




Network Optimization based on Genetic Algorithm for High-Level Data Classification

Janayna M. Fernandes , *Student Member, IEEE*, Gina M. B. Oliveira , *Member, IEEE*, Murillo G. Carneiro , *Senior Member, IEEE*

Abstract—High-level data classification techniques are capable of considering not only physical aspects of the data, such as space, distance, proximity, distribution, but can also consider their functional, topological and structural aspects. High-level techniques are commonly defined in two major steps: the construction of a network from the feature vector data and the uncovering of its underlying patterns using complex networks properties. In the network construction step, heuristics based on k-nearest neighbors strategies have been widely adopted, while several complex network measures (e.g. PageRank) have been modeled to learn high-level patterns of the input data. As both steps are directly related, i.e., the network configuration impacts directly the results obtained by the classifier, in this paper we develop a genetic algorithm (GA) to optimize the network construction step. To be specific, we hypothesize that the salient features of GAs, such as their robust search mechanism and binary representation, may provide a more powerful network representation in the context of the high-level classification based on importance characterization. In summary, extensive experiments with real data sets demonstrate that the networks provided by our GA strategy achieved higher predictive accuracy than those of a widely adopted method based on the nearest neighbors heuristic and competitive results against state-of-the-art ones.

Index Terms—Complex Networks, Genetic Algorithms, Particle swarm optimization, Network Optimization, High Data Classification, Graph Optimization.

I. INTRODUÇÃO

A Internet, as redes sociais, o cérebro humano, a bolsa de valores, os blecautes e os terremotos possuem uma grande característica em comum: são sistemas complexos cuja representação e modelagem podem ser realizadas através de redes complexas [1]. As redes complexas reúnem um conjunto de ferramentas para representar e modelar tais sistemas caracterizadas pela existência de padrões de conexões não triviais: nem completamente regulares, nem caóticos, sendo assim denominados complexos [2].

Similaridades podem ser identificadas em diferentes domínios através da representação em redes complexas. Um exemplo bem conhecido são as redes livres de escala [3] que modelam ligações preferenciais presentes em vários sistemas, tais como a Internet e as redes de citação, por exemplo [4], [5]. No contexto de aprendizado de máquina, a modelagem

e análise dos dados em rede têm contribuído para uma série de tarefas, tais como, detecção de comunidades, classificação multirrotulo e aprendizado transdutivo [6]–[9]. Nesse sentido, as medidas de rede são capazes de caracterizar diferentes aspectos, comportamentos e características subjacentes aos dados, já que são capazes de examinar não somente suas propriedades físicas (distância ou distribuição, por exemplo), mas também informações topológicas a partir da configuração deles em uma rede [9], [10].

As técnicas mais tradicionais, de *baixo nível*, como árvores de decisão, máquinas de vetores de suporte e redes neurais artificiais realizam o processo de classificação com sucesso para vários problemas. Contudo, há cenários em que o uso de grafos podem trazer vantagens, por exemplo quando há muita sobreposição de classes ou distribuição de dados muito arbitrárias [11], [12]. A Fig.1a mostra um exemplo simples da tarefa de classificação, em que existem padrões muito bem definidos. Nela, o objetivo é classificar o objeto de teste, + (azul). As técnicas de baixo nível possuem dificuldade nesse cenário já que consideram apenas aspectos físicos dos dados, negligenciando a estrutura e relação entre os dados [12]. Dessa forma, tais algoritmos possuem sérias limitações para associar o objeto de teste à classe vermelha (o). As técnicas de baixo nível consideram apenas os atributos físicos dos dados de entrada como a similaridade, distância ou distribuição.

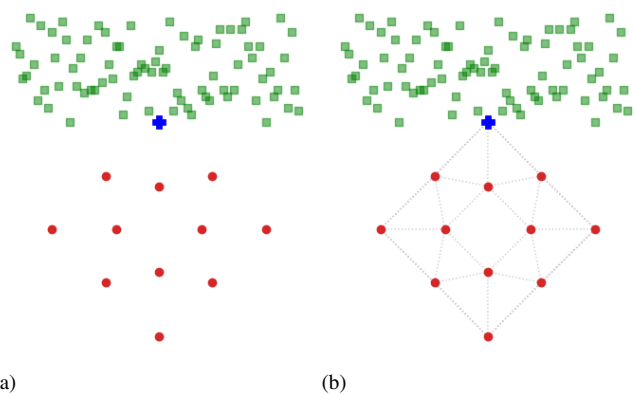


Fig. 1. Objeto de teste azul (+) a ser classificado entre a classe vermelha (o) ou verde (□). (a) Classificadores tradicionais teriam dificuldades pela proximidade que o objeto de teste tem com a classe não estruturada quadrada. (b) Exemplo de classificação de alto nível: o objeto de teste pertence a classe que possui um padrão claro.

Janayna M. Fernandes é aluna de mestrado na Faculdade de Computação, Universidade Federal de Uberlândia, Brasil e-mail: fernandesjnayna@gmail.com

Gina M. B. Oliveira é Professora Titular da Faculdade de Computação, Universidade Federal de Uberlândia, Brasil e-mail: gina@ufu.br

Murillo G. Carneiro é Professor Adjunto da Faculdade de Computação, Universidade Federal de Uberlândia, Brasil e-mail: mgcarneiro@ufu.br

Por outro lado, o uso de redes complexas permite, além de examinar atributos físicos dos dados, também considerar

informações topológicas a partir da configuração deles em uma rede. Esse tipo de classificação é denominada *classificação de alto nível* [11] e tais técnicas são capazes de detectar relações semânticas como a formação de padrão apresentada na Fig. 1a a partir da análise estrutural dos dados em rede [11], tal como apresentado na Fig. 1b. Diferentes métodos da literatura baseados em redes complexas, tais como *conformidade de padrão* e *caracterização de importância* [12], demonstraram que tais informações permitem um melhor desempenho preditivo na detecção do padrão semântico dos dados. Neste artigo investiga-se a técnica de caracterização de importância devido a sua baixa complexidade de parâmetros e a unificação da análise de associações físicas e topológicas em um único algoritmo [12].

