ADV MATH
SCI JOURNAL

Advances in Mathematics: Scientific Journal **9** (2020), no.9, 6687–6698
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.9.9.26 Spec. Issue on CAMM-2020

ENERGY DATA ANOMALY DETECTION USING UNSUPERVISED LEARNING TECHNIQUES

K. ANITHA KUMARI[1], AVINASH SHARMA[2], R. BARANI PRIYANGA[3], AND A. KEVIN PAUL[4]

ABSTRACT. Smart Grid is a rising advancement that can fulfill requests by incorporating prompted Information and Communications Technology (ICT). The specific relationship of the progressed ICT, particularly the sharp metering, creates energy data with high degree of volume, speed, and veracity. The generated big data bring huge benefits for better energy planning, efficient energy generation, and distribution. Existing techniques create a profile and identify the use cases that doesn't match against the profile as anomalies. The proposed research framed as a model-based technique that identifies and separates anomalies rather than profile matching. Isolation Forest (iForest) executes in linear time with less memory utilization considered as a wise choice for the proposed work. Our results analysis proves that iForest performs in favor to Orca, a distance based method with respect to processing in large dimension for large data set. In addition, it works fine for irrelevant attributes. One-Class Support Vector Machines (OC-SVMs) are a natural extension of SVMs. In order to identify suspicious observations, an OC-SVM estimates a distribution that encompasses most of the observations, and then labels as suspicious, those that lie far from it with respect to a suitable metric. An OC-SVM solution is built estimating a probability distribution function which makes most of the observed data more likely than the rest, and a decision rule that separates these observation by the largest possible margin. The computational intricacy of the learning stage is intensive on the grounds of training OC-SVM that involves quadratic programming problem. However, once the prediction function is ascertained, it is utilized to foresee the class name of testing data effectively.

[1] *corresponding author*

2010 *Mathematics Subject Classification.* 68W99.

Key words and phrases. Anomaly detection, Energy data, Unsupervised Learning, isolation forest, one class support vector machine.

1. INTRODUCTION

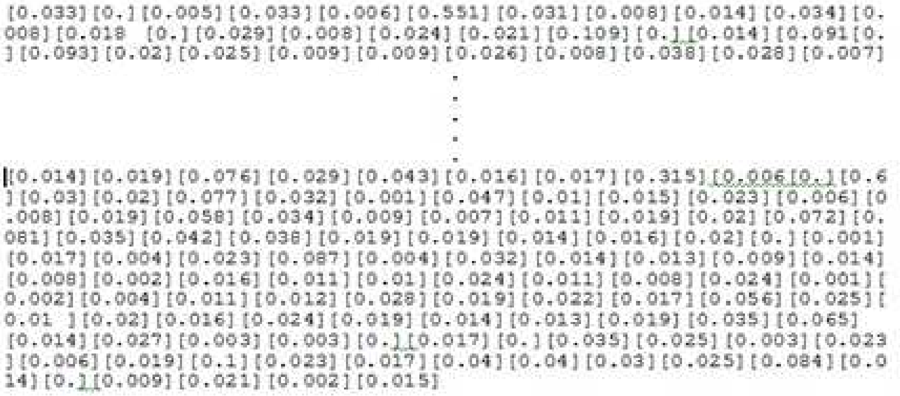
The significance of energy security is growing continuously and with these grounds, it is highly important to safeguard the energy data and its framework. Energy theft in recent years is increasing alarmingly in the industry. It can be detected by applying/using machine learning methods. As of recent times industry has turned into a profoundly directed industry, where operational security and security of the data is inevitable and its rudimentary operation. Every single action and event must be monitored continuously to avoid any undesired behavior and invasive attacks. As outside attacks/threats are expanding in a vast manner, systems are suffering with anomalies that lead to bogus data. To avoid any mishaps, continuous monitoring of data is primary mandate. Behavior of the data is monitored in a continuous manner to detect the anomalies. Anomaly detection gives better insights and enhances the accuracy in detection. It has to be performed with greater speed and scale to enhance the security of the data. Machine Learning has four normal classes of utilizations: grouping, foreseeing next esteem, peculiarity recognition, and finding structure. Among them, Anomaly identifies information that does not fit well with whatever remains of the information. It has an extensive variety of uses, for example, misrepresentation location, observation, finding, information clean-up, and prescient support. Based on the activities or actions performed by the entities, the data is produced. If the desired action is suspicious, it is termed as anomalies and it constitutes information of suspicious activities. Outlier detection is a procedure for detecting the suspicious activities in the system. In [9] introduced decorated guideline to enhance the presentation of abnormality location method. Finally given a few conversations on examination of contextual investigation after simulation and how the abnormality recognition fulfils the models. In [8] cyber-attack caused by the smart meter data while distinguishing the density of trial points is elaborated. Anomalies are detected either by using supervised or semi-supervised or unsupervised anomaly techniques. In [1], classification of unlabelled data using unsupervised learning technique either as suspicious or not is elaborated.

