ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(44), DOI: 10.17485/ijst/2016/v9i44/105161, November 2016* ISSN (Online) : 0974-5645

A Novel Approach to Outlier Detection using Modified Grey Wolf Optimization and k-Nearest Neighbors Algorithm

Reema Aswani[1], S. P. Ghrera[2] and Satish Chandra[3]

[1]Department of CSE, Ajay Kumar Garg Engineering College, Ghaziabad - 201009, Uttar Pradesh, India; Indiareemaswani@gmail.com
[2]Department of CSE and IT, Jaypee University of Information Technology, Waknaghat -173234, Himachal Pradesh, India; sp.ghrera@juit.ac.in
[3]Department of CSE and IT, Jaypee Institute of Information Technology, Noida - 201301, Uttar Pradesh, India; satish.chandra@jiit.ac.in

Abstract

Objectives: Detecting dataset anomalies has been an interesting yet challenging area in this front. This work proposes a hybrid model using meta-heuristics to detect dataset anomalies efficiently. **Methods/Statistical Analysis:** A distance based modified grey wolf optimization algorithm is designed which uses the k- Nearest Neighbor algorithm for better results. The proposed approach works well with supervised datasets and gives anomalies with respect to each attribute of the dataset based on a threshold using a confidence interval. **Findings:** The proposed approach works well with regression as well as classification datasets in the supervised scenario. The results in terms of number of anomalies and the accuracy using confusion matrix are depicted and have been evaluated for a classification dataset considering a minority class to be anomalous in comparison to the majority class. The results have been evaluated based on varying the threshold and 'k' values and depending on the data set domain and data distribution the optimal parameters can be identified and used. **Application/Improvements:** The proposed approach can be used for anomaly detection of datasets of different domains of supervised scenario. It can also be extended to unsupervised scenario by integrating it with K-means clustering.

Keywords: Data Mining, Grey Wolf Optimization, k-Nearest Neighbor, Machine Learning, Outlier Detection

1. Introduction

Real world data is dirty or coarse in nature and usually consists of a mix of normal and abnormal data points that do not conform to the data distribution[1-3], this fact is well documented in the literature. These abnormalities are usually termed as anomalies. The data available in huge amount is used for retrieving important information and thus would prove to give misleading results if these anomalies are not accounted for prior to any data analytics performed on the dataset. Outliers or anomalies are data points in the dataset that do not conform to normal behavior or which lie away from the range of majority of the data points in the dataset. These data points in most of the cases are questionable as well as actionable depending on the domain and the gravity of the problem. The process of detecting such distinguishable patterns in datasets is termed as anomaly detection or outlier detection. The data sets that we use now days for data analytics and business intelligence follow a kind of generating mechanism like a statistical process or distribution. But not all instances in the dataset follow the same and they deviate from this generating mechanism resulting into abnormal behavior or what we call as outliers. In[4] defined outliers as an observation which deviates from other observations to a level that arouses suspicions that the given data was generated by a different method. The outlier detection can be done

* *Author for correspondence*

in different application scenarios like Supervised, Semi-Supervised and Unsupervised. This paper focuses on the supervised scenario for outlier detection where training dataset is available. In supervised both classification and regression datasets are considered where most of the data instances are usually normal i.e. belonging to a particular class and a few anomalous data points embedded on it. Such methods have an added advantage of processing unlabelled data and finding aberrations which can otherwise go unnoticed in case of unsupervised learning mechanism. This scheme primarily works under the assumption that the number of normal instance is very large in number as compared to the anomalous instances and are also qualitatively differentiable. This paper follows a distance based approach that uses a hybrid model of k Nearest Neighbor (kNN) and Grey Wolf Optimization (GWO). The kNN approach for outlier detection was given by in[5] and was further enhanced by in[6]. In this paper we combine the kNN model with GWO to give better results in outlier detection. The GWO algorithm passes the position of wolves to the kNN function to check accuracy and hence produces optimized results.

