International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 14 (2018) pp. 11448-11453

Identification of Rare Diseases:An Outlier Analysis Approach

Lakshmi Sreenivasa Reddy D
Department of Information Technology
Chaitanya Bharathi Institute of Technology
Hyderabad, India.

S Chinaramu
Department of Computer Science & Engineering
Chaitanya Bharathi Institute of Technology
Hyderabad, India.

Abstract

In health industry, identification of most common diseases is very easy from health diagnosis data. Many frequency based algorithms are available to identify most frequent diseases. But there is little number of algorithms available to apply for low frequent diseases. In this paper some of the existing algorithms are applied to identify rare diseases and then optimized these algorithms to identify rare diseases automatically from huge amount of diagnosis data. Cancer data is used for experiments. Cancer data is collected from UCI ML repository [1]. The objective of the proposed work is to model a scientific method to detect rare diseases from diagnosis data. In this paper three models have been trained on the said data and detected these diseases automatically using Gaussian distribution. The main models along with Gaussian used here are Outlier factor by infrequency (OFI), Attribute value frequency (AVF) and BAD. Using these models with Gaussian achieved good accuracy.

Keywords: Cancer data, ECG data, Outlier, Classifier, Accuracy

INTRODUCTION

Outlier analysis is one of the important concepts in most applications. Most existing algorithms concentrated on numerical or ordinal attributes. Health diagnosis data includes more categorical attributes can be generalized and converted into numerical values. This procedure is not always preferable. This paper presents a simple method for categorical data. AVF method is one of the simple and efficient methods to detect outliers in categorical data. It calculates frequency of each attribute value in each attribute and finds their probabilities. This method calculates attribute value frequency score for each record by taking the average of all attribute value frequencies which are included in respected records. Outliers are records with lowest AVF scores. The only parameter used in this method is "k", the no. of outliers. FPOF is another frequent based method which depends on frequent patterns adopted from Apriori algorithm [3]. This method calculates frequent patterns of attribute values at different levels on each record. It calculates FPOF score from these patterns and finds the least FPOF scored records based on FPOF scores. These least score records are called outliers. This method requires two parameters. 'k' , the no. of required outliers and a threshold value to decide frequent patterns. Time complexity of FPOF is more in detecting outliers when compared with AVF method. The parameters used in FPOF are σ, a threshold value which is useful to decide frequent patterns in each record. There are other methods available for categorical data based on Entropy score. Greedy [4] is another method to detect outliers from categorical data. The existing approaches used to detect outliers were.

EXISTING METHODS FOR NUMERICAL DATA

A. Satistical base methods

Statistical Methods describe mostly the distribution of the data and uses univariate. These methods adopt a parametric model [3, 4]. Statistical methods have many drawbacks as the number of variables increases its efficiency decreases [4]. These draw backs can be rectified by applying principal component analysis (PCA). Attribute relevance analysis is another technique to rectify this problem. These ideas are useful for more dimensions in any Dataset.

B. Distance-based methods

Distance based methods do not take any assumptions on the distribution of the data objects because of computing the distances between all records. Complexity of these methods is very high. Large datasets with more records do not prefer these methods. Knorr's et al. [5], explained that apart of dataset records belong to each outlier must be less than some threshold value and achieved some improvements in this regard.

C. Density based methods

Density base methods find frequencies of the data records and identify anomalies as those lying in areas with low frequency. Breunig et al. described a local outlier factor (LOF) to identify local outliers based on a record contains sufficient neighbor around it or not [6]. LOF decides any record as an outlier if its LOF is less than the user defined threshold. Papadimitriou et al. defined similar methods called Local Correlation Integral (LCI). LCI selects the minimum points (min pts) in LOF through statistical methods [7]. The density based methods can detect outliers those are left by techniques with single, global criterion methods.

D. Deviation based methods

Deviation based methods find characteristics of records where as the density based methods find distances, densities and statistical parameters. If any record deviates from the given description, it is treated as outlier. Time complexity of deviation methods is linear with the dataset size. The terminology used in this paper is given below

