OUTLIER DATA MINING WITH IMPERFECT DATA LABELS

Mr.Yogesh P Dawange[1]

[1] *PG Student, Department of Computer Engineering, SND College of Engineering and Research Centre, Yeola, Nashik, Maharashtra, India*

ABSTRACT

Clustering is a technique that group a similar object in a cluster some objects are different .which differently behaves to identify data objects that are markedly different from or inconsistent with the normal set of data is done by the outlier detection. Most existing solutions build a model using normal data and also identify outlier that do not fit represented model very proper. However, in addition to normal data, there also exist some limited negative examples or outliers in many applications that data and information may be corrupted such that the outlier detection data is imperfectly labeled. It creates outlier detection very different than compared to that of traditional ones. To address data with imperfect labels and incorporate limited abnormal example into learning is done by a novel outlier detection approach we are implementing RBF kernel and SVDD for the outlier detection. We are combining these two things and generate a best output for outlier detection.

Keyword : - *Outlier, SVDD, Kernels*

1. INTRODUCTION

Outlier detection method refers to the problem of detecting ,analyzing, checking and observing patterns in data that does not map to expected normal behavior means it can be different from the other one . These patterns or observations are often referred to as outliers, anomalies, different observations or pattern, exceptions, noise, errors, damage, faults, defects, contaminants, surprise or peculiarities in different application domains or a field. Outlier detection has been a widely researched problem in a many fields . It finds important use in a wide variety of application domains such as insurance company , tax, credit card , ATM card fraud detection, military surveillance for enemy activities, fault detection in safety critical systems, intrusion detection for cyber security and many other areas . Outlier detection is important due to the fact that outliers in the data translate to significant information in a wide variety of application domains. For example, in a computer network security, an exceptional pattern could mean that a hacked computer is sending out sensitive data to an unauthorized receiver or a computer system.

Outliers in weather data is that in a summer season some days weather information can be a different from the other days weather information and this different days information is to be a outlier in a weather data of summer. Similarly, in public health data, outlier detection techniques are used to detect exceptional or different patterns in patient medical records which could be symptoms of a new disease. Outliers can also translate to some critical entities such as in exceptional readings from a space craft which a fault in some component of the craft. In military surveillance, the existence of an unusual portion in a satellite image of enemy area could indicate enemy troop movement. Outlier detection has been found to be directly applicable in a large number of domains. This has resulted in a huge and highly distinct literature of outlier detection techniques. Many of these techniques have been developed to solve focused problems pertaining to a particular application domain, while others have been developed in a more generic fashion. This survey deals with providing a structured and comprehensive sketch of the research done in the field of anomaly detection. The key aspects of any outlier detection technique are identified, and are used as dimensions to classify current techniques into different categories.

Outlier means "an observation or pattern which deviates so much from other observations or pattern as to arouse suspicions that it was generated by a different mechanism or techniques". Outlier is an observation point that is different from other observations Outlier detection refers to the problem of finding patterns in data that do not conform to expected or a normal behavior. Previous work done can be broadly classified into the following four types.

1.1. Statistics-based: This outlier detection techniques always fit a statistical model to the given data and then apply a statistical inference test to determine whether an unseen instance satisfies this model or not. In which Instances that have a low probability of being generated pattern from the learned model, based on the applied test statistic cases, are declared as outliers. For example, we can assume the normal examples follow a certain data distribution (such as Gaussian distribution), by estimating the parameter in the model, we can generate a Gaussian model to predict an unseen example into normal class or outliers. The statistics-based techniques always assume knowledge of the underlying distribution and estimate the parameters from the given data such as Gaussian model based , in which the pre-specified data distribution is assumed to fit a Gaussian distribution; regression model based , where outlier detection using regression has been extensively investigated for time-series data; mixture of parametric distributions based , in which techniques use a mixture of parametric statistical distributions to model the data. For this category, the main disadvantage is that these techniques rely on the assumption that the data is generated from a particular distribution. However, this assumption often does not hold true in many applications, especially for high dimensional real data sets.