Outlier detection algorithms usually work on statistical modeling techniques for supervised or an unsupervised scenario. Supervised scenario uses labeled training data sets to create an outlier data model. In the unsupervised scenario, there are direct techniques like statistical, proximity or density based and techniques. In this scenario training data sets are not available to us and thus the classification of objects as outliers is usually done by specifying statistical heuristics. Outlier detection has commonly been the area of research of a number of authors in various books, articles, surveys etc. Researchers from statistics have also done considerable amount of research in this field starting from towards the 20th Century and their works have been published in[7-9], and survey articles[10-12] have provided with an exhaustive survey of outlier detection techniques developed in the area of machine learning and statistics. Anomaly detection techniques for various types of datasets like numeric and symbolic are also very well discussed and illustrated in their work by in[13]. Some authors take anomalies as novelties and an extensive work on novelty detection in conjunction with neural networks is presented by in[14,15] where they have gone for statistical approaches to highlight them. The current research in this area also includes work by in[16] with discussion on unsupervised front of outlier detection where the authors use unlabelled data i.e. dataset instances without the output class for training, and detect instances that do not align themselves with the definition of normality. In[17]also discuss the statistical approach where the data attributes are partitioned into environmental and indicator attributes. Data is anomalous if its indicator attributes cannot be explained in the context of the environmental attributes. A few works unify set of rules for data quality that can include Functional Dependencies, Conditional and Matching Dependencies, Sequential and Order Dependencies which also proved to be useful in this front. In[18,19] detecting patterns, automating the process and applying the concepts to high dimensional datasets are the current additions in the same[20-23]. Since the work proposed in this paper is focused on nearest neighbor based approach for outlier detection. The following part of the section provides a summary of work which focuses on direct methods which are used in supervised scenario for labeled data using nearest neighbor techniques. These techniques require a distance or any similarity metric to be defined between the dataset instances. There are various metrics that can be different measures for the same. For continuous data attributes, Euclidean distance has emerged out to be very popular out of all distance measures. A lot of work has been done in this front and the major assumption for detecting anomalies using nearest neighbors is that normal data records occur in dense neighborhood areas, whereas anomalies occur far from the closest or nearest neighbor. In[24] give a very extensive review of the same in their survey article where they group nearest neighbor anomaly detection into two main categories namely: 1. where anomaly score is taken as distance of a data point from its k^{th} closest neighbor, and 2. Anomaly score is computed using relative density of each data point.

2. Meta Heuristic Otimization Techniques

Meta-Heuristic optimization techniques have become very common over the past few decades due to four main advantages that they have: simplicity, flexibility and avoidance of local optima. Thus, Grey Wolf Optimization[25] based approach gives us all of these advantages over any other Outlier detection approach.

2.1 Grey Wolf Optimization

This problem is inspired from the grey wolf which belongs to the Canidae Family, they are considered to be at the topmost point of the food chain since they are apex predators and thus it is interesting to study their behavior for survival. The Figure 1 shows the hierarchy and hunting of the grey wolves, these usually live in a pack of size 5-12 on an average. Both of these are of great interest and can be mathematically modeled to design GWO and perform optimization. The Grey wolf Optimizer proposed by in[25] uses a wolf population and calculates fitness for search agents and over iterations, it updates the wolf positions to hunt the prey. The pseudo-code in Pseudo-code 1 for the Optimizer provided by in his work depicts the approach in a formal manner. In the pseudo code for GWO given in Figure 2, the author first initializes the grey wolf population X_i which is the initial positions. Vectors a, A and C are used to update the wolf positions while hunting the prey over the Max number of iterations.

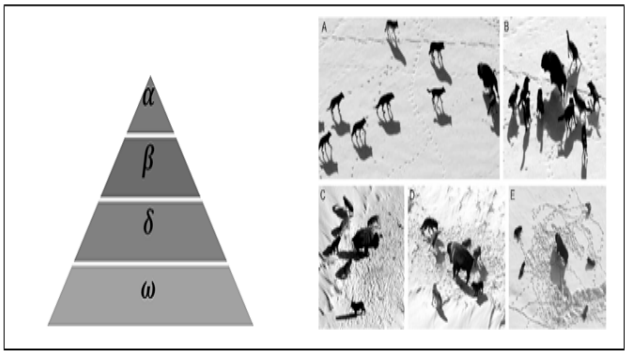

Figure 1. Hierarchy and group hunting of grey wolves.

