 ISSN: 2321-8169
3077 – 3080

Effective And Efficient Approach for Detecting Outliers

M.Sowmya Tanuja
Computer Science Engineering
UCEK, Kakinada
Andhra Pradesh, India
e-mail: tanuja1225@gmail.com

A.Krishna Mohan,Associate Professor
Computer Science Engineering
UCEK, Kakinada
Andhra Pradesh, India
e-mail: Krishna.ankala@gmail.com

***Abstract*—** Now a days in machine learning research anomaly detection is the main topic. Anomaly detection is the process of identifying unusual behavior. It is widely used in data mining, for example, medical informatics, computer vision, computer security, sensor networks. Statistical approach aims to find the outliers which deviate from such distributions. Most distribution models are assumed univariate, and thus the lack of robustness for multidimensional data. We proposed an online and conditional anomaly detection method based on oversample PCA osPCA with LOO strategy will amplify the effect of outliers. We can successfully use the variation of the dominant principal direction to identify the presence of rare but abnormal data, for conditional anomaly detection expectation-maximization algorithms for learning the model is used. Our approach is reducing computational costs and memory requirements.

Keywords-*PCA; LOO strategy; online updating ; power method; Anomaly detection*

*****

I. INTRODUCTION

Anomaly detection aims to identify the outliers which deviates from the existing data. Mainly we observe some small instant of data which is different from other observation or other data because of that we may cause serious problems in the real world.

Practically these anomaly detection are widely used in homeland security cyber security intrusion detection, credit card detection etc.

We can't identify some kind of data which cause severe problems so for that unseen irregular data online anomaly detection is introduced in order to detect the anomalies or outliers in data. Here in this paper we are also using conditional anomaly detection. There are several data attributes which human can't directly identify as anomaly. Accuracy may also suffer if data attributes consider equally. Here for that reason we are introduction conditional anomaly detection. By using this all the attributes will treated equally while detecting the anomalies without high accuracy.

In some conditions these will not be correctly identify the outliers for large data sets for example data mean and least square calculation for linear regression are mainly fragile to outliers. So because of that reason osPCA(Online over sampling Principle component anomaly detection) is used.. In this we are using principle direction to detect the anomalies. We calculate the principle directions for adding and removing the data, by comparing those principle directions we can easily identify the anomalies. LOO strategy is used for calculating the principle direction in this we can take the principle with and without the target instance through which we can detect the anomalies with variation of principle direction. We can also consider the dPCA based detecting the anomalies but it not significantly work for large data sets. For conditionally anomaly detection we are using EM-based model this model will define that the anomaly is a indicator attributes are non identical to the environmental attributes.

II. BACKGROUND AND RELATED WORK

Mainly it deals with the outliers which are present in the real world entity. Many of them introduced many techniques and methods with they are confined to only small amount of data so that large data can't handle by these methods. For that reason we introduces a new technique in order to support large amount of data i.e real world data like credit card faults network intrusion etc. In this first we clean the contaminated data to normal after cleaning pattern extraction is used in order to extract the pattern after pattern is extracted we then detect the data whether is contain errors or not by using principle directions.

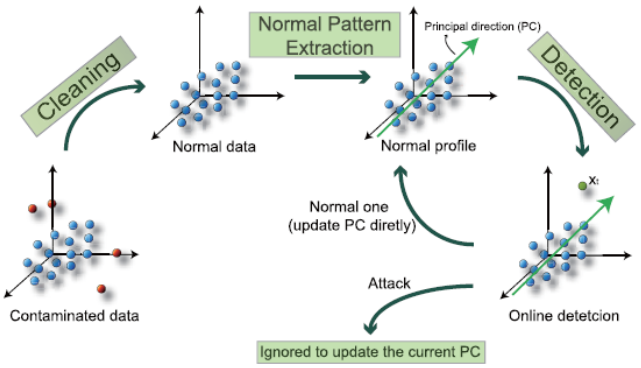

Figure 1: Framework of our approach

III. EXISTING SYSTEM

There are three major approaches for detecting the anomalies, distribution (statistical), distance based method and density based methods. Distribution approach is predetermined and follow some kind of standard and it is more suitable to that kind of distribution. For distance based approach we calculate the distance between each data instance and using that distance we can identify the anomalies. In Density based approach local outlier factor is used. LOF diagnose the outlierness

