iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 Issue: IV Month of publication: April 2020

DOI:

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue IV Apr 2020- Available at www.ijraset.com

Outlier Pattern Detection in Time Series Sequences using Standard Deviation and Mean

Ritubara Chauhan[1], Yash Parashar[2]
[1, 2]Assistant Professor, Indore Institute of Science and Technology, India

***Abstract:** Outlier pattern detection is one of the important data mining tasks. In general everything follows a particular pattern. Finding this patterns in time series data is known as pattern mining. Outlier is unusual or surprising pattern which occur rarely in datasets. Event or sequence of events are frequent when they have frequencies greater than defined frequency. This defined frequency is usually the mean of frequencies corresponding to each pattern. This mean estimator is sometimes a biased estimator, and gives incorrect results.*

Therefore, suffix tree based approach using standard deviation is used. This is pattern searching approach and concerned with the frequency of patterns. This approach uses (mean-3*standard deviation) for the comparison of pattern's frequency instead of mean. As per the (68-95-99.7) rule of the statistics, 99.7 % of the values of the data are in the three standard deviations of the mean. The three-sigma rule states that at least 88.8% accuracy can be achieved even for non-normally distributed variables. Therefore proposed approach is better in eliminating wrong patterns and finding the accurate outlier patterns. Four different datasets have been used. The existing method detects some wrong candidate outlier patterns, which are eliminated in the proposed approach. Results show that 64.27% wrong patterns are eliminated as compared to existing methods with mean; in addition, 16% improvement in time is also achieved over the existing method.

I. INTRODUCTION

Time Series data is one which is recorded at the same interval of time or regularly. The data can be weather records, transaction summery of a store, medical report of patient, road or network traffic, stock price movement, gene expressions etc. The data is same as temporal data. Time series data is used to discover some hidden knowledge which is not possible using simple queries like SQL.

"Pattern Mining" refers to the method of data mining, which is to search for patterns in data. It helps in taking strategic decisions used to predict future events and patterns. Patterns which are unusual or irregular are outliers or surprising patterns. Detection of outliner patterns is more important as compare to a regular one in various areas like fraud detection, weather forecasting, unusual ECG heart beat etc. Outliers can be of many types for instance, in a certain sequence event 'a' and 'b' might not be outliers but a sequence 'aba' might be an outlier sequence. There is many work already has been done in this area. STNR (suffix tree based noise resilient) is most popular algorithm among them. The pattern which are of low frequency (number of appearance) are considered as outliers. There are 3 types of periodicity there i.e. symbol, sequence and segment.

II. RELATED WORK

A. Pattern Mining

In [1] Ashis Kumar, Chanda et al. proposed FPPM (Flexible Periodic pattern mining) previous methods have shown that there is a huge amount of candidate generation. The proposed approach ignores trivial or undesirable events, not treating them as term. This is also used to extract all the 3 types of periodicity i.e. symbol, segment and sequence periodicity. Paper [2] proposed an algorithm which removes noise presented in the data patterns. This deals with all types of noise presented in data. In [3] Comparison of different algorithms like WRAP, CONV, ParPer, and STNR for finding periodic patterns has been shown. STNR can detect all type of periodicity such as symbol, segment and partial periodicity. By using CONV only symbol and segment periodicity is detected. Only segment periodicity is detected by using WRAP. And ParPer discovers only fractional periodicity. From the comparison of those 4 algorithms it is concluded that STNR is found to be the most efficient and resourceful algorithm among all. In [4] proposed a method to efficiently mining similarity profiled temporal association patterns using FP-tree.it is based on creating FP-tree which is usually substantially smaller than the original database and thus save the cost of subsequent mining process.

B. Periodic Outlier detection

In [5] numerous procedures for detecting outlier patterns are examined. Which are discussed below - In [6] Base Algorithm - Fu et al. proposed Base algorithm based on some properties of Haar wavelet transform, Time series dissonances are continuance of a long time series which are at most unlike to all the other of the time series subsequence. Subsequence comparisons are ordered using

Haar transform for effective pruning. In this algorithm, all the possible candidate subsequence in outer loop are extracted, the interval to the nearest non-self-match for each candidate subsequence is found in inner loop. The candidate pattern which has the longest distance to its nearby non-self-match is declared as discord. At each stage of this algorithm different assumptions are followed. This algorithm is sub-ordered as the Outliers classification. In [7] Heuristic Discord Discovery - Keogh et al. proposed a simple algorithm, Heuristic Discord Discovery. This algorithm is three to four magnitude fast as compare to brute force that was competently finding discords. The efficiency of the discord detection algorithm was tested by Keogh et al. on 82 various time-series datasets from different domains. Three possible heuristic strategies: magic, random and perverse are pronounced which is surveyed by Approximation to Magic Heuristics by via SAX (Symbolic Aggregate Approximation). This algorithm falls under Subsequence as Outliers classification.

