Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Using heavy-tailed distributions to stress-test kernel methods for segregating the firms that are likely to survive

Pouyan Hosseinizadeh
Mechanical & Industrial Engineering Department
Ryerson University
Toronto, Canada
pouyan.hosseinizadeh@ryerson.ca

Aziz Guergachi
Ted Rogers School of Information Technology Management
Ryerson University
Toronto, Canada
a2guerga@ryerson.ca

***Abstract*—while kernel-based learning methods have emerged during the last two decades as major tools to effectively manage uncertainty, heavy-tailed distributions remain a major challenge for modelers who aim to predict the future behavior of complex systems. In this article, Weibull distribution has been used to stress-test kernel-based methods and study more specifically the impact of heavy-tailed distributions on the performance of Fisher kernels in identifying the potential for collapse of an enterprise based on its stock price.**

***Keywords*—Fisher kernel, financial time series, modelling, prediction, Weibull distribution, heavy-tailed distributions**

I. Introduction

The last two decades have seen a significant development in the area of statistical learning [12] and kernel-based methods [2] [9] [10] [11]. Based on a solid mathematical framework, these methods have been shown in many applications to outperform the existing tools such as neural networks and conventional statistical techniques. Among their strengths is that they make some of the weakest assumptions about the characterization of the uncertainty that underlie the system under study. In particular, and contrary to the maximum likelihood method, for instance, statistical learning theory and kernel-based methods can be implemented without having any prior knowledge of the probability distribution of the input/output data. Another significant advantage of these methods concern the format of the inputs on the basis of which an output prediction needs to be made. In most of the existing tools, the inputs have to be part of a vector space and, if they are not, they need to be somehow 'vectorized', which would usually lead to some loss of information. Kernel-based learning methods, on the other hand, can process the inputs without having to reduce them to coordinates and vectors [13]. By introducing a mapping from the input space to another space usually referred to as the feature space, kernels allow modelers to define the notion of similarity or closeness among inputs even if they don't belong to a vector space. Thus kernels can handle inputs that are in the format of graphs, trees, text, or time series.

A variety of kernels have been proposed to describe inputs' similarities [3]. The most popular ones are the polynomial and Gaussian kernels. But one kernel that has showed a consistent performance when the inputs are highly complex non-vector objects is the Fisher kernel, which was introduced by Jaakkola and Haussler in 1999 [4]. It has been implemented for a variety of applications including signal processing and computational biology.

In contrast to this notable success of kernel-based methods, a major challenge that is currently facing the systems engineering community concerns the type of uncertainties that arise in the case of heavy-tailed distributions. As opposed to the very popular normal (or Gaussian) distribution, a heavy-tailed distribution would give rise to very large deviations (from the mean, *if* it exists) at a probability that is not so negligible. According to [5], a probability distribution is heavy-tailed if its tail is not exponentially bounded.

Some researchers claim that most of the failures that we witness in the financial markets are due to the fact that the behavior of these markets is consistent with heavy-tailed probability distributions, while the mathematical tools that are used to manage them are based on the assumption that the underlying uncertainty can be modeled by a normal distribution [6]. Mathematicians tend to use normal distributions to describe uncertainty mostly because they make the computations and the modeling process more tractable, certainly not because the phenomena that are under study follow this distribution.

The purpose of the research presented in this paper is to run a set of simulation experiments to stress-test the performance of kernel-based methods using situations where the data arise according to heavy-tailed distributions. Although statistical learning theory, which is the foundation of some kernel-based learning methods (such as support vector machines for instance), would fall apart in the case where large deviations occur [7], this research work intends to test empirically the robustness of kernel-based learning methods by investigating the impact of heavy-tailed distributions on their performances.

The paper is organized as follows: Section II presents the modeling approach, Section III explains the data generation process, Section IV introduces the concept of Fisher kernels, and finally Section V presents the results.

