

Explorando Redes Neurais de Grafos para Classificação de Arritmias

Rafael F. Oliveira¹, Vander L. S. Freitas¹, Gladston J. P. Moreira¹, Eduardo J. S. Luz¹

¹Departamento de Computação – Universidade Federal de Ouro Preto (UFOP)
Ouro Preto – 35.400-000 – MG – Brasil

rafael.fo@aluno.ufop.edu.br, {vander.freitas, gladston, eduluz}@ufop.edu.br

Abstract. *Cardiac arrhythmia is a risky condition and its early diagnosis is of paramount importance. Therefore, the automation process is desirable. In this context, the Graph Neural Network (GNN) is explored to classify arrhythmias in electrocardiogram (ECG) signals in which each heartbeat is transformed in a graph via the Visibility Graph (VG) algorithm. Three models of GNN networks are evaluated: a Graph Convolutional Network with two layers (GCN-2L), another with three layers (GCN-3L), and GraphSAGE (GraphSAGE-4L). Following the inter-patient paradigm, the GraphSAGE-4L model presented the best performance, obtaining an average F1-score of 0.86 for the classes N, S and V.*

Resumo. *Arritmia cardíaca é uma condição de risco e seu diagnóstico precoce é de fundamental importância. Por isso, a automação do processo de identificação de arritmia é desejável. Neste contexto, um modelo de classificação automática de arritmias em sinais de eletrocardiograma (ECG) é proposto baseado em Graph Neural Network (GNN) e o batimento cardíaco representado por meio de um grafo via algoritmo Visibility Graph (VG). São avaliados três modelos de redes GNN: uma rede convolucional de grafos com duas camadas (GCN-2L), outra com três camadas (GCN-3L) e GraphSAGE (GraphSAGE-4L). Sob o paradigma inter-patient, o modelo GraphSAGE-4L apresentou o melhor desempenho, obtendo F1-score médio de 0,86 para as classes N, S e V.*

1. Introdução

O principal exame para diagnóstico de doenças cardíacas é chamado Eletrocardiograma (ECG). Segundo [Cohen 1986], o ECG foi um dos primeiros sinais estudados pelas técnicas de processamento de sinais biomédicos e até os dias de hoje é a técnica mais utilizada no diagnóstico de doenças do coração. Isto é possível por causa da simplicidade do exame e por ser não-invasivo.

O processo de identificação e classificação de arritmias pode ser muito exaustivo para um ser humano e ainda sujeito a erros, visto que seria necessário uma análise de batimento a batimento em horas ou até mesmo dias de registros de ECG capturados [Luz et al. 2016]. Uma alternativa consiste no uso de inteligência artificial para classificação automática dos padrões de arritmia. Atualmente, métodos baseados em aprendizagem de máquina (*machine learning*), tornaram-se bastante populares para a tarefa [Hannun et al. 2019, Luz et al. 2016].

Para que soluções propostas possam ser utilizadas em monitores de sinais vitais ou desfibriladores, o projeto e análise de experimentos deve obedecer normas da *Association for the Advancement of Medical Instrumentation (AAMI)* [ANSI/AAMI 2008]. Manter a conformidade com todas as normas e boas práticas é ainda um grande obstáculo para a introdução de novas técnicas na indústria. As abordagens publicadas na literatura que não seguem estes requisitos não apresentam grande valor prático para fabricantes de equipamentos médicos [Luz et al. 2016]. Logo, o trabalho propõe o uso de boas práticas da AAMI para a construção da metodologia proposta.

Neste trabalho, propõe-se um modelo de classificação automática de arritmias em sinais de ECG baseado em Redes Neurais de Grafos (*Graph Neural Network - GNN*). A metodologia proposta inova por meio da técnica de mapeamento (conversão) de um batimento cardíaco (um canal) em grafo, denominada *Visibility Graph (VG)* [Lacasa et al. 2008]. São investigados três modelos de redes GNNs: *Graph Convolutional Network com duas camadas (GCN-2L)*, *Graph Convolutional Network com três camadas (GCN-3L)* e *GraphSAGE (GraphSAGE-4L)*. O número que vem antes da letra “L” representa o número de camadas (*layers*) utilizadas na construção de cada modelo. Propõe-se aqui uma análise comparativa entre suas performances. Os experimentos foram validados utilizando o conjunto de dados PhysioNet MIT-BIH por meio do paradigma *inter-patient*.

O trabalho é organizado como segue. A Seção 2 apresenta uma breve revisão dos trabalhos relacionados e a Seção 3 descreve brevemente a principal técnica investigada aqui: GNN. A Seção 4 detalha o conjunto de dados utilizado na experimentação. A metodologia proposta é descrita na Seção 5. Além disso, na Seção 6, os experimentos, resultados e discussões são apresentados. Finalmente, as observações finais são reportadas na Seção 7.

