# Investigando o Impacto de Amostras Adversárias na Detecção de Intrusões em um Sistema Ciberfísico *

**Gabriel H. N. Espindola da Silva**[1], **Rodrigo Sanches Miani**[2], **e Bruno Bogaz Zarpelão**[1]

[1]Departamento de Computação – Universidade Estadual de Londrina (UEL)
Rod. Celso Garcia Cid, s/n – 86.057-970 – Londrina – PR – Brasil

[2]Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Uberlândia – MG – Brasil

{gabriel.henrique1,brunozarpelao}@uel.br, miani@ufu.br

***Abstract.*** *In this paper, we investigate the impact that adversarial examples have on machine learning algorithms used to detect intrusions in cyber-physical systems. The study considers a scenario where an attacker manages to get access to data from the target that can be used to train the adversarial model. The attacker aims to generate malicious samples using Adversarial Machine Learning to mislead the machine learning model used for intrusion detection. By running FGSM (Fast Gradient Sign Method) and JSMA (Jacobian Saliency Map Attack) attacks, we observed that prior knowledge on the target algorithm architecture can lead to more critical attacks. Also, the results showed that variations in the amount of data accessed for attack preparation can have different impacts in the experimented algorithms. Finally, FGSM could create more severe attacks than JSMA, but the latter is less intrusive and possibly harder to detect.*

***Resumo.*** *Neste artigo, investigamos o impacto que amostras adversárias causam em algoritmos de aprendizado de máquina supervisionado utilizados para detectar ataques em um sistema ciberfísico. O estudo leva em consideração o cenário onde um atacante consegue obter acesso a dados do sistema alvo que podem ser utilizados para o treinamento do modelo adversário. O objetivo do atacante é gerar amostras maliciosas utilizando aprendizado de máquina adversário para enganar os modelos implementados para detecção de intrusão. Foi observado através dos ataques FGSM (Fast Gradient Sign Method) e JSMA (Jacobian Saliency Map Attack) que o conhecimento prévio da arquitetura do algoritmo alvo pode levar a ataques mais severos, e que os algoritmos alvo testados sofrem diferentes impactos conforme se varia o volume de dados roubados pelo atacante. Por fim, o método FGSM produziu ataques com maior severidade média que o JSMA, mas o JSMA apresenta a vantagem de ser menos invasivo e, possivelmente, mais difícil de ser detectado.*

## 1. Introdução

O avanço em diferentes tecnologias relacionadas a sensores, protocolos de comunicação, e armazenamento e processamento de grandes volumes de dados permitiu que diversos

*O autor Gabriel H. N. Espindola da Silva agradece o apoio financeiro da Fundação Araucária de Apoio ao Desenvolvimento Científico e Tecnológico do Estado do Paraná para a execução deste trabalho.

sistemas embarcados fossem conectados a redes TCP/IP e, sobretudo, à Internet. Surgiu assim uma nova geração de sistemas ciberfísicos, que preveem a utilização massiva de tecnologia da informação para modernizar infraestruturas tradicionais como as redes de energia elétrica. Ironicamente, a mesma tecnologia que pode tornar essas infraestruturas mais eficientes e robustas traz também novos riscos. É importante lembrar que o acesso remoto aos sistemas ciberfísicos, por exemplo, é bastante facilitado por estas soluções. Por isso, um problema frequentemente discutido é a possibilidade de ciberataques explorarem vulnerabilidades nessas novas soluções e trazer graves prejuízos [Kim et al. 2022, Ning and Jiang 2022].

Pesquisadores já têm trabalhado em diversos estudos para proporcionar melhores condições de segurança a esses sistemas. Certamente, os sistemas de detecção de intrusão (IDS - *Intrusion Detection System*) estão entre os mecanismos de defesa mais explorados nesses trabalhos. Eles possuem a capacidade de analisar informações sobre o sistema monitorado em tempo real e detectar sinais de comportamento malicioso. Dessa forma, podem alertar os administradores do sistema para que tomem as medidas cabíveis. Assim como tem ocorrido em outras áreas, as técnicas de aprendizado de máquina têm dominado o horizonte das novas propostas de IDS [Zhang et al. 2022, Zarpelão et al. 2020].

Contudo, mais uma vez, a inserção de uma nova tecnologia pode trazer mais riscos à cibersegurança. Diversos trabalhos têm mostrado consistentemente que algoritmos de aprendizado de máquina podem ser alvos de ataques que prejudicam a sua capacidade preditiva por meio de técnicas de aprendizado de máquina adversário. Estes ataques podem explorar diferentes níveis de acesso às informações sobre o alvo. Nos ataques do tipo caixa preta, o atacante possui pouca ou nenhuma informação sobre o sistema alvo. Nos ataques denominados caixa cinza, o atacante eleva um pouco o seu nível de conhecimento. Ele pode conseguir acessar alguns dados utilizados para treinamento do modelo alvo ou detalhes sobre o algoritmo e os seus hiperparâmetros. Por fim, os ataques caixa branca são aqueles em que o atacante conhece todos os detalhes do modelo alvo. As técnicas utilizadas para realizar o ataque a partir das informações obtidas também varia. Entre elas, uma das mais discutidas consiste em manipular maliciosamente as amostras apresentadas ao modelo para induzi-lo ao erro nas predições. Essas amostras são denominadas amostras adversárias [Alhajjar et al. 2020, Tabassi et al. 2019].