Para realizar a classificação de alto nível em dados de tipo não-grafo (e.g., vetor de atributos, imagens, texto, etc), o primeiro passo é a geração de uma rede (ou grafo) na qual os vértices e as arestas representam respectivamente os objetos e as relações entre eles. A construção da rede é uma etapa crucial, pois a partir do grafo são extraídas as informações subjacentes dos dados para o processo de classificação [13].

O objetivo deste estudo é desenvolver uma solução baseada em algoritmos genéticos (AG) para otimização estrutural de redes, ou seja, encontrar a configuração mais adequada para representação das conexões dos objetos em rede. A principal motivação para a pesquisa é o mecanismo eficiente para representação de problemas discretos dos AGs, os quais podem refletir naturalmente as conexões em uma rede. Dessa forma, a hipótese investigada afirma que AGs podem prover redes mais apropriadas para a classificação via caracterização de importância do que técnicas tradicionais de formação de rede, como por exemplo a rede k NN, ao mesmo tempo em que podem ser competitivos em relação a outros métodos do estado-da-arte por permitirem a manipulação direta das configurações da rede a partir da representação e manipulação de variáveis binárias ao invés de contínuas. As principais contribuições da pesquisa são apresentadas a seguir:

- Desenvolvimento de um algoritmo genético com representação binária para otimização estrutural de redes no contexto de classificação de dados de alto nível;
- Análise comparativa e estatística do desempenho preditivo do método desenvolvido em relação àquele obtido pela rede k NN (método amplamente adotado na literatura) e pelo PSONet (método de otimização estado-da-arte para classificação de dados em redes);
- Caracterização das redes otimizadas pelo algoritmo genético em termos de topologia e estrutura.

O restante do artigo está organizado da seguinte forma. A seção II apresenta conceitos fundamentais necessários para a compreensão adequada deste trabalho, dentre eles a classificação de dados através de redes, a técnica de classificação via caracterização de importância e algoritmos genéticos, bem como outros trabalhos relacionados à construção e otimização de redes. A seção III descreve o método proposto para otimização de redes baseado em AG. A seção IV discute os resultados obtidos por nosso método em comparação com outras técnicas. E a seção V conclui o artigo.

II. FUNDAMENTAÇÃO E TRABALHOS RELACIONADOS

A. Classificação de Dados em Redes

Os dados de entrada de um algoritmo de classificação são um conjunto de instâncias cada uma delas denotada pela tupla (x, y) , sendo que x denota os atributos e y a classe [14]. Assim o número de classes é conhecido e o algoritmo recebe a informação das classes durante o treinamento (aprendizado supervisionado). O objetivo é mapear um conjunto de atributos x no seu rótulo de classe y , de modo a classificar corretamente novas instâncias cujos atributos (x) são conhecidos, porém a classe (y) não.

Diferentemente das técnicas mais conhecidas de classificação que baseiam-se somente nas características físicas dos dados (ex. distância ou distribuição), aprendizado baseado em rede é capaz de considerar também padrões topológicos dos dados ao representá-los em grafo [12], [13]. Para representar o conjunto de dados \mathcal{X} a partir de um vetor de atributos, \mathcal{X} é transformado em um grafo $\mathcal{G} = \{V, E\}$, onde cada vértice $v_i \in \mathcal{V}$ representa um item de dado $i \in \mathcal{X}$ e cada aresta $e_{i,u} \in \mathcal{E}$ representa uma conexão entre os vértices $v_i, v_u \in \mathcal{X}$ [14]. Por isso, \mathcal{G} desempenha um papel fundamental para obtenção dos resultados já que os padrões das classes são diretamente extraídos dele.

B. Classificação via Caracterização de Importância

A classificação via caracterização de importância é uma técnica de classificação de alto nível que avalia individualmente a importância dos itens de dados para determinar um rótulo de uma nova instância [12]. Além disso, tira proveito tanto de propriedades espaciais quanto estruturais ao representar os dados na forma de grafo. Nessa técnica, o conceito de *importância* é derivado de uma medida de centralidade chamada *PageRank*, a qual caracteriza a importância que um determinado objeto possui em uma rede com base no número de arestas que incidem no mesmo de modo que quanto mais conexões incidem no vértice mais importante ele é [15].

Na classificação via caracterização de importância, originalmente definida em [12], a *importância* de um item de teste y , denotada por \mathcal{I} , em relação à classe $l \in \mathcal{L}$ é dada por:

$$\mathcal{I}_y^{(l)} = \sum_{j \in \Lambda_y^{(l)}} \mathcal{I}_j, \quad (1)$$

em que $j \in \mathcal{X}_{train}$ denota um vértice rotulado, $\Lambda_y^{(l)}$ é um conjunto de nós que pertencem à classe l na qual y é temporariamente conectado, e \mathcal{I}_j representa a *importância* do vértice j .

A classificação via caracterização de importância consiste em duas fases principais: a fase de treino, onde o grafo é construído a partir dos dados de entrada (na forma de vetor de atributos) e posteriormente são calculadas uma medida de eficiência no fluxo da informação e outra de importância baseada no PageRank; e a fase de teste, que consiste na inserção virtual de um dado objeto de teste (*query*) no grafo baseado na melhoria da eficiência no fluxo de informação e na atribuição de importância ao objeto de teste, de modo que o mesmo seja atribuído à classe do componente na qual recebeu maior valor de importância [12].

