Aprendizado de Máquina Adversário contra Detectores de Anomalias em Séries Temporais

Felipe Dallmann Tomazeli[1], **Bruno Bogaz Zarpelao**[1]

[1]Departamento de Computação – Universidade Estadual de Londrina (UEL)
Caixa Postal 10.011 – CEP 86057-970 – Londrina – PR – Brasil

felipe.dallmann@uel.br, brunozarpelao@uel.br

Abstract. *The growing availability of devices connected to the Internet of Things (IoT) and also of cyber-physical systems, implies an increase in time series data, since many of these devices transmit continuous data streams. In this scenario, it is important to analyze these data to support decision making. However, unexpected events can cause anomalies in this data. With this in mind, studies indicate the possibility of using Machine Learning to detect anomalies in time series. However, Machine Learning algorithms are known to be vulnerable to Adversarial Machine Learning attacks. Therefore, in view of the applicability and importance of these applications, this work proposes to observe different types of attacks and their impacts against anomaly detectors in time series using Machine Learning, in addition to the implementation of defense techniques to evaluate the protection of the system with them.*

Resumo. *A crescente disponibilidade de dispositivos conectados à Internet das Coisas (Internet of Things - IoT) e também de sistemas ciberfísicos, implica no aumenta dos dados organizados em séries temporais, já que, muitos desses dispositivos transmitem fluxos de dados contínuos ao longo do tempo. Nesse cenário, é importante a análise desses dados para apoiar na tomada de decisões. No entanto, eventos inesperados podem causar anomalias nestas séries. Tendo isso em vista, estudos indicam a possibilidade de utilizar Aprendizado de Máquina para detecção de anomalias em séries temporais. Contudo, sabe-se que algoritmos de Aprendizado de Máquina são vulneráveis a ataques de Aprendizado de Máquina Adversário. Portanto, em vista da aplicabilidade e importância dessas aplicações, este trabalho propõe observar diferentes categorias de ataques e seus impactos contra detectores de anomalia em séries temporais utilizando Aprendizado de Máquina, além da implementação de técnicas de defesa para avaliar a proteção do sistema com elas.*

1. Introdução

Desde o final do século XX, o crescimento acelerado da Internet trouxe diversos benefícios e comodidades à humanidade. Um dos paradigmas onde a Internet faz-se essencial é em sistemas ciberfísicos e dispositivos conectados à Internet das Coisas, como em eletrodomésticos, dispositivos de segurança e aparelhos relacionados à saúde. Esses sistemas são caracterizados pela capacidade de interagir e se comunicar com o mundo físico através da computação [2].

Assim como a internet, sistemas de Inteligência Artificial (IA) estão cada vez mais presentes no cotidiano das pessoas, auxiliando na tomada de decisões e automação

de processos. Este termo refere-se à capacidade dos computadores executarem tarefas comumente realizadas por humanos, como o reconhecimento de objetos em uma imagem. Para possibilitar que sistemas alcancem esse nível de inteligência, é possível utilizar um método de análise de dados, chamado Aprendizado de Máquina (AM), no qual o sistema passa por uma fase de treinamento, acumulando experiências e aprendendo. Após o treinamento, ele passará por fases de teste e validação, para que seja possível avaliar o desempenho na execução da tarefa para a qual foi designado.

Séries temporais são coleções de dados colhidas sequencialmente em ordem cronológica. São comumente aplicadas na previsão de valores futuros ou na identificação de padrões, já que as observações futuras normalmente são dependentes das que já aconteceram. Com o objetivo de analisar esses padrões, diversos modelos foram desenvolvidos. Entre eles, estão os que envolvem Aprendizado de Máquina [11]. Tendo em vista o crescimento na quantidade de séries temporais [22], juntamente com a expansão das tecnologias citadas anteriormente e a possibilidade de utilizá-las com as séries temporais, é notável a importância socioeconômica desses sistemas [17].

