FX FORECASTING WITH HYBRID SUPPORT VECTOR MACHINES

ARAM ZINZALIAN, DEYAN SIMEONOV, AND ELINA ROBEVA

1. Introduction

Foreign exchange rates are notoriously difficult to predict. Deciding which of thousands of market signals to ignore and which to act on poses a challenging problem which potentially lends itself well to techniques in machine learning. Support Vector Machines have been shown to outperform a number of learning algorithms in FX prediction [2], [7]. The nonlinear predictive power and input size independent generalization error afforded by SVMs make them well suited to high dimensional classification problems [9]. In the context of FX forecasting, determining whether an FX rate will rise or fall after a certain number of days can be phrased as a binary classification problem where '1' denotes a rise, and '0' a fall. Previous research indicates that feature and kernel selection are key determinants of an SVM's predictive power [5], [2]. Of the learning algorithms applied to FX forecasting in the literature, hybrid combining models, such as Genetic Algorithm SVMs, achieve the best overall results [2]. Certain currency pairs, like AUS/USD have also been shown to favor certain kernels, e.g. the hyperbolic kernel, over others [2]. In investigating the generalizability of SVMs to forex forecasting we are interested in achieving the best accuracy with the least complexity. At the same time, we do not want to lose a lot of valuable information by developing an SVM off of a very restricted set of input variables. In order to balance these twin goals we thus seek an SVM that best captures the interaction between FX directional movement and market data in as complete as possible an input space. To that end, we propose a hybrid SVM combining a linear combination of Kernel functions with intelligent feature selection. Beyond generic forward/backward feature selection,we further implement a Genetic Algorithm SVM that explicitly defines a fitness function balancing the twin goals of reduced model complexity and low generalization error. We use the USD/CAD currency pair as a basis for evaluating our Hybrid SVM.

2. Features.

One of the main tasks in implementing a support vector machine algorithm for predicting foreign exchange rate movement, just as for every other purpose, is the choice of feature vectors. Our initial set of features consists of two different types of features: technical and non-technical features. The technical features refer to functions of the exchange rates in the past. Most of them are described below. These technical features are mostly used in machine learning algorithms for prediction of stock market indices movement rather than exchange rate movement [1], so it is interesting to see how they apply to our problem. On the other hand, non-technical features refer to the movement of prices of different commodities, other exchange rates, stock market indices, and other macro-economic variables that may be related to the exchange rate we are trying to predict.

Date: December 12, 2009.

2.1. **Technical Features.** Our set of technical features is mostly based on the technical features described in [1] which are used for stock market movement prediction. Each of them depends solely on the recent values of the exchange rate. Our output for day i is 1 if the exchange rate on day $i+5$ is grater than the exchange rate on day i. We do this as suggested by [8], who finds that under the same definition of closing stock prices the output data assumes more normal distribution than daily closing prices .
Assume that the price at time t is C_t.

Technical Feature Description	Formula
Stochastic $\%\mathbf{K_t}$	$\frac{\mathbf{C_t - L_t}}{\mathbf{H_t - L_t}}$,where L_t is the lowest price in the last 10 days.
Stochastic $\%\mathbf{D_t}$, moving average of $\%K$	$\frac{\sum_{\mathbf{i=0}}^{\mathbf{n-1}} \%\mathbf{K_{t-i}}}{\mathbf{n}}$.
Stochastic Slow $\%\mathbf{D_t}$, moving average of $\%D$	$\frac{\sum_{\mathbf{i=0}}^{\mathbf{n-1}} \%\mathbf{D_{t-i}}}{\mathbf{n}}$.
Momentum, price change in last 4 days	$\mathbf{C_t - C_{t-4}}$.
ROC, price rate of change	$\frac{\mathbf{C_t}}{\mathbf{C_{t-10}}}$.
$\mathbf{MA_5, MA_{10}}$, moving averages over the last 5 and 10 days	$\frac{\sum_{\mathbf{i=0}}^{\mathbf{4}} \mathbf{C_{t-i}}}{\mathbf{5}}, \frac{\sum_{\mathbf{i=0}}^{\mathbf{9}} \mathbf{C_{t-i}}}{\mathbf{10}}$.
$\mathbf{Disparity_{5,10}}$, distance of current price and the moving average of 5 or 10 days.	$\frac{\mathbf{C_t}}{\mathbf{MA_5}}, \frac{\mathbf{C_t}}{\mathbf{MA_{10}}}$.
OSCP, price oscillator; shows difference between 2 moving averages.	$\frac{\mathbf{MA_5 - MA_{10}}}{\mathbf{MA_5}}$.

