Quantitative Finance and Economics

http://www.aimspress.com/journal/QFE

QFE, 2(3): 525–553.
DOI:10.3934/QFE.2018.3.525
Received: 13 December 2017
Accepted: 12 February 2018
Published: 19 July 2018

Research article

A hybrid forecasting algorithm based on SVR and wavelet decomposition

Timotheos Paraskevopoulos*and Peter N Posch

TU Dortmund University, Finance, Otto-Hahn-Str. 6, 44227 Dortmund, Germany

* **Correspondence:** Email: timotheos.paraskevopoulos@udo.edu; Tel: +492317553271.

Abstract: We present a forecasting algorithm based on support vector regression emphasizing the practical benefits of wavelets for financial time series. We utilize an effective de-noising algorithm based on wavelets feasible under the assumption that the data is generated by a systematic pattern plus random noise. The learning algorithm focuses solely on the time frequency components, instead of the full time series, leading to a more general approach. Our findings propose how machine learning can be useful for data science applications in combination with signal processing methods. The time-frequency decomposition enables the learning algorithm to solely focus on periodical components that are beneficial to the forecasting power as we drop features with low explanatory power. The proposed integration of feature selection and parameter optimization in a single optimization step enable the proposed algorithm to be scaled for a variety of applications. Applying the algorithm to real life financial data shows wavelet decompositions based on the Daubechie and Coiflet basis functions to deliver the best results for the classification task.

Keywords: regression; spectral analysis; time series; financial markets
JEL codes: C63, C53

1. Introduction

Forecasting stock market index movements is challenging as the underlying process is characterized by significant fluctuations and exhibits nonlinear dynamics. There has been a tremendous amount of research on this topic over the past years and a great deal of effort has been made in order to gain advantages in financial time series predictions. Atsalakis and Valavanis (2009) provide a cohesive presentation and classification of the research on stock market forecasting. The most widely used methods in forecasting such as autoregressive integrated moving average models, the exponential smoothing method, various models for seasonality, structural models, the Kalman filter and nonlinear models (including regime-switching models) could not reliably outperform the random walk (Atsalakis and Valavanis, 2009). Machine learning provides a theory to construct algorithms that are able to learn, memorize and generalize from data. Generalization is essential to produce sufficiently accurate predictions from new and unseen patterns after having experienced learning on a related data set. However, before an algorithm is able to generalize it must be able to recognize the observed patterns which is achieved by the preceding learning process. Therefore, the incremental informative content of the considered learning sample is crucial. Redundant, noisy or unreliable information impair the learning process and undermine the success of machine. Given the assumption that the considered data is generated by a systematic pattern plus random noise, an effective de noising algorithm provides a deeper understanding of the data generating process leading to a more precise forecast. Wavelet transformation, as a signal processing technique, enables the decomposition of a time series into the time and the frequency domain resulting to powerful feature extraction capabilities Li et al. (2002); Kim and Han (2000); Gençay et al. (2001); Zheng et al. (1999); Huang and Wu (2008); DeLurgio (1998); Seo et al. (2015). In this study we introduce a rolling hybrid wavelet support vector regression with a precede recursive feature elimination (WL-RFE-SVR) algorithm, which is apart from traditional econometric and time series techniques. Our proposed algorithm consists of three stages which are conducted every walking forward step. In the first stage we split the sample set into a training, validation and test set. Each sub-sample is subsequently decomposed into its low and high frequency components using wavelet transform which enables us to extract additional information from the considered time series (Chang and Fan, 2008; Li et al., 2002; Lu et al., 2009; Ramsey, 2002). In the second step, we identify the optimal feature set and support vector regression (SVR) hyperparameters using the recursive filter elimination (RFE) and hyperparameter optimization routine based on the training and validation set. In the third step, we apply the identified optimal feature set and SVR parameters on the test and obtain the out-of-sample (OOS) forecasts.

To best of our knowledge, the proposed WL-RFE-SVR methodology has not been presented in the literature, however similar ideas have been discussed by Huang and Wu (2008). They apply a relevance vector machine on wavelet-based extracted features to forecast stock indices. In contrast to our work, the feature space dimension is not reduced by a feature selection algorithm.

An algorithm based on wavelet decompositions and a MLP neural network with a financial application shows promising profitability when the data set is decomposed into low and high frequency components (Zhang et al., 2001). Kao et al. (2013) provide a similar hybrid approach which, however, it requires multiple optimization steps for feature selection and parametrization of the SVR. In addition, their application is based on discrete wavelet transforms (DWT) which requires the sample to be dyadic, ie. observations of pairs each from disjunct sets. This restricts the application of the SVR as the size of the training set cannot be chosen arbitrarily. A recent work by Wang et al. (2017) combines wavelet analysis with correlation analysis to construct multiscale networks. This new perspective on financial markets emphasizes the varying correlation structure across different time scales. The results in Wang et al. (2017) strongly support our idea to process financial time-series separately as time-frequency decompositions.

We contribute to the literature by introducing a hybrid algorithm for processing data based on wavelet transform and SVR which integrates feature selection and parameter optimization in a single optimization step. Thereby we shed light on wavelet decompositions with the purpose to improve the forecast performance of a SVR by systematically comparing several wavelet filters and investigate the impact of short term to long term forecasting horizons on the performance of a trading application.

The remainder of the paper is as follows: We introduce and discuss the proposed method in the following section 2. In sections 2.1, 2.2 and 2.3 we give an introduction into wavelet transform, recursive feature selection and describe the concept of SVR. Section 3 focuses on the application of the method to real life data, while the final section concludes.

2. Methodology

The proposed forecasting algorithm is a purely time-delayed model which captures the relationship between historical and subsequent index log returns. An effective adaptation to changing market conditions is achieved by optimization which follows the purpose of determining the most robust implementation of the forecasting algorithm within a specific time interval. A walk-forward routine is a well-suited systematic approach for this purpose as it allows to measure the robustness of the forecasting algorithm exclusively on the basis of out of sample forecasts. In order to effectively eliminate the likelihood that out of sample results are just a matter of luck, the procedure is conducted over a time span of 4 years resulting in a series of 1200 successive out of sample tests. For each walking-forward step of 100 observations, we split the past 1100 observation into a training set (800 observations), a validation set (200 observations) and a test set (100 observations). Subsequently, we decompose each sub-sample separately into its low and high frequency components which are used as an input. The output is defined as the log return for the point-forecast task while for the classification task we label the data as follows:

$$y_t := \begin{cases} -1 & if \quad y_t < 0 \\ 1 & if \quad y_t > 0 \end{cases} \tag{1}$$

We use the training set to train our SVR with a specific feature set and hyper-parameters while we asses its performance on the validation set. We pick the SVR, which is characterized by a specific feature set and its hyper-parameters, which performs best on the validation set and produce 100, depending on the forecast horizon, 1, 5, 10 and 20 days ahead, forecasts. In particular, the data preprocessing component follows the key objective of de-noising. We obtain the trend part by setting all high frequency wavelet coefficients equal to zero and applying the inverse wavelet transform. To obtain the variation part we set the low frequency wavelet coefficients equal to zero and perform the same procedure. To avoid a looking forward bias, we decompose each sub-set using a rolling window approach with a window length of 200 observations and a step size of one observation. Subsequently, the algorithm decides based on a RFE algorithm the amount of trend and variation components that will be utilized to perform the forecasts. For each feature subset, we perform a grid search optimization in order to identify the optimal SVR hyperparameters.

The flowchart in Figure 1 depicts the proposed methodology in detail. Instead of restricting ourselves to a single filter, we provide a comparative analysis of a wide range of wavelet transform filters which are adequate for financial time series analysis (Percival and Walden, 2006). In order to ensure better comparability, we extended the study by a naive strategy. For this strategy, we do not preprocess the data with a wavelet transform. An important property of this algorithm is the scalability which allows additionally to point and classification tasks, the incorporation of trading strategies into the model.

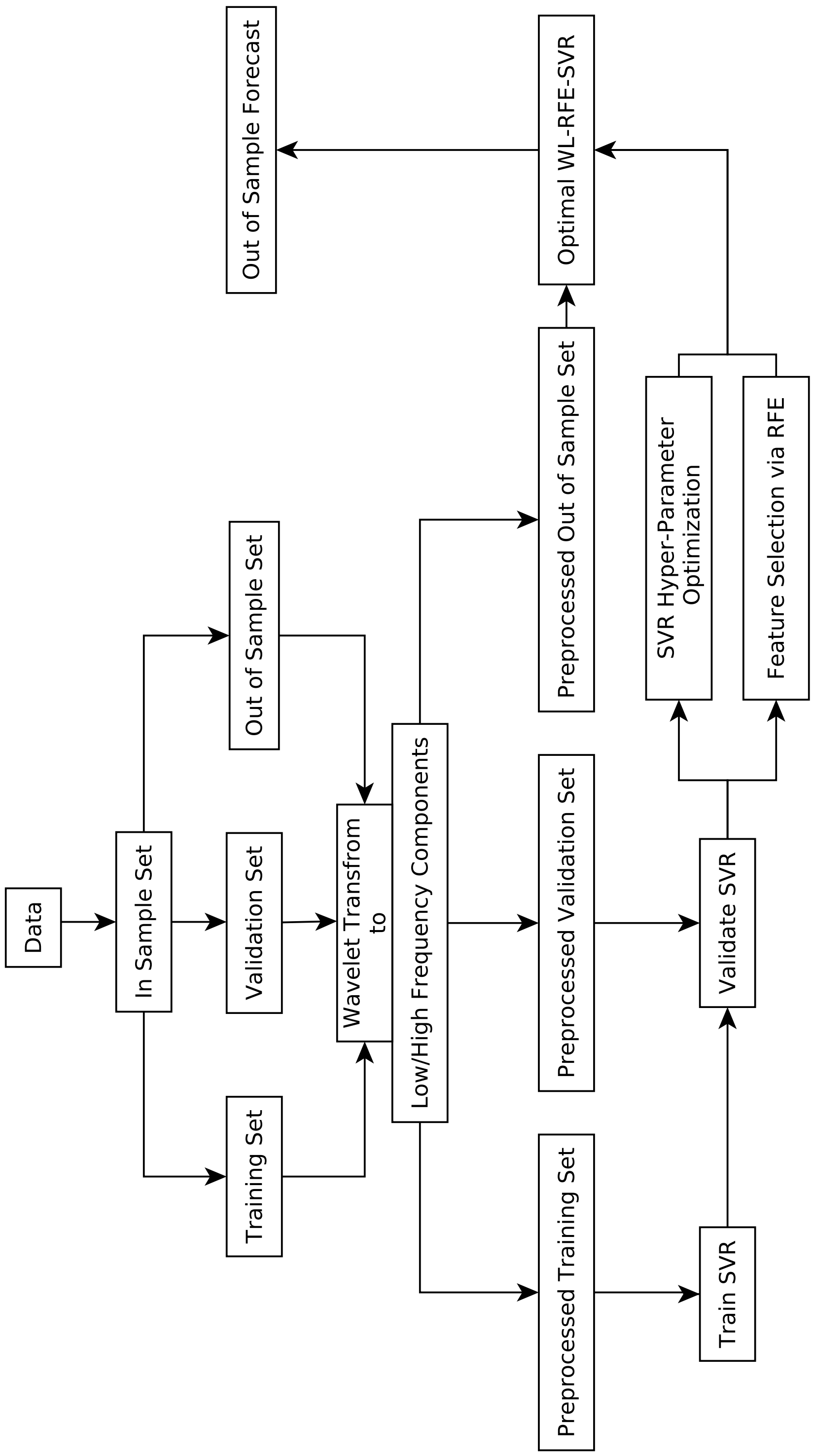

Figure 1. Flowchart of the proposed WL-RFE-SVR algorithm.