2. LITERATURE SURVEY

In recent times, anomalies has been subjected to many research works with advent of internet. There are many works in the literature that discuss about basic and advanced anomaly detection techniques. As all know there is no ideal algorithm, each one have their own benefits and drawback. In [3] an adaptive anomaly detection scheme for cloud computing based on LOF is introduced. In [1], unsupervised learning methods are examined and explored the results. The scheme is configured to the change while detection. In Principal Component Analysis (PCA) and Fuzzy Adaptive Resonance Theory (Fuzzy ART) are used to minimize the large dimensional data vectors and distance between a vector and its projection. In [10], a detailed survey on anomaly techniques is presented by the authors. The degree of outliers is measured using abnormal factor in local outlier factor and global outlier factor. A method for distinguishing workload patterns with an incremental clustering algorithm is present in literature. The COF algorithm [3] enhances the efficacy of local outlier factor when a pattern has alike neighborhood density as an anomaly. In [4] anomaly is detected using k-means clustering technique, and experimented with different k values. K-Means [5] clustering method is used to split instances into k clusters based on Euclidean distance. With the aid of this measure, final anomaly score is computed. In [6] detection of anomaly attack is illustrated by using Apriori algorithm. Generally, association rule mining model for intrusion detection improves the effectiveness. FaridFathnia et al. [8] compares the normal user and abnormal user data using OPTICS density-based at various circumstances. The data disturbance is measured effectively using Local Outlier Factor (LOF) index. In [9], a new RBDT (Rule Based Decision Tree) based machine learning approach is discussed for banking sector. Anomaly detection for household energy consumption data using various data mining techniques like regression-based, entropy-based and clustering-based methods are used. In [7] multi-agent-based unsupervised method is presented. The data is labelled using ensemble models and deep learning techniques are used to detect anomalies. The proposed method is demonstrated with several dataset collected from Tamil Nadu Generation and Distribution Corporation (TANGEDCO) and proves its effectiveness.