2.2. **Non-technical Features.** When running out feature selection algorithms and SVM algorithm, we also used non-technical features. They included:

- USD/EUR, USD/JAP, and USD/AUS exchange rates
- US 3-month T-bill, US 10-Year Government Bond
- US S&P-500
- Wheat, Corn, and Gold futures prices
- Silver and Platinum Prices
- Canadian 3-month Libor, Canadian 3-month T-bill

This information dates from January 1, 2000 to November 11, 2009 and was obtained from www.globalfinancialdata.com .

Instead of considering these values alone, we also took "technical features" from these values, meaning, for each non-technical feature, if it's value at time t was V_t, then we also considered the functions from the above table applied to V_t instead of C_t.

Thus, the initial number of features(technical and non-technical combined) we had was 140.

3. Feature Selection

We tried two different types of feature selection. Both of them use the SVM algorithm.

3.1. **Genetic Algorithm Support Vector Machine (GASVM) Feature Selection.** [7] Cites very promising results for Genetic Algorithm Support Vector Machines (GASVM) applied to stock market prediction and foreign-exchange-rate forecasting. Our implementation of GA-Based feature selection broadly follows the algorithm described by the authors in [8]. Beyond their superior performance, relative to generic SVMs, GASVMs have a number of features that make them especially interesting for FX classification problems. GASVMs have been shown to find near optimal or optimal feature subsets within a reasonable run-time[8]. The random evolutionary process by which feature chromosones are generated also aligns with our preference for a "hands-off" feature selection algorithm, in so far as features are selected in a purely probabilistic manner.

On every iteration of the GA-algorithm, we assign each chromosone in our population a "fitness value," derived from a fitness function that rewards high prediction accuracy and penalizes model complexity. Yu, Wang, and Lai [8] suggest the following Kernel with $\beta = 0.3$.

$$f = \beta \text{RMSE}_{\text{training}} + (1-\beta)\text{RMSE}_{\text{testing}} - \alpha(1 - n_v/n_{\text{tot}}),$$

$$0 \leq \beta \leq 1$$

The higher α is the more we penalize large input spaces. Our GASVM algorithm performed best when we used an α value around 5, but we could not identify a clear relationship between the choice of α and the GASVM's prediction accuracy.

3.2. **Forward and Backward Feature Selection.** Then second, and more standard feature selection algorithm that we used was Forward and Backward Feature Selection. Since we had too many features to start with ($\tilde{1}40$), backward feature selection was considerably slower and it also proved less successful. Forward feature selection on the other hand gave us good results. We were able to select about 20 out of the 140 features so that the success rate of our prediction was around 65%.

4. Linear Combination of Kernels

Past research suggests that SVM prediction accuracy can vary dramatically with the choice of kernel. To address this problem we experimented with a linear combination of a Radial Basis Function (RBF) Kernel with Gamma Value 1 and a Linear Kernel. This turned out to be a successful attempt.

