



BI Norwegian Business School - campus Oslo

GRA 19703

Master Thesis

Thesis Master of Science

Can a trading strategy based on predictions from a nonlinear Support Vector Machine

outperform a passive investor holding the S&P500 index?

Navn: Kristian Opsahl, Marius Skyrud Harsjøen

Start: 15.01.2019 09.00

Finish: 01.07.2019 12.00

Can a trading strategy based on predictions from a nonlinear Support Vector Machine outperform a passive investor holding the S&P500 index?

Master Thesis

By

Marius Harsjøen and Kristian Opsahl

MSc in Finance

ABSTRACT

In this empirical research, we compare the forecasting performance of a supervised Support Vector Machine technique to a passive buy-and-hold strategy on the S&P500 index. By introducing two investment strategies, we find evidence that the application of a nonlinear Support Vector Machine can be superior to linear regression models, as well as to a passive buy-and-hold strategy. The Support Vector Machine model generates both excess returns and reduced volatility for the period between 2013 to 2019. However, when comparing the prediction results of a Support Vector Machine model to that of a linear regression model during the Great Recession, the results are ambiguous, although both models have proven to explicitly outperform the passive buy-and-hold approach.

Supervisor:

Tatyana Marchuk

This thesis is a part of the MSc programme at BI Norwegian Business School. The school takes no responsibility for the methods used, results found, or conclusions drawn.

List of Figures

FIGURE 1 - THE DECISION FUNCTION OF A SUPPORT VECTOR MACHINE	11
FIGURE 2 – INCCURED PENALTIES FROM MISCLASSIFICATION OF DATA POINTS	13
FIGURE 3 - HIGHER DIMENSIONAL DATA SPACE FOR REGRESSION SVM	15
FIGURE 4 - AUTOCORRELATION OF THE LOG RETURNS FOR S&P500 AT VARIOUS LAGS	25
FIGURE 5 - DAILY CUMULATIVE RETURN ESTIMATES OF THE S&P 500 BETWEEN 07/05/2013 – 05/03/2019	30
FIGURE 6 - PERCENTAGE VARIANCE OF LOG RETURNS FOR S&P500 EXPLAINED BY THE ORTHOGONAL COMPONENTS	32
FIGURE 7 - BUY-AND-HOLD VS MODEL STRATEGIES: 07/05/2013 – 05/03/2019	41
FIGURE 8 - MODEL PREDICTION FOR LONG/SHORT STRATEGIES WITH TRANSACTION COSTS	45
FIGURE 9 - PREDICTED CUMULATIVE RETURNS OF THE S&P500 BETWEEN 07/05/2013 – 05/03/2019	47
FIGURE 10 - LONG/SHORT STRATEGY WITH TRANSACTION COSTS BETWEEN 23/10/2006 – 31/12/2010	51
FIGURE 11 - CUMULATIVE RETURNS OF THE S&P500 BETWEEN 07/05/2013 – 05/03/2019	69

LIST OF TABLES

TABLE 1 - DESCRIPTIVE STATISTICS OF DAILY LOG RETURNS BETWEEN 02/01/1900 – 05/03/2019	20
TABLE 2 - LINEAR REGRESSION OF DAILY S&P500 LOG RETURNS FOR DATA BETWEEN 22/01/1990 – 06/05/2013	28
TABLE 3 - LINEAR REGRESSION OF THE DAILY S&P500 LOG RETURNS USING PCA FROM 22/01/1990 TO 06/05/2013	33
TABLE 4 - RESULTS OF PREDICTION ACCURACY FOR THE S&P500 USING SVM REGRESSION MODEL WITH DIFFERENT HYPERPARAMETER VALUES	37
TABLE 5 - PREDICTION COMPARISON OF THE BEST PERFORMING SVM MODEL TO THE LINEAR REGRESSIONS	39
TABLE 6 - INVESTMENT RESULTS FOR THE LONG/SHORT STRATEGIES	43
TABLE 7 - PREDICTION COMPARISON OF THE TWO TEST DATA SETS FOR THE LOG RETURN ESTIMATES OF THE S&P500	49
TABLE 8 - INVESTMENT RESULTS FOR THE LONG/SHORT STRATEGIES OVER THE INVESTMENT PERIOD 23/10/2006 - 31/12/2010	50
TABLE 9 - INPUT VARIABLES USED FOR FORECASTING	54
TABLE 10 - CORRELATION OF DAILY RETURNS OVER THE PERIOD BETWEEN 02/01/1990 TO 05/03/2019	68

Table of Contents

1. INTRODUCTION, MOTIVATION AND THEORY	1
2. HYPOTHESIS	3
3. RELATED LITERATURE	4
4. RESEARCH METHODOLOGY	7
<i>4.1 SUPERVISED MACHINE LEARNING</i>	7
<i>4.1.2 Main Challenges of Machine Learning</i>	8
<i>4.2 DATA</i>	9
<i>4.3 SUPPORT VECTOR MACHINE</i>	10
<i>4.3.1 Hyperparameters</i>	15
<i>4.4 PERFORMANCE MEASURES</i>	16
5. DATA PREPROCESSING AND DESCRIPTIVE STATISTICS	18
<i>5.1 CORRELATION</i>	23
6. DATA ANALYSIS	24
<i>6.1 LINEAR REGRESSION MODEL</i>	24
<i>6.2 PRINCIPAL COMPONENT ANALYSIS</i>	31
<i>6.3 DRAWBACKS OF LINEAR REGRESSION</i>	34
7. SUPPORT VECTOR MACHINE	35
<i>7.1 TUNING OF THE SVM REGRESSION FOR OPTIMAL PREDICTIONS</i>	36
8. RESULTS	39
<i>8.1 LONG-ONLY PORTFOLIOS WITHOUT TRANSACTION COSTS</i>	40
<i>8.2 LONG-SHORT PORTFOLIOS WITH TRANSACTION COSTS</i>	42
<i>8.3 ROBUSTNESS OF THE PREDICTION MODELS AND VALIDITY OF THE RESULTS</i>	46
<i>8.4 TESTING THE INVESTMENT STRATEGIES DURING THE FINANCIAL CRISIS</i>	48
9. CONCLUSION AND FURTHER ANALYSIS	52
APPENDIX	54
REFERENCES	70

1. Introduction, Motivation and Theory

The idea of machine learning is neither a new nor modern term. Arthur Samuel (2000), a pioneer of artificial intelligence research at IBM, coined the term “Machine Learning” back in 1959. However, the extreme advances seen within computational power and speed, have increased the possibilities and use of machine learning within most industries.

The contemporary financial market has been experiencing disruptive changes related to the development of technological progress. The increased implementation of robotic trading, machine learning algorithms, and Big Data processes are some of the factors that are currently shaping and transforming the financial industry. It is a broad consensus among the most significant financial players that the costs of not continuously participating in the development of technology, are severe for their respective investment performance. Human rationality is not particularly good at making fact-based decisions, especially when those decisions involve conflicts of interest. Additionally, we are slow learners, which puts us at a disadvantage in a fast-changing, fast-paced world like finance. It is easier to improve an algorithmic investment process than one relying entirely on human behavior (Agrawal, Gans & Goldfarb, 2018). As technology has amplified financial innovations, nonlinear forecasting tools have increased in popularity by producing profitable trading strategies that are outperforming standard econometric techniques.

We have two primary motivations for researching a topic revolving around machine learning. Firstly, we set the requirement that our thesis will provide us with knowledge and tools that contribute to our technical expertise, which can be beneficial for a future career in finance. Since machine learning and artificial intelligence most likely will be a prominent part of the financial industry, it is critical to master the complexity of the technical aspects. Secondly, compared to most other financial topics, there is a scarcity of academic papers in finance that implements machine learning techniques for forecasting, beyond measuring accuracy.

Stock market forecasts have an extensive literature where previous research on prediction, such as Fama (1998) and Loughran & Ritter (2000), claims that financial

markets are efficient and stock prices have a unit root¹, making it impossible to determine the development of the underlying stock price. Consequently, the distribution of returns presumes that an investor must assess the expected performance relative to risk in order to predict stock price changes. Through this paper, we will challenge the evidence found by previous researchers on the topic of stock market predictions. We will critically evaluate the technique of a supervised Support Vector Machine² (SVM) model and discuss if the method can be implemented as a trading strategy, without having access to unlimited computational power. It will be examined and assessed if the strategy can accurately forecast the development of a stock market. Additionally, we will analyze if the predictions can be transformed into a profitable trading strategy that can consistently outperform a passive buy-and-hold strategy.

Our empirical contribution builds on the foundation of implementing a machine learning technique to predict movements in the underlying S&P500 price index. We intend to incorporate an SVM technique by utilizing readily available data. The data set will comprise of 12 variables, in addition to the S&P500 price index, mainly consisting of other stock market indices, supplemented with interest rates, oil- and gold prices. We implement daily data where it is available, starting from 02/01/1990 until 05/03/2019. Our data period includes several significant events such as the IT-bubble of 2000, the Great Recession, the US election of 2016, and the continued low-interest-rate environment in the aftermath of the Great Recession. As most academic articles on SVM predictions target an audience of professionals within computer science, model specifications and technicalities for machine learning purposes have certain barriers for the financial reader. We aim to fully describe the statistical methods with a clear and concise structure by also considering the financial aspects behind the models.

¹ A unit root means integrated of order 1, which implies that both the mean and variance of stock prices depends on the previous price for the last period.

² Support Vector Machine (SVM), introduced by Boser, Guyon, and Vapnik in the Fifth Annual ACM Conference on Computational Learning Theory in 1992. V.Vapnik has since continued the research on this method and if the reader wants a thorough description, we suggest reading his book: "The nature of statistical learning theory".

2. Hypothesis

With this paper, we will, through an empirical research, evaluate the prediction accuracy of a supervised Support Vector Machine technique. We will apply an active investment strategy based on the estimated predictions and measure the results against a passive buy-and-hold strategy. Both strategies will be implemented by trading the underlying S&P500 index. Based on this, we have formed the following null hypothesis:

H0A: A buy-and-hold investment strategy of the S&P500 is superior to an active investment strategy that applies a Support Vector Machine technique

Against the alternative hypothesis:

H1A: The prediction results of the Support Vector Machine model can be applied to outperform a buy-and-hold strategy of the S&P500

Secondly, we want to determine if a nonlinear SVM model can be a better forecasting tool than a linear regression model when accounting for the prediction accuracy, as well as for the magnitude of the movements for the S&P500. The null hypothesis is:

H0A: The prediction results of applying a nonlinear SVM technique to forecast the returns of the S&P500, will be insignificantly different from the predictions of a linear regression model.

Against the alternative hypothesis:

H1A: The prediction accuracy of a nonlinear Support Vector Machine is superior of forecasting the returns of the S&P500 compared to linear regression models.

3. Related Literature

Stock market predictions are regarded as a challenging task for financial time series data since the stock market is inherently dynamic, nonlinear, complicated, nonparametric, and chaotic in nature (Abu-Mostafa & Atiya, 1996). Besides, a stock market is affected by numerous macroeconomic- and other factors such as political agendas, general economic conditions, policies of firms, environmental factors, expectations of investors, psychology and movement of other stock- and commodity markets. This evidence is supported by research conducted in the financial literature where stock-level predictors such as short-term reversal, momentum change, stock momentum, long-term reversal, recent maximum return, as well as industry momentum are shown to have significant forecasting abilities (See Fama and French, 2016). The interconnectivity of the global financial market has risen as technology has prospered. It provides opportunities to apply nonlinear models that can more accurately capture the interactions between numerous predictors.

Various models have been developed to predict stock market behavior, for example, one- or multi-step ahead price prediction, price change direction, returns and risks, portfolio assets allocation, and trading strategy decisions. Brock, Lakonishok and LeBaron (1992), find nonlinearities in market prices and show that the use of technical analysis indicators, under certain assumptions, may generate efficient trading rules. Hence, the adequacy of financial prediction using nonlinear models has spurred innovations within the industry. Earlier research claims that excess stock return predictability can be explained by a few robust factors using linear regression models (See Basu, 1977; Fama and French, 1988a, 1988b). However, the more generalized econometric model assumptions in the financial literature propose the opportunity of optimizing such models by configuring them to account for nonlinearities.

Kim (2003), introduces Support Vector Machine to predict the future direction of a stock price index. The study compares SVM with Back-propagation (BP)³ and Case-

³ For more information on BP Neural Network see: "The improvements of BP neural network learning algorithms, by Jin, Li, Wei and Zhen, 2000.

based reasoning (CBR)⁴. The main objective of his research is to forecast the direction of daily price changes on the Korean composite stock price index. 12 technical indicators make up the initial attributes and a total sample size of 2928 trading days from January 1989 to December 1998. 20% of the data is used for hold-out and 80% for training. A standard three-layer BP networks and CBR is used for benchmark. Overall, Kim concludes that SVM outperforms the other two techniques, however, not significantly.

Tay and Cao (2001) examine the feasibility of SVM in financial time series forecasting by comparing it with a multi-layer BP neural network. Additionally, they investigate the functional characteristics of SVMs for financial data. They collect data from five real futures listed at the Chicago Mercantile Market and transform the original closing price into a five-day relative difference in the percentage of the price (RDP). The most prominent advantage is that the distribution of the transformed data becomes more symmetrical and follows more closely a normal distribution⁵. The prediction performance is evaluated using the root mean squared error, mean absolute error, directional symmetry, and weighted directional symmetry. For the SVM model, they apply the Gaussian kernel as the kernel function together with the polynomial kernel. A standard three-layer BP network is the benchmark. Their experiment shows that SVM provides a promising alternative to BP neural network for financial time series forecasting. The predicted results of the SVMs forecast was significantly better than the BP network in four of the five futures.

Karathanasopoulos et al. (2013) introduce a novel hybrid Rolling Genetic- Support Vector Regression model (RG-SVR) to predict the directional movement of financial assets on the ASE20 Greek Stock index. The proposed hybrid consists of a combination of genetic algorithms with SVM modified to uncover effective short-term trading models and overcome the limitations of existing methods. Four

⁴ For more information on Case-Based Reasoning neural network see: "A neural network with a case based dynamic window for stock trading predictions, by Chang, Lie, Lin, Fan and Ng, 2009.

⁵ The input variables were determined from four lagged RDP values based on 5-day periods (RDP -5, RDP -10, RDP -15, RDP -20) and one transformed closing price which was obtained by subtracting a 15-day exponential moving average from the closing price (EMA15).

traditional strategies⁶ and a multi-layer perceptron neural network model is their benchmark for comparison. Their hybrid SVM model produces a higher trading performance in terms of annualized return and information ratio than all the benchmarks, even during the Great Recession.

Patel, Shah, Thakkar & Kotecha (2015) conduct a study comparing the prediction performance of SVM, Artificial Neural Network (ANN), random forest and Naïve-Bayes algorithms for the task of predicting stock and stock price index movements. Their dataset includes ten years of data from two stock price indices and two individual stocks⁷. First, they test the models with continuous-valued data. The results show an achieved accuracy ranging from 73.3% (naïve-Bayes model) up to 83.56% with a random forest model. However, when each model is trained using trend deterministic data, all of them improve their accuracy significantly. ANN is slightly less accurate in terms of prediction accuracy compared to the other three models which perform almost identically. The SVM achieved an accuracy of 89.33%, concluding that the model contains significant predictability potential.

The positive performance obtained from implementation of an SVM can be a result of robust model specifications, as well as appropriate input variables used for predictions. In their famous paper published in 2001, Microsoft researchers Michele Banko and Eric Brill suggest through their findings that it might be better to emphasize the analysis of data collection rather than the development of algorithms. Their study proclaims that when given enough data, very different Machine Learning algorithms, including relatively simple ones, perform almost identically well on a complex problem (M. Banko, E. Brill, 2001).

Most of the research conducted with an SVM technique shows promising forecasting results. However, we do want to highlight that from the studies we have examined, few goes beyond measuring the accuracy. The findings imply that a minority of the researchers have actually considered investment strategies from their prediction results. Secondly, from an investor standpoint we find it interesting that it is possible

⁶ A naïve strategy, a buy and hold strategy, a moving average convergence/divergence an autoregressive moving average model

⁷ The indices concern CNX Nifty and S&P BSE, while the stocks are Reliance Industries and Infosys Ltd

to achieve prediction accuracy well above 70 %. We consider it as an improbable possibility to obtain such high predictive power for daily financial data on a consistent basis. As a result, we will challenge these findings and discuss our results in comparison to other research on the topic of SVM forecasting.