Pseudo-code for GWO

Initialize the grey wolf population X_i *(i = 1, 2, ..., n)*
Initialize a, A, and C
Calculate the fitness of each search agent
X_α*=the best search agent*
X_β*=the second best search agent*
X_δ*=the third best search agent*
while *(t < Max number of iterations)*
  for *each search agent*
    Update the position of the current search agent
  end for
  Update a, A, and C
  Calculate the fitness of all search agents
  Update X_α, X_β, *and* X_δ
  t=t+1
end while
return X_α

Figure 2. Grey wolf optimizer pseudocode.

3. Proposed Approach

The proposed approach uses a combination of the concept of grey wolf optimization by in[25] Pseudo-code 1 and k-Nearest Neighbor algorithm. The discussion about optimization problems always includes an objective function where the motive is to optimize the given objective function. Thus, to apply any optimization technique the first step is to identify and select an objective function. In the discussed problem of anomaly detection, for the proposed approach of Distance Based Grey Wolf Optimization for anomaly detection, the objective function is the accuracy of k Nearest Neighbor Algorithm. The paper uses a modified version of GWO where the positions of search agents X_α, X_β, and X_δ are not updated in every iteration. The α value i.e. the best positions for the wolves is used and is obtained from the kNN algorithm, β, δ and ω become the second, third and rest of the positions respectively. The proposed approach is depicted by the Pseudo-code 1.

3.1.1 Pseudo-code for Modified GWO using kNN Algorithm

Begin
Initialize the grey wolf population
Initialize Test Data as Number of grey wolves (NumGW)
for each grey wolf NumGW
  /*calculate Accuracy and Position of the grey wolf */
/* i=1 Iteration 1 */
  R ← random number (between 1 to NumGW) is generated
  R is checked from a database of numbers so that R should not repeat
  /* Take data points one by one from TestData as position*/
Position ← TestData(R)
  Pass Position to kNN as input and get kNN Output
  Accuracy ← Target Value – kNNOutput
  Best_Value ← Acuuracy
  Best_Position ← Position
/* i=2 Iteration 2 */
for each grey wolf NumGW
  if (Best_Value > Acuuracy)
  then

```
                Best_Value=Accuracy
                Best_Position=Position
            End
End
```

Pseudo-code 1. Modified GWO using kNN

The calculation done is the fitness of X_α which is the fitness for the best search agent. Again over an iteration, the best score and best positions i.e. best 'n' independent variables are found, where 'n' is the number of attributes in that dataset. The reason behind this is that in the proposed approach, α is pre calculated using the kNN by optimizing its accuracy and this makes the β, δ and ω solutions insignificant in this context. First, distances between data points are calculated and k neighbors are considered. After the kNN algorithm where the distances are calculated, kNN Accuracy is computed, which is the absolute difference between the Target value to be achieved and the value obtained from the kNN. The input to kNN is the test set or the positions of 30 wolves each with 'n' columns. So, basically 30 data points are passed to the kNN algorithm one by one, each having 'n' values corresponding to 'n' attributes or columns of the dataset. Since, kNN is to find the best set of values (position of wolves) for anomaly detection and a data point of 'n' dimensionality, i.e. a row with 'n' columns as the (*Target-Output)* which is the minimum value is obtained. Thus, this becomes a minimization problem where the objective function for the optimization problem is to minimize the value of *Target- Output* or simply the kNN accuracy. The best set of values that we get as output from the kNN becomes the α positions for the GWO algorithm where the prey is hunt or simple where the best accuracy of kNN is achieved. Once, the 'n' values are obtained from the above algorithm the Euclidean Distance of all the values of each of the rows from that value for all 'n' columns in our training data is taken. Based on a threshold, if the distance is greater than the threshold, that data point is taken as anomaly with respect to that particular attribute/ column of the dataset. The process is repeated for all 'n' attributes and individual anomalies are found out with respect to each best position of that attribute found using the algorithm. The total number of anomalies is the sum of all the data points acting as anomalies with respect to one or more than attributes of that dataset. Calculation of threshold is being done using a confidence interval. The proposed algorithm calculates the best position which is take as the α position (best search agent in GWO). Distance between data points is calculated taking best position and by putting a threshold, the anomalies are calculated. The flowchart in Figure 3 summarizes the proposed approach by taking the number of wolves as 30. The wolves are chosen randomly from the dataset.

