

Predição do Número de Alunos por Municípios do Projeto “Saúde com Agente” Utilizando Redes Neurais de Grafos

Letícia M. Puttlitz¹

Programa de Pós-Graduação em Ciência da Computação, USP, São Paulo, SP

Márcia H. Barbian²

Programa de Pós-Graduação em Estatística, UFRGS, Porto Alegre, RS

Fabiana S. Pires³

Programa de Pós Graduação de Ensino na Saúde, UFRGS, Porto Alegre, RS

Leandro Raizer⁴

Departamento de Ensino e Currículo, UFRGS, Porto Alegre, RS

Daniela R. Knauth⁵, Luciana B. Teixeira⁶

Programa de Pós Graduação em Epidemiologia, UFRGS, Porto Alegre, RS

Resumo. A previsão de dados espaciais tem ganhado crescente importância em diversas áreas do conhecimento. Nesse contexto, modelos baseados em redes neurais e aprendizado profundo emergem como opções viáveis para realizar tais previsões. Destaca-se o papel das redes neurais de grafos (GNN), as quais são exploradas neste estudo para capturar a dependência espacial. Isso ocorre por meio da agregação realizada entre regiões vizinhas, fundamentada na tendência de regiões próximas de apresentarem comportamentos semelhantes. O foco deste trabalho envolveu a exploração de diferentes configurações de GNNs aplicadas a dados espaciais de área, com o objetivo de prever o número de alunos do projeto “Saúde com Agente” por município, considerando informações demográficas específicas. Os resultados indicaram desempenho superior dos modelos que utilizam GNN em comparação ao MLP (Rede Neural de Multicamadas) que considera as observações independentes. Além disso, observou-se uma melhoria na qualidade de predição quando camadas lineares foram adicionadas ao modelo GraphConv, para o modelo GENConv não observou-se a mesma tendência. Os modelos são particularmente sensíveis à inicialização dos pesos, essa variabilidade é evidenciada pelo alto desvio padrão do MSE (Erro Quadrático Médio) para o GraphConv e GENConv.

Palavras-chave. Estatística Espacial, Dados de área, Aprendizado de Máquina, Inteligência Artificial, GNN.

1 Introdução

A predição de dados espaciais tem ganhado crescente relevância em diversos campos do conhecimento, abrangendo áreas como meteorologia, análise de tráfego veicular e saúde.

A estrutura de dependência é uma característica fundamental na análise de dados espaciais e pode se manifestar de diferentes maneiras [1]. Por exemplo, dados de área se referem a informações associadas a dados agregados que são coletados em unidades disjuntas, como por exemplo municípios ou bairros.

Neste contexto, as Redes Neurais de Grafos (GNN) surgem como uma abordagem relevante, as GNN realizam agrupamentos das informações das áreas vizinhas para realizar a predição de uma

¹leticia.puttlitz@ime.usp.br

²mhbarbian@ufrgs.br

³fabianaspires@gmail.com

⁴leandroraiser@ufrgs.br

⁵daniela.knauth@gmail.com

⁶luciana.bteixeira@gmail.com

determinada área, capturando assim a dependência espacial. Isso se justifica pela tendência de áreas próximas apresentarem comportamentos semelhantes.

A adoção das GNN em contraposição a outros modelos estatísticos, como o campo aleatório markoviano gaussiano é motivada pela habilidade das GNNs em enfrentar desafios preditivos associados a um grande número de covariáveis na análise. O conjunto de variáveis preditoras pode apresentar características, como multicolinearidade e confundimento espacial, que o modelo *Conditional Autoregressive* (CAR) não trata de maneira eficiente, devido à complexidade intrínseca desses dados. Dessa forma, torna-se crucial buscar metodologias que não apenas capturem a estrutura espacial, mas também sejam mais flexíveis em relação a essas questões.

Nesse contexto, o objetivo principal deste trabalho foi realizar a predição do número de alunos por município no projeto “Saúde com Agente”. O projeto foi uma colaboração entre a Universidade Federal do Rio Grande do Sul (UFRGS) e o Ministério da Saúde, juntamente com o Conselho Nacional de Secretarias municipais de Saúde (Conasems), destinado a oferecer os cursos de Agente Comunitário de Saúde (ACS) e de Vigilância em Saúde com Ênfase no Combate às Endemias (ACE). Os cursos de ACE e ACS foram oferecidos para agentes de saúde de todo o Brasil, atendendo um total de **235.339 alunos**. Cada agente de saúde que foi aluno dos cursos de ACE ou ACS foi alocado ao município no qual estava vinculada as suas atividades profissionais, logo o projeto poderia atender 5.570 municípios, destes, **5.226 municípios** tiveram pelo menos um aluno matriculado.

