T-outlier and a novel dimensionality reduction framework for high dimensional financial time series

D. Wang[1], P. J. Fortier[2], H. E. Michel[2] & T. Mitsa[2]
[1]*State Street Corporation, U.S.A.*
[2]*Department of Electrical and Computer Engineering, University of Massachusetts Dartmouth, U.S.A.*

Abstract

Outlier trading strategy is one of the more popular strategies in the financial markets. In reality, the outlier trading strategy tends to provide a higher return and is accompanied by a respective higher risk. Even so, more fund managers and investment companies turn to outlier trading strategies to maximize their return. The key is to minimize the risk by finding the most accurate and relevant outliers.

The framework presents a T-outlier concept in high dimensional time series data space. The framework generalizes the characteristics of time series outliers and proposes a novel dimensionality reduction framework in high dimensional time series data space of financial markets to support further mining. The framework consists of horizontal dimensionality reduction and vertical dimensionality reduction algorithms. On the vertical dimension, an Attribute Selection algorithm reduces the vertical dimension by identifying significant dimensions among a given set of attributes that closely represent the data characteristics. On the horizontal dimension, a Segmentation and Pruning algorithm reduces the selected attributes' horizontal data points to concisely represent the data object in piecewise linear representation form. This paper discloses a series of experiment results that illustrate the effectiveness of the framework.

Keywords: T-outlier, outlier, data mining, computational finance, dimensionality reduction, piecewise linear representation, high dimension.

WIT Transactions on Modelling and Simulation, Vol 43, © 2006 WIT Press
www.witpress.com, ISSN 1743-355X (on-line)
doi:10.2495/CF060311

1 Introduction

An outlier is "…an observation that deviates so much from other observations as to arouse suspicion that it was generated from a different mechanism."[1]. Outlier analysis strives to detect data objects whose characteristics are dissimilar from the rest of the objects in a given data collection. Outlier analysis can be categorized into distribution based [2], depth [3] based, and distance based approaches [4, 5]. Recently, additional outlier detection algorithms have been proposed for high dimensional data [6], spatial outlier detection [7], and OLAP based outlier data mining [8].

In this paper, we propose a definition of T-outliers (outliers in time-series data) and a framework to detect outliers efficiently in high dimensional space. The framework is composed of an Attributes Selection (AS) algorithm and a Piecewise Linear Representation (PLR) based a Segmentation and Pruning (SP) algorithms. The Attributes Selection algorithm utilizes the correlation coefficient and other statistics techniques to find significant attributes by applying a unique attribute weighting procedure. The identified significant attributes represent most of the data characteristics of a given data set. The Attribute Selection algorithm processes the vertical dimension reduction technique. Horizontally, we use the PLR based Segmentation and Pruning algorithms to concisely and precisely identify significant attributes' time dimensional data points – changing points.

The rest of the paper is organized as follows: Section 2 presents the problem statement to be addressed and summarizes the major contribution of this work. The background and related work is reviewed in Section 3. Section 4 describes the generalized T-outlier concept, the business similarity from an industry perspective using two definitions: changing point and object definition. Section 5 presents the data transformation algorithms including the Attribute Selection and PLR based Segmentation and Pruning algorithm. The experiments and result analysis are presented in Section 6, followed by Section 7 containing the conclusion and future work discussions.

2 Problem statement and contributions

Outlier trading strategy is a popular strategy in financial markets and tends to provide a higher return though with a respective higher risk. The key is to minimize the risk by finding the most accurate and relevant outliers.

Sophisticated tools are needed to help investors filter out outliers from hundreds or even thousands of stocks, based on their criteria. The goal is to develop a method that not only improves the efficiency in identifying outliers, but also provides a very high level of accuracy, thereby reducing the risk component in investment decision making processes.

The major contributions of the research presented are as following.

- We propose a general dimensionality reduction framework for time series data based on a data object transformation from high dimensional space to low dimensional space.

