Análise Exploratória de Dados do StreamDataNetClass

Gabriel Silva[1], Adriele Colossi[1] Giancarlo Lucca[2], Renata H. S. Reiser[1], Adenauer Yamin[1], Lizandro de Souza Oliveira[2], Helida Santos[3], Bruno M. P. Moura[4], Eduardo Maroñas Monks[1]

[1]CDTec – Universidade Federal de Pelotas (UFPel), Pelotas-RS – Brasil

[2]CCST – Universidade Católica de Pelotas (UcPel), Pelotas-RS – Brasil

[3]C3 – Universidade Federal do Rio Grande (FURG), Rio Grande-RS – Brasil

[4]DTIC – Universidade Federal do Pampa (UNIPAMPA), Bagé–RS – Brasil

{grosilva,akocolossi,reiser,adenauer,emmonks}@inf.ufpel.edu.br

{giancarlo.lucca,lizandro.oliveira}@ucpel.edu.br

helida@furg.br

brunomoura@unipampa.edu.br

Abstract. *Network traffic classification has received more attention from the technological and academic communities as the number of devices and users has grown. Considering the artificial intelligence (AI) field, the existence of valuable and consistent datasets is extremely important. This study presents an Exploratory Data Analysis (EDA), using the Python programming language, in a dataset created for the network streaming traffic classification problem, the* StreamDataNetClass*.*

Resumo. *A classificação de tráfego de rede tem recebido maior atenção das comunidades tecnológica e acadêmica à medida que o número de dispositivos e usuários cresce. Considerando o campo da inteligência artificial (IA), é extremamente importante a existência de conjuntos de dados valiosos e consistentes. Este estudo apresenta uma Análise Exploratória de Dados (AED), utilizando a linguagem de programação Python, em um conjunto de dados criado para o problema de classificação de tráfego de streaming em rede, o StreamDataNetClass.*

1. Introdução

O problema de classificação de tráfego de rede é um campo de pesquisa com muitos desafios para a classificação dos dados ser efetiva, questões como métodos de segurança, criptografia, qualidade de serviço, gerenciamento de rede e contabilidade de serviços [Banihashemi and Akhtarkavan 2022, Draper-Gil et al. 2016, Salman et al. 2020]. Este tópico tem ganhado importância crescente devido ao crescimento exponencial das redes globais, impulsionado pela proliferação de usuários, dispositivos e da Internet das Coisas (IoT), que contribuem para um *throughput* massivo de dados [Qiu et al. 2018].

As técnicas de Inteligência Artificial (IA) têm sido amplamente aplicadas em diversas áreas de conhecimento [Li et al. 2024, Wang et al. 2025, Zhang et al. 2024], incluindo classificação de tráfego de rede [Hattak et al. 2023, M. De Resende et al. 2022].

No entanto, o sucesso dos algoritmos de IA depende da qualidade e consistência dos conjuntos de dados utilizados durante o treinamento. Antes de aplicar técnicas e diferentes modelos de IA em um conjunto de dados, uma análise minuciosa é necessária para entender sua estrutura, características e problemas potenciais. Tais análises auxiliam na preparação dos conjuntos de dados e melhoram o desempenho algorítmico.

Nesse contexto, a Análise Exploratória de Dados (AED) desempenha um papel fundamental, permitindo a observação e conclusões sobre o conjunto de dados trabalhado, através de gráficos e resultados numérico-estatísticos [Komorowski et al. 2016, Fu 2024]. Outro ponto que a AED nos evidencia são as irregularidades de dados que provavelmente influenciariam na eficácia de um modelo, sendo possível ajustar os dados e otimizar o conjunto de dados [Raparthi et al. 2024, Ventocilla and Riveiro 2019].

O conjunto de dados de tráfego de rede utilizado neste estudo, o *StreamDataNetClass*, foi inicialmente introduzido em [Monks et al. 2022] e compreende quatro conjuntos de dados de tamanhos variados. Esses conjuntos de dados capturam características de tráfego de *streaming* de rede para aplicá-las ao *FuzzyNetClass* [Monks 2023], um modelo de classificação que combina lógica fuzzy [Zadeh 1988] com técnicas de IA para classificar tráfego de *streaming* de vídeo [Rao et al. 2011].

