Coefficient based Anomaly Detection

Yash Verma
ML & NLP Engineer
Ernst & Young
Gurgaon, India
yashverma@ieee.org

***Abstract*—The characteristics of various systems give rise to anomalies, which challenges high availability and high performance. Detecting anomalies in systems is a critical step towards building a secure, trustworthy system with early mitigation features, the paper addresses the issue of current Machine learning and Statistical techniques as how they fail to fit on time series with limited data and its changing data equation over a period of time. The paper presents a novel coefficient-based anomaly detection technique which uses coefficients of multinomial expression to detect anomalies and caters the issues such as drift, trend, seasonality without any external rules which current algorithms fail to achieve and then apply Minimum anomaly threshold approach on them, the algorithm also applies on multidimensional datasets.**

***Index Terms*—Anomaly detection, Multidimensional, Time Series**

I. Introduction

ANOMALY- detection is an important research area in data mining as high service availability and performance of systems are critical in any industry.

Traditionally, anomaly detection problems have been addressed using statistical and machine learning techniques [1]–[3]. The significant progress made in recent years with deep learning approaches and compute capacity have lead to performance breakthroughs in various domain specific problems like machine translation [4], [5], natural language processing [5], and speech recognition [6]. Long Short-Term Memory (LSTM) [7] networks have demonstrated the ability to automate feature extraction, handle complex non-linear temporal or sequences data and improve the capability to maintain long-range temporal dependencies [4]. These features inspired the use of LSTM networks in many recent time series anomaly detection tasks including cyber-physical systems [8]–[10], web traffic [11], and spacecraft telemetry [12].

However, they are often bounded by criteria which are difficult to be applicable on wide variety of data which requires experimentation before being deployed. For e.g. Statistical tests are often threshold based with several parametric assumptions about data which until satisfied does offer a viable solution, Machine Learning techniques require to have good amount of data to train them which in real world datasets are difficult to find.

This paper proposes loopholes in current Machine Learning Statistical time-series techniques

1) Data with changing drift, seasonality, and trend if often fitted blindly and trained, as a result the same weights, biases are used with changing characteristics of the same data with time which give wrong prediction for test points and hence wrong reconstruction errors with time.

2) Whenever, the data characteristic will change in future the model will need to be re-trained and hence the same problem as in step 1 will continue.

3) Machine Learning techniques require great amount of data to train which causes several road blocks to find the best-fit algorithm for detecting anomalies.

3) Does not take into account the inter dependency of previous data points for current data point.

This paper proposes a novel technique called Finite Multinomial Coefficients (FDM) which is based on the multinomial equation $(ax_1 + bx_2 + cx_3 \dots)^p$, where $x_1, x_2, \dots, x_n$ are variables and n and p are non-negative integers.The multinomial formula [13] describes how to expand a power p of a sum, in terms of powers of the terms in that sum.

II. System Model

A. Problem Statement

The data is a time-series vector of size T s.t vector $X = [x_1, x_2, \dots, x_n]$, the primary objective is to identify anomaly points with high Precision, Recall F-score. The working of FMC is to not based Machine learning or any other Statistical tests but the multinomial equation $(ax_1 + bx_2 + cx_3 \dots)^p$. The FMC uses p & a coefficients for every point x_t and then apply Find Critical Range procedure and Minimum Anomaly Threshold on them, then, the final output by the above procedures is compared with true anomaly labels to find Precision, Recall & F-score.

