Unsupervised anomaly detection in network traffic

Amine Hrimech

Eötvös Loránd University,Faculty of informatics
3scu@inf.elte.hu
June 7, 2022

Abstract. This document was created for the Data Science Lab 1 Subject as a final - end of year - assignment. The aim of the project is to analyse the data, and select the most suitable Unsupervised Machine Learning model to detect any kind of anomalies caused by network failure or intrusions which has an important role to secure the stability of any system.

Keywords: Anomaly detecion · unsupervised learning· Kmodes · Kmeans · Isolation tree · DBSCAN · HDBSCAN · DBCV

1 Introduction

Anomaly detection is the identification of rare items, events or observations which raise suspicions by differing significantly from the majority of the data. In this project we intend to detect anomalous interactions or behavior which may be indicative of even larger problems in the network. Depending on the specific domain or context, these interactions could indicate the presence of fraudulent individuals, spammers or intrusions. Evidence of anomalous behavior with a high degree of accuracy allow us to analyze and improve our network.

2 Literature review

A new technique were proposed by Kumari, et aL. in [1] which used a combination of K-Means model with spark streaming in order to detect anomalies. Furthermore, the evaluation metric of the model performance was conducted by using the weighted average of entropy as the cluster score . The dataset used in this research was collected from the KDD cup 1999 which consist of about 4.9 connections Finally results showed that the proposed method can form the core for detecting any anomalies in a network traffic based on the features obtained from the data which also can be combined with spark streaming to detect anomalies and turn possible intrusions from data arriving in real time.

In [2] the authors tried to train a machine learning model to learn normal behavior and then look for abnormal behavior or anomalies and raise alerts accordingly. The dataset used in this research was collected from KDD 99 which consists of 41 features. Moreover, the evaluation metric of the methods performance was evaluated using the anomaly score. Finally results showed that Isolation forest is good for high dimensional features like KDD Cup and it is computationally effective.

Authors in [3] performed an investigation about unsupervised learning approach to detect cyber attacks in Cyber-Physical Systems (CPS) using time series predictor.The dataset used in this research was collected from the Secure Water Treatment Testbed (SWaT) which consists of sensors and actuators values over seven days of normal operation and four days with attack.This article proposed Long short-term memory (LSTM) method to detect the anomalies. The Training was performed on a XEON class server with 64GB of RAM using a Nvidia 2GB 750ti GPU. Finally results showed that the proposed method was successful to detect the majority of the attacks with low false positive rates.

The goal of [4] was to train an unsupervised machine learning model that can detect anomalies for a Cyber-Physical System. A dataset from the Secure Water Treatment (SWaT) testbed were used in this research which consists of all networktraffic, sensor data, and actuator data that was collected over 11 days of continuous operation (seven days of normal operation and four days of attacks).The proposed methods for anomaly detection are Deep Neural Network (DNN) which consist of a layer of Long Short-Term Memory architecture followed by feedforward layers of multiple inputs and outputs , and the second one is Support Vector Machine (SVM) which is widely used for anomaly detection.Results showed that Deep Neural Networks has a slightly better F measure than Support Vector Machines since it generates fewer false positives. In the other hand both methods have difficulties in detecting gradual changes of sensor values and anomalous actuator behavior.

In [5] the authors tried to introduce an unsupervised framework for anomaly detection based on an Attention-based Spatio-Temporal Autoencoder.The dataset used in this research was collected from the Secure Water Treatment Testbed (SWaT) which consists of 51 features generated by sensors and actuators on a per-second basis.The proposed model was based upon Bayesian Pearson correlation analysis to construct statistical correlation matrices to characterize the system status across different time steps while Attention-based Convolutional LSTM Encoder-Decoder is used to capture the most significant input features at each time step followed by using the 2D convolutional decoder to reconstruct the correlation matrices.Finally results showed that the proposed correlation matrix approach based on Bayesian Pearson Correlation Analysis is more robust to non-normality of the data.

3 Data preprocessing

The data collection process was implemented on a six-stage Secure Water Treatment (SWaT) testbed which reflects a real-world environment that helps to ensure the quality of the dataset of both normal and attack data terms. In the first seven days the system operated normally but for the rest days certain cyber and physical attacks were launched on SWaT [6].