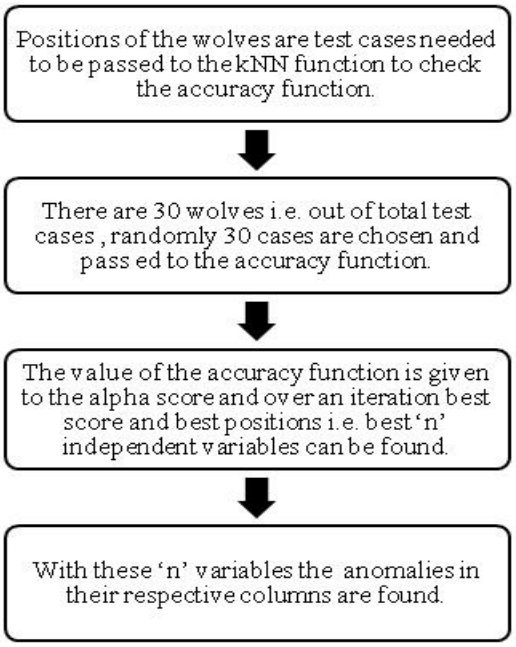

Figure 3. Flowchart for modified GWO using kNN.

3.2 Confidence Interval

A confidence interval estimates a population parameter which consists of a range of values that give a good estimate of the parameter that is unknown and has to be found. It does not describe any single sample, rather the entire population and the value is depicted by a percentage, like by an interval of 99%,it means there is 99% confidence that the true value of the unknown population parameter lies in the specified range of confidence interval. For the purpose of this work a confidence interval of 95% is taken which means there is a significance level of 0.05.The population parameter is taken as the distances of the best position for an attribute from all other instances of that attribute.

4. Implementation

4.1 Experimental Setup

The program was implemented in MATLAB R2013b. The code for GWO was taken from the author's official

page. The kNN function used was the ClassificationKNN. fit provided by MATLAB. The datasets were loaded in the form of .mat files and the entire processing was done in the code. The distance metric used for finding the k nearest Neighbors and the distance between data points for anomaly identification was Euclidean Distance[26] as shown in equation 1.

$$d = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

For n data points distance d is calculated between each par of points x_i and y_i .The data points in blue color and anomalies in red color with respect to each attribute in the data set were plotted. For a data set containing 'n' attributes 'n' graphs were plotted one for each attribute. The x- axis tells the number of instances in each plot. The number of plots for each dataset is equal to the number of attributes, one per attribute. Since, the proposed approach deals with supervised scenario, the last column, i.e. the output in case of regression and class label in case of classification are not taken into consideration. Each data point is plotted for all attributes in the respective plots. The best position which was obtained from the Modified GWO Algorithm becomes the criteria for distance calculation. The distance between each data point and the best position is computed. Based on a selected threshold and confidence interval, the data points are either marked as normal points (in blue) or outliers (in red).

4.2 Results

The Figure 4 shows the anomaly plots for the abalone dataset from UCI Repository[27]. The dataset has 4177 instances and 8 attributes, used for predicting the age of abalone from different physical measurements. The attributes used in this case are Length, Diameter, Height, Whole, Shucked, Viscera and Shell weight and Rings. The eight plots below show anomalies with respect to each attribute of the abalone dataset using the algorithm. The anomalies identified for each attribute are marked in red whereas the normal points are marked in blue color. Categorical values if any are converted to numerical by allotting a number to each category. For example in case of the first graph which is for the attribute Sex of nominal type having values M, F, and I (infant) is converted into integer by allotting 0,1 and 0.5 to each of them and thus we have not considered anomalies for the first column while calculating total number of anomalies. The next dataset on which the algorithm was tested was Combined Cycle Power Plant from UCI Repository[27]. The dataset contains 9568 data points collected from a Combined Cycle Power Plant over a period of 6 years, with the power plant set on full load work. The attributes comprise of Temperature, Pressure, Humidity and Exhaust Vacuum. The four attributes are used to predict the net hourly electrical energy output of the plant. The graphs show anomalies for each of the attributes. The total number of anomalies comes out of 38. Figure 5 shows anomaly plots for the dataset.