Esses ataques começaram a ser estudados na área de visão computacional. Eles eram utilizados, principalmente, para gerar pequenas alterações em imagens que seriam imperceptíveis a olho nu, mas poderiam induzir os modelos de aprendizado de máquina ao erro. Com o tempo, este conhecimento foi transportado para outras áreas que utilizam o aprendizado de máquina, chegando aos IDSs [Anthi et al. 2021, Wang 2018, Pawlicki et al. 2020, Ibitoye et al. 2019]. IDSs que protegem sistemas ciberfísicos são alvos potenciais desses ataques. Muitos deles utilizam dados do nível de aplicação para detectar a ocorrência de intrusões, o que pode facilitar que amostras adversárias passem despercebidas. IDSs propostos para defender redes elétricas inteligentes, por exemplo, podem utilizar dados sobre grandezas elétricas medidas na rede para detectar atividades maliciosas [Zarpelão et al. 2020]. Portanto, variações sutis, de difícil detecção, podem ser artificialmente inseridas em dados como tensão, corrente e frequência do sinal elétrico para produzir amostras adversárias capazes de enganar IDSs baseados em aprendizado de máquina.

O objetivo deste trabalho é investigar o impacto de amostras adversárias produzidas por meio das técnicas FGSM (*Fast Gradient Sign Method*) e JSMA (*Jacobian Sa-*

*liency Map Attack*) em modelos de detecção de intrusão baseados em aprendizado de máquina supervisionado. O FGSM está entre as técnicas mais populares que exploram o uso de métodos baseados em gradiente. O JSMA representa uma variante mais direcionada desse tipo de ataque, pois escolhe manipular determinados atributos da amostra que podem potencializar os efeitos do ataque. Para implementar os modelos de detecção de intrusão, foram utilizados os algoritmos *Random Forest*, Árvore de Decisão J48 e MLP (*Multi Layer Perceptron*). Considera-se também que o IDS alvo utiliza dados sobre o funcionamento da rede elétrica coletados por PMUs (*Phasor Measurement Unit*), ou seja, são grandezas elétricas contínuas que seriam interceptadas e manipuladas pelos atacantes para enganar o sistema de detecção. Os experimentos trabalham cenários do tipo caixa cinza. Mais especificamente, o trabalho buscou investigar três questões:

- Entre o FGSM e o JSMA, qual causa mais impacto negativo ao desempenho preditivo do modelo de detecção de intrusão?
- A variação no volume de dados que o atacante consegue acessar sobre o alvo influencia no impacto das amostras adversárias resultantes?
- As amostras adversárias se tornam mais efetivas caso o modelo de aprendizado de máquina utilizado pelo atacante coincida com o modelo utilizado pelo alvo?

O restante deste artigo é organizado da seguinte forma. A Seção 2 traz os trabalhos relacionados sobre IDS e amostras adversárias. A Seção 3 apresenta detalhes sobre as duas técnicas de produção de amostras adversárias exploradas neste trabalho: FGSM e JSMA. Na Seção 4, discutimos os experimentos desenvolvidos para investigar o impacto desses ataques em um modelo de detecção de intrusão para uma rede elétrica. A Seção 5 apresenta os resultados dos experimentos, enquanto a Seção 6 encerra o artigo com as considerações finais.

## 2. Trabalhos Relacionados

A quantidade de estudos que abordam aprendizado de máquina adversário em IDSs tem crescido nos últimos anos. Essas pesquisas procuram principalmente entender como as amostras adversárias podem afetar a detecção de intrusão. Com este objetivo, eles seguem diferentes metodologias e exploram tipos de ataques e modelos de aprendizado de máquina variados.

Aiken e Scott-Hayward [2019] realizaram ataques contra o *Neptune*, um IDS para redes definidas por software. Esse IDS analisa fluxos de tráfego utilizando classificadores baseados em algoritmos como Regressão Logística, *Random Forest*, SVM (*Support Vector Machine*), e *k-Nearest Neighbors*. Três atributos de tráfego (comprimento da carga útil, taxa de pacotes, e presença de tráfego bidirecional) foram manipulados para fazer com que pacotes de um ataque de negação de serviço distribuído se parecessem com pacotes benignos. Os resultados mostraram que estes ataques de evasão podem ter sucesso contra diferentes classificadores desde que os atributos mais relevantes sejam manipulados de maneira eficaz.

Apruzzese et al. [2019] também se debruçaram sobre ataques que manipulam os atributos mais relevantes para a classificação. No trabalho deles, os atacantes adicionaram ruído a três atributos de fluxos de tráfego gerados por *bots* (bytes trocados durante a comunicação, duração do fluxo e número total de pacotes). Os algoritmos *Random Forest*, MLP e *k-Nearest Neighbors* foram escolhidos como alvos. Os resultados indicaram que os ataques poderiam causar sérios problemas à eficácia dos alvos, fazendo com que a taxa de detecção caísse para um terço do medido sem a presença de amostras adversárias. No

trabalho de [Apruzzese et al. 2019], os autores também implementaram alguns mecanismos de defesa como o *adversarial retraining*.

Outros trabalhos investigaram técnicas de ataques que foram originalmente criadas para a área de classificação de imagens. Ayub et al. [2020] se concentraram em amostras adversárias criadas com o uso da técnica JSMA. Em seu estudo, um IDS baseado em uma MLP era o alvo dos ataques em um cenário do tipo caixa branca. Os experimentos focaram apenas em modificar amostras maliciosas para que fossem classificadas erroneamente como benignas. Os resultados do trabalho apontaram que o JSMA foi eficaz em degradar o desempenho preditivo do modelo baseado em MLP.

Anthi et al. [2021] também trabalharam apenas com o JSMA. Contudo, diferentemente de [Ayub et al. 2020], Anthi et al. usaram o *Random Forest* e a Árvore de Decisão J48 para criar os modelos que seriam alvo das amostras adversárias. O objetivo do seu trabalho foi estudar um cenário caixa cinza, onde um modelo baseado em MLP foi utilizado para gerar as amostras adversárias. Nos experimentos desenvolvidos sobre um conjunto de dados obtido do monitoramento de uma rede elétrica, os autores descobriram que as amostras adversárias produzidas a partir de uma MLP podem afetar o desempenho de modelos baseados no *Random Forest* e no J48. Eles também mostraram que incluir amostras adversárias no conjunto de treinamento pode fazer os modelos resultantes mais resistentes a ataques de amostras adversárias.