- Secondly, we propose novel horizontal and vertical dimension reduction algorithms that maintain significant data characteristics and concisely represent the data in the time axis.
- The resultant dimensionality reduction framework in high dimensional space can also be applied to other industries' data such as medical data, weather data, credit card usage data, and so on.

3 Background and related work

3.1 General outlier research

The outlier notion is formalized in a classic distance based approach by Knorr and Ng [4, 5]: In general, outlier detection research are classified into two categories. The first is distribution based, where a standard distribution is used to find the best fit with outliers defined thereafter. A major drawback of this category is that except for a handful of tests, most of the distributions used are single variant which does not match the reality of most problems.

The second category of outlier studies is referred to as depth based. Each data object is represented as a point in a k-dimension space, and is assigned a depth. Outliers are more likely to be data objects with smaller depths. In theory, depth-based approaches work for large values of k. However, depth-based approaches generally become inefficient for large datasets with $k \geq 4$ in Breunig [3], though there exist efficient algorithms for $k = 2$ or 3 in practice. Their notion generalizes many concepts from the distribution-based approaches, and enjoys better computational complexity than the depth-based approaches for large values of k. Recently, additional outlier concepts have been proposed such as high dimensional outlier [6], spatial outlier [7], and OLAP based outlier [8].

3.2 Dimensionality reduction research

The most promising data mining solution techniques involve dimensionality reduction with multidimensional index structures. Many dimensionality reduction techniques have been developed, including Singular Value Decomposition (SVD), Discrete Fourier transform (DFT), Discrete Wavelet Transform (DWT), and Adaptive Piecewise Constant Approximation (APCA). These are good horizontal dimensionality reduction techniques but can not solely be used for high dimensional data objects such as stock equities. The framework we presented is based on a Piecewise Linear Representation (PLR) of horizontal data with a vertical dimensionality selection algorithm to represent data objects. It produces satisfactory results in our experiments as is shown in section 7.

4 T-outlier concept

4.1 PLR based changing point

In this research, objects are assumed to move in a piecewise linear manner. Specifically, we assume that time is the horizontal continuous dimension and is

one dimension in the total data space. We extend the definition presented in Wu et al [9] to define the changing points including upper changing point and lower changing point which are symmetric. The lower changing point is defined here.

Definition 4.1: **Changing Point Definition**
Given the user specified time window *h* and current time t_c, a lower change point in Price dimension $P_j(\overrightarrow{D_{ij}}, t_j)$ at time j within user specified time period *h*, where i is the i^{th} attribute, should satisfy the following:

- $D_{price\text{-}j}$ = MIN(D_{price} value of current sliding window), where the sliding window can be various sizes which specify *m* as the maximum number of points contained after the last identified end point and *m* is given as a user specified parameter.
- $P_j(\overrightarrow{D_{ij}}, t_j)$ is the last point satisfying the above two properties.
- $\overrightarrow{D_{ij}} = (D_{1j}, D_{2j}, D_{3j}, \ldots D_{nj})$, where n is the total number of selected dimensions.

4.2 T-outlier concept

T-outliers consist of a subset of data that differs from a given super data set and exhibits four major features: periodicity, trend, predictability, and continuity. Continuity is the major differentiating factor when compared with other data mining fields, and also differentiate from streaming data.

We illustrate the T-outliers concept within the following scenarios:

First, consider only one object (stock object) in a two dimensional space (Time is the first dimension, the other dimension is Price). There are generally four types of unusual movements: isolated outliers, level shifts, slope change, and changes in frequency. An isolated outlier could be due to a sudden temporally localized change. A level shift represents a change in value that endures for at least a while and represents a new local mean value for the temporal data. A slope change can be unusual if the previous slope such as a high slope either decreases or increases. Changes in frequency means the frequency of change in the time series data. We believe all the above types apply to T-outliers in two dimensional spaces. We call these types of outliers: horizontal outliers because we only need consider the historical data of a particular object. Much research exist for horizontal T-outliers: such as burst in Zhu and Shasha [10], event in Guralnik and Srivastava [11], shocks in Jong and Penzer [12], or changing point in Yamanishi and Takeuchi [13].

