Time-series entropy data clustering for effective anomaly detection

Valentina Timčenko

Institute Mihajlo Pupin

www.pupin.rs

ICIST 2020, Kopaonik Serbia, March 2020

Introduction

Anomaly detection cutting-edge IDS research directions:

- granulation of the systematic procedures
- extensive data pre-processing
- proper feature selection
- feature engineering.

What are the possibilities of joint use of clustering machine-learning and time-series techniques for the entropy-based anomaly detection in network environment?

Problem Statement

Entropy based detection techniques: efficient but not accurate.

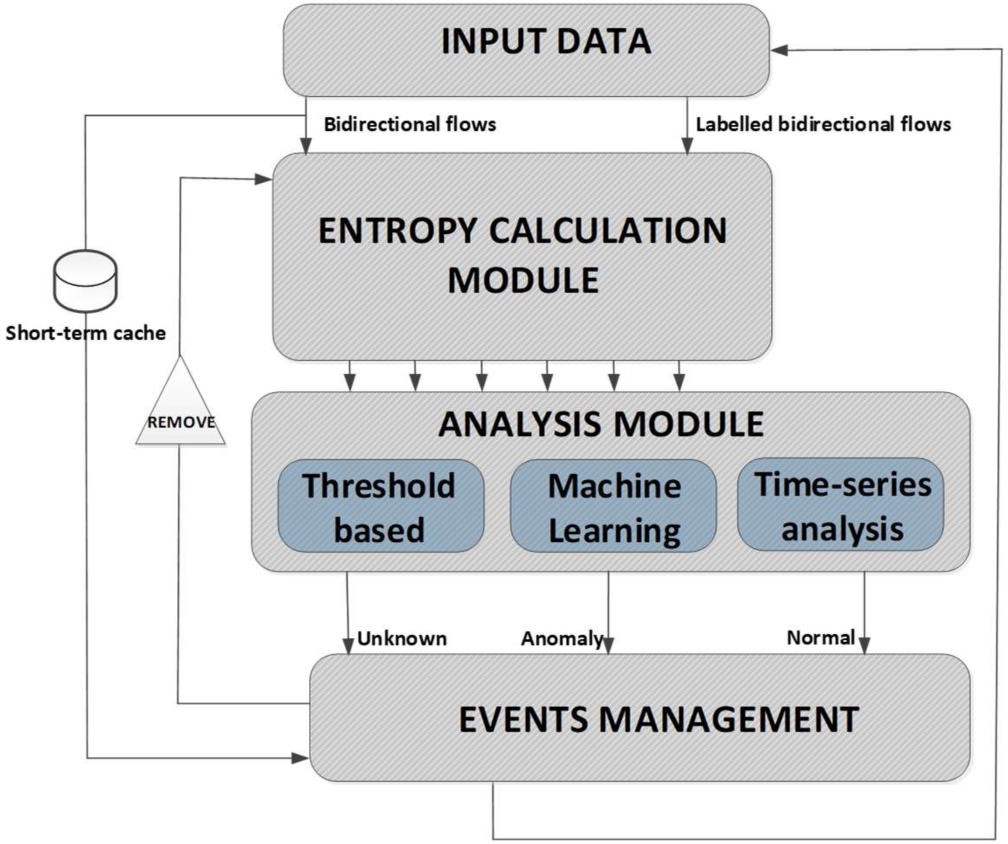
Supervised machine learning: limited implementation in the production on the unpredictable network traffic.

Unsupervised machine learning: groups data according to similarities and differences even though there are no categories provided.

Problem: some anomalious traffic is left undetected.

Challenge: adequate combination of pre-processing, entropy calculations, time-series techniques and machine learning analysis.

A comprehensive flow-based anomaly detection architecture



The idea is to estimate the possibilities to expand the proposed network traffic anomaly detection with time-series algorithms in order to provide more accurate final anomaly detection.

The earlier the detection, the most suitable the method and IDS system is for the real-time applications.

Entropy calculus for anomaly detection

Entropy: a degree of the uncertainty and randomness of a certain stochastic process. It is a measure of network traffic patterns variance: provides mechanisms for tracking the effects of traffic characteristics alteration when changing the features values.

The variations in entropy values: a reliable indication of the existence of the anomaly, attack or some malware activity.

Shannon entropy:

$$H_s(X) = \sum_{i=1}^{n} p(x_i) \log_a \frac{1}{p(x_i)}$$

Unsupervised ML in anomaly detection

- **K-means:** an iterative clustering algorithm, aims to find local maxima (cluster centroids) in each iteration. Predefined number of clusters!

- **Hierarchical clustering:** builds hierarchy of clusters. Starts with all each data point assigned to individual cluster. Two nearest clusters are merged in mutual cluster. It terminates when there is no possibility of further merging or there is a single cluster left.

- **Expectation Maximization (EM):** calculation of the probability density for accurate data allocation to a specific cluster. No strict limitations between the clusters, for each data EM calculates the probability of the membership to the generated clusters.

Time-series techniques

Sequential observations of the process taken at equidistant time points.