Wang [2018], Pawlicki et al. [2020], e Ibitoye et al. [2019] estudaram o impacto de ataques baseados em FGSM e outras técnicas como DeepFool, C&W, BIM (*Basic Iterative Method*), e PGD (*Projected Gradient Descent*). Wang realizou ataques do tipo caixa branca usando as técnicas FGSM, JSMA, DeepFool, e C&W contra um modelo de detecção de intrusão baseado em uma MLP. De acordo com os resultados alcançados, C&W foi a técnica menos efetiva quando comparada com as demais. Além disso, a técnica JSMA foi reportada como a mais atrativa para os atacantes, já que ela diminui a quantidade de atributos manipulados.

De maneira semelhante, Pawlicki et al. [2020] exploraram as técnicas FGSM, BIM, e C&W, incluindo ainda o PGD a esta lista. No seu trabalho, o IDS alvo consiste de uma rede neural artificial treinada para classificar fluxos de tráfego como maliciosos ou benignos. Sua principal contribuição é um mecanismo de defesa que usa as ativações neurais do alvo durante a fase de inferência para classificar as amostras como benignas ou adversárias. Dois classificadores alcançaram os melhores resultados na implementação deste mecanismo de defesa: *Random Forest* e *Nearest Neighbor*.

Ibitoye et al. [2019] analisaram o impacto de amostras adversárias contra modelos baseados em SNN (*Self-normalizing Neural Network*) e FNN (*Feed-forward Neural Network*) para detecção de intrusão em redes de Internet das Coisas. As técnicas FGSM, BIM, e PGD foram utilizadas para criar as amostras adversárias. Os resultados mostraram que os ataques causaram impacto significativo sobre a eficácia dos dois modelos. Na comparação direta entre os dois modelos, o SNN se mostrou mais robusto que o FNN quando submetido aos ataques executados pelos autores do experimento.

Redes Adversárias Generativas (GAN - *Generative Adversarial Networks*) e ZOO são outras técnicas de ataques encontradas nos trabalhos levantados. Yang et al. [2018] buscaram entender melhor o impacto destas duas técnicas, além da C&W, contra redes neurais profundas aplicadas à detecção de intrusão. Os ataques foram todos realizados no modelo caixa preta. Os resultados indicaram que os ataques baseados em ZOO e GAN

causaram maior impacto que o C&W. Apesar desse bom resultado, as técnicas também apresentaram algumas limitações. De acordo com os autores, o GAN ainda é instável durante a fase de treinamento, enquanto o ZOO é computacionalmente custoso.

Todos os trabalhos revisados, com exceção de [Anthi et al. 2021], se concentraram em atacar modelos treinados para classificar tráfego de rede. As técnicas utilizadas para realizar os ataques, assim como os tipos de cenários, são bastante diversificados. O presente trabalho estuda o impacto de amostras adversárias contra um modelo de detecção que analisa, ao invés de tráfego de rede, medições coletadas por meio de PMUs em um sistema de energia elétrica. Em infraestruturas críticas, IDSs que utilizam dados de aplicação, como os coletados por PMUs, são comuns. Isso reforça a importância de estudar o impacto de amostras adversárias contra este tipo de sistema de detecção. Anthi et al. exploraram o mesmo cenário, mas este trabalho tem objetivos diferentes. Anthi et al. estudaram o impacto de ataques baseados em JSMA e a eficácia de mecanismos de defesa baseados em treinamento adversário. No presente artigo, nós incluímos o FGSM entre as técnicas, investigamos se a quantidade de dados de treinamento usados pelo atacante pode influenciar a eficácia dos ataques, e procuramos entender o quanto semelhanças entre o modelo alvo e o modelo do atacante afetam a força dos ataques. A Tabela 1 traz uma visão geral dos trabalhos e permite compará-los por meio de três critérios: classificadores utilizados para implementar os alvos dos ataques ("Alvos"), fontes de dados analisadas por estes classificadores para detectar as intrusões ("Fontes de dados do modelo alvo"), e ataques que foram realizados durantes os experimentos ("Ataques").

**Tabela 1. Aspectos principais dos trabalhos relacionados.**

| **Referência** | **Alvos** | **Fontes de dados do modelo alvo** | **Ataques** |
|---|---|---|---|
| Yang et al. [2018] | Rede Neural Profunda | Estatísticas do tráfego de rede | GAN, ZOO, C&W |
| Wang [2018] | MLP | Estatísticas do tráfego de rede | FGSM, JSMA, DeepFool, C&W |
| Aiken e Scott-Hayward [2019] | Regressão Logística, Random Forest, SVM, k-Nearest Neighbors | Estatísticas do tráfego de rede | Perturbação simples de atributos selecionados do tráfego |
| Apruzzese et al. [2019] | Random Forest, MLP, k-Nearest Neighbors | Estatísticas do tráfego de rede | Perturbação simples de atributos selecionados do tráfego |
| Ibitoye et al. [2019] | SNN, FNN | Estatísticas do tráfego de rede | FGSM, BIM, PGD |
| Ayub et al. [2020] | MLP | Estatísticas do tráfego de rede | JSMA |
| Pawlicki et al. [2020] | Rede Neural | Estatísticas do tráfego de rede | FGSM, BIM, PGD |
| Anthi et al. [2021] | Random Forest, Árvore de Decisão J48 | Medições em um sistema de energia elétrica | JSMA |
| Este trabalho | Random Forest, Árvore de Decisão J48, MLP | Medições em um sistema de energia elétrica | FGSM, JSMA |

## 3. Ataques FGSM e JSMA

O aprendizado de máquina adversário abrange uma série de técnicas que almejam enganar ou confundir modelos de aprendizado de máquina a fim de induzi-los a falhar em suas previsões e tomadas de decisão [Goodfellow et al. 2014]. Uma dessas técnicas trata de adicionar uma pertubação aos dados de entrada com o propósito de confundir o modelo de aprendizado. Porém, alterações bruscas e totalmente aleatórias podem fazer com que os ataques sejam facilmente detectados. A intenção é fazer uma pertubação mínima, mas suficiente, nos dados de entrada com a finalidade de gerar uma classificação incorreta.