II. MODELING APPROACH

In this research, the stress-tests are carried out in the context of the analysis of financial markets. The amount of research that has been devoted to modeling the financial markets in order to understand how stock prices vary over time is overwhelming. In general, however, there are two major approaches to analyze financial data [1]: Fundamental Analysis, which studies the factors that affect financial markets and intends to predict the stock market behavior for next periods based on the analysis of these factors, and Technical Analysis that looks primarily at to the stock prices during the past periods and attempts to forecast the future prices by extracting patterns from the past market behavior.

The method that we use in this research is closer to the Technical Analysis approach. Our goal, however, is not to make predictions about the values (or range of values) of stock prices for the next day, week or month. Rather, our intention is to focus on a problem that is different and, we hope, more feasible: identifying the potential of collapse in an enterprise by looking at its stock price over a certain time window of size m.

In general, there are two types of companies recognizable in each sector of the stock market. First, companies which went bankrupt, merged to or bought by other companies and do not exist in the market as standalone firms anymore. These are referred to as the *Dead* companies. The second type includes the companies that do exist in the market at the present time. These companies are called *Active* companies. The modeling approach that we have adopted in this research consists of the following steps:

(1) The machine sees the stock prices of a collection of companies over a certain period of time; some of these companies are dead, while other ones are active at the present time, i.e., each company is associated with a label in a binary fashion: either dead or active.

(2) To process the stock prices that are supplied to the machine, the latter makes use of a Fisher kernel (more explanation is provided below) to reduce the input stock price data from a time series format to a vector format. In other words, a mapping (called feature mapping) is constructed from the space of time series that contains the input data (financial time series) to a vector space F (called the feature space) that is spanned by Fisher scores (see any textbook on mathematical statistics for the definition of Fisher scores). In this study, the generative probability model that is used to compute the Fisher scores is the Gaussian distribution (see below for more details) that is characterized by two parameters: the mean and the variance. As a result, the feature mapping will involve two Fisher scores and, therefore, the Feature space will be a two-dimensional space.

(3) While it may have very well been possible to apply the support vector machines algorithm to determine the separating hyperplane in the feature space, we limit ourselves in this paper to the plotting and visualization of the points corresponding to the companies being studied as well as their class labels (dead versus active). The success of the implementation of the Fisher kernel is measured by the extent to which the two classes appear, by visual inspection, to be segregated in the two dimensional plot using Fisher scores as coordinates.

In the article [14], the authors showed, using <u>*real*</u> stock market data, that Fisher kernel was able to segregate the two classes for *some* sectors of the stock market. In this study, we intend to investigate the case where the stock prices are affected by a large volatility and, therefore, subject to large deviations caused by a heavy-tailed distribution, i.e., a large amount of uncertainty. To do this and be able to control the nature of uncertainty that underlies the stock market data, we made use of simulations implementing the Weibull distribution to generate the stock prices (more details below in section III).

Based on our modeling approach, the stock price of company i at, say, week j, denoted as "x_{ij}" follows the Gaussian distribution:

$$P(x_{ij}) = \frac{1}{\sigma_j\sqrt{2\pi}} exp\left(-\frac{1}{2}\left(\frac{x_{ij}-\mu_j}{\sigma_j}\right)^2\right) , \qquad (1)$$

where μ_j and $\sigma_j{}^2$ are the mean and variance of the stock price at week j for all available companies of the same type. These values can be shown as the coordinates of the following vectors:

$$\mu = [\mu_1 \quad \mu_2 \quad \mu_3 \quad \cdots \quad \mu_W]_{1\times W}$$
$$\sigma^2 = [\sigma_1^2 \quad \sigma_2^2 \quad \sigma_3^2 \quad \ldots \quad \sigma_W^2]_{1\times W} \qquad (2)$$

This concept is depicted in figure 1.

