# Anomaly Detection using Markov Chain Model

Michael Zheludev

CVC (CyberIcontrol)

e-mail: qukengue@andex.ru;

m.zheludev@cvcontrol.tech

## ABSTRACT

This paper provides a method of mathematical representation of the traffic flow of network states. The flow of states is represented as transitions to the Markov Chains. Anomalies are interpreted as graph transitions with low probabilities.

## Keywords

Kernel Methods, Data Analysis, Markow Chain.

## 1. INTRODUCTION

Network anomalies becomes a serious threat to countries, government agencies, critical infrastructures and business organizations. Some attacks are focused on exploiting software vulnerabilities to implement denial of service attacks, damage or stealing important data, other use a large number of infected machines to implement denial-of-service attacks. In this paper we focus on detecting network attacks by detecting the anomalies in network traffic flow data and anomalous behaviour of network applications. The goal is to detect the beginning of the attack in a real-time and to detect when the system is returned back to the normal state. In this paper we are not focusing on the problem of identifying the source of the attack and the attack mitigation.

The network traffic flow data can be represented by a set of network-level metrics (amount of packets for different protocols, inbound and outbound traffic, etc.) and application-level metrics (like the response duration histogram for web server). These metrics are collected by the traffic analyser at fixed rate. The goal for the state analyser is to detect anomalous network and/or application behaviour basing on these metrics.

The input data for the analyser is statistics matrix that contains a single row for every traffic time slice. Each row contains network-level and application-level features that come from different scales. This matrix is the input for intrusion detection processes (both training and detection steps).

Our method has two sequential steps. Study and analysis of the behaviour of networking datasets and projection of data onto a lower dimensional space - training step. This is done once and updated as the behaviour of the training set changes. During this step we can handle corrupted training sets.

The output from the training step enables online detection of anomalies to which we apply automatic tools that enable real-time detection of problems. Each newly arrived datapoint is classified as normal or abnormal.
Analysis of the indicators of network traffic reveals represent normal behaviour as statistically dependent set, grouped in clusters after reduced dimensionality operation, against which the representation of anomalies. Anomalies is not statistical connected with the basic set of states. They appear as distant from the main cluster points.

The traffic analyser processes the network packets and summarises the network-level statistics. These metrics include: tcp flags usage; number of control tcp packets (packets without payload); number of data tcp packets (packets with payload); number of source (client) packets; number of source control packets; number of source data packets; number of source data bytes; number of destination (server) packets; number of destination control packets; number of destination data packets; number of destination data bytes.

TCP-connections could be reassembled to estimate application-level metrics. Another sources of application-level metrics are the log files from applications (like access-logs on HTTP web server). The analyser processes the application logs to collect and summarise application level metrics (like total amount or requests, total amount of errors, histogram of the response times, histogram of error codes, etc). These metrics can be extended by adding other sources of behaviour metrics, like e-mail server logs, database server logs, cpu/memory metrics. We measure, receive and sense many parameters (features) at every pre-determined time interval – forming high dimensional data. The challenge are: How to cluster and segment high-dimensional data? How to find distances in high-dimensional data? How to find deviations from normal behaviour?

Challenge: How to process an “ocean” of data in order to find abnormal patterns in the data? How to fuse data from different sources (sensors) to find correlations and anomalies? How to find distances in high-dimensional data? They do not exist. How can we determine whether a point belongs to a cluster/segment or not? The goal is to identify points that deviate from normal behaviour which reside in the cluster/segment. How we treat huge high dimensional data that is dynamically and constantly changes? How can we model the high dimensional data to find deviations from normal behaviour?

## 2. MODEL OF STATES FLOW

The traffic state at each time point can be represented by a vector, as shown in Figure1

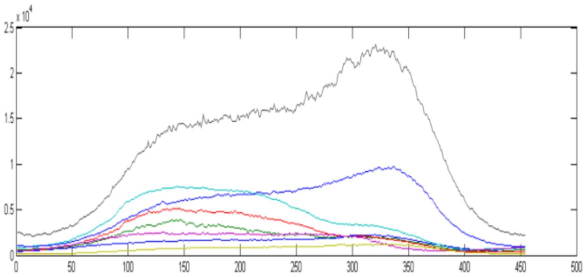

Figure1: traffic behavior in a single day, represented by several general trends.

