Hinc itur
ad astra

Investigation of network intrusion detection using data visualization methods

VIKTORAS BULAVAS

PHD STUDENT AT VISUALIZATION SCHOOL OF DATA SCIENCE AND DIGITAL TECHNOLOGIES INSTITUTE,

MATHEMATICS AND INFORMATICS FACULTY OF VILNIUS UNIVERSITY

OCTOBER 12, 2018

ITMS 2018 Riga

Keywords

- Cybersecurity
- Decision Trees
- Machine-learning
- Network intrusion detection
- Principal Component Analysis
- Visualization

Principal Components Analysis

- In this paper, attention is drawn to linear projection, in particular Principal Component Analysis (PCA), as introduced by Hotelling [29], helping to select the most informative dimensions for intrusion detection.
- Brauckhoff, Salamatian and May [30] discuss implementing PCA method for anomaly detection and issue of right number of Principal Components for analysis.
- PCA, together with Decision Tree, can be successfully used for traffic feature extraction and intrusion classification.

Principal Components Analysis Defined

- The common definition of Principal Component Analysis (PCA) was introduced by Hotelling [3]:
- It is said, that for a set of observed vectors $\{u_i\}$, i ∈ {1,…, N}, where N is number of vectors, the *q* principal axes $\{E_j\}$, $j \in \{1,\ldots q\}$ are those orthogonal axes onto which the retained variance under projection is maximal.
- It can be shown that the vectors E_j are given by the *q* dominant eigenvectors of the covariance matrix C of vector v, such that eigenvectors E_j and corresponding eigenvalues λ_i are solution to $CE_i = \lambda_i E_j$ equation.
- The vector $v_i = E^T(u_i - \bar{u})$, where $E = (E_1 \ldots . E_q)$, is thus a *q*-dimensional reduced representation of the observed vector u_i.

Findings

- Investigation in this research demonstrates, that combination of PCA and Decision Tree methods allows classification of intrusions such as:
    - smurf,
    - satan,
    - neptune,
    - portsweep,
    - ipsweep
- with probabilities higher than 95% with depth of tree set to 4 and PCA components set to 10.
- Nevertheless, nmap and teardrop intrusions are classified purely, therefore deeper Decision Tree is needed to increase classification accuracy.

Abstract

- There are numerous sources for network intrusion detection data: for example, network traffic, system host logs, user activity, such as mail or browsing, use of smart devices and similar. All this data comes in big volumes, velocity and variety.
- Analysis of such data is essential for making anomaly detection and intrusion prevention decisions.
- Common data processing steps, following the acquisition of data, are projection, which helps to reduce the number of dimensions, and visualization, which helps observation of distinct features in real time.
- Both steps, further discussed in this paper are required for better understanding of contained intrusion phenomena, such as data theft, malware activity or hacking attempts.

Abstract (continued)

- Machine learning, which is more and more often used for preparing of network and related activity data, helps reducing data complexity, supports discovery of anomalies and speedups related decision-making.
- Visualization helps further understand data by elaborating the well-hidden data properties and features.
- Numerous methods of multi-dimensional data visualization are currently available to assist data scientist or information security analyst in the broad landscape of intrusion data analysis.

Intrusion detection problem

- Current network intrusion detection (NID) appliances utilize three main technics:
- anomaly detection,
- misuse detection and
- hybrid.

Intrusion detection problem

- Misuse detection systems use signatures that describe already known attacks and require regular ruleset update.
- Anomaly detection, on the other hand, consists of building models from normal data and then detect variations from the normal model in the observed data. Anomaly detection was originally introduced by Anderson [7] and Denning [8].
- The main advantage with anomaly detection algorithms is that they can detect new forms of attacks, because these new intrusions will probably deviate from the normal behavior [8].

Visualization methods

- Dzemyda, Kurasova and Zilinskas [2] classify visualization methods into direct visualization and projection visualization methods:
- 1. Direct visualization methods (when features of a multi-dimensional object are presented in a certain visual form). Using these methods, the selected dimensions of data are presented in a visual form on a two dimensional plane.
  - Direct visualization methods can be further classified as geometric, symbolic and hierarchical.