Este estudo tem como objetivo realizar uma AED para o conjunto de dados *StreamDataNetClass*, utilizando da linguagem Python. A escolha da linguagem, como ferramenta analítica para conduzir a AED, foi devido às suas extensas bibliotecas e ampla utilização na comunidade acadêmica [Sahoo et al. 2019]. O foco é compreender as principais características, através de diferentes gráficos e dados estatísticos, para obter conclusões sobre este conjunto de dados.

Este artigo segue a seguinte estrutura: Na Seção 2 é demonstrada, resumidamente, como foi feita a construção do *StreamDataNetClass*, focando em como os dados foram capturados e convertidos para os valores finais. Em sequência, na Seção 3 está todo o processo da AED realizada nesse conjunto de dados e, ao final, na Seção 4, são apresentados um debate dos resultados e uma perspectiva de trabalhos futuros.

2. Construção do StreamDataNetClass

Nesta seção está detalhada a construção dos quatro *datasets* utilizados nessa AED, criados originalmente para a testagem do classificador de rede *FuzzyNetClass*[1] introduzido por [Monks et al. 2022], cujo foco é a classificação de tráfego de *streaming* de vídeo.

2.1. Formação dos *datasets* originais

As ferramentas utilizadas em [Monks et al. 2022] foram o *Tcpdump* e o *CicFlowMeter*[2], sendo as duas compatíveis com o formato *libpcap*. A captura foi relizada em ambiente controlado, visando apenas a identificação de dados de *streaming* de vídeo.

Ao final de todo o processo foram construídos quatro *datasets*, cada um com um número diferente de amostras, porém com os mesmos onze atributos apresentados na Tabela 1. Com esses *datasets* definidos, cada um passou por um processo de análise envolvendo a remoção de *outliers*. Adicionalmente, todos os dados foram normalizados na escala de 0 a 10, utilizando da técnica MinMax.

[1]https://github.com/emmonks/datasets/tree/main/25092022

[2]https://github.com/ahlashkari/CICFlowMeter

Tabela 1. Descrição dos Atributos presentes no *dataset StreamDataNetClass*.

ID	Atributo	Descrição
0	Fwd_Packet_Length Mean	Tamanho médio de pacotes em fluxo de *upload*
1	Fwd_Packet_Length Std	Desvio padrão do tamanho de pacotes em fluxo de *upload*
2	Bwd_Packet_Length Mean	Tamanho médio de pacotes em fluxo de *download*
3	Bwd_Packet_Length Std	Desvio padrão do tamanho de pacotes em fluxo de *download*
4	Flow_IAT Mean	Tempo médio entre dois pacotes enviados em um único fluxo
5	Flow_IAT Std	Desvio padrão entre dois pacotes enviados em um único fluxo
6	Fwd_IAT Mean	Tempo médio entre dois pacotes em fluxo de *upload*
7	Fwd_IAT Std	Desvio padrão entre dois pacotes em fluxo de *upload*
8	Bwd_IAT Mean	Tempo médio entre dois pacotes em fluxo de *download*
9	Packet_Length Mean	Tamanho médio de pacotes
10	Packet_Length Std	Desvio padrão do tamanho de pacotes

2.2. Ajustes e Organização dos *Datasets* Originais

Para uma melhor análise, todos os quatro *datasets* originais foram agrupados em um único *dataset*, considerando que todos possuem a mesma estrutura. Toda a AED foi realizada utilizando a linguagem Python devido a sua facilidade de desenvolvimento e sua alta utilização para esse tipo de problemática. Também foram utilizadas as bibliotecas Pandas e Numpy para analisar e realizar as operações necessárias no conjunto de dados.

É importante ressaltar que, dado o processo de construção do *dataset* descrito na Seção 2, não existem dados inexistentes ou nulos no conjunto de dados. Outro ponto é que os valores mínimos e máximos de todos os dados no *dataset* está no intervalo [0, 10].

Com o agrupamento dos quatro *datasets*, um único *dataset* foi formado, contendo 13530 linhas e doze colunas. Dessas doze colunas, onze são os atributos e, a última coluna, a classe referente a cada dado.