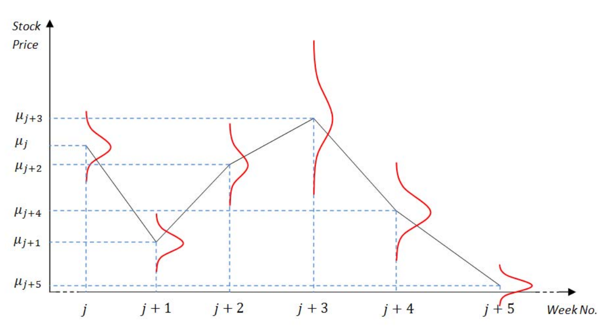

Figure 1. Weekly stock prices of each company can estimate by Gaussian probability distribution

III. Data Generation Process

While the probability model that is used to construct the Fisher kernel is the Gaussian distribution, and since we aim at stress-testing the performance of kernel-based methods, the stock price values are generated in a simulated environment using a heavy tailed probability distribution that allows us to cover a wide range of intensities of stock price volatility (and therefore uncertainty). There are different heavy tailed distributions available; however, given the fact that the random numbers to be generated (stock prices, that is) have to be positive, a right-tailed probability distribution which promises the generation of data from zero to positive infinity is selected. Weibull distribution looks to be a right choice, since depending on the values of its parameters it assumes various shapes including that of a heavy tailed distribution:

$$f(x,\alpha,\beta)=\left(\frac{\alpha}{\beta}\right)\left(\frac{x}{\beta}\right)^{\alpha-1}e^{-\left(\frac{x}{\beta}\right)^{\alpha}}. \quad (3)$$

Adjusting the shape parameter to be less than one ($\alpha < 1$), Weibull distribution becomes a heavy-tailed probability distribution.

To give the reader an idea about the amount of volatility introduced by the Weibull distribution compared to the volatility that is found in the real-world stock market, we present in Figures 2 and 3 the *actual* weekly stock prices of active and dead companies in the “Oil and Gas Producers” sector from 1995 to 1996; these stock prices were collected from various North American stock exchanges. The stock price values vary between 0 to almost \$35 for the active companies in figure 2, and between 0 and about \$71 for dead companies in figure 3 during the same period (from 1995 to 1996). Examination of their variances reveals that it is very difficult to find a general pattern among dead or active companies, so that one can decide whether a given company will be dead or remain active in the future. The high dimension of the financial time series itself adds more challenges to the task of classifying a company as one that will survive or will collapse.

Now using the simulated data generated through the implementation of the Weibull distribution in its heavy-tailed form, this task will supposedly become even more complicated because the simulated stock prices will show deviations from the mean that are a lot larger. Indeed, the randomly generated financial time series data (using equation 3) for active and dead companies are plotted in Figures 4 and 5 respectively and show a higher degree of volatility in stock prices compared to the real-world values.

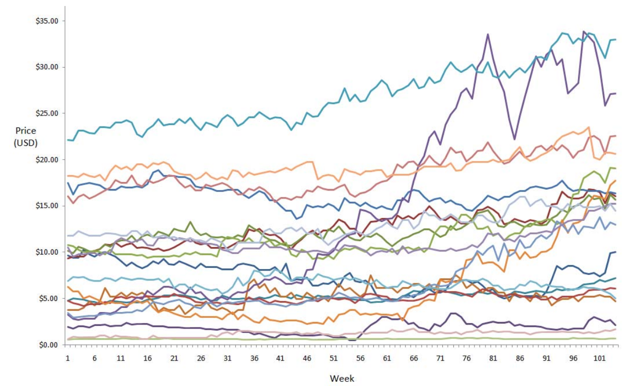

Figure 2. Weekly Stock prices for 20 active companies in “Oil & Gas Producers” sector between 1995 and 1996

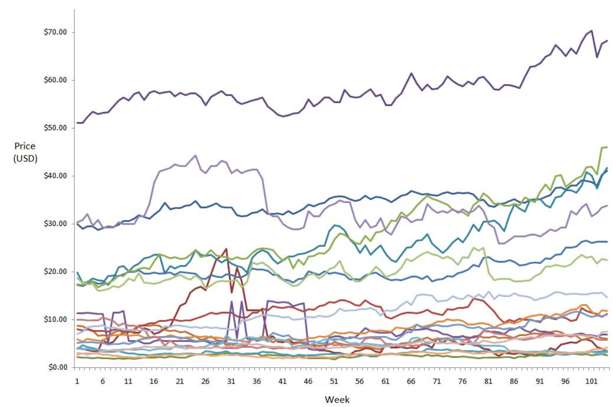

Figure 3.Weekly Stock prices for 20 dead companies in “Oil & Gas Producers” sector between 1995 and 1996