This paper will highlight the forecasting results of a supervised SVM model for both classification and regression. We aim to highlight the strengths of such a model by discussing the importance of the configuration for its hyperparameters. Moreover, we will further address the limitations of an SVM model compared to a more ordinary technique like Ordinary Least Square. We will examine the prediction accuracy for both regression and classification and highlight the results by applying a long-only and a long-short trading strategy based on the forecasted results. By doing this, we can justify if the prediction accuracy can be applied, in order to obtain a profitable trading strategy.

4. Research Methodology

Within the field of machine learning, there are various types of approaches: Supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. In this paper, we will apply a supervised learning technique, and our description will emphasize this.

4.1 Supervised Machine Learning

In the context of artificial intelligence and machine learning, supervised learning is a type of machine learning algorithm that uses a known dataset, named the training dataset, to make predictions. The training data fed to the algorithm includes the desired solutions, named labels. Both input and output data are labeled for classification or regression to provide a learning basis for future data processing. Supervised machine learning systems provide the learning algorithms with known quantities through its training data, to support future judgments and are mostly associated with retrieval-based artificial intelligence. However, they may also be capable of using a generative learning model (Rouse, 2016).

The choice of what specific learning algorithm to use is a critical step. Once the preliminary testing is judged to be satisfactory, the classifier which are mapping from unlabeled instances to classes, is available for testing. The evaluation of the classifier is most often based on prediction accuracy, which we will measure through either correct predictions or estimated regression results. The most important supervised learning algorithms are k-Nearest Neighbors, Linear Regression, Logistic Regression, SVMs, Neural Networks, Decision Trees and Random Forest. We will for the purpose of this paper only focus on SVM and Linear Regression.

Generally, when it comes to utilizing Support Vector Machines for machine learning, the technique tends to perform significantly better when dealing with dimensions and continuous predictors, like stock price returns. Secondly, For SVM, a large sample size is required to achieve its maximum prediction accuracy, and it executes well when multicollinearity is present and nonlinear relationship exists between the input and output predictors (Kotsiantis, 2007). This is often apparent for financial time series data, which can be beneficial for our research.

The key question when dealing with any machine learning classification is not whether a learning algorithm is superior to others, but under what conditions a particular method can significantly outperform others on a given application problem (Kalousis, Gama and Hilario, 2004). After a better understanding of the strengths and limitations of each method, investigating the possibility of integrating two or more algorithms to solve a problem, should be a priority. The object is to utilize the advantages of one approach to complement the weakness of another (Wall, Cunningham, Walsh and Byrne, 2003).

4.1.2 Main Challenges of Machine Learning

Two main elements can cause the Machine Learning process to be unsuccessful. This can either be the algorithm or the data. With insufficient quality of training data or test data with have nonrepresentative observations, our models will not perform well regardless of how good the algorithm is. If the sample size has insufficient observations, we can potentially suffer from sampling noise⁸. On the other hand,

⁸ i.e nonrepresentative data as a result of chance

sampling bias can occur when extensive sample sets are nonrepresentative due to a flawed sampling method. Naturally, if our training data is full of errors, outliers and noise, it will make it nearly impossible for the system to detect underlying patterns. The best way of improving the performance is to clean up the training data as much as possible before its implemented (Géron, 2019). We must be sure not to include irrelevant data points as this would corrupt the training of our models. Lastly, we must be aware of overfitting⁹ and underfitting¹⁰ the training data. This will be further discussed when setting the parameters of the SVM model.

4.2 Data

To conduct this empirical research, we use daily adjusted closing prices starting from 02/01/1990, until 05/03/2019. We have selected 12 variables including the S&P500, where 8 of these are other stock indices, the VIX index which measures the volatility of the S&P500, a Treasury yield variable and two commodities¹¹. We provide a complete list of the variables, ticker names, and data descriptions in **Appendix 1**. We create two fixed subperiods of trading; the first 80% of the observations are our training set. The training data starts 02/01/1990 and ends on 06/06/2013. The last 20%, beginning 07/06/2013 and ends 05/03/2019 is our out-of-sample test data and will be our measurement for prediction accuracy based on the information collected in the training data. Since we are using daily data, we have chosen to start our data collection from the beginning of 1990. In this way, we obtain more consistent daily data from all the variables as some of the predictors have missing data points and inconsistent prices. The complete data set contains 7.350 daily observations. For a full description of the data collection and the processing of the explanatory variables, see **Appendix 2**.

To test our hypotheses, we examine two linear regression models, Ordinary Least Square (OLS) regression, and a regression model that implements a dimension

⁹ When the data does well on the training data but perform poorly in out-of-sample data

¹⁰ When the model is too simple to learn the underlying structure of the data

¹¹ Dow Jones Industrial Average (US), Nasdaq (US), Russel 2000 (US), Euro Stoxx 50 (EU), Euronext 100 (EU), FTSE100 (EU), Hang Seng (HK), SSE Composite (CH), 10-year Treasury yield, Gold price and WTI Crude Oil price

reduction technique through Principal Component Analysis (PCA)¹². We will discuss the advantages of PCA when we make predictions with highly correlated variables. The nonlinear model will be the supervised machine learning technique SVM. Moreover, the SVM techniques will impose both classification predictions, as well as regression predictions.

4.3 Support Vector Machine

A Support Vector Machine is a specific type of a supervised learning algorithm that classifies data from its characteristics. It is a statistical procedure where we transform complex data sets to help us produce better forecasting results. By estimating a function that is minimizing an upper bound of the out-of-sample error, SVM is proved to achieve a high generalization performance which is resistant to the overfitting problem (Huang, Nakamori and Wang, 2005). More generally, this implies that SVM is a better forecasting model than other statistical techniques who are optimizing prediction accuracy only on the training data.

By defining a hyperplane, the model will separate the data points on either side of the hyperplane in its data space (See Figure 1). For classification, it will imply that the optimal hyperplane will effectively try to determine the difference between an up- or a downward movement for the S&P500. The data set used as input for training the forecasting model will be as follows: $D = \{(x_i, y_i)\}_{i=1}^N$ where $x_i \in \mathbf{R}^n$ are all N explanatory variables at observation i used for prediction through their respective log return estimates. For regression results, the corresponding $y_i \in \mathbf{R}^n$ represent the log return estimates of the stock price for S&P500 at observation i and corresponds to the response variable. For classification, $y_i \in \{-1; +1\}$ implies that an increase in the daily log returns will be classified by $+1$, while a decrease in the daily return will be classified as -1 . The reason why we impose two estimates of y is because we aim to use SVM for the objective of comparing the results of both classification and regression. The data set used will be identical to what we will employ for the linear regression models.

¹² For an excellent description of PCA for financial analysis, see for example Kritzman, Li, Page and Rigobon (2011, p.112-126).

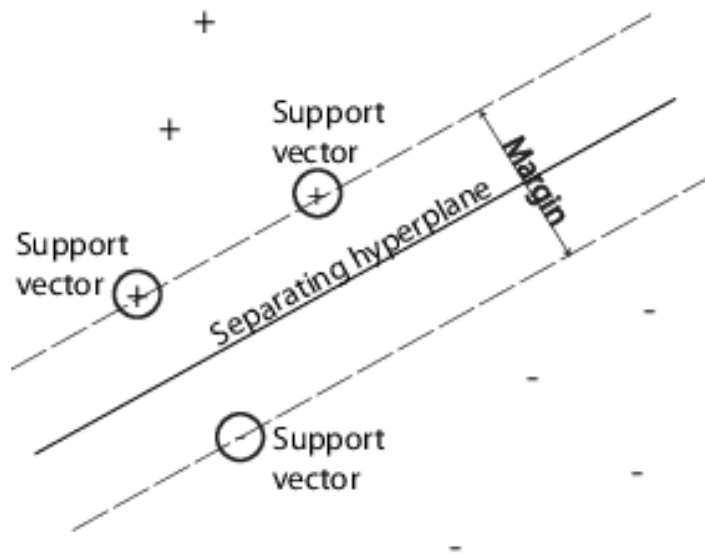
Figure 1 - The Decision Function of a Support Vector Machine

Fig 1. The separating hyperplane has a margin with an equal distance on each side of the hyperplane. The support vectors form the decision boundary and decides on the class which each data point will be assigned to. Illustration retrieved from MathWorks (2019).

The smallest distance between the data points and the hyperplane is named the margin of separation. The points that are the closest to the hyperplane will be assigned as support vectors and form the decision boundary of the hyperplane. The objective is to find the optimal separating point where the margin is maximized. When the margin width increases, the model becomes more efficient at accurately separating the positive data points from the negative ones. For our objective, we aim to predict the daily log returns of the S&P500 by accounting for the data points on either side of the hyperplane. The model will determine the classification based on which side of the hyperplane the data points will be assigned. The utilized model function will form a margin g that have an equal distance to the decision boundary on each side of the hyperplane. If g is the margin of the optimal hyperplane, we can find the support vectors by locating the points which are distance g away from the optimal separating hyperplane. Consequently, the support vectors will lie exactly on the margin g on each side of the hyperplane. To extract the decision boundary for the hyperplane, the following decision function is proposed for a linear SVM classifier:

$$y_i = w^T \theta(x_i) + b \quad (1)$$

Where w^T is the transpose of the feature¹³ weights vector, while b represents the intercept of the model and it is similar to the intercept of a linear regression model. $\theta: \mathbf{R}^n \rightarrow \mathbf{R}^m$ indicates that all the explanatory variable observations are transformed into a nonlinear and high dimensional data space where $n < m$. Hence, the mapping of the variables is a nonlinear function that can be depicted in an infinite high dimensional data space. However, this will never be explicitly calculated due to the kernel trick. For a complete description of the kernel functions and the applied kernel trick, see **Appendix 3**.

The decision function will predict the return of S&P500 for observation i . The prediction is denoted \hat{y}_i . This can be represented as a classification problem where:

$$\hat{y} = \begin{cases} 0 & \text{if } w^T \theta(x) + b < 0, \\ 1 & \text{if } w^T \theta(x) + b \geq 0 \end{cases} \quad (2)$$

The forecasting of \hat{y}_i will predict an up-movement of the S&P500 at observation i if the decision function is greater than or equal to zero, and a down-movement otherwise. The decision boundaries are the intersection of two hyperplanes, where the decision function is equal to zero.

Training the SVM to optimize the prediction accuracy implies finding the value of b and w that maximizes the margin g , while still avoiding margin violations related to overfitting- or underfitting the model. Margin violations will occur whenever a positive data point falls on the negative side of the hyperplane, or vice versa. The weight vector decides on the slope of the decision function. This can be compared to the beta estimate in a linear regression model, which determines the slope of the regression. As w gets smaller, the larger will the margin be. Hence, the objective is to minimize $\|w\|$ with the constraint that the decision function must be greater than 1 when the training instances are positive, while it must subsequently be smaller than -1 when the training instances are negative. We can define $t_i = 1$ for all positive

¹³ A feature represents certain characteristics that helps to describe the data. For our objective, we consider the characteristics of our explanatory variables to describe the movements of the S&P500 index.

instances if $y_i=1$, and $t_i = -1$ when $y_i=0$. This leads to the constraint $t_i(w^T\theta(x_i) + b) \geq 1$ for all i observations. Furthermore, because the data set is inseparable, meaning that we cannot correctly predict all the data points, we introduce a slack variable $\xi_i \geq 0$ for all i observations which measure the violations of the margin. The term imposes a penalty for all points that are assigned to the wrong side of its margin boundary. This proposes a secondary constraint where the objective is to minimize the errors ξ , to reduce the total number of margin violations:

Figure 2 – Incurred Penalties from Misclassification of Data Points

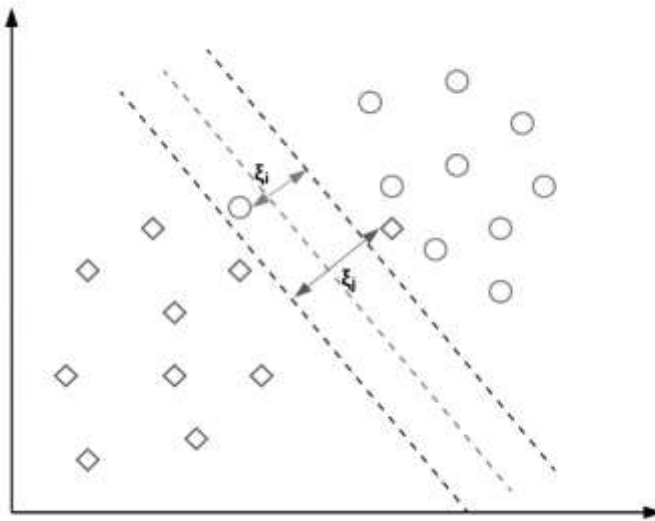


Fig 2. The data points that are classified incorrectly will incur a penalty equal to ξ , for violating the margin. The further away it is from its margin boundary, the larger the penalty that will be assigned to that specific observation. Illustration retrieved from Misra (2019).

This implies that the objective will be to maximize the distance between the positive and the negative data points used for predicting the S&P500 to avoid prediction errors. Furthermore, to make the predictors linearly separable, we will utilize a kernel function to account for the nonlinear characteristics of financial variables. As already described, financial data is nonlinear in nature and the introduction of a kernel function makes the process significantly more computational efficient to apply. The full derivations of the model specification for the SVM classifications are organized to **Appendix 4**. The mathematical derivations are shown for the interested reader to understand the constrained optimization problem in full.

Since we will apply a regression SVM, we have one additional specification that must be included. We must introduce an epsilon parameter ϵ , which is denoted the tolerance hyperparameter. The reason for this is that regression methods, like OLS, will try to predict the exact return estimates for the S&P500. The deviation between the predicted value and the actual observation is the residual value of the regression. In an SVM model, the objective is to find a function where all the predicted values deviate from the corresponding actual observation by no value greater than the epsilon parameter, ϵ . Hence:

$$|y_i - (w^T \theta(x_i) + b)| \leq \epsilon \quad \forall i \quad (3)$$

Where y_i is the actual observation of the log return for the S&P500 at observation i . This implies that the residuals from the decision function must have an absolute value less than ϵ for all observations. The decision function of the SVM is identical to the classification SVM. The difference between a classification- and a regression SVM model is that the regression technique will now decide on the total margin width. In an SVM regression model, the width of the margin from the optimal separating hyperplane is controlled by the epsilon parameter. As epsilon is increased, the margin for the hyperplane boundary is widened, causing more of the data points to be within the hyperplane boundaries. Simultaneously, the objective to limit margin violations where data points are outside the boundary, must be counter-balanced by not setting the parameter excessively low. The prediction for SVM regression will also include the implementation of a kernel function for mapping the variables into a higher dimensional data space. The full derivations of the SVM regression model can be seen from **Appendix 5**.

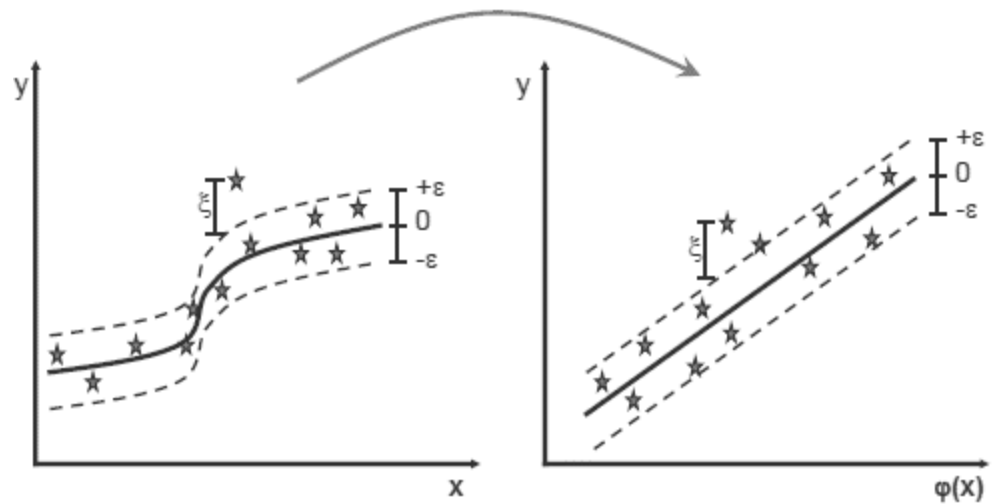
Figure 3 - Higher Dimensional Data Space for Regression SVM

Fig.3. The illustration shows how the kernel function transforms the input variables into a higher dimensional data space, making the decision function linear. ϵ sets the upper limit for the residual value of the predictions. Misclassifications incur a penalty, ξ , that is equal to the distance between the ϵ and the margin violation. Illustration retrieved from Sayad (2017).