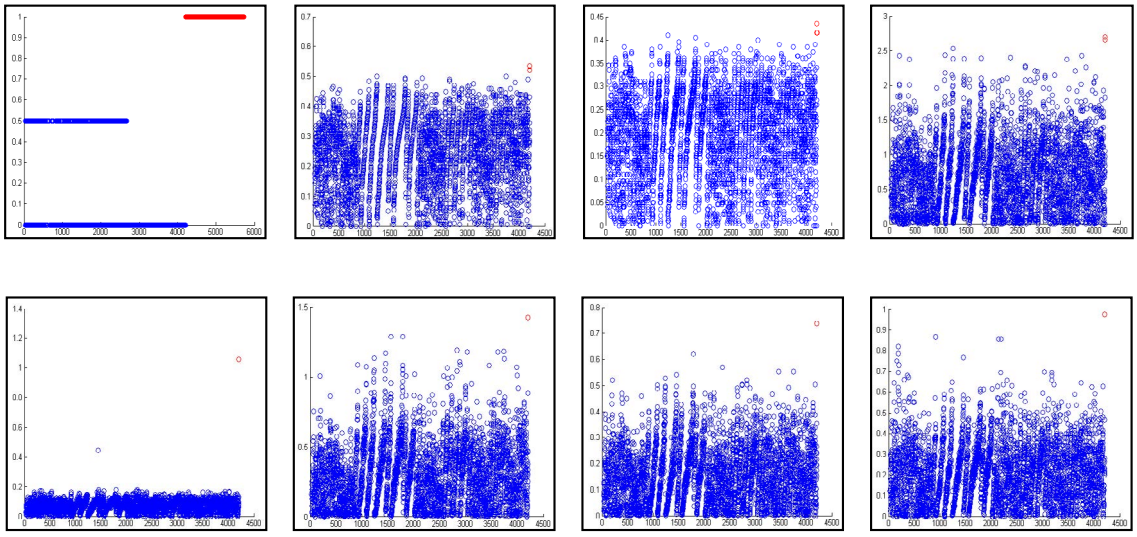

Figure 4. Outlier plots for Abalone dataset (Eight plots, one for each attribute of the dataset).

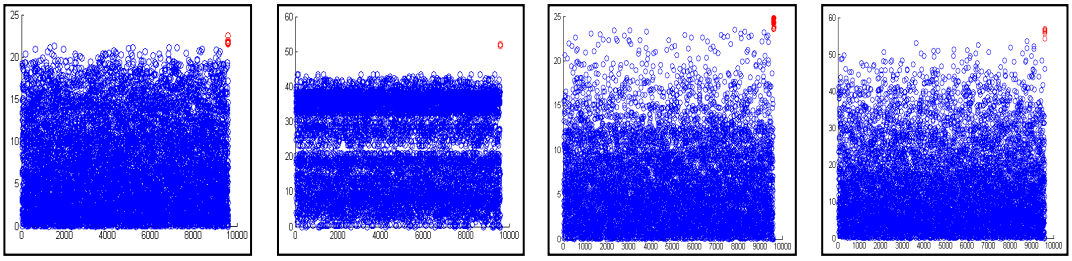

Figure 5. Outlier plots for combine cycle power plat dataset, one per attribute using modified GWO-kNN algorithm.

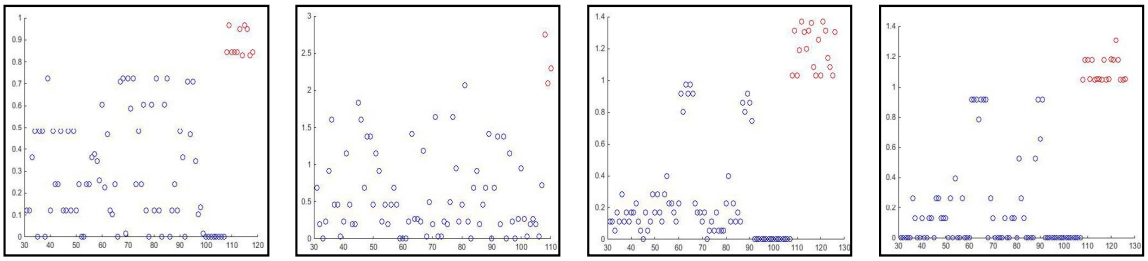

Figure 6. Outlier plots for Iris dataset considering only 80 instances (50 from one class and 30 from other, marked as outliers).