2. Trabalhos relacionados

A detecção de arritmias com base na classificação de ECG tornou-se um tópico de pesquisa muito popular na literatura nos últimos anos [Luz et al. 2016]. Os métodos tradicionais utilizam pontos importantes do sinal de ECG, denominados de pontos fiduciais, como intervalo RR, coeficientes *wavelets*, amplitude, complexo QRS, onda P, onda Q, duração, etc. Dentre os pontos apresentados, temos o segmento QRS, que normalmente é utilizado durante o processo de segmentação de um sinal de ECG.

A identificação dos sinais ocorre a partir de características extraídas dos sinais de ECG. Com isso, diversas características e métodos têm sido propostos na literatura. Em [De Chazal et al. 2004], os autores combinaram características rítmicas, duração de onda e características morfológicas. Em [Zheng et al. 2020], os autores utilizaram mais de 200 características, tais como idade, gênero, características do canal II do ECG, além de características extraídas dos 12 canais (*leads*) presentes no conjunto de dados utilizado. Além disso, um método de seleção de características baseado em transformação *wavelet* com limiar adaptativo, visando reduzir o ruído do sinal de ECG é proposto em [Wang et al. 2018a].

A escolha do classificador ocorre de acordo com o problema a ser tratado. Na literatura há diversos modelos propostos, tais como SVM [Mondéjar-Guerra et al. 2019], random forest [Yang et al. 2021], redes neurais artificiais [Rajpurkar et al. 2017], AdaBoost [Rajesh and Dhuli 2018], redes neurais convolucionais profundas [Hannun et al. 2019]

e XGBoost [Shi et al. 2019]. Com isso, o estudo de novas metodologias de classificação de sinais de ECG torna-se tão importante. Neste contexto, este trabalho propõe uma nova abordagem para classificação de arritmias por meio do uso de GNN em um grafo de ECG gerado pela técnica VG.

3. Redes Neurais de Grafos

O modelo de redes neurais artificiais apresenta crescente interesse pela comunidade científica, principalmente as redes neurais convolucionais, pela sua capacidade de generalização e aplicações. Recentemente, as GNNs tem apresentado notoriedade na literatura [Wu et al. 2020] em diversos domínios, como química [Do et al. 2019], estudo social [Wang et al. 2018b] e biologia [Fout 2017].

O surgimento das GNNs deu-se nos anos 90 [Zhou et al. 2020] com as redes neurais recursivas [Sperduti and Starita 1997] utilizadas em grafos direcionados acíclicos. Posteriormente, redes neurais recorrentes [Scarselli et al. 2008] e redes neurais *feed-forward* [Micheli 2009] foram introduzidas para grafos cíclicos. Até esse momento, as redes eram construídas baseadas em sistemas de transição de estado em grafos e iteram até a convergência, o que restringia a capacidade de extensão e representação. Com o avanço das redes neurais profundas, em especial pelas redes neurais convolucionais profundas (CNN) [LeCun et al. 1998], as GNNs foram aprimoradas.

Em [Wu et al. 2020], os autores categorizaram as GNNs em quatro principais grupos: redes neurais de grafos recorrentes, redes neurais de grafos convolucionais, grafos *autoencoders* e redes neurais de grafos espaço-temporal. Neste trabalho, utiliza-se a rede neural de grafos convolucional.

4. Conjunto de dados

Em razão das propostas de modelos de classificação para o avanço da instrumentação médica [ANSI/AAMI 2008], foi utilizado o conjunto de dados de arritmias PhysioNet MIT-BIH¹ [Moody and Mark 1990] para treinamento e validação da metodologia proposta. O conjunto de dados do MIT-BIH é composto por 48 registros de sinais de ECG de 30 minutos, de 47 pacientes diferentes, amostrados à frequência de 360 Hz. Cada sinal contém duas derivações: derivação V e derivação II. A derivação II é a mais utilizada na literatura e também aqui.

Seguindo recomendações da AAMI, quatro registros (102, 104, 107 e 217) são excluídos do conjunto de dados por conterem batimentos artificiais. [De Chazal et al. 2004] recomenda a divisão do conjunto de dados em duas partes: treinamento (DS1) e teste (DS2), conforme a Tabela 1. Essa divisão faz parte do paradigma *inter-patient* utilizado neste trabalho, em que o conjunto de treinamento e teste apresentam registros diferentes (sem sobreposição de registros de pacientes).