Thus, the traffic can be modeled as a random process related to the vector X (t), where t is time. Define X={X<sub>t</sub>}<sub>t</sub> the dataset of all traffic states X (t), where for each t X (t) belongs to n-dimensional space R<sup>n</sup>.

At the training stage of the algorithm, we collect statistics on the behavior of traffic. We group this behavior into clusters [1] . Each cluster has its main trend (the center of the cluster) and a corridor that characterizes the deviation in it.
We define the vertices of the Markov chain associated with states through lace.
The standard behavior of traffic we want to represent as a set of vertices of the Markov chain.

Let $F = \{f_k\}$ be the set of traffic behavior patterns defined through the cluster centers. $W = \{w_k\}$ are corridors of variation for each pattern.
Let's define the *lace*.
Definition1. The $W$-quantization measure for $F$ is called

$$\eta_w(F) = \sum_{k \neq p} \#\{i | 0 < |f_k(i) - f_p(i)| < w_k(i)\}$$

Where $\#\{.\}$ Stands for the power (number of elements) of the set
Definition2. We define the lace as the set $H = \{h_j\}$, realizing the minimum of the following functional:

$$H = N_w(F) = \sum_k \|h_k - f_k\|^2 + \eta_w(H) \overset{H}{\rightarrow} \min \qquad (1)$$

The traffic state and " the lace" is illustrated in Figure2

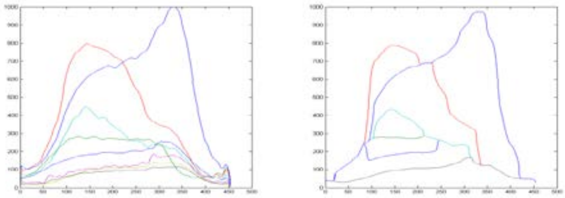

Figure2: traffic behavior in left side. The *Lace* in right side

## 2.1 Constrction of "the Lace"

We associate with the family of vectors $F = \{f_i\}$ the graphical form $G$.

We define $G$ as a matrix of size $m \times n$, where $m$ is the maximum value of $F$, n is the number of coordinates of each of $f_i$. Elements of the matrix $G$ are located in the interval [0,1]. For an element of the matrix $G$ with index $(p, k)$ we define the intensity by the following formula:

$$g(p,k) = \max_r e^{\frac{[p - f_r(k)]^2}{W_p^2(k)}}$$

We obtain a map of the smoothed graph of cluster centers, where the thickness of the line is determined by the weight (corridor) of the component.

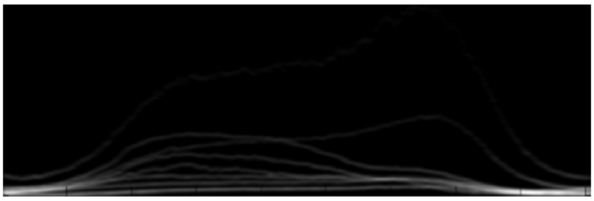

Figure 3: The map of the smoothness: form $G$. To solve the minimization for laces-functional (1) we take as a lace $H = \{h_j\}$, a map of "ridges" (local maximum in the vertical) of the form G. There, the closest laces just mergeed in these ridges. We shift the centers of the clusters along the vertical form and obtain "the lace". Curves with similar values should be merged into a single curve in *the lace*.