3. Proposed Unsupervised Learning Based Models

Unsupervised learning is a kind of machine learning calculation used to draw inferences from datasets comprising of information without marked reactions. The most widely recognized unsupervised learning strategy is cluster learning, which is utilized for exploratory data to discover concealed examples or grouping information. The clusters are displayed utilizing a measure of similitude which is characterized upon measurements, for example, Euclidean or probabilistic separation. Exactly when a dataset is pre-processed with the ultimate objective that it addresses a point irregularity revelation issue, the last progress before the unsupervised abnormality discovery calculation is standardization. In this work min-max normalization method is used for data pre-processing as shown in Fig 1. With the distinctive kinds of information, standardization should be performed by considering foundation learning into account. Min-max normalization is a typical normalization method, where every feature is normalized into a common interval [0, 1]. In common applications, mean value is assumed to be 0 and standard deviation to be 1. Min-max normalization is a normalization strategy which linearly transforms x to y



```
[0.033][0.][0.005][0.033][0.006][0.551][0.031][0.008][0.014][0.034][0.
008][0.018  [0.][0.029][0.008][0.024][0.021][0.109][0.][0.014][0.091[0.
][0.093][0.02][0.025][0.009][0.009][0.026][0.008][0.038][0.028][0.007]
                                  .
                                  .
                                  .
                                  .
                                  .
[0.014][0.019][0.076][0.029][0.043][0.016][0.017][0.315][0.006[0.][0.6
][0.03][0.02][0.077][0.032][0.001][0.047][0.01][0.015][0.023][0.006][0
.008][0.019][0.058][0.034][0.009][0.007][0.011][0.019][0.02][0.072][0.
081][0.035][0.042][0.038][0.019][0.019][0.014][0.016][0.02][0.][0.001]
[0.017][0.004][0.023][0.087][0.004][0.032][0.014][0.013][0.009][0.014]
[0.008][0.002][0.016][0.011][0.01][0.024][0.011][0.008][0.024][0.001][
0.002][0.004][0.011][0.012][0.028][0.019][0.022][0.017][0.056][0.025][
0.01 ][0.02][0.016][0.024][0.019][0.014][0.013][0.019][0.035][0.065]
[0.014][0.027][0.003][0.003][0.][0.017][0.][0.035][0.025][0.003][0.023
][0.006][0.019][0.1][0.023][0.017][0.04][0.04][0.03][0.025][0.084][0.0
14][0.][0.009][0.021][0.002][0.015]
```

FIGURE 1. Pre-processed energy data

- Min-max normalization is often known as feature scaling.
- Feature of data are reduced to a scale between 0 and 1.
- Min-max algorithm transforms the data set from one range to another.
- To transform (normalize) it into a particular range the following formula is applied,

$$v' = (v - min)/(max - min) * (newmax - newmin) + newmin, \tag{3.1}$$

where,

- newmin is the minimum of the normalized dataset,

- newmax is the maximum of the normalized dataset,
- v is old variable,
- v' is transformed variable.
- v=[min, max]
- v'=[newmin, newmax]

3.1. **One Class Support Vector Machine.** One-Class Support Vector Machines (OC-SVMs) are a natural extension of SVMs. One-SVM is a kind of unsupervised anomaly detection model where it partitions a whole unlabelled data into two sets. Ordering new information in OC-SVM is completely diversified. The OC-SVM code has been modified to compute kernel entries dynamically due to memory limitations. They can be prepared with unlabelled data they are a case of unsupervised machine learning. Maps input information into a high dimensional feature space. Iteratively finds the maximal edge in the hyper plane which best isolates the preparation information from the origin. The hyper plane is represented with the equation (3.2).

$$w^T x + b = 0. \tag{3.2}$$

One-class SVM partition the data between points and the origin. The issue in the traditional SVM is made stable by integration of soft margins and kernels. $\phi(\bullet)$, an internal transformation function in one-class SVM is used to propagate the data to higher dimensional space. Decision is based on the hyper plane that segregates the data from the origin. And the point falls on the contrary part are denoted as outliers. A complication arises when replacing the dot product with the non-linear kernel functions. The outcome of the decision function always only relies on the dot product of the vectors in the feature space. A slack variable is introduced, by using this slack variable can define the classification is correct or not. If slack variable is defined small, the classification is correct but not confident. If it is large the classification is wrong. The slack variable is initialized based on the hyper parameter, if the hyper parameter is zero then there is no need of slack variable, as it increases gradually at that time slack variable is introduced. The Gaussian kernel assures the presence of such a decision boundary. If the entries are positive, all values falls in the same quadrant and is suitable for any dataset. The function is defined as

$$g(x) = wT\phi(x) - \rho, \tag{3.3}$$

where w is the vector perpendicular to the decision boundary and ρ is the bias term. Decision function used to identify normal points is shown as