Second, consider many objects (many stock objects) within two dimensional spaces. We define the outlier object as the object whose properties are dissimilar from remaining objects by comparing the recent period. The length of period is a user defined parameter *h*. The object property is denoted as $((t, \overrightarrow{D_1}, \overrightarrow{T_1}), t')$. In Figure 1:(c) we present only one segment to simplify the illustration. There are

two data points at t and t'. $\overrightarrow{D_1}$ and $\overrightarrow{D_2}$ are the normalized Price attribute value of the current point at time t' and the last end point of the price dimension at time t. $\overrightarrow{T_1}$ refers to the normalized degree of change associate with the point at time t. We call this type of time series outliers a two dimensional vertical T-outlier.

Third, consider many data objects within high dimensional spaces. The object property is denoted as $((t_1, \overrightarrow{D_1}, \overrightarrow{T_1}), (t_2, \overrightarrow{D_2}, \overrightarrow{T_2}) \ldots (t_n, \overrightarrow{D_n}, \overrightarrow{T_n}), t_0, t_c)$, where $\overrightarrow{D_1}$ is a vector which includes the normalized attribute value in the Price dimension and normalized values for other attributes. $\overrightarrow{T_1}$ is the normalized degree of change for the Price dimension. (T`-T) is the time difference which equals to the last end point time and current time. As shown in Figure 1:(d), there are multiple additional attributes associated with time t and t'. We should take all of these attributes into account when performing a similarity search to find the outliers. Figure 1, only shows one segment with two changing data points for simplicity with each changing point having three different attributes. Only one attribute has trend and the other two attributes only take the moving average value at a time stamp. There could be many segments and changing data points. We call this type of outlier as high dimensional vertical T-outliers. In this study, we explore this high dimensional vertical T-outlier.

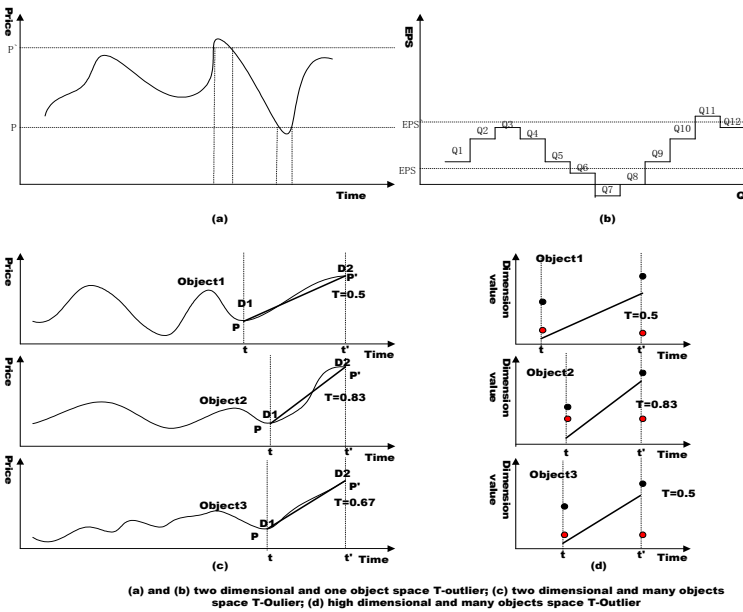

Figure 1: Sample T-outliers.

5 Dimensionality reduction framework

In this section, we will illustrate our vertical and horizontal dimensionality reduction algorithms.

5.1 Attribute Selection algorithm

It is important to know that the rational behind the algorithm is based on the correlation coefficient theory. Higher correlation coefficient value means higher co-dependence. Higher co-dependence means higher contribution to the related attribute. A high number of set co-dependence attributes however is problematic. We need to eliminate some attributes by performing further analysis.

Based on the rational above, we select the significant attributes in this study first and then filter out attributes by examining the co-dependence. Typically, the number of attributes ranges from 4 to 8 in financial market analysis for most of the research problems. We will try different numbers of attribute candidates in our experiments.