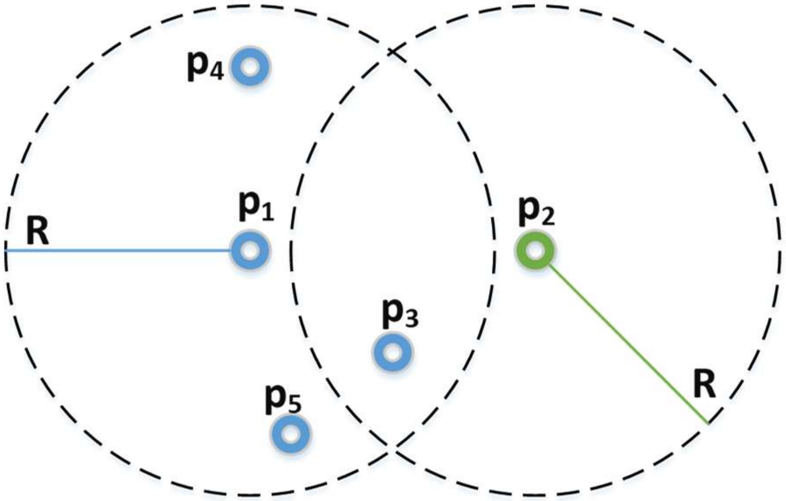
- Shapelets: detect the form of the change, which can be arbitrary. We need to detect a significant change in entropy. Shapelets depend on a large training patterns dataset.

- Statistical techniques: decomposition, Winter's exponential smoothing and autoregressive integrated moving average (ARIMA) techniques.

- Local density cluster-based outliers: distance based outlier calculations, works by clustering samples.

Distance–based outlier detection techniques

- Distance-based outliers: an object x is marked as outlier, if there are less than k objects located at a distance at most R from x.



Object p_2 is an <u>outlier for k=3</u>, since there are less than 3 objects in the R-neighborhood of p_2. The rest of the objects are marked as inliers, because there are at least 3 objects in their R neighborhood.

Sliding windows approach

Solution Improvement:

The analysis is done in real time, the newly arrived data is evaluated based to what is already calculated. The current window is analyzed by checking the data instances one by one.

In the case of the outlier detected, it is not counted as a part of the window (but raise an alarm for it), thus proceed with counting from the moment when there is another normal data instance.

We work over a variable window, all proved inliers are hold fixed in the analysis till reaching the end of defined window size.

Sliding windows approach parameters

Window (W): Contains a set of n objects from the time-series that is analyzed at each iteration. n equals the sum of inR and outR.

Slide: number of elements to move the window over the time-series.

Radius-count: number of points that define threshold for considering a newly analyzed point as outlier.

Margin (M): gives radius (R) value when multiplied with the STD.

Inliers: number of inliers for defined W, R, M.

Outliers: number of outliers for defined W, R, M.

Czech Technical University CTU-13 dataset

Labeled normal, background traffic and 13 malware scenarios captures in real-network environment, further processed to obtain NetFlows.

Our improvements:

Cleaning, labeling other anomalies, flow fragmentation, addition of new features.

Expanded dataset with model-dependent synthetic flows

Data preprocessing: Bidirectional flows (sB, sP, dB, dP), new calculated data (e.g. sPs, dPs, sB, dB).

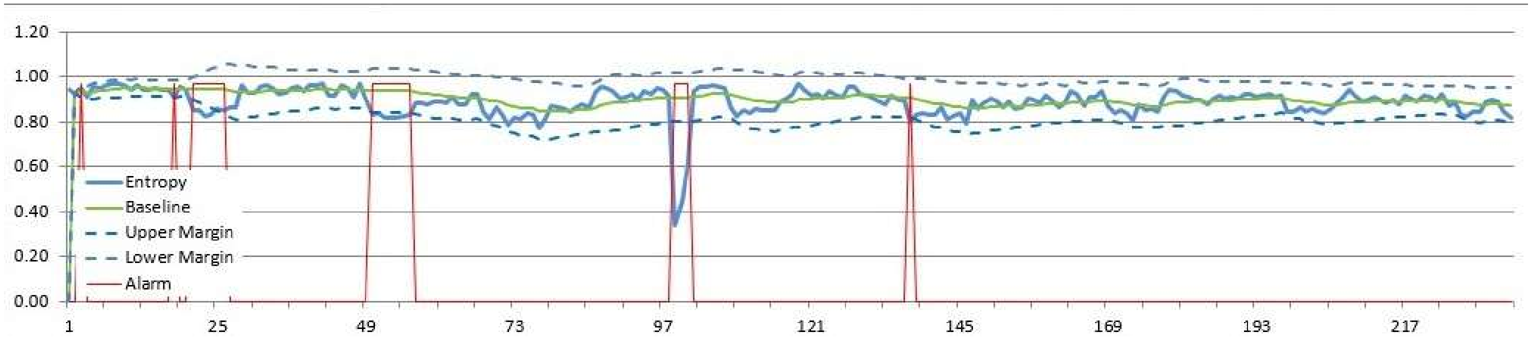
Aggregation: Top talkers, epochs, ID fields, volumetric fields (sP, dP, sB, dB).

Counting entropy for degree fields: e.g. a number of dP for a pair of S-D.