Dentre os 18 tipos de batimentos, a AAMI também recomenda o agrupamento dos batimentos em cinco grupos principais: *Normal (N)*, *Supraventricular ectopic beat (S)*, *Ventricular ectopic beat (V)*, *Fusion beat (F)* e *Unknown beat (Q)*. A Tabela 2 resume o agrupamento dos batimentos dos dados do MIT-BIH após a exclusão dos registros com batimentos artificiais bem como o número de amostras do conjunto de treinamento e teste.

¹<https://physionet.org/content/mitdb/1.0.0/>

Tabela 1. Distribuição dos registros do MIT-BIH em treinamento e teste.

DS1 (Treinamento)	DS2 (Teste)
101, 106, 108, 109, 112, 114, 115, 116	100, 103, 105, 111, 113, 117, 121, 123
118, 119, 122, 124, 201, 203, 205	200, 202, 210, 212, 213, 214, 219
207, 208, 209, 215, 220, 223, 230	221, 222, 228, 231, 232, 233, 234

Os percentuais de batimentos nos conjuntos de treinamento e teste foram de 50,65% e 49,35%, respectivamente. Observa-se que as classes F e Q apresentam poucos batimentos, somando menos de 0,81% de presença em relação ao conjunto de dados total. Por esse motivo, as classes F e Q foram removidas desta análise. Os modelos são validados apenas nas classes N, S e V.

Tabela 2. Agrupamento de tipos de batimento no conjunto de dados do MIT-BIH.

Grupo AAMI	Batimentos	% do total	# Treinamento (DS1)	# Teste (DS2)	# Batimentos total
<i>Normal (N)</i>	N, e, j, L, R	89,47	45844	44238	90082
<i>Supraventricular ectopic beat (S)</i>	A, a, J, S	2,76	944	1837	2781
<i>Ventricular ectopic beat (V)</i>	V, E	6,96	3788	3220	7008
<i>Fusion beat (F)</i>	F	0,80	414	388	802
<i>Unknown beat (Q)</i>	P, f, U, Q	0,01	8	7	15
Total		100	50998	49690	100688

5. Metodologia

A seguir, apresenta-se a metodologia proposta para classificação automática de arritmias em ECG. O diagrama da Figura 1 mostra as etapas do sistema de classificação proposto: segmentação, balanceamento dos dados, divisão de dados, conversão em grafos e classificação.

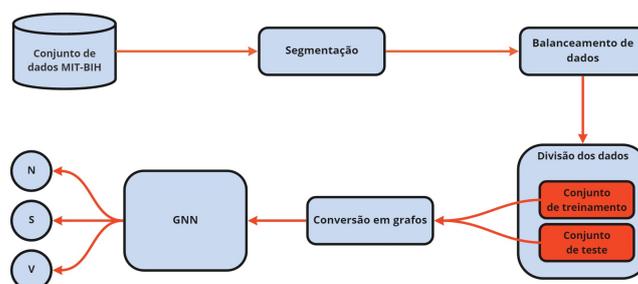


Figura 1. Diagrama do sistema de classificação de arritmia.

5.1. Segmentação

Antes da classificação, é necessária a segmentação do sinal ECG em batimentos individuais. Isto requer a identificação da onda QRS e pontos fiduciais do batimento. No entanto, vários métodos são desenvolvidos na literatura para detecção desses pontos fiduciais com altas acurácias. A Figura 2 ilustra o processo de segmentação. Neste processo de segmentação, utiliza-se a localização do pico R anotada no conjunto de dados do MIT-BIH, segmentando o sinal em séries de batimentos, extraindo-se 150 amostras antes e 150 amostras depois do pico R, e realizando o registro do tipo de batimento de acordo com as anotações do MIT-BIH. No final, cada batimento obtém um tamanho fixo de 300 amostras do sinal de ECG.

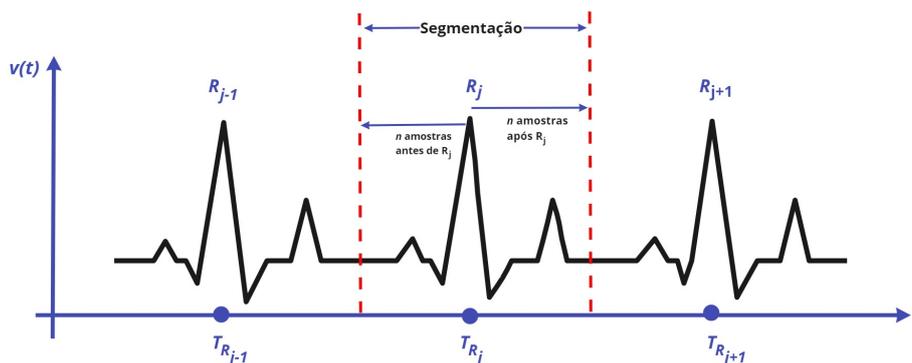


Figura 2. Exemplo de segmentação de sinal ECG. Fonte: Adaptada de [Wang et al. 2021].