2.1. Wavelet transforms

Wavelet transforms are based on Fourier analysis, which represents any function as the sum of the sine and cosine functions. A Fourier transform is not applicable to non-stationary signals as it assumes the signal to be periodic. In contrast, a wavelet transform is a mathematical tool that can be applied to numerous applications, including non-stationary time series. The wavelet transform defines a finite domain which makes it well localized with respect to both time and frequency. Our approach is based on the maximal overlap discrete wavelet transform (MODWT) as we can apply it to an arbitrary large sample whereas a discrete wavelet transform restricts the sample size to be dyadic. Additionally, the trend and variation coefficients of a MODWT can be perfectly aligned with the original time series as they are associated with zero-phase filters (Percival and Walden, 2006). Depending on the normalization rules, there are two types of wavelets within a given function/family. The father wavelets which describe the low-frequency components of a signal and the mother wavelets which describe the high-frequency components:

$$\Phi_{j,k} = 2^{-j/2}\Phi(t - 2^j k/2^j) \tag{2}$$

$$\Psi_{j,k} = 2^{-j/2}\Psi(t - 2^j k/2^j) \tag{3}$$

where $\Phi(t)$ represents the father wavelet, $\Psi(t)$ the mother wavelet and $j = 1, ..., J$ is the J-level wavelet decomposition (Ramsey and Lampart, 1998). The two types of wavelets stated above, satisfy

$$\int \Phi(t)dt = 1 \text{ and } \int \Psi(t)dt = 0. \tag{4}$$

The low-frequency parts can be understood as the smooth baseline trend from a signal whereas the high-frequency parts are any deviations from that. Given a signal $f(t)$, its wavelet transform is formed by (Percival and Walden, 2006):

$$f(t) \approx \sum_k s_{J,k}\Phi_{J,k}(t) + \sum_k d_{J,k}\Psi_{J,k}(t) + \sum_k d_{J-1}\Psi_{J-1,k}(t) \tag{5}$$

$$+ ... + \sum_k d_{1,k}\Psi_{1,k}(t) \tag{6}$$

The wavelet transform coefficients are given by $s_{J,k}$ and $d_{J,k}, ..., d_{1,k}$ where J denotes the maximum decomposition level sustainable by the number of data points. More precisely, 2^j is less than the total number of data points. The parameter k is defined as the dilation parameter ranging from one to the number of wavelet coefficients. The high and low frequency component coefficients $s_{j,k}$ and $d_{j,k}$ can be approximated by the following integrals:

$$s_{J,k} = \int f(t)\Phi_{J,k}(t)dt \tag{7}$$

$$d_{j,k} = \int f(t)\Psi_{j,k}(t)dt. \tag{8}$$

Given the coefficients, we obtain the low-frequency component of a signal by setting all high-frequency coefficients $d_{j,k}$ for $j = 1, ..., J$ to zero while keeping all low-frequency coefficients $s_{J,k}$

Chang and Yadama (2010). Accordingly, we obtain the high-frequency components by setting the low-frequency coefficients $s_{J,k}$ to zero while keeping the high-frequency components.

The corresponding wavelet coefficients are approximated by the specific basis functions $\Phi_{J,k}(t)$ and $\Psi_{J,k}(t)$. Each basis function leads to a different transform as a given signal is weighted over several scales and center time differently. The issue of which basis function generates the best resolution for a specific forecasting application is considered as the most indispensable challenge in wavelet analysis and needs to be addressed by trial and error. On this account, we provide a performance comparison for basis functions appropriate for time series analysis. In general, low order wavelets have good time resolution but poor frequency resolution whereas high order wavelets have good frequency resolution but poor time resolution. As it is yet unknown how this, in particular, affects the performance of our forecasting algorithm during different market regimes we conduct the analysis for several filter lengths and basis functions: Daubechies (D) (4, 6, 8, 12), Least Asymmetric (LA) (8, 10, 12, 14), Best Localized (BL) (14, 18, 20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. Daubechies belong to the most common applied basis functions for time series denoising. CF filters allow specifying vanishing conditions on the associated scaling function leading to remarkably good phase properties (Daubechies, 1992). LA filters have approximately linear phase while minimizing the maximum deviation of the phase function from the linear phase over the low and high frequencies (Percival and Walden, 2006). Based on the idea that deviations from the high-frequency phase are not as important as ones from low frequencies, Doroslovacki (1998) introduced the BL filter which penalizes deviations at low frequencies more heavily than those at high frequencies.

2.2. Feature selection

To achieve the best possible performance with a particular learning algorithm it is necessary for the learning algorithm to focus on the most relevant information in a potentially overwhelming quantity of input variables. A decisive countermeasure to reduce the dimensionality of the input variables is provided by proper feature selection algorithms. Feature selection approaches are divided into two categories: filter and wrapper methods (John et al., 1994). Filter based methods use a mathematical evaluation function to justify a plausible relationship between the input and output variables. The interested reader is referred to Saeys et al. (2007), for a survey regarding filter methods. In contrast to filter methods, wrapper methods are brute force search algorithms which maximize model's performance by adding or removing features while evaluating each feature subset in the context of the learning algorithm. The subset selection takes place based on the learning algorithm used to train the model itself and allows to consider how the algorithm and the training set interact. For the purpose of feature selection, we choose a RFE algorithm which repeatedly constructs our proposed model and ranks features sets according to the achieved root mean square error (RMSE) for the point forecast task and according to accuracy for the classification task. However, an adverse effect on the generalization capability of our forecasting algorithm is the potential issue of overfitting. This occurs when the feature set focus on nuances of the training data which are not found in future samples (i.e. overfitting to predictors and samples). For example, suppose a very large number of uninformative features were collected and one such feature set randomly correlates with the outcome. The RFE algorithm would give a good rank to this feature set and the prediction error (on the same data set) would be lowered. It would take a different test/validation sample set to find out that this predictor was uninformative. This is referred to as selection bias by McLachlan (2002). To avoid the potential issue of over-fitting and selection bias we resampled each training set using a 10-fold cross-validation (Svetnik et al., 2004; Ambroise and McLachlan, 2002).

2.3. Support vector regression

The supervised learning technique SVR is an application form of Support Vector Machines which deals with estimating real-valued functions by allowing the output to take any real number (Vapnik et al., 1997). The key advantage of the SVR is that it allows for noise in the data by addressing the loss of accuracy from large residuals caused by outliers using structural risk minimization.

2.3.1. Structural risk minimization

Given a training set of labelled data drawn from an unknown distribution $P(x, y)$

$$\{(x_1, y_1), (x_2, y_2), (x_3, y_3), ..., (x_l, y_l)\}, x \in \mathbb{R}^n \tag{9}$$

where the input variable $x_i \in \mathbb{R}^l$ is a n-dimensional vector and the output variable $y_i \in \mathbb{R}$ is a continuous value. The problem of learning is to determine a function $f(x, \alpha)$ which approximates the relationship between x_i and y_i with a parameter vector α as accurate as possible. The best approximation of y_i minimizes the following expected risk $R(\alpha)$:

$$R(\alpha) = \int L(y, f(x, \alpha))dP(x, y), \tag{10}$$

where $L(y, f(x, \alpha))$ is a loss function describing the difference between the true y_i and its approximation:

$$L(y, f(x, \alpha)) = (y - f(x, \alpha))^2 \tag{11}$$

consequently Equation (10) can be written as

$$R(\alpha) = \int (y - f(x, \alpha))^2 dP(x, y) \tag{12}$$

Since x_i and y_i follow an unknown distribution, Equation (10) can solely be minimized by an inductive step. Hence, we replace the expected risk $R(\alpha)$ by the empirical risk

$$R_{emp}(\alpha) = \frac{1}{N} \sum_{i=1}^{l} (y - f(x, \alpha))^2 \tag{13}$$

However, minimizing the empirical risk does not necessarily imply a small expected risk in small samples which hampers generalizing from unseen data. A solution to this issue is provided by structural risk minimization principle which simultaneously minimizes the empirical risk and a confidence interval Ω (Vapnik, 1999). Given the VC-dimension, defined as the parameter h, the bound on the expected return can be written as:

$$R(\alpha) \leq R_{emp}(\alpha) + \Omega\left(\frac{l}{h}\right), \tag{14}$$

whereby the VC-dimension can be interpreted as the model capacity or complexity of the learning machine. For a given amount l of data and predetermined VC-dimension h^*, the generalization ability of the machine depends on the confidence interval $\Omega(l/h^*)$. In case that we construct the learning machine too complex, we may be able to minimize the empirical risk $R_{emp}(\alpha)$ down to zero. However, the confidence interval Ω would be in this case large implying a large expected risk. This case is called overfitting and can be avoided by choosing the VC-dimension that minimizes the expected risk while controlling both terms in Equation (14). In support vector methods, the tradeoff between the quality of the approximation and the complexity of the approximating function is determined by the parameter C. This parameter can be interpreted as the degree of penalized loss when a training error occurs and is described in more detail in the following section.