In our Attribute Selection algorithm we require the maintenance of a correlation coefficient matrix for the computation. For each stock object, we calculate the correlation coefficient value of all attributes against Price attribute (plast) and record the value in the matrix. While computing this, we perform pruning according to given conditions. Some stock objects are pruned because of its attribute(s) have 5 consecutive business days values equal to Null, or empty, or ∞. After completion of all computations for every stock object, we sort the values in ascending order. We then read the top 20 fields across all the stock objects, which are a matrix composed of only 20 rows and 500 columns. Finally, we select the attributes that represent 80% of occurrences in the new matrix. In addition, we use an order weighted concept to distinguish the contribution of the attributes having a high correlation coefficient value across different stock objects.

5.2 Segmentation and Pruning algorithm

Our PLR based Segmentation and Pruning algorithm with the selected attributes values is based on a user specified time period h. The algorithm can be separated into two parts. The first performs segmentation for each valid stock. The segmentation is based on solely b%. A b% time series is calculated before this algorithm in the Attribute Selection algorithm. Most of the b% values range from 0 to 1. Only in an occasional scenario the data varies slightly away from 0 or 1. The algorithm starts with the assumption that the first segment is upward. As differentiate from Wu et al [9], our algorithm can handle continuously up or down scenarios. Once the segments are identified as b% series, the time stamps of the changing points are mapped to the Price (plast) series to produce the segments in the Price series. The pruning process is performed in the second part. There are multiple steps of pruning.

First we prune on b% based on the given threshold. We proved in our experiments that a b% pruning threshold of 10~12.5% works for most of the

stock objects and yields adequate segments. The b% pruning process removes the changing points that do not qualify for the threshold and the time stamps associate with the changing points. The result is then mapped according to the new time stamps in Price series. If there is no need to perform b% pruning, the algorithm will prune by Price based on the Price pruning threshold. We proved that a Price (plast) pruning threshold of 10–15% works well for most of the stock objects. The result of the Segmentation and Pruning algorithm produce a PLR form of object which concisely and precisely represents the original Price series.

6 Business implications

6.1 Bollinger Bands and Bollinger Band Percent b%

Bollinger Bands are widely used indicators for American financial markets. The Bollinger bands measure relative definitions of the high and low values for financial time series for the Price dimension. The Bollinger bands are three curves drawn above and below a moving average using a measure of standard derivation. Bollinger Band Percent (b%) is another popular indicator which indicates the current state within the Bollinger bands. We perform segmentation using Bollinger bands in our Segmentation and Pruning algorithm. The reasons are: the b% has a smoothed moving trend similar to the Price movement; it represents a normalized value of the real price (between -1 and 2); and is very sensitive to the price change.

6.2 Attribute selection and construction

The selection of the initial attribute set was conducted with inputs from American financial domain experts and scholars from the financial markets. We selected three categories of attributes: market attributes, financial attributes, and environmental attributes. Besides the attributes we can directly obtain from the market, others were created to better measure stock performance. It is considered best practice in the financial domain to construct composite attributes to better measure stock from different perspectives. The market attributes are the variables gained in the stock market such as daily closing price, daily highest price, daily lowest price, and daily trading volume. In addition, we believe some variable manipulation is important to reflect characteristics of the market, i.e. 5 days average price which measures the relative price stability over a short period of time. The financial variables refer to the variables found in the financial balance sheet, income statement, and cash statements such as sales, revenue, earnings per quarter etc. The environment variables include federal bank interest rates, government behaviors, consumer market indicator, GDP, etc. We believe the Price attribute movement is highly correlated to the selected attributes.