4.3.1 Hyperparameters

The SVM models do require tuning of certain hyperparameters to optimize the prediction accuracy. In addition to the epsilon hyperparameter ϵ , there are two additional parameters which can be tuned and iteratively changed to optimize the model. See for instance Cherkassky and Ma (2004) for a thoroughly discussion on setting these three parameters. These parameters will iteratively be changed to determine their appropriate value. This involves adjusting them to obtain optimized predictions, while simultaneously avoid the issues of either overfitting- or underfitting the data.

Firstly, the box constraint, C , is introduced in the constrained optimization problem shown in **Appendix 4**, equation 3. It helps with the regularization of the model by defining the trade-off between the objective of minimizing the slope $\|w\|$, and the errors ξ , simultaneously. The value of C decides on the regularization of the data and controls the total number of misclassifications for the prediction. As the parameter is set lower, regularization is increased. When the value of C is close to zero, the model

will not be penalized by errors. It implies that even substantial misclassification will be acceptable since the decision boundary will be completely linear. On the opposite, an infinite large value of C will cause a highly penalized model. The classifier can no longer afford to misclassify the data points, and hence overfitting will most likely be introduced. Even though the model will be extremely accurate at predicting the training data, it will most likely fail to forecast adequately when the model is tested on a new data set.

Secondly, the kernel function allows us to pick a value for gamma, γ . Gamma represents the distribution of the input variables and acts as a regularization hyperparameter. For instance, a Gaussian kernel has a gamma parameter that forms the bell-shaped distribution of the variables. For a larger gamma value, the variance of the Gaussian is small, and hence, the bell-shaped curve will get narrower, causing the decision boundary to become more irregular. Each support vectors will have a smaller influence on the prediction of the movements in the S&P500. This can potentially propose the issue of overfitting the model. On the contrary, a low gamma value will cause the decision boundary to end up smoother, and support vectors will have a broader range of influence. This reduces the possibility of overfitting the model but comes at the expense of potentially not extracting the decision boundary which is the best suited to capture the complexity of the movements in the S&P500.

4.4 Performance Measures

Our objective is to predict the log returns of the S&P500 by incorporating a wide range of signals from the tests we implement. As described in 4.1 we create two fixed subperiods of trading, the first 80% of the observations are our training set, and the last 20% are our out-of-sample test where we measure the prediction accuracy based on the information collected in the training data.

Out-of-sample tests are applied for validation purposes and to conclude how robust each model is. We will highlight certain statistical performance measure from forecasting with various error estimations. Both classification of log returns, as well as the absolute log return estimates, will be highlighted. This implies that we will both consider the regression results, as well as its classification of an up- or down

movement in the return estimates. Generally, three types of error measures have been proposed in the financial literature for prediction of stocks when implementing regression results:

$$MSE = \frac{1}{N} \cdot \sum_{t=1}^N e_t^2 \quad (18)$$

$$RMSE = \sqrt{\frac{\sum_{t=1}^N e_t^2}{N}} \quad (19)$$

$$MAE = \frac{1}{N} \cdot \sum_{t=1}^N |e_t| \quad (20)$$

These error measures are the mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE), respectively. N is the total number of observations in the entire sample. e_t denotes the prediction error for the forecasted stock return at time t , where:

$$e_t = \hat{y}_t - y_t \quad (21)$$

\hat{y}_t is the predicted value of the forecasted stock return at time t , while y_t denotes the actual log returns of the S&P500. RMSE is simply the square root of MSE, and it is better known as the standard deviation of the residuals. We also consider the MAE, which accounts for the absolute values of the residual obtained.

Additionally, Atsalakis & Valavanis (2009a) and Leung, Daouk and Chen (2000), are arguing that the most valid performance measure is the accuracy of predicting the success rate of the stock forecasts. The hit rate of stock prediction is calculated as:

$$Hit\ rate = \frac{h}{N} \quad (22)$$

where h denotes the number of correct predictions of the stock trend and N denotes the number of tests conducted for predicting the outcome of the stock. This can be described as a classification technique where we assign a value of 1 for predicting the index to move in the same direction as the actual observation, or a value of 0 when it moves in the opposite direction. However, the hit rate is not able to assess the magnitude of the movements for the index. As a result, the outcome of the hit rate

and the error measurement may deviate and propose substantial different results. We will report all the presented performance measures for comparison. We can then make more adequate conclusions on the advantages and limitations of an SVM model when accounting for both classifications of stock returns, as well as regression results.

5. Data Preprocessing and Descriptive Statistics

The collection of predictive variables can be cumbersome due to the aggregate universe of forecasting predictors proposed in the financial literature. The number of variables and factors affecting the S&P500 are too numerous to list, and we will, therefore, address the importance of normalizing the input data to account for efficient use and reduced computational costs. Atsalakis and Valavanis (2009b) propose a list of various studies with a substantial number of input variables for each of the research papers. Additionally, Gu, Kelly and Xiu (2018) include 94 firm-specific predictors, eight time series variables, and 74 industry sector dummy variables, with more than 900 baseline signals.

To assess and determine the adequacy of the predictive ability of each variable, we can preprocess the data. Highly correlated variables or variables with insignificant power must be left out of the algorithm to reduce the computational costs. We highlight this through a dimension reduction technique. Data normalization will be conducted using Principal Component Analysis. PCA helps to overcome the issue of overfitting the model by using an orthogonal¹⁴ transformation to create a new set of linearly uncorrelated variables. Each succeeding variable will account for as much variation in the data as possible. This technique avoids suboptimal forecasts and helps to reduce noise by isolating the signals from the predictors.

Since stock price returns have properties that are convenient for time series analysis, we have transformed the prices of each respective variable into daily log returns to

¹⁴ For this example, orthogonal variables can be described as a set of variables that are all completely uncorrelated to each other.

overcome the issue of nonstationary stock prices. Prices are assumed to have a log-normal distribution, and by taking the first logarithmic differences of the prices, we are extracting log return estimates.

$$r_{i,t} = \ln\left(\frac{P_{i,t}}{P_{i,t-1}}\right) \quad (23)$$

Where $r_{i,t}$ is the daily log return of variable i at time t . $P_{i,t}$ is the daily closing price of variable i at day t , while $P_{i,t-1}$ is the daily closing price of variable i at day $t - 1$. All variables have been calculated in log returns.

Table 1 - Descriptive Statistics of Daily Log Returns Between 02/01/1900 – 05/03/2019

t=1, ..., T=7350 daily adjusted closing data observation

	Mean	Standard Dev	Skewness	Kurtosis	Augmented-DF	Jarque-Bera
<i>SPX</i>	0,03%	1,11%	-0,2582	11,8054	-90,3534**	23824**
<i>DJIA</i>	0,03%	1,06%	-0,1857	11,1697	-89,5667**	20480**
<i>ESTX50</i>	0,02%	1,34%	-0,1234	8,5039	-86,9046**	9295**
<i>ENX100</i>	0,01%	1,07%	-0,0788	12,9785	-86,535**	30497**
<i>FTSE100</i>	0,01%	1,09%	-0,0896	9,0879	-86,671**	11358**
<i>HSI</i>	0,03%	1,56%	-0,1005	13,8833	-86,4653**	36282**
<i>IXIC</i>	0,04%	1,44%	-0,1083	9,4238	-85,9408**	12650**
<i>RUT</i>	0,03%	1,30%	-0,3778	9,4799	-87,3507**	13032**
<i>SSE</i>	0,01%	2,05%	1,1298	27,208	-84,7844**	181010**
<i>VIX</i>	-0,01%	6,52%	0,8913	9,6485	-92,8838**	14508**
<i>WTI</i>	0,01%	2,39%	-0,7304	17,9172	-87,3253**	68792**
<i>TNX</i>	-0,02%	1,62%	-0,0758	8,7224	-85,3875**	10034**
<i>XAU</i>	0,02%	1,00%	-0,0993	11,8104	-86,1875**	23781**

Significant at 5% Level* *Significant at 1% Level*

Descriptive statistics of the daily log returns for the S&P500 index over the period 02/01/1990-05/03/2019. The mean and the standard deviations are denoted in percentage, while the skewness, kurtosis, Augmented DF, and the JB test are presented in absolute values.

The summary statistics in Table 1.0 presents the daily log return statistics of the provided financial variables for 7350 daily observations from 02/01/1990 to 05/03/2019. The mean daily log returns are close to zero for all the variables included in the data set. The daily volatility measured through the standard deviation of the log

returns are stable around 1-2%, but significantly higher for the CBOE Volatility Index (VIX). The VIX index approximates the expected future realized volatility of the S&P 500 return over the next 30 days. Bardgett, Gourier, and Leippold (2018) concludes that the index has a variance risk premium. Furthermore, they show that some of the dynamic properties of the S&P500 returns are better captured by the VIX and can be a contributing reason to its daily volatility. Since the returns of the VIX and the S&P500 are inversely correlated, an unexpected drop in the price of the S&P500 can cause an amplified increase in the price for the VIX index, leading to a more considerable spike in the daily volatility for the VIX.

To analyze if the log returns follow a normal distribution, we have considered the higher moments of the distribution to make more adequate conclusions. The third and the fourth moments are characterized as the skewness and the kurtosis¹⁵. The daily log returns indicate that the excess kurtosis and the skewness are nonzero for all the estimated variables. The majority of the variables are negatively skewed, while all of them have a leptokurtic distribution with a peak around its mean value, as well as fatter tails compared to a normal distribution. For SSE Composite, both skewness and kurtosis are significantly larger than for the other variables, even after we adjust for the first 606 observations, where the index moves irrationally. The inclusion of these observations would inflate the return statistics even more, as well as the Augmented Dickey-Fuller test¹⁶ and the Jarque-Bera test¹⁷ which would be 167,29 and 8 301 504, respectively. The SSE Composite is known for its inherent volatility and governmental regulations. Recent research conducted by Lin (2018) analyzes the SSE Composite Index and concludes that the returns present large leptokurtosis in its distribution. Furthermore, this is often the case for emerging stock markets with excessive governmental intervention, irrational investment behavior, and undeveloped financial infrastructure. Previous studies have captured similar results of the SSE Composite index (Darrat and Zhong, 2000, p.107).

¹⁵ For a full description of skewness and kurtosis, see Brooks (2019, p.66-67).

¹⁶ The Augmented Dickey Fuller test with a thorough explanation of stationarity in financial times series data are described by Pagan (1996, p.18-21).

¹⁷ Jarque-Bera tests for normality in the distribution of returns. See Pagan (1996, p.34-38) for a discussion on the elements of normality testing.

These results are an unambiguous indication that each of the variables has a distribution that cannot be reflected thoroughly by a normal distribution.

Furthermore, in **Appendix 6** we provide the model specifications of the implemented Augmented Dickey-Fuller and the Jarque-Bera test. From Table 1, it can be shown that the null hypothesis of a unit root is rejected for all the respective variables.

Hence, it implies that all the time series of returns are stationary, and significant at the 1% level. We can conclude that all the variables fluctuate around a constant long-run mean and has a finite variance which is independent of time, making forecasting of the data set feasible to undertake.

The Jarque-Bera test has been conducted to test for the normality of the distribution for each of the variables included in the data set. The test statistics are presented in Table 1. The values related to the Jarque-Bera test clearly states that the null hypothesis of normality is rejected at the 1% significance level for all the return time series. It implies that the error term of the distributions is not normal. However, when the sample size is sufficiently large, as what is the case for our model, the normality assumption can be neglected due to the Central Limit Theorem¹⁸. The random noise between the independent variables and the error term can still propose issues related to financial time series analysis, which can incur certain drawbacks when using a linear statistical model for prediction.

Altay & Satman (2005) points out that financial data violates the assumption of normality. Both skewness and kurtosis can make ordinary least square regressions a potential less efficient tool for forecasting. Hence, these results imply that a prediction procedure that does not require the assumption of normality can increase the accuracy of the model. Based on our descriptive log return statistics, we can form a null hypothesis that the prediction results of a linear regression model will be insignificantly different compared to a nonlinear SVM model. If this is proven wrong, it can be concluded that a nonlinear model can have certain properties which are superior to the linear models when predicting the daily log returns of the S&P500.

¹⁸ The Central Limit Theorem states that for a sufficiently large sample data set from a population with a finite variance, the mean of that data set will approximate the mean of the population.

5.1 Correlation

From Appendix 8 we have extracted the daily correlation for log return estimates among the selected variables. The S&P500 index is highly correlated with the US indices and has a significantly high correlation with the European indices as well. Furthermore, the S&P500 has a low correlation with Hang Seng, SSE Composite, and the WTI Crude oil index. As expected, we see the different indices have a higher correlation with those indices being geographically connected. Ramchand and Susmel (1998) provides interesting findings between volatility and cross-correlation for the US market. In a high variance state, the correlation between the US and other world markets are on average 2 to 3.5 times higher compared to a low variance regime. The findings in monthly postwar US data help to explain the low correlation between excess stock and bond returns. Stock and bond returns are primarily driven by news regarding future excess returns and inflation, respectively. Real interest rates have little impact on returns, although they do affect the short-term nominal interest rate and the slope of the term structure (Campbell and Ammer, 1993). Secondly, if we look at data from the early 1960s, during the 23 times the 10-year Treasury yield rose, the S&P 500 rose more than 80% of the time, indicating a positive correlation even though it is low.

Chang, McAleer, and Tansuchat (2013) study the conditional correlations and volatility spillovers based on the daily returns from 1998 to 2009 of the WTI and Brent markets together with the FTSE100, NYSE, Dow Jones and the S&P500. Their findings indicated a low correlation across markets, which supports our data. Moreover, Bauer and McDermott (2010) conducts a descriptive and econometric analysis of gold on 30 years of data ranging from 1979 to 2009. Their conclusion is that gold indeed was both a hedge and a safe haven, supporting a correlation close to zero and even slightly negative with the S&P500. The VIX index represents the 30-day forward-looking volatility, and as such, it is a natural for it to be negatively correlated with the other variables except for gold.

The daily correlation proposes beneficial opportunities for prediction purposes. However, as we aim to forecast returns for one day ahead, the task of forecasting becomes substantially more challenging. The last column of the table represents the

correlation between the daily return of the S&P500 with the one-day lagged returns of the other variables. The correlations are now consequently smaller than what it is for daily log returns on the same day. This implies explicitly that the task of correctly predicting the movement of the S&P500 index will be more challenging to perform.

6. Data Analysis

6.1 Linear Regression Model

To adequately justify the power of an SVM technique, we have implemented the method of linear regression to compare the prediction accuracy with that of an SVM model. Linear regression is a parametric regression technique where a response variable will be predicted from a fixed formula given in terms of predictor variables. The fixed coefficients will be estimated to minimize the prediction error. This technique is named Ordinary Least Square (OLS). We have looked at the explained variation and the prediction accuracy captured by the regression model, to see how relevant linear models are at explaining changes in the returns of the S&P500 index. Furthermore, the findings will help to understand if the model is dwarfed by the nonlinearity of financial variables and how the variables are interconnected.

In addition to the stated input variables, we will also add lags of the log returns for the S&P500. The past level of prices for the S&P500 can further increase prediction accuracy by accounting for the correlation of the returns at different lags. Consequently, analyzing the autocorrelation¹⁹ for time series of log returns for the S&P500 has been undertaken. The corresponding autocorrelation value between the lags is extracted from a univariate time series by measuring the correlation between y_t and y_{t+k} where y_t is the log return of the S&P500 while k corresponds to lag $k = 0, \dots, K$. Hence, the autocorrelation for lag k will be $\rho_k = \frac{c_k}{c_0}$ where c_0 is the sample variance of the S&P return series.

¹⁹ See Box, Jenkins and Reinsel (2015) for autocorrelation using time series analysis.

Below is the illustration of the autocorrelation between the returns of the S&P500 at different lags. Notice that we have not included the autocorrelation at lag zero:

Figure 4 - Autocorrelation of the Log Returns for S&P500 at Various Lags

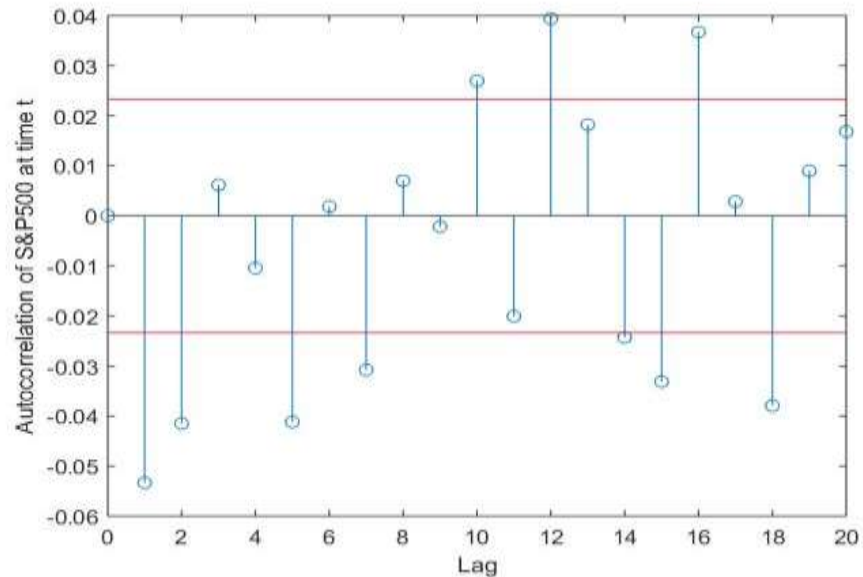


Fig 4. Autocorrelation between the daily log returns of the S&P500 with its lagged values. The figure shows only autocorrelation for lagged values up to the previous twenty days. Notice that the autocorrelation at lag 0 has been discarded.