B. Finite Multinomial Coefficients (FMC)

The FMC algorithm makes assumption about data that any point x_t, can be expressed as a function of terms $x_{t-1}, x_{t-2}, x_{t-3} \dots$.Since it is a time series data and assuming that the current value x_t is a combination of previous $x_{t-1}, x_{t-2}, x_{t-3} \dots$. When fully expanded the multinomial equation takes into account every x_{t-i} term and also x_{t-i} with x_{t-j} terms with varying power. The advantages of multinomial theorem is that it gives a good approximation with all permutations and combination of $x_i's$ fits in a single equation. We just need to find the coefficient $p, a, b, c, d \dots$ For finding coefficients finite differences method can be used but the problem with that is that there are not enough possible points for fitting the equation (1) as data can change its characteristics at any point and propose a problem for accurate anomaly detection, thus derivatives offer a viable solution for finding the equation coefficients. $\frac{\partial x_t}{\partial x_{t-j}}$ can be taken for $0 <= j < i <= T$, based on these derivatives coefficient $p, a, b, c...$ can be derived. According to definition of derivatives a function of degree n when differentiated $n+1$ times yields 0, a similar approach was used in FMC , since exact 0 cannot be expected from a time series equation and thus a variable ϵ is used to approximate the derivatives to find coefficients. The paper defines any point x_t to be a multinomial expression of terms $x_{t-1}, x_{t-2}, x_{t-3} \dots$ given by (1).

Algorithm 1 calculates p for each $i <= T$ and appends to array P for each x_i in D. x_t when differentiated $p+1$ times tends to 0 which can be approximated with a limit $c \in [0.01, 0.1]$

$$x_t = (ax_{t-1} + bx_{t-2} + cx_{t-3} \dots)^p \tag{1}$$

$$d^n x_t / dx_{t-1}^n = a^n p! \tag{2}$$

$$a = (dx_t/dx_{t-1})/p{x_t}^{(p-1)/p} \tag{3}$$

Algorithm 1: FIND P

Input: c, D
Output: P

$p \leftarrow 1$, $i \leftarrow 0$, $\nabla \leftarrow d(D)/dt$
for $i = 1, \dots, len(D)$ **do**
    while *True* **do**
        $q = \nabla_i / \nabla_{i-1}$
        if $abs(q) < c$ **then**
            break ;
        $p \leftarrow p + 1$
    $P \leftarrow p$

Algorithm 2: CALCULATE DRV

Input: D
Output: DRV

$\nabla \leftarrow d(D)/dt$
for $i = 1, \dots, len(D)$ **do**
    $\frac{\partial x_t}{\partial x_{t-1}} = \nabla_i / \nabla_{i-1}$
    $DRV \leftarrow \frac{\partial x_t}{\partial x_{t-1}}$

Coefficients a & p are scaled and added after which Find Critical Range & Minimum Anomaly Threshold is applied to detect anomalies.

Algorithm 3: CALCULATE $COEFFICIENTS$

Input: DRV, P, D
Output: CF

for $i = 1, \dots, len(D)$ **do**
    $CF_i \leftarrow DRV_i / (P_i {D_i}^{(p-1)/p})$

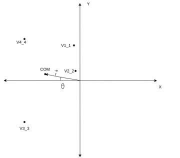

Fig. 1. COM illustration

C. Multidimensional anomaly detection

In multidimensional data of dimensions M, where assuming dependency can be present between different dimensions, centre of mass (COM) [14] for all the dimensions is taken assume that m dimension values when joined together form a figure as shown in.The COM will tell the collective view for all the dimensions.The idea behind this is that in case of normal data COM will be a regular function but in case of anomalies COM will deviate from regularity. All dimensions are scaled and given a pair (v_m, m) for corresponding value v_m of m^{th} dimension, where $m \in M$. From the origin $(0, 0)$ to COM a function of distance r and angle θ i.e $r \cos\theta$ as shown in figure() is computed, the FMC algorithm is then applied on $r \cos\theta$ to calculate anomalies.

D. Find critical range

The critical range algorithm finds the suitable region for range anomalies. It does so by extracting information from population. Coefficients values are divided into buckets of size ϵ. Thereafter a search procedure is applied to the window w to find the best-fit region through population density $pd[i]$ for specific bucket $i\epsilon$ $0 < i < 1/\epsilon$. KL-Divergence (KLD) is then applied on $pd[i]$ at local level and compared to global level, if $KLD[i] > pf(populationfactor)$ then the region is defined as anomaly.