4.3 Analysis of Results

The algorithm was tested by applying on classification dataset for Iris from UCI Repository[27]where for the data points of one class, the rest of the two classes are considered to be anomalous. The algorithm gave correct results on the dataset and is useful for regression and classification anomaly detection by using appropriate threshold and k values for kNN. The 50 rows from the Iris Setosa are taken and 30 rows from the Iris Versicolour class making the latter a minor one. Under the assumption that with respect to Iris Setosa data points the Versicolour should come out to be anomalous since they lie farther. The test proved to be correct and a total of 30 anomalies are calculated by the proposed approach out of 80 total data points. The plots for the four attributes sepal length and width, petal length and width are shown in Figure 6. Table 1 summarizes the effect of varying threshold and 'k' values on the accuracy of anomalies detected. 'N' is the number of anomalies detected. As seen, on varying the k value there is no effect on the number of anomalies detected from k=5 to k=10. But for abalone on varying from k=1 to k=5 the results have substantial improvement. But on varying the threshold value the accuracy changes.

The accuracy calculated is done taking into account the number of true positive and true negative values in the confusion matrix or contingency Table. The false positive and false negative values are not considered while taking into account the number of anomalies detected by the system for accuracy calculation as per equation 2. Seeing the results, this becomes a domain specific problem to improve accuracy beyond a point as the distance between classes plays an important role in analyzing the threshold and 'k' values.

$$Accuracy = \frac{\sum True\ Postive + \sum True\ Negative}{\sum Total\ Population} \quad (2)$$

The No Free Lunch Theorems[28] says that no meta-heuristic approach can be best suited for solving all problems in optimization. In other words, it can be understood that a particular meta-heuristic approach may show excellent results on a problem, but the same approach might give poor results on a different set of problems. Considering the theorem, the proposed approach can be applied on various applications of anomaly detection to explore the results.

Table 1. Data analysis with varying threshold and 'k' values

Dataset	Threshold	k=1		k=5		k=10	
		GWO-kNN		GWO-kNN		GWO-kNN	
		Accuracy	N	Accuracy	N	Accuracy	N
Iris	0.65	96.67	34	96.67	34	96.67	34
	0.75	80	28	80	28	80	28
	0.85	56.67	19	56.67	19	56.67	19
Abalone	0.5	70	61	90	61	90	61
	0.65	30	27	53.33	40	53.33	40
	0.75	23.33	19	26.67	18	26.67	18

5. Conclusion

The proposed algorithm is applied on variety of datasets having attributes and instances ranging from low to high in number. The algorithm works well with regression as well as classification datasets in the supervised scenario. The results in terms of number of anomalies are depicted and have been evaluated for a classification dataset considering a minority class to be anomalous in comparison to the majority class. The algorithm successfully calculates the outliers on the basis of given threshold.

6. Future Scope

The proposed approach can be extended to be used for unsupervised learning scenario by integrated the approach with K-Means Clustering algorithm. The approach may also be scaled for big datasets and integrated with Map Reduce Framework for better and faster results. Domain knowledge can be included to produce more accurate and reliable results. Categorical datasets may also be included for optimization.