In the first notations, let the set $F = \{f_k\}$ be the centers of clusters. Define $G\ (F)$ , a smoothed map.
The lace ( set $G = \{g_k\}$,) we get via the form $G\ (F)$, as the minimization of the functional:

$$g_k = arg \min_g \left[ \lambda \|g - f_k\|^2 - \phi \oint_g G^2\ (F) + \gamma \oint_g \left\| \frac{\partial G(F)}{\partial y} \right\|^2 \right]$$

This task can be solved iteratively:

$$g_k(t+1) = \lambda f_k + \phi \left. \frac{\partial G(F)}{\partial y} \right|_{g_k(t)} - \lambda \left. \frac{\partial^2 G(F)}{\partial y^2} \right|_{g_{k(t)}}$$

Finally, we obtain $g_k = \lim_{t \to \infty} g_k(t)$ a local maximum for

$$N_w(F) = \sum_k \|g_k - f_k\|^2 + \eta_w(G)$$

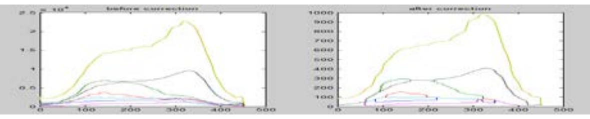

Figure 4: Left - the centers of clusters. Right - lace.
The x-axis is the time.

## 2.2 Constrction of Markow Chain

We select the key points in time according to the formula Key_time = 1: k: t(end), where k be the time interval.
Each point h<sub>i</sub>(t<sub>j</sub>) of the lace at a key point Key_time is corresponds with the vertex of the Markov chain. Each vertex of the chain corresponds to a stable state of the traffic system.

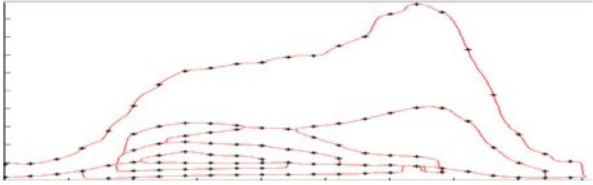

Figure 5: Red correspond lace $H = \{h_j\}$,. Black are the vertex of Markow chain E= { h<sub>i</sub>(t<sub>j</sub>)}.. If two black points are connected by an edge, then the probability of a transition in 1 step is 1, otherwise 0.

Let E= $x_1 x_2, .., x_N$ = {h<sub>i</sub>(t<sub>j</sub>)} be the key points in lace corresponds to a stable state of the traffic system.

Define the arrows $G$ as all possible pairs of neighboring points in laces G= $\{h_k(t_i), h_k(t_{i+1})\}_{k,i}$
So, [xi,xj] belongs to G iff x$_i$=$h_k(t_i)$,   x$_j$=$h_k(t_{i+1})$.

Definition3. We denote the lace graph as $V = (E, G)$ and define the Markov matrix as $P = (p_{ij})$.

$$p_{ij} = \begin{cases} 0, (i,j) \notin G \\ 1, (i,j) \in G \end{cases}$$

$P$ — Matrix transition from state to state in 1 step.
We define the transition matrix for an arbitrary number of steps

$$\hat{P} = \lim_{t \to \infty} \sum_{k=1}^{t} P^k \qquad (2)$$

# 3. MODEL OF ANOMALIES

Given the new traffic behavior in a single day, represented by temporally series y(t) , lace graph $V = (E, G)$ and the Markov matrix $P$, defined in previous training stage of the algorithm.

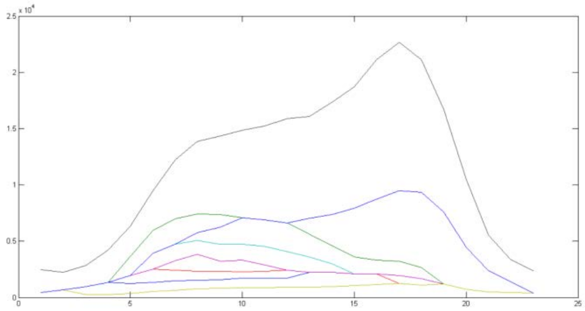

Figure 6: The representation of Markov Chain $V = (E, G)$

The goal is the mapping curve y(t) into lace.