2.3.2. Linear support vector regression

In order to approximate the output vector y, the SVR model performs a linear regression with an error tolerance ϵ. The regression function has the following form:

$$\hat{y}_i = f(x_i) = w_0 + w \cdot x, \tag{15}$$

where w_0 and w^T are the parameter vectors of the function. The tolerated losses within extent of the ϵ-tube as well the penalized losses L_ϵ are defined by the ϵ-insensitive loss function,

$$L_\epsilon(y, \hat{y}) = \begin{cases} 0 & \text{if} \quad |y - \hat{y}| \le \epsilon \\ |y - \hat{y}| - \epsilon & \text{if} \quad |y - \hat{y}| > \epsilon \end{cases} \tag{16}$$

The linear regression $f(x_i)$ is estimated by simultaneously minimizing $||w||^2$ and the sum of the linear ϵ-insensitive loss function that is given by Equation (16). Hence the objective function can be written in the following, not differentiable, convex and unconstrained form:

$$L_\epsilon = C \sum_{i=1}^{N} L_\epsilon(y_i, \hat{y}_i) + \frac{1}{2}||w||^2 \tag{17}$$

where $C = 1/\lambda$ is a regulation constant which controls the trade-off between the complexity and the approximation accuracy of the regression model. By introducing two slack variables ξ_i^+ and ξ_i^-, $i = 1, 2, ..., n$, which represent the upper and lower training errors subject to the ϵ-tube we are able to formulate the problem as a constrained optimization problem:

$$y_i \le f(x_i) + \epsilon + \xi_i^+ \tag{18}$$

$$y_i \ge f(x_i) - \epsilon - \xi_i^-. \tag{19}$$

Consequently, Equation (17) can be transformed into the following constrained form:

$$\min_{w, w_0, \xi_i^+, \xi_i^-} \quad R_{reg}(f) = \frac{1}{2}||w||^2 + C \sum_{i=1}^{l} (\xi_i^+ + \xi_i^-) \tag{20}$$

with subject to:

$$y_i - (w \cdot x_i) - w_0 \le \epsilon + \xi_i^+ \tag{21}$$

$$-y_i + (w \cdot x_i) + w_0 \leq \epsilon + \xi_i^- \tag{22}$$

$$\xi_i^+, \xi_i^- \geq 0, \quad for \quad i = 1, ..., l \tag{23}$$

Minimizing the first term is equivalent to minimizing the confidence interval Ω since we minimize the complexity, whereas minimizing the second term is equivalent to minimizing the empirical risk. The constrained optimization problem can be solved by applying the Lagrangian theory and the Karush Kuhn-Tucker condition as has been done for SVM classifiers (Chang and Fan, 2008; Schölkopf et al., 2005). Introducing a dual set of Langrange multipliers $\underline{a}_i$ and $\bar{\alpha}_i$ enables applying the standard quadratic programming algorithm and thus to solve the optimization problem in the dual form:

$$\min_{\underline{\alpha}_i, \bar{\alpha}_i} \quad \frac{1}{2} \sum_{i,l=1}^{l} (\underline{\alpha}_i - \bar{\alpha}_i)(\underline{\alpha}_i - \bar{\alpha}_i)\langle x_i \cdot x_j \rangle \tag{24}$$

$$+\epsilon \sum_{i=1}^{l} (\underline{\alpha}_i + \bar{\alpha}_i) - \sum_{i=1}^{l} y_i(\underline{\alpha}_i - \bar{\alpha}_i) \tag{25}$$

with subject to:

$$\sum_{i=1}^{l} (\underline{\alpha}_i - \bar{\alpha}_i) = 0 \tag{26}$$

$$0 \leq \underline{\alpha}_i \leq C, \quad i = 1, 2, ..., l \tag{27}$$

$$0 \leq \bar{\alpha}_i \leq C, \quad i = 1, 2, ..., l \tag{28}$$

Once the model is trained, by calculating the Lagrangian multipliers $(\underline{\alpha}_i, \bar{\alpha}_i)$, we are able to make predictions using

$$\hat{y}(x) = \hat{w_0} + \sum_{j,i=1}^{l} (-\underline{\alpha}_i + \bar{\alpha}_i)\langle x_i \cdot x_j \rangle \tag{29}$$

where $\langle x_i \cdot x_j \rangle$ is the inner product of x_i and x_j. The support vectors are given by x_i for which the corresponding coefficients $(-\underline{\alpha}_i + \bar{\alpha}_i)$ are nonzero. These are the data points for which the errors lie outside the ϵ-insensitive tube and thus, support the definition of the approximate function.

2.3.3. Nonlinear support vector regression

The practical use of a linear approximation is very limited when the data is nonlinear. However, the input data x_i can be mapped in a higher dimensional feature space by a nonlinear function $\Phi(x)$ in which it exhibits linearity. Consequently, the higher dimensional feature space can be lineary approximated. The decision function is now given by

$$\hat{y}_i = f(x_i) = w_0 + \langle w \cdot \Phi(x) \rangle, \tag{30}$$

The optimization problem as stated in Equation (20) can be expressed as:

$$\min_{w,w_0,\xi_i^+,\xi_i^-} \quad R_{reg}(f) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i^+ + \xi_i^-) \tag{31}$$

with subject to:

$$y_i - (w \cdot \Phi(x_i)) - w_0 \leq \epsilon + \xi_i^+ \tag{32}$$

$$-y_i + (w \cdot \Phi(x_i)) + w_0 \leq \epsilon + \xi_i^- \tag{33}$$

$$\xi_i^+, \xi_i^- \geq 0, \quad for \quad i = 1, ..., l \tag{34}$$

The dual form of the optimization problem for the nonlinear SVR is given by:

$$\min_{\underline{\alpha}_i,\bar{\alpha}_i} \quad \frac{1}{2}\sum_{i,l=1}^{l}(\underline{\alpha}_i - \bar{\alpha}_i)(\underline{\alpha}_i - \bar{\alpha}_i)\langle\Phi(x_i) \cdot \Phi(x_j)\rangle \tag{35}$$

$$+ \epsilon\sum_{i=1}^{l}(\underline{\alpha}_i + \bar{\alpha}_i) - \sum_{i=1}^{l} y_i(\underline{\alpha}_i - \bar{\alpha}_i) \tag{36}$$

with subject to:

$$\sum_{i=1}^{l}(\underline{\alpha}_i - \bar{\alpha}_i) = 0 \tag{37}$$

$$0 \leq \underline{\alpha}_i \leq C, \quad i = 1, 2, ..., l \tag{38}$$

$$0 \leq \bar{\alpha}_i \leq C, \quad i = 1, 2, ..., l \tag{39}$$

In the final form of the prediction function the scalar products $\langle x_i \cdot x_j\rangle$ are replaced by $\langle\Phi(x_i)\cdot\Phi(x_j)\rangle$ while the scalar product is computed by a kernel function $K(x_i, x_j)$ which enables to avoid a mapping $\Phi(x)$. The key advantage here is that one does not necessarily need to specify the $\Phi(x)$ function in order to yield the inner products $\langle\Phi(x_i) \cdot \Phi(x_j)\rangle$. Once the Lagrangian multipliers are calculated, the kernelized SVR regression function can be written as

$$\hat{y}(x) = \hat{w}_0 + \sum_{j,i=1}^{l}(-\underline{\alpha}_i + \bar{\alpha}_i)K(x_i, x_j) \tag{40}$$

Practically, the choice of the kernel function is restricted to those which satisfy Mercer's theorem (Vapnik, 1999; Vapnik et al., 1997). A popular kernel function which we also apply in our study is the Radial Basis Function (RBF):

$$K(x_i, x_j) = exp(-\gamma\|x_i - x_j\|^2). \tag{41}$$

The choice of the RBF kernel relies on the results of a trial-error-process in which we identified the RBF kernel as the most suitable kernel among the linear, polynomial and sigmoid kernel. This finding is confirmed by further applications on financial time series in the literature (Ince and Trafalis, 2008; Lu et al., 2009).

2.3.4. Parametrization of the nonlinear SVR

The generalization perfomance of the SVR machine strongly depends on the parameter C, the insensitive tube ϵ and the kernel parameter γ (Cherkassky and Ma, 2004). Additionally, the parameters are mutually dependent which hampers the optimization process. The greater the parameter C, the greater the penalty of errors which forces the machine to become more complex in order to achieve a low empirical risk. A smaller C leads to an increased error tolerance which in return leads to a poor approximation. However, such a behaviour may be preferred in case of noisy data. The ϵ-insensitive tube affects the learning machine complexity as it determines the amount of support vectors. A greater ϵ leads to a greater ϵ-insensitive tube. As a result, a larger amount of data points which are within the ϵ-insensitive are ignored by the learning machine resulting in flattening the regression function. We identify adequate hyperparameters for our SVR based on a grid search within a cross-validation routine which we conduct at each walking-forward step on historical data. By doing this, we use estimated parameters from the literature to restrict the scope of our grid search (Cherkassky and Ma, 2004; Mattera and Haykin, 1999). As the scope of this study does not rely on the optimal parametrization of a SVR, the grid search results will not be further discussed.

3. Empirical results

Our data set consists of daily bid and ask closing prices from May, 18th 2007 to Dec, 30th 2015 from nine major stock indices: US (S&P 500), United Kingdom (FTSE100), Japan (Nikkei225), Germany (DAX30), France (CAC40), Switzerland (SMI20), Australia (ASX200) and Hong Kong (HS33). The selection was made with the premise of providing a comprehensive analysis on the most traded indices based on 2200 observations per each index. From the daily closing price P_t the calculate the log returns for a period of l days at time τ as $r_\tau = ln\,(P_\tau/P_{\tau-l})$.

We develop the WL-RFE-SVR algorithm based on the mid price while conducting the trading application based on the bid and ask price to account for transaction costs. As we aim at predicting the logarithmic return 1, 5, 10, 15 and 20 trading days ahead for each index, the parameter l varies respectively from 1 to 20. The first 1000 observations are used to initialize the model. We re-train the model every 100 observations thereby producing 1200 out-of-sample forecast for every index. The descriptive statistics for each index and forecasting horizon are given in Table 1. We observe, with the exception of DAX, a negative skewness for every time series. Additionally, every time series prevails a kurtosis greater than 2.4 and is non-normal as confirmed by the Jarque-Bera statistics at the 99 percent confidence interval.

Table 1. Descriptive statistics for each symbol and granularity. The granularity varies between +1, +5, +10, +15, +20 days.

Symbol	DAX	FTSE100	SP500	ASX200	SMI20	CAC40	HangSeng33	Nikkei225
				1 Trading Day				
Mean	0	0	0	0	0	0	0	0
Median	0.001	0.001	0	0.001	0.001	0.001	0.001	0
Std Dev	0.015	0.013	0.011	0.011	0.011	0.015	0.016	0.015
Skewness	0.02	-0.142	-0.172	-0.49	-0.526	-0.225	-0.021	-0.168
Kurtosis	11.115	12.824	13.747	7.296	11.576	7.508	6.224	8.783
Max	0.158	0.125	0.112	0.077	0.1055	0.116	0.111	0.131
Min	-0.104	-0.111	-0.105	-0.092	-0.121	-0.107	-0.116	-0.111
				5 Trading Days				
Mean	0.002	0.001	0.001	0.001	0.001	0	0.001	0.001
Median	0.004	0.002	0.003	0.003	0.003	0.003	0.003	0.003
Std Dev	0.031	0.025	0.02	0.02	0.02	0.032	0.034	0.032
Skewness	-0.816	-0.751	-0.942	-0.766	-0.995	-0.699	-0.501	-1.006
Kurtosis	4.909	6.791	8.652	4.458	8.164	4.089	6.117	10.460
Max	0.172	0.172	0.154	0.109	0.151	0.152	0.215	0.244
Min	-0.201	-0.174	-0.204	-0.161	-0.239	-0.191	-0.318	-0.311
				10 Trading Days				
Mean	0.003	0.001	0.0016	0.002	0.002	0.001	0.002	0.002
Median	0.008	0.004	0.005	0.005	0.005	0.007	0.005	0.004
Std Dev	0.041	0.033	0.031	0.031	0.033	0.04	0.047	0.043
Skewness	-1.243	-1.136	-1.529	-1.054	-1.089	-1.003	-0.636	-1.125
Kurtosis	4.429	5.334	9.443	3.646	2.895	2.742	4.474	6.459
Max	0.155	0.151	0.134	0.115	0.14	0.164	0.261	0.233
Min	-0.279	-0.259	-0.287	-0.239	-0.227	-0.249	-0.381	-0.415
				15 Trading Days				
Mean	0.004	0.002	0.002	0.002	0.002	0.001	0.002	0.002
Median	0.012	0.007	0.007	0.007	0.007	0.009	0.005	0.008
Std Dev	0.05	0.041	0.037	0.037	0.04	0.047	0.056	0.052
Skewness	-1.279	-1.282	-1.626	-0.989	-0.973	-1.058	-0.676	-1.179
Kurtosis	4.429	5.334	9.43	3.646	2.895	2.742	4.474	6.458
Max	0.153	0.154	0.204	0.159	0.147	0.141	0.260	0.217
Min	-0.298	-0.285	-0.308	-0.241	-0.255	-0.283	-0.477	-0.444
				20 Trading Days				
Mean	0.006	0.002	0.003	0.002	0.003	0.002	0.004	0.003
Median	0.013	0.008	0.01	0.009	0.009	0.010	0.008	0.008
Std Dev	0.056	0.045	0.043	0.042	0.045	0.054	0.066	0.061
Skewness	-1.217	-1.241	-1.673	-0.888	-1.096	-1.021	-0.700	-1.129
Kurtosis	4.075	4.775	8.232	2.589	3.412	2.422	4.522	5.946
Max	0.187	0.162	0.184	0.154	0.17	0.168	0.269	0.264
Min	-0.351	-0.296	-0.31	-0.248	-0.337	-0.287	-0.537	-0.47

The results are solely based on out-of-sample (OOS) forecasts to avoid any forward-looking bias.