The upper and the lower autocorrelation confidence bounds show that several of the lagged values are significantly correlated with the current log returns of the S&P500 at time t . However, the autocorrelation for most lags are small and confirms the hypothesis that the prediction of stock prices on past data is a demanding task. Consequently, we will only include the most significant lags corresponding to lag 1,2,5 and 12 when conducting forecasts on the log returns for the S&P500.

From the proposed input variables and the lagged log returns of the S&P500, we will apply the following linear regression model:

$$\begin{aligned}
 r_{S\&P500,t} = & \alpha + \beta_1 r_{S\&P500,t-1} + \beta_2 r_{S\&P500,t-2} + \beta_3 r_{S\&P500,t-5} \\
 & + \beta_4 r_{S\&P500,t-12} + \beta_5 r_{DJIA,t-1} + \beta_6 r_{ESTX50,t-1} + \beta_7 r_{ENX100,t-1} \\
 & + \beta_8 r_{FTSE100,t-1} + \beta_9 r_{HSI,t} + \beta_{10} r_{IXIC,t-1} + \beta_{11} r_{RUT,t-1} \\
 & + \beta_{12} r_{SSE,t} + \beta_{13} r_{VIX,t-1} + \beta_{14} r_{WTI,t-1} + \beta_{15} r_{TNX,t-1} \\
 & + \beta_{16} r_{XAU,t-1} + \varepsilon_t
 \end{aligned}$$

Where $r_{S\&P500,t}$ is the response variable of the regression, and the daily log return estimate of the S&P500 at time t . $r_{S\&P500,t-k}$ corresponds to the log return of the S&P500 at lag $k=1,2,5,12$. This concludes a total of 16 explanatory variables when excluding the constant term. The input variables are all calculated as first log differences of their respective prices and rates, as well as regressed on the response variables. Notice that both the SSE and the HSI returns are reported at time t due to its closing hours which are prior to the opening of the US stock market. The Beta coefficient β , is the constant beta estimate for each of the respective variables, while ε_t is the error term at time t . The error term captures the residual value through the sum of deviations between the predicted log return of the S&P500 and its actual value. The constant α estimate represents the intercept of the regression line.

It is a possibility that some of the indices are not significant in explaining the variation of the S&P500 due to their intraday trading range between the stock exchanges. For instance, FTSE100 will be affecting the US markets the same day, as the majority of its trading hours are completed before the initiation of the trading day for the S&P500. Hence, the preceding return from the previous day will potentially not explain as much of the variation as the intraday trading returns. Vandewalle, Boveroux, and Brisbois (2000) have found evidence that there is a domino effect in which changes of one stock market index influences the other ones, based on their opening hours. However, we have solely regressed the explanatory variables with its previous day closing log returns to stay consistent. Since both the VIX, Treasury 10-year yield and the WTI Crude Oil have different opening hours and closing hours than stock indices, regressing the log returns based on different time horizons could

potentially distort the validity of the model. Infeasible statistical results, such as those related to look-ahead bias²⁰, can produce prediction results which are ambiguous if we utilize log return observations with overlapping time horizons. Still, the Asian stock market indices are closed prior to the US stock market, and the current day return will be a more accurate predictor to utilize for this objective.

As described earlier, the total sample is split between training and testing. The first period is used as input for training the linear regression model, while the out-of-sample period is employed for prediction results. The statistics of the training data are shown below:

²⁰ Look-ahead bias is related to the usage of fundamental information that would not have been publicly available during the time where the analysis is conducted.

Table 2 - Linear Regression of Daily S&P500 Log Returns for Data Between 22/01/1990 – 06/05/2013

	Coefficients	t-statistics
<i>α</i>	0,00	1,73
<i>DJIA</i>	0.02	0,35
<i>ESTX50</i>	0,05	2,13*
<i>ENX100</i>	-0,07	-2,67**
<i>FTSE100</i>	-0,01	-0,22
<i>HSI</i>	0,16	16,53**
<i>IXIC</i>	0,03	1,16
<i>RUT</i>	-0,00	-0,00
<i>SSE</i>	-0,01	-1,10
<i>VIX</i>	-0,00	-0,28
<i>WTI</i>	-0,00	-0,65
<i>XAU</i>	-0,03	-2,20*
<i>TNX</i>	-0,00	-0,13
<i>Lag12</i>	0,04	3,04**
<i>Lag5</i>	-0,04	-2,72**
<i>Lag2</i>	-0,05	-3,85**
<i>Lag1</i>	-0,20	-2,97**
<hr/>		
R-squared: 0,0576	*Significant at 5% level	**Significant at 1% level
<hr/>		
RMSE: 0,0113	MSE: 0,00	MAE: 0,0078
<hr/>		

The table reports the linear regression statistics for the log returns of the respective input variables. Note that this includes only the training dates which ranges from 22/01/1990 to 06/05/2013.

The R-squared estimate of 0,0576 reflects that the log return estimates of the respective variables explain almost none of the variation in the log returns of the S&P500 index for the next day. This is aligned with the results obtained for the analysis of the correlation, where the log returns of the S&P500 are experiencing a limited interrelationship with the previous day log returns for the respective explanatory variables. Moreover, the challenge of conducting robust predictions on the daily log returns of the S&P500 index is clearly evident.

Interestingly, when regressed for the same day, Hang-Sheng has the most substantial affection on the daily log return of the S&P500. When lagging it by one day, it is not statistically significant anymore. An increase in the stock prices on the Hong Kong index will be an implicit indication of a positive market trend for the US stock market on the same day. Still, the SSE composite does not have the same effect. The European stock index Euronext 100 is also significant at the 1% level and helps to confirm the interconnectivity between the global financial markets. All the lagged log return values of the S&P500 are significant at the 1% level, and the estimate for the previous day has the most considerable influence on the current log return for S&P500. Furthermore, the most recent lags show implicitly that previous positive log returns for the index can potentially yield a negative value for the next day. The log returns of the gold price are also significant at the 5% level. As gold can potentially be a safe haven in challenging times, it is not surprising that it can have an affection on the S&P500 index.

Moreover, we have conducted prediction for the test period to consider the robustness of forecasting the log returns of the S&P500. Based on the data used for training the regression model, we have implemented predictions for the linear regression model for the remaining data observations. The RMSE, MSE, and MAE on the test data report errors of 0,0080, 0,0000, and 0,0056, respectively. Below is a chart of the cumulative forecasted log return predictions for the S&P500 when using linear regression, as well as the actual cumulative log return estimates:

Figure 5 - Daily Cumulative Log Return Estimates of the S&P 500 Between 07/05/2013 – 05/03/2019



Fig 5. Daily prediction of the S&P500 log returns cumulated over the time period 07/05/2013 to 05/03/2019. The red line shows the actual cumulated log return estimates for the S&P500 returns.

As expected, the chart shows the limitations of the linear predictions where the OLS model does not adequately capture the substantial momentum in the actual market movements. The linear model is dwarfed by the aspects of nonlinear relationships in the data and fail to react to short-term spikes and movements in the log return estimates.

Additionally, the linear model is used both for prediction accuracy and through the hit rate. The hit rate accounts for the accuracy of predicting an up or a downward movement of the S&P500 price. We have classified the regression prediction as 1 for a positive log return prediction and subsequently -1 for a negative log return estimate. The hit rate of correctly predicted signs of the log returns for the S&P 500 is 57,40%. This indicates that predictions of the market movements for the S&P500 are somewhat possible to forecast based on the predictor set. Even though the linear aspects of the model struggle to capture the magnitude of the movements, it can be used to make more accurate classification forecasts.

The regression results can be somewhat ambiguous. As both the previous day log return of Dow Jones and Nasdaq are not significant at explaining any of variation of the current log return of the S&P500, it can be questioned how robust the linear regression model is. Furthermore, the lagged log returns of the S&P500 index were significant for the regression, and hence, it can potentially be a sign of multicollinearity, even though it is not apparent. Multicollinearity is an indication of a linear relationship between certain variables which can distort the result of the regression. A solution to this problem is to implement PCA to compose the variation in the data by reducing the number of variables.

6.2 Principal Component Analysis

Implementation of Principle Component Analysis can be advantageous to avoid the issues related to highly correlated variables, as already discussed. PCA is a dimension reduction technique that combines the input variables of the data. It helps us to reduce the total number of components by extracting the variables into a reduced data set, where the variables are orthogonal. For a full description of PCA with model specification, see **Appendix 4**. The percentage of variance explained for the log return estimates of the S&P500 can be fully extracted through a smaller number of components than what was originally used. The result is plotted below:

Figure 6 - Percentage Variance of Log Returns for S&P500 Explained by the Orthogonal Components

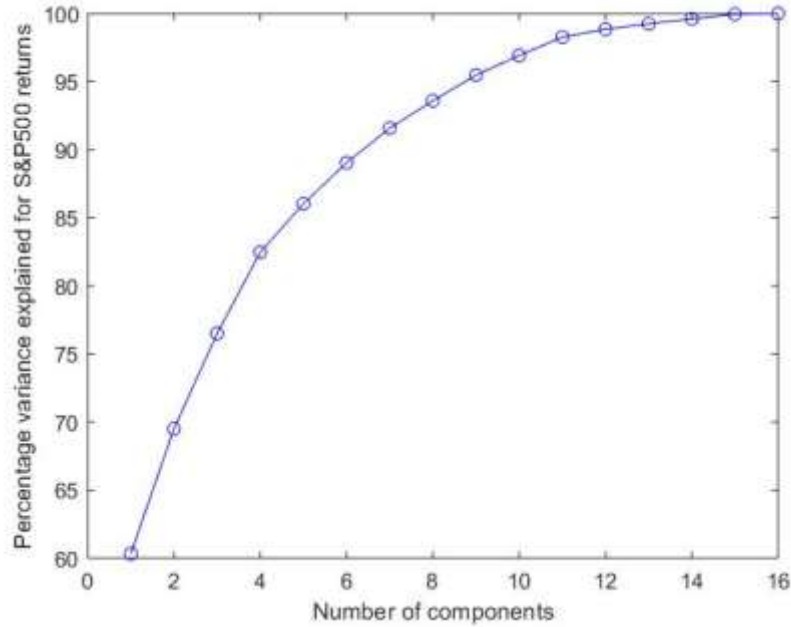


Fig 6. The illustration shows the cumulative percentage variance explained for the S&P500 log returns as the number of orthogonal variables are added. The data set included ranges from 22/01/1990 to 06/05/2013 and consists of all the explanatory variables including the lagged log returns of the S&P500.

As can be seen, 91,58% of the variance in S&P500 log returns can be adequately explained by only seven of the components when we utilize all the predictors. That implies a reduction of nine variables to avoid overfitting and occurrence of potential multicollinearity. Based on the first seven variables obtained through the PCA dimension reduction, we can now regress the new components on the daily log returns of the S&P500:

$$r_{S\&P500,t} = \partial + L_1p_{1,t-1} + L_2p_{2,t-1} + L_3p_{3,t-1} + L_4p_{4,t-1} + L_5p_{5,t-1} \\ + L_6p_{6,t-1} + L_7p_{7,t-1} + u_t$$

Where L is the new regression coefficient that is extracted from the OLS regression. p_{1t-1} is the new representation of the first orthogonal principle component observation, which is extracted from the explanatory variables. p is regressed at $t - 1$, while log returns of the S&P500 are estimated at time t . These factors are

extracted in a way that ensures that all of them are uncorrelated to each other. u_t is the error term observed at time t , and ϑ is the intercept of the estimated regression. The regression is conducted on the training set for the same number of observations as for the original regression. The results of the new regression are presented below:

Table 3 - Linear Regression of the Daily S&P500 Log Returns using PCA from 22/01/1990 to 06/05/2013		
	Coefficients	t-statistics
α	0,000	1,79
<i>PCA1</i>	0,01	2,50*
<i>PCA2</i>	-0,02	-2,62**
<i>PCA3</i>	0,02	3,15**
<i>PCA4</i>	-0,00	-0,56
<i>PCA5</i>	0,12	12,61**
<i>PCA6</i>	-0,03	2,60**
<i>PCA7</i>	0,12	10,47**
R-squared: 0,0485	*Significant at 5% level **Significant at 1% level	
RMSE: 0.0114	MSE: 0.00	MAE: 0.0078

The table reports the linear regression statistics from using PCA on the daily log returns of the explanatory variables. Note that this includes only the training dates which ranges from 22/01/1990 to 06/05/2013.

The regression results show that the R-squared is slightly reduced compared to the original regression. This implies that a dimension reduction technique removes some of the explained variations for the log returns of the S&P500. Furthermore, the MAE stays constant while the RMSE has increased from 0,0113 to 0,0114. However, it helps to keep most of the variation composed in a set of fewer predictors. All the independent variables except the 4th component is significant at the 5% level, and

five of the components are further significant at the 1% level. Moreover, we can conclude that the principal components have removed the potential issue of multicollinearity by using an orthogonal transformation and making six of the principle components significant at explaining the variation of the log returns for the S&P500. However, this comes at the cost of marginally reducing the total explained variation of the S&P500 returns, reflected by the R-squared estimate and the increase in RMSE.

When predicting log return estimates for the S&P500 between 07/05/2013 to 05/03/2019, the RMSE, MSE, and the MAE of the test data are 0,0082, 0,0000, and 0,0058, respectively. This is an increase in the total error for both RMSE and the MAE, compared to the original regression model. The Results also suggest that the dimension reduction technique weakens the robustness of regression forecasting for our test data. The prediction accuracy measured through the hit rate has also been estimated for the same period. Total prediction accuracy is 54,53%, which is a significant decrease compared to the original regression model, which returned a hit rate of 57,40%. We can therefore conclude that dimension reduction techniques are beneficial to compose the variation in the data without losing substantial forecasting accuracy for the training set. However, it clearly fails to improve forecasting of the test set through the magnitude of the movements for the log return estimates reflected by the RMSE and MSE scores. This is also apparent for the prediction accuracy of the classification, which has been explicitly reduced. As we have effectively removed the potential bias of multicollinearity through dimension reduction, the benefits of applying a more simplistic model are not completely apparent. As it does not improve the regression results or the prediction accuracy of the test set, there is evidence that the original regression is superior in providing robust classification forecasts of future log returns for the S&P500.

6.3 Drawbacks of linear regression

The drawback of a linear parameter estimation is the bias of relying on a constant parameter $\hat{\beta}$ to explain the variation in the predicted value of future S&P500 log returns, \hat{y} . The S&P500 has proven to experience multiple periods with significantly

higher volatility than what can be extracted and learned by a linear regression model. Such periods are usually tied to unexpected market anomalies or financial distress where the magnitude of the change in price is substantially larger than during regular periods. Hence, a fixed beta estimate will not adequately capture the amplifying effect of unexpected volatility.

The pairwise interaction between the independent variables used in the regression can further explain more of the variation in the S&P500 log returns. For instance, a major event causing a drop in the Asian stock market could potentially trigger an increase in the global uncertainty indirectly through the VIX index. Even though the US markets can be affected by the Asian economy, the indirect effects that can be extracted from these events are potentially not fully reflected through a linear regression model. Additional variables can be argued to capture these nonlinear movements, but as it would produce an immense number of regressors, it could be unfeasible to consider this, even with a PCA technique. Extracting these interactions in a linear regression model comes at the expense of substantial computational cost. Consequently, the introduction of an SVM model where a nonlinear function can be mapped into an infinite high dimensional data space can address this issue more efficiently.

