Métodos de Aprendizado de Máquina Adversariais na Detecção de Anomalias em Redes de Computadores

Luiz Felipe de Camargo[1], Carlos Reis[1], Pedro Henrique Paiola[1], João Paulo Papa[1], José Remo F. Brega[1], Kelton A. P. da Costa[1]

[1]Universidade Estadual Paulista "Júlio de Mesquita Filho" (Unesp)
Bauru, São Paulo, Brasil
{luiz.felipe, cj.reis, pedro.paiola
joao.papa, remo.brega, kelton.costa}@unesp.br

Abstract. *With the use of intrusion detection systems and the need to improve their classifications and performance, the present work aims to study classifiers using machine learning and, in particular, adversarial methods. This study's main contribution is the validation of the benefit of Adversarial Machine Learning Methods in the detection of anomalies in computer networks. For this, experiments with several classifiers are carried out on a set of data with different attacks through experiments with altered samples to observe the classifiers' behavior. We can observe that the results were promising, consistent with other similar studies, indicating the best classifier and the best metric for each type of attack, the best metric for evaluating the results, and the most relevant parameters for correcting the labeled classifications incorrectly. Therefore, the adversarial methods, based on modified samples to correct erroneous classifications, may be adequate to improve classification methods based on artificial intelligence. The F1-Score metric achieved for each attack category was 0.99 for DoS, 0.99 for Probe, 0.95 for R2L, and 0.65 for U2R, considering the NSL-KDD data set.*

Resumo. *Com a utilização de sistemas de detecção de intrusão e a necessidade de aprimorar suas classificações e desempenho, o presente trabalho objetiva-se no estudo de classificadores utilizando aprendizado de máquina e, em especial métodos adversariais. A principal contribuição deste estudo é a validação do benefício de Métodos de Aprendizado de Máquina Adversariais na detecção de anomalias em redes de computadores. Para tanto, procede-se a realização de experimentos com diversos classificadores em um conjunto de dados com diferentes ataques através de experimentos com amostras alteradas no intuito de observar o comportamento dos classificadores. Desse modo, observa-se que os resultados foram promissores, condizentes com demais trabalhos semelhantes, indicando o melhor classificador e a melhor métrica para cada tipo de ataque bem como a melhor métrica para avaliação dos resultados e os parâmetros mais relevantes para correção das classificações rotuladas incorretamente. Conclui-se que os métodos adversariais, baseados em amostras alteradas para corrigir classificações equivocadas, podem ser adequados no aprimoramento de métodos de classificação baseados em inteligência artificial. A métrica F1-Score alcançada para cada categoria de ataque foi 0,99 para DoS, 0,99 para Probe, 0,95 para R2L e 0,65 para U2R, considerando o conjunto de dados NSL-KDD.*

1. Introdução

Com o advento dos microcomputadores e da computação pessoal, a transmissão de informações através das redes se tornou um importante meio de comunicação, sendo hoje a base de quase todos os demais meios de comunicação como, por exemplo, a telefonia celular 4G, telefonia fixa VoIP e IPTV [Tanenbaum and Wetherall 2021].

Contudo, toda essa movimentação de informações necessita ser realizada de forma segura e implementando as devidas proteções nos sistemas computacionais e nas redes de computadores. Uma ferramenta importante de proteção desses sistemas são os chamados Sistemas de Detecção de Intrusão, em inglês, *Intrusion Detection Systems* (IDS). Existem diversas abordagens de IDS, em que uma das mais utilizadas é a por Assinatura, onde o sistema realiza buscas por padrões de invasão baseando-se em dados passados [Wenke Lee et al. 1999].

A quantidade crescente de dados provenientes do crescimento constante das redes gera uma necessidade de um alto poder computacional para execução de um IDS. Além disso, a variedade de tipos e técnicas de ataques cresce a cada dia, exigindo a capacidade de adaptação de um IDS. Dessa forma, se torna necessário o uso de novos métodos de análise de dados para auxiliar a detecção de intrusos. Dentre eles, podem ser elencadas técnicas de aprendizado de máquina, as quais auxiliam a busca por padrões nas grandes quantidades de dados disponíveis, localizando mais rapidamente comportamentos que caracterizam incidentes de segurança e ataques [Buczak and Guven 2016].

Atualmente, tem-se a Inteligência Artificial como base para os mais diversos tipos de sistemas, com os mais diversos objetivos. Entretanto, a classificação realizada por esse tipo de ferramenta pode ser incorreta, gerando problemas diversos. Se por um lado existe a preocupação do desenvolvedor destas ferramentas em aprimorar sua acurácia, existem também indivíduos que buscam atrapalhar as classificações realizadas, causando diversos problemas.

Uma das abordagens mais utilizadas para geração desse tipo de problema é a criação de amostras adversariais, isto é, de amostras alteradas propositalmente para interferir no processo de classificação, gerando resultados incorretos. Porém, esse tipo de abordagem pode ser utilizada também de forma positiva, proporcionando um aprimoramento das classificações. Um desenvolvedor pode, por exemplo, gerar amostras adversariais e as utilizar em seu modelo de forma proposital, detectando pontos de falha e de maior sensibilidade no sistema de classificação [Marino et al. 2018].