3.1. Point forecast results

We measure the point forecast performance based on the mean absolute percentage error (MAPE) and the RMSE which are given by:

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=0}^{N}(y_t - \hat{y}_t)^2}, \tag{42}$$

$$MAPE = \frac{1}{N}\sum_{t=0}^{N}|y_t - \hat{y}_t| \tag{43}$$

where y_t is the observed value at time t, $\hat{y}_t$ the modelled value at time t and N the sample size. The advantage of MAPE against RMSE is that it allows to compare the point forecast accuracy across different time series as it is scale-independent. Nevertheless, reporting RMSE appears necessary in order to enhance the comparability with other contributions on the topic. The point forecast results are illustrated in Figures 2 to 5. First, it becomes evident that the naive strategy is equal or better than any wavelet filter in terms of MAPE and RMSE. This extends across every forecasting horizon and index. Notwithstanding the naive strategy, the wavelet filters $CF6$, $D4$ and $D6$ outperform every other filter, again, in terms of MAPE and RMSE. We confirm this observation by every forecasting horizon and index. It becomes also evident that the RFE algorithm has a positive effect on the MAPE and the RMSE. This holds for every forecasting horizon and index, which strongly emphasizes the need for a feature selection algorithm. As to the optimal forecasting horizon, the Figures 4 and 5 show that a forecasting horizon of five days delivers the best point forecasting results in terms of MAPE and RMSE. Nevertheless, wavelet transforms appear to be inappropriate for point forecasting as the naive approach outperforms each wavelet filter in terms of MAPE and RMSE.

3.2. Classification results

The forecasted sign of an asset return plays for several financial applications a more important role than its point forecast. For this reason we additionally asses the performance of our model for a binary classification task based on the confusion matrix. Consequently, returns greater than zero belong to class A and returns smaller than zero to class B. Subsequently, we evaluate the classification performance based on the accuracy and the Cohen's Kappa which are given by:

$$Accuracy = \frac{\sum \text{True Positive} + \sum \text{True Negative}}{\text{Total Population}} \tag{44}$$

$$Kappa = \frac{\text{Accuracy} - \text{Random Accuracy}}{1 - \text{Random Accuracy}}, \tag{45}$$

where RandomAccuracy = N^{-1}(Act.False × Pred.False + Act.True × Pred.True). In contrast to Accuracy, Cohen's Kappa provides a more reliable measure of agreement as it takes the amount of agreement that could be expected due to chance alone into account. The higher the Cohen's Kappa is, the better is our classifier compared to a random chance classifier. As we can see in Figure 6 and 7, the wavelet filters $CF6$, $D4$ and $D6$ slightly outperform the naive strategy in terms of Accuracy for most of the indices and forecasting horizons. Additionally, we observe that the RFE algorithm has positive impact on the Accuracy which is in accordance with the point forecast results. The Cohen's Kappa results (Figure 8 and 9) align with the Accuracy results. There is no clear-cut differentiation between the dominating wavelet filters and the naive strategy.

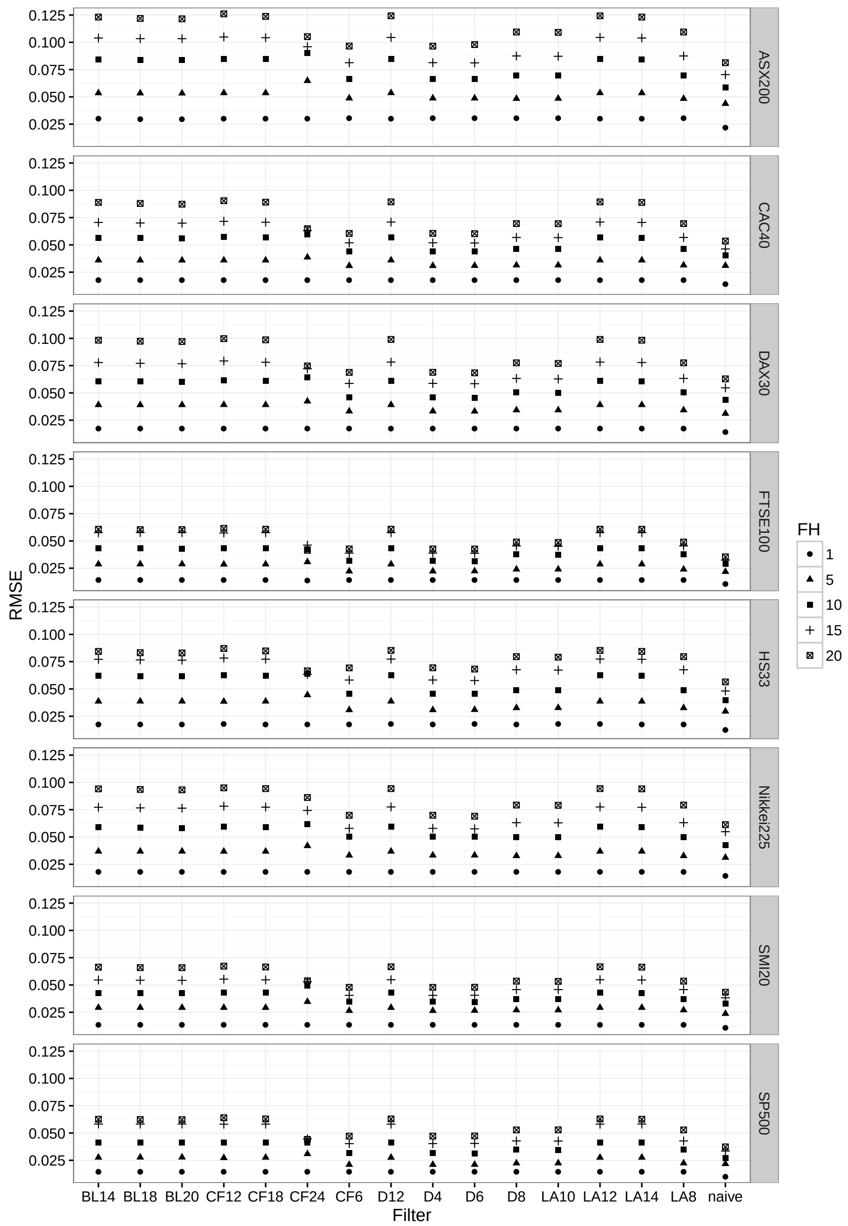

Figure 2. Point forecast performance of each wavelet filter including the naive method based on root mean squared error (RMSE), defined by Equation (42), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

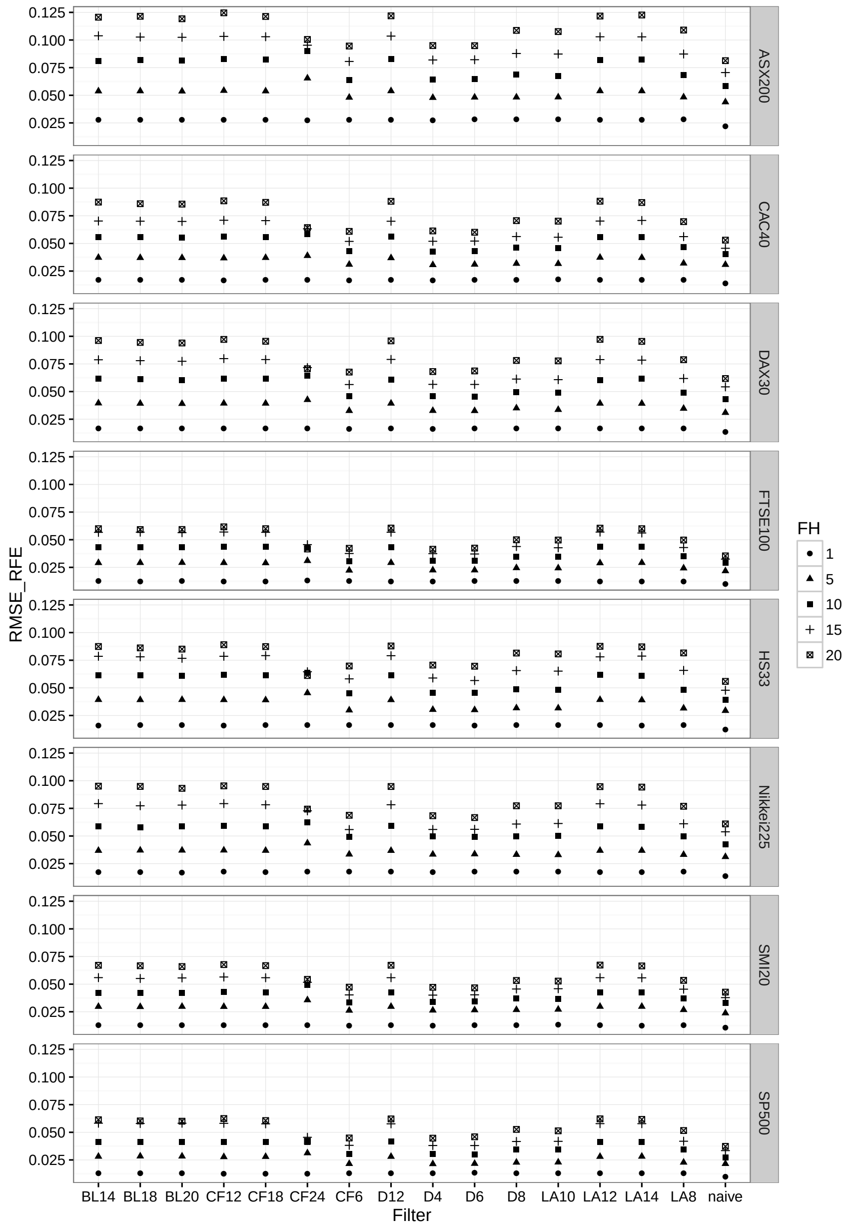

Figure 3. Point forecast performance of each wavelet filter including the naive method based on root mean squared error (RMSE), defined by Equation (42), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong (HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