Experimental environment

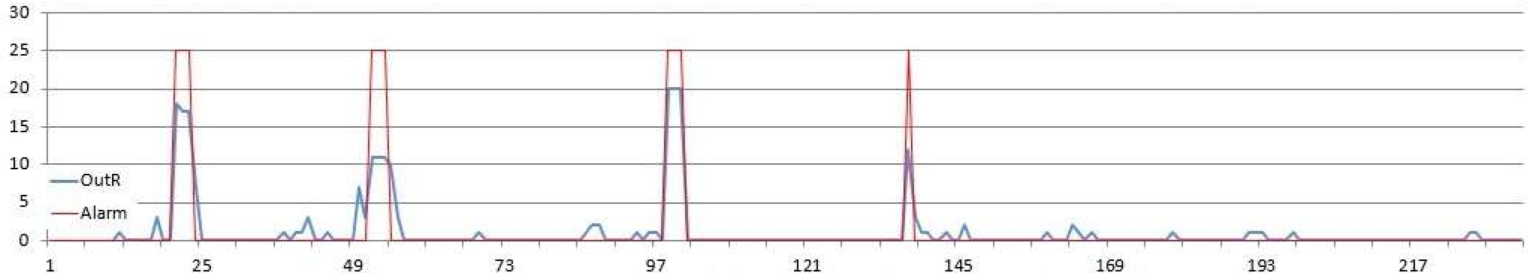
- The experimental evaluation is carried on:
  - Weka, version 3.8.3
  - Windows environment
  - 3GHz Intel(R)Core(TM) i5-2320 and 8GB of RAM
- A number of different network traffic models (real, botnet, synthetic)
- Threshold based evaluation: entropy values estimation
- Clustering: efficient granulation and grouping of instances based on the similarity (k-means, Hierarchical, EM)
- Outlier detection (OD): W = {10, 20, 30, 40}, R, M {4, 5}.

Experimental Results (1a)



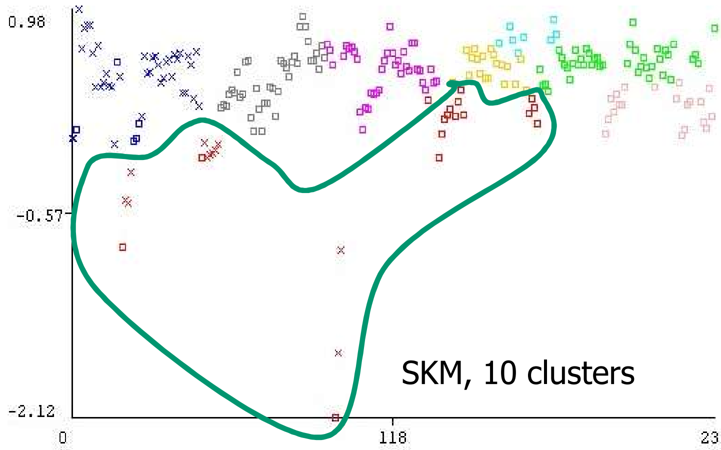
Threshold based entropy analysis: the FP at the beginning is due to the still non-stabilized STD.

TCP traffic:
d.port aggr. by s.IP
k=3 (margin of tolerance)
W,R,M = 20,10,**4**

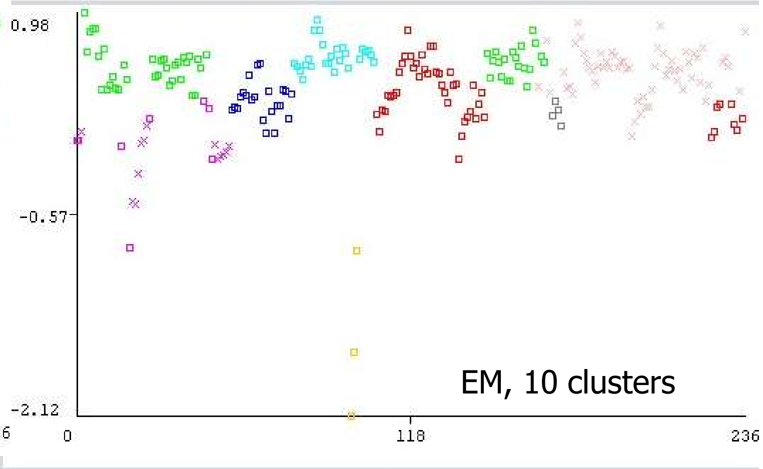


OD analysis is somewhat more efficient because it does not detect 2 FPs at the beginning of the time-series.

Experimental Results (1b)

