

## **FORECASTING THE CAC-40 STOCK INDEX AND ITS RETURNS: EMPIRICAL EVIDENCE FROM GLOBAL AND LOCAL MODELING**

**Stelios E. Papadakis<sup>1\*</sup> and Christos Floros<sup>2,3</sup>**

<sup>1</sup>Dept. of Business Administration, T.E.I. of Crete, Lakonia, Aghios Nikolaos, Crete, Greece

<sup>2</sup>Dept. of Accounting and Finance, T.E.I. of Crete, Estavromenos, Heraklion, Crete, Greece

<sup>3</sup>School of Social Sciences, Hellenic Open University, Patras, Greece

\*Corresponding author: [spap@staff.teicrete.gr](mailto:spap@staff.teicrete.gr)

### **ABSTRACT**

In this paper, we use global and local models to predict the price and returns of CAC-40 stock index (an international stock index). We experimentally show that predicting the stock index, itself, in terms of low absolute percentage error (APE) is feasible but impractical for forecasting the returns. We build a global model based on support vector machines, which achieves forecasting Mean absolute percentage error (MAPE) less than 1%. However, the forecasting of returns for the same period (using the same approach) is about 50%; this implies a “toss a coin to predict the result”. We formulate the claim that a global model cannot guarantee adequate forecasts of the returns. We propose an algorithm to identify those regions of the input space which are “predictable”. For each predictable region, we build a local model by using the data of the respective region. The forecasting of a new, unknown target, is feasible only if the new target belongs to a predictable region- and infeasible otherwise. The final output of our approach has three states a) sell, b) buy, d) hold (no suggestion). Experimental results on CAC-40 index shows that our model can be used as an effective decision support tool for financial analysis and trading.

Keywords: Local Modelling; Forecasting; CAC-40; Returns; Financial trading.

## 1. Introduction

Several stylized facts about financial markets include heavy tails in asset return distributions, absence of autocorrelations of asset returns, volatility clustering and asymmetry between rises and falls [13]. Further, stock markets are affected by a variety of factors (e.g. economic conditions, psychological factors of investors, company's policies, political events etc.), see [6]. There have been many approaches for modeling and forecasting financial prices due to the complexity and uncertainty of the financial markets. Researchers used linear and non-linear methods including ARIMA and GARCH models under low and high frequency datasets. These models provide several advantages (e.g. capturing volatility clustering) and disadvantages on the assumption that stock prices are noisy and non-stationary.

Furthermore, there have been impressive advances in the area of forecasting financial prices using methods from computational intelligence [1]. Artificial neural networks [5], neurofuzzy systems [8], and support vector machines [7] etc., have been successfully used as universal, model free approximators. These non-parametric models are able to capture the input-output relation of a physical system to a desired degree of accuracy from a finite set of observational input-output data. Usually, these models achieve the modeling process by minimizing the empirical risk (e.g. Mean Square Error) between the actual output of the system and the output of the model being used. However, the minimization of empirical risk on the given training set does not necessarily guarantee adequate generalization performance on an unknown set of "unseen" data (testing set).

In contrast, Support Vector Machines (SVM) are models, build on the principle of Structural Risk Minimization instead of the principle of Empirical Risk minimization (SRM). Given a finite set of training data, Vapnik and Chervonenkis introduced the well-known "VC dimension" by providing bounds on the expected generalization error. The minimization of these bounds forms the principle of Structural risk minimization. Support vector machines, introduced by [10], [14], [15], are models build on the principle of SRM. The basic advantage of SVM is the avoiding of over fitting even for a small number of training samples [2]. Moreover, the construction of a SVM model (training) is the result of a convex, quadratic constraint optimization problem and thus it is relatively fast when compared to alternative training algorithms (e.g. gradient based methods).

Despite the capacity of these models to capture the input-output relation, several studies used Machine Learning models (MLM) for predicting stock prices, while other researches used Neural Network Models (NNM). These methods have been extended to large scale machine learning methods which are able to predict financial prices and volatility. However, due to the complexity of these approaches, many studies recently used Artificial Neural Network (ANN) models and show a good generalization (prediction) performance. They argue that ANN outperforms traditional parametric statistical models; however, ANN suffers from the risk of over fitting, while their training algorithm (usually gradient based Empirical risk minimization) suffers from 'local minimum traps and difficulty in determining the structure of the model (e.g. hidden layer neurons, number of layers) and gradient optimization parameters (e.g. learning rate) as well as the hidden larger size' [17].

