data does not exist to make comparisons in order to take the decision to buy or sell. This situation happens in the last 5 days of the training period. In this way, the study is more real. In order to fix this, four experiments have been carried out, for example, compare these 5 days with the last day known, delete these 5 days, compare these 5 days with a simple moving average of those 5 days and compare these 5 days with a weighted average of those 5 days. The best results achieved are shown in the results section.

The following example is presented in order to clarify the previous explanation:

Let us start with the following situation:

	Training									Testing
Day	241	242	243	244	245	246	247	248	249	250
Daily Close Price	7	5	3	3	4	6	10	8	1	7
Decision	Sell	Buy	Buy	Sell	?	?	?	?	?	?

Table 1. Training data with unknown values.

Training data are from day 1 to day 249. In Table 1, data from day 241 to 250 are shown. The Sell/Buy decision is made by comparing the current day value with the value of 5 days ahead. In the case of days 245 to 249, the 5 days ahead value is unknown. Thus, to have a value to compare with, a simple moving average with values of days 245 to 249 is done.

Table 2 would be as follows:

Simple Moving Average: 4 + 6 + 10 + 8 + 1 = 29 / 5 = 5.8

	Training									Testing
Day	241	242	243	244	245	246	247	248	249	250
Daily Close Price	7	5	3	3	4	6	10	8	1	7
Decision	Sell	Buy	Buy	Sell	Buy	Sell	Sell	Sell	Buy	SVM decision

Table 2. Training data with known values.

In consequence, SVM would be trained using the Table 2. Day 250 decision would be taken by the SVM.

The SVM procedure can be described as follow.

Firstly, the SVM analyses the inputs classified in Buy or Sell situations.

Secondly, the SVM tries to separate the different prices of the simulated markets in two classes: Buying and Selling situations, with the inputs mentioned earlier.

Thirdly, the SVM uses the kernel function Heavy Tailed Radial Basis Function (HTRBF, equation [11]) in order to make the forecasting. HTRBF was developed by Chapelle et al. (1999) and is used by SVM-KM Matlab toolbox developed by Canu et al. (2005). The parameter C of the SVM is tested in several experiments and its optimal value is 10.

$$e^{-\rho \sum_j \left| x_j^a - y_j^a \right|^b} \quad \text{with} \quad a \le 1 \text{ and } b \le 2 \qquad [11]$$

Fourthly, the hit ratio is calculated for the different testing periods.

Finally, given a value of the RSI and MACD, the SVM predicts the upward or downward movement for the following week and the intensity of that movement.

4.3. SVM Outputs

The outputs of the SVM are the up or down movements, expected for the index the following week, and its degree of set membership.

5. Experiments

The main results are shown in this section. The SVM trading rule is benchmarked against a Naïve strategy and Buy and Hold strategy.

The trading strategy method is explained below: For each day, the simulated market index is bought or sold depending on the trading system recommendation. After 5 days, the reverse operation over the simulated market index is applied in order to be out of the market. This sequence is repeated every day. 5 contracts can be accumulated as maximum in the generated portfolio. Maximum Drawdown, standard deviation, daily return and Sharpe ratio are calculated based on the achieved results of this strategy. The daily return is annualised, $R_a = 250\,R_d$ and daily volatility is annualised as well, $\sigma_a = \sqrt{250}\,\sigma_d$

A simulated market is composed of 2515 days and this is 503 five days periods (week). The SVM needs an initial training period of 250 days, so 440 weeks are analysed in total.

For each kind of simulated market, three different simulated markets are generated. The obtained results are calculated by an average of these three simulated markets. So, the number of experiments in this study is 1320 in each kind of simulated market.

5.1. Benchmark Models

In this paper, we benchmark our SVM model with 2 traditional strategies, a Buy and Hold strategy (BH) and a Naïve strategy (N). For the sake of simplicity, we do not extend the analysis to other forecasting techniques, as autoregressive models or neural networks, because the goal of this paper is to find which kind of simulated markets is more predictable with the SVM strategy.