$$f(x) = sgn(g(x)). \tag{3.4}$$

The point is normal or not is considered as the outcome of the algorithm. Equation 5 shows the objective of one-class SVMs:

$$wT\phi(xi) \geq \rho - \xi i, \xi i \geq 0, \tag{3.5}$$

where ξi is the slack variable for point i that presents on the decision boundary and is defined as,

$$g(x) = 0. \tag{3.6}$$

And the distance is calculated as

$$d(x) = |g(x)|kwk. \tag{3.7}$$

Single objective is turned into two with the help of this equation (3.7). The detected anomalies in the energy data using OC-SVM algorithm is given in the below Fig's. The metrics are calculated based on the each outlier fraction and their corresponding outputs. The metrics for each outlier fraction is illustrated in Table 1.

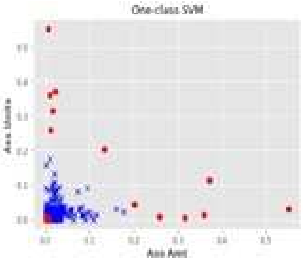


Fig 2 Outlier fraction= 0.08

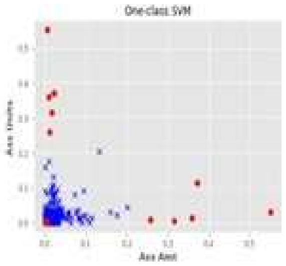


Fig 3 Outlier fraction= 0.07

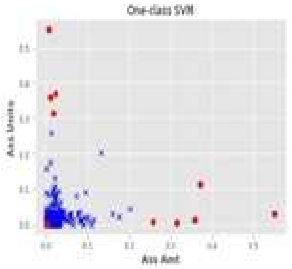


Fig 4 Outlier fraction= 0.05

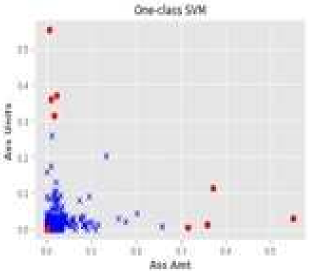


Fig 5 Outlier fraction= 0.04

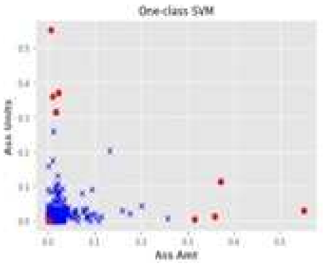


Fig 6 Outlier fraction= 0.03

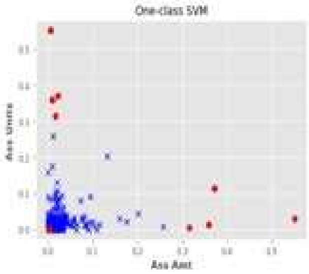


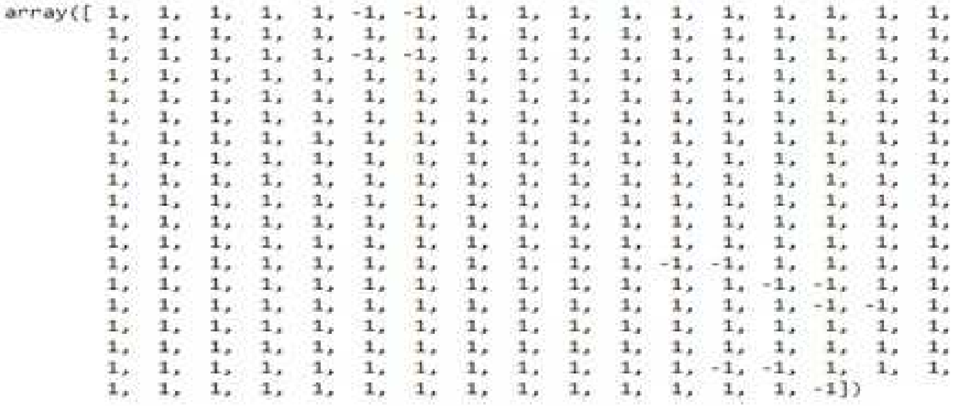
Fig 7 Outlier fraction= 0.01

TABLE 1. Performance analysis of one class SVM algorithm

Outlier fraction	Accuracy	Precision	Recall	F1 Score
0.08	0.959	0.939	0.939	0.938
0.07	0.964	0.944	0.944	0.943
0.06	0.968	0.948	0.948	0.947
0.05	0.973	0.953	0.953	0.952
0.04	0.979	0.959	0.959	0.958
0.03	0.982	0.962	0.962	0.961
0.02	0.982	0.962	0.962	0.961
0.01	0.985	0.965	0.965	0.964

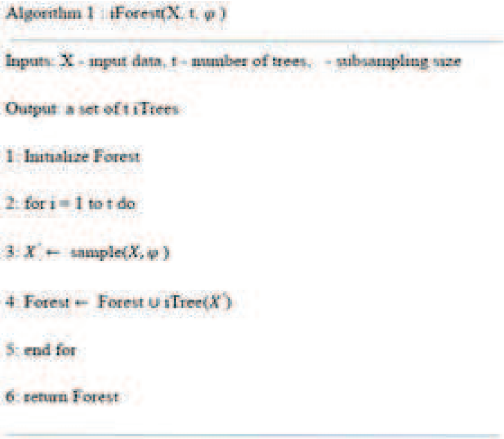
