2010 IEEE International Conference on Data Mining

Financial Forecasting with Gompertz Multiple Kernel Learning

Han Qin
Computer and Information Science
University of Oregon
Eugene, OR 97403, USA
qinhan@cs.uoregon.edu

Dejing Dou
Computer and Information Science
University of Oregon
Eugene, OR 97403, USA
dou@cs.uoregon.edu

Yue Fang
Department of Decision Science
University of Oregon
Eugene, OR 97403, USA
yfang@lcbmail.uoregon.edu

***Abstract*—Financial forecasting is the basis for budgeting activities and estimating future financing needs. Applying machine learning and data mining models to financial forecasting is both effective and efficient. Among different kinds of machine learning models, kernel methods are well accepted since they are more robust and accurate than traditional models, such as neural networks. However, learning from multiple data sources is still one of the main challenges in the financial forecasting area. In this paper, we focus on applying the multiple kernel learning models to the multiple major international stock indexes. Our experiment results indicate that applying multiple kernel learning to the financial forecasting problem suffers from both the short training period problem and non-stationary problem. Therefore we propose a novel multiple kernel learning model to address the challenge by introducing the Gompertz model and considering a non-linear combination of different kernel matrices. The experiment results show that our Gompertz multiple kernel learning model addresses the challenges and achieves better performance than the original multiple kernel learning model and single SVM models.**

***Keywords*-financial forecasting; multiple kernel learning; non-linear kernel combination;**

I. Introduction

Financial forecasting is the basis for budgeting activities and estimating future financing needs [13]. For example, *volatility forecasting* attempts to predict the *risk* of financial markets by examining the historic time series data. Traditional time series models are employed to directly model the financial data. When exploring a large number of time series, the traditional models become impractical [9]. Hence machine learning and data mining models have been applied to financial forecasting for recent years, such as Artificial Neural Networks (ANN) and Support Vector Machine [4] (SVM). Among these models kernel-based methods are most robust and accurate.

In spite of extensive SVM applications in financial forecasting, the SVM models do not address the challenge of learning from multiple data sources. Nowadays, world wide financial markets are not independent under the global economic system [5] while the relationships between the international markets become tighter due to the globalization. This challenge is known as international market integration problem [11], which is one of the important challenges of financial forecasting.

Motivated by kernel-based learning from multiple data sources or models, Multiple Kernel Learning [8] (MKL) was proposed in 2004. MKL considers the linear combination of initial kernels. The initial kernels could be from single or multiple data sources. On the one hand, MKL solves the convex optimization problem of linear combination of single kernels and is guaranteed to achieve global optimal, hence the MKL models theoretically have better performance than SVM. On the other hand, MKL explicitly learns the weights of kernels from different data sources and the relationships among them are learned in the meanwhile.

Due to the above advantages, applying MKL to financial forecasting with multiple data sources is promising. Based on the convex optimization theory, MKL should have better accuracy than applying SVM with any single data source [8]. However, the original MKL models make the assumption that training data and test data have the same distribution while the financial data are *non-stationary*, which means the underlying distribution of the data keeps changing [10]. Our experiment results also indicate that MKL does not necessarily achieve better accuracy than SVM.

Therefore, in this paper we propose a novel MKL model to financial forecasting with multiple data sources and also address the non-stationary problem. Inspired by Localized Multiple Kernel Learning (LMKL) [7], we propose an extended MKL model which combines kernel matrices in a non-linear way by introducing the Gompertz model [20], which is one of mathematical models developed especially for time series. Our Gompertz Multiple Kernel Learning (GMKL) model considers the order of time series data and assigns higher weights to the recent instances. We focus on forecasting the volatility of the stock markets, which means the risk of each trade period. Thus we define a domain specific kernel function based on a well known financial volatility model (Generalized AutoRegressive Conditional Heteroskedasticity [2]). The experiment results show that our extended MKL model achieves better forecasting accuracy than original MKL and single SVM models when applied to five major international stock indexes.

