



Machine Learning Applied to Cyberattack Prediction

Avelino Cosme^{a,*}, Nuno Mateus-Coelho^{a,b,c}

^a*Lusófona University Centro Universitário de Lisboa - COPELABS, Lisbon, Portugal*

^b*CTS - Centre of Technology and System, UNINOVA, Lisbon, Portugal*

^c*LAPI2S - Laboratory of Privacy and Information Systems Security, Porto, Portugal*

^a*Email: avelinocosme@gmail.com*

^{a,b,c}*Email: nuno.coelho@ulusofona.pt*

Abstract

This article summarizes the development of machine learning models to classify network traffic by extracting features from network packets. These models are helpful in intrusion detection and prevention systems (IDPS), which can improve the identification of malicious packets and reduce false positive attacks. Including these models in IDPS systems makes them more efficient and robust. The article summarizes the laboratory developed in the master's dissertation in data science at Universidade Lusófona of Lisbon. In this work, you can find a substrate of the extraction phases at the data source, transformations, preprocessing, features selection, and selection of the model trained and tested with the data set under analysis.

During the laboratory, 11 models were trained and tested. Still, for this article, we present more information about the models that had the best results: Random Forest, XGBoost, and Decision Tree. Three scenarios were created for the same data set, which allowed us to study the behavior of the models with the change of scenarios. The first scenario refers to a binary classification problem, with two classes, that is, class 0 representing Normal Traffic and class 1 representing Attack Attempt. The second scenario refers to simple multiclass classification (different attack types), and the third and last scenario refers to balanced multiclass classification with the smote algorithm.

Ultimately, we propose reflecting on each scenario developed, highlighting two fundamental points. The first is the importance of a precise definition of the problem, which directly influences the transformations applied to the data and the selection of relevant attributes. The second point refers to the fact that achieving high accuracy rates in training and testing the model is not enough — it is crucial to consider the risk of data leakage. Variables that compromise the model's integrity, making its application in real production environments unfeasible, must be carefully identified and excluded.

Keywords: Machine Learning, Intrusion detection system (IDPS), Cyber-attacks, Vulnerabilities.

1. Introduction

With the emergence of new technologies, system vulnerabilities have increased, and cybersecurity has become challenging, as we cannot think of blocking an attack using only human operators. The advanced and automated ways in which attacks have been developed mean that the task of defense must be intelligent and automated, as is the case with the Intrusion Detection and Prevention System (IDPS), which is capable of identifying, creating alerts, and blocking intrusion attempts. These IDPS systems need to be intelligent to learn to classify new forms of attack and identify system vulnerabilities, thus keeping up with technological developments. Due to its particularities, Artificial Intelligence meets this need, as it has played an increasingly important role, especially when coupled with cybersecurity. [4], [5], [10] [11] [12] [13] [14].

2. Theoretical background

The development of IDPS solutions that allow attack prediction through real-time packet capture, feature extraction, and packet classification. If the packet is classified as normal, it is delivered to the destination. If the classification is an attack, it is immediately blocked, and alerts are generated [1], [4], [5], [10].

Table 1 IDPS Functions and Definitions

Sistem	Definition	Main Functions	Typical Actions
Firewall	Packet filtering system that controls traffic between networks. Control access	Block/allow traffic based on rules	Allow or block packets based on IP, port, protocol, etc.
IDS	<i>IDS: Intrusion Detection System</i>	Monitor traffic for malicious patterns -Generate alerts	- Generate alerts - Detect attacks - Send alerts - Does not directly interfere with traffic
IPS	<i>Intrusion Prevention System.</i>	Detect and block malicious traffic - Prevent attacks in real time	- Terminate connections - Block packets - Update rules dynamically

The article presents a case study developed in Data Science that has much more scope for the construction and selection of the machine learning model.

The main objective of this work was not only to find the best solution for classifying network packets, but also to conduct a comprehensive study that would allow an in-depth understanding of the behavior of machine learning models when faced with different formulations of the problem. We explored how the definition of the scenario directly influences the choice and transformation of independent variables, reflecting on the performance and applicability of the models. Throughout the study, we demonstrated that the effectiveness of a model is not limited to accuracy or precision metrics but also depends on critical decisions related to attribute engineering, data leakage prevention, and the feasibility of implementing the model in a real production environment. Based on this, this thesis offers a technical solution and a solid analytical basis for the responsible and robust development of intelligent intrusion detection systems. The data used in this work were obtained from the sets made available by the Canadian Institute for Cybersecurity (CIC), at the University of New Brunswick, available at: <https://www.unb.ca/cic/datasets/>.