An improved approach to the above methodology is called Support Vector Regression (SVR)- under Support Vector Machine (SVM)- which 'has a global optimum and exhibits better prediction accuracy...' (see [17]). SVM successfully provides solutions to regression and classification clustering problems by mapping the low dimensional input space to a high dimensional Hilbert space.

SVM's solution is unique and globally optimal, unlike neural networks training which requires nonlinear optimization with the danger of getting stuck at local minima [6]. Further, the major issue here is overfitting due to risk factors.

Previous studies on the application of SVM in finance include [4],[6],[12], and [7]. In [6], a comparison of SVM's performance with those of Linear Discriminant Analysis, Quadratic Discriminant Analysis and Elman Backpropagation Neural Networks. The experiment results show that SVM outperforms the other methods. In [4] a comparison of SVM performance and back propagation neural networks in forecasting the six major Asian stock markets, while in [7] SVM to KOSPI index in forecasting Korean stock markets, is adopted. They both reported significant accuracy with SVM. Recently, [11] examines the forecasting trends of high-frequency KOSPI20 index data using learning classifiers. They find that KOSPI 1-min time series of spot index is well predicted than the futures index.

Stock prices are essentially dynamic, non-linear, nonparametric, and chaotic in nature [16]. Since 2008 financial crisis, there is a need for accurate forecasting techniques of stock prices and indices. Hence, forecasting stock price movement accurately is of great interest to financial analysts, researchers and investors. In this paper, we use global and local models to predict the price and returns of CAC-40 stock index. The CAC-40 is a benchmark French stock market index which represents a capitalization-weighted measure of the 40 most significant values among the 100 highest market caps on the Euronext Paris stock exchange.

The current study contributes to the existing literature on the empirical analysis and forecasting CAC-40 stock index price and returns. Most studies on the forecasting financial prices using the above approaches have considered standard global modeling under SVM to empirically examine the forecasting accuracy.

The contribution of this paper is twofold: (1) we examine and discuss the CAC-40 stock index prediction problem using widely used techniques (SVM), and (2) we apply a recent method introduced by [9] on local modeling to predict the sign of CAC-40 returns. This is the first study which applies the approach developed by [9] in the area of finance.

The results from a global model show that we are able to identify those regions of the input space which are “predictable”. For each predictable region, we build a local model by using the data of the respective region. The experimental results on CAC-40 index show that our proposed model can be used as an effective decision support tool for financial analysis and trading.

The rest of this paper is organized as follows. Section 2 focuses on the stock index and returns prediction with SVM. Section 3 presents a new approach based on local modeling technique to predict stock index returns.. Section 4 concludes the paper.

## 2. Problem formulation

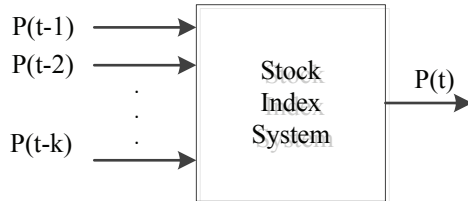
### 2.1 Stock index prediction

Typically, the stock index is a time series  $P(t), t = 1, 2, \dots, N$  of the close value  $P(t)$  of stock index at the end of day  $t$ . We take consider daily data from the CAC40 stock index for the period 1/7/2003 ( $t=1$ ) to 1/10/2010 ( $t=N$ ); the source of our data is *Datastream*. This period includes  $N=1863$  trading days. For each time step  $t$  we formulate an input-output datum  $\mathbf{D}_t$  as a vector of the form:

$$\mathbf{D}_t = [\mathbf{x}_t, y_t] = [P(t-k), \dots, P(t-1), P(t)], t = k+1, \dots, N \quad (1)$$

By using  $k > 1$  lagged values of the index. The last column of the vector is the closing value of the CAC-40 index at time  $t$  while the rest  $k$  values represents the  $k$  previous values of the index. The stock index can be considered as a physical system (Fig. 1) which has  $k$  inputs  $\mathbf{x}_t = [P(t-k), \dots, P(t-1)]$  and one output  $y_t = P(t)$ .