In [7] Tarzan Algorithm - Keogh et al. define a soft match version where the frequency of sample P in Database D is defined using the largest number, which means that each subsequent length of L occurs at least once in D. Surprising patterns can be found using the Tarzan algorithm on dataset which contains the power demand for a Dutch research facility for the entire year of 1997.. This algorithm uses a suffix tree that efficiently encodes the frequency of all observed patterns, and the Markov model allows the prediction of the frequency of the previously observed patterns. First the suffix tree is built. A revelation can be measured for all samples in the new database. The amount of time and space required is simple in database size. Tarzan is not an acquaintance. The name is given because the heart of the algorithm is based on the comparison of two suffix trees. This algorithm falls under the outlier Subsequence in the test time series.

InfoMiner Algorithm -In [8] Yang, R., Wei Wang, and Philip S. Yu. Infominer+: mining partial periodic patterns with gap penalties discover surprising periodic patterns. Their surprise prioritizes those patterns involving less frequency and more support. Support means matching repetition. A new kind of measurement was introduced here. It was called information, which values the degree of surprise of each occurrence of a pattern. Information treats occurrence as a continuous and monotonically decreasing function of its probability of occurrence. Thus patterns with different probability occurrences are handled easily. This information gain concept can address the adverse effects of closed asset infringement through an information gain measurement. It therefore provides an effective solution to this problem. This algorithm falls under the outlier Subsequence in the test time series.

Besides the above, Chuah et al. [9] proposes a variance detection scheme based on time series analysis that allows computers to detect if there is any abnormal heart rate in the real-time sensor data flow. If there is a discrepancy, the time series is transmitted to the physician through the network so that he can diagnose the problem and take appropriate action. The Adaptive Windows Based Discord Discovery scheme that the authors designed was motivated by the two schemes Brute Force Discord Discovery (BFDD) and the Heuristic Discord Discovery (HDD) schemes. [10] Used sub-series join to obtain the similarity relationships among sub-series of the time series data. Then the anomaly detection problems can be converted to graph theoretic problems solvable by existing graph theoretic algorithms.

In [11] Han et al. have detected partial periodic patterns (ParPer) by mining association rule i.e. a pattern is said to be a frequent partial pattern in the time series if its confidence is more or equal as compared to a threshold min+ confidence The effective mining of partial periodic patterns is executed by authors for only a single period as well as for a set of periods.

In [12] Elfeky et al. proposed periodicity detection algorithm based on convolution (CONV). There were 2 types of periodicity have been detected by the author. The first is segment periodicity and the other is symbol periodicity The concept behind this algorithm for segment periodicity detection was to practice the idea of convolution in order to compare and shift the time series for all the most possible values of that period.

In [13] Rasheed et al. proposed Suffix tree Noise Resilient algorithm (STNR) for detecting all types of periodicity. In this process numerical data is decomposed into digital data. The suffix tree representation for numerical data is then developed. The tree is quoted to give the event vector of a substance. The periodicity is given the difference in occurrence positions.

In [14] Chitharanjan et al. have studied various periodicity finding algorithms and done comparison with among four above definite algorithms [8]. CONV gives best time result in comparison to WARP, ParPer and STNR.

[15] Pujeri et al. have suggested a Constraint Based Periodicity Mining methodology where the periodicity is mined for recurrent patterns based on specific constraints on the development of a FP (Frequent Pattern) Tree.

In [16] Huang et al. offered their algorithm for obtaining asynchronous periodic patterns, where the periodic happenings can be shifted in an acceptable range within the time axis.

In [17] F. Rasheed and R. Alhaji et al. associated their work to obtain better outcomes with the InfoMiner algorithm.

In [18] E.Keogh et al. classify subsequence as outliers on 82 various time-series data-sets using symbolic comprehensive estimate.

In [19] Archana et al. presented a transient overview on various different methods for outlier pattern detection in time series data.