Entre as diversas técnicas existentes de aprendizado de máquina adversário, o FGSM [Papernot et al. 2016a] é um ataque bastante conhecido. Sua intenção é adicionar um ruído mínimo aos dados de entrada tal que seja gerada uma classificação errada de alta confiança do modelo. Esse ataque explora o gradiente computado de uma função de custo $J(\theta, x, y)$, onde $\theta$ são os parâmetros do modelo (obtidos por uma estimativa dos dados), **x** é o dado de entrada original, e **y** é a saída real respectiva à entrada **x**.

O ataque, então, utiliza de *backpropagation* para obter $\nabla_x J(\theta, x, y)$. Existe, também, um valor $\epsilon$ que pode ser escolhido de maneira a aplicar uma extensão do ruído gerado nos dados de entrada na direção $sign(\nabla_x J(\theta, x, y))$. Assim, podemos obter $x^*$, que representa a amostra adversária e pode ser calculado conforme a equação (1).

$$x^* = x + \epsilon \cdot sign(\nabla_x J(\theta, x, y)) \tag{1}$$

O JSMA [Papernot et al. 2016b], semelhante em certos pontos ao FGSM, consiste de um ataque mais direcionado, que escolhe, com base em um mapa de saliência, quais são os atributos das entradas (*features*) com maior relevância na decisão de um modelo. Para calcular o mapa de saliência e observar quais são os atributos com maior chance de provocar uma classificação errada dos dados, o JSMA escolhe uma porcentagem $\theta$ inicial desses atributos e aplica um valor $\gamma$ de ruído. Esse processo é repetido várias vezes, até que um mapa de saliência seja construído e, a partir daí, utilizado para gerar as amostras adversárias [Anthi et al. 2021].

Os objetivos do JSMA e do FGSM são os mesmos: a partir de uma amostra $x$ e um ruído $\delta$, obter $x^* = x + \delta$, no qual $x^*$ corresponde à amostra adversária. Ainda, temos que ambos os ataques, normalmente, são aplicados usando um modelo de aprendizado criado por uma rede neural para gerar as amostras adversárias.

## 4. Materiais e Métodos

O objetivo deste trabalho, de maneira geral, é investigar o potencial e as particularidades de dois tipos de ataques que podem ser aplicados contra modelos treinados para detecção de intrusão: o JSMA e o FGSM. Estes dois ataques seguem a mesma sequência de passos. Primeiramente, o atacante precisa acessar dados pertencentes ao alvo para treinar o seu próprio modelo de aprendizado. Depois de obter esse modelo, o atacante passa a interceptar as amostras enviadas para classificação para o alvo. O atacante, utilizando o modelo que ele treinou, modifica a amostra e a reencaminha para o modelo alvo. O modelo alvo, ao tentar classificar a amostra maliciosamente modificada, acaba por ser induzido ao erro. A diferença entre os dois tipos de ataque é a maneira como a amostra é modificada, o que pode ser observado na Seção 3 deste trabalho. No experimento apresentado na seção corrente, o alvo será um modelo de detecção de intrusões que recebe como entrada amostras

coletadas em equipamentos de uma rede de energia elétrica e busca determinar se ela está sob ataque. O estudo foi pensado para responder as seguintes perguntas:

- Qual ataque causa um maior impacto no modelo alvo, FGSM ou JSMA?
- Se diminuirmos o volume de dados de treinamento roubados pelo atacante, o ataque mantém, aumenta ou diminui o seu impacto?
- Em um cenário onde o atacante conhece o algoritmo utilizado pelo alvo para criar o seu modelo de detecção, o ataque causa maior impacto ao sistema?

A seguir, serão apresentados maiores detalhes sobre o cenário de ataque, o cenário do modelo alvo e o conjunto de dados utilizado no experimento.

### 4.1. Cenário do Modelo Alvo

Na composição do nosso estudo, o primeiro ponto que definimos foi o modelo que será alvo dos ataques. Foi projetado um cenário simulado onde o atacante tem acesso aos dados de treinamento do modelo alvo. Em linhas gerais, o alvo se baseia em um modelo supervisionado que passa por uma fase de treinamento, onde é utilizado 60% do conjunto de dados que temos à disposição. Em seguida, os outros 40% do conjunto de dados são utilizados para a fase de inferência do modelo. A fase de inferência pode ser entendida, também, como a fase na qual o IDS classifica as amostras em benignas ou maliciosas, com base no conhecimento construído durante o treinamento.

No processo de treinamento e inferência, para verificarmos a eficácia dos modelos de detecção de intrusões, utilizamos algumas métricas de avaliação de classificadores. Essas métricas são construídas sobre uma matriz de confusão, que nos retorna as seguintes contagens: amostras maliciosas devidamente classificadas como tal (VP - verdadeiros positivos); amostras benignas corretamente classificadas como tal (VN - verdadeiros negativos); amostras begnignas erroneamente classificadas como maliciosas (FP - falsos positivos); e amostras maliciosas erroneamente classificadas como benignas (FN - falsos negativos).