3.2. **Isolation Forest Algorithm (iForest Algorithm).** iForest algorithm and its effectiveness in formulation of anomaly score is discussed in this section. By fine-tuning the height, changes detected in behaviour is analyzed and better isolation model is illustrated by using sub-samples. Anomaly score is computed for various instances during the evaluation stage. To discriminate anomalies, random forest decision tree method is employed. The isolation forest ’isolates’ perceptions by randomly choosing an element, and splitting a value between the most extreme and the base estimations of the selected features. Random partitioning produces shorter paths for oddities. At the point when a forest of random trees aggregately create shorter path lengths for specific examples, they are very likely to be oddities. Isolation forest is one of the speediest anomaly finders and one of only a handful not many that can without much of a stretch scale up to enormous information. The anomalies are identified as -1 and normal data as 1 as shown in Fig 8. During the training stage, repeated partitioning of a subsample X' is done until all instances are well isolated. Algorithm 1 and 2 explain the process in detail. Generally speaking, using ϕ to 2^8 or 256 is sufficient for detecting anomalies over a wide extent of data by setting t = 100.

At the fruition stage, a set of trees is returned for evaluation. Expected time complexity is $O(t\phi^2)$ and the space complexity is O(t ϕ). Evaluation algorithm is describes as follows: The detected anomalies in the energy data using isolation forest algorithm is given in the below fig.

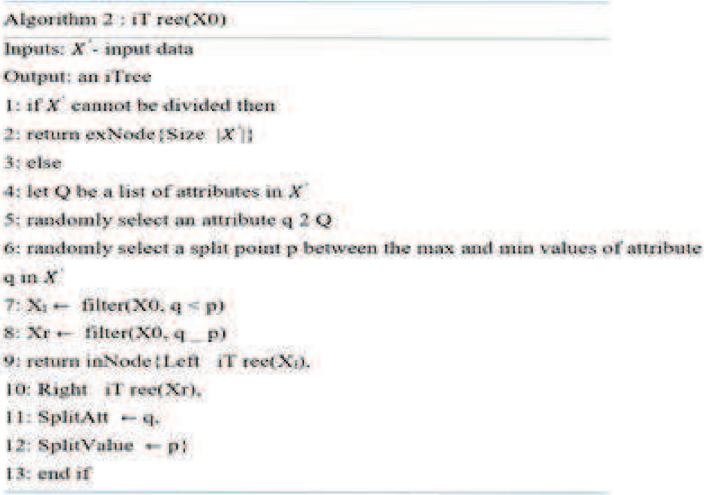


```
array([ 1,  1,  1,  1,  1, -1, -1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1, -1, -1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1, -1, -1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1, -1, -1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1, -1, -1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1, -1, -1,  1,  1,  1,
        1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1,  1, -1])
```

