Mining Distribution Change in Stock Order Streams

Xiaoyan Liu#1, Xindong Wu*†2, Huaiqing Wang$3 Rui Zhang#4 , James Bailey#5, Kotagiri Ramamohanarao#6

# *Department of Computer Science & Software Engineering, University of Melbourne, Australia*
1,4,5,6 {xiaoyanl,rui,jbailey,rao}@csse.unimelb.edu.au

* *School of Compupter Science & Information Engineerring, Hefei University of Technology, China*
† *Department of Computer Science, University of Vermont, USA*
2xwu@uvm.edu

$*Department of Information Systems, City University of Hong Kong, HK*
3iswang@cityu.edu.hk

Abstract— **Detecting changes in stock prices is a well known problem in finance with important implications for monitoring and business intelligence. Forewarning of changes in stock price, can be made by the early detection of changes in the distributions of stock order numbers. In this paper, we address the change detection problem for streams of stock order numbers and propose a novel incremental detection algorithm. Our algorithm gains high accuracy and low delay by employing a natural Poisson distribution assumption about the nature of stock order streams. We establish that our algorithm is highly scalable and has linear complexity. We also experimentally demonstrate its effectiveness for detecting change points, via experiments using both synthetic and real-world datasets.**

I. Introduction

In the financial world, there is great interest in prompt detection of stock price change, which is critical for making intelligent trading decisions. Directly observing stock price usually leads to delayed report of changes, since we will not notice a stock price change until it has already happened. In this paper, we follow an alternative way of detecting stock price change through the detection of *distribution change* in the number of stock *orders*, based on the following well-established findings in financial research. When there is private information (e.g., a company is going bankrupt) available to a small group of traders, the private information will cause abnormal trading behavior that changes the distribution of the number of stock orders preceding the stock price change [5], [1]. In their seminal research paper, Easley et al. [5] argue that "it is private information rather than public information that leads to abnormal trading activity preceding price changes". For example, if some traders know inside news that an oil company has found a new oil source, they will submit large numbers of orders to buy the stocks of that company. This abnormal behavior will cause a change in the distribution of the number of stock orders, and the distribution change of the number of stock orders will in turn cause stock price increase in the near future. If we can monitor the distribution change of the number of stock orders, then we can make a better prediction of the stock price change, even if we do not know the private information (i.e., the inside news in the above example).

It is also widely accepted and validated in the financial research community that stock order arrivals are independent of each other and the number of orders in the cases with private information (available to a smaller group of traders) and the number of orders in the cases without private information follow two different Poisson distributions [5], [1]; this actually serves as the basis of sequential trading models for high-frequency stock data in finance.

In this paper, our goal is to detect the distribution change in the sequence of stock orders. Due to the nature of financial applications, a change point detection algorithm should satisfy the following three key requirements: (1) ***Accuracy***: the algorithm should detect as many as possible actual change points and generate as few as possible false alarms. (2)***Promptness***: the algorithm should detect a change point as early as possible. (3)***Online***: be efficient enough for a realtime environment.

Most of the existing online detection algorithms use a sliding window framework. The windows may slide forward in different ways. Kifer et al. [7] based their detection algorithms on a two-window paradigm. The reference window for the current segment is fixed, and the current window moves forward one step each time if no change point is reported. Another strategy is to move the reference window and current window at the same time. It is used in algorithms such as the kernel change detection method (KCD) in [4]. The difference between the two windows is compared using the likelihood, Bayesian inference, mixture models, K-L divergence, etc [2], [8], [9]. However, these sliding-window-based methods inherently suffer from a high delay.

Existing methods for detecting distribution change such as the Wilcoxon method (WXN) [6] and the kernel change detection method (KCD) [4] are non-parametric methods. We adopt the Poisson assumption for stock orders in this paper. Although various works in statistics have studied the detection of change points for Poisson sequences [3], [10], these methods need to make a hypothesis on the values of pre-change and post-change density parameters first. A maximum likelihood (ML) method is used to detect distribution change in our paper.