[2] G. Dzemyda, O. Kurasova, and J. Zilinskas, Multidimensional Data Visualization. Springer, 2012

Linear and nonlinear projection methods

- 2. Linear and nonlinear projection methods help presenting multidimensional objects in a smaller number of dimensions of space, (also known as dimension reduction methods).
  - Linear projection visualization methods can be further classified into Principal Component Analysis (further - PCA), Linear Discriminant Analysis (further – LDA) and Projection Pursuit.
  - Non-linear projection methods can be further classified into Multi Dimensional Scaling, Locally Linear Embedding, Isometric Feature Mapping, Principal Curves [2].

Data Sources

Network data

- The router or switch has the ability to collect IP network traffic as it enters and exits the interface (flows).

Host Data

- The host has the ability to generate system level and user behavior data, usually not obtainable directly from network flows, but related on a temporal axis.
- Such data would be for example failed login attempts. All of this data has clear temporal dimension, which is needed for real life observation of intrusion.

Datasets defined

- **Primary sources [1] of intrusion detection data, further defined as datasets, are network flows from other network domains and local network, enriched with host-based user behavior and system level content, as shown on *Figure 1*, which is needed to detect anomalous behavior and various types of intrusion attacks.**

[1] R. Koch, “Towards Next-Generation Intrusion Detection,” in *2011 3rd International Conference on Cyber Conflict*, 2011, February, pp. 151–168.

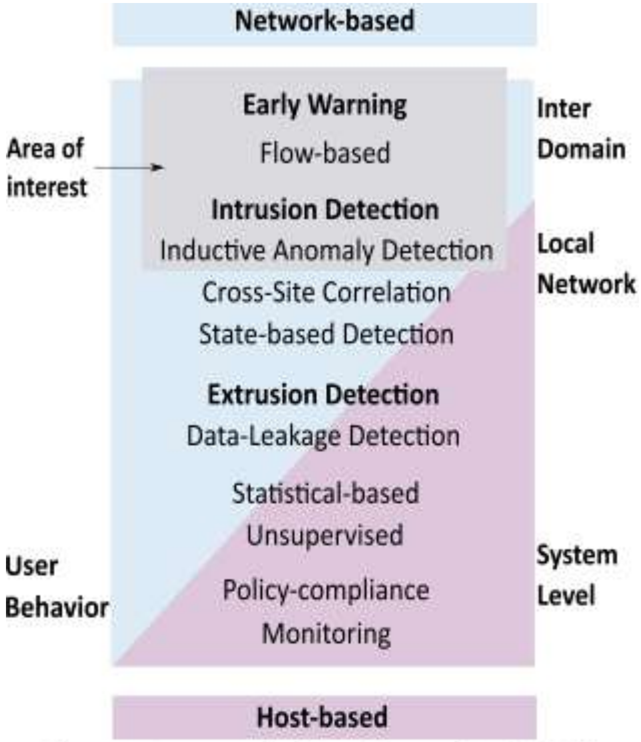


Figure 1. Layers of a Next-Generation IDS [1].

Definition of NetFlow

- A network flow is predominantly defined as a unidirectional sequence of packets that share the exact same packet attributes: ingress interface, source IP address, destination IP address, IP protocol, source port, destination port, and IP type of service.

Dataset used

- One of most easily accessible for research and education purposes intrusion detection datasets is KDD’99, generated for KDD Cup Contest of 1999.
- It has been updated as NSL–KDD and made available for download at University of Brunswick, Canada.
- The NSL–KDD dataset consists of 41 dimensions [14].

[14] Y. Bouzida and F. Cuppens, “Efficient intrusion detection using principal component analysis,” Proc., 2004.

Features in NSL-KDD

- 1) Basic features: attributes that can be extracted from a TCP/IP connection (ingress interface, source IP address, destination IP address, IP protocol, source port, destination port, and IP type of service).
- 2) Host features: examine only the connections in the past 2 seconds that have the same destination host as the current connection, and calculate statistics related to protocol behaviour, service, etc.