C. Algoritmos Genéticos (AGs)

AGs são um framework genérico de busca e otimização em que o espaço de busca da solução é explorado a partir de uma amostragem aleatória de seus pontos utilizando de mecanismos que se baseiam na abstração do conceito de evolução e de operações genéticas como a geração de um novo ser e da possibilidade de mutação [16].

Tal framework considera um determinado período de gerações onde uma população aleatória de soluções candidatas é submetida a procedimentos genéticos (seleção, cruzamento e mutação). Cada indivíduo na população representa uma hipótese do espaço de busca, a qual é avaliada através de uma medida de desempenho que indica o quão boa é aquela hipótese para solução do problema. A medida de desempenho também é conhecida por *função de aptidão* ou *fitness* [17]. Um AG convencional pode ser descrito sucintamente através do pseudo código apresentado em Alg. 1.

Algoritmo 1: Estrutura simples de um algoritmo genético. Adaptado a partir de [16]

Input: Taxa de crossover (Cr); Tamanho da população (Tp); Probabilidade de mutação ($Pmut$); Tamanho do torneio ($tour$);

Output: Melhor solução

- 1 Inicializa a população de Tp indivíduos;
 - 2 Avalia população;
 - 3 **while** critério de parada não é satisfeito **do**
 - 4 Seleciona soluções para cruzamento;
 - 5 Realiza cruzamento;
 - 6 Realiza mutação;
 - 7 Avalia população;
 - 8 Seleciona soluções para a próxima população;
 - 9 Retorna o melhor indivíduo
-

O Alg. 1 tem como parâmetro uma taxa de crossover (Cr), o tamanho da população (Tp), uma taxa de mutação ($Pmut$) e quando aplicável um tamanho de torneio. E tem como saída uma população final que é evoluída, otimizada. Primeiramente, inicializa-se aleatoriamente uma população de soluções candidatas, posteriormente avaliadas utilizando de uma função de aptidão. Então inicia-se o processo de otimização por um determinado número de gerações ou outra condição de parada. Tal processo é caracterizado pelas seguintes etapas: *seleção*, em que indivíduos de melhor avaliação terão preferência para participar da etapa de cruzamento (*crossover*); *cruzamento*, em que os indivíduos selecionados contribuirão na formação de novos indivíduos; *mutação*, que altera, com alguma probabilidade, os novos indivíduos gerados, permitindo uma exploração global do espaço de busca. Em seguida os indivíduos obtidos são avaliados e ocorre a reinserção, com objetivo de selecionar os indivíduos que farão parte da próxima geração. No fim desse processo, é retornado o indivíduo com melhor avaliação.

Um dos grandes desafios no projeto de um AG se refere à representação e avaliação dos indivíduos, bem como na consequente escolha dos métodos adotados nos operadores genéticos de seleção, cruzamento, mutação e reinserção [16]. Esses aspectos serão cobertos na seção II.

D. Construção e Otimização de Redes

Para problemas de aprendizado supervisionado, o método de construção de grafo mais utilizado na literatura é o k -vizinhos mais próximos (kNN) [14], o qual gera um grafo direcionado em que cada vértice é conectado aos k vértices mais próximos, desde que os objetos sejam da mesma classe. Outro método utilizado é a vizinhança de raio ϵ (raio- ϵ), a qual gera um grafo não direcionado cujas conexões entre vértices de mesma classe são definidas a partir de um limiar de distância ϵ .

Vários outros métodos de construção de grafo da literatura também são derivados da rede kNN : i) no kNN simétrico [18] há conexão entre um dado par de vértices (v_i, v_j) se v_i pertence aos k -vizinhos mais próximos de v_j ou vice-versa; ii) no kNN Mútuo [19] há conexão entre um dado par de vértices (v_i, v_j) se ambos (mutuamente) pertencem aos k -vizinhos mais próximos um do outro; iii) o kNN Seletivo [14] retorna um grafo regular pois considera apenas vértices de mesma classe na seleção dos vizinhos mais próximos; iv) no k -Associados Ótimo [20] a rede é obtida a partir de uma variação do método kNN considerando a otimização de uma medida de pureza; e v) em [11] a combinação entre a rede kNN e raio- ϵ tem como objetivo retornar uma rede adaptada tanto para regiões densas (raio- ϵ) quanto esparsas (kNN). Em comum, todos esses métodos geram grafos a partir dos dados de entrada em forma de vetor de atributos e fazem, em maior ou menor grau, suposições fortes sobre os dados, tal como que suas relações podem ser mapeadas a partir de um mesmo número de ligações (k ou ϵ) entre os vértices ou componentes da rede [13].

Uma alternativa recente para lidar com tais limitações foi proposta em [13], que apresenta um framework para otimização estrutural de redes baseada em enxame de partículas denominada PSOnet. O framework foi usado para otimizar uma função de qualidade no contexto da classificação de alto nível via caracterização de importância. A partir de um extenso número de experimentos e comparações, PSOnet demonstrou desempenho preditivo superior aos principais métodos de construção da rede existentes, além de também superar vários algoritmos de classificação amplamente adotados na literatura.

Diferente do PSOnet, que otimiza as configurações da rede a partir de variáveis contínuas, a técnica de otimização apresentada neste artigo (AGNet) manipula as configurações da rede a partir de uma representação binária. Em consequência, AGNet não requer a conversão de variáveis contínuas para obter as configurações da rede, além de conduzir o processo de otimização sobre um espaço de configurações consideravelmente menor (finito).