The main novelty of our work is: we monitor the stock price by detecting the change in stock order streams. We adopt a parametric approach to attack the distribution change problem. The parametric approach yields a higher accuracy, while greatly reducing the computation time. A new way of

978-1-4244-5446-4/10/$26.00 © 2010 IEEE

advancing the reference window leads to a low delay of change point detection. We also provide a theoretical analysis that guides the choice of parameters for our algorithm.

II. Problem description

Motivated by the sequential trading model [5] in finance, our research problem is formulated in Eq.(1): the number of sell orders or buy orders in a period, X, is a random variable governed by the Poisson distribution. $x_1, x_2, \ldots, x_i, \ldots$, are a sequence of sample values of X. It independently follows the Poisson distribution $\lambda_1^x e^{-\lambda_1}/x!$ in the normal status, that is no news occurs. When a news event occurs and takes effect, more orders are submitted. At this time, the order number is in the abnormal status and change to follow the distribution $\lambda_2^x e^{-\lambda_2}/x!$. After the market digests all the information, the order number will recover to the normal status in which the order number follows the distribution $\lambda_1^x e^{-\lambda_1}/x!$, where λ_1 and λ_2 are unknown. We aim to find the change points, τ_1,τ_2,τ_3,..., between the two alternate distributions in an online manner.

$$x_i \in \begin{cases} \textit{Poisson}(\lambda_1) & i = 1, \ldots, \tau_1 - 1 \\ \textit{Poisson}(\lambda_2) & i = \tau_1, \ldots, \tau_2 - 1 \\ \textit{Poisson}(\lambda_1) & i = \tau_2, \ldots, \tau_3 - 1 \\ \quad\cdots & \end{cases} \tag{1}$$

III. Online Change Point Detection

In this section, we will present our online algorithms for change point detection, and provide an analysis of parameters.

A. Online Change Point Detection Algorithm

Different from methods based on sliding windows, our method judges whether each incoming data point is a change point, one by one, meaning a lower delay. The main idea of our method is that given the current segment, if the new incoming point is from the same distribution, the likelihood function ideally should be maximized under the new estimated distribution parameter. To identify the changes in time, we emphasize the influence of the newly arrived data point. But we don't need any *a priori* assumptions on the distribution parameter. It is learnt automatically during the change point detection process from the data stream. The length of our window increases until a change point is detected.

Suppose the sequence x_1,...,x_i,...(i=1, 2,...) is sample values of independent random variables that follow the Poisson distribution. Let $X_{i:j}$ denote the subsequence $x_i, \ldots, x_j$ $(i < j)$. If the latest change point is x_i, and x_j is the current point, then the subsequence $X_{i:j}$ is called the current segment, which means x_i,..., x_j are from the same distribution. Since x_i,..., x_j follow the distribution $\lambda^x e^{-\lambda}/x!$, we can estimate λ by the maximum log likelihood function:

$$\tilde{L}(\lambda|X_{i:j}) == \sum_{t=i}^{j}[x_t \ln(\lambda) - \lambda - \ln(x_t!)], \tag{2}$$

and then get the estimation $\hat{\lambda} = \sum_{t=i}^{j} x_t/(i - j + 1) = \bar{x}$.