6.3 Business aspect of the framework

It is a common practice to find a set of significant independent attributes for a given dependent attribute in the American markets. For instance, find a set of important attributes for implied volatility to support the option pricing model. The common ways are linear regression, multi-linear regression, and correlation coefficient analysis but is usually limited to one object with many attributes. The object here can be equity, option, and bond etc. It is a special challenge to find common significant attributes across objects. In this research, we present a unique attribute selection algorithm to identify common significant attributes across a pool of equities. In big financial institutions, there are terabytes of data sets for financial instruments with hundreds of attributes with over 20 years of historical daily (even intraday) data. The computation of such instruments becomes a nightmare for financial analysis which further supports other financial models in many cases in reality. We observed that there is no need to compute each data point of the object in some cases like outlier-ness calculation between data objects. We presented a novel PLR based Segmentation and Pruning algorithm to reduce the data points in time axis without loosing major data characteristics but with significant performance gain.

In addition, the novel framework presented is not specific to the U.S. equity market but represents a general framework to perform dimensionality reduction in high dimensional time series data. The framework can be applied to U.S. non-equity markets data, international financial markets data, weather time series data, and medical time series data etc. as long as the following properties are satisfied. First, the data object of the problem has a high number of attributes with long historical data in time series formats; second, part of the problem is to find the common significant attributes across objects; third, computational performance is important while not sacrificing result accuracy.

7 Experiment and result analysis

7.1 Raw data and attribute set

The Standard & Poors 500 Composite (S&P 500) stocks from BLOOMBERG© were selected as our raw data. We obtained 15 years of historical data with 95 attributes ranging from the pricing attributes to fundamental financial attributes. In order to adapt the environmental effect, also collected were 11 key economic indicators for the United States over those 15 years from DATASTREAM.

7.2 Experiment environment

The Windows XP platform is used on a machine with 1 GB memory, and a 1.4 GB Intel® Pentium processor. The program is implemented in the Java language with a set of property files.

7.3 Perform Attribute Selection and result analysis

First, cleaned data was generated by changing different values of moving average parameters whose value include 5 days, 10 days, 20 days, 30 days, 40 days, 50 days and 60 days respectively.

Second, for each different number of days scenario, we ran our Attribute Selection algorithm using the correlation coefficient calculation. Then selected the top fields (5~20 user specified) with the highest correlation value for each stock in each scenario. This produces a new matrix with 500 columns and 5~20 rows. The column represents the individual stock and the row represents the attributes for all stocks. We count the occurrence of each attribute in the new matrix and then sort the attributes in ascending order in terms of occurrence value. The algorithm then calculates the top attributes representing 80% of the total occurrences.

In order to analyze parameter significance, we used a PARETO chart. From analysis of the charts, we observed that the slope of the increasing part of the chart is very steep implying that most of the data characteristics are represented by the beginning attributes in the NAME axis. In testing, most of data characteristics could be represented by the top 5–10 attributes occurring in the matrix based on the observation.

Third, we calculate the percentage of occurrence for each attribute in the matrix by taking the occurrence of the attribute as nominator and the total cells of the matrix as denominator. We select the top attributes that represent user specified significant value of the data characteristics. We perform a business attribute analysis afterwards since the data object represents individual stocks in real financial markets and is part of standard practice within the financial industry. Based on both the scientific and business attribute analysis, the following attributes are selected: P2bk, P2sl, and P2ebita are the market data to financial data ratio attributes which reflect some fundamental financial data of the company. P2bk is the price to book value. It presents the current trading price to the company's equity value in the book. P2sl is price to company quarterly sale revenue which reflects company revenue ability. P2ebita is the price to earning before income tax amortization which reflects the profitability ability. pxcncel is related to number of trading get cancelled which is indirectly related to trading volume. plow, phigh, popen, and lsttrd are market trading related attributes which represent the stock market characteristics of the stock. ewapx is the monthly equity weighted average trading prices and dpavg is a constructed attribute which represents the past 20 business days moving average of the price.