Table I. Terminology

Term	Description
k	Target number of outliers
n	Number of objects in Dataset
m	Number of Attributes in Dataset
x_i	i^{th} object in Dataset ranging from 1 to n
A_j	j^{th} Attribute ranging from 1 to m
$D(A_j)$	Domain of distinct values of j^{th} attribute
x_{ij}	cell value in i^{th} object which takes from domain d_j of j^{th} attribute A_j
D	Dataset
V	Set of all distinct values in Dataset D
I	Item set
F	Frequent Item set
$f(x_{ij})$	Frequency of x_{ij} value
$FS(x_i)$	Set of frequent Item sets of x_i object
$IFS(x_i)$	Set of infrequent Item sets of x_i object
Minsup	Minimum support of frequent item set
Support(I)	Support of Item set I
K	Target number of outliers
AVF	Attribute Value Frequency
FPOF	Frequent Pattern Outlier Factor
FDOD	Fast Distributed Outlier Detection
NAVF	Normally Distributed Attribute Value Frequency
FAVF	Fuzzy Distributed Attribute Value Frequency
BAD	Boghapathi-Alisiri-Dirisinapu Score of a record
NBAD	Normally Distributed BAD Score
FBAD	Fuzzy Distributed BAD Score
NN	Neural Network Classifier
LR	Logistic Regression Classifier
CHAID	Chi-Squared Automatic Interaction Detection Classifier
QUEST	Quick Unbiased Efficient Statistical Tree Classifier
C5	C5.0 Classifier(Name of an Classification algorithm)
CRT	Classification and Regression Tree Classifier

EXISTING APPROACHES FOR CATEGORICAL DATASETS

A. Greedy algorithm

If any dataset contain outliers then it deviates from its original behavior and this dataset gives us wrong results in data analysis. Greedy algorithm proposed the idea of finding a small subset of records; these contribute to eliminate the uncertainty of the dataset. This disturbance is also called entropy or disturbance. We can define it formally as 'let us take a dataset D with 'm' attributes A_1, A_2----- A_m and $d(A_j)$ is the domain of distinct values in the variable A_j, then the entropy of single attribute A_j is

$$E\left(A_j\right) = -\sum_{x \in d(Aj)} p(x)\ log_2\left(p(x)\right) \qquad (1)$$

All attributes are considered as independent to each other in this approach, Entropy of the entire dataset D= { A_1, A_2-------- A_m} is defined as

$$E\left(A_1, A_2 --- A_m\right) = E\left(A_1\right) + E\left(A_2\right) + --- E\left(A_m\right) \qquad (2)$$

Greedy algorithm takes k as a parameter to find 'k' outliers as input [2]. Initially all records are treated as non- outliers. All attribute value's frequencies are computed first and using these frequencies the entropy of the dataset initially is calculated. Then, Greedy algorithm scans k times over the dataset to determine the top k outliers keeping aside non-outlier every time. While scanning each time every single non-outlier is temporarily removed from the dataset once and then total entropy is recalculated for the remaining dataset. For any normal records that results in the maximum decrease for the entropy of the remaining dataset is the outlier removed by the algorithm. Complexity of Greedy algorithm is O(k *n*m*d), where 'k' is the required number of outliers as input, 'n' is the number of records in D, 'm' is the number of attributes in D and 'd' is the number of distinct attribute values, per attribute.

B. Attribute Value Frequency (AVF) algorithm

Greedy is linear with respect to data size and it needs k-scans each time. Frequent item set mining (FIM) is another method which depends on creating a large space to store frequent item sets, and then search for these sets in each and every data point. Frequent itemset based methods become very slowly if threshold value to find frequent item sets is low. Another simpler and faster approach to detect outliers that minimizes the scans over the data and does not need to create more space and more scanning for combinations of attribute values or item sets is Attribute Value Frequency (AVF) algorithm. Outlier x_i is defined based on the AVF score as below:

$$\text{AVF Score}\ (xi) = \frac{1}{m} \sum_{j=1}^{m} f(xi\ j) \qquad (3)$$

AVF algorithm requires 'k' as input to find k-outlier based on k-least AVF scores.

The AVF algorithm time complexity is lesser when compared with Greedy algorithm. Since AVF needs only one scan to detect outliers, the time complexity is less. The complexity of

AVF is O (n * m). Frequent pattern outlier factor FPOF [8] discussed frequent pattern based outlier detection. It also requires k-value and another parameter 'σ' is required as threshold.