Uma hipótese de pesquisa foi avaliar se municípios com características demográficas similares possuíam uma quantidade similar de agentes de saúde participando do projeto. Outra característica considerada foi a geográfica, visto que áreas mais distantes e de difícil acesso poderiam possuir uma quantidade menor de alunos. Logo, se municípios similares dado essas características demográficas e geográficas apresentassem uma quantidade de alunos muito diferente da predita, pode ser um indício de que naquele local, não houve uma divulgação apropriada do projeto, no caso do valor ser superestimado, ou que houve muita divulgação do projeto, se o valor predito foi subestimado.

Na predição do número de alunos por município no projeto “Saúde com Agente” foram consideradas características espaciais e covariáveis como número de habitantes, número de estabelecimentos de saúde, IDH, entre outras (disponíveis em fontes como IBGE, DATASUS e RAIS) através do modelo GNN. Como objetivos secundários avaliou-se a eficácia da abordagem das GNNs em comparação com um modelo tabular, que considera as observações independentes para realizar essas previsões. Além disso, buscou-se analisar a eficácia da predição para diferentes inicializações de pesos, a fim de verificar se eventuais melhorias não foram resultado de variações aleatórias. Adicionalmente, foi conduzida uma comparação em termos de desempenho entre modelos que empregaram diferentes camadas de GNN. A pesquisa abordou a avaliação do impacto da inclusão de camadas lineares no final do modelo, com o objetivo de determinar se essa alteração proporcionou um aumento substancial no desempenho preditivo.

2 Representação de Dados Espaciais via Grafos

A representação de dados espaciais por meio de grafos segundo [5] é uma abordagem para capturar e explorar efetivamente a estrutura e as relações espaciais inerentes aos dados. Nesse contexto, as unidades espaciais são modeladas como nós, enquanto as relações espaciais entre elas são representadas por arestas no grafo. A presença de uma aresta entre dois nós indica que as unidades espaciais correspondentes são vizinhas, estabelecendo uma conexão. Vale destacar que o grafo é necessariamente não direcionado. Isso significa que, se uma área é considerada vizinha de outra, a recíproca também é verdadeira.

A matriz de adjacência pode ser considerada uma representação tabular de um grafo, fornecendo uma visão das relações entre os vértices. Seja um grafo $G(V, E)$ com N vértices, a matriz de adjacência \mathbf{A} será uma matriz quadrada de ordem $N \times N$, em que cada elemento A_{vu} representa

a presença ou ausência de uma aresta entre os vértices v e u . Matematicamente, a matriz de adjacência é representada por:

$$A_{vu} = \begin{cases} 1, & \text{se } u \in N(v), \\ 0, & \text{caso contrário} \end{cases}$$

em que $N(v)$ representa a vizinhança do nó v .

Por exemplo, ao transformar os municípios pertencentes aos arredores de Porto Alegre (RS) em uma representação de grafo, pode-se gerar uma visualização semelhante à imagem apresentada na Figura 1.



Figura 1: Exemplo de mapa da região metropolitana de Porto Alegre como um grafo, em que cada círculo no centroide da cidade representa um nó e as linhas brancas representam as arestas. Fonte: Elaborado pelos autores.

A construção da matriz de adjacência em dados espaciais envolve a identificação das relações espaciais entre os elementos. Essa medida de proximidade entre dois polígonos pode ser quantificada de forma binária, determinando se esses elementos compartilham ou não uma fronteira, isto é, se são adjacentes.

Seja \mathbf{A} uma matriz de ordem $N \times N$ que representa a vizinhança entre N áreas, podemos representar essa estrutura através da adjacência entre as áreas:

$$A_{ij} = \begin{cases} 1, & \text{se os polígonos } i \text{ e } j \text{ compartilham fronteiras,} \\ 0, & \text{caso contrário} \end{cases} \quad (1)$$

Existem muitas medidas de proximidade diferentes que podem ser usadas para definir a proximidade entre duas áreas, não apenas a definida na Equação (1). Como ser ponderada de acordo com o tamanho da população.