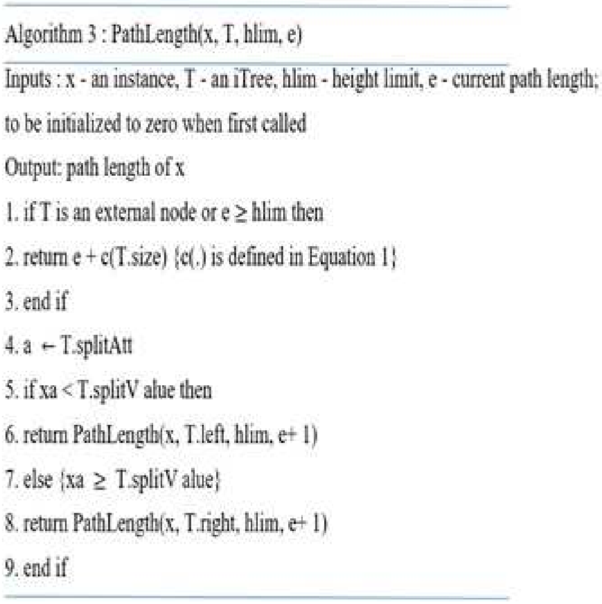
Figure 8:Data representation



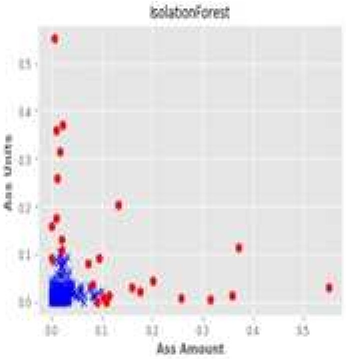
```
Algorithm 1 : iForest(X, t, φ )
Inputs: X - input data, t - number of trees,   - subsampling size
Output: a set of t iTrees
Initialize Forest
for i = 1 to t do
    X' ← sample(X, φ )
    Forest ← Forest ∪ iTree(X')
end for
return Forest
```



```
Algorithm 2 : iT ree(X0)
Inputs: X' - input data
Output: an iTree
if X' cannot be divided then
    return exNode{Size  |X'|}
else
    let Q be a list of attributes in X'
    randomly select an attribute q 2 Q
    randomly select a split point p between the max and min values of attribute
    q in X'
    Xl ← filter(X0, q < p)
    Xr ← filter(X0, q _ p)
    return inNode{Left   iT ree(Xl),
    Right   iT ree(Xr),
    SplitAtt  ← q,
    SplitValue  ← p}
end if
```



```
Algorithm 3 : PathLength(x, T, hlim, e)
Inputs : x - an instance, T - an iTree, hlim - height limit, e - current path length;
to be initialized to zero when first called
Output: path length of x
if T is an external node or e ≥ hlim then
    return e + c(T.size) {c(.) is defined in Equation 1}
end if
a ← T.splitAtt
if xa < T.splitV alue then
    return PathLength(x, T.left, hlim, e+ 1)
else {xa ≥ T.splitV alue}
    return PathLength(x, T.right, hlim, e+ 1)
end if
```



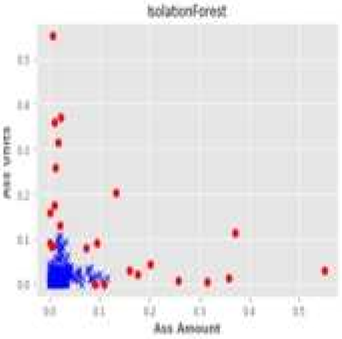
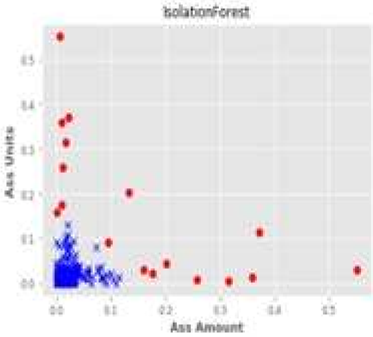