Table 2 Columns of the Dataset

Group A	Group B	Group C	Group D
duration	Hot	num access files	srv error rate
protocol type	num failed logins	num outbound cmds	same srv rate
service	logged in	is host login	diff srv rate
flag	num compromised	is guest login	srv diff host rate
src bytes	root shell	count	dst host count
dst bytes	su attempted	srv count	dst host srv count
land	num root	error rate	dst host same srv rate
wrong fragment	num file creations	srv error rate	dst host diff srv rate
urgent	num shells	error rate	dst host same src port rate
attack	dst host srv error rate	dst host error rate	dst host srv diff host rate
last flag		dst host error rate	dst host srv error rate

Table 3 Attack records in the dataset

Type of attack	Total	Type of attack	Total	Type of attack	Total
NORMAL	76503	snmpguess	331	ftp_write	11
Neptune	45545	Saint	318	Loadmodule	11
Satan	4346	Mailbomb	292	Xlock	9
Ipsweep	3717	Pod	242	Phf	6
Smurf	3288	Snmpgetattack	178	Perl	5
Portsweep	3067	Httpunnel	133	Xsnoop	4
Nmap	1552	Butter_overflow	50	Worm	2
Back	1302	Multihop	25	SQLattack	2
Guess_passwd	1280	Land	25	Udpstorm	2
Mscan	994	rootkit	22	Spy	2
Warezmater	964	Named	17	Apache2	734
Teardrop	897	Ps	15	Processtable	683
Warezcilent	887	Sendmail	14	Xterm	13
imap	12			Total Registos	147500

Table 4 summary of the transformations applied to the data

Reference	Quantity	Action
Duplicated	601	dropped
Invalid Values	0	N/A
Null Values	0	N/A
Numeric Columns		
Data_Leakage	32	dropped
Outliers	N/A	Ajustados entre os quantis 1% e 99%
Coategorical Columns		
Column Removed	1	dropped
Null or Invalid Values	0	N/A
New Column	1	attack class
One-hot-encoding	3	One-hot-Encoding

3. Feature Selection

The selection and organization of input variables (features) play a central role in the effectiveness of machine learning models applied to intrusion detection. This study defined three scenarios (C1, C2, and C3), each with a distinct configuration of final attributes. This distinction is not arbitrary, but rather essential to ensure the models' performance, specialization, and coherence. Including irrelevant attributes can introduce noise into the data, making it difficult for the model to generalize. No feature selection can lead to overfitting, where the model learns spurious patterns from the training set that are not repeated in the real data.

Table 5 Multiclass Designation

Multiclass	Class designation	Attack grouping
0	Normal	Normal
1	DoS (Deny of service)	Neptune, smurf, back, teardrop, pod, land, apache2, udpstorm, processtable, mailbom.
2	Probe (Scanning)	Satan, ipsweep, portsweep, nmap, mscan, saint.
3	Unauthorized Remote Access	Guess_passwd, warezclient, warezmaster, imap, ftp_write, multihop, phf, spy, snmpgetattack, snmpguess, xlock, xsnoop, named, sendmail, worm
4	U2R (user-to-root) Privilege escalation	Buffer_overflow, rootkit, loadmodule, perl, sqlattack, xterm, ps, httptunnel.

Fewer attributes result in lower data dimensionality, which reduces the computational cost regarding training time, prediction time, and memory usage. This efficiency is critical in real-time applications, such as intrusion detection systems in corporate networks.

Different scenarios require different detection strategies. Scenario C1, for example, deals with a binary classification (attack or normal), while C2 and C3 deal with multiple classes of attacks. Adjusting the set of features according to the scenario's objective allows the model to specialize in recognizing specific patterns, increasing its performance.

Scenario 1: Binary Classification:

- Class 0: Normal, non-invasive packet.
- Class 1: Invasive packet, representing an attack.

Scenario 2: Multiclass Classification.

Scenario 3: Balanced Multiclass Classification.

In total, 32 variables representing information derived from analyses or calculations performed after the traffic were discarded. The use of these variables in machine learning models would cause data leakage, as they would not be available when the packet is actually analyzed on the network. To ensure the viability of implementation in a real environment, the model must be based only on variables directly extracted from the IP packets, that is, information accessible in real time.