1550-4786/10 $26.00 © 2010 IEEE
DOI 10.1109/ICDM.2010.68

IEEE computer society

II. Related Work

To extend kernel methods to learn from multiple kernels functions or data sources, Lanckriet *et al.* [8] first proposed multiple kernel learning (MKL) by considering the linear combination of initial *basis* kernels and used a semidefinite programming (SDP) framework to solve the problem. Bash *et al.* [1] proposed the problem of support kernel machine as a second order conic programming (SOCP) problem and cast the multi kernel learning to a SOCP problem. They also developed a sequential minimal optimization (SMO) based algorithm to solve the problem more efficiently.

In 2006, Sonnenburg *et al.* [14] proposed an efficient algorithm to solve the multiple kernel learning problem by formulating the original problem as a semi-infinite linear program (SILP). Rakotomamonjy, Bash *et al.* [12] showed that the SILP approach is exactly equivalent to cutting plane methods and thus suffers from the problem of slow converging due to the fact that cutting planes methods are known for their instability. Therefore they proposed a subgradient descent (SD) approach and addressed the MKL problem through an adaptive 2-norm regularization formulation.

Gonen and Alpaydm [7] elaborated that assigning different weights to a kernel in different regions of the input space will produce a better classifier and proposed their localized multiple kernel learning (LMKL) to learn the weights of regions while treating all kernels equally. Varma and Babu [17] proposed a general multiple kernel learning algorithm that not only considers the linear combination of base kernels but also employs more prior knowledge. Ye *et al.* [22] proposed a QCQP formulation for kernel learning to address both bi-class problems and multi-class problems.

Multiple kernel learning techniques have been applied to a variety of domains. Fu *et al.* [6] applied MKL to image matching with a novel histogram intersection kernel. Ye *et al.* [21] integrated multiple biomedical data sources for Effective diagnosis of Alzheimer's disease. Vedaldi *et al.* [18] showed that MKL classifiers can be successfully used for image object detection. Suard *et al.* [15] presented a pedestrian detection method based on the MKL framework. Barrington *et al.* [16] combined a number of music information retrieval features with the MKL method for the semantic music discovery problem. Wang and Zhu [19] applied a two-step MKL method to predict the buy and sell signals in stock markets with a validation problem determining the regularization parameter.

III. Gompertz-based Multiple Kernel Learning

To address the non-stationary problem of financial data, we propose a novel Gompertz Multiple Kernel Learning (GMKL) model with a domain-specific kernel. In this section, we first introduce the design of the GARCH-based kernel, then describe our model and formulate GMKL to an optimization problem and describe a subgradient algorithm to solve the problem.

A. Design of Domain-specific Kernel

Choosing proper kernels for the special application is the key point of kernel-based methods. We hence develop a GARCH-based kernel to leverage the domain knowledge.

GARCH models are employed commonly in modeling financial time series that exhibit time-varying volatility changing. The return $\mathcal{R}_t$ is defined as $\mathcal{R}_t = lg(\frac{P_t}{P_{t-1}})$, where P_t is the stock price of day t. Let assume that $R_t = \sigma_t \epsilon_t$, where ϵ_ts are independently and identically distributed random variables . And the series σ_t^2 are modeled by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i R_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2$$

where α and β are coefficients.

We define a domain specific kernel, GARCH Kernel, based on the GARCH model. The GARCH kernel function takes the σ^2 and R^2 values of the GARCH model as input. The kernel function based on the GARCH model with parameters (p,q) is defined as $k(day\_1, day\_2) = \Sigma\sigma_{1i}^2 * \sigma_{2i}^2 + \Sigma R_{1j}^2 * R_{2j}^2$ where $i = 1 \ldots p$ and $j = 1 \ldots q$. The kernels can be seen as a mapping from the original price space to higher dimensional space.

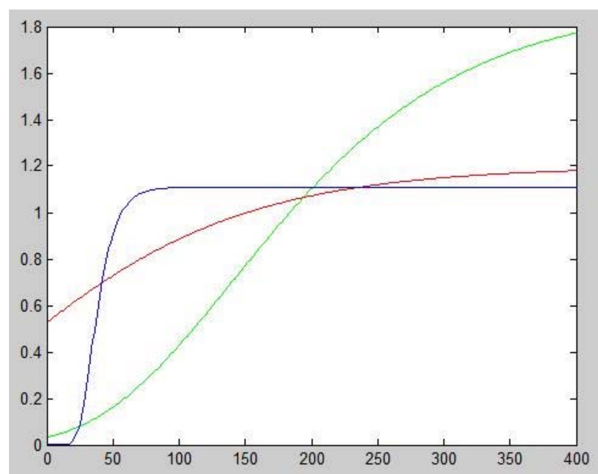

Figure 1. Gompertz model with different sets of parameters

B. The Gompertz Model

The Gompertz model or Gompertz function [3] is one of mathematical models developed for time series. The Gompertz model was originally developed for analyzing reliability growth but it is most applicable when the data set follows a smooth curve. The growth rate of Gompertz model is slowest at the start and the end of a time period.