It is known that $\bar{x}$ is an unbiased estimator of λ. When the new data point x_{j+1} become available, we update the estimator with $\hat{\lambda}' = (1 - \alpha)\hat{\lambda} + \alpha x_{j+1}$ $(\alpha \in [0, 1])$. If the new point x_{j+1} belongs to the same distribution as the current segment, then

$$\begin{aligned} E[\hat{\lambda}'] &= (1 - \alpha)E[\hat{\lambda}] + \alpha E[x_{j+1}] \\ &= (1 - \alpha)\lambda + \alpha\lambda = \lambda, \end{aligned}$$

where $E[\cdot]$ denotes the expectation. $\hat{\lambda}'$ is still the unbiased estimator of λ. $\hat{\lambda}'$ should maximize the probability $p(X_{i:j+1}|\hat{\lambda}')$. So we can get $p(X_{i:j+1}|\hat{\lambda}') \geq p(X_{i:j+1}|\hat{\lambda})$. Due to the estimate error, $\hat{\lambda}'$ may not maximize $p(X_{i:j+1}|\hat{\lambda}')$, that means $p(X_{i:j+1}|\hat{\lambda}')$ may be less than $p(X_{i:j+1}|\hat{\lambda})$. But since they are both the unbiased estimators of λ, $p(X_{i:j+1}|\hat{\lambda}')$ should not be significantly less than $p(X_{i:j+1}|\hat{\lambda})$. That is, $p(X_{i:j+1}|\hat{\lambda}) - p(X_{i:j+1}|\hat{\lambda}') < \delta$, where δ is the user-specified threshold.

Otherwise, the point x_{j+1} is a change point and from the Poisson distribution with $\lambda_1 = \lambda + \epsilon$. Let $\epsilon > 0$. Then

$$E[\hat{\lambda}'] = (1 - \alpha)E[\hat{\lambda}] + \alpha E[x_{j+1}] = \lambda + \alpha\epsilon.$$

If we set the value of α close to 1, $E[\hat{\lambda}']$ is close to λ_2. At this time, $p(X_{i:j+1}|\hat{\lambda}')$ must be much less than $p(X_{i:j+1}|\hat{\lambda})$ according to the interpretation of maximum likelihood. Therefore, if $p(X_{i:j+1}|\hat{\lambda}) - p(X_{i:j+1}|\hat{\lambda}') > \delta$, we say the point x_{j+1} is a change point. Since α can control the closeness of $E[\hat{\lambda}']$ to λ_2, it has an effect on the early detection of the change point. The structure of the algorithm MDD is given in Algorithm 1. In the algorithm, at each step we add one more point in and update the distance measure incrementally. Therefore, the computational complexity of our method for a time series with n points is $O(n)$.

```
Algorithm 1. ML based Distribution Change Point Detection Algorithm (MDD)
Input: data stream $X = (x_1, x_2, \ldots, x_i, \ldots)$, the initial window length $w$, threshold $\delta$, and $\alpha$
Output:change points $c_1, c_2, \ldots, c_n, \ldots$
Initial: $id = 1$; $c_1 = id$, i=1;
Read $w$ points to $Q = \{x_1, \ldots, x_w\}$ from $x_{id}$
$\hat{\lambda} = \sum_{t=1}^{w} x_t/w$
While not the end of the sequence
    Read the point $x_{id+w-1+i}$,
    $\lambda' = (1 - \alpha)\hat{\lambda} + \alpha x_{id+w-1+i}$
    $L_1 = L(Q \cup x_{id+w-1+i}, \hat{\lambda})$
    $L_2 = L(Q \cup x_{id+w-1+i}, \lambda')$
    If $L_2 < L_1 - \delta$
        Report $x_{id+w-1+i}$ as a change point
        $id = id + w - 1 + i$
        $i = 1$
        Read $w$ points to $Q$ from $x_{id}$
    Else
        $Q = Q \cup x_{id+w-1+i}$
        $i = i + 1$
    EndIf
    $\hat{\lambda} = \sum_{x_i \in Q} x_i/|Q|$
EndWhile
```