1.2. Density-based: This outlier detection technique always assumes that normal data instances occur in dense neighborhoods, while outliers occur far from their closest neighbors. One representative method is called LOF (local outlier factor) , which assigns an outlier score to any given data point, depending on its distances in the local neighborhood. In LOF there is a distance of a object or a parameter is be calculated and then as per LOF value the object or parameter is to be considered as a outlier. If LOF value is large than a other object value then this is a outlier. Recently, the work proposed by improves the accuracy of outlier detection by calculating an outlier score based on a Gaussian mixture model (GMM). However, if the data has normal instances that do not have enough close neighbors or if the data has outliers that have enough close neighbors, the technique fails to label them correctly, resulting in missed outliers.

1.3. Clustering-based: This outlier detection technique mainly relies on applying clustering techniques to characterize the local data behavior. As per by-product of clustering, small clusters that contain significantly fewer data points than other clusters are considered as outliers. The performance of clustering based techniques is highly dependent on the effectiveness of the clustering algorithm in capturing the cluster structure of normal instances. In that clustering algorithm like K-means, kernel K-means and other algorithm are used.

1.4. Model-based: This outlier detection techniques are used to learn a model from a set of labeled data instances and then to classify a test instance into one of the classes using the learnt model. Model-based outlier detection techniques operate in a similar two-phase fashion. The training phase learns a classifier using the available labeled training data. The testing phase classifies a test instance as normal or anomalous using the classifier. In this category, SVDD proposed by has been demonstrated empirically to be capable of detecting outliers in various domains. Model based approaches can detect global outliers effectively for high-dimensional data without need to assume the prior distribution of data.

2. LITERATURE SURVEY

Conventional approaches for outlier mining are based on the type of application for which it is designed. As per the literature survey, to best of our knowledge many techniques use predefined distribution of data. Hence some of the current type of techniques does not work for unlabeled type of data. One of the approaches which deal with this type of data use clustering based technique. But this creates a problem when the data is not uniformly distributed. In order to cope up with this problem, density based approach was used to consider varied density problem. But major disadvantage is that it has high sensitivity to the setting of input parameters. Also it is unsuitable for high-dimensional datasets because of the curse of dimensionality phenomenon. Whereas, model based approach generates a model in order to define the behavior of the data. Model based approach generates a hypersphere of minimum volume in order to encapsulate all the normal data with minimum volume

3. PROBLEM STATEMENT

Imperfect class label is one of the difficulties that the supervised classification framework may confront. This issue of uncertainty in labels of the learning data dramatically degrades the classification performance. It is because of the importance of the labeled data in the learning procedure where the classifier tries to appropriately fit to the training data and make decisions about the class of the test sample based on the characteristics of learned data. This uncertain label assignment can occur when differentiating between two or more classes is not easy due to lack of information required for specifying certain labels to the data or the difficulty of labeling complicated data. This motivated proposed system to design classifier from the training set without assuming the prior distribution of data and to reduce the impact of uncertain data on classifier construction. And, thus to detect outliers with very few labeled negative examples and data with imperfect labels.

4. PROPOSED SYSTEM AND ARCHITECTURE

Our proposed approach falls into the model-based category, which is proposed to account for the challenge of outlier detection on uncertain data or imperfect data labels. More specifically, our method only determines the local uncertainty by generating a confidence score for each instance, which indicates the likelihood of this sample belonging to normal class, but also constructs a global outlier detection classifier. Have shown that our proposed approach outperforms state-of-art outlier detection algorithms in terms of performance and sensitivity to noise. Proposed system architecture is as follows

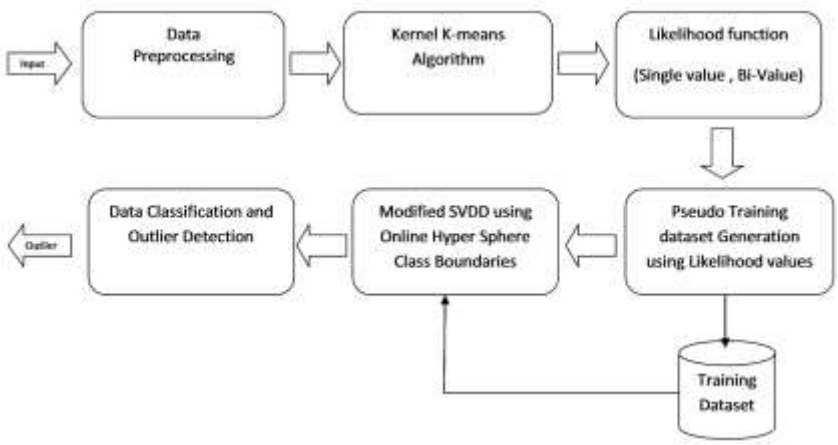


Fig : Proposed System Architecture

Following are the modules of the proposed system:

This system takes a query result page DP as input. After generating Pseudo training Dataset PT consisting of datasets N from using likelihood values. Liklihood Function are formed using input samples and their values. After extracting Liklihood Function values they are forwarded to KKM (Kernal K Means) Algorithm as a input. One or more features are used to minimize objective function and cluster data. Applying the Autonomous SVDD parameter tuning Class boundaries result are been extracted and then classification is been done and outliers has been detected.

4.1 Data Preprocessing

Data preprocessing includes setting the environment for running the proposed system. This includes initializing the dataset for processing. The module also sets the variables and parameters for the system and removes the noise and other things present in a data.

4.2 Kernel K-Means Algorithm

Kernel k-means clustering algorithm takes input as the set of data points and the count of number of clusters. It randomly initialises 'c' cluster center and compute the distance of each data label point and the cluster center in the transformation space. It then assigns a data point to that cluster whose center distance is minimum. These steps are repeated till data points are reassigned

4.3 To calculate the degree of membership value

For the single likelihood model,

$m^t(x_t) = l^p_j / (l^p_j + l^n_j)$ where x_t belongs to the normal class

$Mn\ (x_k) = l^n_j / (l^p_j + l^n_j)$ where x_k belongs to the negative class

For the bi-likelihood model,

$m^t(x_t) = l^p_j / (l^p_j + l^n_j)$ where x_t belongs to the normal class

$m^n(x_t) = l^n_j / (l^p_j + l^n_j)$

4.4 Likelihood value generation function

Single likelihood model:

In this model, each input is associated with a likelihood value $(x_i, m(x_i))$, which represents degree of membership of an example towards its own class label.

Bi-likelihood model:

In the model, each input is associate with bi-likelihood values, denoted as $(x_i, m^t(x_i), m^n(x_i))$, in which $m^t(x_i)$ and $m^n(x_i)$ indicate the degree of an input data xi belonging to the positive class and negative class respectively.

4.5 Pseudo training dataset generation

For the single likelihood model, the generated pseudo training data consists of two parts for the 'l' normal examples and 'n' abnormal examples as follows :

$(x_1, m_t(x_1)), \ldots, (x_l, m_t(x_l)), (x_{l+1}, m_n(x_{l+1})), \ldots, (x_{l+n}, m_n(x_{l+n}))$,

where

$m_t(x_i)$: likelihood of example x_i belonging to the normal class

$m_n(x_i)$: likelihood of example x_i belonging to the abnormal class

Similarly, the generated pseudo training data for bilikelihood model is:

$(x_1, m_t(x_1), m_n(x_1)), \ldots, (x_l, m_t(x_l), m_n(x_l)), (x_{l+1}, m_t(x_{l+1}), m_n(x_{l+1})), \ldots, (x_{l+n}, m_t(x_{l+n}), m_n(x_{l+n}))$

4.6 Autonomously tuning the SVDD parameters

Like most classifiers, SVDD has parameters that massively influence the classification accuracy. While SVDD yields good classification results for the one-class problem, manually Adjusting the parameters C and σ makes it non-applicable for real-world applications. Using grid search, parameter candidates are selected and based on an optimization criterion the best pair {Ci, σi} is determined. The task is to tune the parameters so that the accuracy on the test set and on unseen data is optimal

4.7 Data classification and outlier detection

This module presents the detected outliers by the previous module. This module gives the analysis about the outliers detected.

5. ALGORITHM USED FOR IMPLEMENTATION

5.1 Kernel k-means Algorithm

Let X = {a_1, a_2, a_3, ..., a_n} be the set of data points and *'c'* be the number of clusters.