The formula of Gompertz model is:

$$y(x) = ue^{ve^{wx}}$$

where u is the upper asymptote, v sets the x displacement, w is the growth rates and e is the Euler's number. Note that both v and w should be negative numbers.

Because he Gompertz function is not symmetrical, it is more general than the Sigmoid function. This asymmetry enable us to capture the non-stationarity of financial data. Figure 1 gives three examples of Gompertz function with different sets of parameters.

In this paper, we apply Gompertz model to approximate the weights of different training instances, assigning higher weights to recent instances. We consider the index of a data instance in time order as x. Parameters u, v and w will be learned in the training process, so that the optimal y will be used as weights.

C. Gompertz Multiple Kernel Learning

To apply the Gompertz function, we propose the Gompertz Multiple Kernel Learning (GMKL) model which assigns different weights to training data instances under the multiple kernel learning framework. GMKL actually combines the original kernel matrices in a non-linear way. This understanding is inspired by Localized MKL [7].

Similar as Multiple Kernel Learning, GMKL calculates the summation of inner products of weight w and input vector x in multiple high dimensional spaces. Nevertheless, the weight w is learned with weighted data instances. The following is the function of GMKL:

$$f(x) = \sum_{j=1}^{m} < \mathbf{w_j}, \Phi_j(\mathbf{x}) > +b$$

where $\mathbf{w_j} = \sum_{i=1}^{n} G_j(x_{iindex})\alpha_i x_i$ while Gompertz function $G_j(x_{in} = ue^{ve^{wx_{in}}})$ considers the ordering of training data instances from the jth data source.

The above function is equal to the output function of MKL when $\forall x_{index} G_j(x_{index}) = 1$. Therefore the search space of GMKL is more general than MKL. By penalizing the training errors with the ϵ-insensitive loss function, the optimization problem of the GMKL model is:

$$\begin{aligned}
\text{minimize } & \tfrac{1}{2}\sum_{j=1}^{m} \|\mathbf{w}_m\|^2 + C\sum_{i=1}^{l}(\varepsilon_i + \hat{\varepsilon}_i) \\
\text{subject to } & G_m(x_i) < \mathbf{w}_m \cdot \Phi(\mathbf{x_i}) > +b - y_i \leq \varepsilon - \varepsilon_i \\
& y_i - G_m(x_i)(< \mathbf{w}_m \cdot \Phi(\mathbf{x_i}) > +b) \leq \varepsilon - \hat{\varepsilon}_i, \\
& \varepsilon_i \geq 0,\ \hat{\varepsilon}_i \geq 0, \sum_{j=1}^{m} G_j(w) = 1
\end{aligned}$$

The dual form of the above problem is:

$$\begin{aligned}
\text{maximize } & -\tfrac{1}{2}\sum_{i,j=1}^{l}(\hat{\alpha_i} - \alpha_i)(\hat{\alpha_j} - \alpha_j)K_G(\mathbf{x_i}, \mathbf{x_j}) \\
& + \sum_{i=1}^{l} y_i(\hat{\alpha_i} - \alpha_i) - \sum_{i=1}^{l}\varepsilon(\hat{\alpha_i} + \alpha_i) \\
\text{subject to } & \sum_{i=1}^{l}(\alpha_i - \hat{\alpha_i}) = 0,\ 0 \leq \alpha_i \leq C, \\
& 0 \leq \hat{\alpha_i} \leq C, \sum_{j=1}^{m}\sum_{i=1}^{n} G_j(x_i) = n
\end{aligned}$$

where the combined kernel matrix is defined as:

$$K_G(\mathbf{x_i}, \mathbf{x_j}) = \sum_{j=1}^{m} G_j(x_i)K_j(x_i, x_j)G_i(x_j)$$

D. Subgradient Descent Algorithm

The subgradient-descent learning algorithms are commonly used for solving the optimization problems. Therefore, we propose a subgradient-descent based algorithm to learn the parameters of Gompertz function.