III. BACKGROUNDS

Here, we start with the basic terminologies of periodic pattern. This project is based on STNR algorithms. To understand the proposed approach one need to understand some important terms used in this project

A. Time Series Data

Time Series is a sequence of data points occurred on a regular basis or in a uniform time interval. Time series is a data related to the time and it can be easily represented by graphs. Level of employment measured every month can be considered as an example of time series. Fig 1 shows a simple representation of time series in graph.

B. Periodicity Detection

Periodicity mining is used to predict the behaviour of time series sequences. There are two types of periodicities which are symbol and segment periodicity. While segment periodicity concerns for the periodicity of the whole time series, values or symbol of the time series.

C. Outlier Detection

Outlier patterns are different from other patterns. They are periodic but comparatively they occur less frequently in datasets. For example, the pattern X = ab with period p = 7 is a improved candidate for the outlier pattern in the order.

D. Mean and Standard Deviation

Mean is one of the most widely held events in statistics, used with both continuous and discrete datasets. Mostly used with continuous frequent data. The “mean” is equal to some of all values divided by numbers of all values in dataset.

$$\bar{x} = \frac{(x_1 + x_2 + x_3 + \cdots + x_n)}{n}$$

The formula is usually written slightly differently using the Greek uppercase, clear “sigma”, which means “addition of…”

$$\bar{x} = \frac{\sum x}{n}$$

Standard Deviation is a measure of how much the points are spread within datasets. Formula for the standard deviation formula is:

$$s = \sqrt{\frac{\sum(X - \bar{X})^2}{n - 1}}$$

Where, s = sample standard deviation $\sum$ = sum of...
$\bar{X}$ = sample mean n = number of scores in sample.

IV. ALGORITHM TO DETECT PERIODIC OUTLIER PATTERN

The sequence of proposed approach -

A. Discretization Process

It transforms the time series into a series consisting of a finite set symbols. For example, consider the time series containing the hourly number of transactions in a superstore; For example, consider a time series in a supermarket with the number of transactions per hour; The following mapping is defined by discretization processes taking into account the various limitations of the transaction: 0 transactions: a, f1 to 200g transactions: b, f201 to 400g transactions: c, f401 to 600g transactions: d,f>600g transactions: e. Based on this mapping, the time series T = 243,267,355,511,120,0,0,197 it can be discretized into T’= cccdbaab.

B. Periodicity Detection

A time series is discretized into a finite set of alphabets which results in a string s of length n. fmax is the maximum possible repetitions (or frequency) of the pattern X within the range i_{st} and ending at position i_{end} with period p in string s.

$$f_{max} = \frac{(i_{end} + 1 - |X| - i_{st})}{p} + 1$$

$$con\ f(X, i_{st}, i_{end}\ ) = \frac{f}{f_{max}}$$

f= actual number of repetitions (frequency) in the given range A periodic pattern X is said to be frequent if its confidence is greater than or equal to a user-defined threshold confine.

C. Optimization

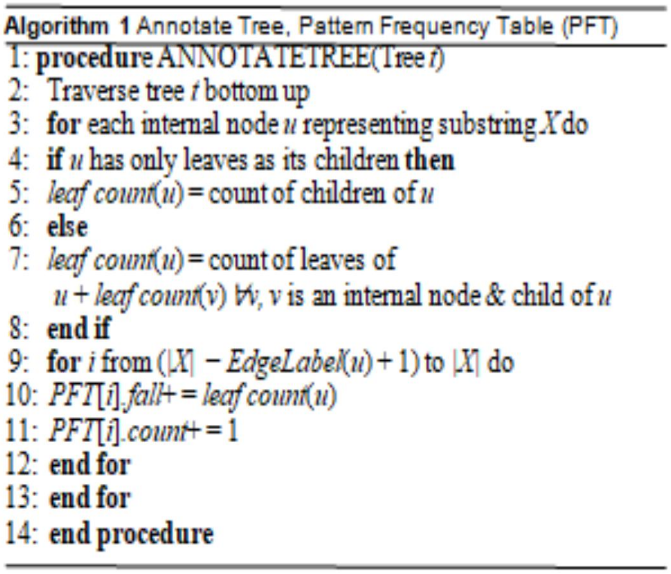
Most existing algorithms expect the user to manually identify which of the reported periods are useful. And hence, non-redundant. We use several parameters to allow the user to specify his preferences and to remove unnecessary redundant periods. (ex. - minSeglen, dmax).

D. Outlier Periodic Pattern

Candidate outlier pattern as the one which is less frequent than the patterns with same length.