5.2. Balanceamento de dados

O grande número de batimentos da classe N, conforme apresentado na Tabela 2, ocasiona um grande tempo de processamento devido ao alto número de grafos gerados, além do problema de desbalanceamento entre as classes. Por esse motivo, os batimentos da classe N foram sub-amostrados para 3000 batimentos, tanto no conjunto de treinamento (DS1) quanto no conjunto de teste (DS2) de forma aleatória. A Tabela 3 resume o conjunto de dados antes e após o balanceamento.

Tabela 3. Descrição da amostragem antes e depois do balanceamento de dados.

Batimentos	Treinamento (DS1)		Teste (DS2)	
	Antes	Depois	Antes	Depois
N	45844	3000	44238	3000
S	944	944	1837	1837
V	3788	3788	3220	3220
Total	50576	7732	49295	8057

5.3. Divisão do conjunto de dados

Os batimentos gerados na etapa de segmentação são divididos em conjunto de treinamento (DS1) e conjunto de teste (DS2), conforme apresentado na Tabela 1. Essa divisão proposta por [De Chazal et al. 2004] permite utilizar o paradigma *inter-patient* de classificação, onde os batimentos do conjunto de treinamento são diferentes do conjunto de teste, permitindo assim a simulação de um cenário do mundo real.

5.4. Representação dos sinais de ECG em grafos

A captura das amostras de uma derivação do sinal de ECG no tempo pode ser vista como uma série temporal de uma dimensão e esta série temporal é, então, transformada em grafo. A Figura 3 exemplifica a representação pelo método *visibility graph* [Lacasa et al. 2008]. Cada ponto da série origina um nó do grafo e dois nós são conectados se eles satisfizerem um critério de visibilidade. Considere dois pontos (t_a, y_a) e (t_b, y_b) , representados pelos nós a e b , onde t_a e t_b representam o tempo e y_a e y_b os valores associados. Eles serão conectados se

$$y_c < y_b + (y_a - y_b) \frac{t_b - t_c}{t_b - t_a} \quad (1)$$

para quaisquer pontos (t_c, y_c) . Em outras palavras, se cada ponto da série fosse uma barra vertical, em um gráfico de barras, duas barras a e b se conectam se não existe nenhuma barra c entre elas, cuja altura impeça que uma linha reta seja traçada entre os picos de a e b .

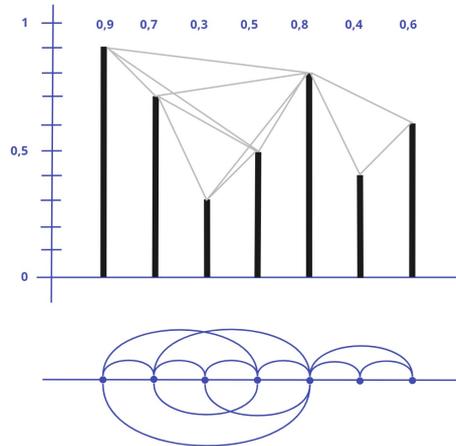


Figura 3. Exemplo de série temporal periódica (07 valores) e grafo derivado pelo método *visibility graph*. Fonte: Adaptada de [Lacasa et al. 2008].

A Figura 4 exemplifica a conversão de batimentos correspondente às classes **N**, **S** e **V**. Observa-se que cada batimento cardíaco em um sinal de ECG é convertido em um grafo, possibilitando a classificação pelas GNNs. O processo realiza a conversão de um batimento P em um grafo $G(V, E)$, em que V corresponde ao conjunto de vértices/nós e E corresponde ao conjunto de arestas/links do grafo G .