C. Our Approach(BAD score Algorithm)

The algorithms discussed above ,need many scans of dataset for each data object, but this algorithm needs only one scan of dataset for all records to find frequency of each value in dataset .This algorithm declares the records with any value having frequency one as outliers. This algorithm finds the disturbance of each record in data set and finds k-records as those highest BAD scores. This algorithm applied on Breast cancer data taken from UCI Machine Learning repository [10]; in this model we have defined the

1) Dataset as D= {A1, A2-------- Am},

2) D (Aj) = Domain of all distinct values in attribute 'j',

3) V=Set of all distinct values in dataset 'D'= D (A_1) $\cup$ D (A_2) $\cup$ D (A_3)….. D (A_m)={$V_{1j,}$ V_{2j} V_{3j} V_{4j} ….V_{kj}}

Where $1 \le k \le n$ and $1 \le j \le m$ for each record, then our approach to find BAD score for each record as below

$$\text{Score}_{1=} -\sum_{j=1}^{m}\left[\sum_{\forall Vkj\in D(Aj)\cap Xij=Vkj}\frac{(f(Vkj)-1)}{(n-1)}\log_{10}\left(\frac{(f(Vkj)-1)}{(n-1)}\right)\right] \quad (4)$$

$$\text{Score2}_{=} -\sum_{j=1}^{m}\left[\sum_{\forall Vkj\in D(Aj)\cap Xij\neq Vkj}\frac{(f(Vkj)}{(n-1)}\log_{10}\left(\frac{(f(Vkj)}{(n-1)}\right)\right] \quad (5)$$

$$\text{BAD Score} = \frac{1}{Score1+score2} \quad (6)$$

All the above models require k as input. It is not known that how many infrequent records need to identify. When gaussian distribution is applied to both AVF and BAD algorithms, these identified optimal number of outliers. These algorithms are called NAVF NBAD

D. Normally distributed Attribute Value Frequency (NAVF)

This model has been implemented on existing model called AVF (Attribute Value Frequency). NAVF finds the objects whose frequencies are under a threshold value "Mean – 3 * Standard Deviation".

$$\text{Mean} = \frac{1}{m*n}\sum_{i=1}^{n}\sum_{j=1}^{m}f(x_{ij}) \quad (7)$$

$$\text{S.D}=\sqrt{\sum_{i=1}^{n}\frac{1}{n}\left(\frac{1}{m}\sum_{j=1}^{m}f(x_{ij})-\frac{1}{m*n}\sum_{i=1}^{n}\sum_{j=1}^{m}f(x_{ij})\right)^2} \quad (8)$$

$$NAVFscore(D)=\left(\frac{1}{m*n}\sum_{i=1}^{n}\sum_{j=1}^{m}f(x_{ij})\right)-3*\sqrt{\sum_{i=1}^{n}\frac{1}{n}\left(\frac{1}{m}\sum_{j=1}^{m}f(x_{ij})-\frac{1}{m*n}\sum_{i=1}^{n}\sum_{j=1}^{m}f(x_{ij})\right)^2} \quad (9)$$

E. Normally distributed BAD(BAD)

The same Gaussian distribution has been applied on BAD Score algorithm. Mean, Standard Deviation.

EXPERIMENTAL RESULTS

Experiments are conducted on Breast Cancer taken from UCI Machine repository [10]. All these experiments conducted using MATLAB tool. Breast Cancer data consists nine attributes, those are "Uniformity of Cell Size", "Uniformity of Cell Shape", "Marginal Adhesion", "Single Epithelial Cell Size", "Bare Nuclei" , "Bland Chromatin", "Normal Nucleoli" , "Mitoses" , "Clump Thickness" and class attribute. This dataset contain two types of classes. One is "benign" ,other one is "malignant". Breast Cancer data is divided into two parts based on class attribute, first part contains 457 records with malignant cancer type, and second part contain 241records with benign cancer type which is used as outliers in experiments. Separation is possible by Clementine 11.1 tool. In first iteration 119 sample objects are selected randomly using Clementine tool as from each two records one is selected. The selected 119 records are mixed up with part one with 457 records and applied "BAD" "NBAD", "AVF" and "NAVF" algorithms to get outliers. Class attribute has two attribute values and all the remaining attributes contain 10 attribute values. The found outliers are given in Tables. In the next iteration 47 records are selected randomly as one record from each five records and mixed up with first part and applied the similar process .The results are given in the Tables. Another sampling is that one record is selected from each eight records and ten records and repeated the same process. Then we have applied our algorithm for 10%, 30%, 50%, 75% and 100% of outliers. Results are given in below Tables. This method has been implemented on Breast cancer which is taken from UCI Machine learning repository. This method has been compared with AVF model in each sample. Comparison graphs are given in the subsequent Figures.