At the end of the data preparation process, 8 numerical and three categorical variables were kept. The latter were transformed into 81 numerical variables using the One-Hot Encoding technique, ensuring that all categorical information was adequately represented for the machine learning algorithms.

Each scenario focuses on different categories of attacks, and certain attributes are more informative in specific contexts. For example, attributes such as wrong_fragment or service_other are more relevant in detecting denial of service (DoS) attacks. In contrast, services such as telnet and pop_3 indicate remote access (R2L) and user-to-root (U2R) attacks. Including only the attributes relevant to each scenario increases the model's sensitivity to the specific anomalies of interest.

Table 6 Features selection per scenario

Features	Meaning	C1	C2	C3
src_bytes	Number of bytes sent from source to destination	Yes	Yes	Yes
last_flag	Number of packets with the same last TCP flag	Yes	Yes	Yes
dst_bytes	Number of bytes sent from destination to source	Yes	Yes	Yes
service_http	Indicates whether the service used is HTTP	Yes	Yes	Yes
flag_S0	Indicates connection initiated but not completed (SYN sent, response absent)	Yes	Yes	Yes
service_private	Indicates services marked as private (non-standard)	Yes	Yes	Yes
flag_SF	Connection established and terminated normally	Yes	Yes	Yes
service_ftp_data	Indicates whether the service used is FTP (data)	Yes	Yes	Yes
duration	Connection duration (in seconds)	Yes	Yes	Yes
protocol_type_tcp	Indicates whether the protocol used is TCP	Yes	Yes	Yes
service_ecr_i	Indicates whether the service is Echo Reply ICMP (used in reverse ping)	Yes	Yes	No
flag_REJ	Connection rejected by destination host	Yes	Yes	Yes
service_domain_u	Indicates whether the service is DNS (UDP)	Yes	No	No
service_smtp	Indicates whether the service used is SMTP (email)	Yes	No	No
service_eco_i	Indicates whether the service is Echo ICMP (ping)	Yes	Yes	Yes
service_other	Service classified as 'other'	No	Yes	Yes
wrong_fragment	Number of wrong IP fragments	No	Yes	No
service_pop_3	Indicates whether the service used is POP3 (email)	No	No	Yes
service_telnet	Indicates whether the service used is Telnet	No	No	Yes

The correlation analysis between variables is essential because it allows us to identify highly correlated variables (with a coefficient close to 1), which may be redundant and therefore discarded to avoid multicollinearity. In addition, variables with low variance, close to zero, are also candidates for removal because they tend to contribute little to the model's predictive capacity, [2,3,4,5,6].

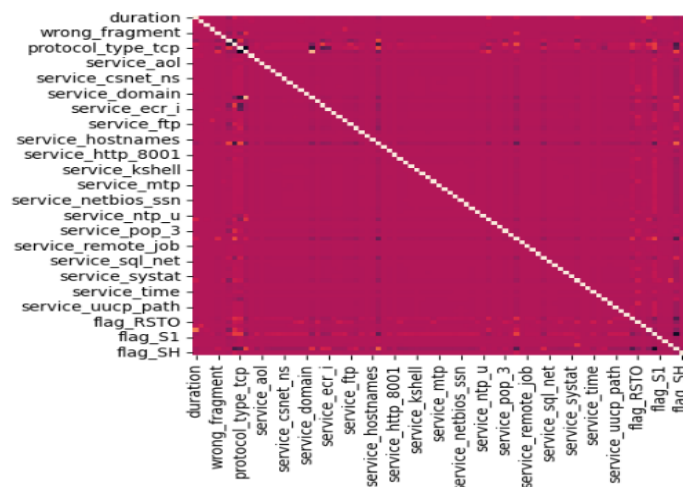


Figure 1 Correlation map of features

4. Metrics

To calculate the accuracy achieved by each model, a confusion matrix was built, which displayed true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [6], [7], [8], [9], [10].