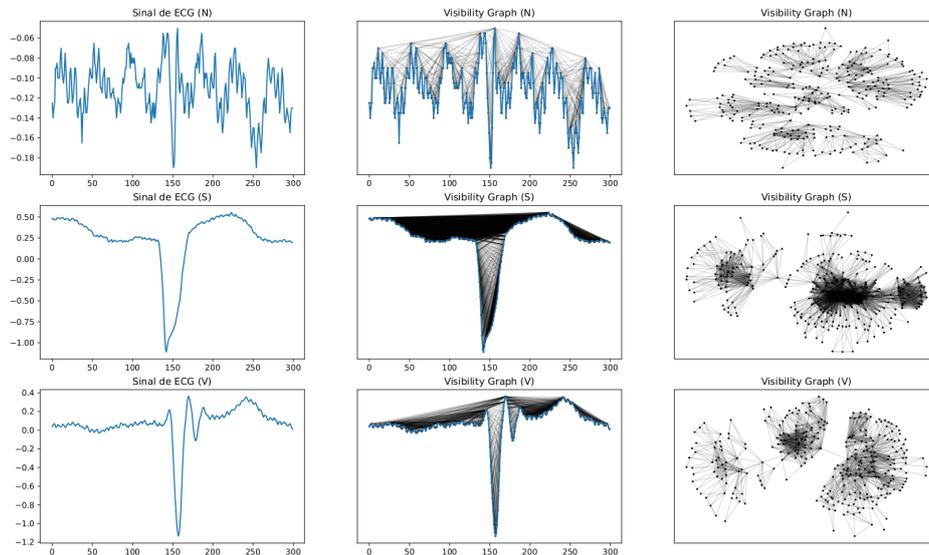


Figura 4. Mapeamento dos sinais de ECG das classes N, S e V utilizando o método *visibility graph*.

Neste trabalho, utilizou-se a abordagem de nível de grafo (*graph-level*) em que a classificação ocorre considerando o grafo por completo, como em [Kojima et al. 2020], em que utiliza-se a GCN na predição de proteínas por meio das estruturas moleculares. As

informações contidas nos nós e arestas são utilizadas durante o processo de classificação. Aqui, apenas duas informações foram configuradas em cada nó p do conjunto de nós V : o grau (número de arestas incidentes no respectivo nó) e *pagerank* (medida da importância do nó no seu respectivo grafo).

5.5. Classificação

A GCN é uma rede neural multicamada aplicada diretamente em grafos que utiliza as informações/propriedades da vizinhança de cada nó. Formalmente, considere um grafo como $G(V, E)$, onde V ($|V| = n$) é o conjunto de nós e E o conjunto de arestas. Cada nó é conectado consigo mesmo (*loop*), isto é, $(v, v) \in E$. Assumindo $X \in \mathbb{R}^{n \times m}$ como a matriz contendo todos os n nós com seus respectivos vetores de atributos, onde m é a dimensão deste vetor, cada linha $x_v \in \mathbb{R}^m$ é o vetor de atributos para v . Considere ainda a matriz de adjacência A de G e a matriz de graus D , onde $D_{ii} = \sum_j A_{ij}$. A diagonal principal de A contém valores 1 por causa dos *loops* de cada nó. Com apenas uma camada, a GCN pode capturar informações apenas dos vizinhos imediatos. Com múltiplas camadas, agrega-se informações de uma vizinhança de maior alcance. Uma GCN de várias camadas segue a seguinte regra de propagação em camadas [Kipf and Welling 2016]:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}), \quad (2)$$

sendo $\tilde{A} = A + I_n$ a matriz de adjacência do grafo com *loops*, I_n a matriz identidade, $\tilde{D} = \sum_j \tilde{A}_{ij}$ a matriz de graus, $W^{(l)}$ a matriz de pesos e σ a função de ativação, isto é, a ReLU. $H^{(l)} \in \mathbb{R}^{n \times m}$ a matriz de ativação da l -ésima camada e $H^{(0)} = X$.

Neste trabalho, três modelos de redes GNN são analisadas: *GCN-2L*, *GCN-3L* e *GraphSAGE-4L*. O modelo *GCN-2L* consiste em duas camadas de convolução de grafos sequenciais. O modelo *GCN-3L* difere do modelo *GCN-2L* apenas pela adição de mais uma camada de convolução. Por último, o modelo *GraphSAGE-4L* consiste em 04 camadas de convolução *GraphSAGE* (SAmple and aggreGatE) sequenciais (ver Tabela 4). Este último utiliza um conjunto de funções agregadoras que aprendem a agregar informações de recursos da vizinhança local de um nó e, à medida que esse processo se repete, os nós ganham cada vez mais informações de alcances mais distantes do grafo. *GraphSAGE* [Hamilton et al. 2017] é definido por

$$h_{\mathcal{N}(v)}^k \leftarrow \text{aggregate}(h_u^{k-1}, \forall u \in \mathcal{N}(v)) \quad (3)$$

$$h_v^k \leftarrow \sigma(\mathcal{W}^k \cdot \text{concat}(h_v^{k-1}, h_{\mathcal{N}(v)}^k)) \quad (4)$$

$$h_k^v \leftarrow \text{norm}(h_k^v, \forall u \in \mathcal{V}). \quad (5)$$

A intuição do *GraphSAGE* é que a informação agregada dos vizinhos locais de cada nó é incrementada cada vez mais, tornando o processo de classificação mais robusto a cada iteração. A cada iteração k , cada nó $v \in \mathcal{V}$ agrega informação dos vizinhos imediatos $h_u^{k-1}, \forall u \in \mathcal{N}(v)$ em um único vetor $h_{\mathcal{N}(v)}^{k-1}$, como em (3). Após a agregação, o *GraphSAGE* concatena a representação atual do nó h_v^{k-1} com a do vizinho $h_{\mathcal{N}(v)}^{k-1}$ aplicada a uma função de ativação σ (4), onde as transformações são usadas na próxima iteração por meio de h_v^k . Por fim, as informações são normalizadas (5) para a próxima iteração.