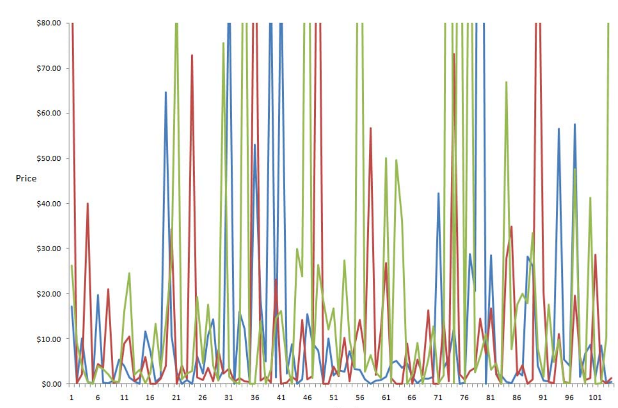

Figure 4. The random generated weekly stock prices for active companies for two years

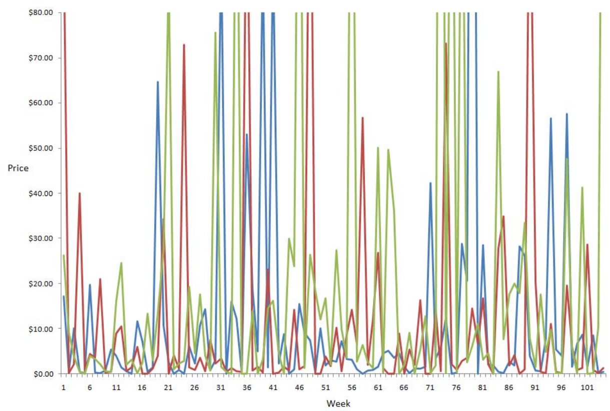

Figure 5. The random generated weekly stock prices for dead companies for two years

The random stock price data is generated separately with different scale parameters for each of the active and dead categories. Each category is associated with a constant shape parameter. Fifty companies of each category are simulated, with the stock price time series spanning a period of hundred weeks.

IV. Fisher Kernel & Its Application in Analysis of the Financial Time Series

Fisher kernel is used in this paper because it has been shown to perform very well in many applications [4]. Generative models such as Gaussian Mixture Models and Hidden Markov Models [8] have been used to model the time series data [1]. However, Fisher kernel is a different strategy that attempts to extract from a generative model more information than simply its output probability. The goal is to obtain internal representation of the data items within the model [3]. Since it was decided that a Gaussian probability distribution will be our generative model (see Section II), for each, say, week j, x_{ij} is the stock price value of company i with probability of $P_{\theta_0}(x_{ij})$ where $\theta_0 = \{\mu_j, \sigma^2{}_j\}$. Therefore, in order to calculate the probability for each data point $X_i = \{x_{ij}\}_{j=1}^{W}$ (note that a data point here represents the whole time series), one needs to multiply W number of Gaussian distributions, each having its specific estimated parameters μ_j, and $\sigma_j{}^2$ as

$$P(X_i) = \prod_{j=1}^{W} P(x_{ij}) = \prod_{j=1}^{W} \frac{1}{\sigma_j\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x_{ij}-\mu_j}{\sigma_j}\right)^2\right) , \quad (4)$$

where W is the total number of, say, weeks over which the stock price data are collected or randomly generated. We define $P(X_i)$ with respect to the Gaussian model to be the likelihood of that specific data point X_i. So for a collection of inputs $X = \{X_i\}_{i=1}^{C}$ with C as number of companies in a set of training data, the model parameters can be learnt by adapting all the points to maximize the likelihood of the training set. In order to get rid of the complicated calculations, in terms of dealing with several multiplications, in practice the log-likelihood of the data points with respect to the generative model is used as the generative model:

$$log\, P(X_i) = \, log \prod_{j=1}^{W} P(x_{ij}). \quad (5)$$

Consider the vector gradient of the log-likelihood with respect to the parameter vector:

$$g(\theta, X_i) = \nabla_{\theta}\, log\, P(X_i) \quad (6)$$

where θ is the parameter vector, in this case it is the mean and variance. As each of the coordinates of the parameter vector is a vector itself, for each data point X_i, $g(\theta, X_i)$ is defined as its Fisher scores vector with respect to the generative model for the given set of parameters θ. The Fisher score gives an embedding into the feature space which is, in this case, the Fisher score space. The Fisher information matrix I_M is used to define a non-standard inner product in that feature space and, therefore, the Fisher kernel *K*:

$$K(X_i, X_j) = g(\theta, X_j)'\, I_M{}^{-1} g(\theta, X_i). \quad (7)$$

The practical Fisher kernel is obtained by replacing the Fisher information matrix by the identity:

$$K(X_i, X_j) = g(\theta, X_j)'\, g(\theta, X_i). \quad (8)$$

In order to perform a visual inspection for segregation of one class from the other we use the summation over all the elements of the each coordinates of the Fisher scores vector, as proposed in [14] to have a two dimensional point representing each company.

V. Experiment, Results and Discussion

In a nutshell, our experiment procedure includes the following steps:

- Random generation of stock price time series using a certain instance of the Weibull probability distribution for each class of data
- Selection of the generative model and estimation of the associate parameters for each data point in the input space
- Calculation of Fisher scores for all data points

- Plot of the Fisher scores and visual inspection

By application of Fisher kernel and using the generative model it is possible to decrease the dimension of the data to be equal to the number of the generative model's parameters. So for every company in each class of data there is a unique Fisher score vector that can be easily plotted. The Fisher scores for the randomly generated data are plotted in Figure 6.

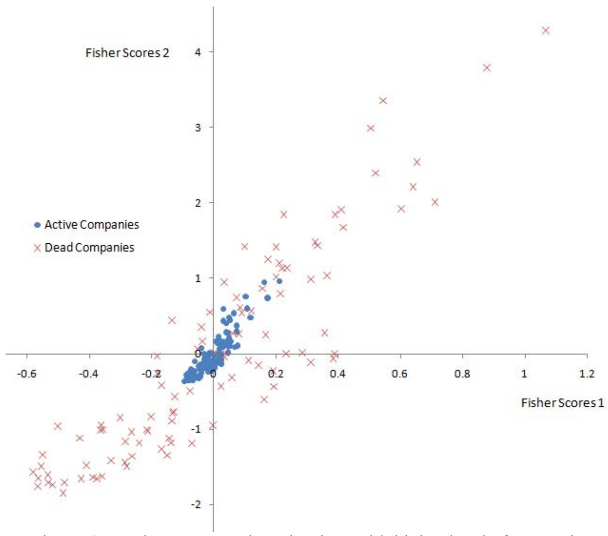

Figure 6. Random generated stock prices with higher level of uncertainty

Figure 6 reveals that although the generated data includes in itself a great deal uncertainty, it is still possible to visually recognize and separate the two classes of dead and active companies. In this way, given a time series of a new company in the same sector, it is possible to simply plot the Fisher score coordinates for this time series and visually inspect if it is close to area where the full circles are concentrated to judge the risk of possible collapse. The two areas (full circles and the x) are not perfectly separable; however, compared to the level of volatility that has been introduced through the heavy-tailed nature of the distribution, it is still possible to have an acceptable judgment of risk of collapse for a new data point. As the deviations from the mean in stock prices decrease, the Fisher scores of each class of data tend to take a more defined shape that resembles parabola. As these deviations decrease even further, it becomes possible to separate the two classes almost and, therefore, the accuracy of the results increases. Figure 7 illustrates the Fisher scores plot for a lower degree of uncertainty in stock prices.