Algorithm 4: Find Critical Range

Input: D,win,ϵ

Output: CR *[Critical range]*

1 $lt \leftarrow 0$
2 $ut \leftarrow win$
3 $CR \leftarrow [0 * len(D)]$
4 **for** $i = 1, \ldots, 1/\epsilon$ **do**
5     $count \leftarrow i * \epsilon < D < i + 1 * \epsilon$
6     $std \leftarrow std(count.index)$
7     $gd[i] \leftarrow \frac{count}{std}$
8 **for** $lt = 0, \ldots, len(D) - win$ **do**
9   *Repeat 4-7 for local density ld and D[lt:ut];* $\delta \leftarrow KLD(ld, gd)$
10   **if** $\delta > pf$ **then**
11         $CR[lt] \leftarrow 1$
12     $ut \leftarrow lt+win$
13   Return CR
14

Algorithm 5: Minimum Anomaly Threshold

Input: *Errors, InitThreshold, MaxLimit, ConvergeLimit,* ϵ

Output: *Minimum Anomaly Threshold* ($\mathcal{M}$)

$\mathcal{M} \leftarrow InitThreshold + \epsilon$, $\mathcal{P}_{max} \leftarrow 0$
$\nabla \leftarrow 0$, $\delta \leftarrow ConvergeLimit$
while $Threshold_l \leq MaxLimit$ **do**
    $Threshold_l \leftarrow InitThreshold + \epsilon$
    $Threshold_u \leftarrow Threshold_l + \epsilon/\rho$
    $N_l \leftarrow |\{Errors|Errors > Threshold_l\}|$
    for $c = 1, \ldots, \rho$ **do**
        if $Threshold_u \geq MaxLimit$ **then**
            Return $\mathcal{M}$
        $N_u \leftarrow |\{Errors|Errors > Threshold_u\}|$
        $\mathcal{P}_{cur} \leftarrow (N_l - N_u) \div N_l$
        $\nabla \leftarrow \mathcal{P}_{cur} - \mathcal{P}_{max}$
        if $\mathcal{P}_{max} < \mathcal{P}_{cur}$ **then**
            $\mathcal{M} \leftarrow Threshold_u$
            if $ConvergeLimit \geq \nabla$ **then**
                if $\delta < \nabla$ **then**
                    Return $\mathcal{M}$
                $\delta \leftarrow \nabla$
            $\mathcal{P}_{max} \leftarrow \mathcal{P}_{cur}$
        $Threshold_u \leftarrow Threshold_u + \epsilon/\rho$

E. Minimum Anomaly Threshold

We propose a non-parametric approach that does not make any assumptions about the distribution of reconstruction errors, first defines a minimum anomaly threshold ($\mathcal{M}$) (Algorithm 5). Algorithm 5 finds $\mathcal{M}$ using maximum percentage decrease ($\mathcal{P}_{max}$) in error between lower threshold ($Threshold_l$) and upper threshold ($Threshold_u$) to ensure that anomalous errors are identified correctly. The current percentage decrease ($\mathcal{P}_{cur}$) is greedily searched at each incremented $Threshold_u$ for every incremental value of $Threshold_l$ until ∇ converges to find the $\mathcal{P}_{max}$. This is done because anomalies occur in a very small number and $\mathcal{P}_{max}$ acts as an indicator that nominal values are filtered and a good estimate of anomaly values has been generated.

III. Performance Evaluation

In this section, performance evaluation of the algorithms are discussed on three different types of datasets. The datasets are publicly available time series anomaly detection datasets, Yahoo! Webscope S5 dataset (YAB) [15] [real, synthetic] which is univariate datasets and Cardiotocography dataset from UCI machine learning C. C. Aggarwal and S. Sathe, "Theoretical foundations and algorithms for outlier ensembles." ACM SIGKDD Explorations Newsletter, vol. 17, no. 1, pp. 24–47, 2015. and Kaggle TimeSeries dataset from https://www.kaggle.com/code/drscarlat/anomaly-detection-in-multivariate-time-series/data.