O processo de classificação de grafos pode ser realizado em três níveis: nós (*node-level*), arestas (*edge-level*) e grafos (*graph-level*), sendo a última utilizada neste trabalho. Para classificação de grafos, a função *readout* agrega atributos dos nós na última iteração para obter a inteira representação do grafo (h_G) [Xu et al. 2018]:

$$h_G = \text{readout}(\{h_v^{(k)} | v \in G\}), \quad (6)$$

sendo $h_v^{(k)}$ o vetor de atributos do nó v na k -ésima iteração/camada com $h_v^{(0)} = X$.

6. Resultados

Nesta seção, apresenta-se as configurações do experimentos, os resultados e suas discussões.

6.1. Configurações e métricas

Os experimentos foram conduzidos em um ambiente Windows 11 com ambiente virtual Anaconda Python 3.9 com Intel®i5-1035G1 com 04 cores (threads) de 1,2GHz, 8GB RAM DDR4 e GPU GeForce MX350 de 2GB de RAM.

Avalia-se o desempenho dos modelos de classificação propostos por meio de três métricas frequentemente utilizadas na literatura: predição positiva ($+P^k$), sensibilidade (Se^k) e F1-score (F_s^k), onde $k \in \{N, S, V\}$ indica a classe correspondente a cada métrica. As métricas são baseadas nas medidas de verdadeiros positivos (TP), falsos negativos (FN), verdadeiros negativos (TN) e falsos positivos (FP). As métricas ($+P^k$), (Se^k) e (F_s^k), são respectivamente: $+P^k = \frac{TP}{TP + FP} \times 100$, $Se^k = \frac{TP}{TP + FN} \times 100$, $F_s^k = 2 * \frac{Se^k * (+P^k)}{Se^k + (+P^k)}$

6.2. Resultados e discussões

As bibliotecas utilizadas na implementação das redes GNN são a *Deep Graph Library - DGL* [Wang et al. 2019] e o *PyTorch*². Para a converção dos batimentos em grafos por meio do VG utiliza-se a biblioteca *ts2vg*³.

Devido a presença de mais amostras no conjunto DS2 após o balanceamento, o conjunto DS2 é utilizado no treinamento e o conjunto DS1 no teste. Não utiliza-se conjunto de validação no treinamento. Os modelos são treinados em 100 épocas (*epochs*) utilizando o otimizador Adam com taxa de aprendizado de 0,01, função de perda *cross entropy* e função de ativação *ReLU (Rectified Linear Unit)*.

A Tabela 4 descreve a arquitetura de cada modelo de GCN, a quantidade de neurônios bem como o tipo de cada camada. É importante observar que nas redes GNN da DGL, a camada de entrada tem como argumento o tamanho do vetor de atributos de cada nó, ou seja, informações que podem ser adicionadas para agregar valor ao grafo. Utiliza-se aqui os atributos *grau* e *pagerank*, portanto um vetor de tamanho 2. A camada de saída possui 03 neurônios correspondentes às três classes (N, S, V).

²<https://pytorch.org/>

³<https://pypi.org/project/ts2vg/>

Tabela 4. Arquitetura de cada modelo de rede GCN.

GCN-2L			GCN-3L			GraphSAGE-4L		
No	Camada	Formato	No	Camada	Formato	No	Camada	Formato
1	GraphConv	2×20	1	GraphConv	2×20	1	SAGEConv	2×30
2	GraphConv	20×3	2	GraphConv	20×20	2	SAGEConv	30×20
3	Readout	-	3	GraphConv	20×3	3	SAGEConv	20×5
4	Softmax	-	4	Readout	-	4	SAGEConv	5×3
			5	Softmax	-	5	Readout	-
						6	Softmax	-

A Tabela 5 apresenta os resultados das matrizes de confusão para os modelos propostos em relação ao conjunto DS1 (teste). Nota-se que apesar da arquitetura dos modelos serem mais simples, a performance é bastante promissora. O modelo *GraphSAGE-4L* apresenta a melhor taxa de classificação em relação às classes **N** e **V**, 99,84% e 85,22%, respectivamente. Para a classe **S**, o modelo *GCN-2L* apresenta a melhor taxa de classificação, 86,33%. Pelo desempenho obtido, observa-se que a classe **S** apresenta o maior desafio de classificação para os modelos propostos. Ainda, a Tabela 6 mostra os resultados das métricas de avaliação para cada modelo proposto. Observa-se que o modelo *GraphSAGE-4L* apresenta os melhores resultados, com uma média ponderada em $+P$, Se e F_s de 0,86, sendo superior aos outros modelos.