3 Redes Neurais para Grafos (GNNs)

Redes neurais convencionais são eficazes na modelagem de dados tabulares, mas apresentam limitações ao lidar com estruturas mais complexas, como grafos. Nesse contexto, as Redes Neurais de Grafos (GNNs) [7] surgem como uma ferramenta para capturar informações em dados estruturados. A peculiaridade das GNNs reside na habilidade de propagar e agregar conhecimento ao longo das arestas do grafo, incorporando as interações entre nodos vizinhos.

Essa abordagem dinâmica das GNNs tem aplicações amplas, desde classificação e regressão de nós em grafos até previsão de links. Seu impacto estende-se além do domínio da ciência da computação, abrangendo áreas diversas como na descoberta de antibióticos, simulações físicas e sistemas de recomendação.

Existem três categorias principais de tarefas de previsão em grafos: previsão em nível de nó, em nível de aresta e em nível de grafo [6]. A previsão de nós em GNNs é empregada para

prever características específicas associadas a cada nó em um grafo. Na análise de sistemas de recomendação, por exemplo, a GNN pode ser utilizada para prever os interesses de um usuário com base em suas conexões e atividades passadas.

Este trabalho concentrou-se particularmente na predição de nós, explorando a capacidade das GNNs em modelar e extrair informações relevantes associadas aos nós em um grafo.

3.1 Rede Neural Convolutacional em Grafos (GCN)

Desenvolvida para operar em dados de grafos, a Rede Neural Convolutacional em Grafos (GCN) [2] é uma arquitetura que se destaca na tarefa de aprendizado em grafos.

A equação que descreve a propagação nas camadas da GCN é dada por:

$$\mathbf{H}^{(l)} = g \left(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l-1)} \mathbf{W}^{(l)} \right), \quad (2)$$

em que $g(\cdot)$ é uma função de ativação e $\tilde{\mathbf{A}}$ representa a matriz de adjacência \mathbf{A} de um grafo não direcionado somada à matriz identidade \mathbf{I} , isto é, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$. Essa adição assegura que cada nó leve em consideração não apenas o *embedding* de seus vizinhos, mas também o seu próprio *embedding* durante o processo de convolução. O termo $\tilde{\mathbf{D}}$ refere-se à matriz diagonal dos graus dos nós, presentes na matriz de adjacência $\tilde{\mathbf{A}}$. Matematicamente $\tilde{D}_{ii} = \sum_{j=1}^N \tilde{A}_{ij}$, em que N representa o número de vértices presente no grafo. $\mathbf{W}^{(l-1)}$ é uma matriz de pesos específica da camada $l-1$ e é aprendida ao longo do treinamento da GCN. Além disso, $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times D}$, em que D é a dimensão dos *embeddings* de cada nó.

3.2 GraphConv

O GraphConv [4] descreve a propagação de informações em grafos da seguinte maneira:

$$\mathbf{h}_v^{(l)} = g \left(\mathbf{B}^{(l)} \mathbf{h}_v^{(l-1)} + \text{AGG}^{(l)} \left(\mathbf{W}^{(l)} \mathbf{h}_u^{(l-1)} \right) \right), \quad (3)$$

em que v representa o nó alvo, e u são seus vizinhos, isto é, $u \in N(v)$. Além disso, a função $g(\cdot)$ caracteriza a função de ativação empregada. Os termos $\mathbf{B}^{(l)}$ e $\mathbf{W}^{(l)}$ são matrizes de pesos da camada l , multiplicando os *embeddings* do nó v , $\mathbf{h}_v^{(l-1)}$, e dos nós vizinhos u , $\mathbf{h}_u^{(l-1)}$, respectivamente. Ambos os conjuntos de pesos são aprendidos durante o treinamento. Além disso, $\text{AGG}^{(l)}$ representa uma operação de agregação, podendo assumir formas como soma, média e máximo, aplicada sobre os *embeddings* dos vizinhos u .