O presente trabalho propõe a utilização de diversos classificadores para identificar diferentes categorias de ataques em um conjunto de dados relacionados à segurança de redes, a NSL-KDD. Objetiva-se, através da abordagem de aprendizado de máquina adversarial, obter um grau adequado de eficácia para os métodos de classificação baseados em inteligência artificial, possibilitando o ajuste dos mesmos para um melhor resultado. Como contribuição principal, pode-se citar a validação de técnicas de aprendizado de máquina adversariais na detecção de anomalias em redes de computadores, obtendo-se, assim, mais detalhes sobre as características que mais impactam nas classificações realizadas de forma incorreta.

Este trabalho conta com as seguintes partes: a Seção 2 apresenta a fundamentação teórica para realização do estudo, a Seção 3 apresenta os trabalhos relacionados e a Seção

4 expõe a metodologia utilizada e os experimentos realizados. Já a Seção 5 apresenta e discute os resultados dos experimentos e a Seção 6 apresenta as conclusões e perspectivas futuras.

2. Fundamentação Teórica

A seguir, são detalhados conceitos que nortearam o desenvolvimento do trabalho apresentado neste artigo.

2.1. Detecção de Intrusos

Os sistemas de detecção de intrusos são compostos por conjuntos de regras e configuração que buscam, através da análise de pacotes de uma rede, emitir alertas sobre possíveis situações que caracterizam um ataque, ou seja, uma situação que coloca em risco seu funcionamento e a segurança de seus dados. Essa análise é realizada geralmente através da comparação com os dados de ataques previamente conhecidos.

Uma das abordagens mais utilizadas atualmente no contexto de IDS é a baseada em Inteligência Artificial. Por meio dela, buscam-se métodos análogos ao processo de aprendizagem humana para utilização em análise de dados, como reconhecimento de padrões e otimização, dentre outras atividades. Em se tratando da aplicação de métodos de aprendizado de máquina, existem diversas abordagens, resultando em diferentes classificadores que podem ser utilizados na classificação de dados de rede. Os classificadores selecionados para o presente estudo foram:

- *Decision Tree* ou Árvores de Decisão é um classificador muito utilizado em tarefas de classificação e regressão, é baseado em uma árvore onde cada nó interno, não terminal, chamado galho, representa um teste ou decisão sobre o item de dado considerado [Goebel and Gruenwald 1999].
- *Random Forest* ou Floresta Aleatória é um algoritmo de aprendizado de máquina baseado em um conjunto de árvores de decisão, que também podem ser utilizadas em tarefas de classificação e regressão.
- *Adaptive Boosting* ou AdaBoost utiliza uma Árvore de Decisão de apenas um nível em sua implementação, sendo considerado como um classificador fácil e simples de ser utilizado.
- *Convolutional Neural Network* ou Rede Neural Convolucional, também conhecida como CNN, é um tipo de rede neural que trabalha em uma ampla gama de problemas, porém é muito utilizada em classificação de imagens. Podem ser treinadas fazendo uso de um conjunto de dados que utiliza algoritmo de retropropagação de erro que envolve o cálculo de erros cometidos pelo modelo.

2.2. Aprendizado de Máquina Adversarial

O aprendizado de máquina adversarial consiste em uma metodologia que busca enganar os modelos de aprendizado, fornecendo informações enganosas que foram previamente alteradas. O objetivo é causar um mau funcionamento em um modelo de aprendizado de máquina. As amostras adversárias são criadas com o objetivo de alterar a saída de um modelo, executando pequenas modificações em uma amostra de referência, usualmente real. Tais amostras são geralmente utilizadas para detectar pontos cegos em algoritmos de aprendizado de máquina. Um exemplo da aplicação desta técnica é a inclusão de

palavras bem pontuadas e a alteração na grafia de palavras mal pontuadas em mensagens de spam com o objetivo de enganar filtros automatizados [Laskov and Lippmann 2010]. O aprendizado de máquina adversarial pode ser baseados em três estratégias:

- Evasão: em ataques baseados na estratégia de evasão, os atacantes buscam evitar a detecção de um classificador: spammers e hackers tentam escapar da detecção ofuscando o conteúdo de e-mails de spam e malware, por exemplo. Nessa abordagem, não há necessidade de alteração dos dados de treinamento.
- Envenenamento: muitos sistemas utilizam aprendizado de máquina durante sua execução como, por exemplo, os IDSs, em que o modelo de classificação pode ser aprimorado através de novos treinamentos realizados em paralelo com a operação. O objetivo da abordagem de envenenamento é contaminar os dados que serão utilizados nesse tipo de treinamento. O atacante pode manipular os dados de operação de forma a alterar o retreinamento, causando problemas ao classificador.
- Roubo de modelo ou extração de modelo: o atacante utiliza a abordagem adversarial para obter informações sobre o funcionamento de um modelo de caixa preta, de forma a conhecer os dados de treinamento para reconstruir o modelo. Essa abordagem gera problemas como o acesso não autorizado a dados sensíveis e confidenciais, bem como as técnicas proprietárias.