Features in NSL-KDD (continued)

- 3) Service features: examine only the connections in the past 2 seconds that have the same service as the current connection. However, now popular slow probing attacks scan the hosts using a time interval as defined by botnet control centre.
- 4) Content features: unlike most of the DoS and Probing attacks, the R2L and U2R attacks don’t have a similar sequential pattern. The R2L and U2R attacks are embedded in the data portions of the packets, and normally represent only a single connection. To detect such attacks, IDS needs specific features in the data portion to recognise as an anomaly, for example a number of failed login attempts. These features are called content features.

Best representing data features for intrusion detection

- Amiri [33], Olusola [34], Zargari [35] and others, based on PCA analysis, proposed methods of selecting the best representing data features of NSL-KDD for intrusion detection:
- Service, Source bytes, Destination bytes and Destination host error rate.
- These features explain about 97% of variance. The remaining 37 features explain up to 99,7%, and 80 network features predict 99,97% of attacks.

Experiment

- The objective of experiment in this research was to visualise different types of attack data, available in the NSL-KDD dataset.
- Particular attention is drawn to linear projection, in particular principal components analysis, helping to select the most informative dimensions.
- Principal components analysis method, that provides indication of anomalies in network and host data are further reviewed and presented in this paper.
- Decision Tree method is utilized to provide decision criteria for anomaly recognition as intrusion.

Open Source Data Analytics Orange 3

- A simple Orange 3 workflow is used for visualization with ScatterPlot

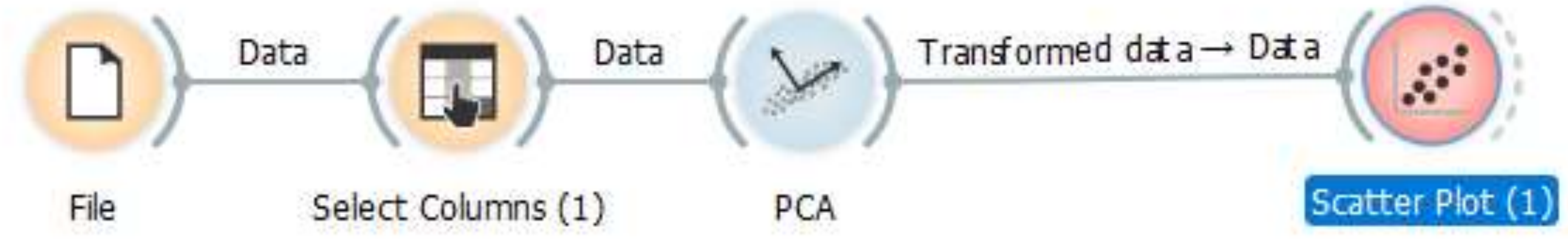


Fig. 4. Principal Component Analysis workflow using Orange 3 software.

