UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Previsão de Arestas em Redes Complexas

Ana Clara Kandratavicius Ferreira

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP
SÃO CARLOS

Ana Clara Kandratavicius Ferreira

Previsão de Arestas em Redes Complexas

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestra em Ciências – Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Francisco Aparecido Rodrigues

USP – São Carlos
Fevereiro de 2021

Ana Clara Kandratavicius Ferreira

Link Prediction in Complex Networks

Master dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. *EXAMINATION BOARD PRESENTATION COPY*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Francisco Aparecido Rodrigues

USP – São Carlos
February 2021

À minha família e amigos.

AGRADECIMENTOS

Ao meu orientador Professor Francisco Aparecido Rodrigues pela orientação e paciência.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

RESUMO

FERREIRA, A. C. K. **Previsão de Arestas em Redes Complexas**. 2021. 46 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

Este trabalho pretende analisar o problema de previsão e reconstrução de arestas por meio da observação das similaridades entre os nós. A previsão de arestas é um problema de ampla relevância para diversas áreas de conhecimento incluindo estudos sociais, neurociência e redes de infraestrutura. Nestes casos temos o conjunto de arestas observáveis, nosso objetivo é por meio da observação destas e das similaridades entre os vértices que se conectam, inferir arestas faltantes ou arestas que se formarão em algum tempo no futuro. Esse estudo permitirá uma melhor compreensão sobre a relação entre a as características estruturais ou particulares dos vértices e a formação de conexões em redes.

Palavras-chave: Redes complexas, previsão de arestas, modelos de redes, medidas de similaridade, mapeamento de nós.

ABSTRACT

FERREIRA, A. C. K. **Link Prediction in Complex Networks**. 2021. 46 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2021.

The present work intends to analyze the problem of forecasting and reconstruction of edges by observing the similarities between the nodes of a network. Link prediction is a problem of wide scope for several areas of knowledge, including social studies, neuroscience and infrastructure networks. In these cases we have a set of observable links, our goal is to observe these and the similarities between the vertices that connect, infer missing links or links that will form at some time in the future. This study allows a better understanding of the relationship between the network structural characteristics or particular atributes of the vertices and the formation of composition in networks.

Keywords: Complex Networks, Link prediction, network models, similarity measures, Node embedding.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE SÍMBOLOS

k_i — Grau vértice i

knn_i — Grau médio dos vizinhos do vértice i

Γ_i — Vizinhança do vértice i

cc_i — coeficiente de aglomeração do vértice i

Cl_i — Centralidade de proximidade do vértice i

B_i — Grau de intermediação do vértice i

π^T — PageRank

D — Densidade da rede.

l — Caminho mínimo.

$l_{max} - - - Dimetro$.

T — Transitividade.

r — Assortatividade.

SUMÁRIO

CAPÍTULO

1

INTRODUÇÃO

Sistemas complexos são formados por muitos elementos apresentando comportamento coletivo, emergência e propriedades que não podem ser preditas a partir da análise dos seus componentes individuais. A estrutura de sistemas complexos pode ser descrita por redes complexas, que matematicamente são representadas por grafos.

Os elementos que constituem um sistema complexo apresentam diversas variáveis que caracterizam a sua dinâmica e estrutura. Por exemplo, em redes sociais, cada indivíduo é representado por um vértice na rede e as conexões define os laços de amizade. Tais indivíduos podem ter variáveis associadas, constituindo o que chamamos de metadados. Essas variáveis pode ser independentes da rede, como idade do indivíduo, ou estar ligada à padrões de conexões, tal como o seu número de conexões e índice de centralidade. Outras diversas variáveis e graus de liberdade podem ser adicionados para a caracterização de um vértice.

Portanto, podemos criar mapeamentos da estrutura de grafo, fazendo uma transformação do vértice para o espaço $\mathbb{R}^n$ onde cada coordenada indica um atributo próprio do indivíduo representado pelo vértice na rede, ou uma característica calculada em relação a rede. Assim definimos um conjunto de dados nos quais podemos aplicar diversos modelos de aprendizado de máquina para classificação de vértices ou previsões de comportamentos.