Amostras adversariais são adequadas do ponto de vista do defensor, visto que podem ser utilizadas para realizar avaliação de vulnerabilidade, estudar a robustez contra interferências, melhorar a generalização e depurar o modelo de aprendizagem de máquina [Marino et al. 2018].

3. Trabalhos Relacionados

O trabalho de [Silva et al. 2019] realiza uma comparação de métodos baseados em aprendizagem de máquina em diferentes bases de dados, verificando a manutenção da acurácia e diminuição do custo computacional. Também buscando uma comparação de diferentes conjuntos de dados e utilizando diferentes algoritmos de aprendizagem de máquina, pode-se citar o trabalho desenvolvido por [Sapre et al. 2019], os quais buscaram uma comparação robusta entre os conjuntos de dados KDDCup99 e o NSL-KDD, avaliando o desempenho de diversos classificadores de aprendizagem de máquina, incluindo nos experimentos um conjunto maior de métricas. A conclusão foi que o conjunto NSL-KDD possui uma maior qualidade nos dados.

Diversos trabalhos abordam a utilização de amostras adversariais interferindo em modelos de classificação baseados em Inteligência Artificial, demonstrando a sensibilidade de sistemas deste tipo a amostras alteradas de forma proposital. O trabalho de [Sharma et al. 2019] estuda amostras adversariais interferindo em sistemas de veículos conectados e autônomos que utilizam Aprendizado de Máquina para automatizar suas tarefas e tomar decisões. A interferência gerando problemas nesse tipo de situação pode causar graves acidentes. [Kuchipudi et al. 2020] trata no seu trabalho sobre o Aprendizado de Máquina Adversarial para filtros de *spam*, em que amostras alteradas interferem na classificação de mensagens como *spam* utilizando processamento de linguagem natural para evitar os filtros. As três técnicas apresentadas se mostram efetivas e podem ser utilizadas para aprimorar os mecanismos de classificação existentes.

[Usama et al. 2019] mostram como as técnicas de aprendizado de máquina são promissoras na classificação de tráfego de rede, mas que também podem ser vulneráveis a ameaças adversariais, principalmente modelos de aprendizado profundo sendo expostos a amostras cuidadosamente alteradas para gerar problemas. No trabalho de [Marino et al. 2018] é apresentada uma interface de IA explicável para diagnosticar IDSs baseados em dados. Apresenta-se uma metodologia para explicar as classificações incorretas feitas pelo modelo seguindo uma abordagem adversarial. Embora a aprendizagem de máquina adversarial seja geralmente utilizada para enganar o classificador, nesse trabalho ela é usada para gerar explicações, encontrando as modificações mínimas necessárias para classificar corretamente as amostras classificadas incorretamente.

A proposta do presente trabalho é utilizar diversas técnicas de aprendizado de máquina para classificar diferentes tipos de ataques presentes no conjunto de dados NSL-KDD e, a partir das amostras classificadas de forma incorreta, gerar amostras adversariais que sejam classificadas de forma correta, permitindo identificar as diferenças que causam a classificação incorreta de uma amostra.

4. Metodologia e Experimentos

A seguir é detalhada a metodologia utilizada no desenvolvimento dos experimentos que compõem este trabalho.

4.1. Seleção dos Classificadores

Para a realização dos experimentos, foi utilizada a base de dados NSL-KDD, que é uma evolução da KDD Cup 1999, bastante utilizada em diversos trabalhos na literatura. No conjunto de dados NSL-KDD, foram realizadas melhorias buscando resolver problemas apresentados no KDD Original. Abaixo podem ser observadas informações sobre cada tipo de ataque presente na base de dados empregada neste trabalho:

- DoS - Visa bloquear ou restringir recursos de um sistema ou serviço de rede para os seus usuários. Serviços Web são os mais atingidos por esse tipo de ataque;
- Probe - Ataque que busca identificar e coletar informações sobre vulnerabilidades de uma rede ou de computador que possam ser utilizados em um possível ataque futuro;
- R2L - É realizada a tentativa de obter o acesso a algum equipamento na rede por um usuário não autorizado remotamente;
- U2R - Para realizar esse tipo de ataque o intruso adquire acesso como um usuário de privilégio normal e então passa a explorar as vulnerabilidades para ganhar acesso de root ou Administrador.