Fig. 1: Stock index as a system



Each datum is an instance of previous  $P(t-k)$  and current  $P(t)$  stock index values.  $N-k$  Instances as input-output observational data can be created. Based on the set of input output observational data, a regression model can be constructed to learn the relation of input output instances. If such a relation exists and the model is correctly structured then the model can abstractly learn the internal behavior of the system from the input-output data (excitation-response) observations, given. Then the model replaces the physical system and it is able to calculate the output for an unseen input (generalization). In that respect, we have a mathematical model (model) acting as a predictor which receives the previous values of the stock index (as inputs) and produces the future (currently unknown) value of the stock index. This approach is the typical modeling approach. Actually, the model is a tunable function  $f(\mathbf{w}, \mathbf{x})$ , which maps an input  $\mathbf{x}$  to an output  $y$  ( $\mathbf{x} \mapsto y$ ). The system identification process is the calculation of the values of the tunable parameters  $\mathbf{w}$ , such that a norm between the real output and the model output is minimized for the given input-output instances. A complex model's structure (large cardinality of  $\mathbf{w}$ ) might overfit the data. That is, the model produces a very small value of error norm for the given data but a large value for any new, unseen data. The risk of overfitting is high for learning algorithms based on the minimization of error norm. Relaxation techniques establish a practical application of Occam's razor principle by pursuing the better approximation with the simplest model's structure the possible. SVM incorporates relaxation through the principle of structural risk minimization, promising adequate generalization performance [2] despite the fact that some criticism on the theoretical guarantee of the generalization performance of SVMs, exists [3].

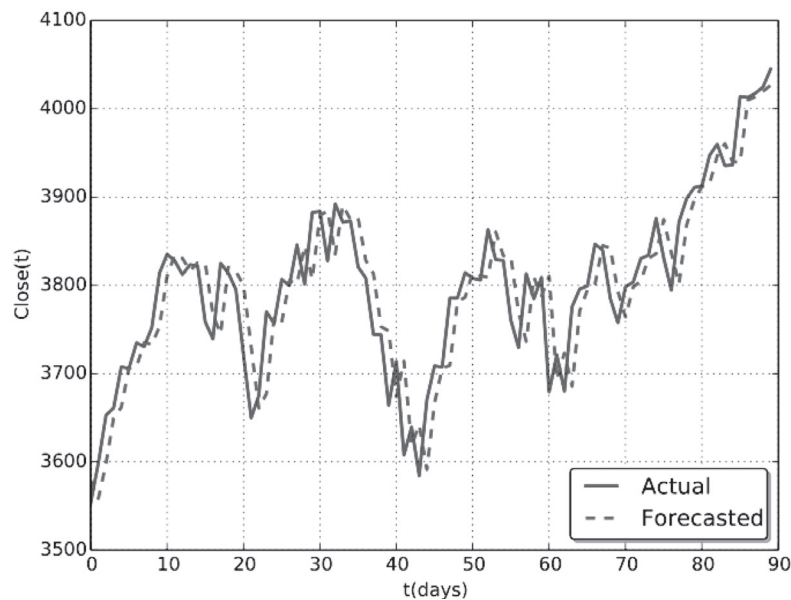
We adopt SVMs to model the stock price system in this paper, because of the abovementioned reasons. We also use the mean absolute percentage error (MAPE) as a model evaluation criterion, because it is an informative error measure. Given  $N$  input output samples  $[x_i, y_i], i = 1, \dots, N$  and a model with a transfer function  $\hat{y}_i = f(\mathbf{w}, \mathbf{x}_i)$ , the mean absolute percentage error between actual and model's output on the  $N$  samples is given by:

$$MAPE = \frac{100}{N} \cdot \sum_{i=1}^N \frac{|\hat{y}_i - y_i|}{|y_i|} \% \quad (2)$$

We split the dataset of the  $N$  observational samples (trading days) into two subsets. The first subset includes the first  $N_T$  samples and it is used as the training set for the construction of the SVM model. The second subset includes the rest  $N_c = N - N_T$  samples. It is the testing set and is only used for evaluating the generalization performance of the model. In our experiments we used  $k = 3$  lagged values. We have to underline that deciding the optimal set of lagged inputs is a crucial step. More than the previous three lagged inputs appeared statistically uncorrelated in terms of Pearson's  $R$ /Spearman's  $\rho$ , correlation coefficients. However, input selection task is out of the scope of this paper. Selecting three lagged inputs we created  $N = 1860$  samples in the form of Eq. . The first  $N_T = 1770$  samples were used for training and the rest  $N_c = 90$  for testing. A support vector regression machine (SVR) with Gaussian kernel was used. SVRs have three parameters, namely, hyper-parameters [10]: a) parameter  $\gamma$  which regulates the width of the Gaussian kernel and b) parameter  $C$  which defines the trade of between approximation performance and function's smoothing. After grid search and cross validation on the training set we decided  $\gamma = 0.3$ ,  $C = 1.0$ .