Figure 2 Metrics

4.1. C1 - Binary Classification Results

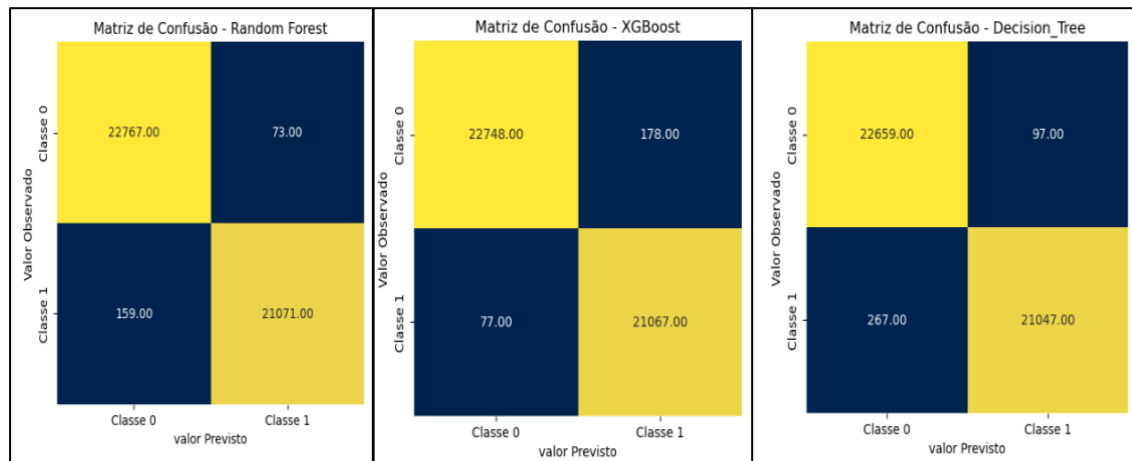


Figure 3 Confusion matrix top 3 models in C1

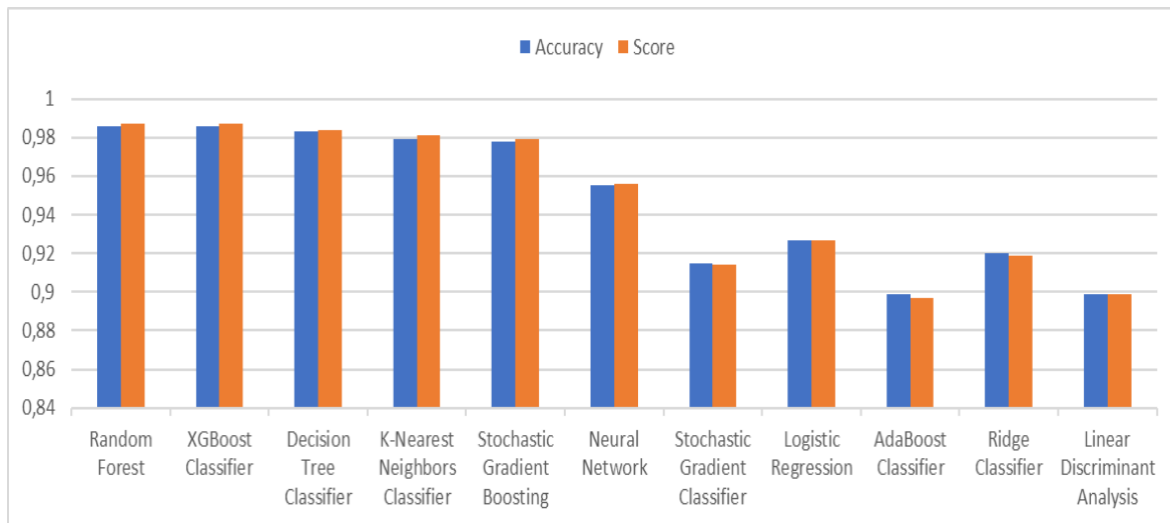


Figure 4 Accuracy and Score Metrics in C1

4.2. C2 - Multiclass Classification Results

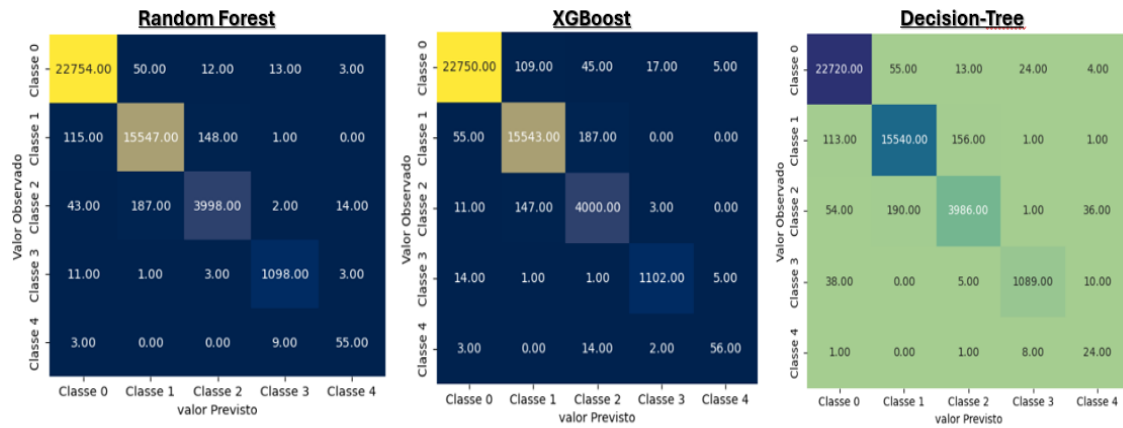


Figure 5 Confusion matrix top 3 models in C2

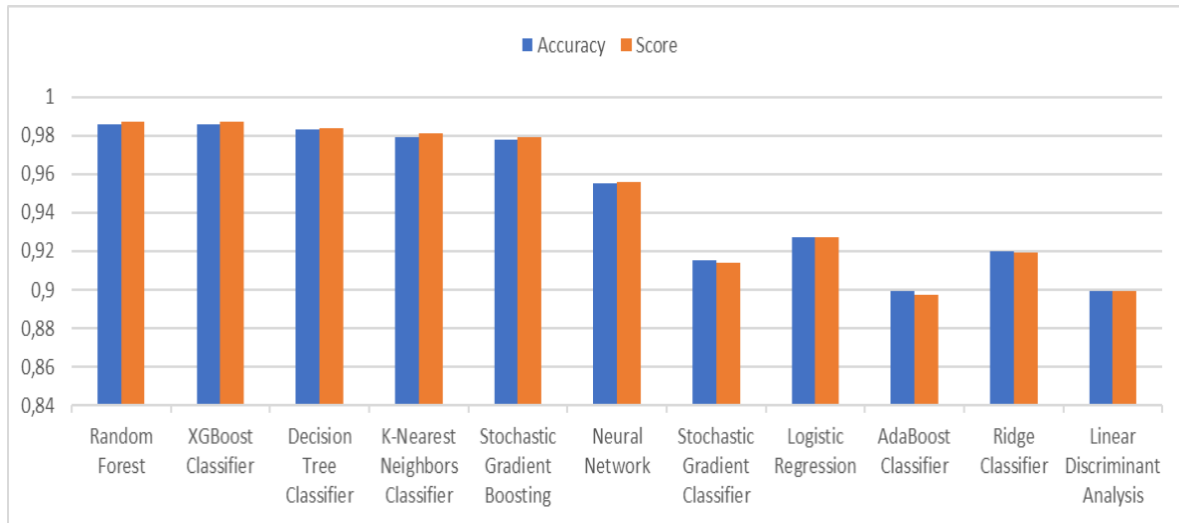


Figure 6 Accuracy and Score Metrics in C2

4.3. C3 - Balanced Multiclass Classification (SMOTE) Results