7. Support Vector Machine

Adjusting a formula to create a defined model with a minimum prediction error can be cumbersome. Predicting a response variable from unknown observations in an out-of-sample test can be enhanced by implementing a nonlinear regression model. When data is not linearly separable, a solution can be to add a kernel function to transform the predictors into a linearly separable dataset. Mullainathan and Spiess (2017) highlight the importance of choices made for implementing a supervised learning model. It must be considered what function to apply and how to measure the accuracy of the respective output. We aim to predict both the direction and the accuracy of the movements for the returns of S&P500, and hence, we will account for both aspects in the SVM regression model. The predictors used as explanatory variables are the same as for the original linear regression model, where the data are all calculated as log

returns. The first part will solely evaluate the prediction based on forecasting the sign of the movement and it will be a classification problem. Secondly, we will consider the regression results for the SVM model and discuss if this is a better forecasting model used for predicting the magnitude of the movements compared to a linear regression. The predicted regression results will be captured by the RMSE, MAE, and the MSE by comparing it to the two linear regression models. The prediction of up- and downward movements will be calculated based on the hit rate for the test data.

7.1 Tuning of the SVM Regression for Optimal Predictions

According to Cao and Tay (2001), the choice of hyperparameter values depend on the box constraint C , the gamma for the kernel scale γ , and the tolerance hyperparameter epsilon, ϵ . Their research show that the performance of SVM is insensitive to ϵ , while both C and γ play an essential role for the prediction accuracy. Furthermore, the values must also be set to avoid the issue of overfitting and underfitting the training data. The values of C , γ , and ϵ have been set iteratively for multiple kernel functions. The kernel functions evaluated are the linear kernel, the gaussian kernel and the polynomial kernel. See **Appendix 3** for their model specifications.

The prediction results for the three kernel functions estimated with different values for the hyperparameters are illustrated below. The performance measures of the SVM regression are reported as the RMSE and the hit rate, where RMSE demonstrates the magnitude of the predicted returns against the actual return of the S&P500, while the hit rate accounts for correct predictions of the regression results from up- or down-movements:

Table 4 - Results of Prediction Accuracy for the Daily Log Returns of the S&P500 Using an SVM Regression Model with Different Hyperparameter Values

	<i>Kernel Function:</i>	Linear	Gaussian	Polynomial
$C = 10$				
$\gamma = 10$	RMSE	0.0080	0.0080	0.0080
$\epsilon = 0.01$	Hit Rate	0.5794	0.5685	0.5726
$C = 100$				
$\gamma = 10$	RMSE	0.0080	0.0080	0.0080
$\epsilon = 0.01$	Hit Rate	0.5678	0.5671	0.5692
$C = 100$				
$\gamma = 10$	RMSE	0.0080	0.0080	0.0080
$\epsilon = 0.001$	Hit Rate	0.5753	0.5692	0.5706
$C = 10$				
$\gamma = 10$	RMSE	0.0080	0.0080	0.0080
$\epsilon = 0.001$	Hit Rate	0.5849	0.5842	0.5719
$C = 10$				
$\gamma = 1$	RMSE	0.0080	0.0082	0.0084
$\epsilon = 0.001$	Hit Rate	0.5746	0.5603	0.5576

Prediction results of the SVM regression models for various hyperparameters. The prediction period ranges from 07/05/2013 to 05/03/2019, and the training data uses log return estimates from the input variables. The best performing function results are highlighted with bold font.

As can be seen, the performance measure of RMSE is relatively stable when the parameters are tuned incrementally. This observation is consistent with the results of

Cao and Tay (2001), where they suggest that these specific boundaries for the values of the hyperparameters help to avoid the issues of overfitting the data. They also argue that the Gaussian kernel is the best suited function to apply for financial time series data. This is contradicting with our research, where the linear kernel tends to outperform both the Gaussian- and the Polynomial kernel for the majority of the models.

Simultaneously, the SVM model applying the linear kernel seems on average to consistently outperform the linear regression models when we consider the prediction accuracy. Since SVM is well-known for the accuracy of predicting classification objectives through the hit rate, it can be superior to both the linear regression models. However, the SVM fails to outperform the linear regression model when we account for the magnitude of the predictions through the RMSE (See Table 5 below for a comparison of forecasting results). Also, both the Polynomial kernel and the Gaussian kernel fails to improve the prediction accuracy relative to the linear models. It implies that even after transforming the input predictors into a high nonlinear dimension space, we are still not able to consistently outperform the simple OLS technique by the hit rate. Regardless, SVM can be a more precise forecasting tool when evaluating investment returns over a defined time period. Consequently, we will adopt a real investment strategy by comparing the absolute investment value generated by the output of the predictions for each respective model.

Table 5 - Prediction Comparison of the Best Performing SVM Model to the Linear Regressions

<i>Test Period 07/05/2013 - 05/03/2019</i>			
	<i>Linear Regression</i>	<i>PCA</i>	<i>Linear SVM</i>
RMSE	0,0080	0,0082	0,0080
MAE	0,0056	0,0058	0,0056
MSE	0,0000	0,0000	0,0000
Hit Rate	57,40%	54,53%	58,49%

The linear kernel has the stated hyperparameters $C = 10$, $\gamma = 10$ and $\epsilon = 0.001$. The test period for prediction ranges between 07/05/2013 to 05/03/2019 and forecasts the daily log returns of the S&P500 index.

8. Results

Calculating the total return per unit dollar for each strategy is a better measurement than purely focusing on the prediction accuracy. As the hit rate can suffer from bias related to predicting a market which only moves upward, it can distort the true accuracy when the market is experiencing a positive trend. Moreover, models that forecast frequent changes in up- and down movements can be significantly affected by the incurred transaction costs from trading securities. Consequently, we account for the absolute investment return to analyze the actual impact of predictions for each model.

We have selected two different investment strategies, a long-only- and a long/short strategy, to conclude on the viability of implementing the prediction models for trading. Secondly, we provide results on the long-only portfolio without transaction costs, as well as including the transaction costs for the long/short strategy. We evaluate both strategies by comparing it to a passive buy-and-hold investment as a benchmark. The overall results will determine if we can reject our null hypotheses

where a passive buy-and-hold strategy is superior to the predictions of a nonlinear SVM model.

8.1 Long-only portfolios without transaction costs

The prediction output for the two linear regression models will be equal to what we have previously estimated. The implementation of the optimal SVM regression model is justified based on the iteration of the different hyperparameter values. We have chosen to apply the best performing SVM model which was deemed the linear kernel with the parameters $C = 10$, $\gamma = 10$ and $\epsilon = 0.001$. We will be conducting investment strategies for each of the three estimated models to determine the adequacy of the predictions when applied to trading the underlying S&P500 index. The predictions will only account for the classification of the log return estimates by either forecasting an up- or a downward movement of the underlying index.

For the first investment comparison, it is assumed that an investor either implements a passive buy-and-hold strategy or an active long-only strategy when investing in the S&P500. The investment period corresponds to the daily test period between 07/05/2013 to 05/03/2019. The buy-and-hold strategy is as follows:

$$BH_T = BH_0 \prod_{t=1}^T (1 + r_{S\&P500,t})$$

Where BH_0 is the absolute value of the total investment for an investor at initiation, while $r_{S\&P500,t}$ is the log return of S&P500 at time t . At initiation, $t = 0$, the investor buys an amount equivalent to \$1 of the S&P500 index, and the strategy corresponds to holding the index until time T , which concludes 1467 days. The investment strategy for the three models, Linear Regression, PCA and SVM is based on classification of the regression prediction for the S&P500 log returns. At time t , if the model predicts a positive log return for the S&P500 then:

$$M_{t+1} = M_t(1 + r_{S\&P500,t})$$

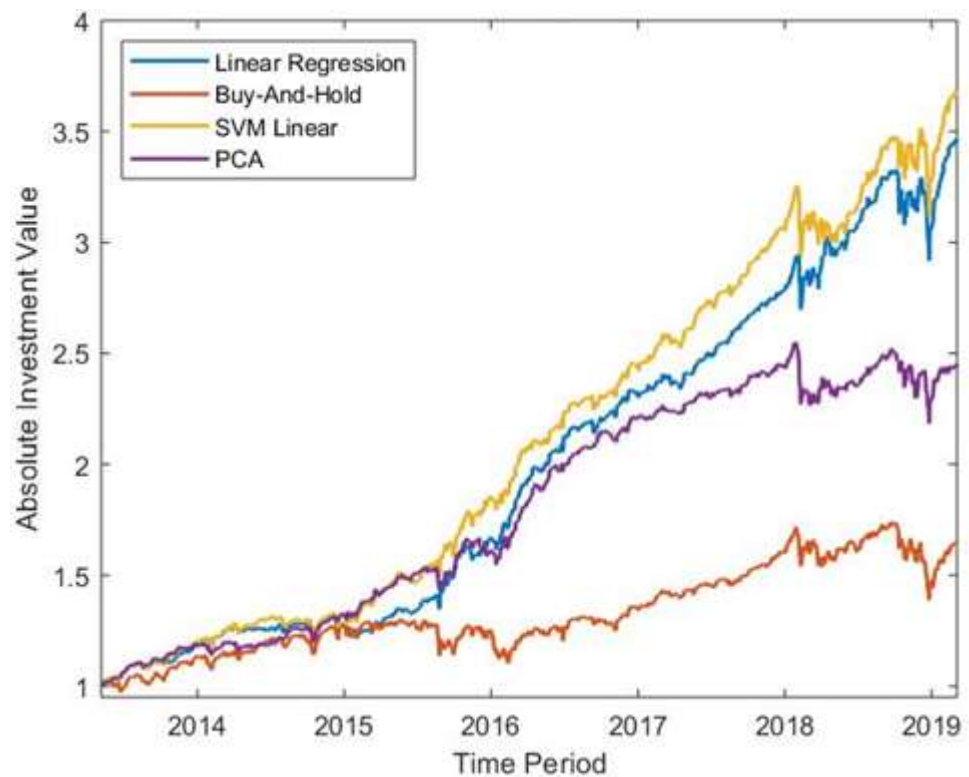
Where M_t is the absolute value of the total investment for the respective model at time t . If the model predicts a negative log return for the S&P500 index at time t , the

model will sell its long position if it holds the index. The absolute value of the total investment will be:

$$M_{t+1} = M_t$$

When the investor is not holding any positions in the S&P500 and the model predicts a negative return, the investor will keep its cash holdings. For this simplistic model, we have assumed that there exist no frictions in the market, and hence, there will not be any transaction costs, bid-ask-spreads²¹, rollover costs²² or inefficiencies when buying or selling the positions. We have also assumed that the investor earns no interest on its cash holdings. The total absolute investment value over the test period is illustrated below:

Figure 7 - Buy-And-Hold vs Model Strategies: 07/05/2013 – 05/03/2019



²¹ Bid (buy) – Ask (sell) spread is the current price difference between what positions the investors have placed in the open market to buy or sell a security. The Ask price is always equal to or higher than the bid price

²² This is a fee the market maker charges for investors who desire to keep their position in derivatives overnight.

Fig 7. The absolute investment value cumulated over the time period between 07/05/2013 – 05/03/2019 for each of the respective forecasting strategies when trading the S&P500 index, including the buy-and-hold strategy. No transaction costs or market frictions are assumed.

All three models are significantly outperforming the buy-and-hold strategy when excluding all costs and frictions in the market. From its prediction accuracy, it can be concluded that the prediction models avoid adverse outcomes over the test period, to some extent. The total absolute investment value for the buy-and-hold strategy, the linear regression model, the PCA model and the SVM are \$1,64, \$3,45, \$2,44 and \$3,67, respectively at the end of the period. The results are an explicit indication that all the prediction models can consistently outperform a passive investor holding the S&P500 over the specific time period, when we relax the additional consequences of financial trading. The SVM model has also proven to be superior relative to the other prediction models by generating excessive returns compared to the linear regression models over almost the entire test period.

However, the absolute investment value will be significantly affected when accounting for transaction costs. Over the stated test period of 1467 days, the linear regression model, the PCA model and the SVM model undertake a total of 726 trades, 793 trades and 616 trades, respectively. Clearly, the SVM model performs substantially fewer trades than the two other models. This strengthens the hypothesis that SVM can be an even more robust forecasting model after deducting for transaction costs over the test period.

8.2 Long-short portfolios with transaction costs

For the long/short portfolio, we have compared the buy-and-hold strategy to the prediction results for the three forecasted models. The investment strategy of each of the models is again based on the classification of the prediction for the daily S&P500 log returns. At time t , if the model predicts a positive log return for S&P500 then:

$$M_{t+1} = M_t(1 + r_{S\&P500,t})$$

Where the calculations are identical as for the long-only strategy. Whenever the model predicts a negative log return for the S&P500 at time t , the model will sell its long position, if it holds the index, and go short the S&P500. Since we are using daily closing prices in our data, we have assumed for all portfolios that the market price will be identical when exiting the current position and when entering the new position. The absolute value of the total investment after selling the S&P500 index will be:

$$M_{t+1} = M_t \left(1 + (-r_{S\&P500,t}) \right)$$

Consequently, a correct prediction when the index yields a negative log return will correspond to an increase in the absolute value of the total investment. This implies that we will always be fully invested in the market. The table below shows the mean return, the volatility of the returns and the Sharpe Ratio²³ of each of the respective investment strategies, including the buy-and-hold strategy:

	<i>Buy and Hold</i>	<i>Lin Reg</i>	<i>PCA Reg</i>	<i>SVM Reg</i>
Mean	0,04%	0,14%	0,09%	0,14%
Standard Dev	0,82%	0,81%	0,82%	0,81%
Sharpe Ratio	0,0452	0,1674	0,1081	0,1789

The mean value represents the average daily log return obtained over the period between 07/05/2013 to 05/03/2019 by investing in the S&P500 by either staying long or short, depending on the prediction for each respective model. The standard deviation is the variation around the mean value, while the Sharpe ratio measures the mean value of a variable, divided by its standard deviation.

The linear SVM model seems to incur the largest Sharpe ratio, closely followed by the original linear regression model. All the three prediction models seem to be outperforming the buy-and-hold strategy. The findings imply that the forecasting information provides the investor with a strategy that is more profitable than purely

²³ To stay consistent, we have assumed a risk-free rate of zero. The calculations will then simply be the model mean divided by its standard deviation.

holding the underlying index. Still, we need to account for transaction costs before concluding which strategy that is the most viable to apply.

We have incorporated a transaction cost of 0,1% for each trade undertaken. The CME Group (2016) has conducted research for active individual traders to compare Exchange Traded Fund (ETF) costs for S&P500. Assuming we will trade an ETF of the underlying S&P500 index, it can be expected to be incurred a total cost of 5bps-8bps for each long transaction. For short transactions, additional costs related to borrowing stocks and possible margin requirements can make the total transaction cost to be in the range of 12bps-15bps. Therefore, we will assume an average transaction cost per trade of 10bps to be a plausible estimate. This assumption is consistent with earlier research on the topic of predicting the S&P500. For instance, Gestel, et al., (2001) estimated a 10bps transaction cost. More recent research conducted by Fischer and Krauss (2018) imposed a transaction cost of 5bps per half-turn, but this was merely for long-only investments. Hence, 10bps can implicitly be a reasonable estimate when we include both long- and short sales over the period between 2013 to 2019. Furthermore, when the model changes its holding from long to short- or the other way around, transaction costs will be incurred twice. We assume that when liquidating the current position, a new position will be undertaken simultaneously. The total absolute investment value cumulated over the out-of-sample period for each of the respective models are provided below:

Figure 8 - Model Prediction for Long/Short Strategies with Transaction Costs

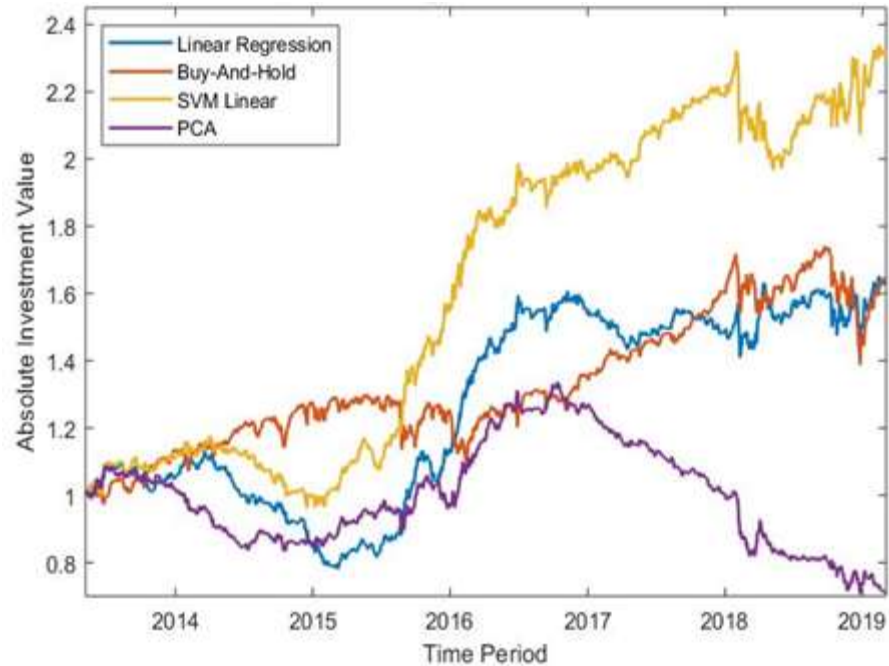


Fig 8. The absolute investment value cumulated over the period between 07/05/2013 – 05/03/2019 for each of the respective forecasting strategies for trading the S&P500 ETF, including the buy-and-hold strategy. Notice that a transaction cost of 0.1% is included for each trade undertaken.