Neste trabalho estamos interessados na previsão de arestas. A previsão de arestas visa inferir a formação ou perda de conexões entre nós, indicando como a rede irá evoluir. Outra aplicação é determinar a existência ou inexistência de um link para reconstruir uma rede em que não conhecemos as relações entre os elementos. Para isto iremos nos basear em características dos vértices que podemos extrair ou calcular.

A previsão de arestas é baseada na noção de similaridade em grafos e na observação empírica de que vértices similares tendem a se conectar. A similaridade é um conceito aberto e assume significados diferentes dependendo do contexto. Embora muito importante, esse problema de predição da arestas a partir de características dos vértices é bastante complicado, pois está

relacionado com a estrutura da rede (e.g. aleatória, sem escala, com estrutura de comunidades), e da natureza das conexões. Além disso, as redes tendem a ser esparsas, dificultando, a predição da estrutura da rede a partir de conexões preexistentes. A previsão de arestas é uma linha de pesquisa em desenvolvimento dentro da Teoria das Redes Complexas.

Muitos métodos têm sido desenvolvidos para prever a formação de arestas a partir de mapeamentos dos vértices. Embora alguns métodos ofereçam resultados precisos em algumas aplicações, a estimação depende de muitos fatores e é esperado que não exista um método que seja geral o suficiente para qualquer aplicação, como nos problemas de aprendizado de máquina (ver *No free lunch theorem* (**??**)). Assim, estudos que comparem diferentes métodos em diferente configurações podem trazer importantes avanços nessa área de pequisa.

No presente trabalho, estamos interessados em contribuir para o estudo de métodos para inferir conexões na rede. Para isso, vamos considerar diferentes metodologias. A primeira será pelo cálculo de diferentes medidas de similaridade presentes na literatura. Em seguida utilizaremos as medidas de similaridade em conjunto com medidas de centralidade do vértice e determinaremos um vetor de características que descreve os vértices e arestas. Sobre esse conjunto de dados será aplicado diferentes métodos de aprendizado de máquina. A terceira abordagem dispensa a definição das características pois automatiza o processo por meio de mapeamento automático que transforma um vértice em um vetor d-dimensional por meio de caminhadas aleatórias pela rede. Esse estudo será feito para diferentes topologias de redes, tais como redes sem escala, aleatórias, assortativas e com estrutura de comunidades.

1.0.1 Objetivos

Os objetivos do nosso trabalho podem ser resumidos em:

- Comparar diferentes medidas de similaridade usadas na previsão da arestas e avaliar o desempenho das previsões feitas a partir de medidas predefinidas em comparação com a caracterização automática dos *Embeddings*.
- Determinar os melhores métodos para prever arestas em diferentes estruturas de redes.
- Verificar como a estrutura da rede interfere na predição variando o nível de assortatividade, heterogeneidade e estrutura de comunidades da rede.

Esses trabalhos irão permitir um melhor entendimento sobre a relação entre a estrutura e métodos de previsão de arestas em redes, permitindo assim uma melhor previsão da estrutura em situações prática, como neurociências, mercado financeiro, e redes sociais onde queremos reconstruir a estrutura ou prever a formação de conexões

CAPÍTULO

2

REDES COMPLEXAS

Existem diversos sistemas que nos cercam. Podemos considerar, por exemplo, os sistemas aéreos e rodoviário que interconectam cidades por meio de rotas aéreas e terrestres, a sociedade da qual participamos e e interagimos com outras bilhões de pessoas; os neurônios no nosso cérebro que em conjunto interpretam o mundo a nossa volta. Todos estes sistemas complicados são composto por elementos e suas relações, são chamados de Sistemas Complexos e, muitas vezes, apesar de diferentes, apresentam características em comum.