A. For K=10% of outliers

If we considered 10% of outliers from the original outliers in each sample and applied "BAD" "NBAD", algorithms gave good results without any false positives and "AVF" and "NAVF" given false positives. Comparison graph of these two algorithms results has given in the Fig1.Comparision of these two algorithms results have given in Table II.

Table II. COMPARISON FOR K=10% OF ACTUAL OUTLIERS

Sample Method	*K=10%*	*AVF*		*BAD*		*NAVF*		*NBAD*	
		T P	*FP*	*TP*	*FP*	*T P*	*FP*	*TP*	*FP*
1-in-2	12	11	1	*12*	0	11	1	*12*	0
1-in-5	4	4	0	*4*	0	4	0	*4*	0
1-in-8	3	2	1	*3*	0	2	1	*3*	0
1-in-10	2	1	1	*2*	0	1	1	*2*	0

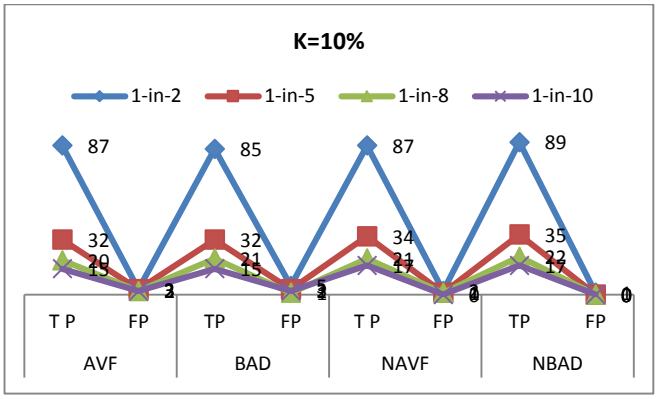

Figure 1: COMPARISON FOR K=10% OF ACTUAL OUTLIERS

B. For K=30% of outliers

Table III. COMPARISON FOR K=30% OF ACTUAL OUTLIERS

Sample Method	K=30%	AVF		BAD		NAVF		NBAD	
		T P	FP	TP	FP	T P	FP	TP	FP
1-in-2	36	35	1	35	1	35	1	35	1
1-in-5	15	14	1	14	1	15	0	15	0
1-in-8	9	8	1	9	0	8	1	9	0
1-in-10	6	5	1	6	0	5	1	6	0

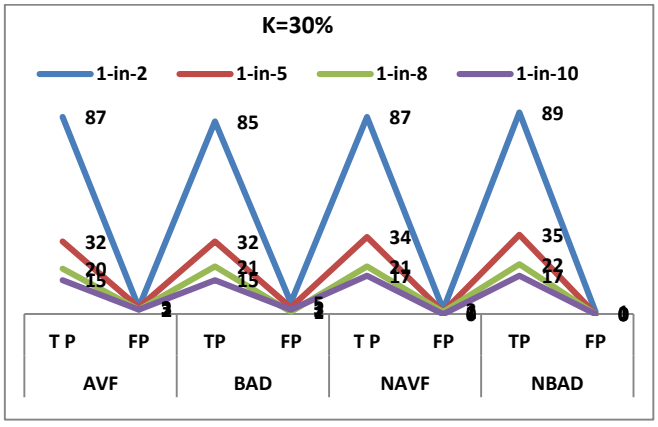

Figure 2:: COMPARISON FOR K=30% OF ACTUAL OUTLIERS

This sample considered 30% of outliers from the original outliers in each sample respectively are 36, 15, 9, and 6 and "BAD" "NBAD", "AVF" and "NAVF" algorithms revealed that 34, 14, 9, and 6 respectively from those samples and AVF and NAVF have found 35,14,8 and 5 outliers. In these AVF and NAVF have given more false positives than BAD and NBAD . The results are given in Table III and graph is given in Fig2

C. For K=50% of actual outliers

Table IV. COMPARISON FOR K=50% OF ACTUAL OUTLIERS

Sample Method	*K=50%*	*AVF*		*BAD*		*NAVF*		*NBAD*	
		T P	*FP*	*TP*	*FP*	*T P*	*FP*	*TP*	*FP*
1-in-2	60	58	2	56	4	58	2	58	2
1-in-5	25	23	2	23	2	23	2	24	1
1-in-8	15	13	2	15	0	13	2	15	0
1-in-10	10	9	1	10	0	9	1	10	0