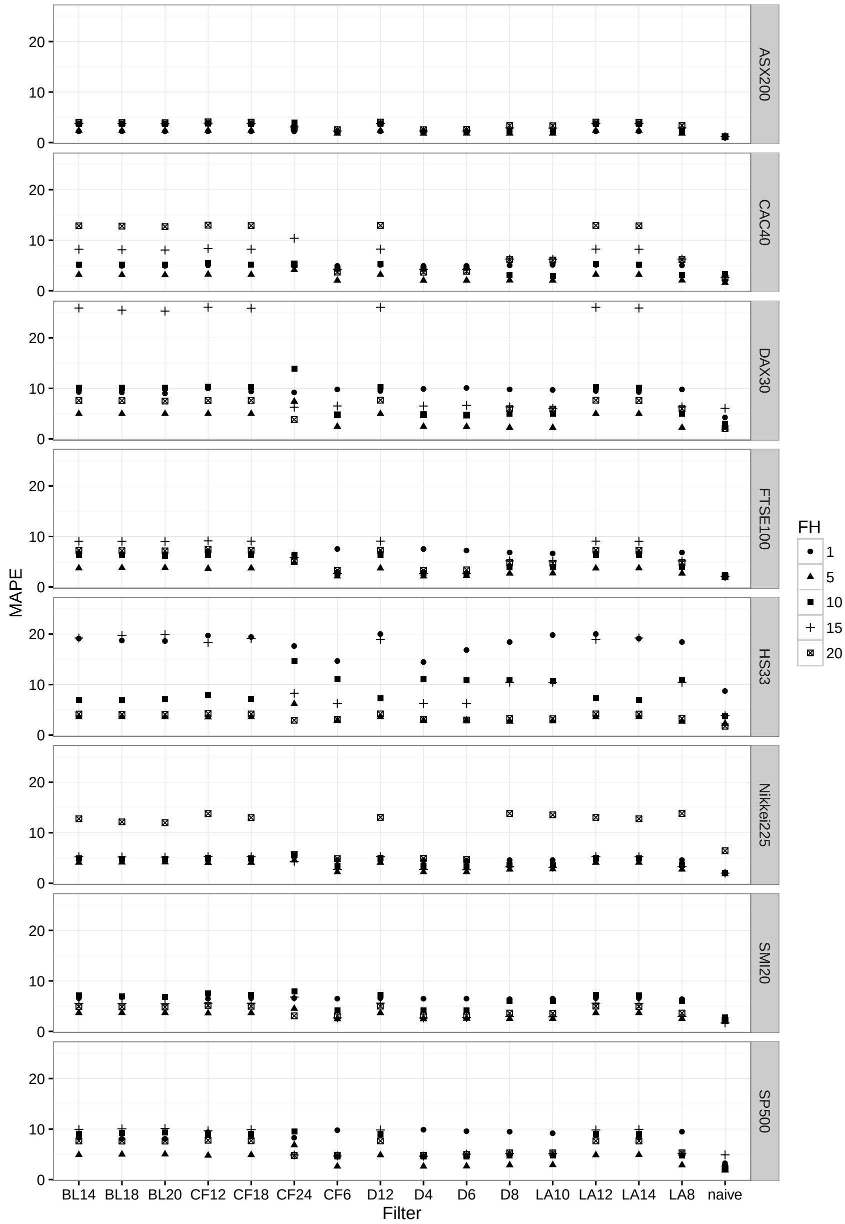

Figure 4. Point forecast performance of each wavelet filter including the naive method based on mean absolute percentage error (MAPE), defined by Equation (43), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

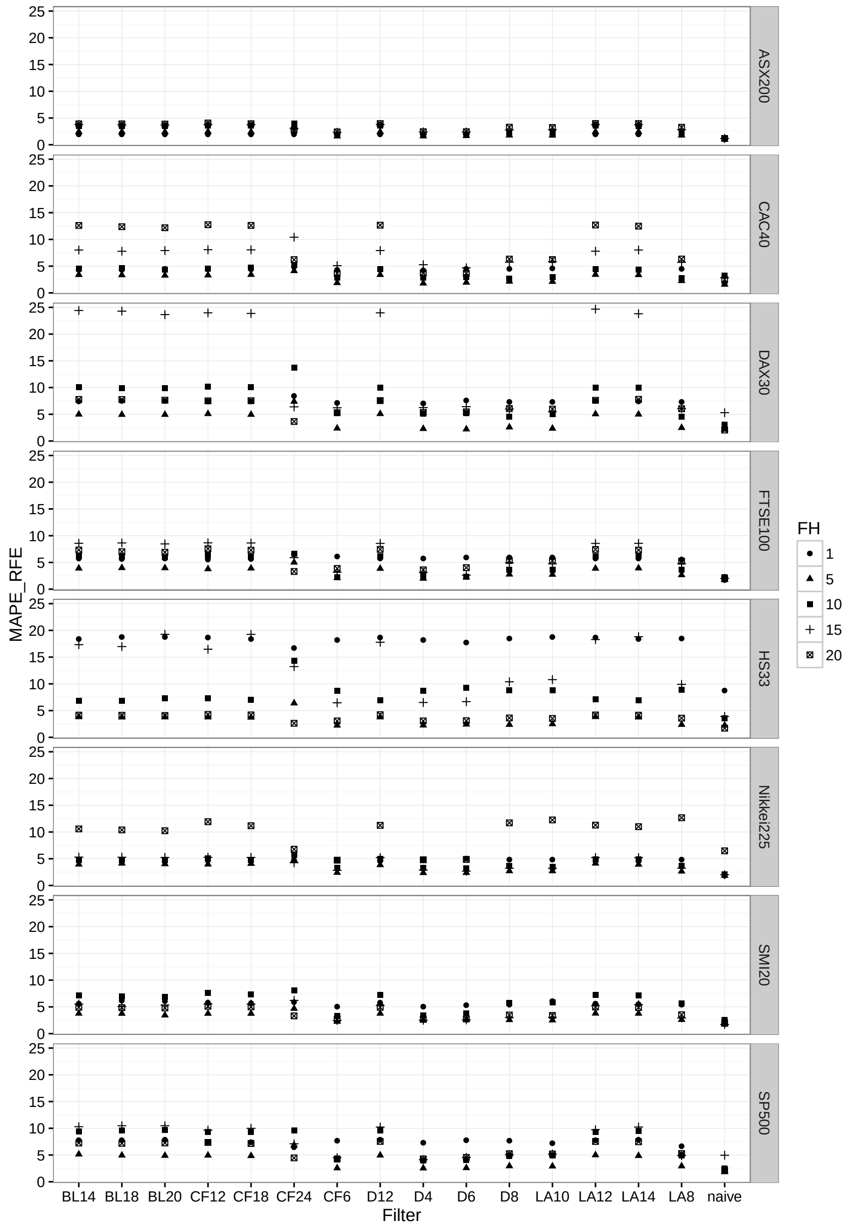

Figure 5. Point forecast performance of each wavelet filter including the naive method based on mean absolute percentage error (MAPE), defined by Equation (43), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong (HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

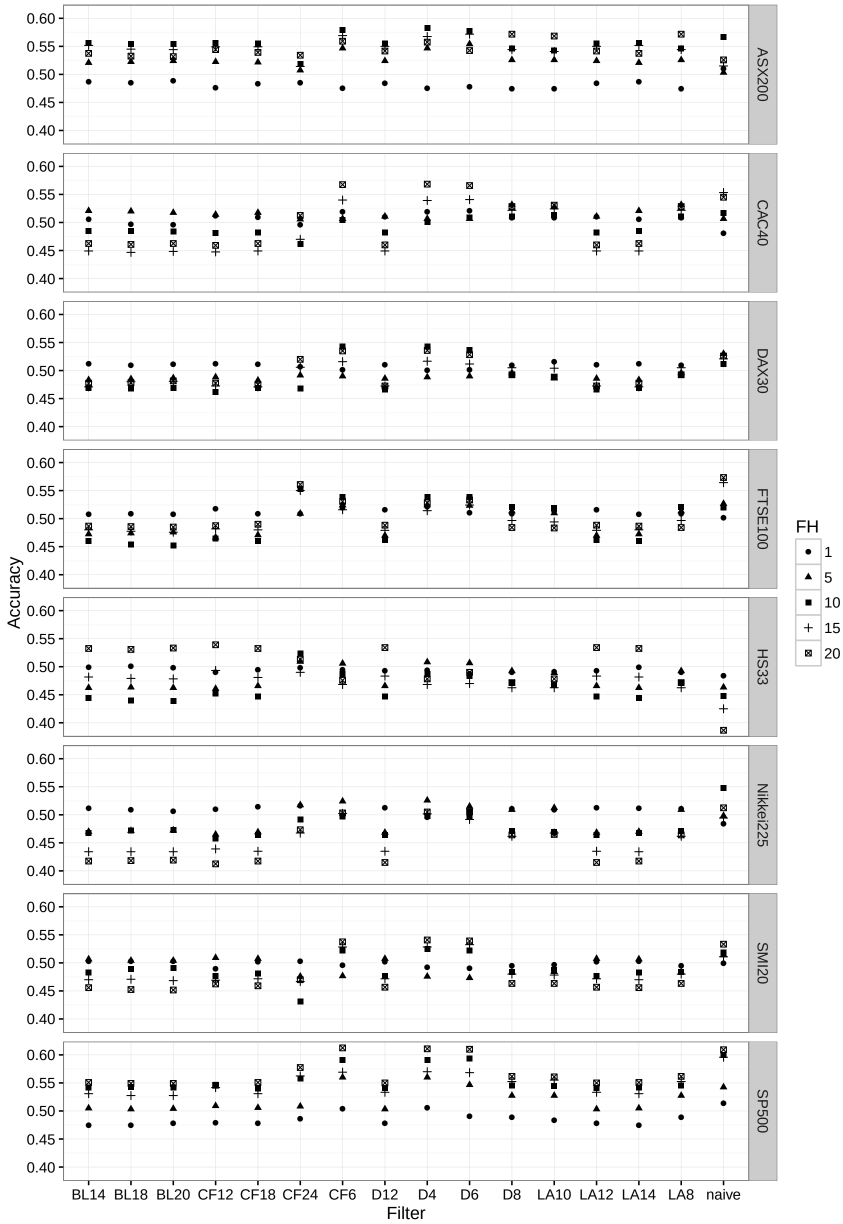

Figure 6. Classification forecast performance of each wavelet filter including the naive method based on Accuracy, defined by Equation (44), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