7.4 Piecewise Linear Representation via Segmentation and Pruning algorithm

Cleaned data was then processed by our PLR segmentation and pruning algorithm using different periods of days (5–60) to find out which values best simulate the original data. In our experiment, we found that a 20 day period moving average yields good results. The segmentation pruning threshold (segbpcntThreshold) over a b% value of 10% works best for all scenarios. Most

of the b% values fall between 0 and 1. Therefore we can use standard values such as 10% as the threshold. We varied segmentation and pruning thresholds (segplastThreshold) from 0%, 2%, 5%, 7.5%, 10%, 12.5%, 15%, 17.5%, 20%, to 22.5%. The experimental results show that 12.5% and15% yields better results.

The example of STT (State Street Corp.)for the period from January 3, 2000 to December 31, 2003, Figures 2–5 shows a detailed comparison. As we can see from the figures intuitively, Figure 3: precisely represents the raw data object from Figure 2 without losing any major data characteristics. Figure 3: is the result of performing segmentation on b% series and mapping the time stamps that identified b% back to the Price dimension. The logic is explained in the Segmentation and Pruning algorithm. Also, the data in Figure 3: produced an appropriate zigzag shape according to our theory.

Since the b% values fall between 0 and 1, we use a uniform threshold for pruning our object based on the b%. A b% of 10% is proven to be suitable for most of the stock objects [9]. We prove here through our experiments that this is true. As we can see in Figure 4 and 5, the data gets further reduced concisely, without losing any major characteristics.

Based on the observations from experiments we can conclude two important rules. First rule, the degree of simulation accuracy depends on the plast pruning threshold. The smaller the threshold the closer simulation though it yields more segments. More segments adds more complexity to the HAC clustering and yields fewer similar objects but with higher similarity. The second rule, the user specified time period *h* is balanced with the plast pruning threshold. The shorter period of h requires a smaller plast pruning threshold to produce a reasonable amount of segments. The longer period of h requires a larger plast pruning threshold to produce segments. The algorithm itself is flexible to the user.

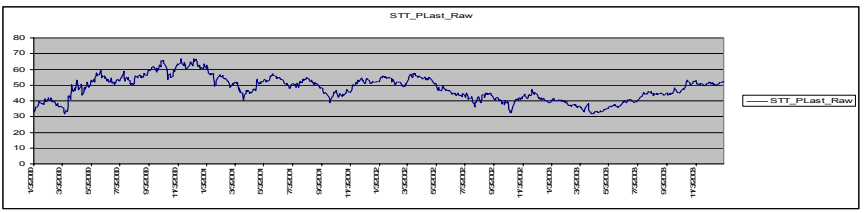

Figure 2: The raw plast data stream of STT.

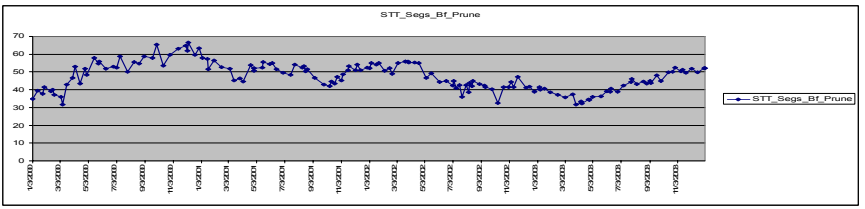

Figure 3: Segmentation results of raw data stream without any pruning based on sliding window size 10 and average moving period 20 days.

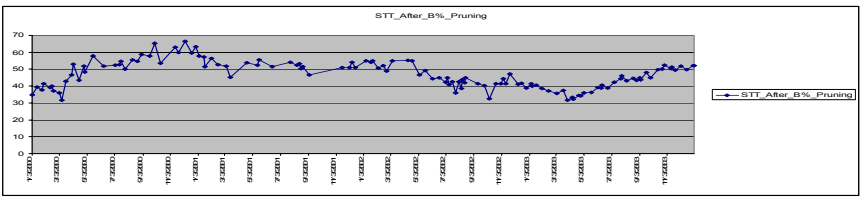

Figure 4: The result after perform b% pruning before performing plast pruning(b% = 10%).