3.3 GENeralized Graph Convolution (GENConv)

O GENeralized Graph Convolution (GENConv) [3] opera de acordo com a seguinte formulação:

$$\alpha_{i,j} = \text{MLP} \left(\mathbf{h}_v^{(l-1)} + \text{AGG} \left(\left\{ \text{ReLU} \left(\mathbf{h}_u^{(l-1)} + \mathbf{e}_{uv}^{(l-1)} \right) \right\} \right) \right), \quad (4)$$

em que v representa o nó alvo, e u são seus vizinhos, isto é, $u \in N(v)$. Além disso, a função $g(\cdot)$ caracteriza a função de ativação empregada. Os *embeddings* do nó v são representados por $\mathbf{h}_v^{(l-1)}$, e dos nós vizinhos u , $\mathbf{h}_u^{(l-1)}$. O termo $\mathbf{e}_{uv}^{(l-1)}$ representa os *embeddings* da aresta entre os nós u e v , se houver. Além disso, $\text{MLP}(\cdot)$ representa uma rede neural com várias camadas de neurônios, organizadas em uma estrutura hierárquica.

4 Banco de Dados e Configuração dos Modelos

O Projeto “Saúde com Agente” foi uma colaboração entre a Universidade Federal do Rio Grande do Sul (UFRGS) e o Ministério da Saúde, juntamente com o Conasems (Conselho Nacional de Secretarias municipais de Saúde), destinado a oferecer os cursos de Agente Comunitário de Saúde

(ACS) e de Vigilância em Saúde com Ênfase no Combate às Endemias (ACE). Esses cursos foram ministrados em diversos estabelecimentos de saúde em todo o território brasileiro, o que gerou uma grande quantidade de dados espaciais. O programa disponibilizou os cursos ACE e ACS em todo o Brasil, embora em alguns casos a participação não tenha ocorrido devido à ausência de alunos interessados ou à falta de adesão ao projeto por parte de alguns municípios.

Os municípios nos quais esses cursos foram realizados passam a ser tratados como nós em um grafo, tendo uma aresta conectando dois municípios se eles compartilham fronteiras. A variável que o modelo GNN buscou prever foi o número de alunos em cada um desses municípios. Como *features* para os nós, características demográficas desses municípios foram consideradas, obtidas de fontes como o Instituto Brasileiro de Geografia e Estatística (IBGE), Departamento de Informática do Sistema Único de Saúde (DATASUS) e Relação Anual de Informações Sociais (RAIS).

A seleção das variáveis para compor as *features* de cada município priorizou aquelas que poderiam apresentar alguma relação com a variável resposta, a qual é representada pelo número de alunos. Nesse contexto, foram destacadas variáveis ligadas à saúde, como o número de estabelecimentos de saúde e de profissionais na área. Aspectos educacionais e socioeconômicos foram ponderados, uma vez que municípios com elevadas taxas de analfabetismo trazem restrições na participação nos cursos. Variáveis relacionadas ao rendimento médio também se mostraram relevantes, dado que poderiam influenciar na busca por qualificação profissional. Adicionalmente, variáveis mais abrangentes, como o Índice de Desenvolvimento Humano (IDH) e o Índice de Gini, foram incorporadas na análise, considerando seu potencial impacto sobre o número de alunos.

A implementação dos modelos foi realizada em Python versão 3.11.3. Dentre as bibliotecas empregadas, destacam-se: **torch_geometric (v2.4.0)**, **torch (v2.1.2)**, **numpy (v1.23.5)** e **pandas (v2.0.2)**.

4.1 Definição dos Hiperparâmetros

Os dados foram divididos em conjuntos de treinamento e teste por meio da abordagem transitiva, com 70% dos dados destinado ao treino e 30% ao teste, resultando em 3642 municípios para o conjunto de treino e 1561 para o teste.

A definição dos hiperparâmetros, envolvendo elementos como o número de camadas, a quantidade de neurônios e a taxa de aprendizagem, foi conduzida por meio de uma técnica conhecida como *Hyperopt*. A abordagem de treinamento em lote foi adotada para todos os modelos, ou seja, o conjunto de treino completo foi utilizado em cada iteração do algoritmo.

O modelo MLP, que trata as observações de forma independente, foi empregado para comparação com os modelos de GNN, com o objetivo de analisar se a incorporação de conexões melhorou a acurácia das previsões.

Foram analisadas diferentes arquiteturas de GNN, GraphConv e GENConv. Foram explorados modelos compostos unicamente por camadas de GNN, assim como modelos que integram camadas de GNN com camadas lineares. Vale destacar que uma camada linear executa uma transformação linear nas entradas, envolvendo a multiplicação das entradas pelos pesos e a adição de um *bias*. A semente foi fixada em 42 em todos os modelos, assegurando a reprodutibilidade dos resultados.