Principal Component Analysis of NSL-KDD using Orange 3 software

Fig. 5. Principal Component PC1-PC2

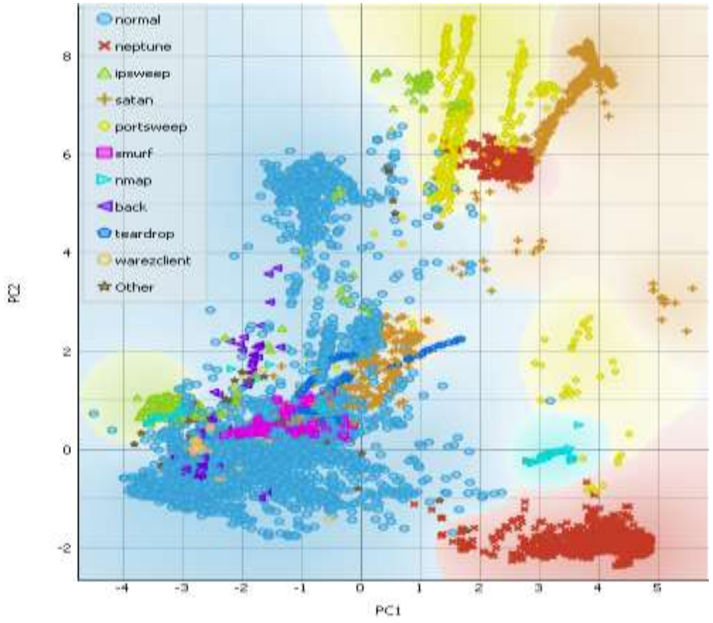
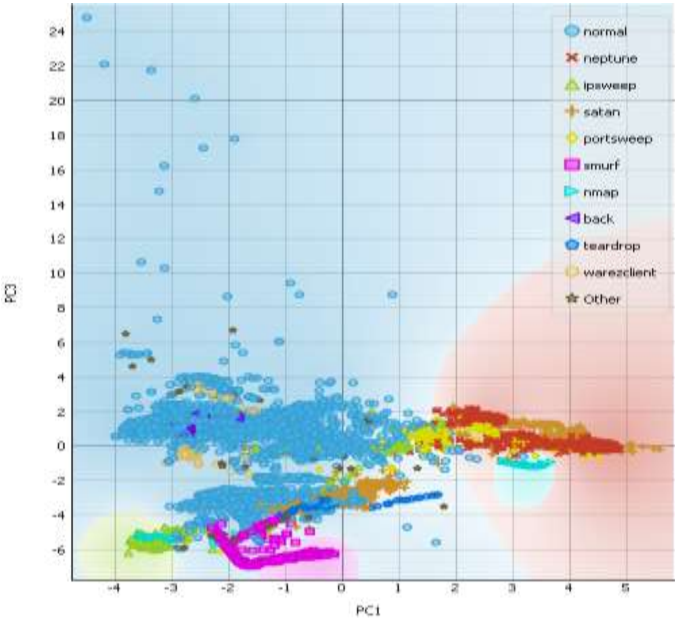


Fig.6. Principal Component PC1-PC3



Open Source Data Analytics Orange 3

- For the purpose of experiment reproducibility, related Orange workflow for PCA analysis with Decision Tree is presented in Fig. 7.