Além disso, séries temporais estão sujeitas à ocorrência de anomalias [8]. Anomalias são valores que fogem do padrão observado ao longo da série, podendo significar um erro na leitura do valor ou um acontecimento inesperado. Portanto, essas irregularidades devem ser detectadas e tratadas quando necessário. Para tal fim, Algoritmos de Aprendizado de Máquina podem ser utilizados também na detecção dessas anomalias em séries temporais [18].

Neste cenário, é importante garantir segurança e confiabilidade nos modelos de Aprendizado de Máquina [2]. Em vista do exposto, o trabalho em questão pretende analisar os impactos, técnicas e mecanismos de defesa para ataques de Aprendizado de Máquina Adversário contra detectores de anomalias em séries temporais. Espera-se, pelos resultados, analisar a severidade dos ataques e possíveis estratégias de defesa.

2. Fundamentação Teórico-Metodológica e Estado da Arte

2.1. Séries Temporais

As séries temporais representam dados coletados sequencialmente, em ordem cronológica, com intervalos de tempo equidistantes. São utilizados para encontrar padrões ou realizar um rastreamento das mudanças nos dados ao longo do tempo [10] visando prever os valores futuros. Basicamente, as análises em séries temporais levam em consideração as observações passadas para determinar os valores futuros.

Exemplos de aplicações que lidam com séries temporais:

- Análise de mercado financeiro em preços de ações
- Atividade elétrica do cérebro
- Atividade elétrica do coração
- Vendas anuais
- Previsão do tempo

As séries temporais podem ser classificadas como estacionárias ou não estacionárias, sendo que, as não estacionárias contam com dois elementos principais: sazonalidade e tendência [16] [23] [21].

- **Tendência**: ocorre quando há um padrão de crescimento ou decrescimento, a longo prazo, nos dados da série.

- **Sazonalidade**: padrões nos dados que se repetem periodicamente, como, por exemplo, a variação da temperatura ao longo do dia.

- **Autocorrelação**: é a característica das séries temporais que define a relação entre uma observação e as observações anteriores. Autocorrelação de primeira ordem é a relação da observação atual com a anterior, de segunda ordem é a relação da atual com as duas anteriores, e assim por diante.

- **Estacionariedade**: característica de séries temporais que apresentam média, variância e autocorrelação constantes. De maneira geral, as propriedades estatísticas da série não variam ao longo do tempo.

  Exemplo de série temporal [1] [13]:

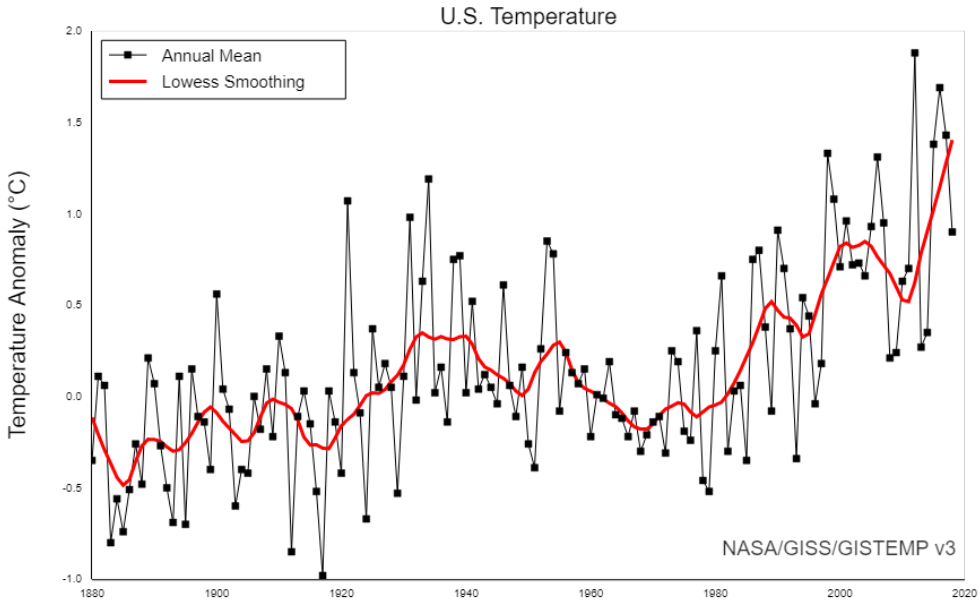


Figura 1. Mudança de temperatura média anual nos Estados Unidos

Dada a importância e ampla aplicabilidade dessas coleções de dados ao longo dos últimos anos, estudos propuseram diferentes métodos de análise [3] [9]. Dentre eles, estão os de Aprendizado de Máquina [6] [11].

2.2. Aprendizado de Máquina

Aprendizado de Máquina é um campo de estudo que tem como objetivo viabilizar que algoritmos de computadores aprendam a realizar determinadas tarefas de maneira semelhante aos humanos [19], acumulando conhecimento com base nos seus próprios erros. De maneira geral, esses algoritmos aprendem na fase de treinamento, quando recebem os dados de entrada. Posteriormente, para serem avaliados, são submetidos a uma fase de teste.

Enquanto grande parte dos algoritmos de computadores recebem os dados e, através de uma função pré determinada, geram uma saída, os algoritmos de Aprendizado de Máquina recebem os dados de entrada e, com base em propriedades estatísticas, geram a própria função [20].

A fase de treinamento, na qual se acumulam experiências, leva em consideração uma função chamada função de perda, que representa o erro na saída do algoritmo. Com isso, parâmetros da função gerada pelo próprio algoritmo de AM são ajustados, visando diminuir esse erro, através do gradiente descendente, por exemplo.

Na fase de testes, o algoritmo de Aprendizado de Máquina será avaliado, com base em métricas estatísticas, por suas predições. Uma vez que o sistema foi treinado, é necessário utilizar dados ainda não vistos pelo algoritmo de AM. Dessa forma, é possível analisar o comportamento do modelo, simulando um cenário real com possíveis novos dados de entrada.

Existem diferentes abordagens para as técnicas de Aprendizado de Máquina, sendo elas: aprendizado supervisionado e não supervisionado.

Aprendizado supervisionado é a técnica de construir modelos que fazem predições baseadas em evidências. Para isso, os algoritmos utilizam, durante o treinamento, dados de entrada já rotulados. Espera-se que, depois de treinados, sejam capazes de realizar predições razoáveis, com entradas parecidas às do treinamento. Duas abordagens diferentes podem ser tomadas no aprendizado supervisionado: problemas de regressão e de classificação

- **Regressão**: consiste em prever respostas contínuas, inferindo a relação entre uma variável dependente com uma ou mais variáveis independentes, como, por exemplo, alterações de temperatura ao longo do tempo.
- **Classificação**: consiste em classificar entradas em determinadas categorias, como, por exemplo, o reconhecimento de objetos em uma imagem.

No aprendizado não supervisionado, o algoritmo aprende a identificar padrões nos dados sem nenhum tipo de classificação ou rotulação. Nesse caso, o sistema aprende apenas com os dados de entrada. Diferente do supervisionado, não passa por uma fase de validação, que leva em consideração as entradas e suas respectivas saídas, mas o algoritmo apenas aprende a agrupar os dados de acordo com seus atributos.

2.3. Detecção de Anomalias

Detecção de anomalias em séries temporais tornou-se um campo de interesse de pesquisadores quando se trata desse tipo de coleção de dados. Valores atípicos ou anomalias são observações que desviam do comportamento padrão da série temporal causadas por eventos inesperados.

Como citado por [14]:

"An observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism".

Anomalias podem ser classificadas em dois tipos: eventos indesejados e eventos de interesse [5].