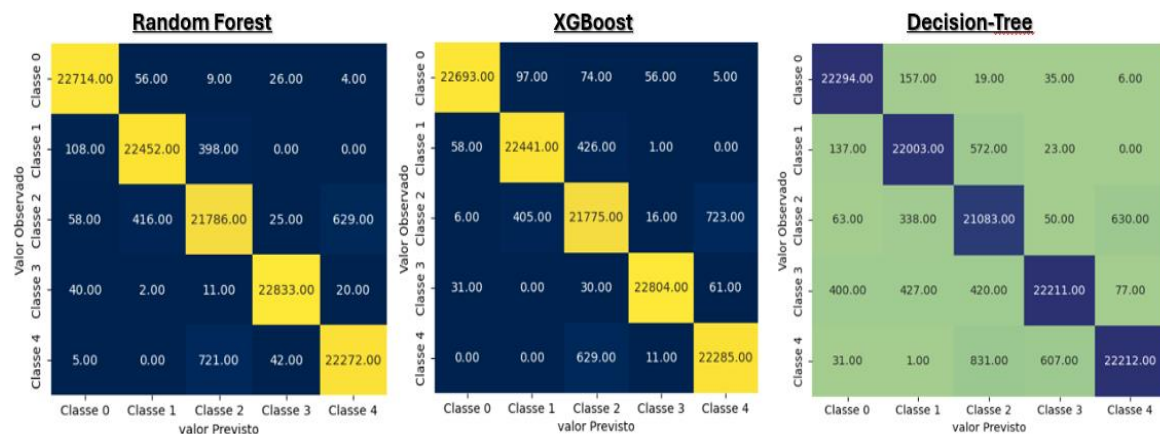


Figure 7 Confusion matrix top 3 models in C3

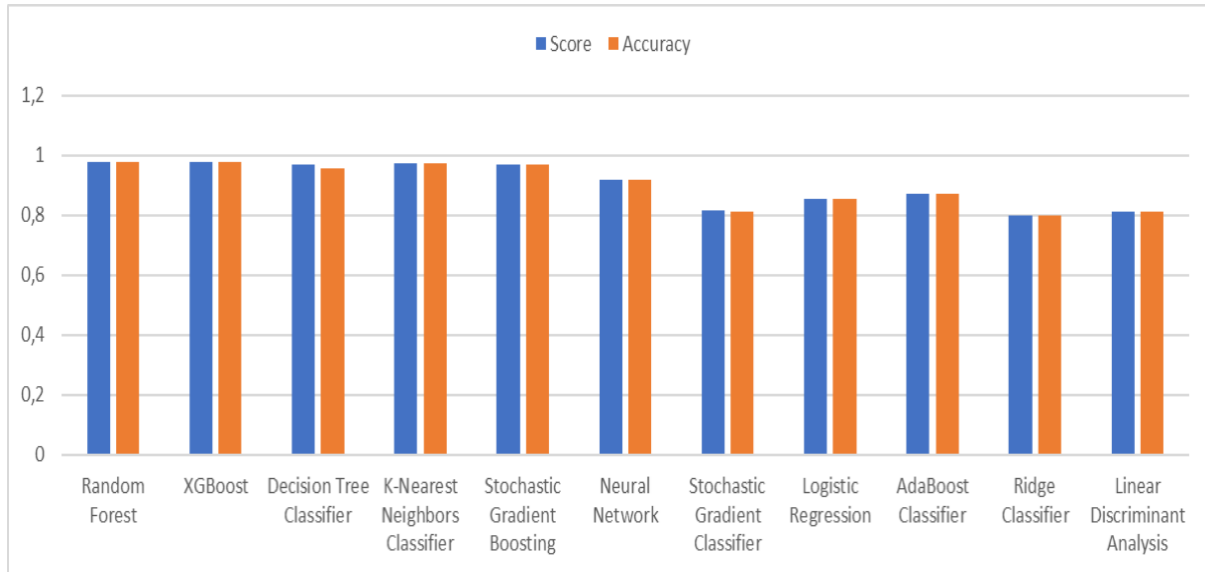


Figure 8 Accuracy and Score Metrics in C3

5. CONCLUSIONS

Among the eleven models evaluated, the three with the best performance were Random Forest, XGBoost, and Decision Tree. This predominance of algorithms based on decision trees reveals that the problem presents non-linear relationships between the variables, which are well captured by this type of approach. Random Forest stood out for its robustness against overfitting, in addition to its generalization capacity. At the same time, XGBoost presented competitive performance by iteratively optimizing the residual errors, proving effective in identifying complex patterns. The good performance of the simple decision tree indicates that, even with a basic structure, the model was able to learn relevant patterns, a direct result of the quality of the selected variables and the transformations applied.