Fig 9 Outlier fraction= 0.08 **Fig 10 Outlier fraction= 0.07**



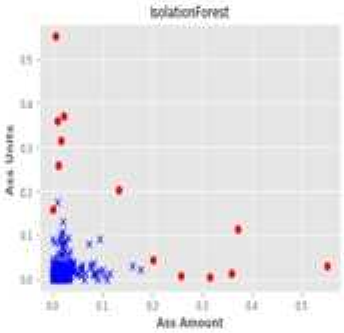


Fig 11 Outlier fraction= 0.05 **Fig 12 Outlier fraction= 0.04**

4. Result Analysis

To prove the efficiency of the proposed algorithms, accuracy, precision, recall and F1 score are considered as the evaluation metrics. In this proposed

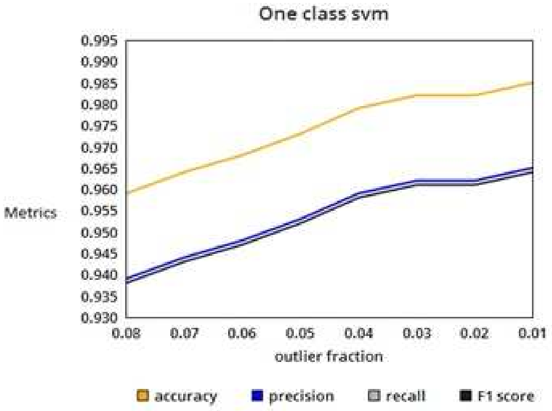
TABLE 2. Performance metrics of isolation forest

Outlier fraction	Accuracy	Precision	Recall	F1 Score
0.08	0.959	0.939	0.939	0.938
0.07	0.965	0.945	0.945	0.944
0.06	0.970	0.950	0.950	0.949
0.05	0.979	0.959	0.959	0.958
0.04	0.981	0.961	0.961	0.960
0.03	0.984	0.964	0.964	0.963
0.02	0.984	0.964	0.964	0.963
0.01	0.989	0.969	0.969	0.968

research work, the anomaly is detected using OC-SVM and iForest algorithms. Accuracy is calculated to determine the predicted observation to the total observation whereas positive observations to the total predicted positive observations is measured by precision. Recall measures the positive observations to the all observations in actual class and F1 Score is calculated using the weighted average of Precision and Recall. Fig 13 and 14 shows the statistical analysis of OC-SVM and iForest algorithm by varying the outlier fractions. The fraction of 0.01 gives better accuracy when compared to all other outlier fractions. Fig 13 and 14 shows the statistical analysis of OC-SVM and isolation forest algorithm by varying the outlier fractions. The fraction of 0.01 gives better accuracy when compared to all other outlier fractions. It is inferred from the results, both the algorithms are performing equivalently well in detecting the anomalies when outlier fraction is 0.01.

5. CONCLUSION

In the proposed system, electricity consumption is analyzed to detect anomalies in the energy data - phase, load, and capacity. Anomalies are detected using OC-SVM and iForest algorithms where iForest randomly selects a feature and perform arbitrary split. Both the algorithms are performing equivalently well in terms of accuracy, precision, recall and F1 score for detecting the anomalies. To verify the proposed method, several experiments are conducted based



. **Fig 13 Statistical analysis of OC-SVM algorithm**

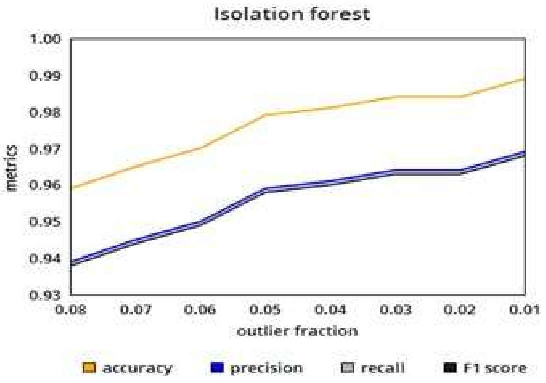


Fig 14 Statistical analysis of Isolation forest algorithm

on the outlier fraction on the real time energy dataset and the final results show the effectiveness of our method.