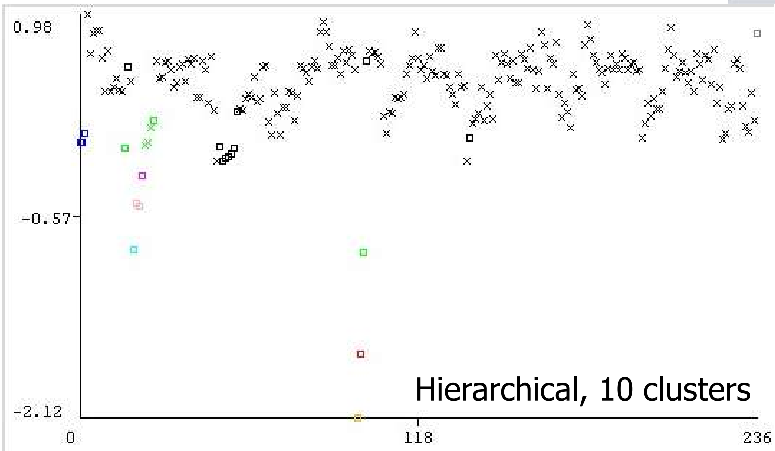


SKM, 10 clusters



EM, 10 clusters

TCP traffic:
d.port aggr.
by s.IP
k=3
W,R,M =
20,10,**4**

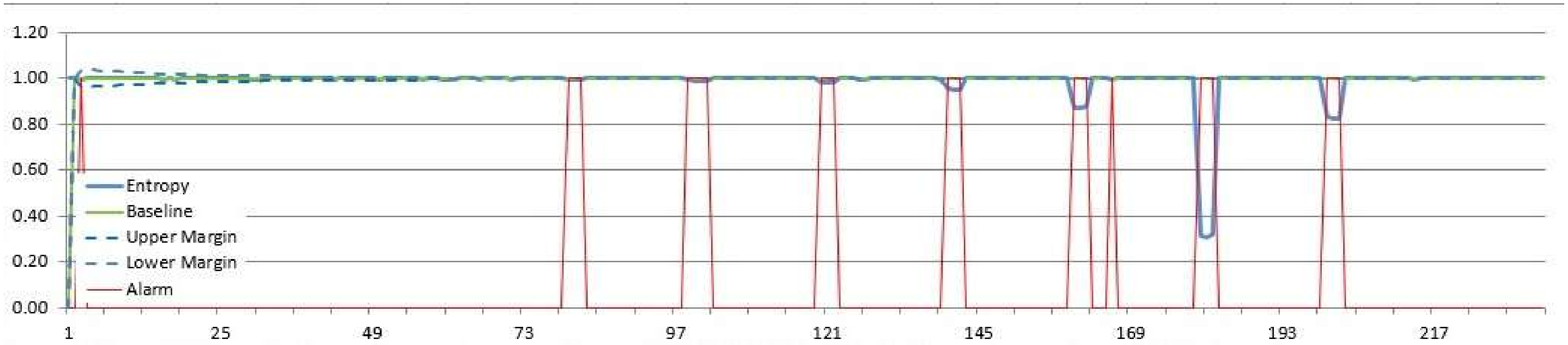


Hierarchical, 10 clusters

SKM – the most effective for this type of anomaly.
Hierarchical (H) – separates well each anomaly into an isolated cluster. The more anomalies the more clusters are needed.

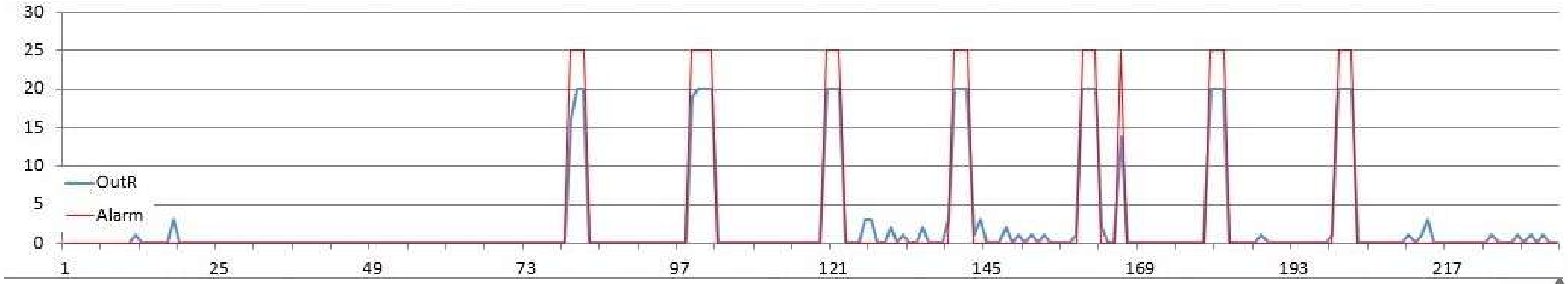
Disadvantages: SKM and H clustering require multiple repetitions for different number of clusters to find the optimal number.

Experimental Results (2a)

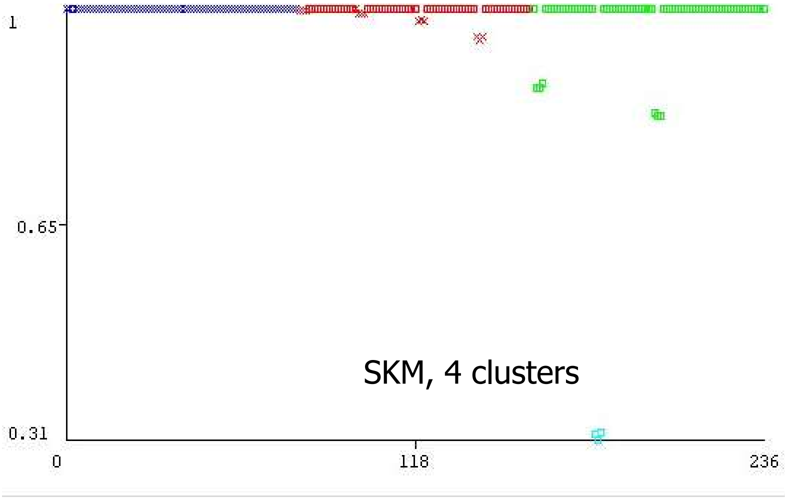


The entropy based approach and OD provide a similar result. DO is somewhat more efficient because it does not detect 1 FP at the beginning of the time-series.