Tabela 5. Matrizes de confusão do conjunto de teste (DS1).

		GCN-2L					GCN-3L					GraphSAGE-4L		
		Predição					Predição					Predição		
		N	S	V			N	S	V			N	S	V
Verdade	N	2161	475	364	Verdade	N	2343	157	500	Verdade	N	2995	2	3
	S	25	815	104		S	14	655	275		S	13	435	496
	V	84	1823	1881		V	26	1356	2406		V	28	532	3228

Tabela 6. Performance dos modelos em relação ao conjunto de teste (DS1).

Grupos	GCN-2L			GCN-3L			GraphSAGE-4L		
	+P	Se	F_s	+P	Se	F_s	+P	Se	F_s
N	0,95	0,72	0,82	0,98	0,78	0,87	0,99	1,00	0,99
S	0,26	0,86	0,40	0,30	0,69	0,42	0,45	0,46	0,45
V	0,80	0,50	0,61	0,76	0,64	0,69	0,87	0,85	0,86
Acc			0,63			0,70			0,86
Média Macro	0,67	0,69	0,61	0,68	0,70	0,66	0,77	0,77	0,77
Média Ponderada	0,79	0,63	0,67	0,79	0,70	0,73	0,86	0,86	0,86

As Figuras 5 e 6 mostram as curvas de desempenho da acurácia e da função perda (*loss*) ao longo de 100 épocas de treinamento. Vale ressaltar, que apesar da Figura 6 mostrar uma tendência de queda na função de perda, o aumento no número de épocas ocasiona a diminuição nos desempenhos dos modelos. Novamente, percebe-se que o modelo *GraphSAGE-4L* apresenta os melhores resultados em relação aos outros modelos, tanto na acurácia ($0,86 \pm 0,058$) quanto na função de perda ($0,19 \pm 0,13$). Esse desempenho pode

ser justificado pela agregação e concatenação de informações dos nós utilizada pelo modelo a cada iteração. Portanto, o objetivo não é realizar uma comparação com o estado-da-arte da literatura, mas sim, investigar a viabilidade de classificação de arritmias mediante a representação de um batimento em forma de grafo utilizando VG e por meio de GNNs.

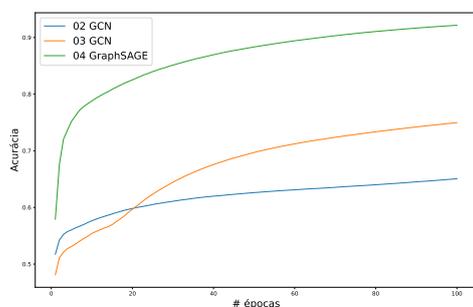


Figura 5. Curva da acurácia ao longo de 100 épocas.

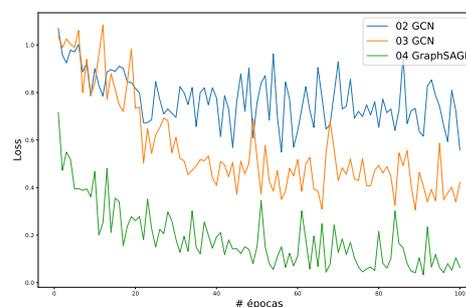


Figura 6. Curva da função de perda ao longo de 100 épocas.

7. Conclusões

Este trabalho propõe um modelo de classificação automática de arritmias em sinais de ECG baseado em Redes Neurais de Grafos (*Graph Neural Network - GNN*) por meio do mapeamento dos sinais de ECGs em grafos e classificação a nível de grafo (*graph-level*). Investiga-se três modelos de redes GNNs: *GCN-2L*, *GCN-3L* e *GraphSAGE-4L* por meio do paradigma *inter-patient*.

O modelo *GraphSAGE-4L* é superior aos outros modelos, com F_5 médio de 0,86 para as classes **N**, **S** e **V**. O desempenho pode ser justificado pela agregação e concatenação de informações dos nós utilizada pelo modelo *GraphSAGE-4L* a cada iteração. Os resultados ainda mostram que os modelos propostos apresentam baixo desempenho de classificação para a classe **S**, deixando assim espaço para novas propostas de melhorias.

A utilização de modelos baseados em GNNs apresenta potencial como novo método de classificação de arritmias em sinais de ECG. Algumas perspectivas de continuidade são: investigar novas arquiteturas de GNNs e adição de informações extrínsecas ao grafo.

Agradecimentos

Os autores agradecem a *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Financiamento código 001*, *Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG, APQ-01518-21)*, *Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq)* e *Universidade Federal de Ouro Preto (UFOP)* pelo suporte no desenvolvimento do presente trabalho.