 ISSN: 2321-8169
3077 – 3080

which provides ranking through that we detect the anomalies. These models mainly expect as univariate so these models can't deal with multidimensional data and also these models may suffer with the noise present in the data.

IV. PROPOSED SYSTEM

PCA is used to deal with multidimensional data which determines the principle directions but the matrix storage can't easily extended and also it is not support for large databases. So for that reason we are using osPCA with online updating technique which can be suitable for large data. Here in this technique we can calculated the eigen vectors without the use of data covariance matrix.

A. *Working process*

Working process will have the following steps:

1st step: We must clean the data by using osPCA before start of the detecting phase. Here we can set some threshold value so that data can be cleaned based on that value.

2nd step: Now we can start our anomaly detection by using the threshold value. The newly added data will be compared with the given threshold if the value is beyond the value given then it will consider as anomaly.

After that we can also calculate the conditional anomalies for which we can consider the entire attributes. This will also consists of some steps

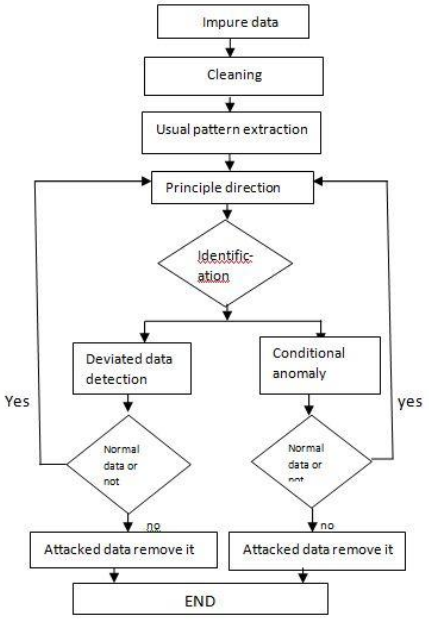

Figure 2: Working process of the method

B. *Explanaiton*

Adding one outlier will not effect the principle direction when the data that we are using is large. So we introduce this technique for large scale data problems.

In this we first calculate the principle direction for newly added data than the normal or original data. All the times it's not enough to calculate only the eigen vectors and leaving the remaining. Our method perform well without sacrificing the memory and computational cost. Instead of calculating eigen vector we can determine the dominant principle direction in online model. If u calculate the newly added data n times