1) Randomly initialize 'c' cluster center.

2) Compute the distance of each data point and the cluster center in the transformed space using:

$$\mathcal{D}(\{\pi_c\}_{c=1}^{k}) = \sum_{c=1}^{k} \sum_{\mathbf{a}_i \in \pi_c} \|\phi(\mathbf{a}_i) - \mathbf{m}_c\|^2, \text{ where } \mathbf{m}_c = \frac{\sum_{\mathbf{a}_i \in \pi_c} \phi(\mathbf{a}_i)}{|\pi_c|}.$$

$$\phi(\mathbf{a}_i) \cdot \phi(\mathbf{a}_i) - \frac{2\sum_{\mathbf{a}_j \in \pi_c} \phi(\mathbf{a}_i) \cdot \phi(\mathbf{a}_j)}{|\pi_c|} + \frac{\sum_{\mathbf{a}_j, \mathbf{a}_l \in \pi_c} \phi(\mathbf{a}_j) \cdot \phi(\mathbf{a}_l)}{|\pi_c|^2}.$$

where,

c^{th} cluster is denoted by π_c.

'm_c' denotes the mean of the cluster π_c.

'$\Phi(a_i)$' denotes the data point a_i in transformed space.

$\Phi(a_i).\ \Phi(aj)$ = $\exp^{-(\|ai - aj\|)*q}$ for gaussian kernel.

= $(c + a_i.a_j)$^d for polynomial kernel.

3) Assign data point to that cluster center whose distance is minimum.

4) Until data points are re-assigned repeat from step 2).

5.2 SVDD classifier

Input: The previous training set X_O ,the newly added training set X_I

Output: SVDD classifier **Ω** and the retained training set X_O.

Step 1: By training SVDD classifier **Ω** on X_O, the previous training set is partitioned into SV set SVo and non-SV set NSV_O .

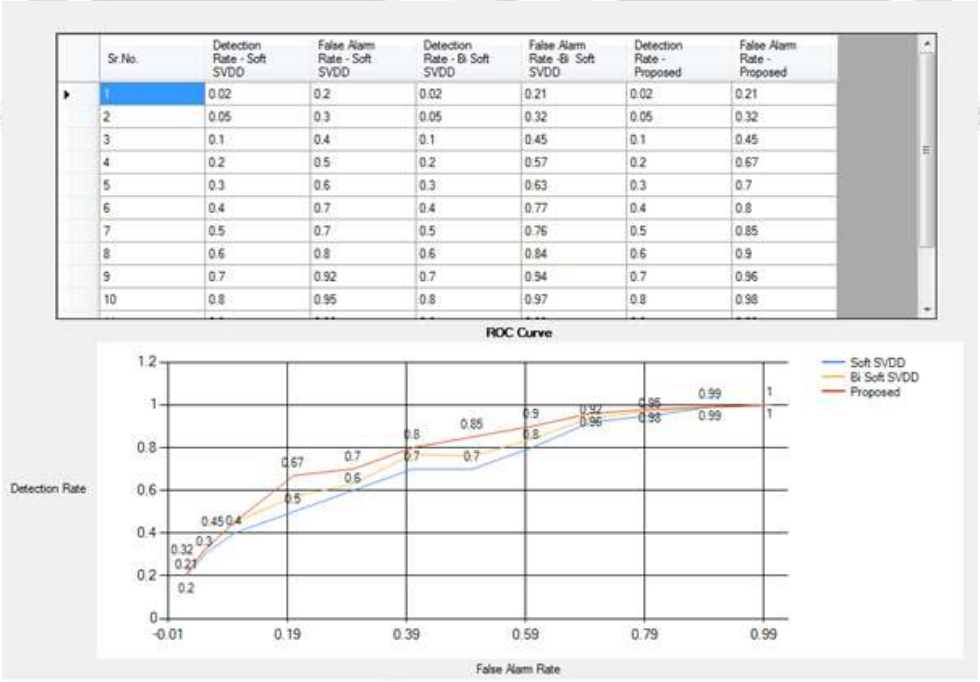
Step 2: Verify that whether there is a training sample in X_I violating KKT conditions of **Ω** .If there isn't, **Ω** and X_O will be the result of the incremental learning. Terminate. Otherwise, choose the samples violating KKT conditions of **Ω** and represent them as X^V_I .