REFERENCES

[1] W. CHEN, F. KONG: *A Novel Unsupervised Anomaly Detection Approach for Intrusion Detection System*, IEEE 3rd international conference on big data security on cloud (bigdatasecurity), ieee international conference on high performance and smart computing (hpsc), and IEEE international conference on intelligent data and security (ids), Beijing, 2017, 69-73. doi: 10.1109/BigDataSecurity.2017.56

[2] D. WANG, L. LIU, M. QIU, Q. ZHANG, T. HUANG, Y. ZHU, Y. ZHU: *An LOF-based Adaptive Anomaly Detection Scheme for Cloud Computing*, IEEE 37th Annual Computer Software and Applications Conference Workshops, Japan, 2013, 206-211. doi: 10.1109/COMPSACW.2013.28

[3] J. TANG, Z. CHEN, A.W. FU, D.W. CHEUNG D.W.: *Enhancing Effectiveness of Outlier Detections for Low Density Patterns*, In: Chen MS., Yu P.S., Liu B. (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2002. Lecture Notes in Computer Science, **2336**. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-47887-6_53

[4] M.K. SINGH, N.K. SINGH, R. JHA, R. KUMARI, SHEETANSHU: *Anomaly Detection in Network Traffic using K-mean clustering*, 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, 2016, 387-393. doi: 10.1109/RAIT.2016.7507933

[5] Y. YASAMI, S. KHORSANDI, S.P. MOZAFFARI, A. JALALIAN: *An Unsupervised Network Anomaly Detection Approach by K-MeansClustering & ID3 Algorithms*, IEEE Symposium on Computers and Communications, Marrakech, 2008, 398-403. doi: 10.1109/ISCC.2008.4625717.

[6] E. Saboori, S. Parsazad, Y. Sanatkhani: *Automatic Firewall rules generator for Anomaly Detection Systems with Apriori Algorithm*, 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE), Chengdu, 2010, pp. V6-57-V6-60. doi: 10.1109/ICACTE.2010.5579365

[7] Y. Weng, N. Zhang, C. Xia: *Multi-Agent-Based Unsupervised Detection of Energy Consumption Anomalies on Smart Campus*, IEEE Access, **7**, 2169-2178, 2019. doi: 10.1109/ACCESS.2018.2886583

[8] F. Fathnia, F. Fathnia, D.B.M.H. Javidi: *Detection of Anomalies in Smart Meter Data: A Density-Based Approach*, Smart Grid Conference (SGC), Tehran, 2017, 1-6. doi: 10.1109/SGC.2017.8308852

[9] G. Reddy Jidiga, P. Sammulal: *Anomaly Detection using Machine Learning with a Case Study*, IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, 2014, 1060-1065 doi: 10.1109/ICACCCT.2014.7019260

[10] R. Barani Priyanga, K. Anitha Kumari, D. Dharani: *A Survey on Anomaly Detection using Unsupervised Learning Techniques*, International Journal of Creative Research Thoughts, **6**(2) (2018), 845-850.

Department of IT, PSG College of Technology, Coimbatore, India
E-mail address: anitha.psgsoft@gmail.com

Maharishi Markandeshwar Deemed University, Haryana, India
E-mail address: asharma@mmumullana.org

Tata Consultancy Services, Bangalore, India.
E-mail address: barani.rajappan@gmail.com

Department of IT, PSG College of Technology, Coimbatore, India.
E-mail address: kevinpaul1100@gmail.com