III. ALGORITMO GENÉTICO PARA OTIMIZAÇÃO DE REDES

A Fig. 2 apresenta o método de otimização de redes baseado em AG para classificação de alto nível, denominado AGNet. O método é dividido em duas fases principais: treino e teste. Durante o treino ou otimização, a partir dos dados de treino, o AG é empregado para obter uma boa configuração da rede. Na fase de teste, a melhor solução obtida em treino é utilizada para criar a rede, agora lidando com os dados de teste, utiliza-se da técnica de classificação via caracterização de importância e da rede otimizada para classificar os dados.

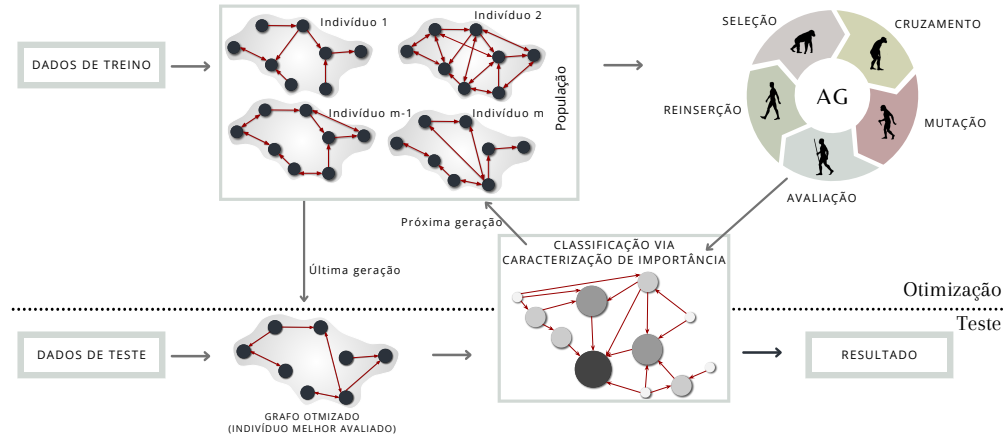


Fig. 2. Ilustração do método de otimização estrutural via AG aplicado ao aprendizado em rede.

De maneira formal, o AG manipula uma população de indivíduos $P = \{I_1, I_2, \dots, I_m\}$, na qual cada indivíduo $I_i \in P$ é denotado por:

$$I_i = \{v_1, v_2, \dots, v_n\}, \quad (2)$$

em que $v_i \in I_i$ representa as conexões de um dado vértice v_i (associado a um objeto $x_i \in X$) definidas por:

$$v_i = \{e_{i1}, e_{i2}, \dots, e_{iq}\}, \quad (3)$$

no qual $j \in \{1, 2, \dots, q\}$ denota as q possíveis conexões de v_i e $e_{ij} \in \{0, 1\}$ a existência ou ausência de uma dada conexão do vértice v_i para o vértice vizinho $Mapa_{ij}$ na rede, como é ilustrado na Fig. 3. Os vizinhos de cada vértice v_i são definidos baseado na heurística de mapeamento *MapAll*, originalmente proposta em [13] e que define a matriz *Mapa* a partir dos seguintes passos:

- Calcular a similaridade entre cada par de itens de dados;
- Selecionar para cada vértice v_i seus q vértices mais similares;
- Dado que $1 \leq z \leq q$, criar a matriz $Mapa_{n \times q}$ tal que:

$$Mapa_{iz} = \begin{cases} v_z & \text{se } l_i = l_z \\ \emptyset & \text{caso contrário.} \end{cases} \quad (4)$$

Note que $Mapa_{iz}$ é vazio se o vértice v_i não pertence à mesma classe que v_z . Pela formulação apresentada é importante observar também que, diferente do método de otimização contínuo apresentado em [13], o AG desenvolvido aqui realiza a otimização em um espaço discreto de soluções.

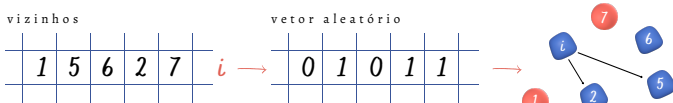


Fig. 3. Exemplo ilustrativo de *Map-all* em que a partir de um vetor aleatório os vértices de mesma classe se conectam. Neste exemplo não existe a conexão entre os vértices i e 7 pois são de classes diferentes, não obstante, mesmo os vértices i e 6 sendo de mesma classe não há a conexão entre eles pois não existe tal ligação no vetor *Map-all*.

Na Fig. 2 pode-se visualizar que, primeiramente, divide-se os dados em treino, validação e teste, gera-se uma população aleatória com m indivíduos com base na heurística de mapeamento *MapAll* a partir do conjunto de dados de treino. Cada indivíduo da população passa pelo processo de evolução, seleção, cruzamento e mutação, por m gerações. Durante a fase de otimização cada indivíduo I_k é avaliado através de uma função de aptidão sob o conjunto de dados de validação. Converte-se I_k em uma rede $\mathcal{G}_k = \{V_k, E_k\}$, onde $\mathcal{V}_k = \{1, \dots, n\}$ representa os vértices associados a cada item de dado e \mathcal{E}_k as arestas entre tais vértices. A função de aptidão adotada foi a *classificação via caracterização de importância* proposta em [12] que é uma técnica de classificação de alto nível que captura ambas características topológicas e físicas dos dados e utiliza a medida PageRank para classificar o objeto de teste no componente mais importante. Na fase de teste, utiliza-se o melhor indivíduo obtido no final das m gerações para criar a rede a ser utilizada para a classificação de alto nível a partir do conjunto de dados de teste.

Em relação aos operadores genéticos adotados pelo AGNet, eles são apresentados a seguir:

Seleção. Dois métodos foram avaliados para a etapa de seleção: roleta e torneio [21]. Na *roleta*, a probabilidade de cada indivíduo ser selecionado é dada de acordo com o seu valor de aptidão. No *torneio* seleciona-se aleatoriamente t indivíduos para formarem grupos e então seleciona-se o melhor indivíduo de cada grupo para o cruzamento [21].