B. Analysis of Parameter Behavior

In our algorithm, w, α, δ are the parameters that need to be specified by the user. Suppose $x_1, \ldots, x_n$ are known to independently and identically follow the Poisson distribution with parameter λ, and x_{n+1} is the newly arrived data point. According to our algorithm,

$$\begin{aligned} L_1 &= L(\hat{\lambda}|X_{1:n+1}) = \ln(\hat{\lambda})\sum_{i=1}^{n+1} x_i - (n+1)\hat{\lambda}, \quad (3)\\ L_2 &= L(\lambda'|X_{1:n+1}) = \ln(\lambda')\sum_{i=1}^{n+1} x_i - (n+1)\lambda', \quad (4) \end{aligned}$$

where $\hat{\lambda} = \sum_{i=1}^{n} x_i/n$ is the maximum likelihood estimator of λ, and $\lambda' = (1-\alpha)\hat{\lambda} + \alpha x_{n+1}$. Then, we have

***Theorem* 1:** If x_{n+1} is from the same distribution as $x_1, \ldots, x_n$, then $E[p(X_{1:n+1}|\hat{\lambda}) - p(X_{1:n+1}|\lambda'))] = 0$.

In practical computation, we use the difference in log likelihoods. Let's consider $L_2 - L_1$,

$$\begin{aligned} L_2 - L_1 &= (n\hat{\lambda} + x_{n+1})\ln[1 + \alpha(\frac{x_{n+1}}{\hat{\lambda}} - 1)] \quad (5)\\ &\quad -(n+1)\alpha(x_{n+1} - \hat{\lambda}), \end{aligned}$$

By means of the second order Maclaurin polynomial series of $\ln(1+x)$,

$$\ln[1 + \alpha(\frac{x_{n+1}}{\hat{\lambda}} - 1)] = \alpha(\frac{x_{n+1}}{\hat{\lambda}} - 1) - \frac{\xi^2}{2}, \quad (6)$$

where

$$\xi = \frac{\alpha(\frac{x_{n+1}}{\hat{\lambda}} - 1)}{1 + \theta\alpha(\frac{x_{n+1}}{\hat{\lambda}} - 1)}(0 < \theta < 1).$$

Substituting Eq.(6) into Eq.(5), and we get

$$L_2 - L_1 = \alpha(\frac{x_{n+1} - \hat{\lambda}}{\sqrt{\hat{\lambda}}})^2 - (n\hat{\lambda} + x_{n+1})\frac{\xi^2}{2}, \quad (7)$$

Then

$$L_2 - L_1 < \alpha(\frac{x_{n+1} - \hat{\lambda}}{\sqrt{\hat{\lambda}}})^2. \quad (8)$$

In Eq.(8), if x_{n+1} is not a change point, the expression on the right side follows the distribution $\chi^2(1)$. Since the expectation of $\chi^2(1)$ equals 1, it follows that $E[L_2 - L_1] < \alpha$.

In Eq.(6), the expansion of the Maclaurin polynomial series requires $|\alpha(x_{n+1}/\hat{\lambda} - 1)| < 1$, then $\alpha < |\hat{\lambda}/(x_{n+1} - \hat{\lambda})|$. From Chebishev's inequality, we know

$$p(|x_{n+1} - \lambda| > k\lambda) \leq 1/k^2.$$

If we set $k = 5$, then $p(|x_{n+1} - \lambda| > 5\lambda) \leq 0.04$. So if we set $\alpha \leq 0.2$, then $|\alpha(x_{n+1}/\lambda - 1)| < 1$ is satisfied at the probability 96%. In a practical setting, we usually set $\alpha \leq 0.5$.