SWaT dataset consist of network and physical properties stored in a CSV file formats which contains 52 Unlabeled features in total.

Data cleaning

The data set was completed and it has not any missing values. When we loaded the data we got 52 features and 13661 records.

However we decided to remove features with only one value since it won't be useful to build our model, Also we removed the time stamp feature so we end up by having only 39 features.

Features Selection

During this stage, we aimed to go through the data set and remove redundant and irrelevant features through different analysis techniques that analyze the variable distributions and their correlations and associations

As a first step we started creating the correlation matrix which is basically a simple table in which we can identify the correlation coefficients between features [7]. Each cell in the table shows the correlation between two features.
For example in case of two features that are having high correlation coefficient it's enough to use only one of them (see Fig. 4).

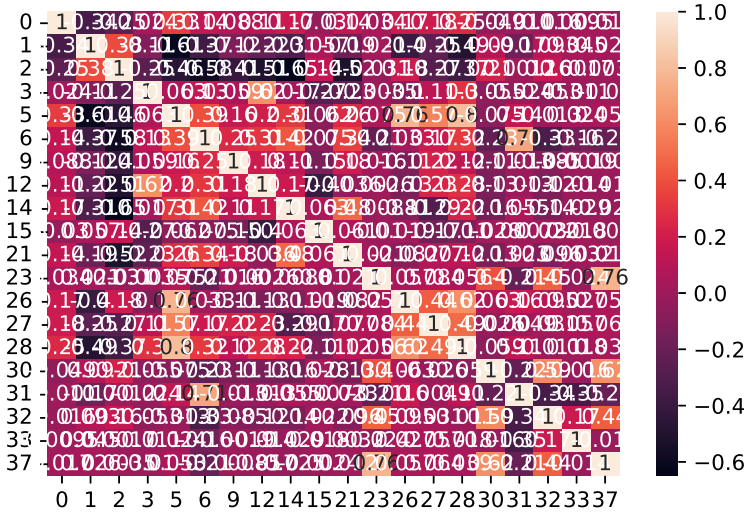

Fig. 1. Correlation matrix

As a next step we calculated the variance threshold [8] which removes all the low variance features from the data set that are of no great use in modeling.

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

Further, we decided to calculate the mean absolute difference [9] as well to double check the important of features and compare it with selected ones using the variance threshold in order to improve our selection criteria and make it more accurate.

$$MAD = \frac{1}{n}\sum_{i=1}^{n}|x_i - m|$$

The chart below presents the mean absolute difference among all our features(see Fig. 2)

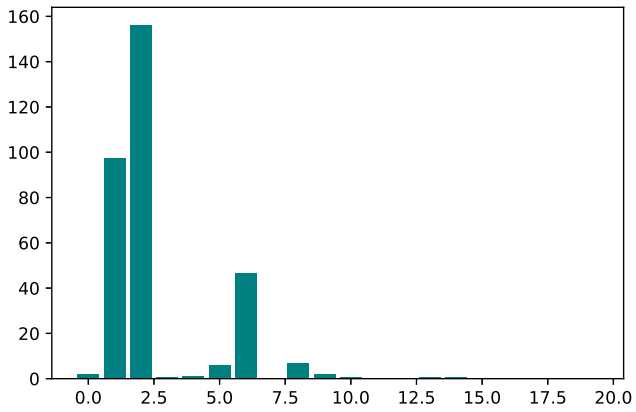

Fig. 2. Features Mean Absolute Difference

Afterwards we applied all the above techniques we managed to reduce the number of our features from 40 to only 14 which seems to be suitable for our future model.

Feature Encoding

Most of the machine learning models can only work with numerical values. For this reason, it is necessary to transform the categorical values of the relevant features into numerical representation.

Categorical variables can be addressed using multiple encoding techniques.we applied one-hot encoding technique [10]. In the label encoding, each categorical variable of features is assigned a unique integer. This can create the bias in the encoded variables and classification misinterpretation .