We consider the objective value of the dual optimization problem as Γ and use gradient-descent to get the derivation of Γ over the parameters v and w of the Gompertz function:

$$\begin{aligned}
\tfrac{\partial \Gamma}{\partial v} &= -\tfrac{1}{2}\sum_{i,j=1}^{l}(\hat{\alpha_i} - \alpha_i)(\hat{\alpha_j} - \alpha_j)K_j(x_i, x_j)u(e^{ve^{wi}}ve^{2wi} \\
&\quad + e^{ve^{wj}}ve^{2wj}) \\
\tfrac{\partial \Gamma}{\partial w} &= -\tfrac{1}{2}\sum_{i,j=1}^{l}(\hat{\alpha_i} - \alpha_i)(\hat{\alpha_j} - \alpha_j)K_j(x_i, x_j)u \\
&\quad (e^{ve^{wi}}ve^{2wi}i + e^{ve^{wj}}ve^{2wj}j)
\end{aligned}$$

Algorithm 1 describes the learning process.

Algorithm 1 Extended MKL Learning with Gompertz model

Input: The convergence threshold t
  Initialize the values of v and w
  repeat
    Calculate the $K_G(x_i, x_j)$ with the Gompertz Function
    solve the MKL problem with $K_G(x_i, x_j)$ to generate the value of u, w and b and objective value Obj^{r+1}.
    $v^{r+1} = v^r - \mu^r \frac{\partial \Gamma}{\partial v}$
    $w^{r+1} = w^r - \mu^r \frac{\partial \Gamma}{\partial w}$
  until $|Obj^{r+1} - Obj^r| < t$

Although our GMKL model is inspired by LMKL, there are several main differences:

- LMKL makes the assumption that training data and test data have same distributions. GMKL addresses the non-stationary problem by favoring recent data.
- LMKL is proposed for learning from single data source but different kernel functions. GMKL learns the relationship between different data sources with same kernel function.
- LMKL solves the problem of discovering which kernel function is better for a certain region of the kernel matrix. GMKL assigning the weights to different regions by considering the order of time series data.

IV. EXPERIMENTS

Based on the convex optimization theory, MKL should have better accuracy than applying SVM with any single data source. However, our experiment results indicate that when applying to financial data the MKL models do not always achieve better accuracy than the single SVM models. We apply our Gompertz Multiple Kernel Learning model (GMKL) to the *volatility* forecasting with five major international stock indexes and compare the results with MKL and SVM.

A. Experiment Setting

Intuitively, volatility represents the *risk* of a period. The higher the volatility is, the more the price changes. Therefore the definition of volatility is related to the definition of return R_t. The volatility $\mathcal{V}_t$ is then defined as $\mathcal{V}_t = \{\log \mathcal{R}_t\}^2$.

We use daily data from January 2007 to December 2009 for five major international stock indexes: Dow Jones Industrial Average (United States), S&P500 (United States), FTSE 100 (England), Hengsheng (Hong Kong), Nikkei 225 (Japan). For each index, we train and forecast for a certain number of shifting periods and calculated the average performance. In each single period, we take a certain number of days as training and test the learned model with the following day. Figure 2 describes the training and forecasting with the shifting periods.

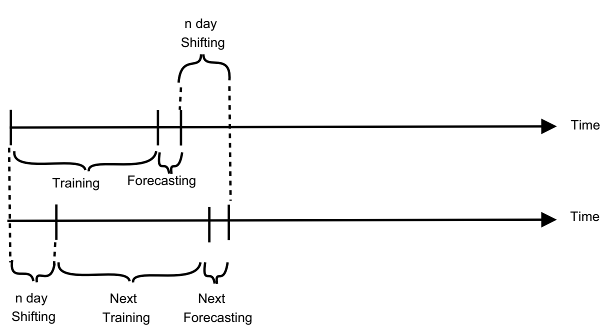

Figure 2. Training and Forecasting with the Shifting Periods

We calculate the average performance of SVM, MKL and GMKL models with shifting training periods and employ the relative absolute error (RAE) as our measurement. The definition of RAE is $\frac{\sum |y_m - y|}{\sum |y_{rw} - y|}$, where y_m is the forecasting volatility value of certain model m, y_{rw} is the forecasting volatility value of the random walk model and y is the real volatility value. If the RAE measure is less than 100%, the performance of the model is better than the random walk model. The less the RAE measure is, the better performance the model has. Ideal model (perfect forecasting) will give 0% REA measure.