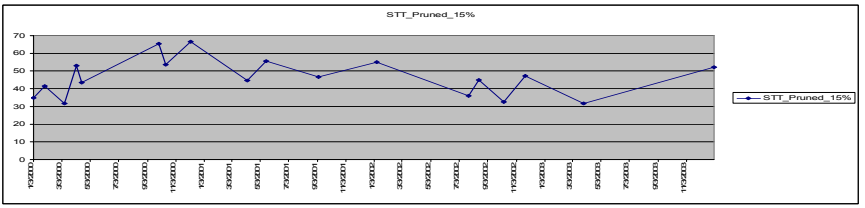

Figure 5: The pruning threshold of plast is 15%.

8 Conclusion and future work

In this study, we introduced a new T-outlier concept and a novel dimensionality reduction framework based on a Segmentation and Pruning algorithm over high dimensional financial data along with an Attribute Selection algorithm. Experiments show that vertical and horizontal dimension reduction algorithms are able to present the objects in concise and accurate form for further mining in the future.

The experiments illustrate that our algorithms are able to reduce the dimensionality efficiently within a large set of high dimensional data and could be used as foundation of developing more sophisticated tools.

This research focused on static time series data. Extending this research to include moving data, may improve predictive capabilities. The second extension looks to perform clustering and classification to find out regularities among large set of data objects based on data representation transformed by this framework. Additionally, we can perform prediction analysis based data object representation by this framework. Lastly, the framework itself can also be extended to other time series applications such as weather outlier detection, POS transaction outlier detection, network intrusion outlier detection and so on.

References

[1] Hawkins, D.M., Identification of outliers. Chapman and Hall, London, 1980.

[2] Barnett, V., and Lewis, T., Outliers in Statistical Data (Wiley Series in Probability & Mathematical Statistics). John Wiley and Sons Ltd. February, 1994.
[3] Breunig, M.M., Kriegel, H.P., Ng, R.T., and Sander, J., LOF: identifying density-based local outliers, The 2000 ACM SIGMOD international conference on Management of data, Pages: 93 - 104, 2000.
[4] Knorr, E.M., and Ng, R.T., Algorithms for Mining Distance-Based Outliers in Large Datasets, The 24th International Conference on Very Large Data Bases, Pages: 392 - 403, 1998.
[5] Knorr, E.M., and Ng, R.T., Finding Intensional Knowledge of Distance-Based Outliers. The VLDB Journal, Pages: 211-222, 1999.
[6] Aggarwal, C.C., and Yu, P.S., Outlier Detection from High Dimensional Data, The 2001 ACM SIGMOD international conference on Management of data, Pages: 37 – 46, 2001.
[7] Shekhar, S., Lu, C. T., and Zhang, P., A Unified Approach to Spatial Outliers Detection. University of Minnesota - Computer Science and Engineering Technical Report Abstract, December 10, 2001.
[8] Lin, S., and Brown, D.E., Outlier-based Data Association: Combining OLAP and Data Mining, Department of Systems and Information Engineering, University of Virginia, Charlottesville, VA, http://www.sys.virginia.edu/techreps/2002/sie-020011.pdf.
[9] Wu, H., Salzberg, B., and Zhang, D., Online Event-driven Subsequence Matching over Financial Data Streams, ACM SIGMOD international conference on Management of data, Pages: 23 - 34, 2004.
[10] Zhu, Y., and Shasha, D., Efficient elastic burst detection in data streams, The ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Pages: 336 - 345, 2003.
[11] Guralnik, V., and Srivastava, J., Event Detection from time series data, The fifth ACM SIGKDD international conference on Knowledge discovery and data mining, Pages: 33 - 42, 1999.
[12] Jong, P.D., and Penzer, J., Diagnosing Shocks in Time Series. Journal of the American Statistical Association, 93, Pages: 796 - 806, 1998
[13] Yamanishi, K., and Takeuchi, J., A unifying Framework for Detecting Outliers and Change Points from Non-Stationary Time series Data. The eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Pages: 676 - 681, 2002.