Referências

ANSI/AAMI (2008). Testing and reporting performance results of cardiac rhythm and ST segment measurement algorithms. American National Standards Institute, Inc. (ANSI), Association for the Advancement of Medical Instrumentation (AAMI). ANSI/AAMI/ISO EC57, 1998-(R)2008.

- Cohen, A. (1986). *Biomedical Signal Processing: Compression and automatic recognition*. CRC Press, 1st edition.
- De Chazal, P., O'Dwyer, M., and Reilly, R. B. (2004). Automatic classification of heartbeats using ecg morphology and heartbeat interval features. *IEEE transactions on biomedical engineering*, 51(7):1196–1206.
- Do, K., Tran, T., and Venkatesh, S. (2019). Graph transformation policy network for chemical reaction prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 750–760.
- Fout, A. M. (2017). *Protein interface prediction using graph convolutional networks*. PhD thesis, Colorado State University.
- Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1025–1035.
- Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., and Ng, A. Y. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature medicine*, 25(1):65.
- Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907*.
- Kojima, R., Ishida, S., Ohta, M., Iwata, H., Honma, T., and Okuno, Y. (2020). kgcn: a graph-based deep learning framework for chemical structures. *Journal of Cheminformatics*, 12(1):1–10.
- Lacasa, L., Luque, B., Ballesteros, F., Luque, J., and Nuno, J. C. (2008). From time series to complex networks: The visibility graph. *Proceedings of the National Academy of Sciences*, 105(13):4972–4975.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.
- Luz, E. J. d. S., Schwartz, W. R., Cámara-Chávez, G., and Menotti, D. (2016). Ecg-based heartbeat classification for arrhythmia detection: A survey. *Computer methods and programs in biomedicine*, 127:144–164.
- Micheli, A. (2009). Neural network for graphs: A contextual constructive approach. *IEEE Transactions on Neural Networks*, 20(3):498–511.
- Mondéjar-Guerra, V., Novo, J., Rouco, J., Penedo, M. G., and Ortega, M. (2019). Heartbeat classification fusing temporal and morphological information of ecgs via ensemble of classifiers. *Biomedical Signal Processing and Control*, 47:41–48.
- Moody, G. B. and Mark, R. G. (1990). The mit-bih arrhythmia database on cd-rom and software for use with it. In *[1990] Proceedings Computers in Cardiology*, pages 185–188. IEEE.
- Rajesh, K. N. and Dhuli, R. (2018). Classification of imbalanced ecg beats using re-sampling techniques and adaboost ensemble classifier. *Biomedical Signal Processing and Control*, 41:242–254.

- Rajpurkar, P., Hannun, A. Y., Haghpanahi, M., Bourn, C., and Ng, A. Y. (2017). Cardiologist-level arrhythmia detection with convolutional neural networks. *arXiv:1707.01836*.
- Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2008). The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80.
- Shi, H., Wang, H., Huang, Y., Zhao, L., Qin, C., and Liu, C. (2019). A hierarchical method based on weighted extreme gradient boosting in ecg heartbeat classification. *Computer methods and programs in biomedicine*, 171:1–10.
- Sperduti, A. and Starita, A. (1997). Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, 8(3):714–735.
- Wang, L., Sun, W., Chen, Y., Li, P., and Zhao, L. (2018a). Wavelet transform based ecg denoising using adaptive thresholding. In *Proceedings of the 2018 7th International Conference on Bioinformatics and Biomedical Science*, pages 35–40.
- Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., Zhou, J., Ma, C., Yu, L., Gai, Y., Xiao, T., He, T., Karypis, G., Li, J., and Zhang, Z. (2019). Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv:1909.01315*.
- Wang, T., Lu, C., Sun, Y., Yang, M., Liu, C., and Ou, C. (2021). Automatic ecg classification using continuous wavelet transform and convolutional neural network. *Entropy*, 23(1):119.
- Wang, Z., Chen, T., Ren, J., Yu, W., Cheng, H., and Lin, L. (2018b). Deep reasoning with knowledge graph for social relationship understanding. *arXiv:1807.00504*.
- Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. (2020). A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24.
- Xu, K., Hu, W., Leskovec, J., and Jegelka, S. (2018). How powerful are graph neural networks? *arXiv:1810.00826*.
- Yang, P., Wang, D., Zhao, W.-B., Fu, L.-H., Du, J.-L., and Su, H. (2021). Ensemble of kernel extreme learning machine based random forest classifiers for automatic heartbeat classification. *Biomedical Signal Processing and Control*, 63:102138.
- Zheng, J., Zhang, J., Danioko, S., Yao, H., Guo, H., and Rakovski, C. (2020). A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients. *Scientific Data*, 7(1):1–8.
- Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81.