Nestes sistemas, conhecer um elemento que os compõe não os explica em sua totalidade, como conhecer um neurônio não explica os processos cognitivos. Os elementos se comportam coletivamente e independente de um controle central, se influenciam mutuamente e se auto-organizam (**????**). Estruturalmente observamos a tendência dos elementos se organizarem em comunidades e o surgimento de alguns elementos com muitas conexões enquanto a maioria possuem poucas (BARABÁSI, 2007). Dada a importância e a interdisciplinaridade dos sistemas complexos, torna-se necessário sua compreensão, descrição e previsão de seus comportamentos. Consequentemente surge, no final da década de 90, a teoria das redes complexas (WATTS; STROGATZ, 1998; ALBERT, 1999).

Redes complexas são representações de sistemas complexos. Cada elemento que compõe o sistema é representado por um vértice, as interações entre os pares de vértices são representadas por arestas. Matematicamente, utilizamos a Teoria de Grafos para criar uma espécie de mapa das relações que compõe o sistema, indicando quais vértices estão relacionados e quais caminhos de arestas podemos percorrer para chegar de um elemento à outro. Definimos um grafo como $G = (V, E)$. Os vértices, também chamados de nós, formam o conjunto V e as arestas formam o conjunto E. Podemos considerar um terceiro conjunto W que atribui pesos às arestas, muito úteis quando estamos analisando, por exemplo, viagens entre aeroportos. Neste caso, cada aeroporto é um nó e as ligações se dão pela existência de um voo entre aeroportos, os valores das arestas são os preços dos voos. O grafo que mapeia este sistema é uma tripla $G = (V, E, W)$ e podemos utilizá-lo pra calcular a rota mais barata entre cidades.

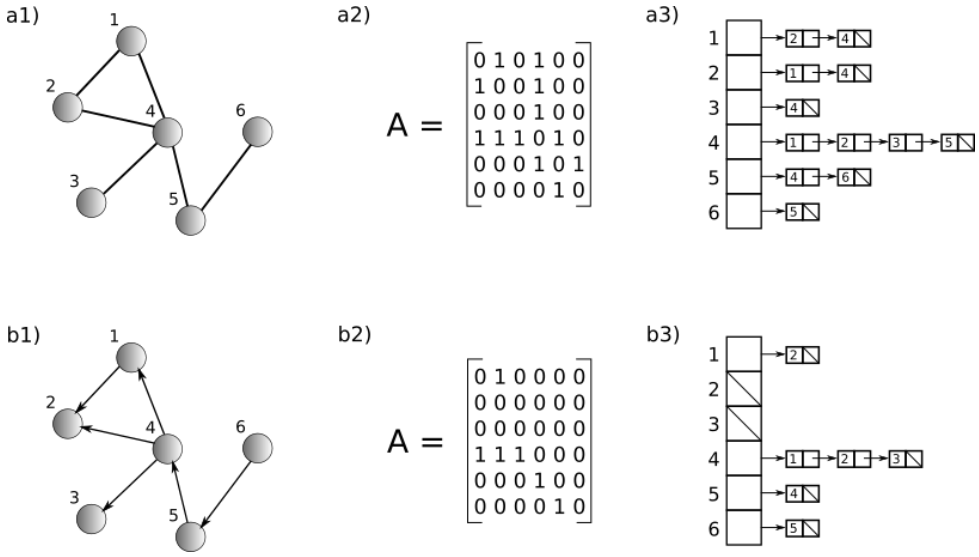

Figura 1 – Redes complexas podem ser representadas por matrizes ou listas de adjacência. Em (a) temos uma rede não-direcionada e em (b) uma rede direcionada. Em (a2), temos a matriz cujos elementos a_{ij} são iguais a 1 se há uma ligação entre os vértices i e j e iguais a zero, caso contrário. Já em (b2), os elementos da matriz a_{ij} são iguais a 1 se existe uma conexão dirigida do vértice i para o vértice j, e em (a3) e (b3) temos a representação em listas de adjacências

Outra característica dos sistemas que estamos interessados em mapear está na natureza das relações entre os componentes, elas podem ser recíprocas ou unilaterais. A reciprocidade ocorre frequentemente em redes sociais, pois as interações entre duas pessoas são normalmente bilaterais, neste caso teremos um grafo não-direcionado, por outro lado, se queremos representar uma cadeia alimentar, a direção da relação deve ser levada em conta.