The performance of each strategy is evaluated in terms of trading performance via a simulated trading strategy.

5.1.1 Buy and Hold

The Buy and Hold strategy consists in buying the spot and holding the investment in the time without more decisions. In our case the investment period is 5 days. So we buy $S_{t-5},\dots,S_{t-1}$ and sell $S_t,\dots,S_{t+4}$.

5.1.2 Naïve strategy

The Naïve strategy takes the most recent period change as the best prediction of the future change, i.e. a simple random walk. The model is defined by:

$$\hat{R}_{t+1} = R_t$$

where the forecast rate of return $\hat{R}_{t+1}$ is based on the actual rate of return R_t.

As we are trading on 5 days periods, we have implemented this strategy buying the market S_{t-5} if $S_{t-6} > S_{t-7}$ or selling the market S_{t-5} when $S_{t-6} < S_{t-7}$, and doing the reverse operation 5 days later.

5.2 Experiment Results

The obtained results are shown in table 3, table 4, table 5 and table 6. Each table is corresponding with a kind of simulated market.

In the first set of columns, the achieved points are shown with each strategy, in the second set of columns the annualised return, in the third set of columns the annualised standard deviation of daily returns, in the fourth set of columns the Sharpe ratio and in the fifth set of columns the maximum Drawdown are shown. On the left side of the table, it can be seen the three bullish simulated markets, and the last row shows the average of the results.

The highlighted numbers present the best result of each strategy for a determinate indicator.

Table 3 explains the results for a bullish simulated market. BH achieves 774.8 points; SVM strategy achieves 60.8 points and Naïve strategy -403.9 points. En términos de rentabilidad, estos puntos de índice equivalen a una rentabilidad anual media de 6.4% en BH, 0.57% en SVM y -3.2% en Naïve. The annual volatility is not too high in the investments and takes 4.5% for BH, 6% for SVM strategy and 7% for Naïve strategy.

The Buy and Hold strategy is not a good benchmark in this kind of market because we are analysing a bullish trend so BH beats the other two strategies. SVM strategy improves the obtained results from the Naïve strategy, particularly; SVM strategy has a positive Sharpe Ratio instead of Naïve strategy with negative Sharpe Ratio. It is note to worth that SVM strategy achieves 60.8 points and Naïve strategy achieves -403.9 points.

Bullish Trend	SP Points (annual)			Ra(%)			σa(%)			SR=Ra/σa			MDD % (annual)		
Simulations	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N
Exp. 1	337.0	800.1	-425.5	2.6	6.4	-3.6	6.1	4.6	7.1	0.4	1.4	-0.5	33.9	36.8	38.1
Exp. 2	-303.3	1147.3	-592.5	-1.8	9.3	-4.5	6.1	4.6	7.1	-0.3	2.0	-0.6	36.1	35.6	37.4
Exp. 3	148.8	377.1	-193.8	0.9	3.4	-1.3	5.8	4.4	6.9	0.2	0.8	-0.2	33.2	36.6	35.8
Mean	**60.8**	**774.8**	**-403.9**	**0.57**	**6.4**	**-3.2**	**6.0**	**4.5**	**7.0**	**0.10**	**1.4**	**-0.5**	**34.4**	**36.3**	**37.1**

Table 3. Bullish trend results.

In Table 4, the results for the bearish simulated markets are shown. On the on hand, the greater annual mean return is 1.63% that corresponds to the SVM strategy. On the other hand, BH strategy and Naïve strategy obtain a negative annual return. We can also highlight that best results of Maximum drawdown and Sharpe ratio are obtained with SVM strategy.