## 3.1 Splitting traffic curve.

Splitting traffic is defined as a uniformly distributed set of states between the upper and lower boundary of the corridor, as shown in Figure9

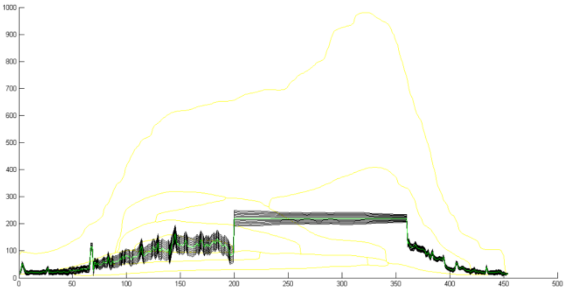

Figure 7: Yellow: the lace; green: current traffic; black: the traffic splitting.

Each layer of split traffic is compared with the lace. Looking for the lace phrase, most similar to the behavior of traffic.
Let $X = \{x_k\}$ be the splitting traffic curve and $H = \{h_j\}$ be the lace. For each traffic layer $i$, we look for its projection on the lace based on the minimization of the functional :

$$L_k(y) = \|y - x_k\|^2 + \left\| \min_i (y - h_i) \right\|^2 \xrightarrow{y} \min$$

Where $y_k = arg \min(L_k)$

We seek a solution using map of the smoothness (form $G$) via iteration series.

$$X_{k+1} = X_k + (dG/dy)(X_k)$$

As a result, we get the traffic splitting projection onto the lace Y = {y$_k$} as shown in Figure 10

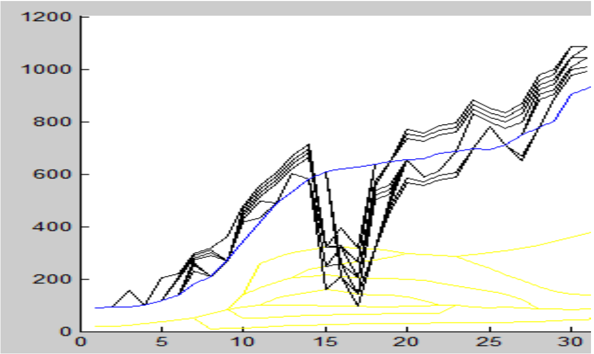

Figure 8: Yellow: the lace; black: splitting of current traffic metric after mapping; blue: the lace phrase, most similar to the behavior of traffic.

In notation of 2.2 let {t$_k$} be the key points of the time.
Then for any moment t$_k$ we have distribution of states {Y$_p$(t$_k$)}$_p$. Define p$_i$(k)=#{p| Y$_p$(t$_k$)}= h$_i$(t$_k$)}/N where N be the length of traffic splitting. Then W(k)={p$_i$(k)}$_i$ be probability distribution at the time moment k for current traffic to be in the lace states. So, we have the representation of current traffic behavior via flow of probability distribution W(k) to be in the Markov Chain $V = (E, G)$ states.

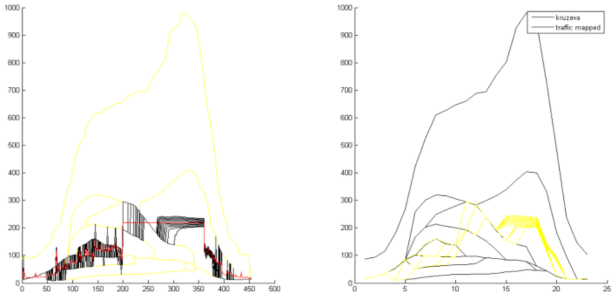

Figure 9: Left side: projection of traffic splitting into lace.
Right side: black is the representation of Markov Chain.
Yellow is flow of probability distribution W(k)

## 3.2 The probabilistic model of the anomalies.

At each key point in time, we have a random process $W = \{W_k\}$, where at each moment of time

$$W_k = \langle p_1, p_2, \ldots, p_N \rangle$$

is the probability distribution be in one or another state of the lace, N is the length of the splitting,

$$\sum_i p_i = 1, \quad 0 \leq p \leq 1$$

Now let $M$ be the Markov matrix (2), and $W$ the current state. Then $W_f = MW$ is the probability distribution of the transition from a given state to another in the future. $W_p =$