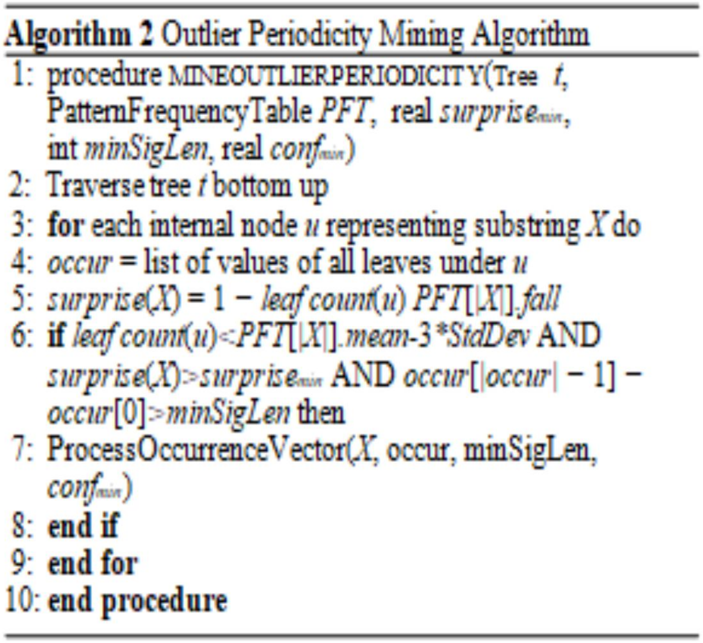
E. Periodicity Detection for Outlier Patterns

1) Build a suffix tree for the input sequences; 2) annotate the suffix tree such that each internal node records the length of substring it represents (the string obtained by tracing from the root till the node) and the frequency of the substring in the sequence; 3) Build a pattern frequency table (PFT) for recording the frequency of different length (up to the maximum pattern length); 4) Identify the candidate outlier pattern;



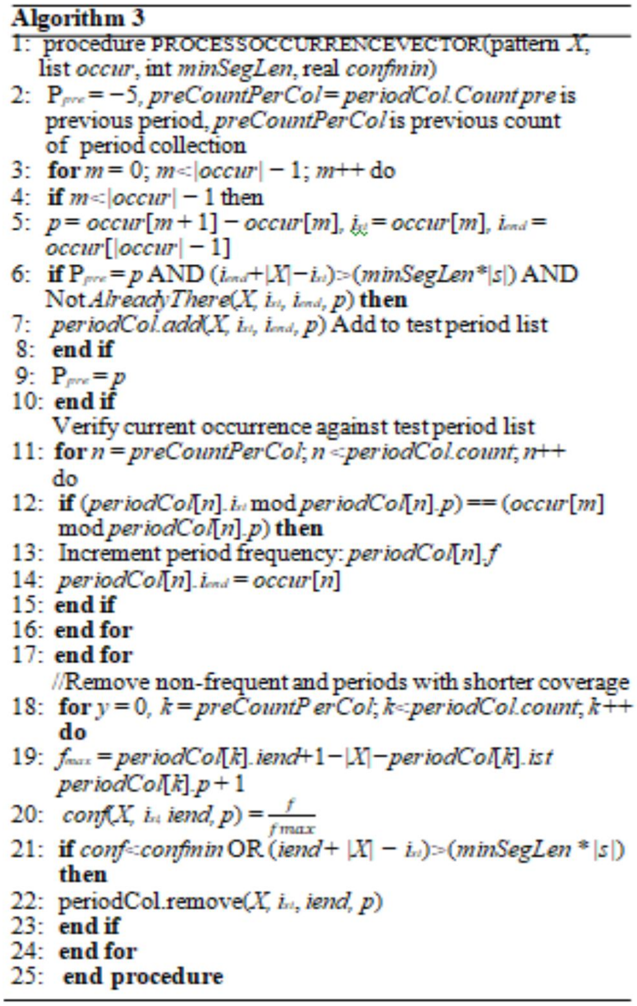
Algorithm 1 Annotate Tree, Pattern Frequency Table (PFT)

```
procedure ANNOTATETREE(Tree $t$)
    Traverse tree $t$ bottom up
    for each internal node $u$ representing substring $X$ do
        if $u$ has only leaves as its children then
            $leaf\ count(u)$ = count of children of $u$
        else
            $leaf\ count(u)$ = count of leaves of
              $u + leaf\ count(v)\ \forall v$, v is an internal node & child of $u$
        end if
        for $i$ from $(|X| - EdgeLabel(u) + 1)$ to $|X|$ do
            $PFT[i].fall + = leaf\ count(u)$
            $PFT[i].count + = 1$
        end for
    end for
end procedure
```