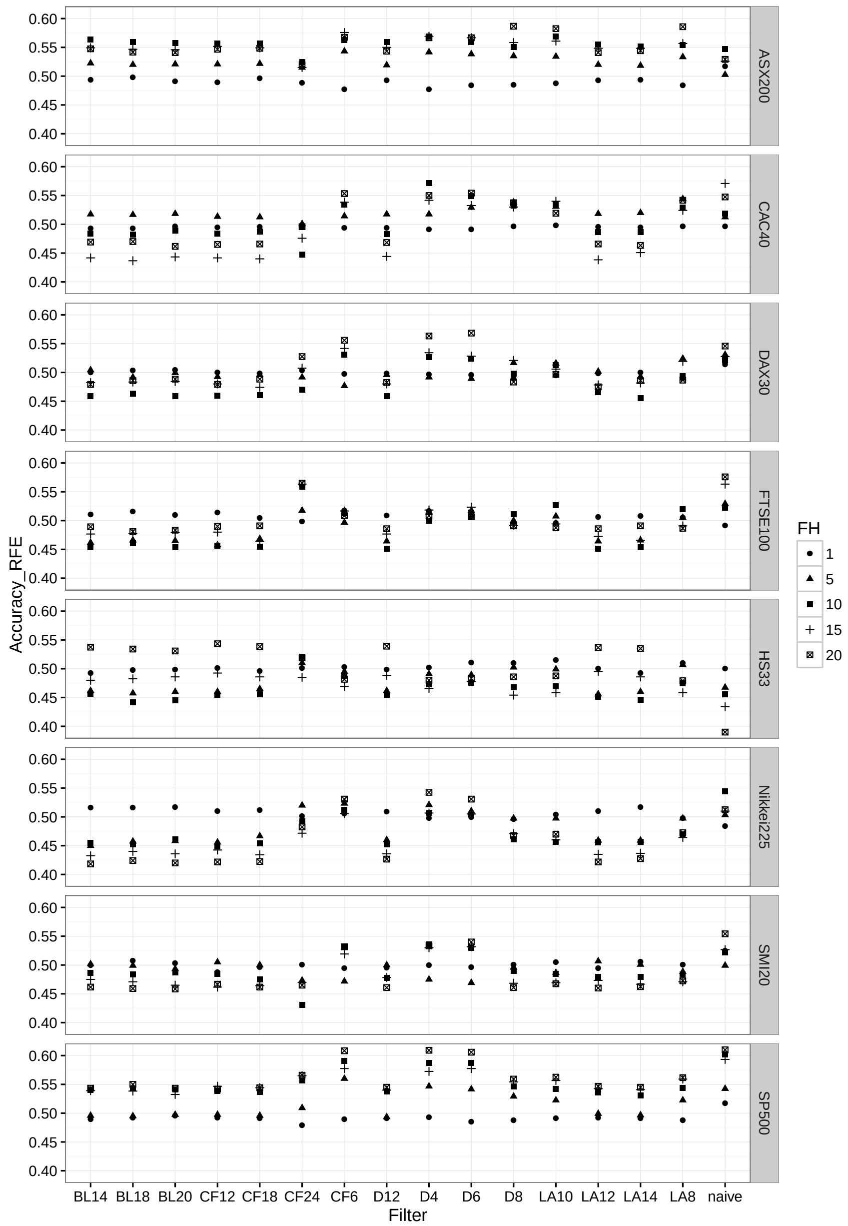

Figure 7. Classification forecast performance of each wavelet filter including the naive method based on Accuracy, defined by Equation (44), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

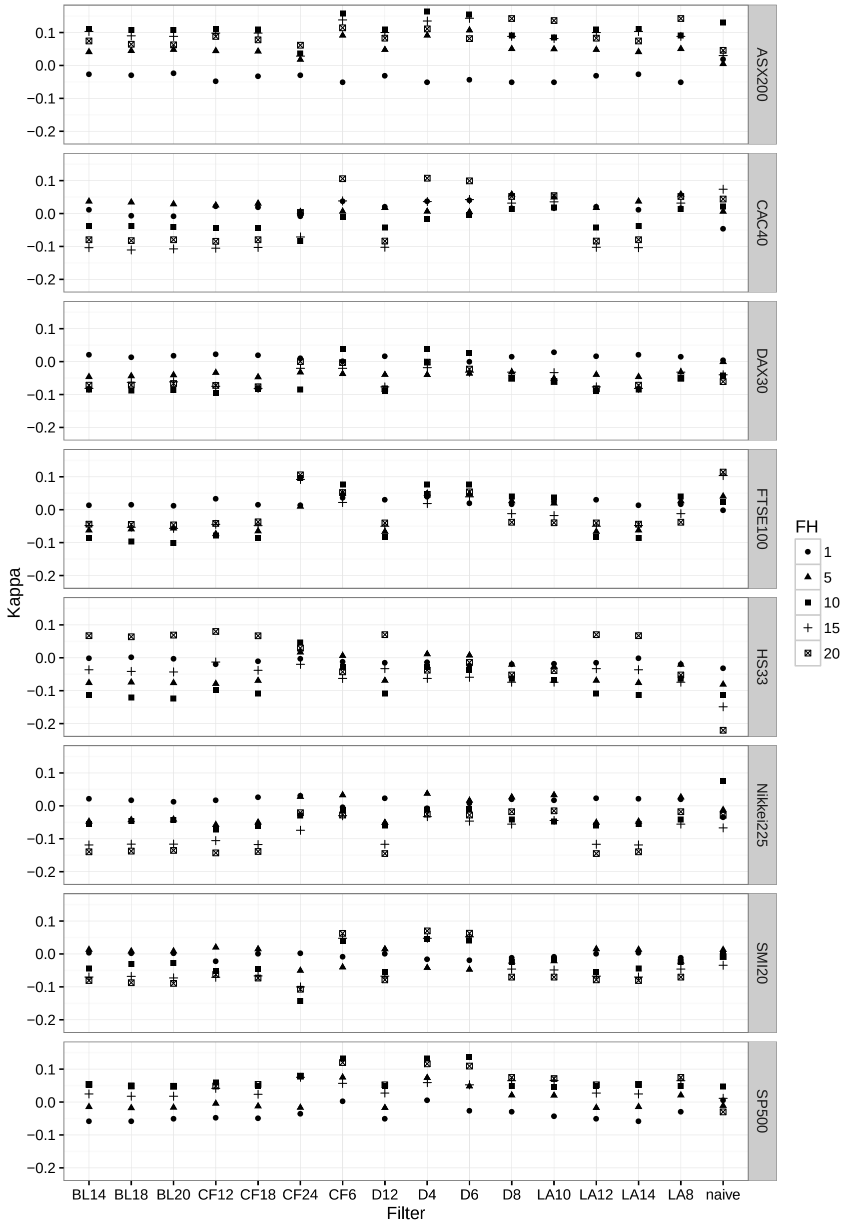

Figure 8. Classification forecast performance of each wavelet filter including the naive method based on Kappa, defined by Equation (45), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

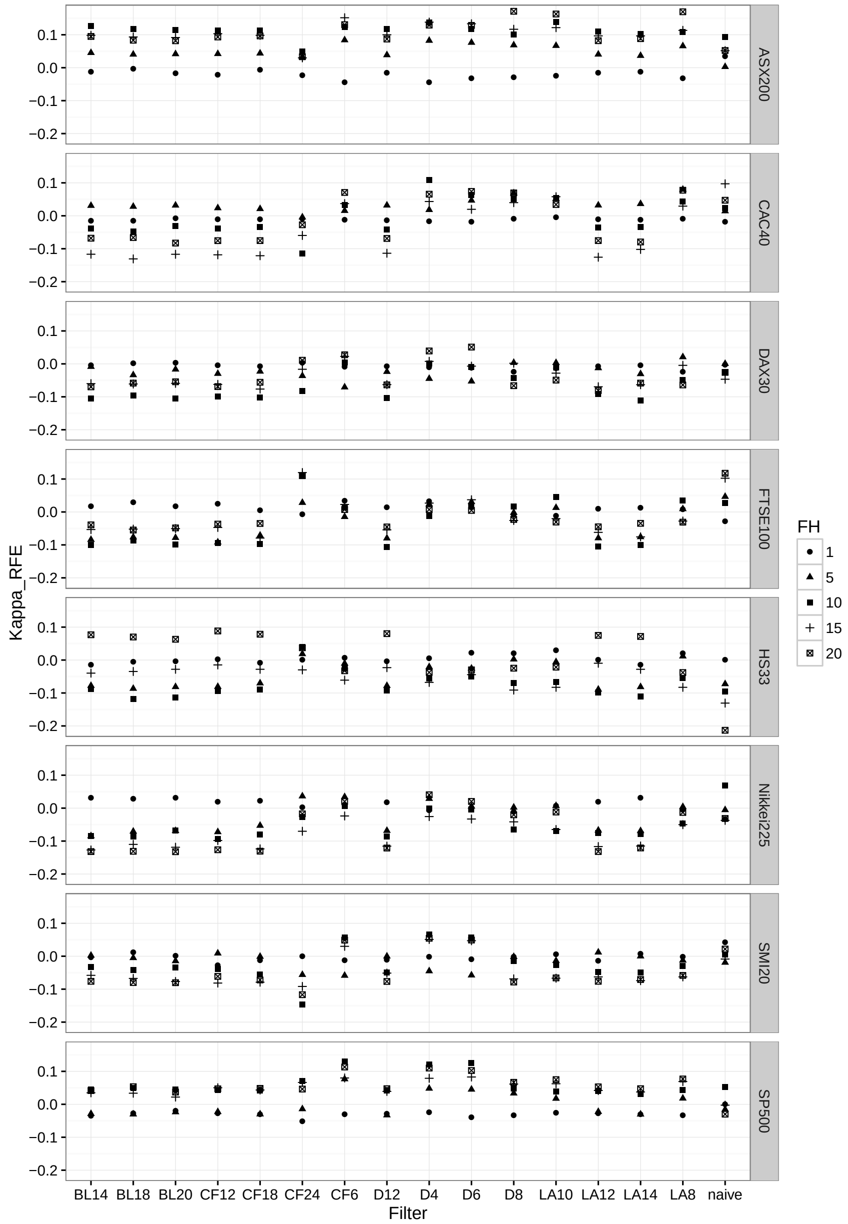

Figure 9. Classification forecast performance of each wavelet filter including the naive method based on Kappa, defined by Equation (45), for 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The considered stock indices are from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

3.3. Trading results

In practice, an ultimate goal of a forecasting algorithm in finance is to confirm its OOS profitability as statistical accuracy is not always synonymous with the financial profitability of the deriving foreceasts. Therefore, based on the above defined classification rule, we implement a most basic trading simulation in order to extend the results by a financial application and improve the comparability of the applied wavelet filters. We measure the profitability of our trading algorithm based on the following metrics:

Total return

$$tr = \sum_{t=0}^{T} r_t. \tag{46}$$

Annualized return

$$ar = 252\frac{1}{T}\sum_{t=1}^{T} r_t. \tag{47}$$

Sharpe Ratio

$$sr = \frac{\frac{\sum_{t=1}^{T} r_t}{T} - \frac{r_{fix}}{252}}{\sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(r_t - \bar{r})^2}} \tag{48}$$

Standard deviation

$$sd = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(r_t - \bar{r})^2} \tag{49}$$

Maximum Drawdown

$$mdd = max_{\tau\in(0,T)}\left[max_{\tau\in(0,T)}X(t) - X(\tau)\right] \tag{50}$$

where r_t denotes the realized return and $X(t)$ the realized equity at time t. The trading strategy is identical for all models and is defined as follows: enter a long position when the return forecast is greater zero and enter a short position when the return forecast is below zero. Trading costs have been considered to the extent that long positions are executed based on the ask quote and short positions based on the bid quote of the respective instrument. As shown in Figure 10 and 11, the wavelet filters $D4$, $D6$ and $CF6$ outperform every other considered wavelet filter including the naive strategy. In contrast to the point forecast and classification results we are able to make a clear distinction in the performance between the $D4$, $D6$ and $CF6$ filters and the remainder. The highest total return and annualized return is achieved for the forecasting horizon of 20 days. On the other hand, the increasing drawdown in forecasting horizon illustrates the increasing downside risk which comes along. This trade-off between return on investment and downside risk is present in all forecasting horizons. Again, the RFE algorithm has a positive effect on the overall return and maximum drawdown for the dominant wavelet filters.