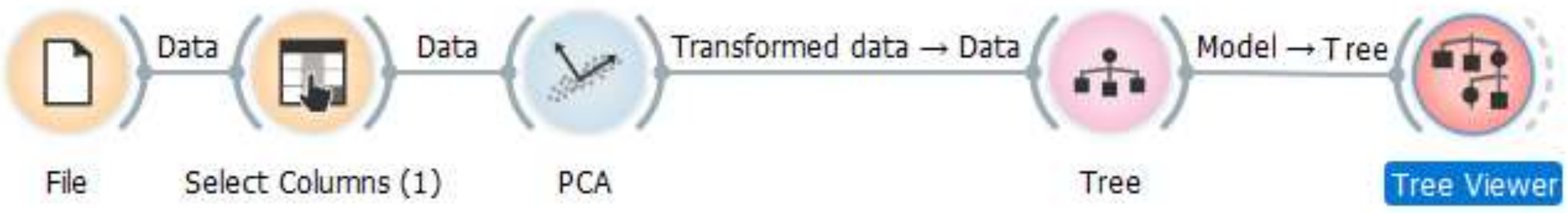


Fig.7. PCA and Decision Tree Analysis workflow using Orange 3 software

Decision Tree for NSL-KDD

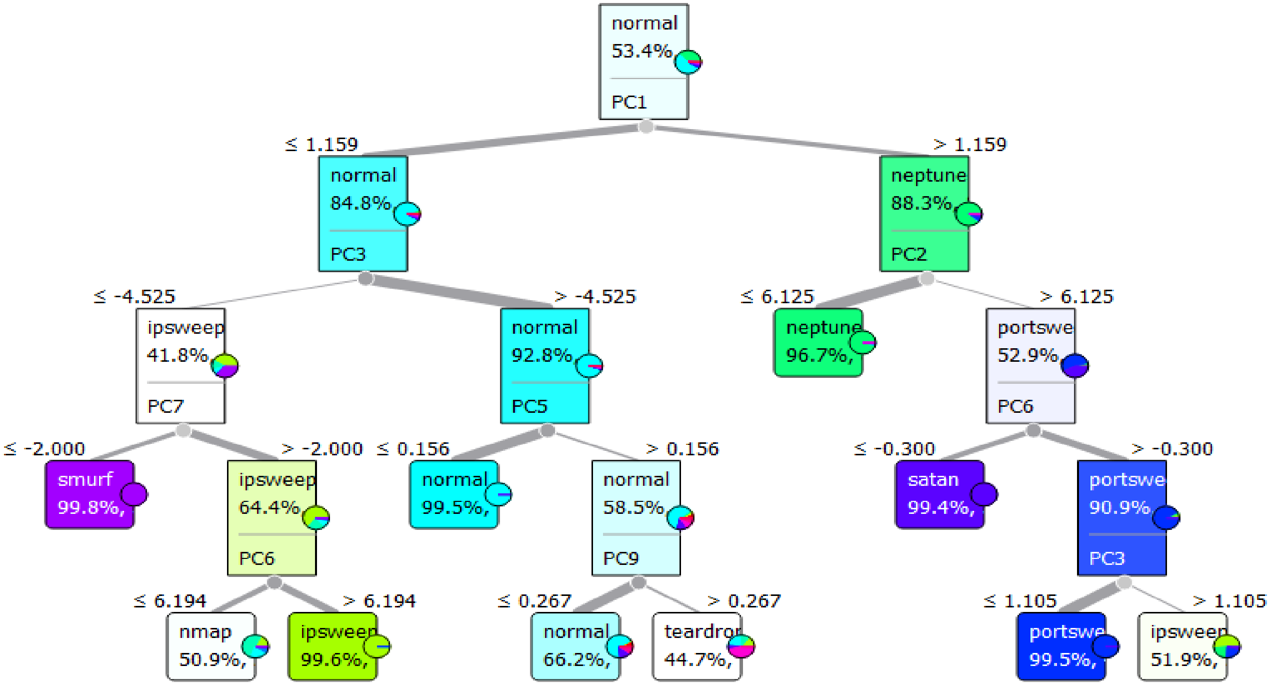


Fig.8. Decision Tree for NSL-KDD data using Orange 3 software.

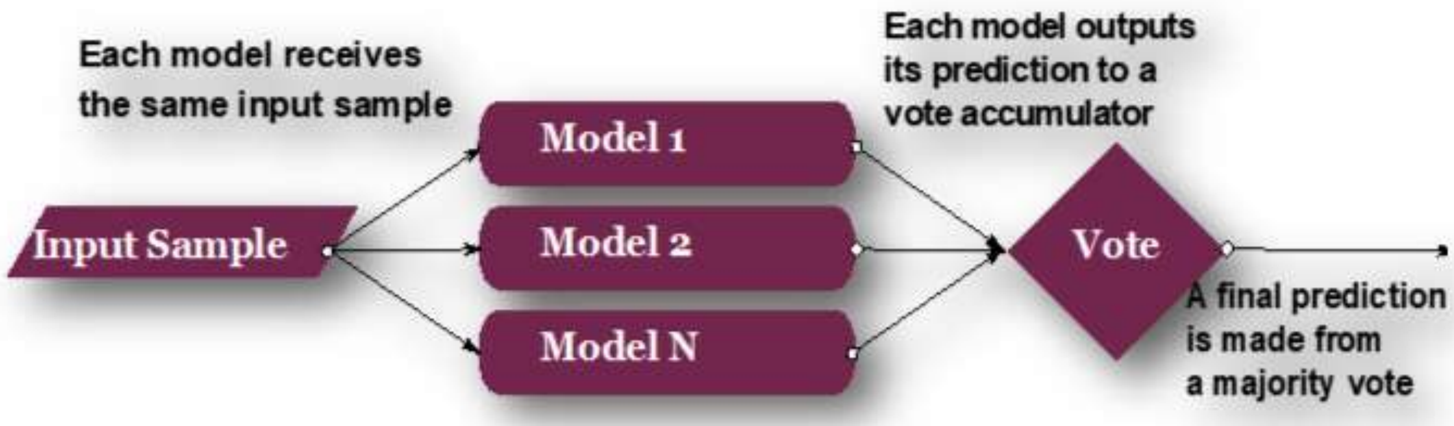
Machine learning approaches used in intrusion detection

ML approaches used in intrusion detection include:

- Decision Trees,
- Inductive Learning,
- Naive Bayes,
- Random Forest,
- Artificial Neural Networks,
- Fuzzy Systems,
- Evolutionary Computation,
- Artificial Immune Systems,
- Hidden Markov,
- Sequential Pattern Mining,
- Swarm Intelligence
- and other [25], [26].