$$\sum_{\tilde{A}} \tilde{u}_t = \lambda \tilde{u}_t$$

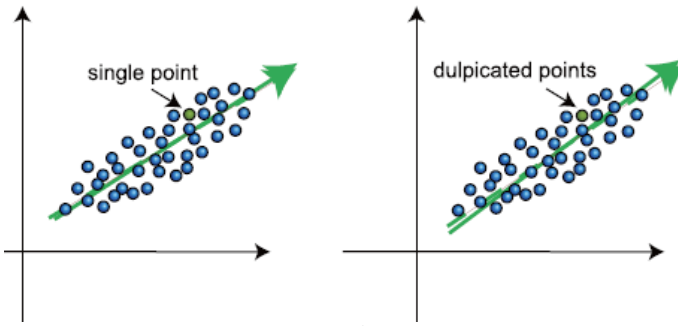

Figure 3: Principle direction of normal data

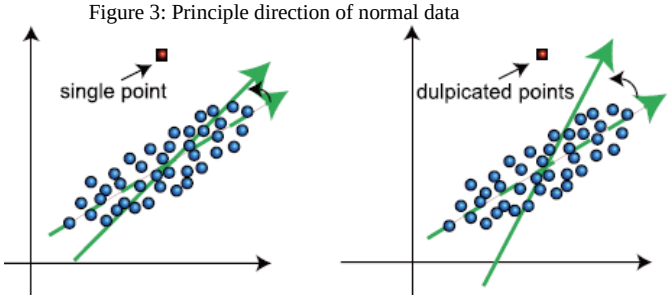

Figure 4: Principle direction of outlier data

C. *Power Method*

In many methods if we want PCA we need to find out the n principle directions for n instances which will drastically reduces the computational speed and also the memory. For this purpose we use power method which can find out the outlier were easily i.e if we duplicate the target instance we can easily find out the difference between the outlier and the normal data can be easily find out.

$$\tilde{\boldsymbol{\mu}} = \frac{\boldsymbol{\mu} + r \cdot \mathbf{x}_t}{1 + r}$$

and

$$\boldsymbol{\Sigma}_{\tilde{A}} = \frac{1}{1+r}\mathbf{Q} + \frac{r}{1+r}\mathbf{x}_t\mathbf{x}_t^\top - \tilde{\boldsymbol{\mu}}\tilde{\boldsymbol{\mu}}^\top,$$

Here we define the parameter r which reduces the size while oversampling. From the above equation we can keep only the matrix and its not necessary to create entire covariance matrix all the time.

Once the covariance matrix is calculated we can easily find out the principle directions which will solve the eigenvector problems which automatically leads to computational problems. So in order to avoid that problem we can use power method

$$\min_{\tilde{\mathbf{U}} \in \mathbb{R}^{p \times k}, \mathbf{U}^\top \mathbf{U} = \mathbf{I}} J_{ls}(\tilde{\mathbf{U}}) \approx \sum_{i=1}^{n} \|\bar{\mathbf{x}}_i - \tilde{\mathbf{U}}\mathbf{y}_i\|^2 + \tilde{n}\|\bar{\mathbf{x}}_t - \tilde{\mathbf{U}}\mathbf{y}_t\|^2.$$

ISSN: 2321-8169
3077 – 3080

TABLE 1 Comparing our method with the remaining

	osPCA [19] (power method)	Online osPCA	Fast ABOD [5]	LOF [2]
Computation complexity	$O(mp^2)$(or $O(mnp)$)	$O(p)$	$O(n^2p+k^2p)$	$O(n^2p+k)$
Memory requirement	$O(p^2)$(or $O(np)$)	$O(p)$	$O(np)$	$O(np)$

In this p and n are dimensionality and size of the data and also we have iterations m, k neartest neighbours.

If we want to increase the convergence rate we can use the normal data without oversample by using the following equation

$$\min_{\tilde{\mathbf{U}}\in\mathbb{R}^{p\times k},\mathbf{U}^\top\mathbf{U}=\mathbf{I}} J_{ls}(\tilde{\mathbf{U}}) \approx \beta\left(\sum_{i=1}^{n}\|\bar{\mathbf{x}}_i-\tilde{\mathbf{U}}\mathbf{y}_i\|^2\right)+\|\bar{\mathbf{x}}_t-\tilde{\mathbf{U}}\mathbf{y}_t\|^2,$$

Now we consider the least square problem which can be solved by approximating the x and y values so that we can write as follows

$$\min_{\mathbf{U}_t\in\mathbb{R}^{p\times k},\mathbf{U}^\top\mathbf{U}=\mathbf{I}} J_{ls}(\mathbf{U}_t)=\sum_{i=1}^{t}\|\bar{\mathbf{x}}_i-\mathbf{U}_i\mathbf{y}_i\|^2$$