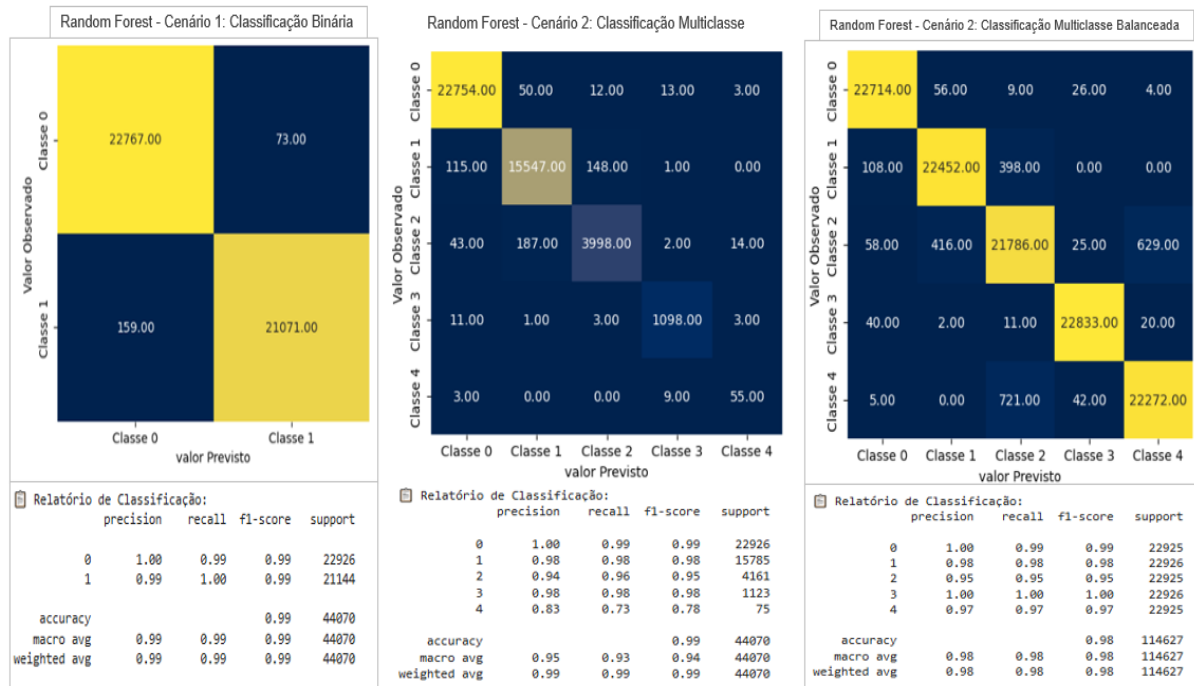


Figure 9 Confusion matrix of the best model per scenario (1,2,3)

DISCUSSION

5.1. Which solution should be chosen for attack prediction, binary or multiclass classification?

The choice of the best solution depends on the expected result, looking at elements such as prediction speed, model maintenance, inclusion of new attacks, and complexity of the problem, as described in the table below:

Table 7 Classification attack prediction

Elements	Binary Classification (Normal vs Attack)	Multiclass Classification (Normal, DoS, Probe, scanning, etc.)
Complexity	Simpler	More complex
Prediction speed	Faster (Lighter model)	A little slower (more calculations)
Explanation for Analysts	Detects if something is an attack, but doesn't say what type	Tell exactly what type of attack
Automatic action (IPS)	Can block all suspicious traffic at once	Can apply different blocks (e.g.: light warning for probe, strong block for DoS)
Model Maintenance	Simpler (fewer classes to train)	More delicate (requires class balancing)
Detection of new attacks	Suitable for general flagging	Better for understanding the nature of the attack and prioritizing response

Future work

- Explore other datasets and new attack types;
- Explore other techniques for class balancing.
- Explore other decision-tree based models, such as CatBoost and LightGBM.

5.2. Application of the machine learning model in IDS (Intrusion Detection System) or IPS (Intrusion Prevention System).

Table 8 Machine Learning Application

STEPS	IDS	IPS
1. Collect network data	Captures packets (PCAPs) or traffic records (NetFlow, logs, etc.)	Captures packets in real time
2. Preprocess the data	Extracts features such as source/destination IP, ports, packet size, protocol, TCP flags, and time between packets.	Same as IDS, but faster and simpler
3. Train a model	Uses supervised algorithms (e.g., XGBoost, Random Forest, Neural Networks) with "normal" or "attack" labels.	Same, but optimized for low latency.
4. Integrate the model into the system	IDS analyzes mirrored traffic (a copy of traffic) and generates alerts if it detects an attack.	IPS applies the real-time model: if it detects an attack, it blocks the traffic automatically.
5. Actions taken	Generates an alert (admin decides action).	Drops packets, disconnects sessions, and updates dynamic rules.
6. Update the model	Periodic retraining to capture new attack patterns.	Same, but with more care to avoid generating false positives that block legitimate traffic.