Bearish Trend	SP Points (annual)			Ra(%)			σa(%)			SR=Ra/σa			MDD % (annual)		
Simulations	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N
Exp. 1	153.1	-17.0	-277.2	3.0	0.2	-3.5	6.9	5.2	8.0	0.4	0.0	-0.4	38.2	44.4	41.7
Exp. 2	104.8	-389.9	-135.7	0.6	-4.6	-1.5	5.8	4.4	6.9	0.1	-1.0	-0.2	34.1	39.7	35.7
Exp. 3	-9.0	-241.3	-150.0	1.2	-2.3	-1.8	5.3	4.1	6.2	0.2	-0.5	-0.3	30.8	36.3	33.0
Mean	**83.0**	**-216.1**	**-187.6**	**1.63**	**-2.2**	**-2.3**	**6.0**	**4.6**	**7.0**	**0.27**	**-0.5**	**-0.3**	**34.4**	**40.2**	**36.8**

Table 4. Bearish trend results.

In Table 5, we show the results for the high volatility simulated markets. This kind of market is probably the more difficult to forecast. The SVM annual return is 4.37%, this result is better than the other models. The lowest MDD is obtained by the SVM strategy, 69.1%. In Rosillo et al. (2014) use the Volatility Index (VIX) to forecast the S&P500 like an input for the SVM strategy to improve the results. This is a simulated market, so in this situation is not possible to use other index to improve results, although SVM strategy beats the other two strategies (BH and Naïve strategy).

High Volatility Simulations	SP Points (annual)			Ra(%)			σa(%)			SR=Ra/σa			MDD % (annual)		
	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N
Exp. 1	-284.8	655.8	297.0	-5.9	7.3	1.1	11.1	8.2	12.9	-0.5	0.9	0.1	65.1	70.1	68.6
Exp. 2	2599.4	22.8	789.5	11.0	2.3	4.3	14.4	10.7	16.8	0.8	0.2	0.3	76.1	92.4	88.8
Exp. 3	288.9	-755.5	-458.1	8.0	-16.8	-10.1	12.5	9.8	14.6	0.6	-1.7	-0.7	66.0	87.5	80.0
Mean	**867.8**	**-25.6**	**209.5**	**4.37**	**-2.4**	**-1.6**	**12.6**	**9.6**	**14.8**	**0.35**	**-0.3**	**-0.1**	**69.1**	**83.3**	**79.1**

Table 5. High volatility results.

In Table 6 we show the results for the low volatility market. The mean annual return is low in the three strategies, but only SVM strategy achieves profits. The lowest mean annual volatility is achieved by BH with 2.1%, but the best maximum drawdown and the highest Sharpe ratio are achieved by SVM strategy with the values: 16.1% and 0.02.

Low Volatility Simulations	SP Points (annual)			Ra(%)			σa(%)			SR=Ra/σa			MDD % (annual)		
	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N	SVM	BH	N
Exp. 1	145.2	-43.9	-84.4	1.4	-0.4	-0.8	2.9	2.2	3.4	0.5	-0.2	-0.2	16.6	19.1	17.9
Exp. 2	9.1	-181.8	-3.8	0.1	-1.9	-0.2	2.7	2.0	3.2	0.0	-1.0	-0.1	15.5	18.6	16.4
Exp. 3	-111.8	143.1	33.8	-1.4	1.6	0.3	2.8	2.0	3.2	-0.5	0.8	0.1	16.2	17.8	17.2
Mean	**14.2**	**-27.5**	**-18.2**	**0.05**	**-0.2**	**-0.3**	**2.8**	**2.1**	**3.3**	**0.02**	**-0.1**	**-0.1**	**16.1**	**18.5**	**17.2**

Table 6. Low volatility results.

6. Conclusions

The aim of this paper is to analyse in which kind of simulated market (bullish trend, bearish trend, high volatility and low volatility) the SVM trading strategy is useful. We have researched the profitability of simple technical trading rule based on SVM models.

An SVM algorithm has been chosen in order to make the quantitative decision. The main inputs of this algorithm are RSI and MACD. The outputs are the up or down movements expected for the index weekly change, and its degree of set membership (bullish or bearish class).

Three trading strategies are compared holding a maximum of five contracts at the same time in order to analyse the relevance of SVM in the different simulated markets during the quantitative decision making, although the most important thing is to compare the results of the SVM strategy in the four simulated markets.