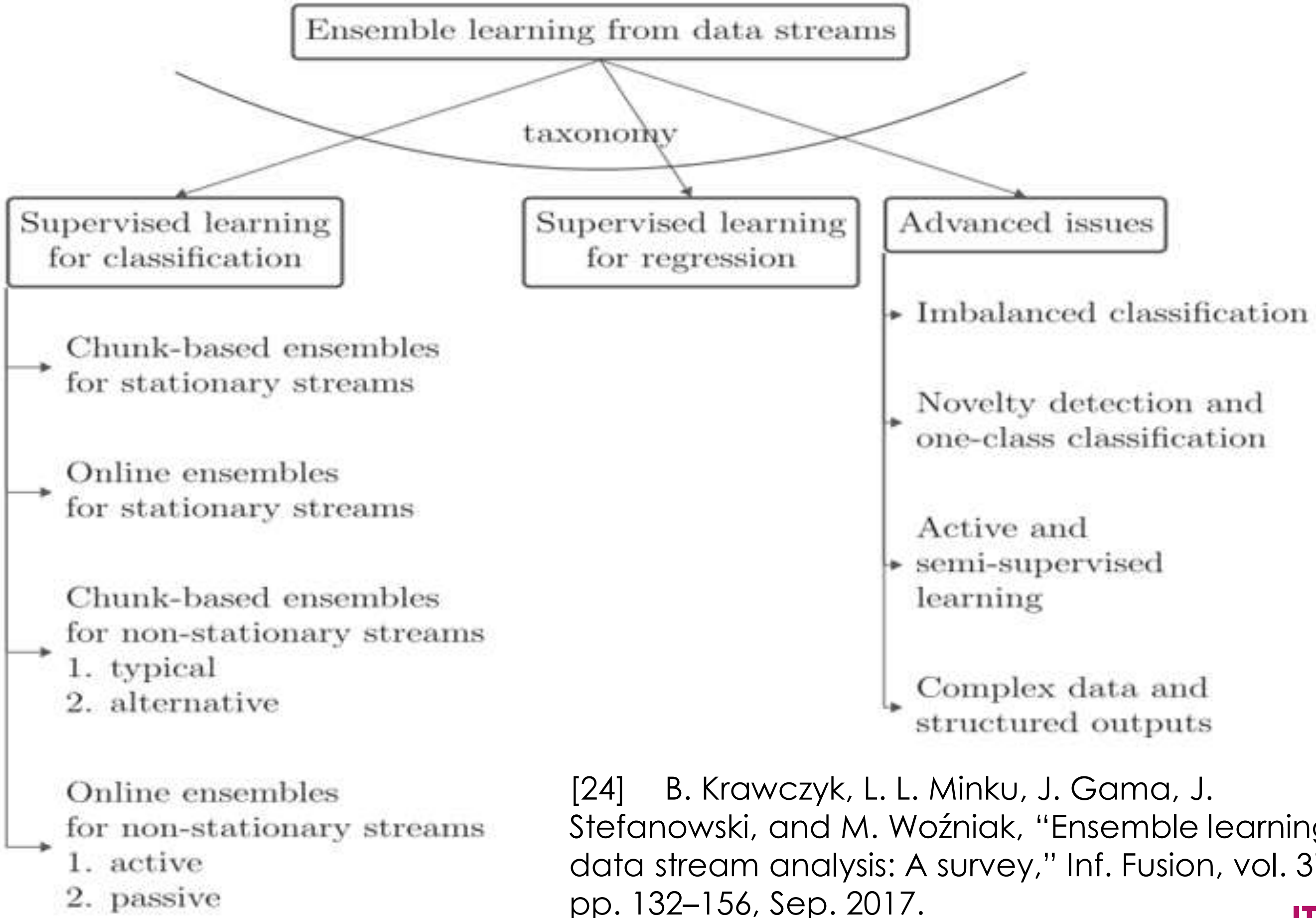
Implementing Ensemble

- Ensemble of Machine Learning models



Each model receives the same input sample
Input Sample
Model 1
Model 2
Model N
Each model outputs its prediction to a vote accumulator
Vote
A final prediction is made from a majority vote