Cruzamento. Dois métodos foram avaliados para a etapa de cruzamento: dois pontos e uniforme [22]. O cruzamento de *dois pontos* é um dos métodos de cruzamento mais simples em que escolhe-se aleatoriamente dois pontos de cortes nas mesmas posições para os dois pais, troca-se o material genético entre os pontos e dois novos filhos são gerados a partir deles. No cruzamento *uniforme* gera-se aleatoriamente um vetor do tamanho do indivíduo indicando se o gene virá de um indivíduo 1 ou 2 (novo indivíduo 1), e também o complemento desse vetor (novo indivíduo 2).

Mutação. Novos indivíduos são selecionados aleatoriamente para a mutação. Nesse caso, o bit que representa a conexão entre dois vértices é alterado: se existe conexão, passa

a não existir ou vice-versa.

Reinserção. Dois métodos de reinserção foram avaliados para a etapa de reinserção: pura e ordenada [17]. Na reinserção *pura* apenas um percentual (*elit*) da população original é mantida, enquanto o restante dos indivíduos originais é substituído pelos novos indivíduos gerados. Na reinserção *ordenada* a população total, tanto indivíduos originais quanto gerados, é avaliada e ordenada, selecionando-se então T_p melhores.

IV. RESULTADOS EXPERIMENTAIS

Para a realização dos experimentos foram consideradas seis bases de dados reais. As bases reais utilizadas estão disponíveis publicamente no repositório de dados da UCI Machine Learning [23] e são, sucintamente, apresentadas na Tabela I através de uma meta-descrição dos dados em termos de números de objetos, atributos e classes.

TABELA I

BREVE DESCRIÇÃO DOS CONJUNTOS DE DADOS EM ANÁLISE EM TERMOS DE NÚMERO DE ITENS DE DADOS (#*Inst.*), ATRIBUTOS (#*Atrib.*) E NÚMERO DE CLASSES (#*Classes*).

Nome	#Inst.	#Atrib.	#Classes
Iris	150	4	3
Teaching	151	5	3
Glass	214	9	7
Libras	360	90	15
Appendicitis	106	7	2
Balance	625	4	3

O ambiente experimental foi desenvolvido em linguagem Python. Cada experimento foi executado cinco vezes. Os parâmetros testados foram o γ , q , relacionados à heurística de mapeamento; dois operadores de seleção: torneio ($t = 3$) e a roleta; dois operadores de reinserção: a ordenada e a pura (*elit* = 20%); para o cruzamento foram utilizados o de dois pontos e o uniforme. Quanto à estrutura da rede foi considerado o grafo ponderado e quanto à medida de rede o *PageRank*, o tamanho da população e o número de gerações utilizados foram 100 e o percentual de cruzamento foi 80%. As combinações de parâmetros testados são apresentadas na Tab. II, totalizando 16 configurações.

A Tabela III apresenta o desempenho preditivo alcançado pelo método de classificação usando a rede *k*NN e as configurações do AG proposto. Com exceção da base Teaching, em todas as demais bases de dados houve melhoria considerável do resultado para várias configurações do AG. O teste estatístico de Friedman [24] foi conduzido para análise dos resultados considerando um nível de significância $\alpha = 0.05$. A hipótese nula afirma que o desempenho preditivo dos métodos de construção da rede são equivalentes. Tal hipótese é rejeitada pelo teste. Em seguida, adotamos o pós-teste de Nemenyi para identificar quais métodos possuem diferença estatística. O resultado do pós-teste é apresentado na Fig. 4, que apresenta o diagrama crítico de Nemenyi. De acordo com a figura, é possível observar que as configurações AGNet-C, AGNet-K, AGNet-E e AGNet-G possuem os melhores rankings médios. Em comum, essas quatro configurações de AG usam a seleção

TABELA II
DIFERENTES CONFIGURAÇÕES SOB INVESTIGAÇÃO COMPOSTOS PELOS PARÂMETROS γ , q E PELOS MÉTODOS DE SELEÇÃO, REINSERÇÃO E CRUZAMENTO.

Configuração	γ	Valor de q	Seleção	Reinserção	Cruzamento
AGNet-A	1	3	Torneio	Ordenada	Dois pontos
AGNet-B	1	3	Roleta	Ordenada	Dois pontos
AGNet-C	1	3	Torneio	Pura	Dois pontos
AGNet-D	1	3	Torneio	Ordenada	Uniforme
AGNet-E	1	5	Torneio	Ordenada	Dois pontos
AGNet-F	1	5	Roleta	Ordenada	Dois pontos
AGNet-G	1	5	Torneio	Pura	Dois pontos
AGNet-H	1	5	Torneio	Ordenada	Uniforme
AGNet-I	2	3	Torneio	Ordenada	Dois pontos
AGNet-J	2	3	Roleta	Ordenada	Dois pontos
AGNet-K	2	3	Torneio	Pura	Dois pontos
AGNet-L	2	3	Torneio	Ordenada	Uniforme
AGNet-M	2	5	Torneio	Ordenada	Dois pontos
AGNet-N	2	5	Roleta	Ordenada	Dois pontos
AGNet-O	2	5	Torneio	Pura	Dois pontos
AGNet-P	2	5	Torneio	Ordenada	Uniforme

por torneio e o cruzamento de dois pontos e foram capazes de superar estatisticamente a rede *k*NN e a configuração AGNet-M. Tal resultado revela o potencial de arquiteturas de otimização de redes baseadas em AGs para a classificação de dados.