References

- [1]. José Arnaud, Paulo Matos “Artificial Intelligence as a Support Tool to Cybersecurity Activities”. ARIS2 – Advanced Research on Information Systems Security, vol. 2, n.º 1, pp. 2–12. 2022. [Online]. Available at: https://www.academia.edu/105506701/Artificial_Intelligence_as_a_Support_Tool_to_Cybersecurity_Activities?sm=b
- [2]. Dr. Nimisha. M.N, “ARTIFICIAL INTELLIGENCE IN CYBERSECURITY: A STUDY”. International Journal of Research Culture Society, vol. 7, n.º 10, pp. 69–72.2023. [Online]. Available at: https://www.academia.edu/111932020/ARTIFICIAL_INTELLIGENCE_IN_CYBERSECURITY_A_STUDY?sm=b
- [3]. Kandala kalyana Srinivas, D. Vijay Sai, , N.Saketh, I. Neelima, B.Alekhyia, “Artificial Intelligence Techniques for Prevention of Cyber Attacks and Detection of Security Threats”. International Journal of Engineering Research and Applications (IJERA), vol. 12, n.º 6, série V, pp. 37–44. 2022. [Online]. Available at: https://www.academia.edu/82300646/Artificial_Intelligence_Techniques_for_Prevention_of_Cyber_Attacks_and_Detection_of_Security_Threats?sm=b
- [4]. Micah Musser and Ashton Garriott, “Machine Learning and Cybersecurity”. Center for Security and Emerging Technology (CSET), Georgetown University. 2021. [Online]. Available at: https://www.academia.edu/118045038/Machine_Learning_and_Cybersecurity?sm=b
- [5]. Amol Dhondse and Sachchidanand Singh, “Redefining Cybersecurity with AI and Machine Learning”. Asian Society for Scientific Research; Asian Journal for Convergence in Technology, vol. 5, n.º 2. 2019. [Online]. Available at: https://www.academia.edu/68985017/Redefining_Cybersecurity_with_AI_and_Machine_Learning
- [6]. Bhavani Thuraisingham, “Artificial Intelligence and Data Science Governance: Roles and Responsibilities at the C-Level and the Board”. IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI). 2020 Download on February 9, 2024 [Online]. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9191400>
- [7]. Chao Zhang, “Data Security in Enterprise Management Under Big Data Technology”. 2nd International Conference on Information Science and Education (ICISE-IE). 2021. Download on February 8, 2024 [Online]. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9742543>
- [8]. Dan Lo, Hossain Shahriar, kai Qian, Michael Whitman, Fan Wu, “Colab Cloud Based Portable and Shareable Hands-on Labware for Machine Learning to Cybersecurity”. IEEE International Conference on Big Data (Big Data). 2021. Download on February 8, 2024 [Online]. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9671541>
- [9]. Veneta Yosifova, “Compositional Machine Learning Method for Computer Security Incidents Response Taxonomy Classification”. International Conference Automatics and Informatics (ICAI). 2022. Download on February 8, 2024 [Online]. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9960124>
- [10]. Tamara AlMasri, Mohammad Abu Snober, Qasem Abu Al-Haija, “IDPS-SDN-ML: An Intrusion Detection and Prevention System Using Software-Defined Networks and Machine Learning”. st International Conference on Smart Technology, Applied Informatics, and Engineering (APICS), pp. 133–137. 2022. Download on February 9, 2024 [Online]. Available at: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9918804>
- [11]. Nuno Mateus-Coelho, Maria Cruz-Cunha, “Exploring Cyber Criminals and Data Privacy Measures”, IGI Global, 2023. ISBN: 9781668484241.
- [12]. Nuno Ricardo Mateus-Coelho et al., “POSMASWEB: Paranoid Operating System Methodology for Anonymous and Secure Web Browsing”, in Handbook of Research on Cyber Crime and Information Privacy, Maria Manuela Cruz-Cunha and Nuno Mateus-Coelho (eds.), IGI Global Scientific Publishing, 2021, pp. 466–497. DOI: 10.4018/978-1-7998-5728-0.ch023.
- [13]. Nuno Mateus Coelho, Miguel Peixoto, Maria Manuela Cruz-Cunha, “Prototype of a paranoid mobile operating system distribution”, in 2019 7th International Symposium on Digital Forensics and Security (ISDFS), 2019, pp. 1–6. DOI: 10.1109/ISDFS.2019.8757551.
- [14]. Pedro Almeida Perdigão, Nuno Mateus Coelho, José Cascais Brás, “AI-Driven Threats in Social Learning Environments - A Multivocal Literature Review”, ARIS2 - Advanced Research on Information Systems Security, vol. 5, no. 1, 2025, pp. 4–37. DOI: 10.56394/aris2.v5i1.60.