A partir dessas informações, podemos obter métricas como *Recall*, *Precision* e *F1-score*, sendo que esta última representa uma média harmônica entre *Recall* e *Precision*. As equações (2), (3), e (4) mostram como *Precision*, *Recall* e *F1-score* são calculados, respectivamente. A métrica *Precision* determina qual é o grau de acerto do modelo com relação às amostras que ele classificou como maliciosas. A métrica *Recall* já traz outra perspectiva. Ela mostra quantas amostras maliciosas foram corretamente detectadas com relação ao total que havia no conjunto de dados analisado. *F1-score* é uma métrica bastante útil pois representa uma visão mais abrangente, unindo as perspectivas das métricas *Precision* e *Recall*.

$$precision = \frac{VP}{VP + FP} \tag{2}$$

$$recall = \frac{VP}{VP + FN} \tag{3}$$

$$f1 = 2 \cdot \left( \frac{precision \cdot recall}{precision + recall} \right) \tag{4}$$

Três classificadores supervisionados diferentes foram testados para implementar o modelo alvo: *Random Forest*, Árvore de Decisão J48, e MLP. O algoritmo *Random Forest* é um representante da família dos *ensembles*. Neste tipo de algoritmo, algoritmos mais básicos são utilizados em conjunto para que haja uma melhora no desempenho preditivo final. No caso da *Random Forest*, são utilizadas várias árvores de decisão. Esse algoritmo foi selecionado já que diversos trabalhos o indicam como uma boa opção para detecção de intrusões [Resende and Drummond 2018]. Além disso, gostaríamos de observar o comportamento de um algoritmo do tipo *ensemble* frente às amostras adversárias. O algoritmo J48 é uma árvore de decisão simples, que foi escolhida para servir como uma referência para a análise do *Random Forest*. Por fim, a MLP é uma rede neural e foi selecionada para que haja um cenário onde o modelo utilizado pelo atacante e o modelo alvo coincidam, já que umas das questões que busca-se responder aqui é a influência do conhecimento sobre o modelo alvo no impacto do ataque.

### 4.2. Cenário de Ataque

O cenário preparado para o atacante pode ser considerado do tipo caixa cinza quando o alvo está utilizando os classificadores *Random Forest* e J48. Neste tipo de ataque, o atacante possui acesso (não autorizado) aos dados de treinamento, mas não possui detalhes do classificador utilizado pelo alvo. Nos testes em que o alvo utiliza o classificador MLP, temos um cenário do tipo caixa branca, pois o atacante possui acesso aos dados e utiliza o mesmo classificador que o alvo para executar o ataque. É compreendido também que o atacante consegue interceptar os dados de entrada que serão utilizados na fase de inferência, manipular esses dados e, em seguida, alimentar o modelo de detecção de intrusões para gerar classificações erradas. Há diferentes situações práticas possíveis nas quais o atacante poderia alcançar as condições requeridas para realizar esses ataques. O atacante pode ser um colaborador interno que, de posse de alguns privilégios de acesso, manipularia esses dados sem ser impedido pelos controles de segurança presentes. Em outra hipótese, a rede e os sistemas envolvidos teriam sido configurados de maneira inadequada ou estariam utilizando softwares com vulnerabilidades que permitiriam que mensagens fossem interceptadas e manipuladas, apesar dos controles de segurança.

Para respondermos as questões apresentadas no início desta seção, utilizamos os dois ataques mencionados na Seção 3, variando seus parâmetros e os reaplicando para cada iteração. Para o FGSM, variamos $\epsilon$ de $0,1$ até $0,9$, enquanto, para o JSMA, variamos $\theta$ e $\gamma$ de $0.1$ até $0,9$. $\theta$ é o parâmetro que varia mais devagar, ou seja, fixamos $\theta$, por exemplo, em $0,1$ e variamos $\gamma$ de $0,1$ a $0,9$.

Em ambos os ataques, é utilizada uma MLP treinada com os dados roubados do alvo. Entretanto, o volume de dados de treinamento da MLP utilizado pelo atacante será diminuído progressivamente em 6 rodadas diferentes para obtermos respostas referentes ao impacto do volume de dados roubados. Iniciamos utilizando no treinamento do modelo do atacante o mesmo volume de dados utilizado para treinar o modelo do alvo, ou seja, $60\%$ do conjunto de dados. Em seguida, os testes serão realizados considerando porções de $50\%, 40\%, \cdots, 10\%$ do conjunto de dados. A MLP do atacante sempre é retreinada para cada configuração de ataque, ou seja, para cada variação de $\epsilon$, $\theta$ e $\gamma$. Para calcular a severidade dos ataques, é utilizada a métrica definida de acordo com a equação (5). A métrica varia de 0 a 1, onde resultados mais próximos a 1 indicam ataques mais severos, enquanto resultados mais próximos a 0 representam ataques com menos impacto no *F1-score*.

**Tabela 2. Combinação dos modelos de classificação supervisionada entre atacante e alvo.**

| Atacante | Alvo |
|---|---|
| MLP | Random Forest |
| MLP | Árvore de Decisão J48 |
| MLP | MLP |

$$AS = 1 - \frac{\text{F1-Score depois do ataque}}{\text{F1-Score antes do ataque}} \tag{5}$$

A modificação maliciosa das amostras ocorrerá de maneira a simular uma situação onde o atacante altera apenas aquelas que ele infere que seriam classificadas como maliciosas pelo alvo, pois ele tem o interesse de que elas sejam classificadas como benignas. Para que o atacante escolha estas amostras, ele vai tentar inferir se elas seriam classificadas como maliciosas ou não utilizando o seu próprio modelo treinado com dados roubados. Em suma, toda amostra que for classificada pela MLP treinada pelo atacante como maliciosa será modificada. É importante salientar, portanto, que não estão sendo considerados os rótulos das amostras para que elas sejam escolhidas para manipulação. A Tabela 2 resume as combinações de modelos do alvo e do atacante que teremos nos cenários do experimento.

Os ataques foram implementados por meio da biblioteca *CleverHans*[1][Papernot et al. 2018], que disponibiliza implementações em linguagem Python de diversos ataques baseados em amostras adversárias já reportados na literatura. A versão da biblioteca utilizada nestes estudos foi a 3.1.0. Utilizamos também o framework TensorFlow para construir a MLP tanto para o alvo quanto para o atacante. Para desenvolver a *Random Forest* e a Árvore de Decisão J48, foi utilizada a biblioteca *scikit-learn*[2].