$W^T M$ is the probability distribution of states from which one can come to the state $W$.
Now calculate the transition probability $(t) \to W(t+1)$ . If the probability is 0, then we assume that this transition is anomalous. Let $W(t)$ be the current state, $r$ the parameter of the determinism of the model ($r = 1$, the Markov model, $r > 1$, the Bayesian model) $W(t-1),..,W(t-r)$ are previous states. $W_f(t-1),..,W_f(t-r)$ are the states to go from previous states via Markov chain. The question is: what is the relationship with current state W(t) and the history of previous states $W(t-1),..,W(t-r)$ ?
The following formula shows the probability distribution of such a state that it would be possible to flow from past states along the Markov chain

$$Wd_t = \prod_{k=1}^{d} W_f\,(t-k)$$

The following formula characterizes the measure of the anomaly of the current state.

$$Xi(t,d) = P(t|t-1,..,t-d) = \sum W(t)\, Wd_t \qquad (4)$$

This probability measure (between 0 and 1) shows the possibility of being in the current state $W(t)$ via history of past states $W(t-1),..,W(t-r)$. In the case when the nonzero probability support of $Wd_t$ overlaps the support of the nonzero probability W (t), then the probability of the transition between $W(t-1),..,W(t-r)$ and current state W(t) will be 1. Conversely, if the supports are not intersect, then the probability of transaction is 0. In this case we have an anomaly in $W(t) \to W(t+1)$.
Thus, we have obtained a function $Xi(t,d)$, that characterizes the measure of anomaly of the current state. The parameter d is associated with long history considered. In conclusion, we give some examples of the behavior of the anomaly measure on real data.

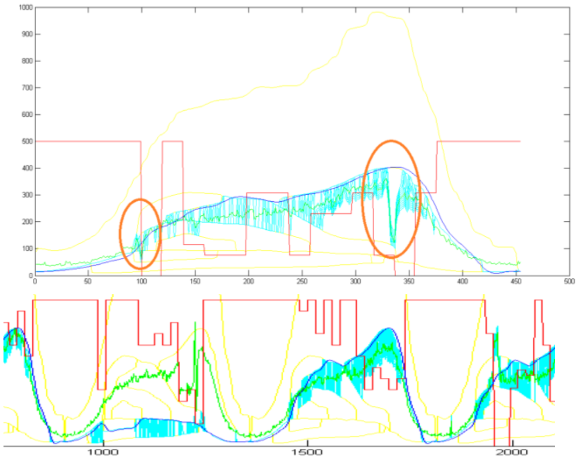

Figure 10. Yellow: the lace; Green: traffic metrica; Cyan: splitting of current traffic metric after mapping; blue: the lace phrase, most similar to the behavior of traffic; Red: the anomaly measure $Xi(t,d)$

## 4. TEST

During research 4 domains from database were tested. Anomalies were successfully detected by 97%. Example

Comparison of the obtained present method with the projection on the PCA [1] we afford in the form of confusion matrix

| Column1 | anomalies | background |
|---|---|---|
| anomalies | 0,97 | 0,03 |
| background | 0,02 | 0,98 |

Table 1: distribution of the “false-positive” and “true-negative” for the result of presented algorithm.

| Column1 | anomalies | background |
|---|---|---|
| anomalies | 0,63 | 0,37 |
| background | 0,29 | 0,71 |

Table 2: distribution of the “false-positive” and “true-negative” for the result of projection on PCA.

## REFERENCES

[1] Zheludev Michael, Nagradov Evgeny. Traffic anomaly detection and DDOS attack recognition using diffusion map technologies, *IEEE Computer Science and Information Technologies (CSIT)*, 2015

[2] Amir Averbuch and Michael Zheludev, Two Linear Unmixing Algorithms to Recognize Targets Using Supervised Classification and Orthogonal Rotation in *Airborne Hyperspectral Images Remote Sens*. 2012, 4(2), 532-560; doi:10.3390/rs4020532

[3] Michael Zheludev & Valery Zheludev, Unmixing and Target Recognition in Airborne Hyper-Spectral Images Amir Averbuch, *Earth Science Research*; Vol. 1, No. 2; 2012 ISSN 1927-0542 E-ISSN 1927-0550 Published by Canadian Center of Science and Education