The performance of each model has been significantly affected by the costs incurred from trading. The absolute investment value of the buy-and-hold strategy, the linear regression, the PCA and the SVM are \$1,64, \$1,63, \$0,71 and \$2,32, respectively. The results are an unambiguous identification of the successful prediction accuracy of the SVM. SVM avoids excessive trading, which can potentially erode the total investment value over time, due to trading costs. Compared to the long-only strategy, where all models outperform the buy-and-hold strategy substantially, the successive trading signals from the two linear regression models will prevent the gains from compounding to the same extent. Therefore, both the linear regression models are now inefficient compared to the buy-and-hold strategy. Moreover, the PCA model

shows explicit limitations relative to the original linear regression, indicating that the dimension reduction technique has led to a sub-optimal investment strategy.

8.3 Robustness of the prediction models and validity of the results

As we have already elaborated, the testing period can have clear prediction limitations as the cumulative log returns of S&P500 have increased over most of the duration between 2013 to 2019. Hence, a model that mainly predicts positive log returns, will on average obtain attractive investment results. See **Appendix 9** for an illustration of the cumulative log return predictions for all the three models. Consequently, we will justify the validity of the models by applying the strategies in a completely different environment.

Firstly, we include another SVM model for the purpose of comparison. This model implements a Gaussian kernel where each of the observations for the predictors in the training data set has been scaled by its variable mean and its standard deviation. Preprocessing of the data can enhance the forecasting accuracy by making predictions more feasible, without incurring look-ahead bias since we only apply this on the training data. The chart below illustrates the cumulative return forecasts of the SVM regression for a Gaussian and a linear kernel. Both models have the same hyperparameters as the former linear SVM regression that we have applied for predictions. These return estimates are compared against the actual cumulative log returns for the S&P500:

Figure 9 - Predicted Cumulative Log Returns of the S&P500 Between 07/05/2013 – 05/03/2019

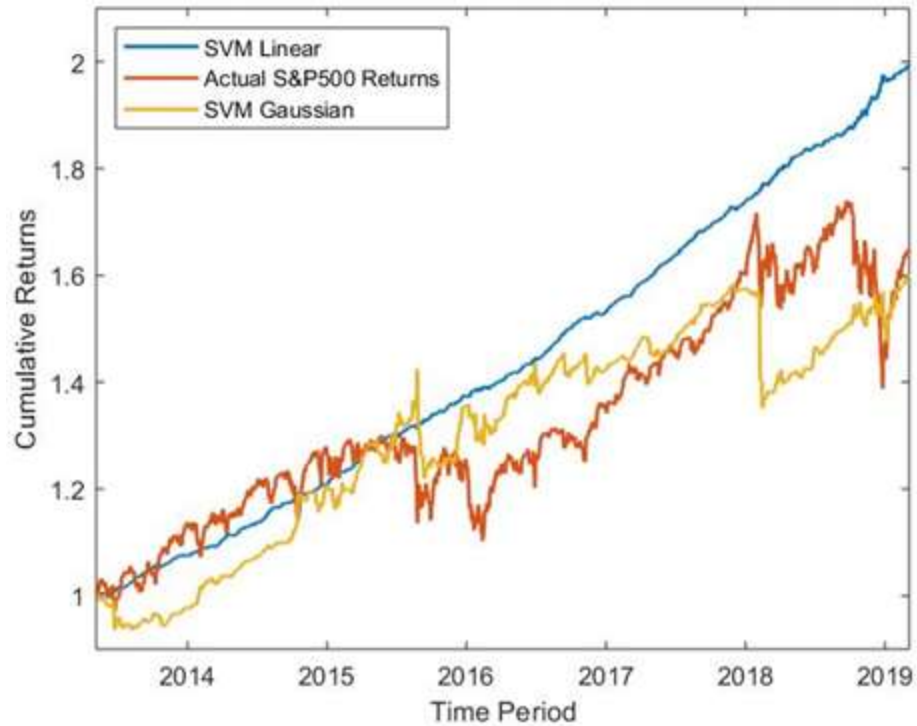


Fig 9. Both the Gaussian kernel and the Linear kernel have the stated hyperparameters $C = 10$, $\gamma = 10$ and $\epsilon = 0.001$ with preprocessed log return estimates and raw log return estimates, respectively. The preprocessed data has been normalized by configuring all the training observations with their variable mean and standard deviation. The benchmark is the cumulative log returns of the actual S&P500 index between 07/05/2013 - 05/03/2019.

As can be observed, the Gaussian SVM seems to implicitly capture the movements in the actual cumulative log returns more adequately. Large volatility spikes in the S&P500 index are only predicted by the Gaussian SVM model, however at an amplifying rate. The linear SVM model shows limitations by predicting the index to move upwards consistently. Moreover, the RMSE of the Gaussian SVM model is 0,0099 while for the Linear SVM model, the RMSE is only 0,0080. The hit rate of the Gaussian model is 53,31%, while the Linear model is significantly higher at 58,49%. This evidence concludes that the issues of daily financial data are apparent from the substantial noise incurred by market indices. A model that is consistently predicting the index to retain positive log returns can on average predict accurately when the market is trending upwards. Still, there could potentially be an indication that the

Gaussian model can be implemented for markets that experience significant volatility. Even though the model fails to imitate the S&P500 log returns accurately, it can potentially be a better forecasting tool to predict magnifying changes over a short-term horizon.

8.4 Testing the investment strategies during the financial crisis

To consider the validity and robustness of our SVM model, we have accounted for a different time period to justify how viable the investment strategy would be when the financial environment is changing. We have implemented the same model specifications for the data set only ranging between 02/01/1990 to 31/12/2010, which constitute a total of 5281 observations. The training period is composed of 80% of the data and ranges from 22/01/1990 to 20/10/2006. It yields 4225 training observations and 1056 test observations. The prediction period will be from 23/10/2006 until 31/12/2010. The period allows us to capture the anomalies experienced through the Great Recession to conclude the robustness of the model when unexpected market events occur. The table below presents the regression results for the different models through RMSE, MAE and MSE. The hit rate denotes the prediction accuracy of the models. Notice that we have included the SVM model with the Gaussian kernel to inspect if it can be better at capturing unexpected spikes in the S&P500 index:

Table 7 - Prediction Comparison of the Two Test Data Sets for the Log Return Estimates of the S&P500

<i>Test Period 07/05/2013 - 05/03/2019</i>				
	<i>Linear Regression</i>	<i>PCA</i>	<i>SVM</i>	<i>SVM Gaussian</i>
RMSE	0,0080	0,0082	0,0080	0,0099
MAE	0,0056	0,0058	0,0056	0,0064
MSE	0,0000	0,0000	0,0000	0,0000
Hit Rate	57,40%	54,53%	58,49%	53,31%
<i>Test Period 23/10/2006 - 31/12/2010</i>				
	<i>Linear Regression</i>	<i>PCA</i>	<i>SVM Linear</i>	<i>SVM Gaussian</i>
RMSE	0,0163	0,0167	0,0165	0,0257
MAE	0,0107	0,0108	0,0107	0,0153
MSE	0,0003	0,0003	0,0003	0,0007
Hit Rate	56,25%	53,41%	56,16%	54,17%

Prediction comparison of the forecasting models for two different periods. The first test period ranges from 07/05/2013 to 05/03/2019, while the second test period ranges from 23/10/2006 to 31/12/2010.

It is not surprising that the forecasting results have been aggravated for the test period between 2006 to 2010. As the frictions of the financial markets rose, unexpected events were not anticipated by the market participants. Even though the predicted log returns fail to capture the development in the S&P500, they are still consistent when changing between the linear regression model and the linear SVM. For the Gaussian SVM, regression predictions are substantially less accurate than for the other three models, although the hit rate has increased from the first to the second test period. As mentioned, the Gaussian model tends to capture the unexpected spikes in the price of

the S&P500, but at an amplifying rate. This is not the case during the prediction period of the Great Recession. The Gaussian model fails to predict the magnifying changes in the returns, and the regression errors are considerably larger.

The hit rate of each of the respective models indicates that classification forecasting is still possibly a viable investment strategy. Even though all models are marginally less consistent than for the first prediction period, we can make forecasting decisions that potentially can generate excess returns relative to a buy-and-hold investor. Moreover, the hit rate of the linear regression model is now higher than for the linear SVM model. The results can be an indication that the SVM model fails to be consistently superior to other statistical techniques when changing the data set and the forecasting period. Furthermore, we have accounted for the classification of predicting the daily log returns of the S&P500 over the test period. The investment results for each of the models are depicted in the table below:

	<i>Buy and Hold</i>	<i>Lin Reg</i>	<i>PCA Reg</i>	<i>Lin SVM</i>	<i>Gaus SVM</i>
Mean	-0,01%	0,27%	0,10%	0,24%	0,09%
Std	1,69%	1,67%	1,69%	1,68%	1,69%
Sharpe	-0,0047	0,1613	0,0614	0,1447	0,0549

The investment results for daily log returns obtained over the period between 23/10/2006 to 31/12/2010 from investing in the S&P500 by either staying long or short, depending on the prediction for each respective model.

The superiority of the linear SVM model for the first test period has now been dwarfed by the excess returns generated from the linear regression model. The linear regression model is simultaneously experiencing the least volatility in the obtained daily log returns, resulting in the most significant Sharpe ratio. Furthermore, the Gaussian SVM is not able to produce better results even though the financial market is struck by intensifying volatility. Notice that we will still have to consider the transaction costs before we can conclude on the profitability of the strategies.

By implementing the classification results of the models, we can now assess the actual profitability for investing in the S&P500 index by accounting for transaction costs. Again, we have considered the long/short strategy for each respective model over the new prediction period where the initial investment is \$1 dollar. The figure below illustrates the cumulated absolute investment value over the period from 23/10/2006 to 31/12/2010 for the various models. Both the Gaussian- and the linear SVM models are included for comparison. The linear SVM model has the same hyperparameters as estimated for the full test set, while the Gaussian model has the same hyperparameters, but with standardized data:

Figure 10 - Long/Short Strategy with Transaction Costs Between 23/10/2006 – 31/12/2010

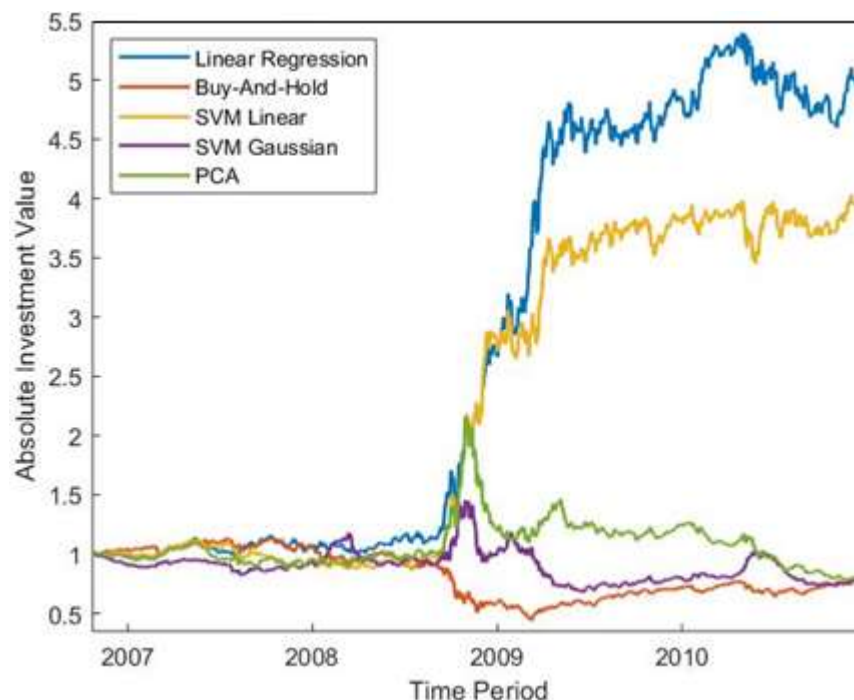


Fig 10. The absolute investment value cumulated over the period between 23/10/2006 – 31/12/2010 for each of the respective forecasting strategies by trading the S&P500 ETF, including the Gaussian SVM model. Notice that a transaction cost of 0.1% is added for each trade undertaken.

The absolute investment value at the end of the period for the buy-and-hold strategy, linear regression model, PCA, linear SVM and Gaussian SVM are \$0,79, \$4,97, \$0.81, \$3,96 and \$0,77, respectively. Clearly, only the original linear regression model and the linear SVM outperform the buy-and-hold strategy substantially over the period. These two models profit on its predictions by shorting the S&P500 index

when the it falls considerably in the period between 2008 to 2009. This evidence proves the robustness of the forecasting ability for both strategies. However, the choice of implementing the most profitable forecasting model is ambiguous. The linear regression model is significantly generating excess returns compared to the linear SVM for the second test period. The outperformance of the linear SVM for the full data set is now the least viable investment opportunity among the two strategies. Furthermore, the application of a Gaussian SVM, which is better at predicting urgent spikes and volatility of the index has apparent drawbacks when forecasting during the financial crisis. Even though all models can implicitly avoid some of the negative periods, the Gaussian SVM and the PCA models have limitations of taking advantage of the negative log returns.

Moreover, the transaction cost per trade of 0.1% can be an unlikely estimate due to the illiquidity and the issues of finding a counterparty to trade with, during the financial crisis. Additionally, shorting was even more problematic, resulting in exceptionally high costs and a lack of available securities in the open market. The linear regression, PCA, Linear SVM and Gaussian SVM conducts a total of 546, 580, 516 and 545 trades, respectively. This reflects a substantial number of trades necessary to undertake during the total of 1056 trading days, to profit from the strategies. Once again, the strength of the linear SVM is apparent as it consistently performs fewer transactions than the other models. Hence, the linear SVM would be more beneficial at times when markets are experiencing frictions, and trading will be more challenging to undertake.

9. Conclusion and further analysis

Our results have shown that the implementation of a supervised SVM regression model that forecasts log returns of the S&P500 can outperform a passive buy-and-hold strategy, both by generating excess returns and reduced volatility. SVM has proven superior to the linear regression models when we conduct investment strategies for the out-of-sample period between 2013 to 2019. However, when changing the prediction period to the Great Recession, SVM is outperformed by the

linear regression model. Such a result is in line with previous research concluding that the choice of input data and the size of the data set, is at least equally important as the choice of a model technique. This supports the fact that no algorithm is consistently superior to others. Regardless, the buy-and-hold strategy is outperformed by both prediction models for our two selected test periods. As such, we have proven that implementing a trading strategy based on the predictions of an SVM regression model could potentially be a more viable option compared to a passive approach. However, conducting additional research for multiple time periods should be examined before concluding that implementation of this active trading strategy can generate excess returns and reduced volatility, consistently over time.

While the SVM model is a feasible investment option for our findings, unexpected anomalies and other adverse events can distort the investment results for the SVM model. Without additional research, our model is not necessarily superior to the passive buy-and-hold strategy over multiple time periods but can potentially be a supplement for investors that want to include an active strategy to their portfolio. We suggest that adding a more diverse selection of variables with predictive abilities can help to explain additional variability for the log returns of the S&P500. This can further be advantageous for an SVM model and enhance the prediction accuracy by extracting nonlinear relationships between variables. Such factors can be associated with currency pairs, macroeconomic factors, and individual stocks.

While both the SVM- and the linear regression model show promising prediction results, the ambiguous outcome of deciding on the best-qualified model should be further analyzed by accounting for the total number of trades undertaken. We have proven that when analyzing the potential of a trading strategy, the frequency of trading must be accounted for in addition to prediction accuracy. We leave the discussion of accurately measuring transaction costs, market frictions, and correction for bid-ask-spreads, to further research. If the costs and issues of trading securities are apparent, SVM can then explicitly be the preferred model based on our findings, outperforming the linear regression strategy by conducting substantially fewer trades on average. This evidence should be further discussed and analyzed by conducting predictions for multiple periods to adequately conclude if the SVM is consistently performing better than a buy-and-hold strategy, as well as a linear regression model.