By using this we can calculate the principle direction easily by using the approximation values and when it likned with our method we have

$$\min_{\tilde{\mathbf{U}}\in\mathbb{R}^{p\times k},\mathbf{U}^\top\mathbf{U}=\mathbf{I}} J_{ls}(\tilde{\mathbf{U}}) \approx \sum_{i=1}^{n}\|\bar{\mathbf{x}}_i-\tilde{\mathbf{U}}\mathbf{y}_i\|^2+\|\bar{\mathbf{x}}_t-\tilde{\mathbf{U}}\mathbf{y}_t\|^2$$

We can also change into the following problem

$$\min_{\tilde{\mathbf{U}}\in\mathbb{R}^{p\times k},\mathbf{U}^\top\mathbf{U}=\mathbf{I}} J_{ls}(\tilde{\mathbf{U}}) \approx \beta(\sum_{i=1}^{n}\|\bar{\mathbf{x}}_i-\tilde{\mathbf{U}}\mathbf{y}_i\|^2)+\|\bar{\mathbf{x}}_t-\tilde{\mathbf{U}}\mathbf{y}_t\|^2$$

D. Algorithm:

1.We have to calculate the principle direction u by considering approximate values of x and y

$$\bar{\mathbf{x}}_{proj}=\sum_{j=1}^{\bar{n}} y_j\bar{\mathbf{x}}_j \text{ and } y=\sum_{j=1}^{n} y_j^2$$

2. For loop i=1 to n do

$$\tilde{\mathbf{u}} \leftarrow \frac{\beta\bar{\mathbf{x}}_{proj}+y_i\bar{\mathbf{x}}_i}{\beta y+y_i^2}$$

$$s_i \leftarrow 1-\left|\frac{\langle\tilde{\mathbf{w}},\mathbf{w}\rangle}{\|\tilde{\mathbf{u}}\|\|\mathbf{u}\|}\right|$$

3.End the for loop.

V. EXPERIMENTAL RESULTS

A. Comparing with real world data

We verify the our algorithm take the example of real world data so here we can take some sample data sets which can be defined in the following table

TABLE 2 Comparison of real world data

Data set	Size	Attributes	Classes
pima	768	8	2
splice	1000	60	2
pnedigits	7494	16	10
adult	48842	123	2
cod-rna	59535	8	2
kdd_tcp	190065	38	5

Here we take the 90% of normal data and in order to find out the outliers we take 10% of outlier data. Here we repeat the same process with five trails and we calculate the average for those values. We can notice that the power method with our method the ratio r will vary from 0.1 to 0.2 while if we take the least square it will produce fast results and perform best from some comparisons.

Now take the another example like 2-D data and the results for that will be represented as

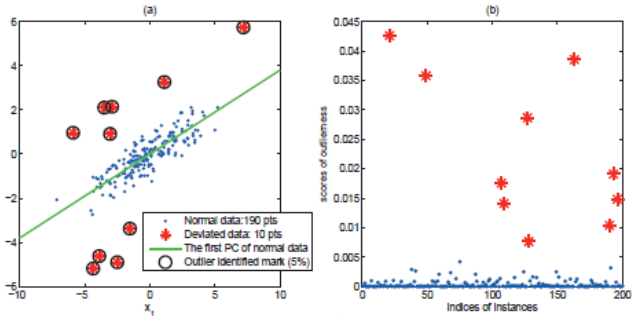

Figure 5: Example for 2-D data using our method

VI. CONCLUSION

In this paper we show how we detect the outliers from large amount of data using principle direction. By calculating the principle direction we can find out the effected data easily by using our method and also update the data without using the eigen vectors. It doesn't maintain any matrix and the eigen vector values in order to detect the errors or the anomalies using which we can reducing the computational speed. So it is mainly useful for large scale data like credit card fault or the network intrusion etc.

Further research will be done on the finding the data when we are using multiple clustering because as it is more complex in order to use the principle direction method in order to find out the error in multi clustered data. For high dimensional data we can come across the dimensionality problem and using PCA is

not an recommended approach so we have to go through all these study in the future scope.