A remoção de dados duplicados foi realizada, sendo esperada uma alta quantidade devido à junção dos quatro *datasets* originais. Ao todo, 5108 linhas do *dataset* original foram removidas, resultando no *StreamDataNetClass* contendo 8422 instâncias. A distribuição desses dados em cada classe está de acordo com a Figura 1.

3. AED na Problemática de Classificação de Tráfego de *Streaming*

Nessa seção descreve-se o processo da AED efetuada no *StreamDataNetClass*.

3.1. Valores Estatísticos

Como primeira etapa nessa AED, foram identificados os principais valores estatísticos, que podem ser vistos na Tabela 2. Os principais objetivos dessa etapa é extrair o comportamento dos dados em cada atributo, e analisar valores que possam indicar alguma característica marcante do conjunto de dados.

Nota-se que a coluna “Moda” de cada atributo, parece deter a maior quantidade de valores nos extremos. Tal característica pode indicar, desde essa primeira análise,

Tabela 2. Estatísticas descritivas dos atributos.

Atributos				
At	Média	DP	Moda	Mediana
0	1.320	2.174	0.0	0.365
1	4.296	3.702	0.0	0.221
2	3.780	3.337	0.0	0.864
3	5.428	3.330	0.0	0.0
4	4.091	4.488	10.0	1.148
5	1.996	3.447	0.0	0.022
6	4.721	4.581	10.0	0.930
7	1.841	3.239	0.0	0.007
8	5.515	4.687	0.0	0.010
9	2.815	3.267	0.0	0.976
10	4.693	3.636	0.0	2.701

Siglas: At - Atributo, DP - Desvio Padrão.

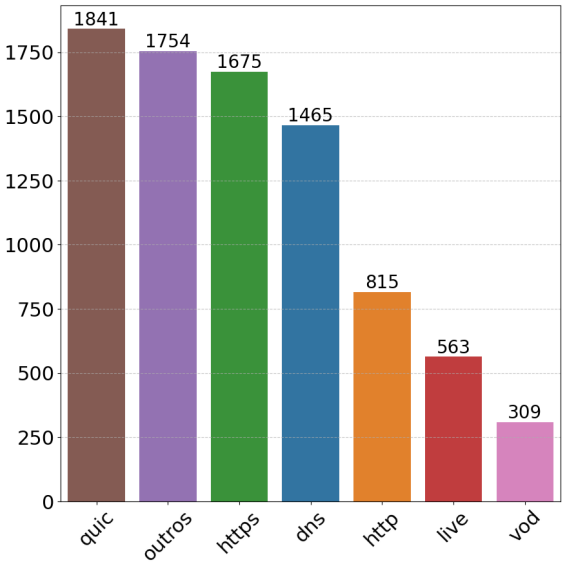

Figura 1. Número de instâncias em cada protocolo.

que esse conjunto de dados não segue uma distribuição normal de dados. Observando a coluna "Mediana", fica evidente que existem muitos valores próximos a zero, indicando que é bem provável a presença de vários *outliers*, algo que será demonstrado na Seção 3.2.

3.2. Visualização dos Dados

Nesta etapa são utilizados gráficos de densidade, *boxplot* e dispersão para entender o comportamento dos dados no *dataset* estudado.

3.2.1. Gráfico de *Boxplot*

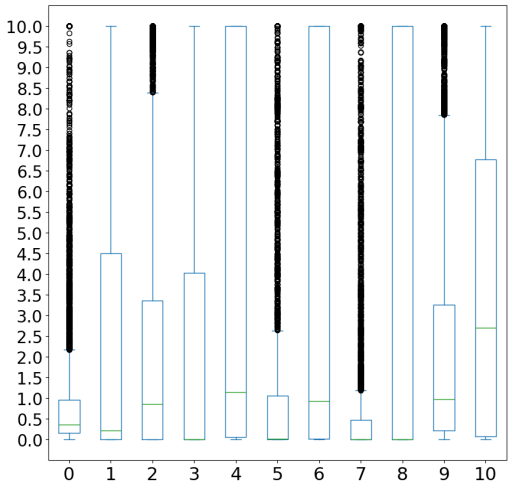

Figura 2. Boxplots de cada atributo.