Taxonomy of Ensemble learning from data streams [24]



Ensemble learning from data streams

taxonomy

Supervised learning for classification

- Chunk-based ensembles for stationary streams
- Online ensembles for stationary streams
- Chunk-based ensembles for non-stationary streams
  1. typical
  2. alternative
- Online ensembles for non-stationary streams
  1. active
  2. passive

Supervised learning for regression

Advanced issues

- Imbalanced classification
- Novelty detection and one-class classification
- Active and semi-supervised learning
- Complex data and structured outputs

[24] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, “Ensemble learning for data stream analysis: A survey,” Inf. Fusion, vol. 37, pp. 132–156, Sep. 2017.

Future work

- Future experiment and analysis could be performed using more detailed data source CIC IDS 2017 [6], with ML implemented on open source Tensorflow framework. According to Sharafaldin, Lashkari, and Ghorbani [13], the abovementioned source, enriched with 80 network features, contains more than 28 informative principal components.
- Implement model of conversion of network data into data frame, reproducing algorithms implemented by Kim and Reddy [23].
- Implementing Ensemble and checking if solutions, brought by Hinton et al with Capsule Networks, with learning layers, eliminating need of retraining with feeding of all the data from the beginning.

Thank you for attention!

Viktoras Bulavas

E-mail: viktoras.bulavas@itpc.vu.lt

References

- [1] D. a. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler, “Challenges in Visual Data Analysis,” in Tenth International Conference on Information Visualisation (IV’06), 2006.
- [2] G. Dzemyda, O. Kurasova, and J. Zilinskas, Multidimensional Data Visualization. Springer, 2012.
- [3] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” J. Educ. Psychol., vol. 24, no. 6, pp. 417–441, 1933.
- [4] J. P. Anderson, “Computer security threat monitoring and surveillance,” 1980.
- [5] D. E. Denning, “An Intrusion-Detection Model,” in 1986 IEEE Symposium on Security and Privacy, 1986, pp. 118–118.
- [6] R. Koch, “Towards Next-Generation Intrusion Detection,” in 2011 3rd International Conference on Cyber Conflict, 2011, no. February, pp. 151–168.
- [7] KDD, “KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html,” KDD Cup 1999 Data, 1999. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.
- [8] S. Hettich and S. D. Bay, “The UCI KDD Archive [http://kdd.ics.uci.edu],” Univ. California, Dep. Inf. Comput. Sci., 1999.
- [9] R. P. Lippmann et al., “Evaluating intrusion detection systems without attacking your friends: The 1998 DARPA intrusion detection evaluation,” DARPA Inf. Surviv. Conf. Expo. 2000. DISCEX ’00. Proc., pp. 12–26 vol.2, 1999.

References

- [10] The Cooperative Association for Internet Data Analysis, “CAIDA - The Cooperative Association for Internet Data Analysis,” CAIDA. 2010.
- [11] Lawrence Berkeley National Laboratory, “The Internet Traffic Archive,” 2010. [Online]. Available: http://ita.ee.lbl.gov/index.html.
- [12] The Shmoo Group, “Defcon,” 2011. .
- [13] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization,” in Proceedings of the 4th International Conference on Information Systems Security and Privacy, 2018, no. January, pp. 108–116.
- [14] Y. Bouzida and F. Cuppens, “Efficient intrusion detection using principal component analysis,” Proc., 2004.
- [15] K. Lakkaraju, W. Yurcik, and A. J. Lee, “013 NVisionIP: NetFlow Visualizations of System State for Security Situational Awareness,” Proc. 2004 ACM Work. Vis. data Min. Comput. Secur. - VizSEC/DMSEC ’04, 2004.
- [16] X. Y. X. Yin, W. Yurcik, Y. L. Y. Li, K. Lakkaraju, and C. Abad, “113 VisFlowConnect: providing security situational awareness by visualizing network traffic flows,” IEEE Int. Conf. Performance, Comput. Commun. 2004, 2004.
- [17] S. Musa and D. J. Parish, “Visualising communication network security attacks,” in Proceedings of the International Conference on Information Visualisation, 2007.
- [18] J. McPherson, K.-L. Ma, P. Krystosk, T. Bartoletti, and M. Christensen, “PortVis: A Tool for Port-Based Detection of Security Events,” Proc. Int. Symp. Vis. Cyber Secur. - VizSec, p. 73, 2004.