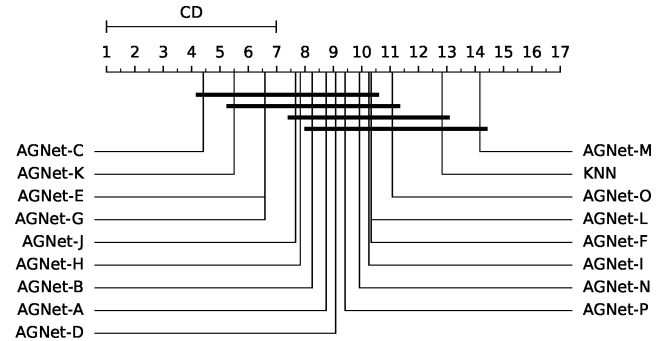


Fig. 4. Diagrama crítico de Nemenyi comparando o ranking médio de desempenho preditivo dos modelos analisados.

Em seguida, conduzimos outra análise relacionada à caracterização das redes AGNet. A Fig. 5 apresenta as redes obtidas pelo *k*NN (esquerda) e pelo AG (direita) respectivamente para as bases de dados Iris e Appendicitis. Na figura, é possível observar que além de melhorar o desempenho preditivo para a maioria das bases de dados, o processo de otimização também é responsável por representar as relações entre os vértices com menor número de arestas, o que contribui para reduzir a complexidade do processo de classificação. Por outro lado, isso também nos ajuda a explicar melhor a dificuldade das configurações de AG para a base Teaching, a qual parece estar relacionada com *overfitting* durante o processo de otimização da rede.

A Tabela IV traz a caracterização das redes obtidas pelo *k*NN, pelas duas melhores configurações AGNet-C e AGNet-K e pela pior configuração AGNet-M em função das medidas de assortatividade (*ASS*), *closeness* (*CLO*), menor caminho médio (*MCM*) e coeficiente de agrupamento (*CA*). Em síntese, as melhores configurações de AG possuem em comum maiores

TABELA III

ACURÁCIA (%) SEGUIDA DE DESVIO PADRÃO PARA AS DIFERENTES CONFIGURAÇÕES DO MODELO AGNET EM COMPARAÇÃO COM A TÉCNICA k NN. OS MELHORES RESULTADOS ESTÃO EM NEGRITO.

Alg.	Iris	Teaching	Glass	Libras	Appendicitis	Balance	Avg. Rank
k NN	96.89 ± 5.09	60.47 ± 8.69	69.89 ± 8.68	75.28 ± 6.14	80.79 ± 10.25	91.79 ± 2.91	12.8 ± 5.5
AGNet-A	97.33 ± 2.49	49.03 ± 5.55	72.68 ± 6.05	79.72 ± 3.99	84.55 ± 10.6	92.96 ± 1.63	8.8 ± 2.1
AGNet-B	97.33 ± 2.49	48.39 ± 5.4	74.63 ± 5.25	79.44 ± 4.06	84.55 ± 10.6	92.96 ± 1.55	8.3 ± 3.9
AGNet-C	98.0 ± 2.67	50.32 ± 4.38	73.66 ± 6.62	80.56 ± 4.56	87.27 ± 7.27	93.12 ± 1.65	4.4 ± 1.6
AGNet-D	97.33 ± 2.49	49.68 ± 1.58	72.2 ± 7.33	79.44 ± 3.45	84.55 ± 10.6	92.96 ± 1.99	9.1 ± 1.7
AGNet-E	98.0 ± 1.63	52.26 ± 9.44	73.17 ± 7.24	79.44 ± 3.66	85.45 ± 6.68	92.48 ± 1.57	6.6 ± 2.0
AGNet-F	98.0 ± 1.63	49.68 ± 10.32	71.22 ± 9.68	78.06 ± 3.66	85.45 ± 6.68	91.04 ± 2.79	10.3 ± 3.9
AGNet-G	97.33 ± 2.49	48.39 ± 4.08	73.66 ± 7.77	82.78 ± 4.08	87.27 ± 6.68	92.64 ± 1.85	6.6 ± 4.6
AGNet-H	98.0 ± 1.63	50.97 ± 8.01	73.66 ± 7.77	79.17 ± 5.2	86.36 ± 5.75	90.4 ± 2.68	7.8 ± 4.9
AGNet-I	96.67 ± 5.16	49.03 ± 5.91	72.2 ± 7.65	80.28 ± 4.76	85.45 ± 6.68	91.68 ± 2.3	10.3 ± 4.0
AGNet-J	96.67 ± 5.16	51.61 ± 8.89	73.66 ± 7.62	76.11 ± 3.22	85.45 ± 7.82	93.44 ± 1.92	7.7 ± 5.7
AGNet-K	98.67 ± 1.63	47.74 ± 8.51	74.15 ± 7.65	80.83 ± 4.43	84.55 ± 8.43	93.28 ± 1.2	5.5 ± 5.0
AGNet-L	96.67 ± 5.16	50.32 ± 7.8	71.22 ± 8.22	80.83 ± 3.97	83.64 ± 8.43	92.32 ± 1.87	10.3 ± 4.6
AGNet-M	98.0 ± 2.67	44.52 ± 8.01	69.76 ± 6.83	76.94 ± 2.58	82.73 ± 10.52	91.52 ± 0.64	14.2 ± 3.8
AGNet-N	98.67 ± 1.63	53.55 ± 6.95	70.73 ± 7.71	72.5 ± 6.3	83.64 ± 10.98	92.32 ± 2.3	9.9 ± 6.1
AGNet-O	98.0 ± 1.63	46.45 ± 5.24	73.17 ± 8.02	79.17 ± 4.48	83.64 ± 11.71	91.84 ± 1.06	11.1 ± 3.4
AGNet-P	98.0 ± 2.67	48.39 ± 8.16	71.22 ± 4.97	79.44 ± 2.55	83.64 ± 10.98	93.28 ± 2.24	9.4 ± 4.1