Algorithm 2 Outlier Periodicity Mining Algorithm

```
procedure MINEOUTLIERPERIODICITY(Tree $t$,
  PatternFrequencyTable $PFT$, real $surprise_{min}$,
  int $minSigLen$, real $conf_{min}$)
    Traverse tree $t$ bottom up
    for each internal node $u$ representing substring $X$ do
        $occur$ = list of values of all leaves under $u$
        $surprise(X) = 1 - leaf\ count(u)\ PFT[|X|].fall$
        if $leaf\ count(u) < PFT[|X|].mean\text{-}3*StdDev$ AND
          $surprise(X) > surprise_{min}$ AND $occur[|occur| - 1] -$
          $occur[0] > minSigLen$ then
            ProcessOccurrenceVector($X$, occur, minSigLen,
              $conf_{min}$)
        end if
    end for
end procedure
```



Algorithm 3

```
procedure PROCESSOCCURRENCEVECTOR(pattern $X$, list $occur$, int $minSegLen$, real $confmin$)
    $P_{pre} = -5$, $preCountPerCol = periodCol.Count$ $pre$ is previous period, $preCountPerCol$ is previous count of period collection
    for $m = 0; m < |occur| - 1; m{+}{+}$ do
        if $m < |occur| - 1$ then
            $p = occur[m+1] - occur[m]$, $i_{st} = occur[m]$, $i_{end} = occur[|occur| - 1]$
            if $P_{pre} = p$ AND $(i_{end} + |X| - i_{st}) > (minSegLen * |s|)$ AND Not$AlreadyThere(X, i_{st}, i_{end}, p)$ then
                $periodCol.add(X, i_{st}, i_{end}, p)$ Add to test period list
            end if
            $P_{pre} = p$
        end if
        Verify current occurrence against test period list
        for $n = preCountPerCol; n < periodCol.count; n{+}{+}$ do
            if $(periodCol[n].i_{st}$ mod $periodCol[n].p) == (occur[m]$ mod $periodCol[n].p)$ then
                Increment period frequency: $periodCol[n].f$
                $periodCol[n].i_{end} = occur[n]$
            end if
        end for
    end for
    //Remove non-frequent and periods with shorter coverage
    for $y = 0$, $k = preCountPerCol; k < periodCol.count; k{+}{+}$ do
        $f_{max} = periodCol[k].iend + 1 - |X| - periodCol[k].ist$ $periodCol[k].p + 1$
        $conf(X, i_{st}, iend, p) = \frac{f}{fmax}$
        if $conf < confmin$ OR $(iend + |X| - i_{st}) > (minSegLen * |s|)$ then
            periodCol.remove($X$, $i_{st}$, $iend$, $p$)
        end if
    end for
end procedure
```

TABLE I: An Example of a Table

One	Two
Three	Four

It is recommended that to use a text box for inserting a graphic (that should be ideally a 300 dpi EPS or TIFF file, including all fonts set in) as in a document, this technique is to some extent more stable comparatively directly inserting the picture.

Fig. 1: Amorphous magnetic core and reluctance of the oscillating winding on the DC bias magnetic field.

Figure Label: Use the 8 Point Times New Roman to label. Use words instead of symbols or acronyms when writing figure axis labels to avoid confusing the reader. For instance, write the quantity O`MagnetizationO´ , or O`Magnetization, MO´ , not just O`MO´. If the label contains units, display them in parentheses. Do not label only units with axes. For example, write O`Magnetization (A/m)´O or O`Magnetization A[m(1)]O´ , not just O` A/mO´ . Do not label axes with proportions of sizes and units. Such as, write ` OTemperature (K) ´ O, not ` OTemperature/K.´O

V. CONCLUSION

Pattern mining is a branch of data mining and many work has been already done in this field. Detection and estimation of outliner patterns can be described using mean estimation. In this proposed approach standard deviation for detecting candidate outlier is used. In this project suffix tree based method is proposed, First of all patterns are detected using this tree, then Pattern on the basis of their periodicity is detected and then Outlier candidates are recognized on the basis of some constraints.

A. Appendix

Appendixes should come before the acknowledgment.