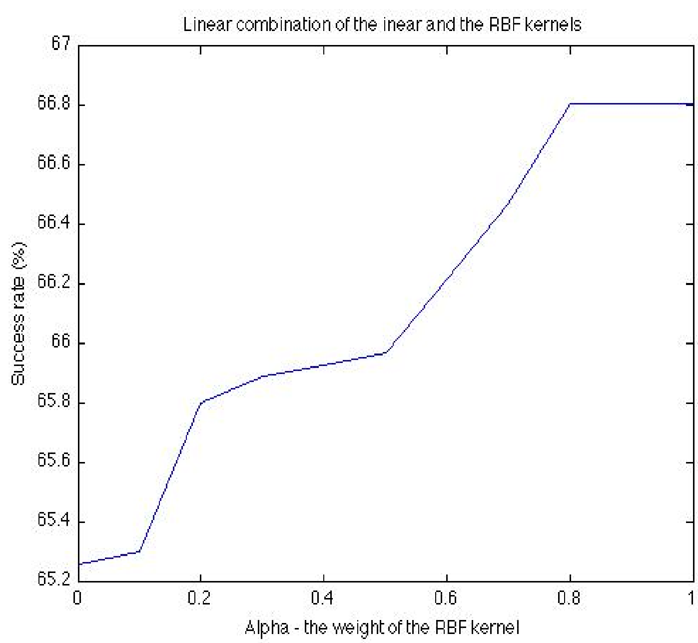
5. Results

One of the important things we noticed was that when one trains an SVM on training data consisting of prices in the past several years, one needs to shuffle the dates before choosing a training and a testing set. For example, during the period 2000 - 2003 the CAD/USD exchange rate was mostly increasing, so if we trained the SVM on that period, it wouldn't have much information of how days in which the rate is decreasing should look like. When we trained it on this, and tested it on the rest of the days, its best prediction was to say that the price always increases, so it would return a vector of 1's most of the time. The success rate on the test period: 2003 - 2009 was always around (mostly) under 60% no matter what Feature selection we used or what kernels we used. However, when we first shuffled the data and then we did feature selection and chose the training and the testing set, we obtained much better results.

After running a forward selection algorithm which selects at most 30 features out of the 140, our forward selection algorithm returned 25 features. Note that this algorithm also randomized the training and the testing data, so that the accuracy was better. Then we tried different types of kernels. The best result was the linear kernel, which gave us 67% accuracy, next was the RBF kernel which gave us 63% accuracy, and even worse was the polynomial kernel, which gave about 58% accuracy.

If we just use different kernels with no feature selection, the linear kernel gives 55% accuracy, the RBF kernel gives 60% accuracy. The RBF kernel gives 0 training error which is why we use a linear combination of the RBF kernel with the linear kernel. In order to find out what the best weights of the two kernels should be, we tried different linear combinations of the two kernels on the data with only technical features. We achieved pretty good results - our highest accuracy 67% was obtained when the weight of the RBF kernel was bigger (i.e. when it is between 0.7-1.0, which is easily seen on the figure below).

Moreover, when we tried using a feature selection algorithm on all 140 features and then plugging linear combinations of these two kernels, we achieved quite weak results - the best success rate was under 60%.



The GASVM feature selection algorithm that we ran used an RBF, $\sigma = 1$, kernel and produced features with 59% accuracy.

6. Conclusion

We achieve success rates of roughly 67% with either: only technical features (we have 18 such) and and SVM with a linear combination RBF and Linear Kernels; or forward feature selection on the whole feature set (which contains 140 features) and SVM with linear kernel. We were generally disappointed by the fact that the GASVM algorithm did not give better results than the other algorithms we tried. In order to possibly make the GASVM algorithm give better results, we could try modifying the fitness function.

We strongly believe that further testing of different combinations of kernels and selection algorithms could give even better results.

References

[1] K. Kim, *Financial time series forecasting using support vector machines*, Neurocomputing 55 (2003) 307 319

[2] C. Ulrich, D. Seese and S. Chalup, *Foreign Exchange Trading with Support Vector Machines*, Advances in Data Analysis, Berlin 2007

[3] W. Huang, Y. Nakamori, S. Wang, *Forecasting stock market movement direction with support vector machines*, Computers & Operations Research 32 (2005) 2513 2522

[4] L. Cao, F. Tay, *Application of support vector machines in financial time series forecasting*, Omega 29 (2001) 309317

[5] Y. Tan, J. Wang, *Support Vector Machine with a Hybrid Kernel and Minimal Vapnik-Chervonenkis Dimension*, Transactions on Knowledge and Data Engineering, Vol. 16, No. 4, April 2004

[6] J. Platt, *Fast Training of Support Vector Machines using Sequential Minimal Optimization*, Microsoft Research, Redmond, 1999

[7] Lean Yu, Shouyang Wang, Kin Keung Lai, pg. 171, *Foreign-exchange-rate forecasting with artificial neural networks*

[8] Lean Yu, Shouyang Wang, Kin Keung Lai, *Mining Stock Market Tendency Using GA-Based Support Vector Machines*

[9] Ping-Feng Pai, Chih-Shen Lin, Wei-Chiang Hong, Chen-Tung Chen, *A Hybrid Support Vector Machine Regression for Exchange Rate Prediction* , Information and Management Sciences Volume 17, Number 2, pp. 19-32, 2006