Feature Scaling

We normalized and scaled the continuous variables. We used the standard min-max scaling [11], this normalization method scales the data to [0,1] range as follows:
For feature X, we define:

$$X_{norm} = (X - X_{\min}) \,/\, (X_{\max} - X_{\min})$$

4 Clustring

Since the algorithm needs to be unsupervised, the options for the types of models are limited. In an unsupervised setting, anomaly detection can be done by trying to learn the general behavior of the model in the face of noisy data.

4.1 Kmeans and Kmodes

K-means

The main parameter for k-means is the K number of clusters, In order to find an optimal K number of cluster we used Elbow method in which we plot the sum of squared distance between each point and the centroid in a cluster with K value and we got an Elbow curve from which picked the elbow of the curve as the number of clusters to use and in our case it was 7.

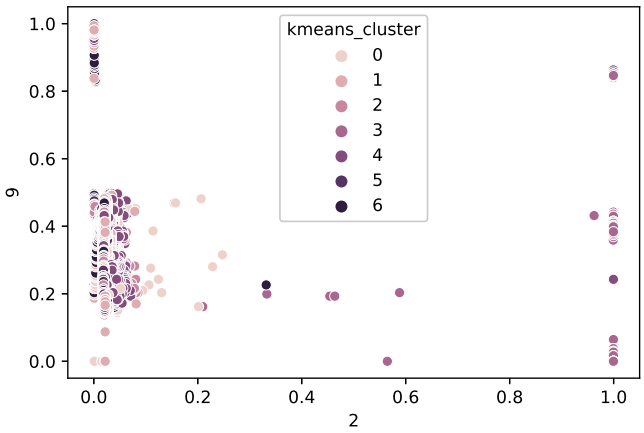

Fig. 3. K-means clusters

K-modes

K-modes is pretty much the same as K-means, But instead of distances it uses dissimilarities and it is usually applied to categorical data. We applied Elbow method again to obtain an optimal K number of clusters and got 7 cluster just as k-means.

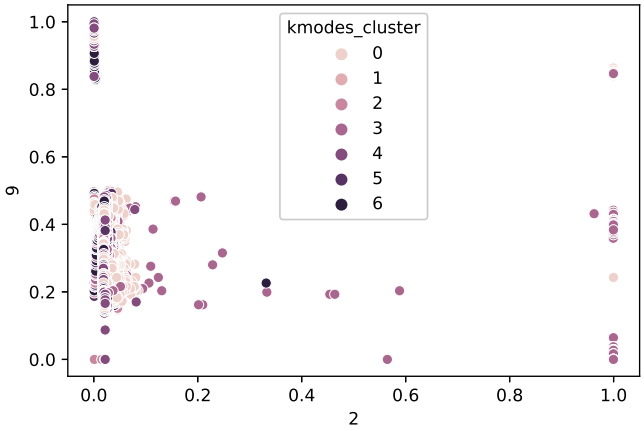

Fig. 4. K-modes clusters

Results

Despite the fact that k-means is one of the simplest and popular unsupervised machine learning algorithms, It also comes with some drawbacks such as it only handle numerical data and it assumes that all clusters are equally sized and have the same variances which is not true in most of the cases and last but not least it does not perform well with arbitrary shapes. but yet these disadvantages does not make it impossible to work with k-means, only we should take into account that results of the analysis might be affected.

Same can be said on K-modes clustering algorithm except the part of handling numerical data since k-Modes is basically an improvement form of the k-Means for categorical data type.

Table 1. Kmeans vs Kmodes results comparison

Algorithm	clusters	Silhouette Coefficient
K-means	6	0.60
K-modes	6	0.35

4.2 DBSCAN and HDBSCAN

DBSCAN

DBSCAN [12] can form clusters with arbitrary shape and size unlike k-means. Also it performs faster analysis compared to other clustering methods. However, in case of clusters with different densities are located to each other it will not be able to distinguish them.

The main hyper-parameters for DBSCAN are epsilon and minimum number of samples, In order to find optimal ones we ran our model in a pre-defined range for both eps and min samples and we chose the parameters of the one that performs better.

HDBSCAN

On the other hand, HDBSCAN [13] focuses on high density clustering, which reduces this noise clustering problem and allows a hierarchical clustering based on a decision tree approach as well as it reduce the speed of clustering in comparison with other methods.