References

- [19] K. Abdullah, C. Lee, G. Conti, J. A. Copeland, and J. Stasko, "IDS RainStorm: Visualizing IDS alarms," in IEEE Workshop on Visualization for Computer Security 2005, VizSEC 05, Proceedings, 2005.
- [20] Y. Livnat, J. Agutter, S. Moon, R. F. Erbacher, and S. Foresti, "A visualization paradigm for network intrusion detection," Proc. from 6th Annu. IEEE Syst. Man Cybern. Inf. Assur. Work. SMC 2005, 2005.
- [21] J. R. Goodall, "Introduction to Visualization for Computer Security," in VizSEC 2007, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–17.
- [22] F. Fischer and D. a Keim, "VACS: Visual Analytics Suite for Cyber Security," IEEE VAST Chall. 2013, 2013.
- [23] S. S. Kim and A. L. N. Reddy, "A study of analyzing network traffic as images in real-time," in Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies., 2005, vol. 3, pp. 2056–2067.
- [24] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," Inf. Fusion, vol. 37, pp. 132–156, Sep. 2017.
- [25] F. Gharibian and A. A. Ghorbani, "Comparative Study of Supervised Machine Learning Techniques for Intrusion Detection," in Fifth Annual Conference on Communication Networks and Services Research (CNSR '07), 2007.
- [26] A. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," IEEE Commun. Surv. Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.
- [27] D. H. Wolpert, "The Supervised Learning No-Free Lunch Theorems," Proc. 6th Online World Conf. Soft Comput. Ind. Appl., vol. 50 Suppl, pp. 25–42, 2001.

References

- [28] Y. Chen, A. Abraham, and B. Yang, "Feature selection and classification using flexible neural tree," Neurocomputing, vol. 70, no. 1–3, pp. 305–313, Dec. 2006.
- [29] D. Brauckhoff, K. Salamatian, and M. May, "Applying PCA for Traffic Anomaly Detection: Problems and Solutions," in IEEE INFOCOM 2009 - The 28th Conference on Computer Communications, 2009, pp. 2866–2870.
- [30] H. Ringberg, A. Soule, J. Rexford, and C. Diot, "Sensitivity of PCA for traffic anomaly detection," ACM SIGMETRICS Perform. Eval. Rev., vol. 35, no. 1, p. 109, Jun. 2007.
- [31] C. Issariyapat and K. Fukuda, "Anomaly detection in IP networks with principal component analysis," 2009 9th Int. Symp. Commun. Inf. Technol., pp. 1229–1234, 2009.
- [32] K. Keerthi Vasan and B. Surendiran, "Dimensionality reduction using Principal Component Analysis for network intrusion detection," Perspect. Sci., vol. 8, pp. 510–512, 2016.
- [33] F. Amiri, M. Rezaei Yousefi, C. Lucas, A. Shakery, and N. Yazdani, "Mutual information-based feature selection for intrusion detection systems," J. Netw. Comput. Appl., vol. 34, no. 4, pp. 1184–1199, 2011.
- [34] A. A. Olusola, A. S. Oladele, and D. O. Abosede, "Analysis of KDD & apos ; 99 Intrusion Detection Dataset for Selection of Relevance Features Analysis of KDD ' 99 Intrusion Detection Dataset for Selection of Relevance Features," vol. I, no. January, pp. 16–23, 2016.
- [35] S. Zargari and D. Voorhis, "Feature selection in the corrected KDD-dataset," in Proceedings - 3rd International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2012, 2012.
- [36] J. Demšar et al., "Orange: Data Mining Toolbox in Python," J. Mach. Learn. Res., vol. 14, pp. 2349–2353, 2013.