Todos os modelos foram treinados utilizando a função de perda MSE. O otimizador Adam foi empregado em todos os modelos, com as únicas variações ocorrendo nos hiperparâmetros associados à taxa de aprendizado e ao *Weight Decay*. A não-linearidade foi aplicada utilizando a função ReLU após cada camada.

Em nenhuma das configurações de camadas lineares, seja para modelos constituídos unicamente por camadas lineares (MLP) ou para modelos com GNN seguidas por camadas lineares, foi implementado o uso de *dropout*, um regulador que desativa aleatoriamente alguns neurônios durante o treinamento. Os hiperparâmetros adotados para o modelo que apresentou o melhor desempenho, o GENConv + linear, estão detalhados na Tabela 1.

Tabela 1: Valores considerados na otimização dos hiperparâmetros⁽¹⁾ para o modelo GENConv + Linear e os hiperparâmetros ótimos encontrados⁽²⁾. O tempo de execução foi de 108min.

Hiperparâmetro	Valores considerados na otimização ⁽¹⁾	Hiperparâmetro otimizado ⁽²⁾
Camadas (GENConv)	0, 1 ou 2	1
Neurônios (GENConv)	8, 16 ou 32	32
Taxa de aprendizagem	0,001 a 0,01	0,004
<i>Weight Decay</i>	0 a 0,1	0,08
Épocas	1.000 ou 2.000	2.000
Dropout	0 a 0,6	0,3
Camadas (linear)	1, 2 ou 3	2
Neurônios (linear)	8, 16 ou 32	8

4.2 Resultados

Os mapas das Figuras 2 e 3 apresentam os erros quadráticos dos modelos com os melhores desempenhos, as cores no mapa indicam a grandeza da diferença ao quadrado entre o número de alunos observado (y) e o número predito de alunos (\hat{y}) para cada município pertencente ao banco de teste.

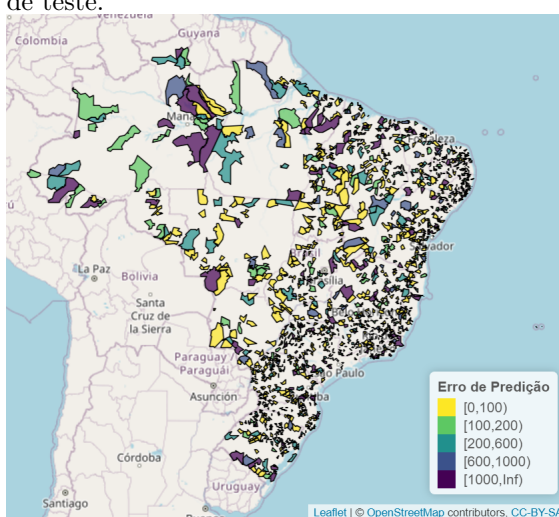


Figura 2: Mapa do erro MSE do modelo GENConv.

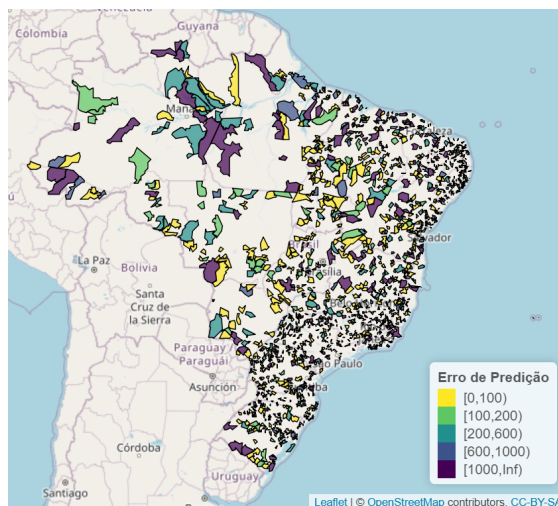


Figura 3: Mapa do erro MSE do modelo GENConv + Linear.

Fonte: Elaborado pelos autores

Tabela 2: O MSE é calculado no conjunto de teste para o respectivo modelo com semente igual a 42⁽¹⁾. MSE no conjunto de teste apresentado na forma de média (desvio padrão), obtido a partir de 30 simulações com diferentes sementes⁽²⁾. Tempo de execução das simulações em minutos⁽³⁾.