B. Experiment Results

We conduct two different kind of experiments to test the performance of our GMKL model. For the first scenario we focus on forecasting the DJI index with DJI and one of other international stock indexes. For the second scenario we forecast five indexes with all five indexes. In both scenarios, we consider the single index data for SVM as the benchmark. We use multiple (i.e., two and five) index data sets for both MKL and GMKL.

1) Forecasting DJI with DJI and another index data source: Based on the domain knowledge, the training period should not be too short or too long. Hence we focus on the training periods between 300 days to 400 days.

Figure 3 shows the comparison of forecasting DJI indexes with SVM and MKL models. The *SVM* represents the result by applying SVM with the DJI data only, which is treated as the benchmark. The *MKL-FTSE* represents MKL with the DJI data and the FTSE data. Other *MKL-* lines are similar to *MKL-FTSE*. As one can see, *MKL-FTSE* performs worse than SVM, which means that the MKL model is not robust for the financial forecasting problem, such as the non-stationary problem. This may be caused by the limitation of the linear combination of kernel matrices. The global optimal solution of training data may not always be the same for testing data. On the other hand, our GMKL model considers a non-linear combination of kernel matrices and favor recent data (distribution). Therefore GMKL tries to discover the solution in a more general search space and hence achieves a better accuracy. Figure 4 demonstrates the comparison of forecasting DJI indexes with SVM and GMKL models. As can be seen, the GMKL models not only are more robust than the MKL models but also achieve better accuracies.

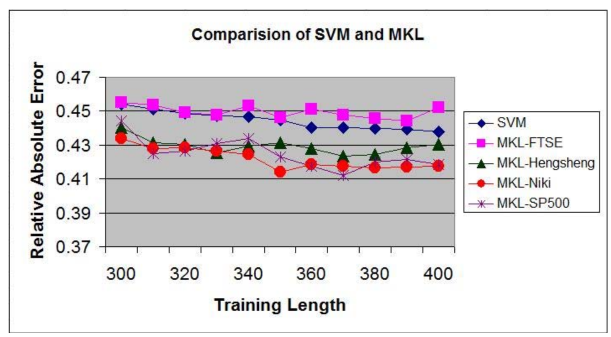

Figure 3. Comparison of SVM and MKL on forecasting DJI

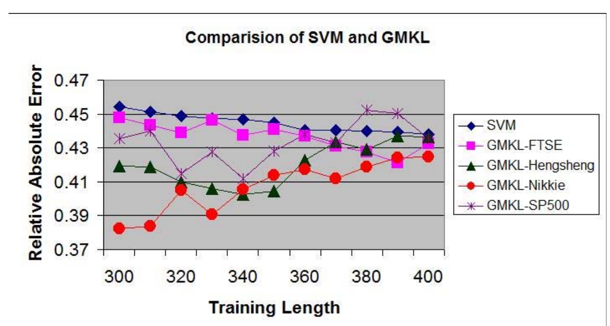

Figure 4. Comparison of SVM and GMKL on forecasting DJI

2) Forecasting five indexes with all five indexes data sources: To further verify our GMKL model, in this more general scenario we forecast five indexes with all five indexes data sources. Figure 5, 6, 7, 8 and 9 provide the results of forecasting the five indexes with SVM, MKL and GMKL respectively. As shown, when forecasting DJI MKL performs worse than SVM with all the training lengths while our GMKL achieves lower error rate than SVM with all the training lengths. In the FTSE test case, both MKL and GMKL perform better than SVM while GMKL beats MKL in 7/10 training lengths. Since both of DJI and FTSE are the largest stock markets in the world, these two cases show that GMKL is more robust than MKL. The Hengsheng test case is similar to the FTSE test case except GMKL performs clearly better than MKL. As the earliest stock mark in the world, the Nikkie market is special. When trading agents of other markets make prediction, they always consider the Nikkie index as one of the most important sources. Therefore combining other sources to forecasting the Nikkie index may have no advantages. That is one of the possible reason that MKL performs much worse than SVM. However, our GMKL model performs similar to SVM. Although both the S&P 500 and DJI indexes belong to United States, the results of these two test cases have slight differences, which may be due to the fact that S&P 500 and DJI reflect different economical aspects.