Overall, this study shows that SVM strategy produces better results to Naïve strategy or Buy and Hold strategy in bearish markets, high volatility markets and low volatility markets. However, Buy and Hold strategy achieves higher returns in bullish markets, this situation is logical because Buy and Hold always earns more points in bullish markets, so it is not a good benchmark for bullish movements. If SVM strategy and Naïve strategy are compared for bullish markets, SVM strategy beats Naïve strategy as it can be seen in results section. These results are in line with Fernandez-Rodriguez et al. (2000) when applying nonlinear predictors to the Spanish Stock Market Index.

It is note to worth that SVM strategy achieves better results in high volatility markets than other kind of markets. These results are in line with Rosillo et al. (2014) where use VIX like an input of the SVM to forecast the S&P500. The SVM strategy allows a reduction in the Maximum Drawdown and a reduction in the annualised standard deviation. Also, Sharpe ratio is improved using SVMs. Furthermore, SVM trading strategy reduces the global risk of the investment.

The SVM strategy analysed would be useful in high volatility markets. The use of this algorithm would generate profits during financial crises.

However, some limitations are found in our work. For example, the algorithm is not able to achieve interesting results in low volatility markets. This could be due to the fact that SVM models works better in markets with high volatility because the price does not change significantly in low volatility markets.

As further work, it would be advisable to use a trend indicator in order to determine when the market is going to be immersed in a high volatility movement. Also, it would be interesting to combine this algorithm with an expert system in order to avoid days which are expected to be extremely volatile such as political crisis or FED decisions.

Another research would be to use XBRL. XBRL is a freely available and global standard for exchanging business information. XBRL allows us to obtain more information and to calculate more ratios in fundamental analysis for indexes and companies. It would be interesting, to combine different fundamental analysis indicators to improve the results.

7. Acknowledgments

Financial support given by the Government of the Principality of Asturias is gratefully acknowledged. The authors would like to thank Dr. Yuri Álvarez for his advices and support.

References

Allen, H. L. and Taylor, M. P. (1990) Charts, Noise and Fundamentals in the London Foreign Exchange Market. Economic Journal, 100, 49-59.

Berndt, E.K., B.H. Hall, R.E. Hall and J.A. Hausman (1974), Estimation inference in nonlinear structural models, Annals of Economy and Social Measurement, 4, 653-665.

Bollerslev, T. and J.M. Wooldridge (1992), Quasi Maximum Likelihood Estimation and Inference in Dynamic Models with time Varying Covariances. Econometric Reviews, 11, 143-172.

Bollerslev, T., R.F. Engle and D.B. Nelson (1994), “ARCH Models”, The Handbook of Econometrics, Volume 4, pp. 2959-3038. Elsevier Science B.V.

Brock,W., Lakonish, J., LeBaron, B. (1992), Simple technical rules and the stochastic properties of stock returns. Journal of Finance 47, 1731–1764.

Burges, C. (1998) A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2, 121-167.

Canu, S., Grandvalet, Y., Guigue, V., & Rakotomamonjy, A. (2005). SVM and Kernel Methods Matlab Toolbox, Perception Systèmes et Information, INSA de Rouen, Rouen, France.

Cao, L. and Tay, F. (2001) Financial forecasting using support vector machines. Neural Computing Applications, 10, 184-192.

Cao, L. and Tay, F. (2003) Support vector machine with adaptive parameters in financial time series forecasting, IEEE Transactions on Neural Networks, 14, 1506-1518.

Chapelle, O., Haner, P., Vapnik, V.N. (1999). Support Vector Machines for histogram-based image classification. IEEE Transactions on Neural Networks, 10(5):1055-1064

Chong, Terence Tai-Leung and Ng, Wing-Kam. (2008) Technical analysis and the London stock exchange: testing the MACD and RSI rules using the FT30, Applied Economics Letters, 15, 1111-14.

Cristianini, N. and Taylor, JS. (2000) An introduction to support vector machines and other kernel-based learning methods. New York: Cambridge University Press.