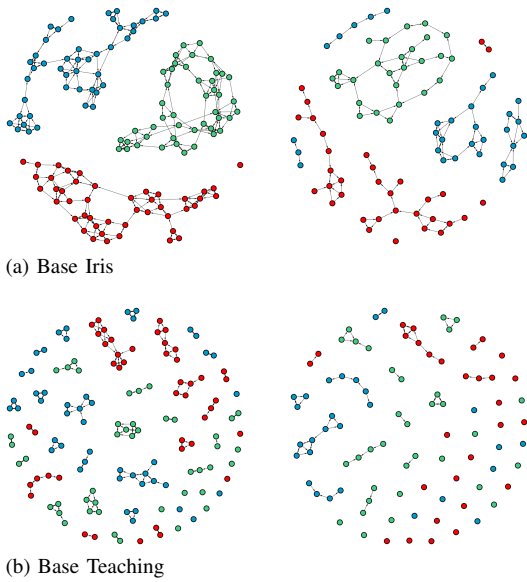


Fig. 5. Comparação visual entre as redes obtidas pelo k NN (esquerda) e pelo AGNet-C (direita).

valores de CLO e menores valores de MCM, enquanto a rede k NN provê redes com maiores valores de ASS e CA. Por outro lado, em comparação com AGNet-C e AGNet-K, a pior configuração de AG (AGNet-M) se caracteriza por valores maiores de MCM e menores de CLO. Esses resultados demonstram o potencial do processo de otimização de redes em transformar a estrutura e a topologia da rede k NN, de modo a adaptar a rede resultante para o problema de classificação considerado, contribuindo para um melhor desempenho preditivo na maioria dos casos.

A Tabela V traz uma comparação entre a melhor configuração de AG obtida pelo presente estudo (AGNet-C) e PSOnet, um método estado-da-arte para otimização estrutural de redes. Como pode ser visto, PSOnet alcança os melhores resultados para as bases de dados Iris, Teaching e Balance, sendo superado por AGNet-C para Glass, Libras e Appendicitis. Para

TABELA IV
MEDIDAS SUMARIZANDO AS CARACTERÍSTICAS TOPOLÓGICAS EM TERMOS DE ASSORTATIVIDADE MÉDIA (ASS), *Closeness* (CLO), MENOR CAMINHO MÉDIO (MCM), COEFICIENTE DE AGRUPAMENTO (CA) NA DIFERENTES CONFIGURAÇÕES DE REDES. TÉCNICAS SINALIZADAS COM “*” ALCANÇARAM O MELHOR RESULTADO NA TABELA III.

Dataset	Algs.	ASS	CLO	MCM	CA
Iris	k NN	0.16	0.26	3.51	0.44
	AGNet-C	0.01	0.47	1.33	0.27
	AGNet-K*	-0.01	0.51	0.97	0.26
	AGNet-M	0.03	0.33	3.00	0.36
Appendicitis	k NN	0.34	0.28	5.33	0.43
	AGNet-C*	-0.08	0.43	1.24	0.24
	AGNet-K	0.11	0.53	0.94	0.24
	AGNet-M	0.04	0.33	2.19	0.34
Teaching	k NN*	0.42	0.82	0.84	0.63
	AGNet-C	0.00	0.84	0.41	0.30
	AGNet-K	0.12	0.85	0.49	0.45
	AGNet-M	0.05	0.74	0.84	0.39

analisar estatisticamente o desempenho de ambos os métodos foi realizado o teste de Wilcoxon. Considerando um nível de significância $\alpha = 0.05$, o teste aponta que o desempenho dos modelos são equivalentes. Este é um resultado interessante, pois indica que as redes obtidas pelo AG além de apresentarem desempenho significativamente melhor do que aquelas obtidas pelo método k NN, são competitivas em relação às redes obtidas pelo PSOnet. Uma vez que PSOnet é baseado em uma técnica de otimização sofisticada para problemas de grande escala, contribui para tal resultado a representação binária do espaço de busca (finito) e facilita o processo de otimização.

V. CONCLUSÕES

Este artigo apresenta um AG para otimização estrutural de redes no contexto da classificação de dados de alto nível.

TABELA V

COMPARAÇÃO DA MELHOR CONFIGURAÇÃO DE AGNET EM RELAÇÃO A PSONET, MÉTODO ESTADO-DA-ARTE PARA OTIMIZAÇÃO ESTRUTURAL DE REDES.

Base de dados	AGNet-C	PSONet [13]
Iris	98.00 ± 2.67	100.0 ± 0.00
Teaching	50.32 ± 4.38	62.58 ± 5.24
Glass	73.66 ± 6.62	67.80 ± 3.58
Libras	80.56 ± 4.56	77.50 ± 2.97
Appendicitis	87.27 ± 7.27	82.72 ± 1.81
Balance	93.12 ± 1.65	95.36 ± 0.93

Dezesseis configurações diferentes foram exploradas com o objetivo de obter uma boa configuração de rede que pudesse ser aplicada em conjunto com a técnica de classificação de dados via caracterização de importância. Experimentos conduzidos em seis bases reais revelaram que tal método pode aprimorar a extração de informação subjacente nos dados. Nesse sentido, os resultados obtidos demonstraram que redes otimizadas por nosso método foram capazes de superar estatisticamente redes geradas pela rede k NN, técnica mais adotada na literatura, além de obterem resultados competitivos em relação ao estado-da-arte. A caracterização das redes geradas ressaltou ainda a capacidade do AG proposto em explorar configurações de rede com estrutura e topologia consideravelmente diferentes daquelas obtidas pela rede k NN e entre as diferentes configurações do AG avaliadas. Como trabalhos futuros, pretendemos desenvolver mecanismos que auxiliem no processo de busca do AG, de modo a lidar melhor com problemas de *overfitting*, além de estender as nossas simulações e análises experimentais.