Considering the parameter δ, if $x_1, \ldots, x_n$ and x_{n+1} are from the same distribution *Poisson*(λ_1), then the maximum log likelihood is

$$L_1 = \ln(\lambda_1)\sum_{i=1}^{n+1} x_i - (n+1)\lambda_1. \quad (9)$$

If $x_1, \ldots, x_n$ are from the distribution *Poisson*(λ_1), and x_{n+1} is from the distribution *Poisson*(λ_2), then the corresponding true maximum log likelihood is

$$L_3 = \ln(\lambda_1)\sum_{i=1}^{n} x_i - n\lambda_1 + x_{n+1}\ln(\lambda_2) - \lambda_2. \quad (10)$$

L_1 and L_3 both approximate the true probability, but L_2 departs significantly from the true probability. Therefore, if x_{n+1} is a change point, $|L_2 - L_1| \geq |L_3 - L_1|$. The ideal δ is

$$\delta \geq |L_3 - L_1| = |x_{n+1}\ln(\lambda_2/\lambda_1) - (\lambda_2 - \lambda_1)|. \quad (11)$$

IV. Experiments

Our experiments are conducted on synthetic Poisson sequences and real stock data. The hardware used is an Intel 2.1GHz CPU with 2G memory. Programs were written in C++ and run using Windows Vista.

A. Synthetic dataset

Since the numbers of orders are assumed to follow the Poisson distribution empirically in finance research [5], we first conduct experiment with the synthetic Poisson sequences.

Our algorithm is compared with the WXN method in [7] and the KCD method in [4]. The evaluation criteria are

$$\begin{aligned} \textit{Precision} &= \frac{\textit{Number of correct detections}}{\textit{Number of detections}},\\ \textit{Recall} &= \frac{\textit{Number of correct detections}}{\textit{Number of real changes}},\\ F_1 &= \frac{2 \times \textit{Recall} \times \textit{Precision}}{\textit{Recall+Precision}}, \end{aligned}$$

Larger precision and recall mean better results for the algorithm. But it is difficult to simultaneously maximize these two indicators. So the F-measure (F_1) is used to take Precision and Recall into account together. To determine whether a detected change point is a real change point, we relax the condition for correct detection as follows: if

$$\textit{Dist}(d_i, \textit{RCP}) = \min_{r_j \in \textit{RCP}}\{|d_i - r_j| \leq \epsilon\}. \quad (12)$$

then $d_i \in \textit{DCP}$ is viewed as a real change point, where *RCP* and *DCP* respectively denote the position set of real change points and that of detected change points.

Table I gives the average results on 100 Poisson sequences under the best set of parameters. DlyT denotes the average delayed time points of the correct detections. The listed result of WXN is obtained under the confidence level 0.001 when both windows contain 10 points. The parameter of the KCD method is determined by following the method in [4]. The result of our algorithm MDD is obtained at $\alpha = 0.2$, $\delta = 1.2(\lambda_2 - \lambda_1)$, and the initial window lengths $w = 1, 10$. From Table I, we can see WXN gets the best result for precision and our method achieves a similar recall as WXN. The F_1 results of our method is satisfactory. However, the average delayed detection time of our algorithm is the smallest. It is crucial for online decision making. Figure 1 shows the computing time of

TABLE I

Results on Poisson Sequences

Method	Pre	Rec	F_1	DlyT
WXN	0.90	0.83	0.86	9.68
KCD	0.52	0.53	0.52	9.49
MDD($w = 1$)	0.76	0.80	0.78	2.17
MDD($w = 10$)	0.79	0.82	0.80	**1.63**

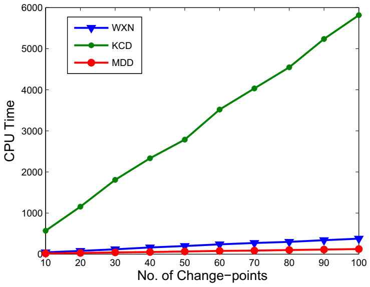

Fig. 1. Computing Time by CPU clock

each method under the numbers of change points. Our method is the fastest one.