7. References

1. Hernandez MA, Stolfo SJ. Real-World Data is Dirty: Data Cleansing and the Merge/Purge Problem, Data Mining and Knowledge Discovery.1998 Jan; 2(1):9–37.
2. Bell G, Hey T, Szalay A. Beyond the Data Deluge, Science. 2009 Mar; 323(5919):1297–98.
3. Swartz N. Gartner Warns Firms of Dirty Data, Information Management Journal. 2007; 41(3):1–6.
4. Hawkins D. Identification of Outliers, Chapman and Hall, London, New York, 1980.
5. Ramaswamy S, Rastogi R, Kyuseok S. Efficient Algorithms for Mining Outliers from Large Data Sets, ACM SIGMOD Record USA. 2000 Jun; 29(2):427–38.
6. Angiulli F, Pizzuti C. Fast Outlier Detection in High Dimensional Spaces. Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery, UK. 2002 Sep, p. 15–26.
7. Angiulli F, Pizzuti C. Outlier Mining in Large High-Dimensional Data Sets, IEEE Transactions on Knowledge and Data Engineering. 2005 Feb; 17(2):203–15.
8. Rousseeuw PJ, Leroy AM. Robust Regression and Outlier Detection. New York, NY, USA, John Wiley & Sons, 1987.
9. Barnett V, Lewis T. Outliers in Statistical Data. 3rd Edition, Chichester: John Wiley & Sons. 1994.
10. Beckman RJ, Cook RD. Outliers, Technometrics. 1983; 25(2):119–49.
11. A comparative Study for Outlier Detection Techniques in Data Mining. Date Accessed: 7/06/2006. Available at: http://ieeexplore.ieee.org/document/4017846.
12. Hodge V, Austin J. A Survey of Outlier Detection Methodologies, Artificial Intelligence Review. 2004 Oct; 22(2):85–126.
13. Agyemang M, Barker K, Alhajj R. A Comprehensive Survey of Numeric and symbolic Outlier Mining Techniques, Intelligent Data Analysis. 2006 Dec; 10(6):521–38.
14. Markou M, Singh S. Novelty Detection: A Review-Part 1: Statistical Approaches, Signal Processing. 2003 Dec; 83(12):2481–97.
15. Markou M, Singh S. Novelty Detection: A Review-Part 2: Neural Network Based Approaches, Signal Processing. 2003 Dec; 83(12):2499–521.
16. Das K, Schneider J. Detecting Anomalous Records in Categorical Datasets. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA. 2007 Aug, p. 220–29.
17. Conditional Anomaly Detection. Date Accessed: 26/03/2007. Available at: http://ieeexplore.ieee.org/document/4138201/.
18. Fan W, Geerts F, Li J, Xiong M. Discovering Conditional Functional Dependencies, IEEE Transactions on Knowledge and Data Engineering. 2011 May; 23(5):683–98.
19. Szlichta J, Godfrey P, Gryz J. Fundamentals of Order Dependencies, Proceedings of the VLDB Endowment. 2012 Jul; 5(11):1220–31.

20. Discovery of Complex Glitch Patterns: A Novel Approach to Quantitative Data Cleaning. Date Accessed: 11/04/2011. Available at: http://ieeexplore.ieee.org/document/5767864/.
21. Lee YJ, Yeh YR, Wang YCF. Anomaly Detection via Online Oversampling Principal Component Analysis, IEEE Transactions on Knowledge and Data Engineering. 2013 Jul; 25(7):1460–70.
22. Lan Z, Zheng Z, Li Y. Toward Automated Anomaly Identification in Large-Scale Systems, IEEE Transactions on Parallel and Distributed Systems. 2010 Feb; 21(2):174–87.
23. Niu K, Huang C, Zhang S, Chen J. ODDC: Outlier Detection Using Distance Distribution Clustering, Springer Berlin Heidelberg, 2007 May, p. 332–43.
24. Chandola V, Banerjee A, Kumar V. Anomaly Detection for Discrete Sequences - A Survey. IEEE Transactions on Knowledge and Data Engineering, 2012 May; 24(5):823–39.
25. Mirjalili S, Mirjalili SM, Lewis A. Grey Wolf Optimizer, Advances in Engineering Software. 2014 Mar; 69:46–61.
26. Danielsson PE. Euclidean Distance Mapping, Journal Computer Graphics and Image Processing. 1980 Nov; 14(3):227–48.
27. UCI Machine Learning Repository. Date Accessed: 13/11/2015. Available at: http://service.re3data.org/repository/r3d100010960.
28. Wolpert DH, Macready WG. No Free Lunch Theorems for Optimization, IEEE Transactions on Evolutionary Computation. 1997 Apr; 1(1):67–82.