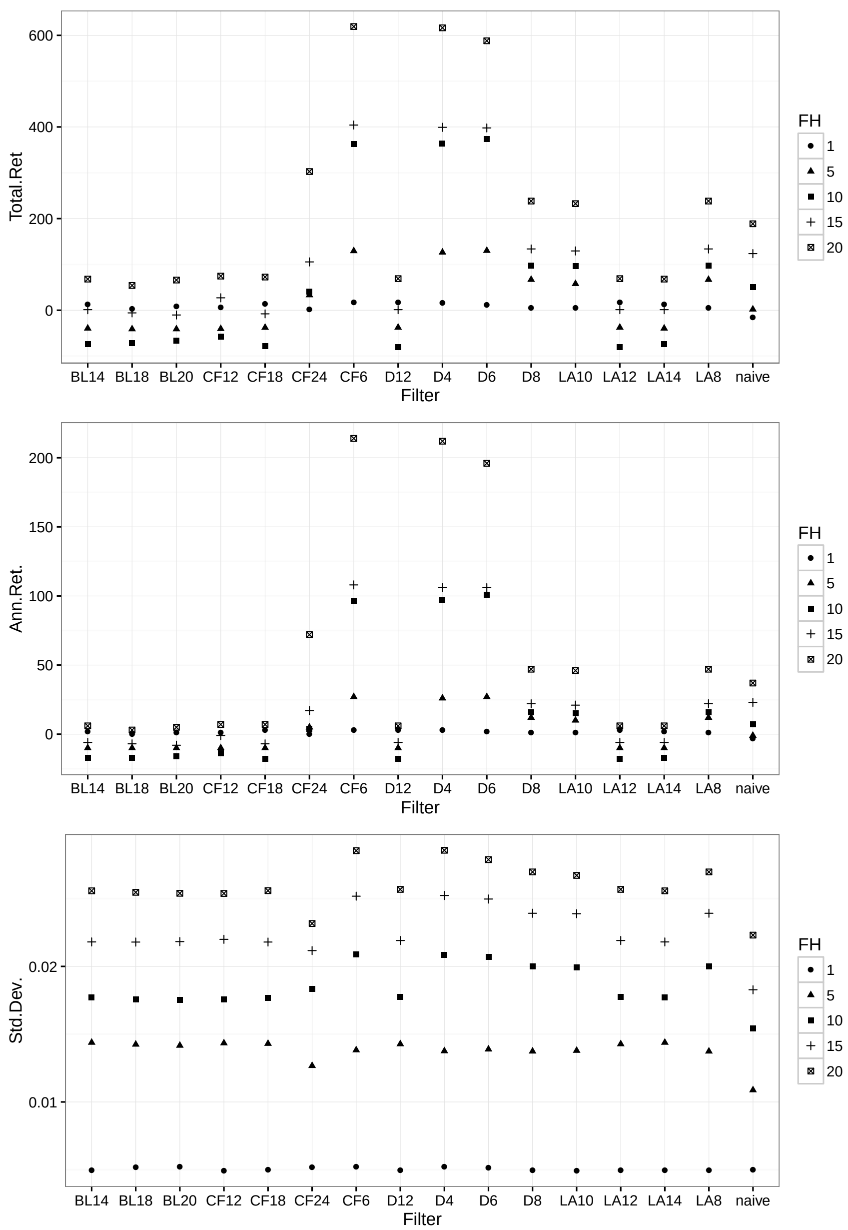

Figure 10. Trading performance in terms of Total Return (Equation (46)), Annual Return (Equation (47)) and Standard Deviation Equation (Equation 49) for the equally weighted portfolio consisting of the major stock indices from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The results are based on 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

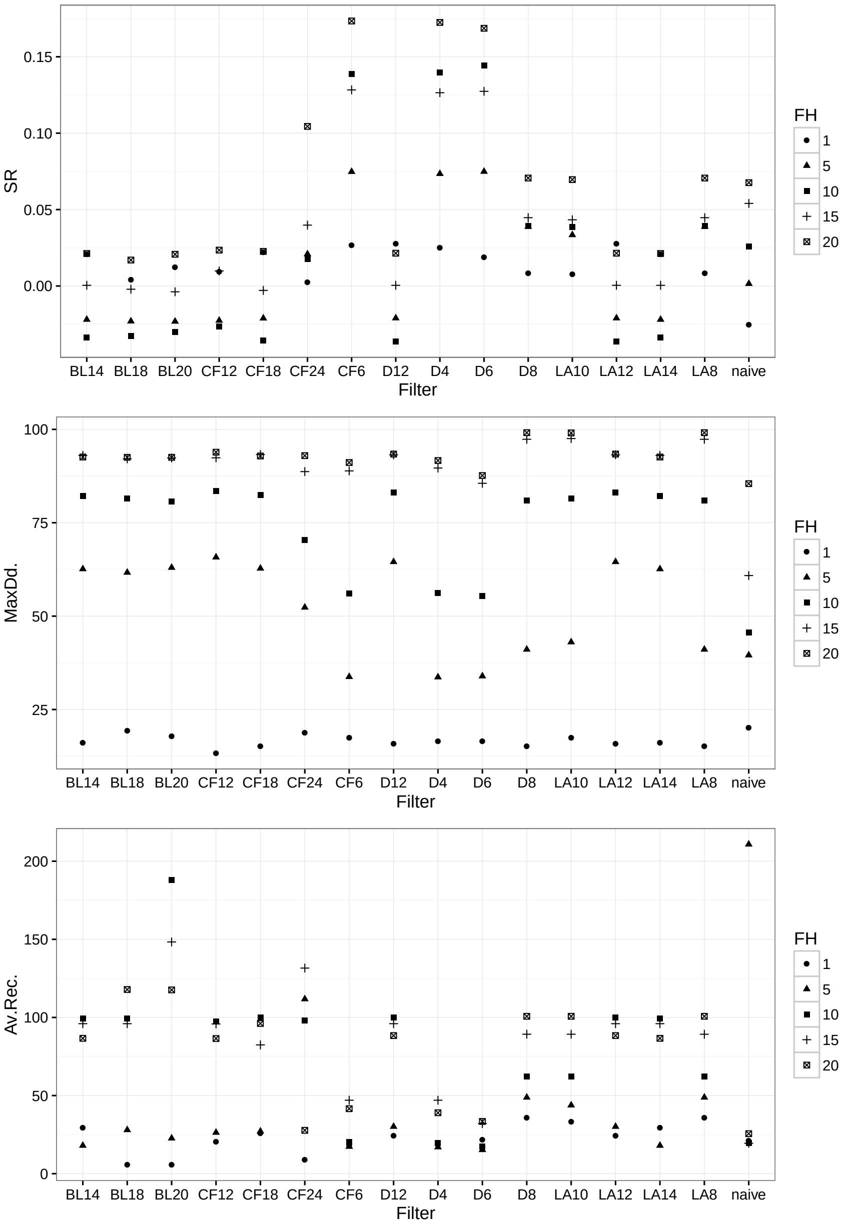

Figure 11. Trading performance in terms of Sharpe Ratio (Equation (48), Maximum Drawdown (Equation 50) and Average Recovery for the equally weighted portfolio consisting of the major stock indices from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). Average Recovery is defined as the mean duration of a equity drawdown period. The results are based on 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

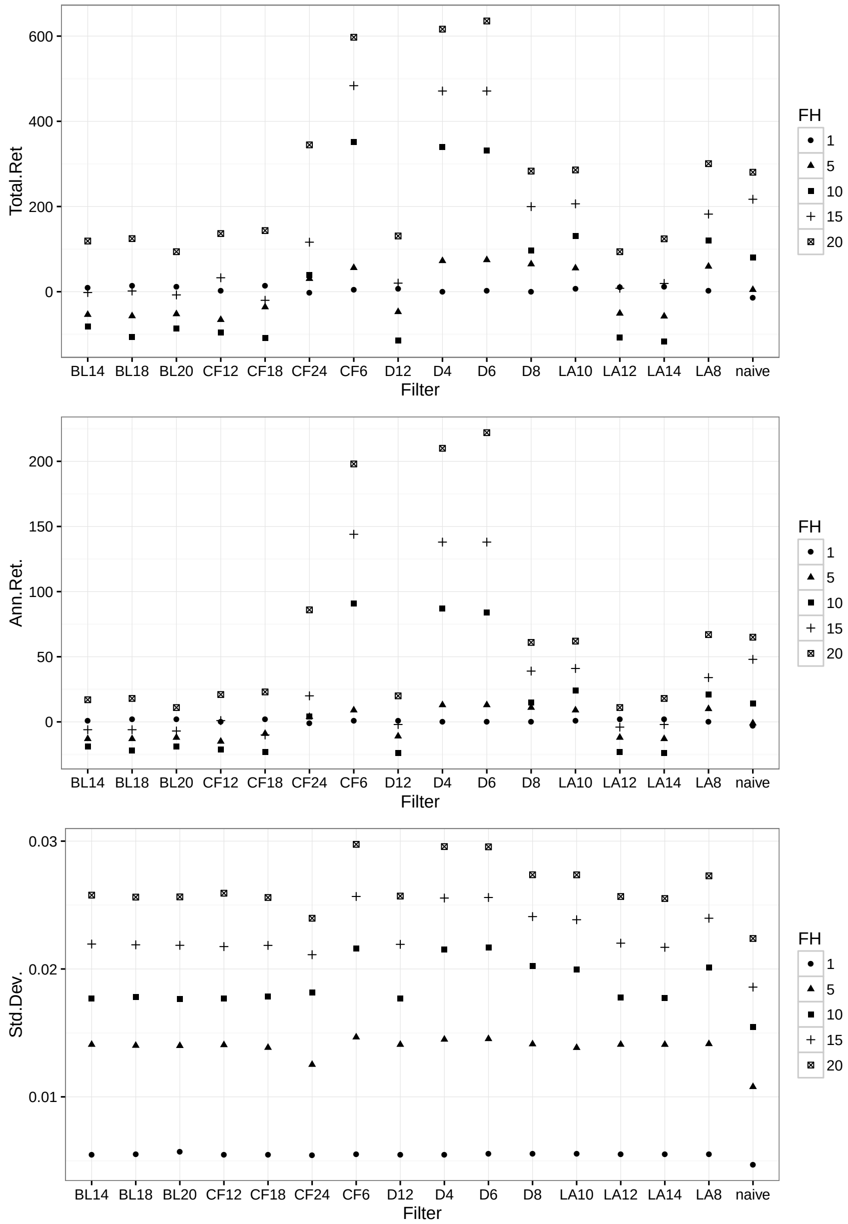

Figure 12. Trading performance in terms of Total Return (Equation (46)), Annual Return (Equation (47)) and Standard Deviation Equation (Equation 49) for the equally weighted portfolio consisting of the major stock indices from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). The results are based on 1200 out-of-sample forecasts for the period from 29.20.2010 to 26.06.2015. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