A base de dados NSL-KDD é estruturada originalmente em oito arquivos divididos em treinamento e teste, conforme detalhado abaixo:

- KDDTrain+ .TXT - É o conjunto de dados de treinamento completo, incluindo rótulos de tipo de ataque e nível de dificuldade em formato CSV.
- KDDTrain+ .ARFF - É o conjunto de dados de treinamento completo em formato binário ARFF.
- KDDTrain+ _20Percent.TXT - É um subconjunto do KDDTrain+.TXT, possui 20% dos dados de treinamento;

- KDDTrain+ _20Percent.ARFF - É um subconjunto do KDDTrain+ .ARFF, possui 20% dos dados de treinamento;
- KDDTest+ .TXT - É o conjunto de dados de teste completo, incluindo rótulos de tipo de ataque e nível de dificuldade em formato CSV;
- KDDTest+ .ARFF - É o conjunto de dados de teste completo em formato binário ARFF;
- KDDTest- 21.TXT - Um subconjunto do arquivo KDDTest+.TXT que não inclui registros com nível de dificuldade de 21 até 21;
- KDDTest- 21.ARFF - Um subconjunto do arquivo KDDTest+.TXT que não inclui registros com nível de dificuldade de 21 até 21.

Para o desenvolvimento do trabalho, foram utilizados dois arquivos da base NSL KDD : KDDTrain + .TXT e KDDTest + .TXT. Foram necessários passos de adequação da base de dados para aplicação de classificadores com aprendizado de máquina. Tais etapas incluem normalização, tratamento de atributos categóricos utilizando codificação distribuída e segmentação da base de dados em quatro partes, considerando cada classe de ataque.

No ambiente desenvolvido para realização dos testes, foi utilizada a biblioteca Keras para implementação do classificador CNN e a biblioteca scikit-learn para os demais classificadores. A biblioteca scikit-learn não utiliza os núcleos gráficos especiais CUDA presentes no hardware utilizado para as simulações, gerando assim uma redução de desempenho em relação aos demais métodos.

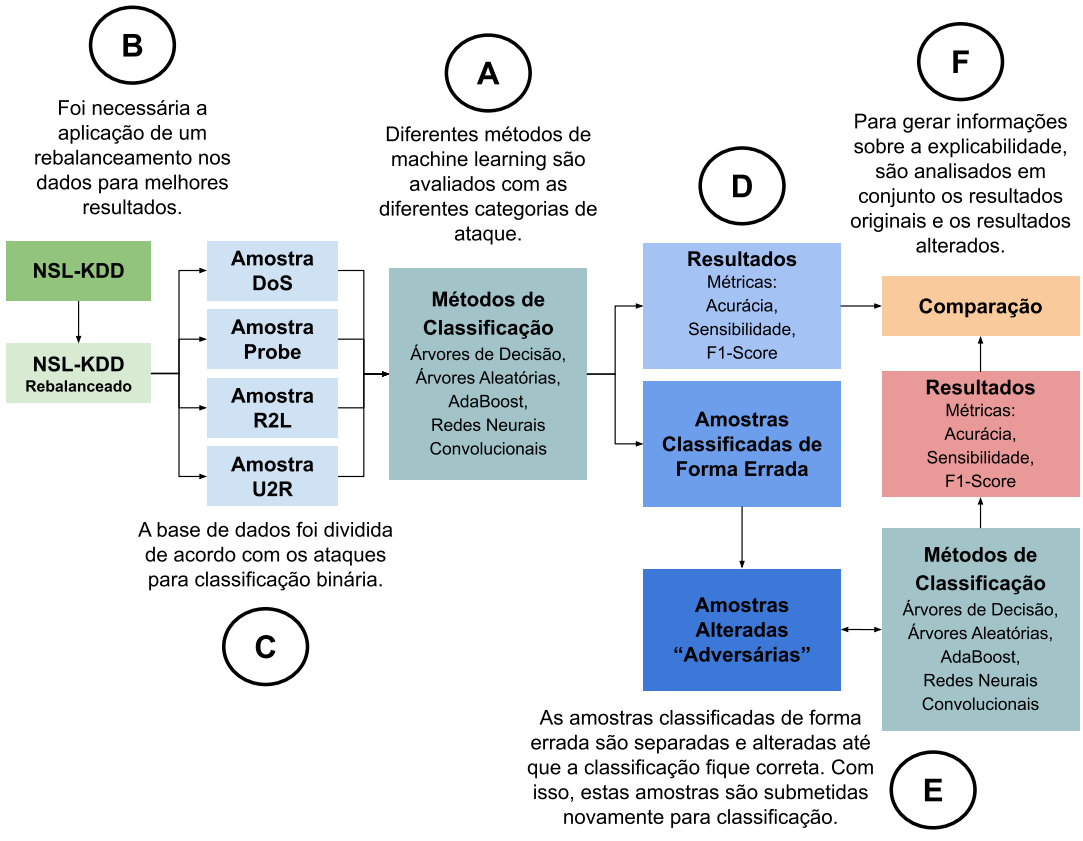

Figura 1. Etapas dos experimentos realizados.