A. Parameter Estimation

The hyperparameters used across all the experiments as illustrated in Table I. The hyperparameters were chosen with custom variations from dataset to datset so as to get a good construction of the coefficients at all time steps and also to maximize Precision, Recall, F-score .

TABLE I
Hyperparameters of the proposed anomaly detector.

Type	Parameters
Window length	[20, 200]
InitThreshold	[75, 80]th error percentile
MaxLimit	[0.9]
ConvergeLimit	[0.001, 0.01, 0.1]
pf	[0.85, 0.9]
ϵ	[0.01, 0.1]
c	[0.001, 0.1]

B. Results and Discussions

Once the coefficients a, p are computed using Algorithm 1, 2 and 3. Find Critical Range and Minimum Anomaly Threshold is applied to find range based anomaly and point-anomalies. Discriminating coefficients were obtained for anomaly and normal points when c was in the range

TABLE II
Precesion, Recall F-score on datasets

Dataset	F-score	Precision	Recall
real28	0.979	0.979	0.979
real17	0.88	0.9	0.89
real3	0.81	0.73	0.76
real32	0.73	0.59	0.62
real55	0.88	0.8	1
synthetic41	0.85	0.75	1
Cardiotocography	0.75	0.69	0.88
TimeSeries	0.76	0.83	0.71

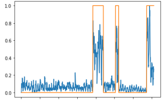

Fig. 2. real17 dataset in Blue and anomalies in Red

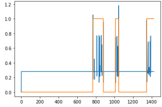

Fig. 3. real17 coefficients in Blue and anomalies in Red

[0.001,0.1]. Coefficients a and p were taken for anomaly detection because they can completely represent family of points for differentiating between normal and anomaly points.The population factor or the percentile for KLD was chosen in [0.85,0.95] because normal points constitute most of the population.

Fig.3 Coefficients clearly show clear distinction between normal and anomalous points.For normal points the coefficients were nearly constant and in case of anomalies the coefficients showed variation.

In case of multidimensional data, COM was taken and similar procedure as in case of uni-dimensional data was applied. Fig 6 clearly shows that coefficients changed their values on anomalies,hence clear showing distinction between normal and anomalous points.

IV. Related Work

Papers Chandola et al. [1] and Ibidunmoye et al. [16] gives an excellent description of approaches for the identification of anomalies in general domain and cloud system respectively.There have been a lot of development on time series anomaly of which major are in statistical analysis and machine learning.

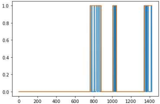

Fig. 4. real17 predicted anomalies in Blue and actual anomalies in Red

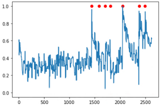

Fig. 5. COM for dataset in Blue and Anomalies in Red Dots

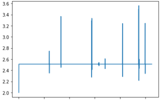

Fig. 6. Coefficients for COM for multidimensional TimeSeries dataset

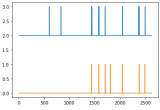

Fig. 7. Time Series datset Predicted anomalies and Actual anomalies in Red

Statistical analysis are generally parametric tests like chi squared test, generalized Student's t-test [17], and non-parametric tests like Kolmogorov-Smirnov goodness-of-fit test, or probability density [18], and relative entropy [19].

Machine learning techniques exploits supervised, unsupervised, and semi-supervised learning to find normal or anomalous data [16]. Supervised learning requires hup labelled data for normal/anamolous behavior, requiring data of faulty signatures,typically obtained by training machine learning models with seeded faults [20]. Supervised methods work on trained subsets and not useful in general.