### 4.3. Conjunto de Dados

O conjunto de dados utilizado no projeto foi gerado a partir de um framework de um sistema de energia implementado pela *Mississipi State University* e *Oak Ridge National Laboratory*. Apesar de ser um sistema considerado pequeno, seu comportamento abrange e representa o funcionamento de sistemas de energia maiores e mais complexos [Anthi et al. 2021] [Beaver et al. 2013]. O funcionamento e configuração usados para gerar os cenários do sistema implementado são bem detalhados no estudo feito por [Beaver et al. 2013]. Os dados gerados por esse sistema foram disponibilizados publicamente[3].

O sistema de energia usado na geração do conjunto de dados possui dois geradores de energia, que estão ligados a quatro *breakers*. Cada *breaker* é controlado por um IED (*Intelligent Electronic Device*), que pode ativar ou desativar o *breaker*. Os IEDs recebem comandos dos operadores, mas também podem ser manipulados de maneira maliciosa por um atacante. Cada amostra presente no conjunto de dados traz valores coletados por meio

[1]https://github.com/cleverhans-lab/cleverhans

[2]https://scikit-learn.org/stable/

[3]https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets

**Tabela 3. Média das métricas para os modelos supervisionados sem a ocorrência de amostras adversárias.**

| Modelo | Recall | Precision | F1-score |
|---|---|---|---|
| Random Forest | $\approx 98\%$ | $\approx 95\%$ | $\approx 96\%$ |
| Arvore de Decisão J48 | $\approx 94\%$ | $\approx 93\%$ | $\approx 93\%$ |
| MLP | $\approx 90\%$ | $\approx 87\%$ | $\approx 89\%$ |

**Tabela 4. Média da severidade dos ataques sobre os algoritmos de aprendizado de máquina para ambos os ataques.**

| Ataque | MLP | Random Forest | J48 |
|---|---|---|---|
| JSMA | 0,27 $\pm 0,07$ | 0,09 $\pm 0,03$ | 0,12 $\pm 0,03$ |
| FGSM | 0,32 $\pm 0,02$ | 0,07 $\pm 0,07$ | 0,20 $\pm 0,07$ |

de PMUs instalados na rede elétrica. Estes dispositivos retornam leituras realacionadas a grandezas elétricas como tensão, corrente, e frequência em diferentes pontos da rede [Beaver et al. 2013].

Entre as opções dos conjuntos de dados disponíveis, escolhemos trabalhar com as classes binárias, onde temos as classes *ataque* e *natural*, que representam, respectivamente, as amostras maliciosas e benignas. No contexto de *natural*, são agrupadas as amostras de *eventos naturais* e situações onde não há *ocorrências de eventos*. A classe *ataque* agrupa 28 cenários diferentes de ataques gerados para danificar o sistema de energia. No total, são 37 cenários que são aleatoriamente divididos entre 15 arquivos [Beaver et al. 2013]. O conjunto de dados também possui as seguintes propriedades: cada observação possui 128 características (*features*); 71% das amostras são rotuladas como ataque e 29% como natural para cada um dos 15 arquivos; os 15 arquivos juntos totalizam $78.377$ amostras, sendo $55.663$ maliciosas (ataques) e $22.714$ benignas (naturais).

## 5. Resultados

Os primeiros resultados apresentados nesta seção são relacionados à performance preditiva dos modelos baseados em *Random Forest*, Árvore de Decisão J48 e MLP construídos para o alvo e podem ser visualizados na Tabela 3. Esses resultados retratam as métricas *recall*, *precision* e *f1-score* em um momento onde ainda não havia amostras adversárias. As métricas observadas sugerem que os três modelos alcançaram uma boa performance preditiva. Há uma clara vantagem para o modelo baseado em *Random Forest*, seguido pela J48 e a MLP. Com estes resultados iniciais apresentados, tem-se uma base para discutir os demais resultados, que envolvem amostras adversárias.

### 5.1. Qual ataque causou maior impacto no modelo alvo, FGSM ou JSMA?

A Tabela 4 apresenta as médias e os desvios padrão da severidade dos ataques FGSM e JSMA, englobando todos os diferentes volumes de dados de treinamento utilizados pelo atacante. Observando estes valores, é possível perceber que o FGSM apresentou uma severidade média maior que o JSMA sobre a MLP e a Árvore de Decisão J48. Ambos os ataques tiveram um impacto semelhante sobre a *Random Forest*, que foi o algoritmo que sofreu menos impacto e já havia alcançado os melhores resultados nos testes iniciais sem a presença de amostras adversárias.

Tem-se assim, à primeira vista, o FGSM como uma melhor opção para o atacante em termos de severidade, tendo um impacto mais significativo sobre os algoritmos alvo.

Ainda assim, vale ressaltar que o FGSM é mais invasivo que o JSMA, o que pode torná-lo mais detectável. Apesar da menor severidade, o JSMA obteve resultados próximos ao do FGSM, aplicando alterações menos abrangentes para produzir as amostras adversárias. Como resultado, o JSMA pode ser mais difícil de ser detectado.

### 5.2. Se diminuirmos o volume de dados de treinamento utilizados pelo atacante, o ataque mantém, aumenta ou diminui seu impacto?

Durante os experimentos, as variações do tamanho do conjunto de dados utilizado para treinamento pelo atacante causaram efeitos diferentes para os três algoritmos. Ao observar a Figura 1, que contém resultados para o FGSM, percebe-se que a diminuição do volume de dados roubados implica em uma queda na severidade do ataque mais acentuada sobre a MLP e bem mais leve para a *Random Forest*. Para o J48, o comportamento não teve uma tendência tão clara, já que houve até mesmo um aumento de severidade com a diminuição do volume de dados utilizado pelo atacante de $30\%$ para $20\%$. Ainda, para situações onde há um volume maior de dados sendo utilizados para treinamento, como nos volumes de $60\%$, $50\%$ e $40\%$, o FGSM alcançou severidades médias semelhantes para diferentes volumes de dados roubados. Finalmente, a severidade demonstra uma queda considerável para volumes menores de dados, como em $30\%$.