Conforme apresenta a Figura 1, a primeira etapa do trabalho consiste na seleção dos modelos a serem utilizados para classificar as amostras e, posteriormente, serem usados na aprendizagem adversarial para identificar e entender quando e porquê estes modelos erram durante a classificação. Os modelos testados foram: Árvores de Decisão, Florestas Aleatórias, AdaBoost e Redes Neurais Convolucionais. O detalhe A na Figura 1 indica essa etapa. Os modelos utilizados mostraram-se eficazes para cada tipo de ataque. É importante ressaltar que os melhores resultados foram atingidos após o balanceamento realizado nas amostras. A etapa de balanceamento é indicada no detalhe B da Figura 1. Este balanceamento foi realizado na tentativa de melhorar o desempenho dos classificadores de ataques do tipo U2R e R2L. A base de dados utilizada apresenta mais amostras destes dois tipos de ataque no conjunto de teste do que no conjunto de treinamento, tanto em valores absolutos quanto relativos. Por este motivo, foram retiradas algumas amostras do conjunto de testes e inseridas no conjunto de treinamento.

Os dados de cada tipo de ataque foram separados para a classificação. Dessa forma, foram utilizados classificadores binários, que visam decidir se uma determinada amostra é um ataque ou não. Este processo é mais simples do que construir um classificador multiclasse. Essa separação dos dados é mostrada no detalhe C da Figura 1.

Para a avaliação dos classificadores foram utilizadas as medidas de acurácia, sensibilidade e F1-Score. Em especial, a medida F1-Score é mais relevante e considerada para esse estudo, pois une as informações da acurácia e da sensibilidade, sendo bastante útil quando se trabalha com bases de dados desbalanceadas, como neste caso. Os resultados são indicados no detalhe D da Figura 1. A Tabela 1 indica a matriz de confusão, em que se tem as categorias que compõem o resultado da classificação. Já na Tabela 2 é apresentado um detalhamento das métricas utilizadas, apresentando as fórmulas de acordo com as categorias presentes na matriz de confusão.

Tabela 1. Matriz de confusão.

		Detectada	
		Sim	Não
Real	Sim	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Tabela 2. Métricas de avaliação.

Métrica	Objetivo	Fórmula
Acurácia	Indica uma performance geral do modelo. Dentre todas as classificações, quantas o modelo classificou corretamente.	(VP + VN) / (VP + VN + FP + FN)
Precisão	Dentre todas as classificações de classe Positivo que o modelo fez, quantas estão corretas.	VP / (VP + FP)
Revocação (Recall, Sensibilidade)	Dentre todas as situações de classe Positivo como valor esperado, quantas estão corretas.	VP / (VP + FN)
F1-Score	Média harmônica entre precisão e recall.	2 * (Precisão * Revocação) / (Precisão + Revocação)

Os resultados dos testes não foram satisfatórios em um primeiro momento, em especial para os ataques R2L e U2R, utilizando a divisão padrão proposta pelo conjunto de dados, com 20% dos dados para treinamento e os 80% para testes. Os classificadores se adaptaram bem ao conjunto de treinamento, mas não apresentavam bons resultados para o conjunto de teste. Para tentar alcançar melhores resultados, foi realizada uma redistribuição dos dados, retirando 20% das amostras do conjunto de teste e as inserindo no conjunto de treinamento. Os resultados obtidos foram consideravelmente melhores, como pode ser visualizado na Tabela 3. O modelo de classificador selecionado foi Florestas Aleatórias, que demonstrou ser o melhor método para todas as categorias de ataque. A redistribuição dos dados necessária é indicada no detalhe B da Figura 1.

Tabela 3. Resultados antes e depois da redistribuição da base de dados.

Ataque	Melhor Método	Base de Dados Original - F1 Score	Melhor Método	Base de Dados Redistribuída - F1 Score
DoS	CNN	0.92	**Floresta Aleatórias**	0.99
Probe	Adaboost	0.80	**Floresta Aleatórias**	0.99
R2L	Adaboost	0.11	**Floresta Aleatórias**	0.95
U2L	Árvores de Decisão	0.50	**Floresta Aleatórias**	0.64

Na Tabela 4 pode-se observar de forma detalhada as matrizes de confusão para cada tipo de ataque. Na Figura 2 observa-se os resultados alcançados com as classificações realizadas em forma de gráfico, para os diferentes tipos de ataques.

Na Tabela 5 podem ser observados os valores de cada uma das medidas para a

Tabela 4. Matriz de confusão com os resultados.

Ataque DoS		Detectada	
		Sim	Não
Real	Sim	5916	28
	Não	9	7784
Ataque Probe		Detectada	
		Sim	Não
Real	Sim	1902	31
	Não	19	7754
Ataque R2L		Detectada	
		Sim	Não
Real	Sim	2184	149
	Não	90	7654
Ataque U2R		Detectada	
		Sim	Não
Real	Sim	22	25
	Não	0	7776

classificação do tipo de ataque no conjunto de dados redistribuído.

Tabela 5. Resultados antes e depois da redistribuição da base de dados.