Os gráficos de *Boxplot* ajudam a entender melhor a quantidade de *outliers* presentes no conjunto de dados estudado. Na Figura 2 é apresentado o *boxplot* simples de cada

atributo no *dataset*. Cada atributo é indicado pelo número do eixo X referente a Tabela 1. Os círculos em preto indicam os dados que foram considerados *outliers*.

Ressalta-se na Figura 2 que os atributos que possuem a menor média, mostrados na Tabela 2, foram os que obtiveram alguma presença de *outliers*. Sendo os atributos $0, 5$ e 7 os que obtiveram a maior quantidade de *outliers*, possivelmente devido ao baixo valor em sua mediana.

3.2.2. Gráficos de Estimação de Densidade por *Kernel*

Outro tipo de gráfico utilizado nesse estudo foi o gráfico de Estimação de Densidade por *Kernel* (KDE). Considerando a alta presença de *outliers* vista na Figura 2 e os baixos valores de média e mediana da Tabela 2, o atributo 7 (Fwd_IAT_Std) foi escolhido para visualizar qual dos protocolos está influenciando mais esses valores.

Na Figura 3, cada classe, ou protocolo identificado por uma cor, configura uma das curvas do gráfico. Nesse tipo de gráfico é estimada, através da distribuição de *kernels*, a densidade de probabilidade dos dados presentes no *dataset*. Quanto maior o valor da densidade indica que mais dados estão presentes naquele intervalo.

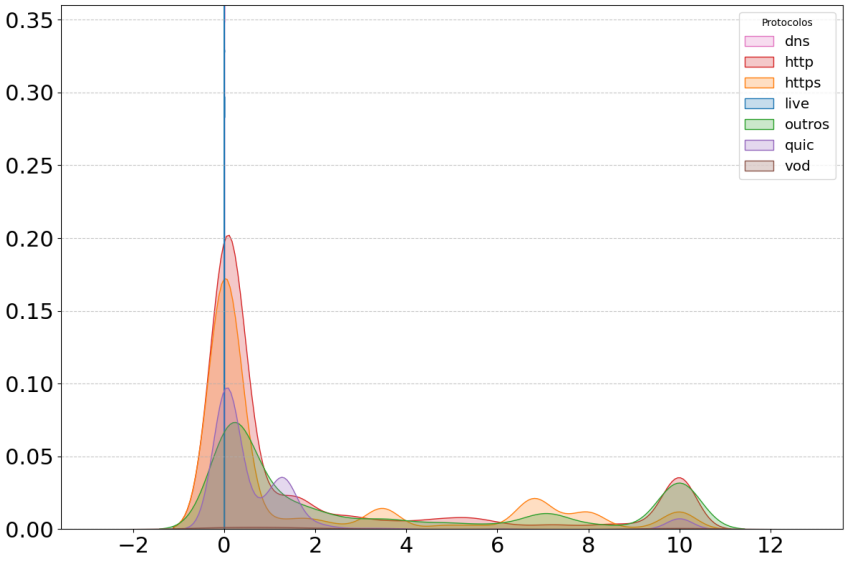

Figura 3. Estimação de Densidade por *Kernel* do atributo 7 – Fwd_IAT_Std.

Com a Figura 3, visualiza-se que o principal protocolo que está afetando os valores dessa variável é o protocolo ”Live”, seguido da classe ”DNS”. É interessante notar que exceto esses dois protocolos, os outros estão muito distribuídos por todo o *dataset*. Isso indica que, considerando a variável analisada, esses dois protocolos possuem um baixo desvio padrão entre seus dados.

3.2.3. Gráficos de Dispersão

Gráficos de Dispersão representam como os dados comportam-se, de acordo com duas variáveis distintas, geralmente indicando uma certa relação entre as variáveis selecionadas. Nessa AED, utilizamos dois atributos que demonstram a diferença entre tamanhos

de pacotes para *upload* e *download* (atributos 0 e 2, respectivamente). A escolha deles deu-se principalmente pela diferença entre quantidade e dispersão de *outliers* presentes. Além disso, uma possível relação entre eles, por se tratarem de direções de fluxo de dados opostas, para uma mesma unidade de medida, no caso, o tamanho de pacotes.