Appendix

Appendix 1:

Table 9 - Input Variables used for forecasting

No.	Acronym	Full Name	Factor	Frequency	Characteristics
1	SPX	S&P 500	Price	Daily	Stock Index
2	DJIA	Dow Jones Industrial Average	Price	Daily	Stock Index
3	ESTX50	Euro Stoxx 50	Price	Daily	Stock Index
4	ENX100	EuroNext 100	Price	Daily	Stock Index
5	FTSE100	Financial Times Stock	Price	Daily	Stock Index
6	HSI	Hang Seng Index	Price	Daily	Stock Index
7	IXIC	Nasdaq Composite	Price	Daily	Stock Index
8	RUT	Russell 2000	Price	Daily	Stock Index
9	SSE	Shanghai Stock Exchange	Price	Daily	Stock Index
10	VIX	CBOE Volatility Index	Price	Daily	Volatility
11	WTI	WTI Crude Oil	Price	Daily	Commodity
12	XAU	Gold Spot	Price	Daily	Commodity
13	TNX	Treasury Yld Index- 10Yrs	Yield	Daily	Interest Rate

Appendix 2:

Data Collection

The data is gathered from multiple open source sites online and is accessible for anyone who desires the same or similar data inputs. Most of the stock price data and the commodity data are collected from Yahoo finance and Bloomberg. The stock price observations for FTSE100 is gathered from the home page of London Stock Exchange, while the data regarding the treasury data is all collected from the homepage of the US Department of the Treasury.

Each country has various holidays where the domestic stock indices are closed. Darrat and Zhong (2000) express the possible problems of daily financial data which can be affected by biases due to bid-ask spreads and holidays when certain markets are closed. However, we will assume that the previous day closing price for each of the financial observation is used whenever the market is closed due to holidays. This is consistent for all the variables. Moreover, both the stock indices of Euronext100 and the SSE Composite were not present at the beginning of our sample period. We therefore assume that the opening price of the stock index at initiation will be constant until trading is undertaken. Additionally, we exclude those days where the S&P 500 index was closed, as this is the index we aim to forecast.

Since we have a timeframe of almost 30 years there is a high probability that significant changes within the indices have occurred. To keep the consistency throughout our dataset we collect the adjusted closing price for all our variables. The adjusted closing price takes into account factors such as dividends, stock splits and new stock offerings and makes it possible to compare newer data observations directly with older observations.

For the S&P 500 we have 7.350 observations, and most of the selected variables have the same amount of observations. Still, there are two variables that has less observations than the other variables.

1. The SSE Composite were traded from the initiation period of December 19,1990. However, the index was officially launched on February 21, 1992 and we therefore assume a price equal to its trading price at the official launch prior to this date. This is backed up by the reason that prices are moving irrational between these two periods, causing inconsistent data observations. In our dataset we have 7105 observations, and the observations prior to the official launch have been kept constant at the first official price of 1266,49.
2. Euronext 100 was established in 2000 with the merger of three European exchanges and as such, we only have 4822 observations available. Observations prior to year 2000 will be kept constant at the price of 1000, which was the initiation price.

Appendix 3:

For SVM, a kernel can be referred to as a method of using a linear classifier, even though we have a nonlinear data set. Since financial data is nonlinear in nature, kernel proposes the opportunity to transform the input variables into a data set that can be more easily applied for a statistical technique. Notice that for $\theta(x_i)$, vector i for the observations of the explanatory variables are transformed into an unknown high dimensional data space. This is possible by applying the kernel trick²⁴. The dual problem has the properties $\theta(x_i)^T \theta(x_j)$ where a 2nd-degree polynomial

²⁴From the Mercer's theorem, there exists a function θ that maps two variables, a and b , into another space, such that $K(a, b) = \theta(a)^T \theta(b)$ Therefore, you can apply a kernel K since you know that θ exists, even without having knowledge of θ . See a full description of the Mercer's theorem from Minh, Niyogi and Yao (2006).

transformation can replace the dot product of the input vector i and j . Hence, avoiding the trouble of transforming the variables are making the algorithm significantly more computational efficient from applying the kernel trick. The kernel trick will help to express the inner product of the predictors in a higher dimensional data space. The chosen kernel computes the dot product of $\theta(x_i)^T \theta(x_j)$ only based on their original vectors.

We can rephrase the original vectors to be:

$$K(a, b) = \theta(x_i)^T \theta(x_j)$$

The introduction of the dual problem, which is derived in Appendix 4, proposes certain benefits which help us to avoid the computation of transforming the data into a higher dimensionality space. We have accounted for three kernel functions in this paper. The kernel functions ²⁵ evaluated are the following:

$$\text{Linear Kernel } K_L(a, b) = (x_i)^T x_j \quad (4)$$

$$\text{Polynomial Kernel } K_P(a, b) = (1 + (x_i)^T x_j)^d \quad (5)$$

$$\text{Gaussian RBF Kernel } K_G(a, b) = \exp(-\|x_i - x_j\|^2) \quad (6)$$

Where K_P represents a polynomial kernel of order d , where 3 is set at default. We will evaluate each of these functions for comparison to conclude which is the best suited to predict financial returns for the S&P500.

Appendix 4:

To fully understand the statistical procedure of conducting a linear SVM technique, we must implement optimization of certain hyperparameters to obtain better forecasting accuracy. By finding the value of b and w that maximizes the margin g , we can generate prediction estimates that are optimal for the proposed decision function (Equation 1). Simultaneously, the slack variable ξ must be minimized to reduce the total margin violations of prediction errors. We must introduce another

²⁵ For a discussion on the kernel functions, see Amari and Wu (1999).

hyperparameter, C , which is described as the box constraint. It helps with the regularization of the model by defining the trade-off between the objective of minimizing $\|w\|$ and ξ simultaneously.

We are now ready to denote the constrained optimization problem in the following equation:

$$\min_{w,b,\xi} \quad \frac{1}{2}w^T w + C \sum_{i=1}^N \xi_i \quad (7)$$

s.t

$$t_i(w^T \theta(x_i) + b) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0 \quad \forall i$$

This is the primal formalization for inseparable variables, such as for financial time series data. We are minimizing the feature weight $\|w\|$ such that we can maximize the margin g from each side of the decision boundary. Concurrently, we are minimizing the sum of the slack variable ξ to reduce the total amount of misclassifications for the estimated model, while still allowing some instances of misclassification. The slack variable is crucial to avoid the common issue of overfitting or underfitting the data for financial time series analysis. When the model classifies the data points correctly, $\xi = 0$, while data points that are predicted incorrectly will have $\xi > 0$. This implies that we conduct a convex quadratic optimization. The objective function is convex quadratic in w , while the constraints are linear in w and ξ .

Moving from the primal formalization above to the soft-margin formulation, requires the implementation of the Lagrange multipliers to minimize the constraint optimization problem. By deriving the primal objective function, we can move to the SVM dual problem. This helps us to transform a constrained optimization objective into a new unconstrained one. By introducing two Lagrange multipliers, α and μ , we can find the stationary points of the Lagrangian, such that we will obtain the solution

to the constrained optimization problem. Since we include an inequality constraint, α and μ will be of the form Karush-Kuhn-Tucker (KKT) where $\alpha_i, \mu_i \geq 0 \forall i$. From equation (7), we can restate the minimization problem into the Generalized Lagrangian for the soft-margin formulation:

$$L(w, b, \alpha, \mu) = \frac{1}{2} w^T w + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (t_i (w^T \theta(x_i) + b) - (1 - \xi_i)) - \sum_{i=1}^N \mu_i \xi_i \quad (8)$$

s.t

$$\alpha_i \geq 0 \forall i$$

$$t_i (w^T \theta(x_i) + b) \geq 1 \forall i$$

We can now compute the partial derivatives of the equation to locate the stationary points \hat{w}, \hat{b} and positive $\hat{\xi}$ for reaching a solution. The complementary slack condition implies that either $\alpha_i = 0$ or that if $\alpha_i > 0$, the i^{th} observation is classified as a support vector since it lies exactly on the decision boundary.

The partial derivatives of the equation above are as follows:

$$\frac{\partial}{\partial w} L(w, b, \alpha, \mu) = w - \sum_{i=1}^N \alpha_i t_i \theta(x_i)$$

$$\frac{\partial}{\partial b} L(w, b, \alpha, \mu) = - \sum_{i=1}^N \alpha_i t_i$$

$$\frac{\partial}{\partial \xi} L(w, b, \alpha, \mu) = C - \sum_{i=1}^N \alpha_i - \sum_{i=1}^N \mu_i$$

From there, we can set both partial derivatives equal to zero. We then obtain the properties of the stationary points:

$$\hat{w} = \sum_{i=1}^N \hat{\alpha}_i \hat{t}_i \theta(x_i) \quad (9)$$

$$\sum_{i=1}^N \hat{\alpha}_i \hat{t}_i = 0 \quad (10)$$

$$\sum_{i=1}^N \hat{\alpha}_i = C - \sum_{i=1}^N \hat{\mu}_i \quad (11)$$

By rearranging the Generalized Lagrangian equation and including the above definitions for the stationary points, we have the final dual formulation for the SVM:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j t_i t_j \theta(x_i)^T \theta(x_j) - \sum_{i=1}^N \alpha_i \quad (12)$$

s.t

$$\hat{\alpha}_i \geq 0 \quad \forall i$$

$$\sum_{i=1}^N \hat{\alpha}_i \hat{t}_i = 0$$

$$0 \leq \hat{\alpha}_i \leq C$$

We will have to find the vector for $\hat{\alpha}$ that minimizes the function above for all the observations. This is named the dual form of the SVM problem. Notice that the final set of inequalities, $0 \leq \hat{\alpha}_i \leq C$, define the box constraint C . It constraints the value of

α in a bounded region. Once the vector has been obtained, we are now able to calculate \hat{w} from equation (9). Moreover, the support vectors must consequently verify that $\hat{t}_i(\hat{w}^T\theta(x_i) + \hat{b}) = 1$, such that if the m^{th} observation is a support vector ($\hat{\alpha}_m > 0$), then we can obtain $\hat{b} = \hat{t}_m - \hat{w}^T\theta(x_m)$. We perform an average computation over all i support vectors to finally obtain an intercept value which is stable and more precise. This implies:

$$\hat{b} = \frac{1}{N-1} \sum_{i=1}^N (\hat{t}_i - \hat{w}^T\theta(x_i)) \quad (13)$$

Ultimately, since the estimated \hat{w} will have an equal number of dimensions as $\theta(x)$, it will not be computationally efficient to calculate it. Therefore, we can extract the solution of \hat{w} from Equation (9) and rearrange it into the decision function (Equation 1). By using the kernel trick for a new instance x_j , we can make classified predictions:

$$\hat{\phi}_{\hat{w},\hat{b}}(\theta(x_j)) = \hat{w}^T\theta(x_j) + \hat{b} = \left(\sum_{i=1}^N \hat{\alpha}_i t_i \theta(x_i) \right)^T \theta(x_j) + \hat{b}$$

$$\hat{\phi}_{\hat{w},\hat{b}}(\theta(x_j)) = \sum_{i=1}^N \hat{\alpha}_i t_i K(x_i, x_j) + \hat{b} \quad (14)$$

s.t

$$\hat{\alpha}_i > 0 \quad \forall i$$

Note that when $\hat{\alpha}_i > 0$, we have a support vector. The predictions will only compute the outcome on a new input x_j if it is a support vector. Concurrently, the support vectors will form the decision boundary. Hence, we will avoid the computational cost of including all the training instances. The bias term \hat{b} can be obtained in a similar fashion by combining equation (13) with equation (9).

Appendix 5:

Since the function shown in equation (3) will not be possible to obtain for inseparable data sets, we introduce a second slack variable ξ^* . Consequently, the minimization problem must include another hyperparameter which must be minimized. The new primal objective function is then the following:

$$\min_{w,b,\xi,\xi^*} \quad \frac{1}{2}w^T w + C \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (15)$$

S.t

$$y_i - (w^T \theta(x_i) + b) \leq \epsilon + \xi_i$$

$$(w^T \theta(x_i) + b) - y_i \leq \epsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \geq 0 \quad \forall i$$

This allows the errors of the regression to maximum be as large as the two slack variables while still satisfying the stated conditions. The loss function for ϵ is linear and ignores the regression errors that are smaller than the epsilon value:

$$\widehat{L}_\epsilon = \begin{cases} 0 & \text{if } |y - w^T \theta(x) + b| \leq \epsilon \\ |y - w^T \theta(x) + b| - \epsilon & \text{otherwise} \end{cases}$$

Hence, the loss measures the total distance between the observed value of y and the ϵ boundary. Moving from the primal formulation to the dual formulation is similar to what we did for the linear SVM classifier. The SVM regression problem is convex and satisfies the constrained optimization objective when moving to the dual formulation. As for the linear classification model, the regression model also introduces two Lagrange multipliers, α and μ . Both are of the form KKT where $\alpha_i, \mu_i \geq 0 \forall i$. Moving from the primal formulation to the dual formulation, the process is similar to that of the linear SVM specifications. The nonlinear SVM dual formula will have to find the coefficients that minimize:

$$\begin{aligned} \min_{\alpha, \mu} L = & \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i - \mu_i)(\alpha_j - \mu_j) K(x_i x_j) + \varepsilon \sum_{i=1}^N (\alpha_i + \mu_i) + \\ & \sum_{i=1}^N y_i (\alpha_i - \mu_i) \end{aligned} \quad (16)$$

s.t

$$\hat{\alpha}_i, \hat{\mu}_i \geq 0 \quad \forall i$$

$$\sum_{i=1}^N (\hat{\alpha}_i - \hat{\mu}_i) = 0$$

$$0 \leq \hat{\alpha}_i \leq C \quad \forall i$$

$$0 \leq \hat{\mu}_i \leq C \quad \forall i$$

Whenever the observations lie inside the epsilon boundary, the Lagrange multipliers will both be strictly equal to zero. However, when this is not the case, the respective observations will be assigned as a support vector. Notice that the box constraint value C will still decide on the boundary region for the Lagrange multipliers. The final prediction function will then only depend on the support vectors. The prediction function for the SVM regression shown below will apply the kernel function and introduce the two Lagrange multipliers.

$$\hat{\varphi}^* \left(\theta(x_j) \right) = \sum_{i=1}^N (\hat{\alpha}_i - \hat{\mu}_i) K(x_i, x_j) + \hat{b} \quad (17)$$

S.t

$$\hat{\alpha}_i (\varepsilon + \hat{\xi}_i - y_i + \hat{\varphi}^*) = 0 \quad \forall i$$

$$\hat{\mu}_i (\varepsilon + \hat{\xi}_i^* + y_i - \hat{\varphi}^*) = 0 \quad \forall i$$

$$\hat{\xi}_i (C - \hat{\alpha}_i) = 0 \quad \forall i$$

$$\hat{\xi}_i^* (C - \hat{\mu}_i) = 0 \quad \forall i$$

This finally implies that both the Lagrange multipliers will be zero whenever the observations fall strictly inside the epsilon tube. When the Lagrange multipliers are

nonzero, they will be assigned as support vectors used for predictions. This is exactly the same as for what we obtained when we derived the classification model for the SVM. Forming the decision boundary of the model will simply be determined by the support vectors.

Appendix 6:

The Augmented Dickey Fuller test and the Jarque-Bera test has been conducted to test our data set for stationarity and normality, respectively. By conducting an Augmented Dickey-Fuller test, we test for a unit root in the return series. The test statistics are defined as:

$$TS = \frac{\hat{\gamma}}{SE(\hat{\gamma})} \quad (24)$$

where $\hat{\gamma}$ is estimated through the following regression $\Delta y_t = \gamma y_{t-1} + u_t$ and Δy_t is the change in y between observation $t - 1$ and observation t . y is here denoted as the log first differences of the price estimate for the respective input variables.

For the Jarque-Bera test, we test for the normality. By applying both skewness and kurtosis, we consider the third and fourth moments of the distribution. We can test whether the coefficient of skewness and excess kurtosis are jointly equal to zero. The test statistics of Jarque-Bera is the following:

$$JB = T \left[\frac{Skew^2}{6} + \frac{(Kurt-3)^2}{24} \right] \quad (25)$$

Where *Skew* and *Kurt* represents the skewness and the kurtosis presented below, while T is the total number of observations

$$Skew = \frac{\frac{1}{N-1} \sum_{t=1}^T (y_t - \bar{y})^3}{(\sigma^2)^{3/2}} \quad (26)$$

$$Kurt = \frac{\frac{1}{N-1} \sum_{t=1}^T (y_t - \bar{y})^4}{(\sigma^2)^2} \quad (27)$$

\bar{y} is the mean value of the log first difference for the respective variable and σ is the corresponding standard deviation for the variables.

