Improving Classification of Patterns in Ultra-High Frequency Time Series with Evolutionary Algorithms

Piotr Lipinski
Computational Intelligence Research Group
Institute of Computer Science
University of Wroclaw,
Wroclaw, Poland
piotr.lipinski@cs.uni.wroc.pl

Krzysztof Michalak
Department of Information Technologies
Institute of Business Informatics
Wroclaw University of Economics, Wroclaw, Poland
krzysztof.michalak@ue.wroc.pl

Adrian Lancucki
Computational Intelligence Research Group
Institute of Computer Science
University of Wroclaw,
Wroclaw, Poland
adrian.lancucki@cs.uni.wroc.pl

ABSTRACT

This paper proposes a method of distinguishing stock market states, classifying them based on price variations of securities, and using an evolutionary algorithm for improving the quality of classification. The data represents buy/sell order queues obtained from rebuild order book, given as price-volume pairs. In order to put more emphasis on certain features before the classifier is used, we use a weighting scheme, further optimized by an evolutionary algorithm.

1. INTRODUCTION

The aim of this paper is to leverage the knowledge acquired from stock market order books to classify market states into those for which we expect an abrupt change in stock price and those for which the price is expected to remain fairly stable. The use of order book data for this task is motivated by opinions of some economists [2], who suggest that important information, helpful to explain the stock market behavior such as liquidity, can be extracted from ultra-high frequency financial time series. Particular computational approaches confirm these opinions [3].

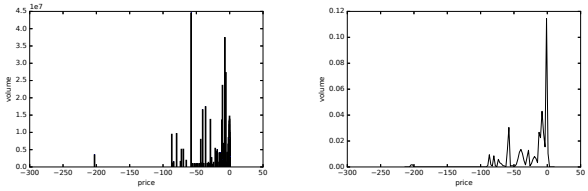
In this paper we use Support Vector Machine (SVM) [1] to perform classification based on order queue shapes. SVM is a well-established classification method used for, among others, classification problems involving class imbalance. Because we expect orders from different parts of the order book to be of different importance, we propose a method in which the Differential Evolution (DE) algorithm [4] is used for adjusting weights of features used for classification with the SVM.

2. ORDER BOOK CLASSIFICATION

Our approach to order book classification can be divided into several steps: a) conversion of the original order book data to a feature-based representation, b) application of a weighting scheme to adjust the importance of the features, c) training of the SVM classifier using weighted features, d) use of an evolutionary algorithm to adjust the parameters of the weighting scheme in order to improve the results of classification.

2.1 Data Representation

The data comes from the London Stock Exchange Rebuild Order Book (LSEROB) database and lists detailed orders placed on the market. Through reconstruction of order queues, we retrieve snapshots of buy/sell order queues at given times. Each snapshot is then represented by a Gaussian Mixture Model (GMM) with 50 components fitted with the Expectation-Maximization (EM) algorithm, as shown in Figure 1.



(a) (b)

Figure 1: Representation of a buy order queue of HSBC Holding at 10:00 am, 2/09/2013: a) initial buy order queue, b) a fitted GMM of 50 components

2.2 Optimization of Representation

Sampling the fitted GMMs, we represent each order book snapshot by two vectors $x^{buy}, x^{sell} \in \mathbb{R}^{1000}$. As orders the farthest from the mean buy/sell price have the lowest contribution to the overall market state, we introduce a weight vector $w \in \{0,1\}^{1000}$, which indicates how relevant are particular dimensions of snapshots. A weight vector is modelled by a Gaussian curve

$$w_i = f(i;\mu,\sigma),$$

$$\text{where} \quad f(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(x-\mu)^2}{2\sigma^2}).$$

Therefore, the representation optimization aims at finding two curves, one for the buy and one for the sell order queue,

GECCO'16 Companion July 20-24, 2016, Denver, CO, USA

ACM ISBN 978-1-4503-4323-7/16/07.

DOI: http://dx.doi.org/10.1145/2908961.2909042

defined in total by four parameters $\{(\mu_1, \sigma_1), (\mu_2, \sigma_2)\}$ with respect to a certain cost function.

2.3 Objective Function

Each candidate solution $\{(\mu_1, \sigma_1), (\mu_2, \sigma_2)\}$ is evaluated with the SVM classifier. The input data include all shapes from the training dataset weighted with Gaussians defined by the candidate solution. The target data consist of 1s for the cases with rises or falls of the mean price over an interval of 300 seconds after the occurence of the shape for more than a certain threshold $\theta = 0.15\%$, and 0s for the other cases. Then, SVM is trained and scored with F_1 measure, calculated on the obtained output. As the classifier's performance may depend on a particular run of the SVM training process, 10 SVM classifiers are trained, and the average value of their F_1 measure is taken as the objective function value of the candidate solution.

2.4 Evolutionary Algorithm

In order to optimize the order book representation, the DE algorithm is used to maximize the average F_1 measure of the 10 SVM classifiers trained on the order queues weighted with the particular Gaussian curves. The optimization problem is of a low dimensionality, but of a time-consuming objective function, so, the algorithm is run with a population of 10 individuals over 30 iterations, with the binomial crossover operator, and parameters set to $F = 0.75$ and $Cr = 0.25$.