Dunis, CL., Rosillo, R., De la Fuente, D. and Pino, R. (2012). Forecasting IBEX-35 moves using support vector machines. Neural Computing and Applications. DOI: 10.1007/s00521-012-0821-9

Dunis, C., Likothanassis, S., Karathanasopoulos, A., Sermpinis, G. and Theofilatos, K. (2013) A hybrid Genetic Algorithm-Support Vector Machine Approach in the task of forecasting and trading the ASE 20, Journal of Asset Management, pp. 1-20, 2013, doi: 10.1057/jam.2013.2

Engle, R.F. and A.J. Patton (2001), What good is a volatility model?, Quantitative Finance, 1, 237-245.

Evgeniou, T., Pontil, M. and Poggio, T. (2000) Regularization networks and support vector machines. Advances in Computational Mathematics, 13, 1-50.

Fernández Rodríguez, F., González Martel, Ch. and Sosvilla Rivero, S. (2000). On the profitability of technical trading rules based on artificial neural networks: Evidence from the Madrid stock market. Economics Letters. Vol. 69, 89-94.

Huang, S. and Sun, Z. (2001) Support vector machine approach for protein subcellular localization prediction. Bioinformatics, 17, 721-728.

Huang, W., Nakamori, Y. and Wang, S. Y. (2005) Forecasting stock market movement direction with support vector machine. Computers & Operations Research, 32, 2513-2522.

Ince, H., and Trafalis, T. B. (2006) A hybrid model for exchange rate prediction. Decision Support Systems, 42(2), 1054-1062.

Kim, K. (2003) Financial time series forecasting using support vector machines, Neurocomputing, 55, 307-319.

Kwon, K. Y. and Kish, R. J. (2002) Technical trading strategies and return predictability: NYSE, Applied Financial Economics, 12, 639–53.

Lee, M.-C. (2009): Using support vector machine with a hybrid feature selection method to the stock trend prediction. Expert Systems with Applications, 36 (8), 10896–10904.

Menkhoff, L. and Taylor, M. P. (2007) The obstinate passion of foreign exchange professionals: Technical Analysis. Journal of Economic Literature. 45, 936-972.

Mills TC (1997) Technical analysis and the London stock exchange: testing trading rules using the FT30. Int J Finance Econ, 2, 319–331

Murphy, J. J. (1999) Technical Analysis of the Financial Markets. Institute of Finance, New York.
Rodriguez-Gonzalez, A., Garcia-Crespo, A., Colomo-Palacios, R. (2011) CAST: Using neural networks to improve trading systems based on technical analysis by means of the RSI financial indicator. Expert Systems with Applications, 38 (9), 11489-11500.
Rosillo, R., De la Fuente, D., Brugos, and JAL. (2013): Technical analysis and the Spanish stock exchange: testing the RSI, MACD, momentum and stochastic rules using Spanish market companies. Applied Economics, 45, 1541-1550.
Rosillo, R., Giner, J., and D. de la Fuente. (2014): The effectiveness of the combined use of VIX and Support Vector Machines on the prediction of S&P 500. Neural Computing and Application, Springer. To be published.
Tay, F. and Cao, L. (2001a) Application of support vector machines in financial time series forecasting. Omega, 29, 309-317.
Tay, F. and Cao, L. (2001b) A comparative study of saliency analysis and genetic algorithm for features selection in support vector machines. Intelligent Data Analysis, 5, 191-209.
Taylor, M. P. and Allen, H. L. (1992) The use of technical analysis in the foreign exchange market, Journal of International Money and Finance, 11, 304–14.
Vapnik, V. N. (1998) Statistical Learning Theory. New York. Wiley.
Vapnik, V. N. (1999) An overview of statistical learning theory, IEEE Transactions of neural networks, 10, 988-999.
Welles Wilder, J. Jr. (1978) New Concepts in Technical Trading Systems, Greensboro, N.C.: Hunter Publishing Company.
Xu, X., Zhou, C. and Wang, Z. (2009) Credit scoring algorithm based on link analysis ranking with support vector machine. Expert Systems with Applications, 36, 2625–2632.