Step 3: Choose samples satisfying from NSV_O and represent them as S NSV_O .

Step 4: Let X_O be the union of SVo , S NSV_O and V X_I and train the SVDD classifier **Ω** on X_O .

6. RESULT

The proposed method first captures the local uncertainty by computing likelihood values for each example based on its local data behavior in the feature space, and then builds global classifiers for outlier detection by incorporating the negative examples and the likelihood values in the SVDD-based learning framework**.**
Following figure shows the result of proposed system



Sr.No.	Detection Rate - Soft SVDD	False Alarm Rate - Soft SVDD	Detection Rate - Bi Soft SVDD	False Alarm Rate -Bi Soft SVDD	Detection Rate - Proposed	False Alarm Rate - Proposed
1	0.02	0.2	0.02	0.21	0.02	0.21
2	0.05	0.3	0.05	0.32	0.05	0.32
3	0.1	0.4	0.1	0.45	0.1	0.45
4	0.2	0.5	0.2	0.57	0.2	0.67
5	0.3	0.6	0.3	0.63	0.3	0.7
6	0.4	0.7	0.4	0.77	0.4	0.8
7	0.5	0.7	0.5	0.76	0.5	0.85
8	0.6	0.8	0.6	0.84	0.6	0.9
9	0.7	0.92	0.7	0.94	0.7	0.96
10	0.8	0.95	0.8	0.97	0.8	0.98

Fig: Proposed System Result

7. CONCLUSIONS

Our proposed method first captures the local uncertainty by computing likelihood values for each example based on its local data behavior in the feature space, and then builds global classifiers for outlier detection by incorporating the negative examples and the likelihood values in the SVDD-based learning framework. We have proposed four variants of approaches to address the problem of data with imperfect label in outlier detection. Extensive experiments on ten real life data sets have shown that our proposed approaches can achieve a better tradeoff between detection rate and false alarm rate for outlier detection in comparison to state-of-the-art outlier detection approaches. We plan to extend our work in several directions. First, we would like to investigate how to design better mechanisms to generate likelihood values based on the data characteristics in a given application domain

8. ACKNOWLEDGEMENT

With all respect and gratefulness, I would like to thanks all people who have helped me directly or indirectly for the paper presentation. I am grateful to my HOD , Prof. I. R. Shaikh, for his guidance and support. I wish to express my sincere thanks to the, PG Coordinator Prof. V. N. Dhakane and HOD Prof I. R. Shaikh for their support. Lastly I would like to thank staff member of Department of Computer, SND COE & RC, Yeola, Nashik, India for making all the requirements possible and simply available whenever required.

9. REFERENCES

[1] Bo Liu, Yanshan Xiao, Philip S. Yu, Zhifeng Hao, and Longbing Cao " An Efficient Approach for Outlier Detection with Imperfect Data Labels " IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 7, JULY 2014.

[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM CSUR*, vol. 41, no. 3, Article 15, 2009.

[3] V. J. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artif. Intell. Rev.*, vol. 22, no. 3, pp. 85–126, 2004.

[4] D. M. Hawkins, *Identification of Outliers.* Chapman and Hall, Springer, 1980.

[5] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection for discrete sequences: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 5, pp. 823–839, May 2012.

[6] V. Barnett and T. Lewis, *Outliers in Statistical Data.* Chichester, U.K.: Wiley, 1994.

[7] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, New York, NY, USA, 2000, pp. 93–104.

[8] S. Y. Jiang and Q. B. An, "Clustering-based outlier detection method," in *Proc. ICFSKD*, Shandong, China, 2008, pp. 429–433.

[9] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Mach. Learn.*, vol. 54, no. 1, pp. 45–66, 2004.D. M. J. Tax, A. Ypma, and R. P. W. Duin, "Support vector data description applied to machine vibration analysis," in *Proc. ASCI*, 1999, pp. 398–405