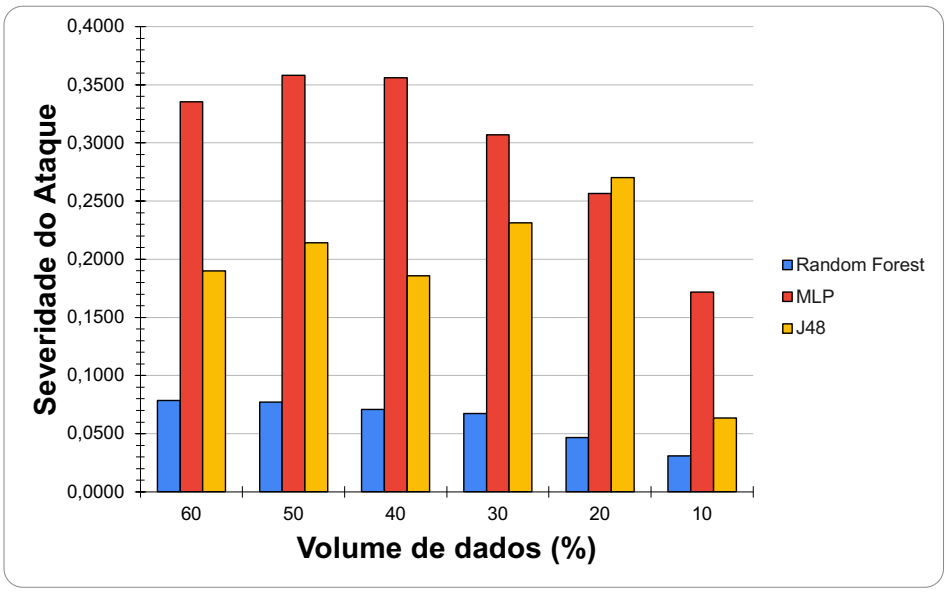


**Figura 1. Severidade média dos ataques FGSM para diferentes volumes de dados utilizados pelo atacante para treinamento.**

Os resultados para o JSMA apresentam algumas diferenças, conforme observado na Figura 2. A redução da severidade do ataque na MLP ocorre até para volumes ainda consideráveis de dados utilizados. Isso pode ser justificado por conta da natureza do ataque. O JSMA depende mais da variedade de dados de treinamento para montar ataques mais eficazes, já que a escolha dos atributos a serem alterados considera a relevância deles para o modelo. Dessa forma, um treinamento menos extenso aparenta afetar negativamente o desempenho do JSMA mais que do FGSM.

Com relação à *Random Forest* e à J48, temos resultados um pouco diferentes da MLP. Como vemos na Figura 2, houve aumento da severidade do ataque com a redução

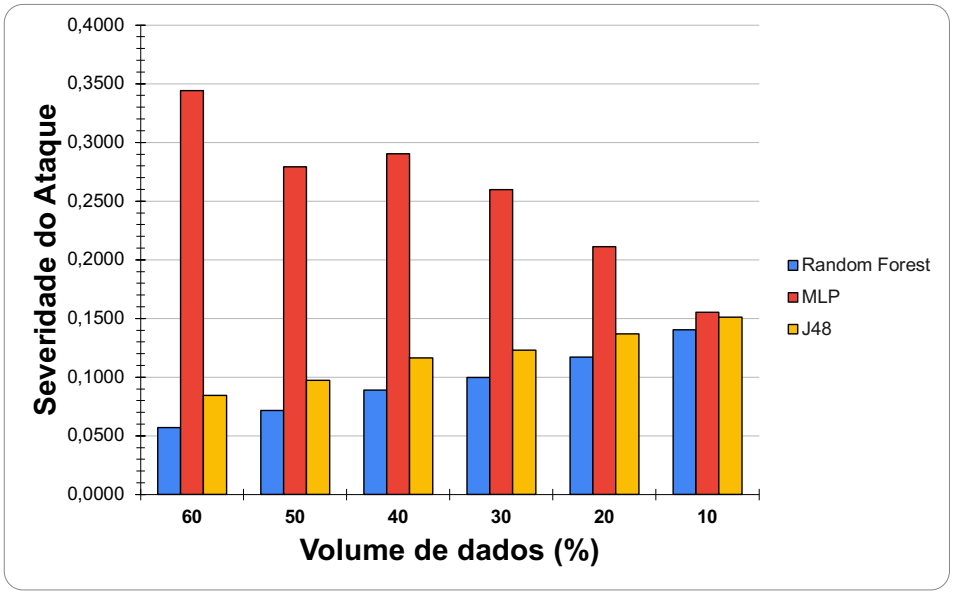


**Figura 2. Severidade média dos ataques JSMA para diferentes volumes de dados utilizados pelo atacante para treinamento.**

do volume de dados roubados tanto para a *Random Forest*, quanto para a J48. Durante os nossos estudos, não foi encontrado um motivo claro para estas diferenças nos comportamentos identificados nos três algoritmos alvo.

### 5.3. Em um cenário onde se conhece o algoritmo utilizado pelo alvo, o ataque se torna mais crítico ao sistema?

Os resultados na Tabela 4 mostram que quando há coincidência entre o modelo do atacante e o modelo alvo, ambos baseados no algoritmo MLP, os ataques tiveram uma severidade próxima a 0,3. Por outro lado, em cenários onde o modelo do atacante e do alvo não coincidem, a severidade caiu. Isso se deve, provavelmente, ao fato que, ao utilizar o mesmo modelo do alvo, o atacante pode preparar exemplos adversários com mais precisão e consegue produzir ataques mais eficazes. Quando há diferença entre o modelo alvo e o modelo atacante, ainda temos ataques com um impacto significativo, mas menos agressivos que no primeiro caso.

Entre os três algoritmos alvo, o *Random Forest* foi o que melhor se saiu contra os adversários, como podemos ver pela Tabela 4. Tanto o *Random Forest* quanto o J48 são algoritmos baseados em árvores de decisão. Porém, o *Random Forest* é uma implementação baseada em um método de *ensemble*, operando com várias árvores de decisão. Sendo assim, seu desempenho tende a ser superior ao da J48 quando treinado sobre os mesmos dados, o que os resultados sugerem que se confirma mesmo quando há amostras adversárias.