AGRADECIMENTOS

Esta pesquisa é parcialmente financiada pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq (439556/2018-0) e pela Fundação de Amparo à Pesquisa do Estado de Minas Gerais – FAPEMIG (APQ 00410-21).

REFERÊNCIAS

- [1] A. Barrat, M. Barthelemy, R. Pastor-Satorras, and A. Vespignani, “The architecture of complex weighted networks,” *Proceedings of the national academy of sciences*, vol. 101, no. 11, pp. 3747–3752, 2004.
- [2] M. G. Carneiro, *Redes complexas para classificação de dados via conformidade de padrão, caracterização de importância e otimização estrutural*. PhD thesis, Universidade de São Paulo, 2017.
- [3] A.-L. Barabási and R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
- [4] R. Van Der Hofstad, *Random graphs and complex networks*, vol. 43. Cambridge university press, 2017.
- [5] M. Drobyshvskiy and D. Turdakov, “Random graph modeling: A survey of the concepts,” *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, pp. 1–36, 2019.
- [6] L. M. Freitas and M. G. Carneiro, “Community detection to invariant pattern clustering in images,” in *Brazilian Conference on Intelligent Systems*, pp. 610–615, IEEE, 2019.
- [7] Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S. J. Hwang, and Y. Yang, “Learning to propagate labels: Transductive propagation network for few-shot learning,” *arXiv preprint arXiv:1805.10002*, 2018.
- [8] J. E. Van Engelen and H. H. Hoos, “A survey on semi-supervised learning,” *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.
- [9] V. H. Resende and M. G. Carneiro, “Analysis of complex network measures for multi-label classification,” *International Journal on Artificial Intelligence Tools*, vol. 30, no. 04, p. 2150023, 2021.

- [10] M. G. Carneiro, B. C. Gama, and O. S. Ribeiro, “Complex network measures for data classification,” in *International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2021.
- [11] T. C. Silva and L. Zhao, “Network-based high level data classification,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 6, pp. 954–970, 2012.
- [12] M. G. Carneiro and L. Zhao, “Organizational data classification based on the importance concept of complex networks,” *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3361–3373, 2018.
- [13] M. G. Carneiro, R. Cheng, L. Zhao, and Y. Jin, “Particle swarm optimization for network-based data classification,” *Neural Networks*, vol. 110, pp. 243–255, 2019.
- [14] M. G. Carneiro and L. Zhao, “Analysis of graph construction methods in supervised data classification,” in *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pp. 390–395, IEEE, 2018.
- [15] L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web,” tech. rep., Stanford InfoLab, 1999.
- [16] S. Katoch, S. S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future,” *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 8091–8126, 2021.
- [17] S. Mirjalili, “Genetic algorithm,” in *Evolutionary algorithms and neural networks*, pp. 43–55, Springer, 2019.
- [18] T. Guo, K. Yu, M. Aloqaily, and S. Wan, “Constructing a prior-dependent graph for data clustering and dimension reduction in the edge of aiot,” *Future Generation Computer Systems*, vol. 128, pp. 381–394, 2022.
- [19] Y. Zhang, S. Ding, L. Wang, Y. Wang, and L. Ding, “Chameleon algorithm based on mutual k -nearest neighbors,” *Applied Intelligence*, vol. 51, no. 4, pp. 2031–2044, 2021.
- [20] M. G. Carneiro, J. L. G. Rosa, A. A. Lopes, and L. Zhao, “Network-based data classification: combining k -associated optimal graphs and high-level prediction,” *Journal of the Brazilian Computer Society*, vol. 20, no. 1, pp. 1–14, 2014.
- [21] S. L. Yadav and A. Sohal, “Comparative study of different selection techniques in genetic algorithm,” *International Journal of Engineering, Science and Mathematics*, vol. 6, no. 3, pp. 174–180, 2017.
- [22] P. Kora and P. Yadlapalli, “Crossover operators in genetic algorithms: A review,” *International Journal of Computer Applications*, vol. 162, no. 10, 2017.
- [23] A. Asuncion and D. Newman, “Uci machine learning repository,” 2007.
- [24] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.



Janayna Moura Fernandes received the B.S. degree in Information Systems from the Federal University of Uberlândia, Brazil in 2019. She is currently a M.Sc. candidate in Computer Science at Federal University of Uberlândia and has interests in the topics of nature-inspired optimization and complex systems.



Gina Maira Barbosa de Oliveira received the B.S. degree from the Federal University of Uberlândia, Brazil, in 1990, and the M.Sc. and Ph.D. from the Aeronautics Institute of Technology, Brazil, respectively in 1992 and 1999. She is a Professor with the Faculty of Computing, Federal University of Uberlândia, Brazil and has experience on the following topics: genetic algorithms, cellular automata, evolutionary computing and artificial intelligence.



Murillo Guimarães Carneiro (M'17-SM'20) received the PhD degree from the University of São Paulo, Brazil in 2016; the M.Sc. degree from the Federal University of Uberlândia, Brazil in 2012; and the Tech. degree from the Goiano Federal Institute, Brazil in 2008. He is an Assistant Professor with the Faculty of Computing, Federal University of Uberlândia, Brazil. His research interests include machine learning, complex networks, network-based learning and nature-inspired computing. Dr. Carneiro was a recipient of the Google Latin America Research Awards in 2020 and 2022.