If we considered 50% of outliers from the original outliers in each sample respectively are 60, 25, 15 and 10, similarly as the above we have applied BAD" "NBAD", "AVF" and "NAVF". BAD Algorithm found 56, 23, 15, and 10 respectively and NBAD found 58, 24, 15 true positives from those samples and AVF has found 58, 23, 13 and 9 outliers. In this too AVF has given false positives in almost all samples and BAD has given in two big samples namely 1-in-2 and 1-in 3. Among all these models NBAD has reduced the false positive rate. The results are given in Table IV and graph is given in Fig3

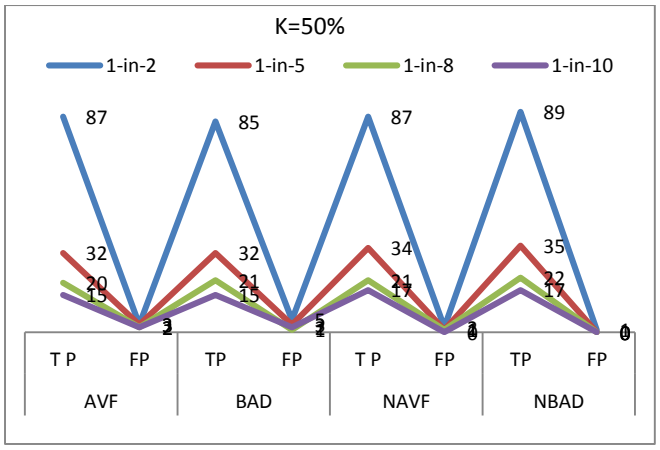

Figure 2: COMPARISON FOR K=50% OF ACTUAL OUTLIERS

D. For K=75% of outliers

Table V. COMPARISON FOR K=75% OF ACTUAL OUTLIERS

Sample Method	*K= 75%*	*AVF*		*BAD*		*NAVF*		*NBAD*	
		T P	*FP*	*TP*	*FP*	*T P*	*FP*	*TP*	*FP*
1-in-2	90	87	3	85	5	87	2	89	1
1-in-5	35	32	3	32	3	34	1	35	0
1-in-8	22	20	2	21	1	21	1	22	0
1-in-10	17	15	2	15	2	17	0	17	0

If we considered 75% of outliers from the original outliers in each sample respectively 90, 35, 22 and 17, similarly as the above we have applied both AVF and BAD score algorithms. BAD score Algorithm has found 85, 32, 21, and 15 respectively from those samples and AVF has found 87, 32, 21 and 15 outliers. In this too BAD has given false positives less than or equal to AVF except at 1-in-2 sample. Among all these models NBAD has reduced the false positive rate The results are given in Table V and graph is given in Fig4.

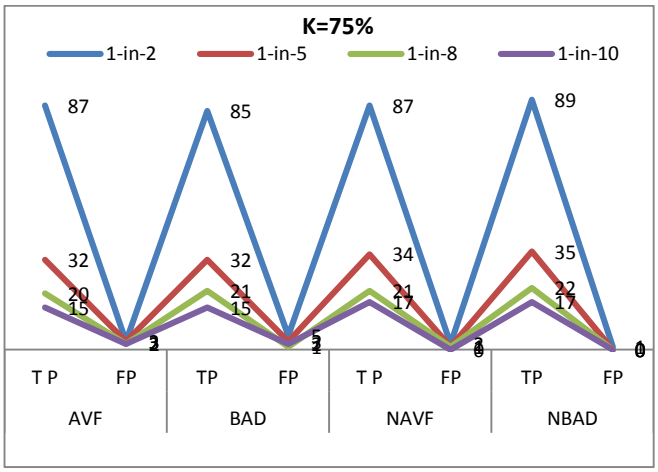

Figure 4 : COMPARISON FOR K=75% OF ACTUAL OUTLIERS

TABLE VI. comparison of Time complexity of both models

Sample		1-in-2	1-in-5	1-in-8	1-in-10
10%	AVF	0.07137	0.0713	0.07137	0.07137
	BAD	0.07957	0.0795	0.07957	0.07957
	NAVF	0.07156	0.07156	0.07156	0.07156
	NBAD	0.08012	0.08012	0.08012	0.08012
30%	AVF	0.07137	0.0713	0.07137	0.07137
	BAD	0.08075	0.0807	0.08075	0.08075
	NAVF	0.07046	0.07046	0.07046	0.07046
	NBAD	0.08116	0.08116	0.08116	0.08116
50%	AVF	0.07197	0.0719	0.07197	0.07197
	BAD	0.08081	0.0808	0.08081	0.08081
	NAVF	0.07269	0.07269	0.07269	0.07269
	NBAD	0.08121	0.08121	0.08121	0.08121
75%	AVF	0.07252	0.0725	0.07252	0.07252
	BAD	0.08226	0.0822	0.08226	0.08226
	NAVF	0.07326	0.07326	0.07326	0.07326
	NBAD	0.08423	0.08423	0.08423	0.08423
100%	AVF	0.07204	0.0720	0.07204	0.07204
	BAD	0.08106	0.0810	0.08106	0.08106
	NAVF	0.07364	0.07364	0.07364	0.07364
	NBAD	0.08576	0.08576	0.08576	0.08576