B. Stocks dataset

We also test the proposed algorithm on 30 stocks from the Shanghai Stock Exchange. There are about 12000 points prepared for each stock's buy/sell sequence. In practice, we do not know the real change points in the order flow. Since the changes in the order number flow will precede the change in stock price mentioned in Section 1. we use "*lift*" as our criterion to test the effectiveness of our method. Let $P_i^{(J)}$ and $Pc_i^{(J)}$ denote the price change between the J time intervals before and after point i and change point i, respectively. The definition of "*lift*" is given as follows:

$$\mathit{lift} = \frac{\sum_{i=1}^{m} Pc_i^{(J)}/m}{\sum_{i=1}^{n} P_i^{(J)}/n}, \qquad (13)$$

where n is the number of total points in the stock sequence and m is the number of detected change points. That is the ratio of average price change on the detected points to the average price change on the whole sequence.

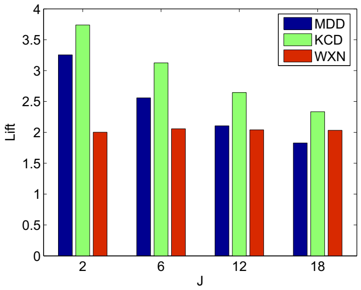

Fig. 2. Lifts under J=2, 6, 12, and 18

We calculate the average lift under $J = 2$, 6, 12, and 18. That means the lift is measured at 10 minutes, 30 minutes, 1 hour, and 90 minutes before and after the points. Figure 2 shows the average lifts of the three methods on 30 stocks. The lift of the WXN method is not as significant in short time intervals (J=2,6) as in long time intervals (J=12,18) compared with other methods. the KCD method is a little better than ours. The average computing time for WXN, KCD, and ICD is 190.9, 3057.47, 185.83 by CPU clock respectively. But KCD takes more time to find change points when these points are being identified. If taking into account both detection delay and real-time decision, our algorithm is the better choice for the online detection of changes in stock data than the other two methods.

V. Conclusions

In this paper, we have proposed an online algorithm, named MDD, for detecting distribution change in stock order streams, through a parametric approach. The MDD algorithm is characterized by low computational time and low delay in detection, and simplicity in choosing parameters with satisfactory accuracy. We have verified the efficiency and effectiveness of our algorithm through experiments on both synthetic and real datasets. Our idea of incrementally monitoring certain parameters can also be potentially extended to time series following other distributions.

Acknowledgments

This work has been supported by the Australian Research Council (ARC) under project DP0880250, the National Basic Research Program of China (973 Program) under award 2009CB326203, and the National Natural Science Foundation of China (NSFC) under award 60828005.

References

[1] R. Albuquerque, E. De Francisco, and L.B. Marques. Marketwide private information in stocks: Forecasting currency returns. *Journal of Finance*, 63(5):2297–2343, 2008.

[2] C. W. Baum and V. V. Veeravalli. A sequential procedure for multihypothesis testing. *IEEE Transactions on Information Theory*, 40(6):1994–2007, 1994.

[3] H. Boudjellaba, MacGibbon B., and Sawyer P. On exact inference for change in a poisson sequence. *Communications in Statistics: Theory and Methods*, 30(3):407–434, 2001.

[4] F. Desobry, M. Davy, and C. Doncarli. An online kernel change detection algorithm. *IEEE Transactions on Signal Processing*, 53(8):2961–2974, 2005.

[5] D. Easley, N. Kiefer, M. O'Hara, and J. P. Paperman. Liquidity, information and infrequently stocks. *Journal of Finance*, 51(4):1405–1436, 1996.

[6] M. Hollander and D. A. Wolfe. *Nonparametric Statistical Methods (2nd Ed.)*. New York: Wiley & Sons, 1999.

[7] D. Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, 2004.

[8] M. Seck, I. Magrin-Chagnolleau, and F. Bimbot. Experiments on speech tracking in audio documents using gaussian mixture modeling. *In IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 601–604, 2001.

[9] D. Siegmund. *Sequential analysis*. Springer-Verlag, 1985.

[10] R.W. West and R.T. Ogden. Continuous-time estimation of a change-point in a poisson process. In *Technical Report 185*. University of South Carolina Department of Statistics, 1994.