O primeiro corresponde às variações que ocorrem por causa de ruídos ou erros na captação dos dados, como, por exemplo, um sensor de batimentos cardíacos com defeito. De maneira geral, esse tipo de anomalia não é interessante para o estudo e compreensão dos dados analisados. Portanto, devem ser deletados ou corrigidos para que não atrapalhem na qualidade das análises [12].

Por outro lado, alguns eventos atípicos podem ser decorrentes de fenômenos relevantes para a análise dos dados. Mesmo os dados com características sazonais estão sujeitos a esse tipo de fenômeno - como a recente pandemia causada pela COVID-19. Neste cenário, as técnicas de detecção de anomalia não procuram retirar valores indesejados, mas entender esses fenômenos e seus impactos. Um exemplo claro disso é a constatação de fraudes, que detecta e analisa comportamentos inusitados.

2.4. Aprendizado de Máquina Adversário

Devido à importância e aplicabilidade dos algoritmos de aprendizado de máquina, é de se esperar que agentes mal-intencionados tentem explorar vulnerabilidades a fim de obter vantagens ou causar danos [4]. Uma das maneiras de explorar essas fragilidades é através de técnicas de Aprendizado de Máquina Adversário.

Este tipo de ataque consiste em tentar enganar o algoritmo de AM a fim de prejudicar a confiabilidade, integridade e segurança do sistema [15]. Para executar essa técnica, o atacante pode usar algumas estratégias diferentes [7][24], tais como: ataques de evasão e envenenamento.

O primeiro consiste em aplicar pequenas perturbações nos dados na fase de teste, resultando em classificações incorretas. Nesse caso, as técnicas de evasão podem ter algumas abordagens diferentes, levando em consideração se será um ataque direcionado a uma classe específica dos dados ou se será indiscriminado. Já o segundo diz respeito aos ataques onde o algoritmo de AM é acometido já na fase de treinamento, com dados contaminados, fazendo com que, posteriormente, tenha um baixo desempenho preditivo.

Além dessas técnicas, um possível atacante levaria em consideração a disponibilidade dos dados e informações do modelo alvo, configurando dois cenários de conhecimento [24]: caixa branca e caixa preta.

- **Caixa branca**: cenário onde o atacante tem total conhecimento sobre o alvo do ataque, incluindo dados de treinamento e de teste, parâmetros do modelo e suas configurações.
- **Caixa preta**: cenário onde o atacante não tem total conhecimento sobre o alvo, apenas as saídas e resultados do modelo.

2.4.1. Defesas contra Aprendizado de Máquina Adversário

Com o intuito de reduzir os impactos, diferentes métodos de defesa podem ser utilizados. Diferentes tipos de ataque implicam em diferentes técnicas de defesa, levando em consideração se é um ataque de evasão ou de envenenamento:

- **Defesa contra ataques de envenenamento**: os ataques de envenenamento ocorrem durante a fase de treinamento do modelo, desse modo, tem como objetivo reduzir as possíveis distorções causadas pelo ataque. Um exemplo claro é a

sanitização dos dados, retirando amostras que aumentam a ocorrência de erros do modelo.

- **Defesa contra ataques de evasão**: tem como objetivo aumentar a robustez do modelo. Nesse caso, devem ser tomadas medidas proativas, precedendo a fase de teste do algoritmo. Um exemplo dessa técnica é a aplicação de dados já contaminados pelo atacante na fase de treinamento, mas contendo a classificação correta para aquela amostra. Dessa forma, espera-se que o modelo saiba lidar melhor com amostras manipuladas por um possível atacante.

De qualquer forma, essas técnicas podem causar um efeito prejudicial à capacidade preditiva do modelo [7].

3. Objetivos

O objetivo deste trabalho é analisar a severidade dos diferentes tipos de ataque pela técnica de Aprendizado de Máquina Adversário contra detectores de anomalia em séries temporais, explorando as particularidades da estruturação desse tipo de dado

Para tal fim, um modelo de Aprendizado de Máquina será definido com base em seus resultados preditivos para detecção de anomalias em séries temporais e, posteriormente, esse modelo será o alvo dos ataques. Além disso, um bom conjunto de dados deve ser escolhido, possibilitando explorar as características das séries temporais.