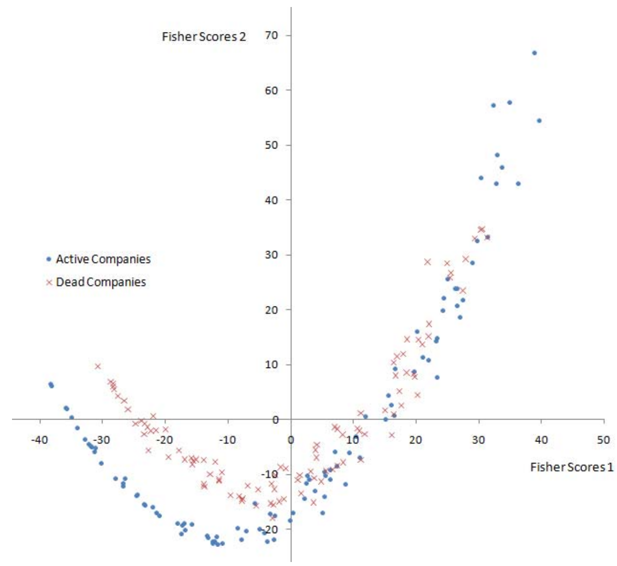

Figure 7. Random generated stock prices with lower level of uncertainty

VI. Future Works

As one of our goals in this research is to test the robustness of the proposed approach, we needed to keep a high level of uncertainty in the randomly generated stock prices and test the performance of the method in such conditions. As part of future work, there is need to look at ways to select a generative model which fits better with the nature of financial time series, superimpose the application of support vector machines over the visualization method explained above, and make use of the proposed method to estimate the parameters used in the inclination analysis [15].

Acknowledgment

Sincere thanks to NSERC, CFI, OIT and Ryerson University for funding the research.

References

[1] Y. Zhang, "Prediction of Financial Time Series with Hidden Markov Models", School of Computer Science, vol. Master of Applied Science: Simon Fraser University, 2004.

[2] N. Cristianini and J. SH. Taylor, "An introduction to support vector machines and other kernel-based learning methods". United kingdom: Cambridge University Press, 2000.

[3] J. SH. Taylor and N. Cristianini, "Kenel methods for pattern analysis", United kingdom: Cambridge University Press, 2004.

[4] T.S. Jaakkola and Haussler, "Exploiting generating models in discriminanative calssifiers", Advances in neural information processing systems, 1998.

[5] Asmussen, Søren (2003). Applied probability and queues. Berlin: Springer.

[6] The (Mis)behavior of Markets: A Fractual View of Risk, Ruin, and Reward; Benoit Mandelbrot, Richard L. Hudson; Basic Books edition

[7] V. Vapnik, Statistical Learning Theory, Wiley-Interscience, New York, 1998.

[8] Lawrence R. Rabiner, “A tutorial on Hidden Markov Models and selected applications in speech recognition”. Proceedings of the IEEE 77 (2): 257-286, 1989.

[9] Claus Bahlmann, Bernard Haasdonk and Hans Burkhardt, “On-line Handwriting Recognition using Support Vector Machines - A kernel approach”. In Int. Workshop on Frontiers in Handwriting Recognition (IWFHR) 2002, Niagara-on-the-Lake, Canada, August 2002

[10] Abdul Rahim Ahmad, Marzuki Khalid, Christian Viard-Gaudin, Emilie Poisson, “Online handwriting recognition using support vector machine”, ©2004 IEEE

[11] Kyoung-jae Kim, “Financial time series forecasting using support vector machines”, 2003 Elsevier

[12] Vladimir Vapnik. The Nature of Statistical Learning Theory. Springer-Verlag, 1995.

[13] NICOTRA, L., A. MICHELI and A. STARITA, “Fisher kernel for tree structured data”. Neural Networks, 2004 IEEE

[14] Yashodhan Athavale, Pouyan Hosseinizadeh, Sridhar Krishnan, Aziz Guergachi,; “Identifying the potential for Failure of Businesses in the Technology, Pharmaceutical and Banking Sectors using Kernel-based Machine Learning Methods”. SMC 2009

[15] Pouyan Hosseinizadeh, Aziz Guergachi, Vanessa Magness, “Predicting System Collapse: Two Theoritical Models”.SMC 2009