CTU-1N1N, *SD:d*
k=4
W, R,M = 20,10,**4**

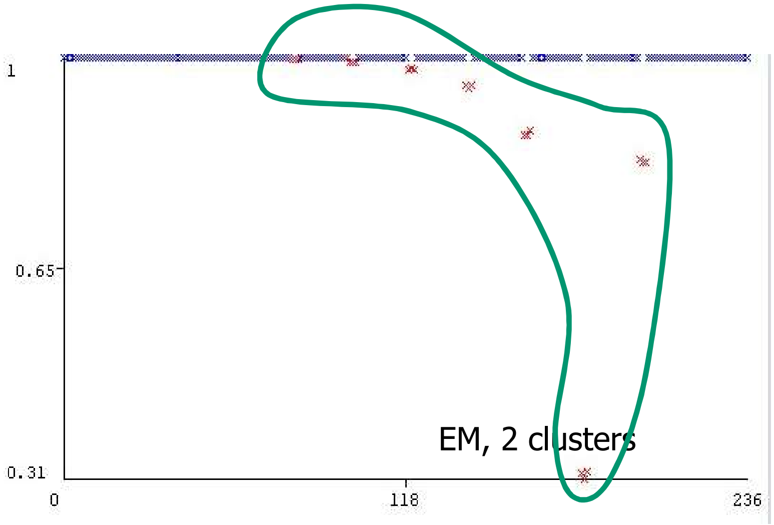


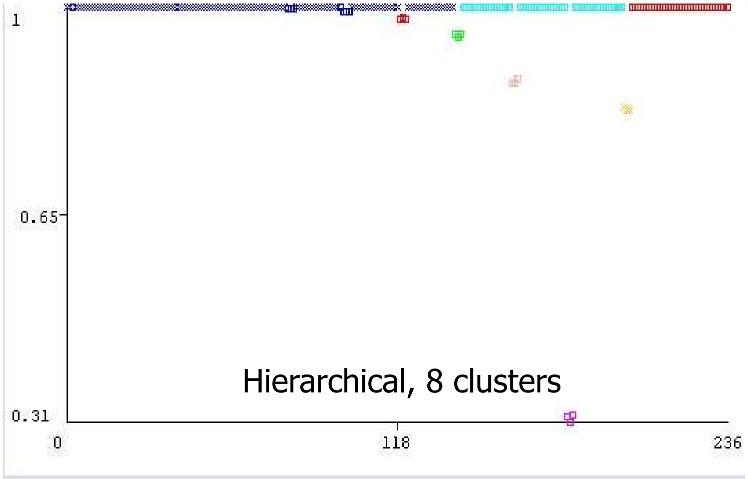
Experimental Results (2b)



CTU-1N1N, *SD:d*
k=4, Entropy
W, R, M = 20,10,**4**

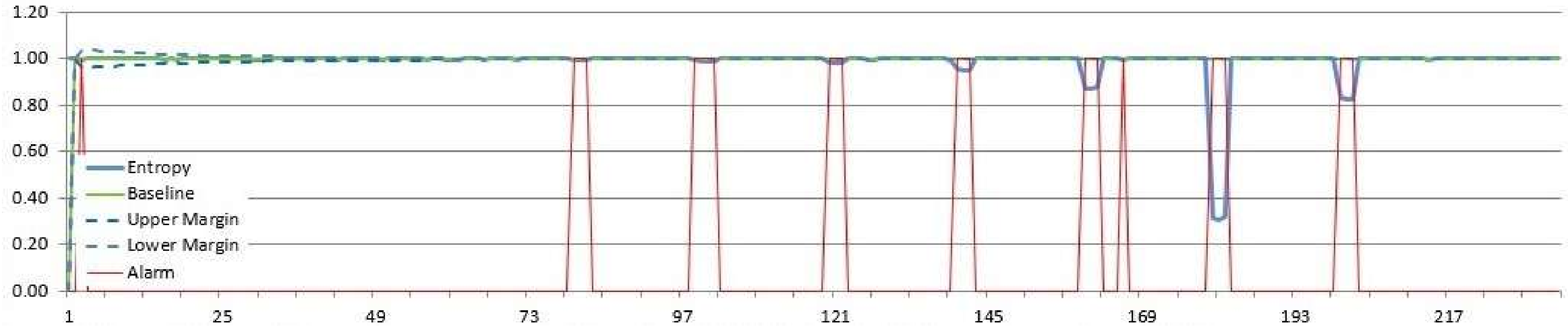
The EM clustering with 2 clusters provides the perfect separation of the anomalies and normal traffic instances. <u>Advantage</u>: The algorithm itself estimates the optimal number of clusters.



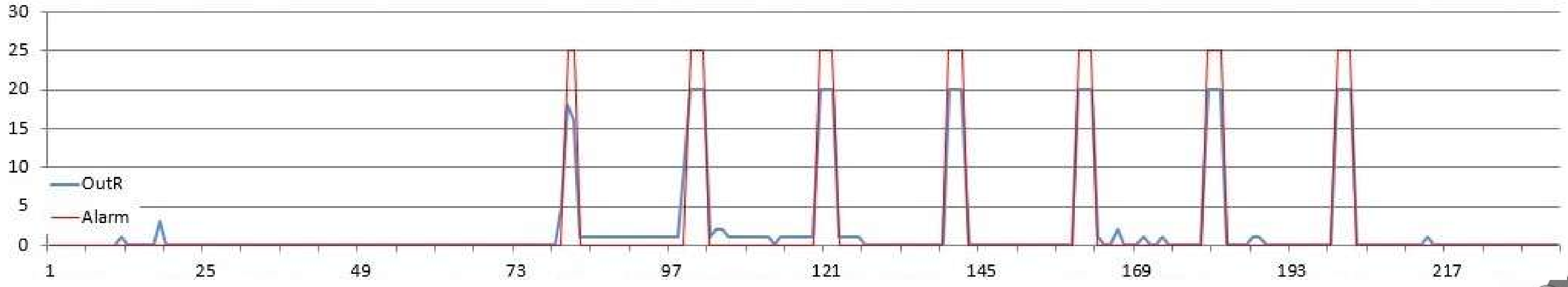


Experimental Results (2c)