Tendo o modelo e seus resultados já estabelecidos, serão exploradas diferentes técnicas de Aprendizado de Máquina Adversário, como as técnicas de envenenamento e de evasão. Dessa forma, é possível explorar quais são mais severas.

Por fim, será possível aplicar estratégias de defesa, visando diminuir ou mitigar os impactos causados. Com os resultados obtidos nas etapas anteriores e nesta última, será possível ter um parâmetro de quais ataques são mais severos e quais mecanismos de defesa são mais eficazes.

4. Procedimentos metodológicos/Métodos e técnicas

Primeiramente, será realizada uma revisão bibliográfica de estudos que abrangem os temas de Aprendizado de máquina para detecção de anomalias em séries temporais, possibilitando um melhor entendimento técnico do assunto. Tendo essa base de conhecimento estabelecida, o foco dos estudos será sobre ataques de Apredizado de Máquina Adversário para que, então, seja possível relacionar ambos.

Posteriormente, será determinada uma boa base de dados que permita explorar as particularidades das séries temporais observadas nos estudos anteriores. Sucessivamente, um modelo de aprendizado de máquina será construído utilizando algum *framework* como *TensorFlow, Keras* ou *Pytorch*. Dessa forma, será possível analisar o desempenho preditivo do modelo sobre a base de dados escolhida.

Uma vez que o modelo alvo foi construído, a ferramenta para realizar os ataques será definida e, então, os ataques serão efetuados variando as técnicas e suas particularidades. Isso possibilitará que seja feita uma análise sobre o tipo de ataque e as condições em que é mais severo, levando em consideração os resultados preditivos obtidos anteriormente e nessa etapa.

A seguir, técnicas de defesa serão aplicadas a fim reduzir os impactos causados pela etapa anterior. Essas técnicas são caracterizadas por se aplicarem aos ataques de evasão ou de envenenamento. Portanto, as condições do ataque devem ser levadas em consideração para a aplicação da defesa.

Por fim, com todos os resultados obtidos anteriormente, será possível realizar uma análise quanto aos ataques escolhidos e aos métodos de defesa para cada cenário estudado. Essa análise será consolidada na escrita de um relatório, contendo as métricas utilizadas para determinar a capacidade preditiva, a severidade dos ataques e a possível redução desses impactos com a aplicação de técnicas de defesa.

5. Cronograma de Execução

Atividades:

1. Revisão Bibliográfica;
2. Levantamento de um conjuntos de dados público que permita explorar as características das séries temporais e realizar os experimentos;
3. Implementação dos ataques estudados no passo 1 para construção dos diferentes cenários de teste;
4. Implementação do código de aprendizado de máquina para detecção de anomalias em séries temporais. Este será o alvo dos ataques;
5. Preparação dos cenários cenários de experimentação;
6. Implementação das possíveis técnicas de defesa contra os ataques;
7. Aplicação dos métodos de defesa nos cenários de ataque do passo 5 para observação do impacto dos ataques e eficácia da estratégia de defesa;
8. Escrita do Trabalho de Conclusão de Curso;

Tabela 1. Cronograma de Execução

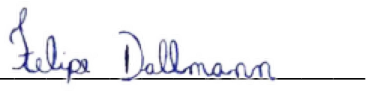
	ago	set	out	nov	dez	jan	fev	mar	abr	mai
Atividade 1	X									
Atividade 2		X								
Atividade 3		X	X							
Atividade 4			X	X						
Atividade 5			X	X	X					
Atividade 6					X	X				
Atividade 7					X	X	X	X	X	
Atividade 8						X	X	X	X	X

6. Contribuições e/ou Resultados esperados

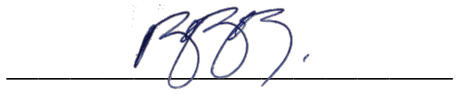
Com o desenvolvimento deste trabalho, espera-se levantar questionamentos sobre a segurança de algoritmos de aprendizado de máquina para detecção de anomalias em séries temporais. Pretende-se avaliar a seguranças desses algoritmos através de diferentes ataques e variações, de forma a entender a nocividade de cada um deles. Além disso, técnicas de defesa serão aplicadas e avaliadas em relação à eficácia contra os ataques.