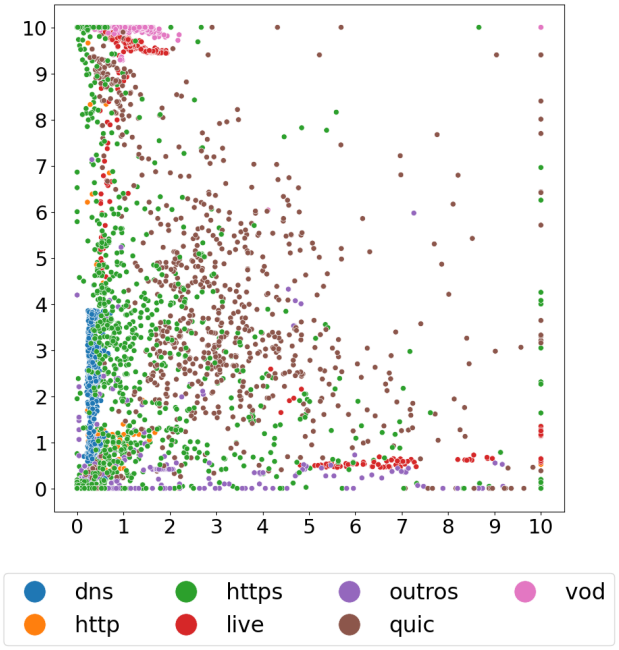

Figura 4. Gráfico de Dispersão da Média de Tamanho de Pacote entre o fluxo de *Upload* (Eixo X) e o fluxo de *Download* (Eixo Y).

Observando a Figura 4, fica claro que a maioria dos valores nesses atributos está próxima aos valores mínimo e máximo (0 e 10, respectivamente). Entretanto, observando apenas os protocolos ”Live” e ”Vod” (*Video on Demand*), é visível que eles possuem os valores mais altos em relação a todos os outros protocolos. Outro fator relevante é que, apesar da ideia inicial de que eles teriam algum tipo de relação, esta hipótese foi invalidada considerando que os dados estão muito dispersos.

3.2.4. Testes de Hipótese

A etapa final desta AED foi a aplicação de testes de hipóteses. Nesse estágio são explorados aspectos fundamentais que devem ser analisados em qualquer *dataset*, como a identificação do tipo de distribuição em que os dados estão, sendo ela a distribuição normal (Distribuição Gaussiana). Com as informações desse teste de distribuição, utilizamos um método de correlação de variáveis para preparação do *dataset* para as próximas etapas.

Foi aplicado o teste *Anderson-Darling* [Nelson 1998] de hipótese para entendimento da distribuição de dados devido a sua performance em *datasets* de larga escala e seu intenso uso por pesquisadores da área. Esse teste baseia-se no uso de duas hipóteses: A hipótese nula ($H0$), que considera que o *dataset* segue a distribuição normal, e a hipótese alternativa ($H1$), que considera que o *dataset* não segue a distribuição normal.

Utilizando da biblioteca SciPy e considerando um nível de significância de 0.05,

o resultado obtido foi muito maior do que o valor crítico para aceitação. Logo, a $H0$ foi completamente rejeitada. Logo, qualquer outro teste a ser aplicado nesse *dataset* necessita ser um teste que lide com uma distribuição não-normal.

Na etapa final nessa AED, foi aplicado um teste de correlação de variáveis visando entender melhor como cada variável interage umas com as outras. Considerando o resultado do teste de distribuição, o coeficiente de correlação de *Spearman* [Spearman 1961] foi o escolhido para realizar essa tarefa. O coeficiente de *Spearman* é usado para entender se duas variáveis diferentes possuem uma relação monotônica, atribuindo valores de -1 a 1. O valor -1 significa que as variáveis são inversamente relacionadas, 1 diretamente relacionadas e 0, não relacionadas.