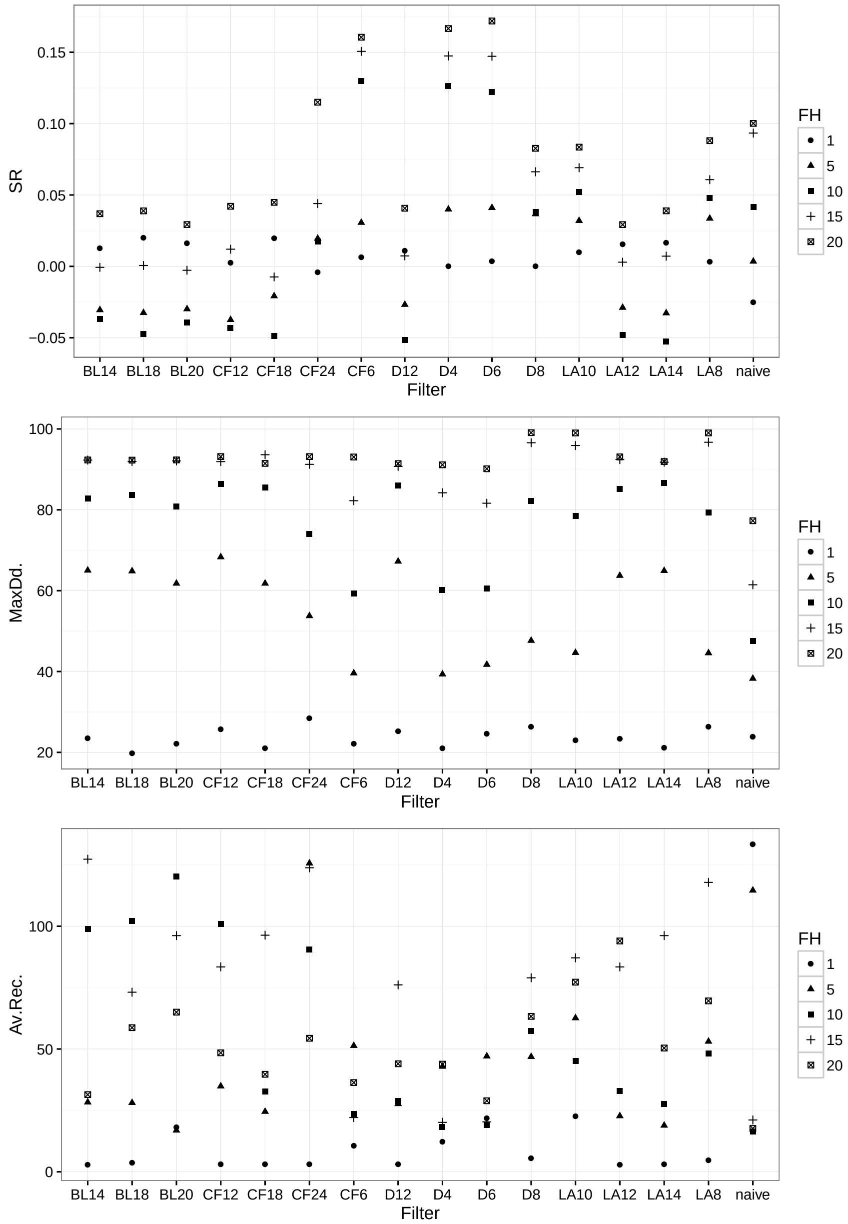

Figure 13. Trading performance in terms of Sharpe Ratio (Equation (48)), Maximum Drawdown (Equation (50)) and Average Recovery for the equally weighted portfolio consisting of the major stock indices from Germany (DAX30), United Kingdom (FTSE100) ,US (SP500), France (CAC40), UK (FTSE100), Switzerland (SMI20), Australia (ASX200), Japan (Nikkei225) and Hong Kong(HS33). Average Recovery is defined as the mean duration of a equity drawdown period. The feature set has been selected based on the results from the Recursive Filter Elimination algorithm, which is described in Section 2.2. The wavelet filters are denoted as Daubechies (D) (4,6,8,12), Least Asymetric (LA) (8,10,12,14), Best Localized (BL) (14,18,20) and Coiflet (CF) (6, 12, 18, 24), where the integers in brackets indicate the wavelet filter length. For the naive method the time series has not been preprocessed by a wavelet filter. As a result the input vector consists of 20 lagged returns. The forecasting horizon is denoted as FH and varies between +1, +5, +10, +15 and +20 days.

4. Conclusions

In this study we introduce a WL-RFE-SVR algorithm with an optimal parameter and feature selection which is applied to the task to the task of forecasting and trading multiple steps ahead the worlds major stock market indices. The proposed model is tested for the period 16/01/2007-22/11/2010 while its out-of-sample performance is validated over the period from 23/11/2010 through 31/12/2015 based on a walking forward analysis. The study shows that despite the adverse statistical properties of daily data, proper transformation and pre-processing can reveal reoccurring patterns that can be source for a trading algorithm. Even though we are not able to obtain an advantage over the naive strategy in the task of point forecasting, the classification and especially the trading results reveal a different picture. The positive results for three wavelet filter out of fifteen, namely $D4$, $D6$ or a $CF6$, show that the process of price formation in major stock indices is not completely governed by noise. Furthermore, results provide evidence to support the hypothesis of a serial dependence at least when the data is preprocessed by a $D4$, $D6$ or a $CF6$ wavelet filter. In the light of limited computational power, we are not able to exploit the full potential of the proposed hybrid method. Hence, the results can be considered as a lower bound on profits that can be achieved with an WL-RFE-SVR trading algorithm. A more sophisticated implementation of entry and exit rules and the testing of a wider/smaller range of inputs may increase forecasting and trading performance. The same applies to the utilized money management. An extension would include an adjustment of the size of the position depending on the strength of the trading signal or on volatility. Concerning some future directions of research, the proposed hybrid method could be applied to other forecasting problems in the domain of finance to prove its effectiveness. Additional research may be conducted towards the implementation of more refined feature selection algorithms and a deeper investigation of how wavelet filter influence the performance of the algorithm.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

Ambroise C, McLachlan GJ (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proceedings of the National Academy of Sciences of the United States of America* 99: 6562–6566.

Atsalakis GS, Valavanis KP (2009) Surveying stock market forecasting techniques - part 2: Soft computing methods. *Expert Syst Appl* 36: 5932–5941.

Chang PC, Fan CY (2008) A hybrid system integrating a wavelet and tsk fuzzy rules for stock price forecasting. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 38: 802–815.

Chang SI, Yadama S (2010) Statistical process control for monitoring non-linear profiles using wavelet filtering and b-spline approximation. *Int J Prod Res* 48: 1049–1068.

Cherkassky V, Ma Y (2004) Practical selection of svm parameters and noise estimation for svm regression. *Neural networks* 17: 113–126.

Daubechies I (1992) Ten Lectures on Wavelets, *Society for Industrial and Applied Mathematics*.

DeLurgio S (1998) Forecasting principles and applications, Boston: Irwin/McGraw-Hill.

Doroslovacki M (1998) On the least asymmetric wavelets. *IEEE T Signal Proces* 46: 1125–1130.

Gençay R, Selçuk F, Whitcher BJ (2001) An introduction to wavelets and other filtering methods in finance and economics. Academic press.

Huang SC, Wu TK (2008) Combining wavelet-based feature extractions with relevance vector machines for stock index forecasting. *Expert Syst* 25: 133–149.

Ince H, Trafalis TB (2008) Short term forecasting with support vector machines and application to stock price prediction. *Int J Gen Syst* 37: 677–687.

John GH, Kohavi R, Pfleger K, et al. (1994) Irrelevant features and the subset selection problem. In *Machine Learning: Proceedings of the Eleventh International Conference*, 121–129.

Kao LJ, Chiu CC, Lu CJ, et al. (2013) A hybrid approach by integrating wavelet-based feature extraction with MARS and SVR for stock index forecasting. *Decis Support Syst* 54: 1228–1244.

Kim KJ, Han I (2000) Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert syst Appl* 19: 125–132.

Li T, Li Q, Zhu S, et al. (2002) A survey on wavelet applications in data mining. *ACM SIGKDD Explorations Newsletter* 4: 49–68.

Lu CJ, Lee TS, Chiu CC (2009) Financial time series forecasting using independent component analysis and support vector regression. *Decis Support Syst* 47: 115–125.

Mattera D, Haykin S (1999) Support vector machines for dynamic reconstruction of a chaotic system. In *Advances in kernel methods*, MIT Press, 211–241.

Percival DB, Walden AT (2006) In *Wavelet methods for time series analysis*, Cambridge University Press.

Ramsey JB (2002) Wavelets in economics and finance: past and future. *Stud Nonlinear Dyn E* 6: 1090.

Ramsey JB, Lampart C (1998) The decomposition of economic relationships by time scale using wavelets: expenditure and income. *Stud Nonlinear Dyn E* 3: 49–71.

Saeys Y, Inza I, Larrañaga P (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics* 23: 2507–2517.

Schölkopf B, Smola A (2005) Support vector machines and kernel algorithms. In *Encyclopedia of Biostatistics 2. ed.*, 5328–5335.

Seo Y, Kim S, Kisi O, et al. (2015) Daily water level forecasting using wavelet decomposition and artificial intelligence techniques. *J Hydrol* 520: 224–243.

Svetnik V, Liaw A, Tong C, et al. (2004) Application of breimans random forest to modeling structure-activity relationships of pharmaceutical molecules. In *Multiple Classifier Systems*, 334–343.

Vapnik V (1999) An overview of statistical learning theory. *IEEE T Neural Network* 10: 988–999.

Vapnik V, Golowich SE, Smola A (1997) Support vector method for function approximation, regression estimation, and signal processing, In M. Mozer and M. Jordan and T. Petsche, *Advances in Neural Information Processing Systems*, MIT Press, 281–287.

Wang GJ, Xie C, Chen S (2017) Multiscale correlation networks analysis of the us stock market: a wavelet analysis. *J Econ Interact Coordination* 12: 561–594.

Zhang BL, Coggins R, Jabri MA, et al. (2001) Multiresolution forecasting for futures trading using wavelet decompositions. *IEEE T Neural Network* 12: 765–775.

Zheng G, Starck J, Campbell J, et al. (1999) Multiscale transforms for filtering financial data streams. *J Comput Intell Financ* 7: 18–35.