The training  $MAPE$  was 0.57% while the testing set  $MAPE$  was 0.93%, which is extremely low in terms of  $MAPE$ . This result (i.e. low  $MAPE$ ) is extremely helpful for predicting daily stock indices, but this doesn't imply the same conclusion for the stock index returns. Most traders and investors would like to know whether to sell or buy an asset (i.e. the prediction of sign of returns). Therefore, we continue our analysis with the prediction of returns, given that most finance studies focus on modeling and forecasting returns which is an important issue for financial analysts and researchers. In other words, due to the stylized facts of financial markets (as given in the Introduction), modeling and forecasting financial returns accurately help financial managers in decision making.

**Fig. 2: The prediction (dashed line) vs the actual (solid line) of the stock index for a period of 90 trading days of CAC-40. The  $MAPE$  error was 0.93%.**



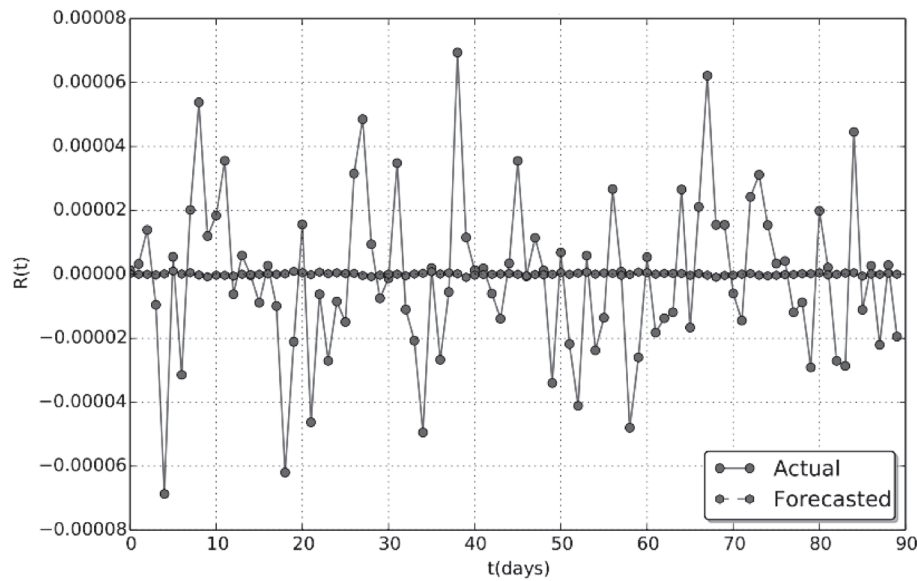
## 2.2 Predicting the returns

The underline importance of predicting the stock index is the prediction of returns. Returns are a time series  $R(t)$  created by the time series of the stock index as:

$$R(t) = \log(P(t-2)) - \log(P(t-1)) \quad (3)$$

Where  $\log(\cdot)$  is the physical logarithm.  $R(t)$  denotes the change in the trend of stock index along the time. A positive return value means that the index increases, while a negative value means that the index decreases. The prediction of  $R(t)$  sign is important since it motivates investors to sell or buy. Predicting the sign of returns from empirical data by using a systemic approach (Fig. 1) can be formulated as a classification problem where the inputs for each datum are the previous values  $\mathbf{D}_i = [R(t-3), R(t-2), R(t-1), \text{sign}(R(t))]$  where  $\text{sign}(R(t))$  is the output of the system. The last 90 samples (trading days) were used for classification while the rest for training a two class SVM classifier. The success classification performance on the testing set of 90 samples was 50%, which is equivalent to “toss for to guess the result”.

**Fig 3.:** The prediction vs. the actual return value by using a SVR model. The MAPE was 99.8%. The respective success classification rate when predicting the sign of the returns was 50%. The SVM global model failed to predict the returns despite the fact that it predicted the index with low MAPE.



### 3. The proposed approach for predicting the return

The global SVM model of the §2.2 failed to predict the sign of stock index, besides the fact that SVR successfully predicted the stock index itself, successfully. We claim that a global model is not able to predict the changes of the daily stock index. However our proposed methodology is based on the assumption that, maybe, there exist some regions of the input space which are predictable. That is, a local model, created from the data of the region, exists which is able to provide a success classification rate greater than 50%. We give the following definition of a predictable region:

**Definition 1:** A region of the input space is predictable at level  $\ell \in (0,1]$  by a model  $M$  if the average ten-fold cross-validation classification rate of the model  $M$  on the data belonging to the region is at least  $\ell \cdot 100\%$ .

If a new target datum arrives then we dynamically identify its neighbors in time  $O(\log n)$  following the methodology in [9]. Then, a ten-fold cross-validation is performed by using a SVM which is built from the data of the specific region. If the average success classification rate of the ten-fold cross-validation is greater than  $\ell$  then the region is considered as predictable and the prediction of the datum is realized by the model. Otherwise, no decision is feasible for the specific target.