The results of these five test cases indicate many interesting relationships between the major international indexes. However, trying to discover and explain these relationship is not the focus of this paper. In summary, GMKL always performs better than both SVM and MKL in all the test cases.

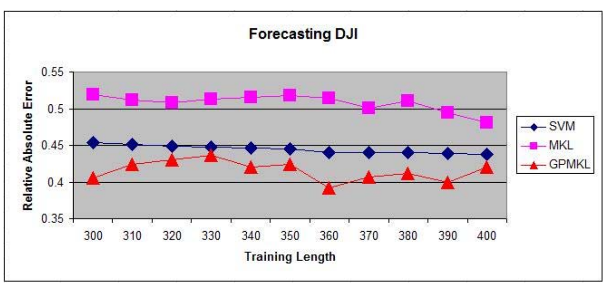

Figure 5. Forecasting DJI with all five indexes training data

V. Conclusion

In our paper we propose the Gompertz Multiple Kernel Learning (GMKL) method, which extends the Multiple Kernel Learning methods to address the non-stationary problem in financial forecasting. GMKL considers the non-linear combination of kernel matrices with Gompertz model which assigns different weights to former and recent training data instances. This non-linear combination sacrifices the convexity of the optimization problem, but it enlarges the search space of optimal global kernel matrix. Therefore the performance of GMKL also relies on the initial values. We apply our GMKL model to five major international stock indexes and compare the results with single SVM models and the MKL model. The experiment results indicate that GMKL performs better than both SVM and MKL. Moreover, our GMKL model is more robust than MKL when considering more training data sources.

Our main contributions to data mining are:

- We propose a novel multiple kernel model to integrate multiple financial time series data sources.
- We propose a domain specific kernel function to leverage domain knowledge in the mining process.
- We conduct real data experiments to verify both of our hypothesis and GMKL model.

Since our model is proposed specifically for financial time series data, we also have the following contributions to financial forecasting:

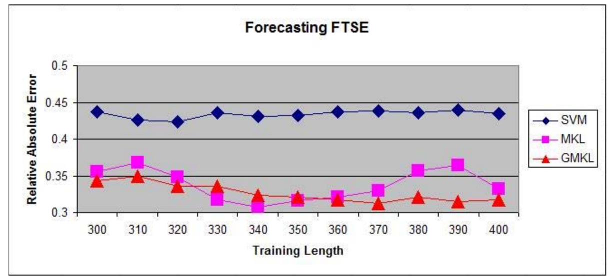

Figure 6. Forecasting FTSE with all five indexes training data

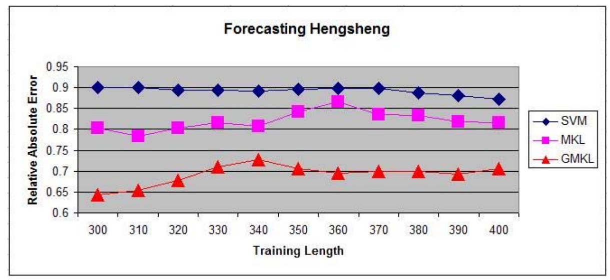

Figure 7. Forecasting Hengsheng with all five indexes training data

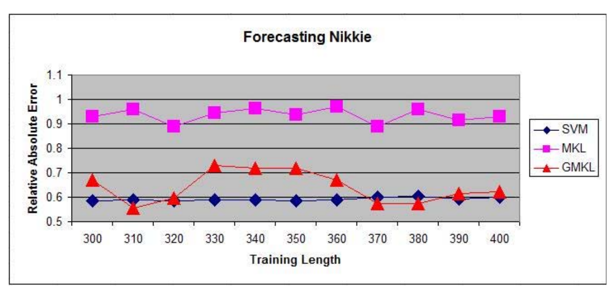

Figure 8. Forecasting Niki with all five indexes training data

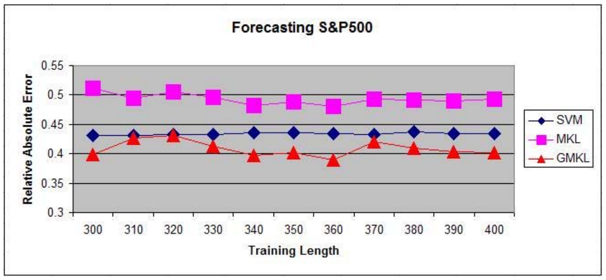

Figure 9. Forecasting S&P500 with all five indexes training data

- We apply state-to-art multiple kernel methods to address the international market integration problem.
- We employ Gompertz model to address the non-stationarity of the financial time series data.
- Our experiment results reveal some interesting relationships among multiple international stock markets.

Although we focus on volatility forecasting of stock markets in this paper, our GMKL model could be applied to more general financial forecasting problems. Therefore in the future we will apply our GMKL model for other financial markets, such as exchange markets, and also define more domain specific kernels. Furthermore, we will explore more sophisticated models to assign weights of former and recent data instances besides the Gompertz function.

ACKNOWLEDGMENT

We would like to thank Dr. Jieping Ye for insightful discussion.

REFERENCES

[1] F. R. Bach, G. R. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *ICML*, 2004.

[2] T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

[3] N. L. Bowers, H. U. Gerber, J. C. Hickman, D. A. Jones, and C. J. Nesbitt. *Actuarial Mathematics*. Society of Actuaries; 2 edition, 1997.

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

[5] D. G. DICKINSON. Stock market integration and macroeconomic fundamentals: an empirical analysis, 1980-95. *Applied Financial Economics*, 10:261–276, 2000.

[6] S. Fu, S. Guo, Z. Hou, Z. Liang, and M. Tan. Multiple kernel learning from sets of partially matching image features. In *ICPR*, pages 1–4, 2008.

[7] M. Gönen and E. Alpaydin. Localized multiple kernel learning. In *ICML*, pages 352–359, 2008.

[8] G. R. Lanckriet, N. Cristianini, P. L. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[9] F. Martinez, A. Troncoso, J. C. Riquelme, and J. S. Aguilar. Lbf: A labeled-based forecasting algorithm and its application to electricity price time series. *Data Mining, IEEE International Conference on*, 0:453–461, 2008.

[10] J. Moody. Economic forecasting: Challenges and neural network solutions. In *International Symposium on Artificial Neural Networks*, 1995.

[11] K. N. Mukherjee. Review of stock market integration: An international perspective. *The IUP Journal of Applied Economics*, 3:96–124, 2004.

[12] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. More efficiency in multiple kernel learning. In *ICML*, pages 775–782, 2007.

[13] J. K. Shim. *Schaum's Outline of Financial Management*. McGraw-Hill; 2 edition, 1998.

[14] S. Sonnenburg, G. Rätsch, and C. Schäfer. A general and efficient multiple kernel learning algorithm. In *Advances in Neural Information Processing Systems*, 2005.

[15] F. Suard, A. Rakotomomonjy, and A. Bensrhair. Model selection in pedestrian detection using multiple kernel learning. In *Intelligent Vehicle Symposium*, 2007.

[16] D. Turnbull, L. Barrington, G. R. Lanckriet, and M. Yazdani. Combining audio content and social context for semantic music discovery. In *SIGIR*, pages 387–394, 2009.

[17] M. Varma and B. R. Babu. More generality in efficient multiple kernel learning. In *ICML*, page 134, 2009.

[18] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[19] L. Wang and J. Zhu. Financial market forecasting using a two-step kernel learning method for the support vector regression. *Annals of Operations Research*, 174(1):103 – 120, 2010.

[20] T. Wheldon. *Mathematical Models in Cancer Research*. Bristol: Adam hilger, 1988.

[21] J. Ye, K. Chen, T. Wu, J. Li, Z. Zhao, R. Patel, M. Bae, R. Janardan, H. Liu, G. Alexander, and E. Reiman. Heterogeneous data fusion for alzheimer's disease study. In *SIGKDD*, pages 1025–1033, 2008.

[22] J. Ye, S. Ji, and J. Chen. Learning the kernel matrix in discriminant analysis via quadratically constrained quadratic programming. In *SIGKDD*, pages 854–863, 2007.