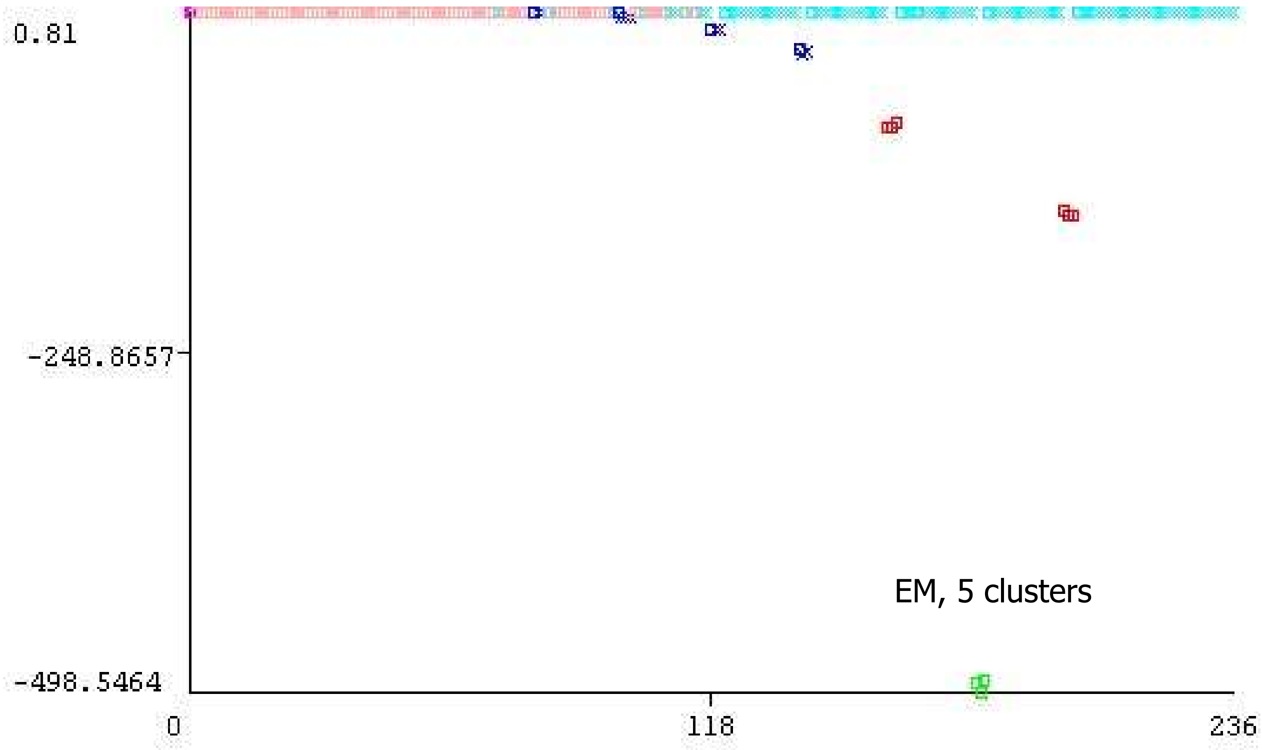
CTU-1N1N, *SD:d,* k=4
W, R,M = 20,10,**5**



Compared to the entropy based approach, DO is more efficient because it does not detect 2 FPs (one at the beginning of the time-series and another immediately after the 5th detected anomaly).