We applied that approach by defining  $\ell = 0.65$  and  $\delta = 0.1$ . For the definition of  $\delta$  see [9]. We experimentally found that the 9.75% of the input data belong to predictable regions, in the sense of definition 1. For the rest 90.25% of the input data any prediction is doubtful. Regarding the predictable targets, the average success classification rate of the respective local models was 60.1%, which is far away from “toss for to guess the result”.

## 4. Conclusion

The financial variable prediction has been an important topic targeted by many researchers since successful prediction helps to make profits as well as avoid risks [11]. This paper extends the existing literature by using global and local models to predict the closing prices and returns of CAC-40 stock index. We experimentally show that predicting the stock index, itself, in terms of low absolute percentage error (APE), is feasible but impractical for forecasting the returns. Using a recent method proposed by [9], we argue that a global model cannot guarantee adequate forecasts of the returns. In other words, we identify those regions of the input space which are “predictable”. For each predictable region, we build a local model by using the data of the respective region. The experimental results on CAC-40 stock index returns show that our proposed model can be used as an effective decision support tool for financial analysis and trading. Future research should consider other advanced methods including the PSO algorithm for predicting daily and intraday returns of European and US stock indices.

## References

- [1] G. S. Atsalakis and K. P. Valavanis, “Surveying stock market forecasting techniques—Part II: Soft computing methods,” *Expert Syst. Appl.*, vol. 36, no. 3, pp. 5932–5941, 2009.
- [2] P. Bartlett and J. Shawe-Taylor, “Generalization performance of support vector machines and other pattern classifiers,” *Adv. Kernel Methods—Support Vector Learn.*, pp. 43–54, 1999.
- [3] C. J. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, 1998.
- [4] W.-H. Chen, J.-Y. Shih, and S. Wu, “Comparison of support-vector machines and back propagation neural networks in forecasting the six major Asian stock markets,” *Int. J. Electron. Finance*, vol. 1, no. 1, pp. 49–67, 2006.
- [5] E. Guresen, G. Kayakutlu, and T. U. Daim, “Using artificial neural network models in stock market index prediction,” *Expert Syst. Appl.*, vol. 38, no. 8, pp. 10389–10397, 2011.
- [6] W. Huang, Y. Nakamori, and S.-Y. Wang, “Forecasting stock market movement direction with support vector machine,” *Comput. Oper. Res.*, vol. 32, no. 10, pp. 2513–2522, 2005.
- [7] K. Kim, “Financial time series forecasting using support vector machines,” *Neurocomputing*, vol. 55, no. 1, pp. 307–319, 2003.
- [8] R. Kuo, “A sales forecasting system based on fuzzy neural network with initial weights generated by genetic algorithm,” *Eur. J. Oper. Res.*, vol. 129, no. 3, pp. 496–517, 2001.
- [9] S. E. Papadakis, V. A. Stykas, G. Mastorakis, and C. X. Mavromoustakis, “A hyper-box approach using relational databases for large scale machine learning,” in *2014 International Conference on Telecommunications and Multimedia, TEMU 2014*, 2014, pp. 69–73.
- [10] A. J. Smola and B. Schölkopf, *Learning with kernels*. Citeseer, 1998.
- [11] Y. Son, D. Noh, and J. Lee, “Forecasting trends of high-frequency KOSPI200 index data using learning classifiers,” *Expert Syst. Appl.*, vol. 39, no. 14, pp. 11607–11615, 2012.
- [12] F. E. Tay and L. Cao, “Application of support vector machines in financial time series forecasting,” *Omega*, vol. 29, no. 4, pp. 309–317, 2001.
- [13] J.-J. Tseng and S.-P. Li, “Asset returns and volatility clustering in financial time series,” *Phys. Stat. Mech. Its Appl.*, vol. 390, no. 7, pp. 1300–1314, 2011.
- [14] V. Vapnik, “Principles of risk minimization for learning theory,” in *Advances in neural information processing systems*, 1992, pp. 831–838.
- [15] V. Vapnik, *The nature of statistical learning theory*. Springer Science & Business Media, 2000.
- [16] J.-Z. Wang, J.-J. Wang, Z.-G. Zhang, and S.-P. Guo, “Forecasting stock indices with back propagation neural network,” *Expert Syst. Appl.*, vol. 38, no. 11, pp. 14346–14355, 2011.
- [17] C.-Y. Yeh, C.-W. Huang, and S.-J. Lee, “A multiple-kernel support vector regression approach for stock market price forecasting,” *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2177–2186, 2011.