Medida de Avaliação	Ataque DoS	Ataque Probre	Ataque R2L	Ataque U2R
Acurácia	0,99	0,99	0,97	0,99
Precisão	0,99	0,99	0,96	1,00
Revocação	0,99	0,98	0,94	0,47
F1-Score	0,99	0,99	0,95	0,64

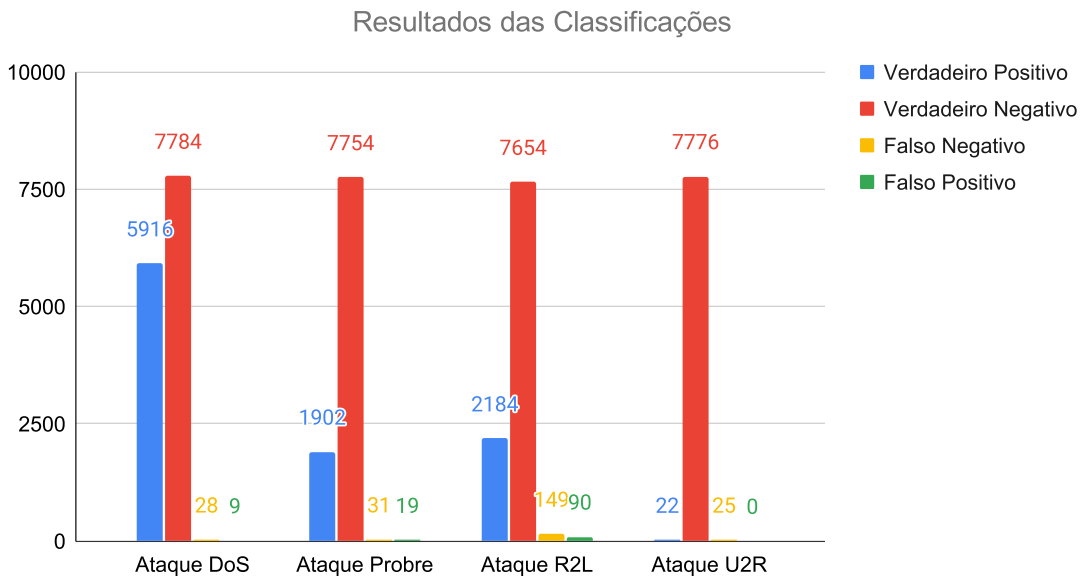

Figura 2. Resultados das Classificações.

4.2. Aplicação do Aprendizado de Máquina Adversarial

Com os classificadores treinados e os conjuntos de testes avaliados, as amostras classificadas incorretamente são separadas e utilizadas como entrada para a próxima etapa, indicada pelo detalhe E da Figura 1, que consiste na fase do aprendizado de máquina adversarial.

Esta etapa foca na correção da classificação destas amostras realizando o mínimo de modificações possíveis nas amostras originais. Dessa forma, ao avaliar a diferença entre as amostras originais e as amostras adversárias, deve-se observar quais características são mais relevantes para levar uma amostra a ser classificada incorretamente. Para isso, os experimentos realizados se basearam especialmente no trabalho de [Marino et al. 2018]. A principal diferença deste trabalho é que são apresentados os resultados para as quatro classes de ataques presentes no NSL-KDD, enquanto o trabalho citado apresentava apenas para ataques dos tipos DoS e Probe.

Na prática, o aprendizado adversarial aplicado neste trabalho consiste na resolução do seguinte problema de otimização:

$$\min_{\hat{x}} \quad H(\hat{y}, p(y|\hat{x}, w))\alpha I_{(\hat{x},\hat{y})} + ||x_0^{'} - \hat{x}^{'}||_2 \quad \text{s. a} \quad x_{min} \leq \hat{x} \leq x_{max} \tag{1}$$

em que:

- (x_0, y) é uma amostra de referência do conjunto de classificações incorretas;
- $\hat{x}$ é amostra adversária, gerada a partir de modificações da amostra e resultante da resolução do problema de otimização;
- $\hat{y}$ é a classe-alvo;

- $H(\hat{y}, p(y|\hat{x}, w))\alpha I_{(\hat{x},\hat{y})}$ é a entropia cruzada entre a classe alvo $\hat{y}$ e a classe estimada por $p(y|\hat{x}, w)$ para a amostra adversária $\hat{x}$ e os parâmetros w do classificador;
- $I_{(\hat{x},\hat{y})}$ indica se a classe alvo já foi atingida, retorna 0 caso a classificação esteja correta e 1 caso contrário;
- α é um fator de escala que define o peso da contribuição da entropia cruzada H para a função objetivo;
- $||x_0^{'} - \hat{x}^{'}||_2$ é a distância euclidiana entre as amostras original $x_0^{'}$ e a amostra adversária $\hat{x}^{'}$ normalizadas, de forma que todos as features possuam a mesma escala;
- x_{min} e x_{max} restringem o domínio das amostras.

Para resolver este problema de otimização, e assim obter as amostras adversárias $\hat{x}$, foi utilizado o Método de Powell, implementado pelo módulo de otimização da biblioteca SciPy para Python. O Método de Powell foi escolhido, pois ele é destinado para otimização com restrições e não necessita que a função objetivo seja diferenciável.