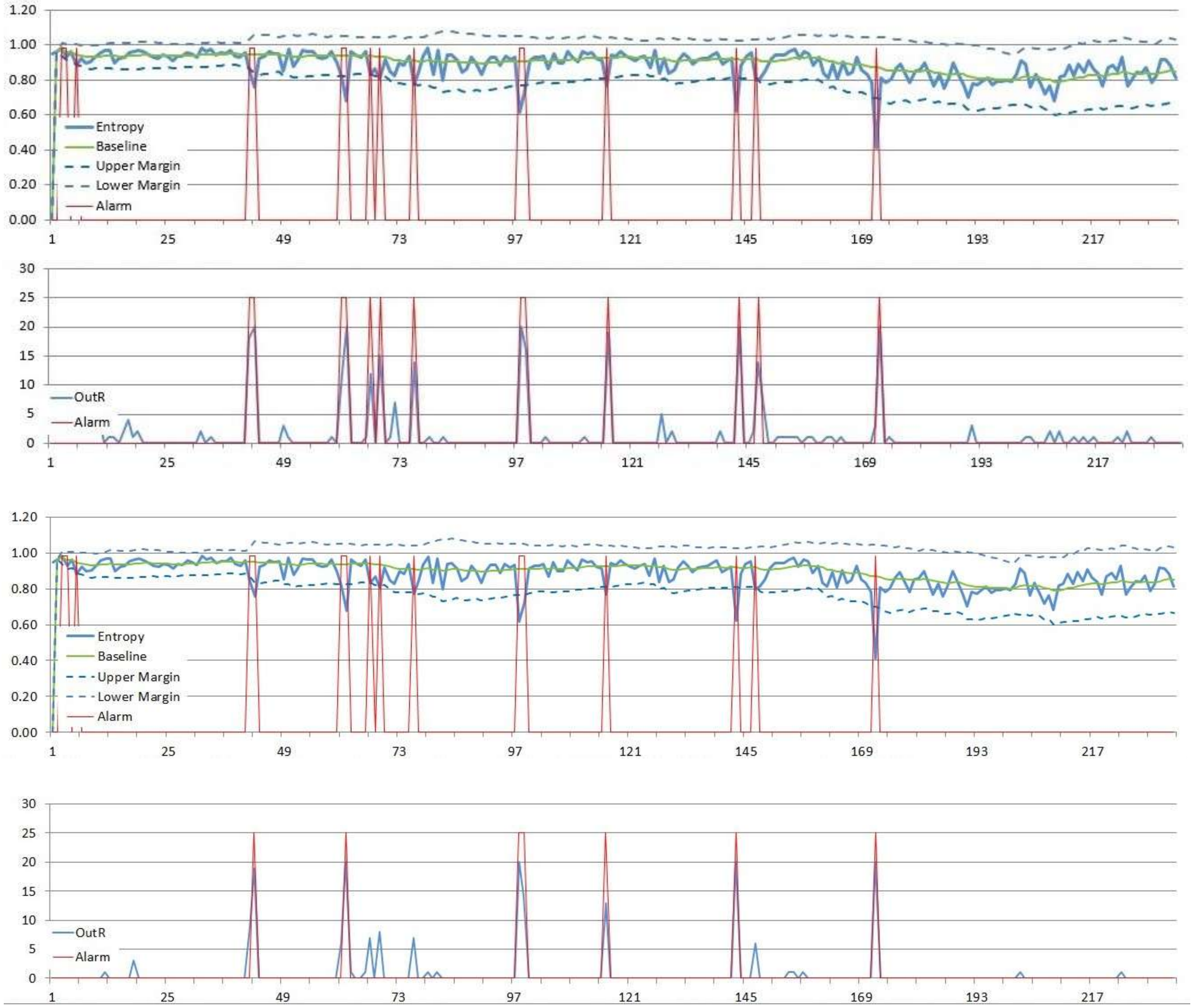
Experimental Results (2d)



CTU-1N1N, *SD:d*
k=4
W, R, M = 20,10,**5**

When compared to SKM and H, the EM clustering with 5 clusters provides successful clustering of anomalies into 3 "anomaly" clusters.

Experimental Results (3a)



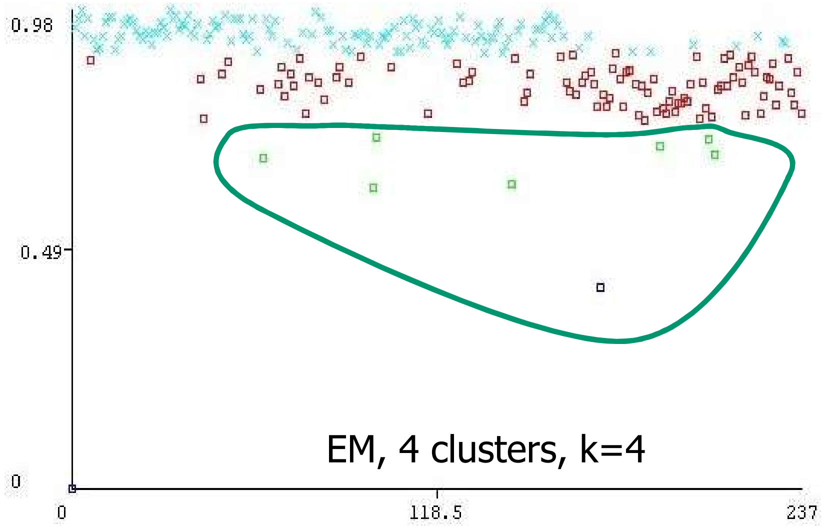
CTU-43.icmp, *S:d*, k=4
W, R = 20,10

M=4: the entropy based approach and OD yield a similar result.

M=5: the entropy based approach detects a large number of FPs while DO provides better results.

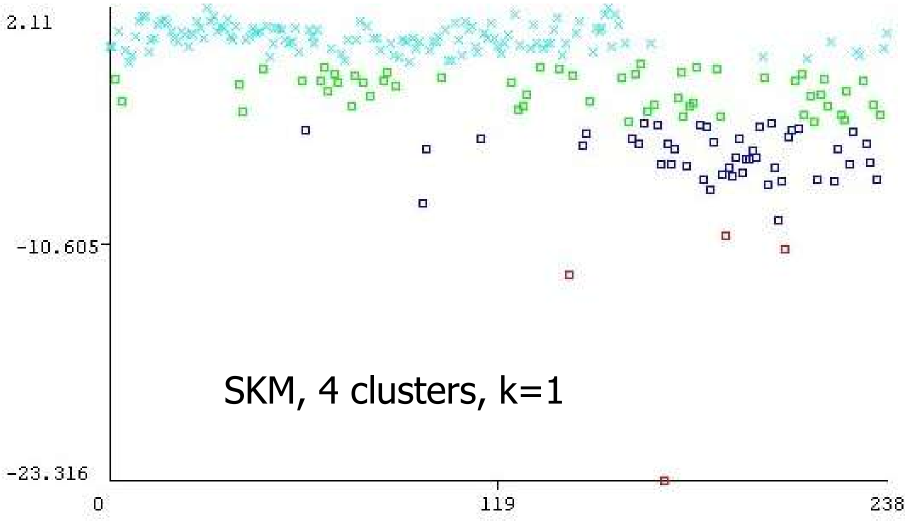
Experimental Results (3b)