VI. ACKNOWLEDGMENT

The ideal spelling of the expression ` Oacknowledgment´O in America is without an ` Oe´O after the O` gO´. Avoid the stilted expression, ` OOne of us (R. B. G.) thanks . . . ´O Instead, try O` R. B. G. thanksO´. Put supporter acknowledgments in the numberless annotation on the first page. The references are essential to the reader; hence, for each reference must be comprehensive and correct. If it is possible, references should be universally existing publications.

REFERENCES

[1] Chanda, Ashis Kumar, et al. ”An efficient approach to mine flexible periodic patterns in time series databases.” Engineering Applications of Artificial Intelligence 44 (2015): 46-63.

[2] Elfeky, Mohamed G., Walid G. Aref, and Ahmed K. Elmagarmid. ”WARP: time warping for periodicity detection.” Data Mining, Fifth IEEE International Conference on. IEEE, 2005.

[3] Chitharanjan, K. ”Periodicity detection algorithms in time series databases-a survey. ”International Journal of Computer Science and Engineering Technology” 1.4 (2013): 22-28.

[4] Agrawal, Mamta, Asha Ambhaikar, and Lokesh Kumar Sharma. “Efficient Similarity Profiled Temporal Association Mining using FPtree.” Database Systems: Proceedings of the International Conference on Computer Applications: 24-27 December 2010, Pondicherry, India. Research Publishing Services, 2010.

[5] Archana, N., and S. S. Pawar. ”Survey on Outlier Pattern Detection Techniques for Time-Series Data.” IJSR, December (2014).

[6] Fu, Ada Wai-Chee, et al. ”Finding time series discords based on Haar Transform.” International Conference on Advanced Data Mining and Applications. Springer, Berlin, Heidelberg, 2006.

[7] Keogh, Eamonn, Jessica Lin, and Ada Fu. “Hot sax: Efficiently finding the most unusual time series subsequence.” Data mining, fifth IEEE international conference on. Ieee, 2005.

[8] Yang, Jiong, Wei Wang, and Philip S. Yu. “Infominer: mining surprising periodic patterns.” Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001.

[9] Chuah, Mooi Choo, and Fen Fu. ”ECG anomaly detection via time series analysis.” Frontiers of High Performance Computing and Networking ISPA 2007 Workshops. Springer Berlin Heidelberg, 2007

[10] Lin, Yi, Michael D. McCool, and Ali A. Ghorbani. ”Motif and anomaly discovery of time series based on subseries join.” IAENG International Conference on Data Mining and Applications, ICDMA. 2010

[11] Han, Jiawei, Guozhu Dong, and Yiwen Yin. Efficient mining of partial periodic patterns in time series database. Data Engineering, 1999. Proceedings. 15th International Conference on. IEEE, 1999.

[12] Elfeky, Mohamed G., Walid G. Aref, and Ahmed K. Elmagarmid. ”Periodicity detection in time series databases.” IEEE Transactions on Knowledge and Data Engineering 17.7 (2005): 875-887

[13] Rasheed, Faras, Mohammed Alshalalfa, and Reda Alhajj. ”Efficient periodicity mining in time series databases using suffix trees.” IEEE Transactions on Knowledge and Data Engineering 23.1 (2011): 79-94.

[14] Chitharanjan, K. ”Periodicity detection algorithms in time series databases, a survey.” International Journal of Computer Science and Engineering Technology (2013).

[15] Pujeri, Ramachandra V., and G. M. Karthik. ”Constraint based periodicity mining in time series databases.” International Journal of Computer Network and Information Security 4.10 (2012): 37.

[16] Huang, Kuo-Yu, and Chia-Hui Chang. ”SMCA: a general model for mining asynchronous periodic patterns in temporal databases.” IEEE Transactions on Knowledge and Data Engineering 17.6 (2005): 774 - 785.

[17] F. Rasheed and R. Alhajj, “A Framework for s Outlier Pattern Detection in Time-series Sequences,” Cybernetics, IEEE Transactions on, vol. 44, pp. 569- 582, May 2014.

[18] E. Keogh, J. Lin, and A. Fu, “Hot sax: Efficiently finding the most unusual time series subsequence,” in Data Mining, Fifth IEEE International Conference on, pp. 8 pp., Nov 2005.

[19] Archana, N., and S. S. Pawar. ”Survey on Outlier Pattern Detection Techniques for Time-Series Data.”, IJSR, December 2014.

[20] Divya, D., and Suvanam Sasidhar Babu. ”Methods to detect different types of outliers.” Data Mining and Advanced Computing (SAPIENCE), International Conference on. IEEE, 2016.