Modelo	MSE ⁽¹⁾	MSE ⁽²⁾	Tempo ⁽³⁾
MLP	1.945,48	3.679,94 (675,7)	3min
GraphConv	2.841,21	263.833,52 (479.655,1)	21min
GENConv	707,37	5.847,59 (7439,5)	59min
GraphConv + Linear	877,50	2.175,72 (1.092,1)	6min
GENConv + Linear	824,21	2.034,06 (792,1)	102min

Na Tabela 2, estão apresentados as configurações de hiperparâmetros dos modelos, que incluem o MSE no conjunto de teste utilizando a semente 42, seguido do MSE médio (com desvio padrão entre parênteses) obtido a partir de 30 simulações com diferentes sementes aleatórias, além do tempo total de execução das simulações. Essa abordagem visou analisar como a mesma configuração se comportou para diferentes sementes fixadas. Observa-se a presença de um desvio padrão

elevado em alguns modelos, sugerindo uma sensibilidade à inicialização dos pesos que é diretamente influenciada pela semente. Além disso, houve um aumento no erro médio calculado a partir das simulações, indicando que a melhor configuração encontrada com a semente 42 pode não ser uma escolha ótima para as demais sementes. A simulação foi feita com a configuração do modelo que gerou o melhor desempenho utilizando a semente 42.

5 Considerações Finais

Em conclusão, a pesquisa abordou a aplicação de diversas arquiteturas de GNN, como GraphConv e GenConv. Adicionalmente, foi conduzido um experimento com 30 diferentes sementes para avaliar a robustez e eficiência dos métodos GNN no contexto espacial. Os resultados revelaram que, embora a presença de uma observação atípica (Rio de Janeiro) tenha impacto no MSE para o modelo MLP, os modelos GENConv, GENConv + Linear e GraphConv + Linear demonstraram uma capacidade superior de lidar com esses *outliers*, apresentando um desempenho superior.

Alguns modelos são particularmente sensíveis à inicialização dos pesos. Como resultado, configurações idênticas com diferentes sementes resultaram em desempenhos distintos. Essa variabilidade é evidenciada pelo alto desvio padrão do MSE durante as simulações nos modelos GraphConv e GENConv. Observou-se uma melhoria geral no desempenho de todos os modelos GNN quando camadas lineares foram adicionadas, com exceção do modelo GENConv. Como perspectivas futuras, há diversas possibilidades a serem exploradas. Entre elas, destaca-se a investigação de previsões espaço-temporais utilizando modelos GNN.

Agradecimentos

Os dados apresentados são provenientes do Projeto Saúde com Agente, que tem como Coordenação Geral Prof. Dr. Leandro Raizer (Faculdade de Educação, UFRGS), Profa. Dra. Luciana Barcellos Teixeira (Escola de Enfermagem, UFRGS), Profa. Dra. Daniela Riva Knauth (Faculdade de Medicina, UFRGS) e Profa. Dra. Fabiana Schneider Pires (Faculdade de Odontologia, UFRGS). O Projeto teve o apoio financeiro do Ministério da Saúde e foi aprovado pelo Comitê de Ética e Pesquisa da UFRGS (CAAE 60867922.6.0000.5347).

Referências

- [1] N. Cressie, **Statistics for Spatial Data**. John Wiley e Sons, 2015.
- [2] T. N. Kipf e M. Welling, “Semi-supervised classification with graph convolutional networks,” **arXiv preprint arXiv:1609.02907**, 2016.
- [3] G. Li, C. Xiong, A. Thabet e B. Ghanem, “Deepergcn: All you need to train deeper gcn,” **arXiv preprint arXiv:2006.07739**, 2020.
- [4] C. Morris, M. Ritzert, M. Fey et al., “Weisfeiler and leman go neural: Higher-order graph neural networks,” em **Proceedings of the AAAI conference on artificial intelligence**, vol. 33, 2019, pp. 4602–4609.
- [5] J. Qian, V. Saligrama e Y. Chen, “Anomalous cluster detection,” em **2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**, IEEE, 2014, pp. 3854–3858.
- [6] B. Sanchez-Lengeling, E. Reif, A. Pearce e A. B. Wiltschko, “A Gentle Introduction to Graph Neural Networks,” **Distill**, 2021, <https://distill.pub/2021/gnn-intro>. DOI: 10.23915/distill.00033.
- [7] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner e G. Monfardini, “The graph neural network model,” **IEEE transactions on neural networks**, v. 20, n. 1, pp. 61–80, 2008.