CTU-43.icmp, *S:d*
W, R, M = 20,10,**4**



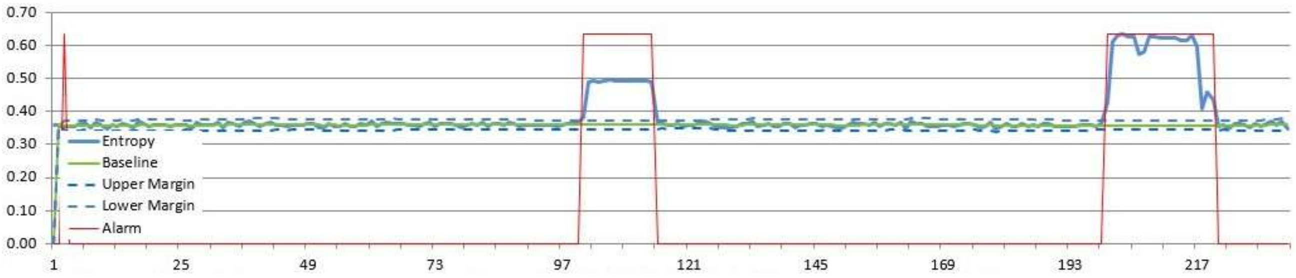
EM, 4 clusters, k=4

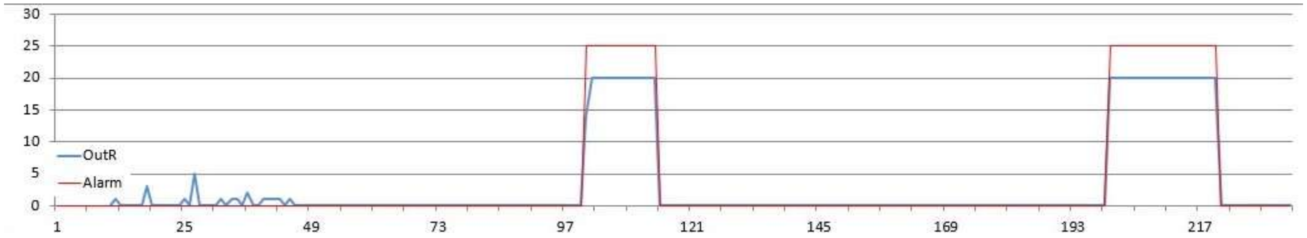
EM with 4 clusters – the most effective for this type of anomaly. It isolates anomalies into two “anomaly” clusters.



SKM, 4 clusters, k=1

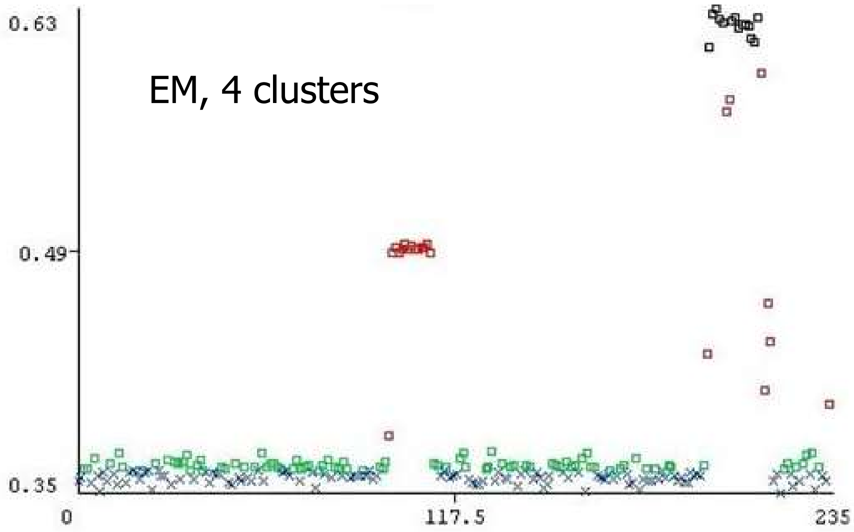
Experimental Results (4a)





Botnet, *d:f*
W, R, M = 20,10,**4**

OD is somewhat more efficient as it does not detect FP at the begining, while clearly identifying two attack blocks.



EM with 4 clusters ensures successful clustering of anomalies into 2 “anomaly” clusters. It recognized that the anomalies were of the same type (red) and that within the second block there is a superposition with another attack type, distinguished with a separate cluster (black).

Conclusion

Real-time data application. The newly arrived data is processes based on the already calculated data relations.

This approach gives a slight advantage to entropy and outlier based analysis, as these approaches deal only with the last arrived data instance, while the clustering approach takes into account all the currently windowed data.

Outliers however depend on STD, and this depends on EMA baselining and entropy.

Clustering advantage: no need to calculate and specify EMA and STD. Clustering drawback: definition of a number of clusters and identification of the normal and anomaly clusters.

ICIST2020

Thank you!

Valentina Timčenko

valentina.timcenko@pupin.rs

Slavko Gajin

slavko.gajin@rcub.bg.ac.rs

University of Belgrade, Serbia

This research is partially done by the project “**Technical solution for security threat detection in computer networks**”, supported by Innovation fund of RS, Innovation voucher number 240 (2018-2019), and supported by Ministry of Education, Science and Technological Development of RS.

www.pupin.rs/en

institut MIHAJLO PUPIN