7. Espaço para assinaturas

Londrina, *12 de setembro de 2022*.



Aluno



Orientador

Referências

[1] GISS Surface Temperature Analysis (GISTEMP v4). `https://data.giss.nasa.gov/gistemp/`. [Accessado em: 06/09/2022].

[2] Rasim Alguliyev, Yadigar Imamverdiyev, and Lyudmila Sukhostat. Cyber-physical systems and their security issues. *Computers in Industry*, 100:212–223, 2018.

[3] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data mining and knowledge discovery*, 31(3):606–660, 2017.

[4] Marco Barreno, Blaine Nelson, Russell Sears, Anthony D Joseph, and J Doug Tygar. Can machine learning be secure? In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pages 16–25, 2006.

[5] Ane Blázquez-García, Angel Conde, Usue Mori, and Jose A Lozano. A review on outlier/anomaly detection in time series data. *ACM Computing Surveys (CSUR)*, 54(3):1–33, 2021.

[6] Gianluca Bontempi, Souhaib Ben Taieb, and Yann-Aël Le Borgne. Machine learning strategies for time series forecasting. In *European business intelligence summer school*, pages 62–77. Springer, 2012.

[7] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018.

[8] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), jul 2009.

[9] Chris Chatfield. *Time-series forecasting*. Chapman and Hall/CRC, 2000.

[10] Philippe Esling and Carlos Agon. Time-series data mining. *ACM Comput. Surv.*, 45(1), dec 2012.

[11] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963, mar 2019.

[12] Prasanta Gogoi, Dhruba K Bhattacharyya, Bhogeswar Borah, and Jugal K Kalita. A survey of outlier detection methods in network anomaly identification. *The Computer Journal*, 54(4):570–588, 2011.

[13] J. E. Hansen, R. Ruedy, M. Sato, M. Imhoff, W. Lawrence, D. Easterling, T. Peterson, and T. Karl. A closer look at united states and global surface temperature change. *J. Geophys. Res.*, 106:23947–23963, 2001.

[14] Douglas M Hawkins. *Identification of outliers*, volume 11. Springer, 1980.

[15] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 43–58, 2011.

[16] Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.

[17] Kyoung-Dae Kim and P. R. Kumar. Cyber–physical systems: A perspective at the centennial. *Proceedings of the IEEE*, 100(Special Centennial Issue):1287–1308, 2012.

[18] Amer Malki, El-Sayed Atlam, and Ibrahim Gad. Machine learning approach of detecting anomalies and forecasting time-series of iot devices. *Alexandria Engineering Journal*, 61(11):8973–8986, 2022.

[19] Tom M Mitchell and Tom M Mitchell. *Machine learning*, volume 1. McGraw-hill New York, 1997.

[20] Maria Carolina Monard and José Augusto Baranauskas. Conceitos sobre aprendizado de máquina. *Sistemas inteligentes-Fundamentos e aplicações*, 1(1):32, 2003.

[21] Douglas C Montgomery, Cheryl L Jennings, and Murat Kulahci. *Introduction to time series analysis and forecasting*. John Wiley & Sons, 2015.

[22] Diego Silva, Rafael Giusti, Eamonn Keogh, and Gustavo Batista. Speeding up similarity search under dynamic time warping by pruning unpromising alignments. *Data Mining and Knowledge Discovery*, 32, 07 2018.

[23] Steven Wheelwright, Spyros Makridakis, and Rob J Hyndman. *Forecasting: methods and applications*. John Wiley & Sons, 1998.

[24] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9):2805–2824, 2019.