Comparing all the model in time complexity NBAD takes the more time. But the accuracy is high than any other. Fig5 shows that the rare diseases are indicated in red spots and the other diseases indicated in blue spots.

CONCLUSION AND FUTURE WORK

To sum up, the proposed work that if there are more number of rare diseases 'NBAD' is better with highest precision rate and it decides the rate of rare cases. These models are good for categorical datasets with less number of outliers. These models can be used for any type of diseases with categorical values.

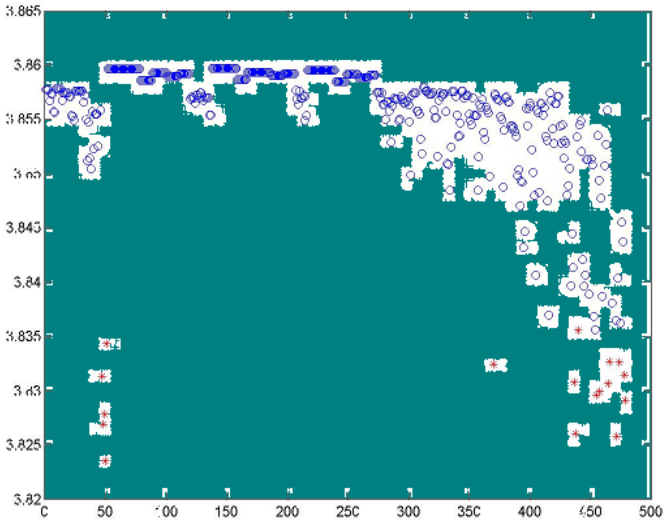

Figure 5: COMPARISON FOR K=75% OF ACTUAL OUTLIERS

REFERENCES

[1] Frank, A. & Asuncion, A. (2010). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[2] He, Z., Deng, S., Xu, X., "A Fast Greedy algorithm for outlier mining", Proc. of PAKDD, 2006.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] P. Tan, M. Steinbach, and V. Kumar, Introduction to Data Mining: Pearson Addison-Wesley, 2005

[5] E. Knorr, R. Ng, and V. Tucakov, "Distance-based outliers: Algorithms and applications," VLDB Journal, 2000.

[6] "LakshmiSreenivasaReddy.D, B.RaveendraBabu" "Outlier Analysis of Categorical Data using FuzzyAVF", presented at IEEE international conference ICCPCT-2013, pp 1259-1263.

[7] "LakshmiSreenivasaReddy.D, B.RaveendraBabu" and etc, "Learning Styles Vs Suitable Courses" IEEE international conference -MITE-2013, pp 52-57.

[8] "LakshmiSreenivasaReddy.D, B.RaveendraBabu" "Efficient Model to Find Outliers in Categorical Data Using Outlier Factor by Infrequency", presented at IEEE international conference ICCPCT-2014, pp 1324-1328.

[9] "LakshmiSreenivasaReddy.D, B.RaveendraBabu" and A.Govardhan, "A Novel Approach to Find Outliers in Categorical Dataset" presented at Elsevier - AEMDS-2013 pp 925-932.

[10] "LakshmiSreenivasaReddy.D, B.RaveendraBabu" and A.Govardhan, "A model for Improving Classifier Accuracy for Categorical data using Outlier Analysis", International Journal of Computers and Technology" vol 7, 2013. pp 500-509.

[11] "LakshmiSreenivasaReddy.D, .B.RaveendraBabu" and A.Govardhan, "Outlier Analysis of Categorical Data using NAVF"', Informatica Economica vol 17, Cloud computing issue 1, 2013

[12] M. E. Otey, A. Ghoting, and and A. Parthasarathy, "Fast Distributed Outlier Detection in Mixed-Attribute Data Sets," Data Mining and Knowledge Discovery, 2012