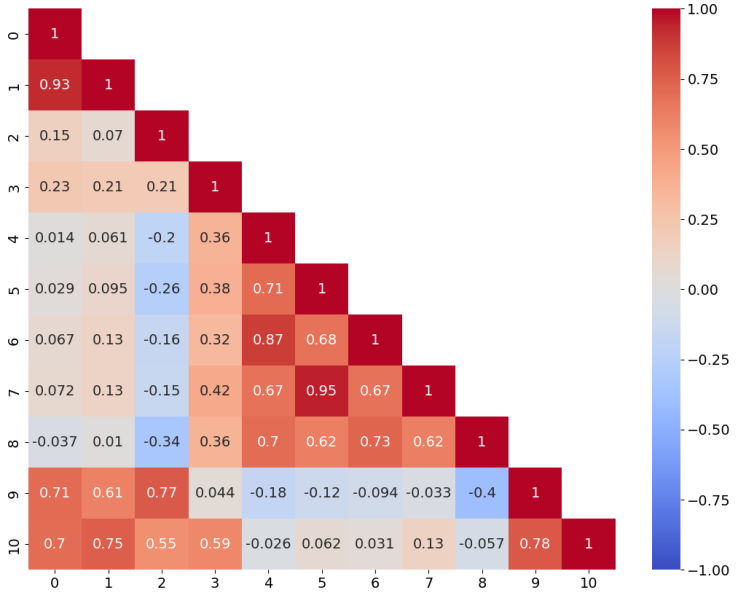

Figura 5. Matriz do Coeficiente de Correlação de *Spearman*.

Com essa matriz de correlação, é possível ratificar análises anteriormente realizadas. Um dos pontos é a falta de relação entre as variáveis (atributos) 0 e 2 (confira a Tabela 1) que possuem uma baixa relação entre si como foi visto na Figura 4. Por fim, observa-se que o atributo 9 possui uma alta relação com os três primeiros (0, 1 e 2) e o atributo 10, possui uma alta relação com os atributos 0, 1, 2 e 3, possivelmente por terem a mesma medida de tamanho de pacotes.

4. Conclusão

A problemática de classificação em *streaming* de vídeos é uma área de estudo que tem avançado cada vez mais, com o alto crescimento de usuários na rede. Esse crescimento exponencial, em conjunto com a criação de variados métodos para segurança de dados, necessita de conjuntos de dados que sejam precisos e robustos.

Neste trabalho o conjunto de dados *StreamDataNetClass* foi explorado, utilizando todas as etapas necessárias em uma AED, apresentando variados gráficos e valores estatísticos que puderam auxiliar na compreensão desse *dataset*.

Com os resultados obtidos e discutidos por todo o artigo é notável que esse *dataset* possui dados recentes e consistentes. Além disso, os quatro *datasets* originais, comenta-

dos na Seção 2, foram construídos de maneira a simular um ambiente real de tráfego de rede, algo que se mostrou presente na distribuição dos dados por todo o *dataset*.

Outro ponto que corrobora com o uso desse *dataset* para treino de modelos de classificação é o fato dos valores estarem todos concentrados dentro do intervalo $[0, 10]$. E estar nessa faixa de valores auxilia no treinamento e pré-processamento de dados antes da aplicação dos modelos de IA.

Como próximos passos, para além dessa AED, é necessária a realização de técnicas de seleção de atributos para filtragem dos atributos mais relevantes, além da otimização e pré-processamento nos dados do *dataset* para aplicação em variados modelos de IA.

Agradecimentos

Esta pesquisa foi parcialmente financiada pelas seguintes agências de fomento: CAPES (88887.158560/2025-00), CNPq (309559/2022-7, 409696/2022-6), FAPERGS (21/2551-0002057-1, 24/2551-0000631-1, 24/2551-0001396-2) e FAPERGS/CNPq (23/ 2551-0000126-8).

Referências

Banihashemi, S. B. and Akhtarkavan, E. (2022). Encrypted network traffic classification using deep learning method. In *2022 8th International Conference on Web Research (ICWR)*, pages 1–8.

Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I., and A. Ghorbani, A. (2016). Characterization of encrypted and vpn traffic using time-related features. In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy - ICISSP*, pages 407–414. INSTICC, SciTePress.

Fu, W. (2024). Exploratory data analysis and machine learning models for stroke prediction. In *Proceedings of the 1st International Conference on Data Analysis and Machine Learning - DAML*, pages 211–217. INSTICC, SciTePress.