Appendix 7:

Principle Component Analysis:

PCA is a dimension reduction technique utilized in situations where multiple variables are highly correlated. By decomposing the structure of the variables into a set of factors, the new transformed variables will be uncorrelated to each other. More specifically, we have k explanatory variables which will be transformed into k uncorrelated variables.

We can construct principal components that are independent linear combinations of the raw data:

$$\begin{aligned} p_1 &= \tau_{11}x_1 + \tau_{12}x_2 + \cdots + \tau_{1k}x_k \\ &\quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ p_k &= \tau_{k1}x_1 + \tau_{k2}x_2 + \cdots + \tau_{kk}x_k \end{aligned}$$

s.t

$$\begin{aligned} \tau_{11}^2 + \tau_{12}^2 + \cdots + \tau_{1k}^2 &= 1 \\ &\quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ \tau_{k1}^2 + \tau_{k2}^2 + \cdots + \tau_{kk}^2 &= 1 \end{aligned}$$

p_k is the k th principle component estimated through independent linear combinations of the original data and x_k is the k th explanatory variable used in the regression. τ_{ij} denotes the coefficient on the j th explanatory variable which is estimated for the i th principal component. The estimated coefficients must meet the criterion that the sum of the squares of all coefficients for each respective component must sum to one.

After obtaining the principal components, a new regression can be estimated with the principal components of the first g principal components which are sufficiently

capturing the variation of the data in the response variable. The remaining $k - g$ principal components will be discarded. The proportion of the total variation in the original data set can be estimated as:

$$\mu_i = \frac{\varphi_i}{\sum_{i=1}^k \varphi_i}$$

φ_i denotes the eigenvalue of principle component i , and reflect how much of the total variation in the original data each principal component explains through μ_i . The new obtained principal components will now be orthogonal, and we have effectively removed the statistical issues related to multi-collinearity.

Appendix 8:

Table 10 - Correlation of Daily Log Returns Over the Period Between 02/01/1990 to 05/03/2019

	SP500	Dow	ESTX	Euronext	FTSE	HangSeng	Nasdaq	Russel	SSE	VIX	WTI	Treas10y	Gold	SP500t1*
SP500	1,00	0,96	0,53	0,51	0,51	0,18	0,88	0,86	0,02	-0,71	0,11	0,27	-0,05	-0,05
Dow	0,96	1,00	0,52	0,49	0,50	0,18	0,79	0,80	0,02	-0,68	0,09	0,26	-0,06	-0,05
ESTX	0,53	0,52	1,00	0,85	0,82	0,36	0,48	0,51	0,06	-0,41	0,12	0,28	-0,04	-0,04
Euronext	0,51	0,49	0,85	1,00	0,78	0,31	0,45	0,48	0,07	-0,39	0,18	0,32	-0,02	-0,06
FTSE	0,51	0,50	0,82	0,78	1,00	0,36	0,44	0,47	0,06	-0,39	0,16	0,24	0,00	-0,04
HangSeng	0,18	0,18	0,36	0,31	0,36	1,00	0,18	0,19	0,19	-0,13	0,07	0,10	0,02	-0,03
Nasdaq	0,88	0,79	0,48	0,45	0,44	0,18	1,00	0,87	0,02	-0,64	0,08	0,25	-0,05	-0,03
Russel	0,86	0,80	0,51	0,48	0,47	0,19	0,87	1,00	0,02	-0,65	0,13	0,29	-0,01	-0,04
SSE	0,02	0,02	0,06	0,07	0,06	0,19	0,02	0,02	1,00	-0,03	0,02	0,03	0,02	-0,01
VIX	-0,71	-0,68	-0,41	-0,39	-0,39	-0,13	-0,64	-0,65	-0,03	1,00	-0,08	-0,21	0,03	0,02
WTI	0,11	0,09	0,12	0,18	0,16	0,07	0,08	0,13	0,02	-0,08	1,00	0,17	0,22	-0,03
Treas10y	0,27	0,26	0,28	0,32	0,24	0,10	0,25	0,29	0,03	-0,21	0,17	1,00	-0,11	-0,03
Gold	-0,05	-0,06	-0,04	-0,02	0,00	0,02	-0,05	-0,01	0,02	0,03	0,22	-0,11	1,00	-0,02
SP500t1*	-0,05	-0,05	-0,04	-0,06	-0,04	-0,03	-0,03	-0,04	-0,01	0,02	-0,03	-0,03	-0,02	1,00

*Note that SP500t1 is the correlation between the one-day lagged log returns of the respective variables and the daily log returns of the S&P500

Appendix 9:

Figure 11 – Cumulative Log Returns of the S&P500 Between 07/05/2013 – 05/03/2019

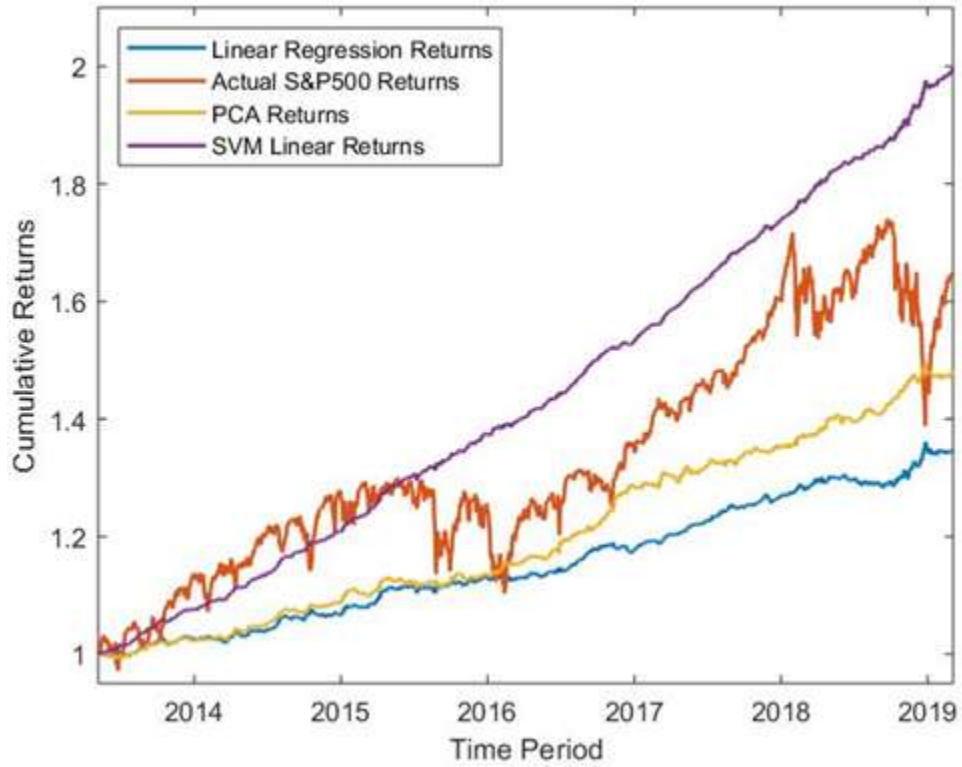


Fig 11. Predicted cumulative log returns of the S&P500 for daily data between 07/05/2013 to 05/03/2019. The red line illustrates the actual cumulative log returns of the index, while the other three lines illustrate the predicted cumulative log returns for the linear regression model, the PCA model and the Linear SVM model.

References

- Agrawal, A., Gans, J., & Goldfarb, A. (2018). *Prediction machines: the simple economics of artificial intelligence* (1st ed.). Boston, Massachusetts: Harvard Business Review Press.
- Alaty, E., & Satman H, M. (2005). Stock Market Forecasting: Artificial Neural Network and Linear Regression Comparison in An Emerging Market. *Journal of Financial Management and Analysis*, 18(2). Retrieved from https://www.researchgate.net/publication/251368933_Stock_Market_Forecasting_Artificial_Neural_Network_and_Linear_Regression_Comparison_in_An_Emerging_Market
- Atsalakis, G. S., & Valavanis, K. P. (2009). Forecasting stock market short-term trends using a neuro-fuzzy based methodology. *Expert Systems with Applications*, 36(7), 10696–10707. <https://doi.org/10.1016/J.ESWA.2009.02.043>
- Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques—Part II: Soft computing methods. *Expert Systems with Applications*, 36(3), 5932-5941. <https://doi.org/10.1016/j.eswa.2008.07.006>
- Banko, M., & Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics - ACL '01*, 26–33. <https://doi.org/10.3115/1073012.1073017>
- Basu, S. (1977). Investment Performance of Common Stocks in Relation to Their Price-Earnings Ratios: A Test of the Efficient Market Hypothesis. *The Journal of Finance*, 32(3), 663. <https://doi.org/10.2307/2326304>
- Baur, D. G., & McDermott, T. K. (2010). Is gold a safe haven? International evidence. *Journal of Banking & Finance*, 34(8), 1886–1898. <https://doi.org/10.1016/J.JBANKFIN.2009.12.008>
- Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. Retrieved from <https://www.wiley.com/en-us/Time+Series+Analysis%3A+Forecasting+and+Control%2C+5th+Edition-p-9781118675021>
- Brock, W., Lakonishok, J., & LeBaron, B. (1992). Simple Technical Trading Rules and the Stochastic Properties of Stock Returns. *The Journal of Finance*, 47(5), 1731–1764. <https://doi.org/10.1111/j.1540-6261.1992.tb04681.x>
- Campbell, J. Y., & Ammer, J. (1993). What Moves the Stock and Bond Markets? A Variance Decomposition for Long-Term Asset Returns. *The Journal of Finance*, 48(1), 3. <https://doi.org/10.2307/2328880>

- Chang, C.-L., McAleer, M., & Tansuchat, R. (2013). Conditional correlations and volatility spillovers between crude oil and stock index returns. *The North American Journal of Economics and Finance*, 25, 116–138. <https://doi.org/10.1016/J.NAJEF.2012.06.002>
- Chang, P.-C., Liu, C.-H., Lin, J.-L., Fan, C.-Y., & Ng, C. S. P. (2009). A neural network with a case based dynamic window for stock trading prediction. *Expert Systems With Applications*, 36, 6889–6898. <https://doi.org/10.1016/j.eswa.2008.08.077>
- Chen, W.-H., Shih, J.-Y., Wu, S., Chen, W.-H., Shih, J.-Y., & Wu, S. (2006). In *International Journal of Accounting and Information Systems* (Vol. 1). Retrieved from <http://webmail.svms.org/finance/ChenShihWu2006.pdf>
- Darrat, A. F., & Zhong, M. (n.d.). *On Testing the Random Walk Hypothesis: A Model-Comparison Approach*. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=286182
- Fama, E. F. (1998). Market efficiency, long-term returns, and behavioral finance. *Journal of Financial Economics*, 49(3), 283–306. [https://doi.org/10.1016/S0304-405X\(98\)00026-9](https://doi.org/10.1016/S0304-405X(98)00026-9)
- Fama, E. F., & French, K. R. (2016). Dissecting Anomalies with a Five-Factor Model. *Review of Financial Studies*, 29(1), 69–103. <https://doi.org/10.1093/rfs/hhv043>
- Fama, E. F., & French, K. R. (1988). Dividend yields and expected stock returns. *Journal of Financial Economics*, 22(1), 3–25. [https://doi.org/10.1016/0304-405X\(88\)90020-7](https://doi.org/10.1016/0304-405X(88)90020-7)
- Fama, E. F., & French, K. R. (1988). Permanent and Temporary Components of Stock Prices. *Journal of Political Economy*, Vol. 96, pp. 246–273. <https://doi.org/10.2307/1833108>
- Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669. <https://doi.org/10.1016/J.EJOR.2017.11.054>
- Géron, A. (2017). *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. O'Reilly Media Inc., California, US.
- Huang, W., Nakamori, Y., & Wang, S.-Y. (2005). Forecasting stock market movement direction with support vector machine. *Computers & Operations Research*, 32(10), 2513–2522. <https://doi.org/10.1016/J.COR.2004.03.016>

- Kalouisis, A., Gama, J., & Hilario, M. (2004). On Data and Algorithms: Understanding Inductive Performance. *Machine Learning*, 54(3), 275–312. <https://doi.org/10.1023/B:MACH.0000015882.38031.85>
- Karathanasopoulos, A., Theofilatos, K. A., Sermpinis, G., Dunis, C., Mitra, S., & Stasinakis, C. (2016). Stock market prediction using evolutionary support vector machines: an application to the ASE20 index. *The European Journal of Finance*, 22(12), 1145–1163. <https://doi.org/10.1080/1351847X.2015.1040167>
- Kim, K.-J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, 55, 307–319. [https://doi.org/10.1016/S0925-2312\(03\)00372-2](https://doi.org/10.1016/S0925-2312(03)00372-2)
- Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. In *Informatica* (Vol. 31). Retrieved from [https://datajobs.com/data-science-repo/Supervised-Learning-\[SB-Kotsiantis\].pdf](https://datajobs.com/data-science-repo/Supervised-Learning-[SB-Kotsiantis].pdf)
- Kritzman, M., Li, Y., Page, S., & Rigobon, R. (2011). Principal components as a measure of systemic risk. *The Journal of Portfolio Management*, 37(4), 112–126. <https://doi.org/10.3905/jpm.2011.37.4.112>
- Lin, Z. (2018). Modelling and forecasting the stock market volatility of SSE Composite Index using GARCH models. *Future Generation Computer Systems*, 79, 960–972. <https://doi.org/10.1016/J.FUTURE.2017.08.033>
- Loughran, T., & Ritter, J. R. (2000). Uniformly least powerful tests of market efficiency. *Journal of Financial Economics*, 55(3), 361–389. [https://doi.org/10.1016/S0304-405X\(99\)00054-9](https://doi.org/10.1016/S0304-405X(99)00054-9)
- Misra, R. (2019). Support Vector Machines—Soft Margin Formulation and Kernel Trick (Online Image). Retrieved June 27, 2019, from <https://towardsdatascience.com/support-vector-machines-soft-margin-formulation-and-kernel-trick-4c9729dc8efe>
- Mullainathan, S., & Spiess, J. (2017). Machine learning: an applied econometric approach. *Journal of Economic Perspectives*, 31(2), 87–106.
- Pagan, A. (1996). The econometrics of financial markets. *Journal of Empirical Finance*, 3(1), 15–102. [https://doi.org/10.1016/0927-5398\(95\)00020-8](https://doi.org/10.1016/0927-5398(95)00020-8)
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. *Expert Systems with Applications*, 42(1), 259–268. <https://doi.org/10.1016/J.ESWA.2014.07.040>
- Rouse, M. (2016). What is supervised learning? Retrieved May 19, 2019, from Search Enterprise AI website: <https://searchenterpriseai.techtarget.com/definition/supervised-learning>

- S. Sayad. (2019). Support Vector Machine - Regression (Online Image). Retrieved June 27, 2019, from 2019 website:
https://www.saedsayad.com/support_vector_machine_reg.htm
- Tay, F. E. H., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega*, 29(4), 309–317. [https://doi.org/10.1016/S0305-0483\(01\)00026-3](https://doi.org/10.1016/S0305-0483(01)00026-3)
- Van Gestel, T., Suykens, J. A. K., Baestaens, D.-E., Lambrechts, A., Lanckriet, G., Vandaele, B., ... Vandewalle, J. (2001). Financial time series prediction using least squares support vector machines within the evidence framework. *IEEE Transactions on Neural Networks*, 12(4), 809–821.
<https://doi.org/10.1109/72.935093>
- Vandewalle, N., Boveroux, P., & Brisbois, F. (2000). Domino effect for world market fluctuations. *The European Physical Journal B*, 15(3), 547–549.
<https://doi.org/10.1007/s100510051158>
- Vapnik, V. N. (1999). *The nature of statistical learning theory* (2nd ed.). Springer.
- Wall, R., Cunningham, P., Walsh, P., & Byrne, S. (2003). Explaining the output of ensembles in medical decision support on a case by case basis. *Artificial Intelligence in Medicine*, 28(2), 191–206. Retrieved from
<http://www.ncbi.nlm.nih.gov/pubmed/12893119>
- Support Vector Machines for Binary Classification - MATLAB & Simulink - MathWorks Nordic (Online Image). (2019). Retrieved June 27, 2019, from
<https://se.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html>
- The Big Picture: A Cost Comparison of Futures and ETFs For the Active Individual Trader*. (2016). Retrieved from <https://www.cmegroup.com/trading/equity-index/files/report-the-big-picture-for-the-active-individual-trader.pdf>