## 6. Conclusão

Neste artigo, estudamos o impacto que amostras adversárias causam contra algoritmos de aprendizado de máquina utilizados para detecção de intrusão. Para explorarmos esse cenário, utilizamos um conjunto de dados público que foi gerado a partir de um sistema de energia e treinamos alguns algoritmos de aprendizado de máquina como *Random Forest*,

J48 e MLP sobre esses dados. Em seguida, para efetuar os ataques, utilizamos os mesmos dados de treinamento aplicados sobre os modelos de aprendizado de máquina para gerar amostras maliciosas através do FGSM e JSMA.

Os ataques mostraram que o conhecimento prévio do modelo utilizado pelo alvo pode levar a impactos mais significativos aos sistemas. Já quando ocorre a redução do volume dos dados roubados para treinamento do atacante, os ataques FGSM contra a *Random Forest* e a MLP se tornaram menos severos, enquanto não percebeu-se uma tendência clara para a Árvore de Decisão J48. No caso do JSMA, a severidade dos ataques contra a MLP caiu junto com a redução dos dados utilizados para treinamento, ocorrendo o oposto com a *Random Forest* e a J48. Temos também que de ambos os ataques utilizados, o FGSM obteve uma leve superioridade sobre o JSMA em termos de severidade média. Porém, o JSMA foi o menos invasivo dos dois ataques, sendo, possivelmente, mais difícil de ser detectado por mecanismos de defesa.

Como trabalho futuro, o objetivo é expandir esta investigação, incluindo outras técnicas para geração de amostras adversárias, outros conjuntos de dados e outros modelos alvo. Além disso, também é importante aprofundar a discussão no sentido de desenvolver mecanismos de defesa que sejam capazes de lidar com as diferentes técnicas de ataque, já que os modelos de aprendizado de máquina supervisionado se mostraram vulneráveis às duas técnicas empregadas neste trabalho.

## Referências


Aiken, J. and Scott-Hayward, S. (2019). Investigating adversarial attacks against network intrusion detection systems in SDNs. In *2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, pages 1–7.

Alhajjar, E., Maxwell, P., and Bastian, N. D. (2020). Adversarial machine learning in network intrusion detection systems. *CoRR*, abs/2004.11898.

Anthi, E., Williams, L., Rhode, M., Burnap, P., and Wedgbury, A. (2021). Adversarial attacks on machine learning cybersecurity defences in industrial control systems. *Journal of Information Security and Applications*, 58:102717.

Apruzzese, G., Colajanni, M., Ferretti, L., and Marchetti, M. (2019). Addressing adversarial attacks against security systems based on machine learning. In *2019 11th International Conference on Cyber Conflict (CyCon)*, volume 900, pages 1–18.

Ayub, M. A., Johnson, W. A., Talbert, D. A., and Siraj, A. (2020). Model evasion attack on intrusion detection systems using adversarial machine learning. In *2020 54th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6.

Beaver, J. M., Borges-Hink, R. C., and Buckner, M. A. (2013). An evaluation of machine learning methods to detect malicious SCADA communications. In *2013 12th International Conference on Machine Learning and Applications*, volume 2, pages 54–59.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Ibitoye, O., Shafiq, O., and Matrawy, A. (2019). Analyzing adversarial attacks against deep learning for intrusion detection in IoT networks. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6.

Kim, S., Park, K.-J., and Lu, C. (2022). A survey on network security for cyber–physical systems: From threats to resilient design. *IEEE Communications Surveys Tutorials*, 24(3):1534–1573.

Ning, X. and Jiang, J. (2022). Design, analysis and implementation of a security assessment/enhancement platform for cyber-physical systems. *IEEE Transactions on Industrial Informatics*, 18(2):1154–1164.

Papernot, N., Faghri, F., Carlini, N., Goodfellow, I., Feinman, R., Kurakin, A., Xie, C., Sharma, Y., Brown, T., Roy, A., Matyasko, A., Behzadan, V., Hambardzumyan, K., Zhang, Z., Juang, Y.-L., Li, Z., Sheatsley, R., Garg, A., Uesato, J., Gierke, W., Dong, Y., Berthelot, D., Hendricks, P., Rauber, J., and Long, R. (2018). Technical report on the cleverhans v2.1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768*.

Papernot, N., McDaniel, P., and Goodfellow, I. (2016a). Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. (2016b). The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroSP)*, pages 372–387.

Pawlicki, M., Choraś, M., and Kozik, R. (2020). Defending network intrusion detection systems against adversarial evasion attacks. *Future Generation Computer Systems*, 110:148–154.

Resende, P. A. A. and Drummond, A. C. (2018). A survey of random forest based methods for intrusion detection systems. *ACM Comput. Surv.*, 51(3).

Tabassi, E., Burns, K. J., Hadjimichael, M., Molina-Markham, A. D., and Sexton, J. T. (2019). A taxonomy and terminology of adversarial machine learning. *NIST IR*, pages 1–29.

Wang, Z. (2018). Deep learning-based intrusion detection with adversaries. *IEEE Access*, 6:38367–38384.

Yang, K., Liu, J., Zhang, C., and Fang, Y. (2018). Adversarial examples against the deep learning based network intrusion detection systems. In *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*, pages 559–564.

Zarpelão, B. B., Barbon Junior, S., Acarali, D., and Rajarajan, M. (2020). *How Machine Learning Can Support Cyberattack Detection in Smart Grids*, pages 225–258. Springer International Publishing, Cham.

Zhang, J., Pan, L., Han, Q.-L., Chen, C., Wen, S., and Xiang, Y. (2022). Deep learning based attack detection for cyber-physical system cybersecurity: A survey. *IEEE/CAA Journal of Automatica Sinica*, 9(3):377–391.