Hattak, A., Iadarola, G., Martinelli, F., Mercaldo, F., and Santone, A. (2023). A method for robust and explainable image-based network traffic classification with deep learning. In *Proceedings of the 20th International Conference on Security and Cryptography - SECRYPT*, pages 385–393. INSTICC, SciTePress.

Komorowski, M., Marshall, D. C., Salciccioli, J. D., and Crutain, Y. (2016). Exploratory data analysis. *Secondary analysis of electronic health records*, pages 185–203.

Li, H., Xue, T., Zhang, A., Luo, X., Kong, L., and Huang, G. (2024). The application and impact of artificial intelligence technology in graphic design: A critical interpretive synthesis. *Heliyon*.

M. De Resende, A. A., D. De Melo, P. H. A., Souza, J. R., Cattelan, R. G., and Miani, R. S. (2022). Traffic classification of home network devices using supervised learning. In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence - Volume 3: ICAART*, pages 114–120. INSTICC, SciTePress.

Monks, E. M. (2023). *Abordagem Híbrida FuzzyNetClass: Uma Contribuição à Classificação do Tráfego de Streaming de Vídeo Integrando Lógica Fuzzy Valorada*

Intervalarmente e Aprendizagem de Máquina. Phd tese, Universidade Federal de Pelotas, Pelotas, RS.

Monks, E. M., Moura, B., Scheneider, G. B., Yamin, A. C., Reiser, R. H. S., and Santos, H. (2022). Abordagem fuzzy valorada intervalarmente para classificação de tráfego de streaming de vídeo. In *Anais do XLIX Seminário Integrado de Software e Hardware*, pages 70–81. SBC.

Nelson, L. S. (1998). The anderson-darling test for normality. *Journal of Quality Technology*, 30(3):298–299.

Qiu, T., Chen, N., Li, K., Atiquzzaman, M., and Zhao, W. (2018). How can heterogeneous internet of things build our future: A survey. *IEEE Communications Surveys & Tutorials*, 20(3):2011–2027.

Rao, A., Legout, A., Lim, Y.-s., Towsley, D., Barakat, C., and Dabbous, W. (2011). Network characteristics of video streaming traffic. In *Proceedings of the Seventh COnference on Emerging Networking EXperiments and Technologies*, CoNEXT '11, New York, NY, USA. Association for Computing Machinery.

Raparthi, M., Gayam, S. R., Kasaraneni, B. P., Kondapaka, K. K., Putha, S., Pattyam, S. P., Thuniki, P., Kuna, S. S., Nimmagadda, V. S. P., and Sahu, M. K. (2024). Exploratory data analysis techniques-a comprehensive review: Reviewing various exploratory data analysis techniques and their applications in uncovering insights from raw data. *Australian Journal of Machine Learning Research & Applications*, 4(1):215–225.

Sahoo, K., Samal, A. K., Pramanik, J., and Pani, S. K. (2019). Exploratory data analysis using python. *International Journal of Innovative Technology and Exploring Engineering*, 8(12):4727–4735.

Salman, O., Elhajj, I. H., Kayssi, A., and Chehab, A. (2020). A review on machine learning–based approaches for internet traffic classification. *Annals of Telecommunications*, 75(11):673–710.

Spearman, C. (1961). The proof and measurement of association between two things. *The American journal of psychology*.

Ventocilla, E. and Riveiro, M. (2019). Visual growing neural gas for exploratory data analysis. In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019) - IVAPP*, pages 58–71. INSTICC, SciTePress.

Wang, C., Li, T., Lu, Z., Wang, Z., Alballa, T., Alhabeeb, S. A., Albely, M. S., and Khalifa, H. A. E.-W. (2025). Application of artificial intelligence for feature engineering in education sector and learning science. *Alexandria Engineering Journal*, 110:108–115.

Zadeh, L. A. (1988). Fuzzy logic. *Computer*, 21(4):83–93.

Zhang, H., Ding, Y., Niu, J., and Jung, S. (2024). How artificial intelligence affects international industrial transfer—evidence from industrial robot application. *Journal of Asian Economics*, 95:101815.