3. VALIDATION OF THE APPROACH

The data comes from the LSEROB database in the form of complete order books, containing detailed information about all orders and trades of a particular financial instrument. The datasets portray quotes between the 1st and 15th of September 2013 of the 19 securities chosen from the FTSE100 index.

Table 1 presents the summary of results. We did not consider the accuracy of the classifier due to high class imbalance. Neutral order book snapshots, i.e. not presenting any abrupt changes, are by far the most frequent. Therefore, we compare the F_1 measure of the optimized classifiers with the baseline F_1 measure, which denotes a classifier which classifies all order books as neutral.

In all cases, the classifier outperformed the baseline on the training dataset. It means that the knowledge included in the order books as well as its representation enable to distinguish neutral and indicated order books. However, it might also be an effect of overfitting. Therefore, performance of the classifier was validated on the testing dataset. In most cases, the classifier outperformed the baseline, which confirms the relevance of the approach. For a few particular stocks, the classifier was unable to distinguish the indicated order books, probably due to a low threshold of decrease or increase of stock price, as the frequency of significant decreases and increases was relatively high.

4. CONCLUSIONS

This paper presents a method of distinguishing states of the stock market leading to abrupt changes in stock price. Stock market states are defined by order book shapes represented through GMM using the EM algorithm. The optimized classifier, based on SVM reinforced by optimization of the data representation with DE, allows to increase the probability of occurring selected events in the indicated stock market states.

Acknowledgment

Calculations have been carried out using resources provided by Wroclaw Centre for Networking and Supercomputing (http://wcss.pl), grant No. 405.

5. REFERENCES

[1] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[2] R. Engle, M. J. Fleming, E. Ghysels, and G. Nguyen. Liquidity and volatility in the us treasury market: Evidence from a new class of dynamic order book models. *Federal Reserve Bank of New York Working Paper*, 2011.

[3] P. Lipinski and A. Brabazon. Pattern mining in ultra-high frequency order books with self-organizing maps. In *Applications of Evolutionary Computation*, pages 288–298. Springer, 2014.

[4] R. Storn and K. Price. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

Table 1: Summary of results

Name	training dataset				testing dataset			
	precision	recall	F_1	F_1 base	precision	recall	F_1	F_1 base
HSBC HOLDINGS PLC	0.6507	0.4346	0.3030	0.2703	0.1017	0.5249	0.1704	0.1660
VODAFONE GROUP PLC	0.8476	0.5870	0.6881	0.1265	0.0375	0.8305	0.0718	0.0573
BP PLC	0.9980	0.4466	0.6169	0.0502	0.2214	0.2417	0.2311	0.1132
GLAXOSMITHKLINE PLC	0.8375	0.4005	0.5310	0.0966	0.0593	0.4375	0.1045	0.1343
ROYAL DUTCH SHELL PLC CLASS A	0.9247	0.2137	0.3303	0.1137	0.0200	0.3973	0.0381	0.0704
BRITISH AMERICAN TOBACCO PLC	0.8167	0.3538	0.4875	0.1123	1.0000	0.1970	0.3291	0.0639
ROYAL DUTCH SHELL PLC CLASS B	0.9506	0.4177	0.5778	0.0866	0.2063	0.2680	0.2332	0.0925
DIAGEO PLC	0.8096	0.3325	0.4641	0.1626	0.0665	0.4241	0.1150	0.1464
BG GROUP PLC	0.7876	0.4496	0.5399	0.1203	0.3761	0.5301	0.4400	0.1533
BHP BILLITON PLC	0.8871	0.4106	0.5604	0.1520	0.2238	0.7124	0.3406	0.3687
BARCLAYS PLC	0.8072	0.4270	0.5428	0.1348	0.4160	0.6420	0.5049	0.1499
RIO TINTO PLC	0.8141	0.5754	0.6728	0.1132	0.2665	0.5477	0.3586	0.1810
LLOYDS BANKING GROUP PLC	0.7592	0.4592	0.5225	0.1477	0.2559	0.4888	0.3359	0.1635
UNILEVER PLC	0.8248	0.3112	0.3992	0.2567	0.3700	0.2782	0.3176	0.1247
TESCO PLC	0.8854	0.3949	0.5433	0.0843	0.0429	0.6866	0.0808	0.0648
SABMILLER PLC	0.8130	0.3710	0.4795	0.0843	0.4192	0.4023	0.4106	0.1601
STANDARD CHARTERED PLC	0.6042	0.5857	0.4735	0.4054	0.2201	0.8908	0.3529	0.2055
RECKITT BENCKISER GROUP PLC	0.7370	0.3994	0.4314	0.2183	0.2267	0.3571	0.2773	0.2127
PRUDENTIAL PLC	0.7793	0.4023	0.5134	0.2175	0.1384	0.4127	0.2072	0.3299