The Hyper-parameter of HDBSCAN are the cluster selection epsilon and min samples. We defined a range for both of them so we can train our model with all the possible combinations and choose the ones fits better.

Results

Silhouette Coefficient assumes convex clusters because it consider the distance to each cluster point, but neither of DBSCAN and HDBSCAN generate convex clusters. So obviously Silhouette is not really meant to be used with noise labels as it is our case. As alternative we decided to rely more on DBCV which can validate clustering assignments on non-globular, arbitrarily shaped clusters.

Density Based Clustering Validation (DBCV) [14] considers both density and shape properties of clusters. To formulate DBCV it defines the notion of all-points-core-distance which is the inverse of the density of each object with respect to all other objects inside its cluster.
Using such Minimum SpanningTrees (one for each cluster), DBCV finds the lowest density region in each cluster and the high-est density region between pairs of clusters.

Table 2. DBSCAN vs HDBSCAN results comparison

Algorithm	clusters	anomalies	Silhouette Coefficient	DBCV score
DBSCAN	16	221	0.50	0.37
HDBSCAN	17	107	0.52	0.38

Based on the obtained results we can observe that HDBSCAN performed slightly better than DBSCAN in term of DBCV score.

4.3 Isolation Forest

Isolation Forest (IF) is an unsupervised algorithm for detecting anomalous data. Isolation Forest assigns an anomaly score based on distance from the root node. At the basis of the Isolation Forest algorithm, there is the tendency of anomalous instances in a data set to be easier to separate from the rest of the sample (isolate), compared to normal points [15].

Finding the best hyper-parameters is very computationally heavy. That is why we applied GridSearchCV [16] algorithm to test a pre-defined range of different hyper-parameters to find the optimum parameters that fits better our model. On each iteration of the Grid Search, our model will be trained with a new set of parameters, and record it's mean squared error. Once all of the permutations have been tested, the optimum set of model parameters will be returned at the end.

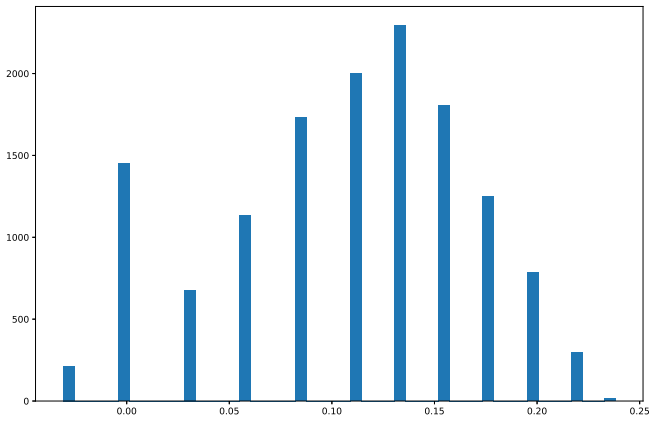

Fig. 5. Anomalies scores

We plotted the graph above (see Fig. 5) to identify the anomalies scores from which we observed that most of our data-set are normal

Results

The trained model managed to detect 214 anomalies based on the calculated score of each record.

5 Evaluation

To evaluate the effectiveness of the proposed techniques, silhouette coefficient, DBCV score were calculated depends on the used algorithm, noise points were identified as well. The clustering algorithms performance is evaluated by comparing the obtained noise points.

The detailed evaluation results are presented in Table 1 and 2. As a post analysis, the proposed unsupervised algorithm using Isolation Forest performs significantly better than the K-means and K-modes algorithms in terms of noise points detection, However DBSCAN and HDBSCAN both gives simillar results to Isolation Forest, Despite the fact that they got a low DBCV score.

Table 3. Clustering algorithms comparison

	DBSCAN	HDBSCAN	Isolation Forest
Outliers (Anomalies)	221	107	214

6 Conclusion

To conclude my research, I would like to say that the experimental results show that Isolation forest model performs favorably in terms of outliers detection. In addition, the algorithm is simple to implement, suitable for dealing with high-dimensional data and has a good parallelization potential.

As a future plan we would try to identify the number of anomalies detected in our K-means and K-modes clusters.