REFERENCES

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

[2] M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in Proceeding of the 2000 ACM SIGMOD International Conference on Management of Data, 2000.

[3] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Computing Surveys, vol. 41, no. 3, pp. 15:1–58, 2009.

[4] L. Huang, X. Nguyen, M. Garofalakis, M. Jordan, A. D. Joseph, and N. Taft, "In-network pca and anomaly detection," in Proceeding of Advances in Neural Information Processing Systems 19, 2007.

[5] H.-P. Kriegel, M. Schubert, and A. Zimek, "Angle-based outlier detection in high-dimensional data," in Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 2008.

[6] A. Lazarevic, L. Ert¨oz, V. Kumar, A. Ozgur, and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," in Proceedings of the Third SIAM International Conference on Data Mining, 2003.

[7] X. Song, M.Wu, and C. J. andSanjay Ranka, "Conditional anomaly detection," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 5, pp. 631–645, 2007.

[8] S. Rawat, A. K. Pujari, and V. P. Gulati, "On the use of singular value decomposition for a fast intrusion detection system," Electronic Notes in Theoretical Computer Science, vol. 142, no. 3, pp. 215–228, 2006.

[9] W. Wang, X. Guan, and X. Zhang, "A novel intrusion detection method based on principal component analysis in computer security," in Proceeding of the International Symposium on Neural Networks, 2004.

[10] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-based detection and prediction of outliers," IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 2, pp. 145–160, 2006.

[11] V. Barnett and T. Lewis, Outliers in statistical data. John Wiley& Sons, 1994.

[12] W. Jin, A. K. H. Tung, J. Han, and W. Wang, "Ranking outliers using symmetric neighborhood relationship," in Proceeding of Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2006.

[13] N. L. D. Khoa and S. Chawla, "Robust outlier detection using commute time and eigenspace embedding," in Proceeding of Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2010.

[14] E. M. Knox and R. T. Ng, "Algorithms for mining distance based outliers in large datasets," in Proceedings of the International Conference on Very Large Data Bases, 1998.

[15] H.-P. Kriegel, P. Kr¨oger, E. Schubert, and A. Zimek, "Outlier detection in axis-parallel subspaces of high dimensional data," in Proceeding of Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2009.

[16] C. C. Aggarwal and P. S. Yu, "Outlier detection for high dimensional data," in Proceeding of ACM SIGMOD international conference on Management of data, 2001.

[17] D. Pokrajac, A. Lazarevic, and L. Latecki, "Incremental local outlier detection for data streams," in Proceeding of IEEE Symposium on Computational Intelligence and Data Mining, 2007.

[18] T. Ahmed, "Online anomaly detection using KDE," in Proceedings of IEEE conference on Global telecommunications, 2009.

[19] Y.-R. Yeh, Z.-Y. Lee, and Y.-J. Lee, "Anomaly detection via oversampling principal component analysis," in Proceeding of the First KES International Symposium on Intelligent Decision Technologies, 2009, pp. 449–458.

[20] G. H. Golub and C. F. V. Loan, Matrix Computations. Johns Hopkins University Press, 1983.

[21] R. Sibson, "Studies in the robustness of multidimensional scaling: perturbational analysis of classical scaling," Journal of the Royal Statistical Society B, vol. 41, pp. 217–229, 1979.

[22] B. Yang, "Projection approximation subspace tracking," IEEE Transaction on Signal Processing, vol. 43, pp. 95–107, 1995.

[23] S. Papadimitriou, J. Sun, and C. Faloutsos, "Streaming pattern discovery in multiple time-series," in Proceedings of the 31st international conference on Very large data bases, 2005.

[24] S. Haykin, Adaptive Filter Theory. Prentice Hall, 1991.

[25] A. Asuncion and D. Newman, "UCI repository of machine learning databases," 2007, http://www.ics.uci.edu/∼mlearn/mlrepository.html.

[26] A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," Pattern Recognition, vol. 30, pp. 1145–1159, 1997.