As amostras adversárias obtidas são classificadas corretamente pelo modelo utilizado, e, idealmente, com os atributos pouco alterados em relação às amostras originais. Desta forma, ao comparar quais atributos foram alterados têm-se grandes pistas sobre o porquê destas amostras terem sido classificadas incorretamente, assim como nos permite entender melhor o que o classificador está considerando para predizer a classe de cada amostra.

A comparação das amostras originais com as adversárias consistiu na última etapa, destaque F na Figura 1, em que através da geração de gráficos e interpretação destes, buscou-se comparar os resultados dos dados originais classificados e dos dados alterados. A partir das informações de explicabilidade extraídas nessa etapa, foram consolidados os dados a serem apresentados na próxima seção.

5. Resultados e discussão

Uma vez obtidos os resultados das classificações com os dados originais e com os dados alterados com os métodos de aprendizagem adversarial, foram realizadas diversas comparações entre esses resultados por meio de técnicas de exploração de dados e visualização da informação.

Uma técnica empregada nos dados foi a t-SNE, em que é possível gerar uma visualização com duas dimensões a partir de um conjunto de dados que possui n¿2 dimensões. Após a geração das visualizações dos dados, antes e depois das alterações, as duas situações foram plotadas no mesmo gráfico, sendo possível uma comparação demonstrando as alterações que foram necessárias através da distância entre os dados originais e os dados alterados. Na Figura 3 são exibidas as amostras originais, em diferentes cores para diferentes categorias de ataques, e na Figura 4 são exibidas em vermelho as amostras alteradas, sobrepostas às amostras originais.

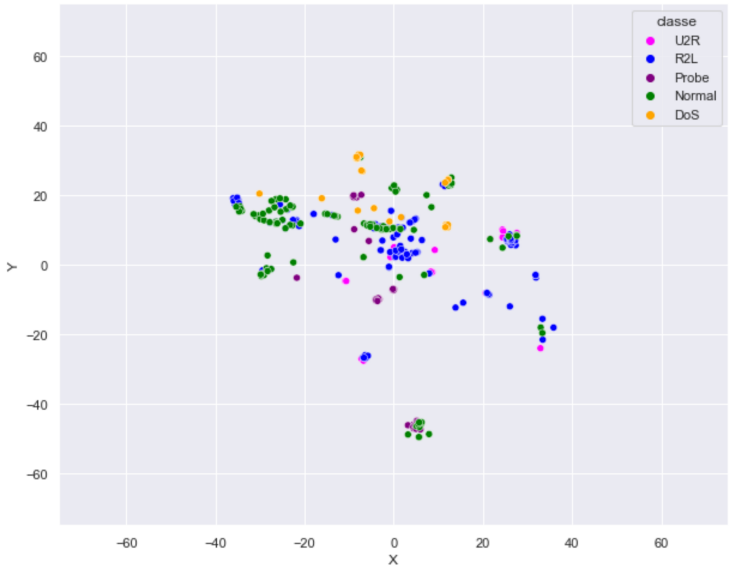

Figura 3. Amostras originais exibidas com a técnica t-SNE.

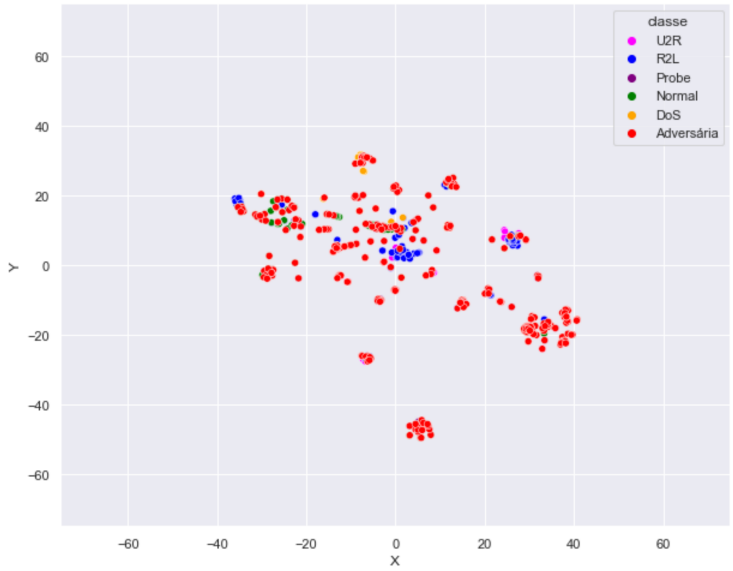

Figura 4. Amostras originais e alteradas exibidas com a técnica t-SNE, em que os pontos vermelhos representam as amostras alteradas.

Para cada categoria de classificação incorreta, foi identificado a característica que mais foi alterada nas amostras adversárias, para que as amostras fossem classificadas de forma correta. Na Tabela 6 podem ser observados os tipos de amostras e as informações das características mais relevantes.