Unsupervised learning infer data patterns and structures embedded in the unlabeled data used for training, but results in less accuracy as the actual case might differ from what is inferred Ibidunmoye [18] developed two techniques, prediction-based anomaly detection (PAD) and behaviour-based anomaly detection (BAD), which combines statistical analysis and kernel density estimation (KDE) with highly unbalanced data, dividing the cost of KDE into small sliding windows. The accuracy of these techniques are sensitive to the optimal window size.

V. Conclusion

Anomaly detection plays a vital role in major systems throughout the world, therefore accurately identifying them in cases with data in varying characteristics, size, labelling becomes immediate necessity.The paper proposed a unique way of identifying data without any model training or making any assumptions about data or its labelling.Coefficients clearly showcase family of data point which help identify anomalies and normal points based on population.

The future work can be broadened as:

a) Getting real value of power coefficient p instead of positive integer as taken in Algorithm 1.

b) Calculating accurate derivatives in Algorithm 2 for better convergence and calculation of coefficient a.

c) Increasing Precision, Recall, F-score and overall accuracy so as to directly deploy in online systems.

References

[1] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[2] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 303–336, 2014.

[3] X. Xu, H. Liu, and M. Yao, "Recent progress of anomaly detection," *Complexity*, vol. 2019, 2019.

[4] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[5] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[6] P. Liu, Z. Zeng, and J. Wang, "Multiple mittag–leffler stability of fractional-order recurrent neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 8, pp. 2279–2288, 2017.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[9] C. Yin, S. Zhang, J. Wang, and N. N. Xiong, "Anomaly detection based on convolutional recurrent autoencoder for iot time series," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–11, 2020.

[10] J. Goh, S. Adepu, M. Tan, and Z. S. Lee, "Anomaly detection in cyber physical systems using recurrent neural networks," in *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*. IEEE, 2017, pp. 140–145.

[11] T.-Y. Kim and S.-B. Cho, "Web traffic anomaly detection using c-lstm neural networks," *Expert Systems with Applications*, vol. 106, pp. 66–76, 2018.

[12] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 387–395.

[13] Wikipedia contributors, "Multinomial theorem — Wikipedia, the free encyclopedia," 2022, [Online; accessed 29-March-2022]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Multinomial_theorem&oldid=1070551425

[14] L. Bai and D. Breen, "Calculating center of mass in an unbounded 2d environment," *Journal of Graphics Tools*, vol. 13, no. 4, pp. 53–60, 2008. [Online]. Available: https://doi.org/10.1080/2151237X.2008.10129266

[15] N. Laptev, S. Amizadeh, and I. Flint. (2015) Online dataset for anomaly detection. [Online]. Available: https://webscope.sandbox.yahoo.com/catalog.php?datatype=s\&did=70.

[16] O. Ibidunmoye, F. Hernández-Rodriguez, and E. Elmroth, "Performance anomaly detection and bottleneck identification," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, p. 4, 2015.

[17] J. Hochenbaum, O. S. Vallis, and A. Kejariwal, "Automatic anomaly detection in the cloud via statistical learning," *arXiv preprint arXiv:1704.07706*, 2017.

[18] O. Ibidunmoye, A. Rezaie, and E. Elmroth, "Adaptive anomaly detection in performance metric streams," *IEEE Transactions on Network and Service Management*, vol. 15, no. 1, pp. 217–231, March 2018.

[19] C. Wang, K. Viswanathan, L. Choudur, V. Talwar, W. Satterfield, and K. Schwan, "Statistical techniques for online anomaly detection in data centers," in *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops*. IEEE, 2011, pp. 385–392.

[20] C. Sauvanaud, M. Kaâniche, K. Kanoun, K. Lazri, and G. D. S. Silvestre, "Anomaly detection and diagnosis for cloud services: Practical experiments and lessons learned," *Journal of Systems and Software*, vol. 139, pp. 84–106, 2018.