References

1. Kumari, Rashmi, MK Singh, R Jha, NK Singh .: *Anomaly detection in network traffic using k-mean clusteringAnomaly detection in network traffic using K-mean clustering. 2016 3rd international conference on recent advances in information technology (RAIT)*, 387–393. IEEE, 2016.

2. Vikram, Aditya .: *Anomaly detection in network traffic using unsupervised machine learning approachAnomaly detection in network traffic using unsupervised machine learning approach.* *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, 476–479. IEEE, 2020.
3. Goh, Jonathan, Sridhar Adepu, Marcus Tan ZiShan Lee: *Anomaly detection in cyber physical systems using recurrent neural networksAnomaly detection in cyber physical systems using recurrent neural networks.* *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*, 140–145. IEEE, 2017.
4. Inoue, Jun, Yoriyuki Yamagata, Yuqi Chen, ChristopherM Poskitt Jun Sun: *Anomaly detection for a water treatment system using unsupervised machine learningAnomaly detection for a water treatment system using unsupervised machine learning.* *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, 1058–1065. IEEE, 2017.
5. Macas, Mayra Chunming Wu: *An unsupervised framework for anomaly detection in a water treatment systemAn unsupervised framework for anomaly detection in a water treatment system.* *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, 1298–1305. IEEE, 2019.
6. Goh, Jonathan, Sridhar Adepu, KhurumNazir Junejo Aditya Mathur: *A dataset to support research in the design of secure water treatment systemsA dataset to support research in the design of secure water treatment systems.* *International conference on critical information infrastructures security*, 88–99. Springer, 2016.
7. Numpacharoen, Kawee Amporn Atsawarungruangkit: *Generating correlation matrices based on the boundaries of their coefficientsGenerating correlation matrices based on the boundaries of their coefficients.* PLoS One, 7(11):e48902, 2012.
8. Munson, MArthur Rich Caruana: *On feature selection, bias-variance, and baggingOn feature selection, bias-variance, and bagging.* *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 144–159. Springer, 2009.
9. Song, MH, ES Kang, CH Jeong, MY Chow B Ayhan: *Mean absolute difference approach for induction motor broken rotor bar fault detectionMean absolute difference approach for induction motor broken rotor bar fault detection.* *4th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, 2003. SDEMPED 2003.*, 115–118. IEEE, 2003.
10. Al-Shehari, Taher RakanA Alsowail: *An insider data leakage detection using one-hot encoding, synthetic minority oversampling and machine learning techniquesAn Insider Data Leakage Detection Using One-Hot Encoding, Synthetic Minority Oversampling and Machine Learning Techniques.* Entropy, 23(10):1258, 2021.
11. Patro, S KishoreKumar Sahu: *Normalization: A preprocessing stageNormalization: A preprocessing stage.* arXiv preprint arXiv:1503.06462, 2015.
12. Thang, TranManh Juntae Kim: *The anomaly detection by using dbscan clustering with multiple parametersThe anomaly detection by using dbscan clustering with multiple parameters.* *2011 International Conference on Information Science and Applications*, 1–5. IEEE, 2011.
13. McInnes, Leland, John Healy Steve Astels: *hdbscan: Hierarchical density based clustering.hdbscan: Hierarchical density based clustering.* J. Open Source Softw., 2(11):205, 2017.
14. Moulavi, Davoud, PabloA Jaskowiak, RicardoJGB Campello, Arthur Zimek Jörg Sander: *Density-based clustering validationDensity-based clustering validation.*

Proceedings of the 2014 SIAM international conference on data mining, 839–847. SIAM, 2014.

15. Ahmed, Saeed, YoungDoo Lee, SeungHo Hyun Insoo Koo: *Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forestUnsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest.* IEEE Transactions on Information Forensics and Security, 14(10):2765–2777, 2019.
16. Tai, Johnathan, Izzat Alsmadi, Yunpeng Zhang Fengxiang Qiao: *Machine learning methods for anomaly detection in industrial control systemsMachine Learning Methods for Anomaly Detection in Industrial Control Systems. 2020 IEEE International Conference on Big Data (Big Data)*, 2333–2339. IEEE, 2020.