Tabela 6. Características mais relevantes para cada categoria de amostra de acordo com a incidência.

Tipos de amostras	Número de amostras	Característica mais vezes alterada	Incidência do parâmetro
DoS detectadas como Normais	28	wrong_fragment	13
Normais detectadas como DoS	9	duration	4
Probe detectadas como Normais	31	duration	8
Normais detectadas como Probe	19	logged_in	9
U2R detectadas como Normais	25	num_shells	12
Normais detectadas como U2R	sem amostras classificadas de forma errada		
R2L detectadas como Normais	149	num_failed_logins	102
Normais detectadas como R2L	90	duration	90

Na Tabela 7 podem ser observadas brevemente as definições de cada característica encontrada, indicando o que representa a variável em uma conexão de rede.

Tabela 7. Definição das características encontradas.

Característica	Definição	Tipo de variável
wrong_fragment	Número de fragmentos ”danificados”na conexão	Contínua
duration	Duração (número em segundos) da conexão	Contínua
logged_in	Valor ”1”para conexões onde o login acontece com sucesso, ”0”para outras situações	Discreta
num_shells	Número de prompts de shell na conexão	Contínua
num_failed_logins	Número de tentativas de login que falharam na conexão	Contínua

A partir das informações extraídas nas explorações de dados, têm-se aspectos de explicabilidade, que permitem compreender de forma mais detalhada o funcionamento e a atuação dos classificadores baseados em aprendizado de máquina. Dessa forma, a situação onde se tem o aprendizado de máquina como uma caixa preta é amenizada, contando com informações mais detalhadas sobre o processo de classificação e as características das conexões de rede que mais impactam nessa classificação.

6. Conclusões e Trabalhos Futuros

Com base no que foi apresentado neste trabalho, pode-se concluir que a utilização de técnicas de classificação baseadas em aprendizado de máquina em conjunto com a abordagem adversarial, onde amostras são alteradas para auxiliar na compreensão do funcionamento dos classificadores, auxiliam na evolução dos IDSs, demonstrando os pontos que devem ser aprimorados para aumentar as taxas de sucesso dos classificadores.

Após diversas simulações, com todo o processo de exploração de dados realizado, os resultados obtidos se mostram válidos e condizentes com outros estudos realizados na área, indicando que a principal contribuição proposta pelo trabalho foi validada, demonstrando a utilidade dos métodos de aprendizado de máquina adversários na detecção de anomalias em redes de computadores. Desta forma, espera-se que o presente estudo sirva como base para pesquisadores que desejem dar continuidade nas simulações e na exploração de dados utilizando abordagens semelhantes.

Referências

Buczak, A. L. and Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys Tutorials*, 18(2):1153–1176.

Goebel, M. and Gruenwald, L. (1999). A survey of data mining and knowledge discovery software tools. *SIGKDD Explor. Newsl.*, 1(1):20–33.

Kuchipudi, B., Nannapaneni, R. T., and Liao, Q. (2020). Adversarial machine learning for spam filters. In *ACM International Conference Proceeding Series*, pages 1–6, New York, NY, USA. Association for Computing Machinery.

Laskov, P. and Lippmann, R. (2010). Machine learning in adversarial environments.

Marino, D. L., Wickramasinghe, C. S., and Manic, M. (2018). An adversarial approach for explainable ai in intrusion detection systems. In *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*, pages 3237–3243.

Sapre, S., Ahmadi, P., and Islam, K. (2019). A Robust Comparison of the KDDCup99 and NSL-KDD Intrusion Detection Datasets by Utilizing Principle Component Analysis and Evaluating the Performance of Various Machine Learning Algorithms:. *Journal of Student-Scientists' Research*, 1.

Sharma, P., Austin, D., and Liu, H. (2019). Attacks on Machine Learning: Adversarial Examples in Connected and Autonomous Vehicles. In *2019 IEEE International Symposium on Technologies for Homeland Security, HST 2019*. Institute of Electrical and Electronics Engineers Inc.

Silva, L., Silva, A., Filho, A. F., and Filho, A. B. (2019). Estudo comparativo de métodos de aprendizagem de máquina aplicados em sistemas de detecção de intrusão. In *Anais da VII Escola Regional de Computação do Ceará, Maranhão e Piauí*, pages 135–142, Porto Alegre, RS, Brasil. SBC.

Tanenbaum, A. S. and Wetherall, D. J. (2021). *Computer Networks, Global Edition*. Pearson Education, 6th edition.

Usama, M., Qayyum, A., Qadir, J., and Al-Fuqaha, A. (2019). Black-box adversarial machine learning attack on network traffic classification. In *2019 15th International Wireless Communications and Mobile Computing Conference, IWCMC 2019*, pages 84–89. Institute of Electrical and Electronics Engineers Inc.

Wenke Lee, Stolfo, S. J., and Mok, K. W. (1999). A data mining framework for building intrusion detection models. In *Proceedings of the 1999 IEEE Symposium on Security and Privacy (Cat. No.99CB36344)*, pages 120–132.