Computacionalmente, a rede pode ser representada por meio de uma matriz ou uma lista de conexões. Lista é uma estrutura de dados que armazena os pares de vértices (i, j) que possuem ligações. No caso da matriz, definimos uma matriz A onde se dois vértices i e j estão ligados, a entrada a_{ij} na matriz A será igual a 1 e igual a 0, caso contrário. A Figura 1 mostra um exemplo de uma rede não-direcionada e de uma direcionada em listas e matrizes, note que no caso da rede não-direcionada, a matriz A é simétrica.

A maioria dos sistemas complexos são formados por milhares, ou mesmo milhões, de componentes. Sendo assim, é necessário usar medidas descritivas para caracterizar a topologia do sistema. A seguir serão descritas algumas medidas fundamentais que tornam possível a análise desses sistemas.

2.1 Caracterização de Redes Complexas

2.1.1 Medidas de Centralidade

Medidas de centralidade caracterizam os nós da rede, permitindo que possamos compará-los e medir sua importância perante a outros nós.

O grau, ou conectividade, de um vértice, é igual ao seu número de conexões. É calculada em termos da matriz de adjacência A pela equação

$$k_i = \sum_{j=1}^{N} a_{ij}, \tag{2.1}$$

o valor do grau indica quais são os componentes mais conectados, ou *hubs*. Em uma rede social, por exemplo, são as pessoas com muitos amigos, estas pessoas são ditas mais centrais.

Ainda relacionado ao grau, podemos caracterizar um nó pelo grau médio de seus vizinhos. Quanto mais conectados os vizinhos de um vértice, mais central ele é. A média dos graus dos vizinhos de um nó i é definida por

$$knn_i = \frac{1}{k_i} \sum_{j \in \Gamma_i} k_j, \tag{2.2}$$

onde Γ_i é a vizinhança do nó i.

O coeficiente de aglomeração, ou *clustering coefficient* calcula a frequência que os vizinhos de um nó se conectam entre si, formando triângulos ou *loops*. No caso de uma rede de amizades, significa a probabilidade de dois amigos de uma pessoa serem amigos também. Em, matriz de adjacências é calculado por

$$cc_i = \frac{2e_i}{k_i(k_i-1)} = \frac{\sum_{j=1}^{N}\sum_{m=1}^{N} a_{ij}a_{jm}a_{mi}}{k_i(k_i-1)}, \tag{2.3}$$

onde e_i representa o número de conexões entre os vizinhos do vértice i.

Menores caminhos, ou caminhos geodésicos, são a menor sequência de vértices e arestas, sem repetições, entre dois vértices. Podemos armazenar os menores caminhos entre dois nós por uma matriz de distâncias D, cujos elementos d_{ij} expressam o valor do menor caminho entre os vértices i e j, obtidos por algoritmos de busca. Caso não exista um caminho entre i e j, definimos $d_{ij} = \infty$. Usualmente considera-se apenas o maior componente conectado, ou seja, o maior conjunto de vértices e arestas no qual sempre existe um caminho entre dois vértices.

Baseado no conceito dos menores caminhos, temos as medidas de centralidade de proximidade, *closeness centrality*, e grau de intermediação, *betweenness centrality*. A centralidade de

proximidade de um nó i é dada pela soma do inverso da distância entre i e todos os outros nós j, $j \neq i$, ou seja

$$Cl_i = \frac{N}{\sum_{j=1, j\neq i}^{N} d_{ij}}, \tag{2.4}$$

em que d_{ij} é a distância entre os vértices i e j. O grau de intermediação quantifica a proporção de caminhos mínimos entre a e b que passam pelo vértice i em relação à todos os caminhos mínimos entre a e b, matematicamente

$$B_i = \frac{\sigma(a,i,b)}{\sigma(a,b)}, \tag{2.5}$$

$\sigma(a,i,b)$ é o número de caminhos mínimos entre a e b que passam por i e $\sigma(a,b)$ é o número de caminhos mínimos entre a e b.

Ademais, temos o *Google PageRank* que calcula a probabilidade de uma caminhada aleatória na rede passar por determinado vértice, esta probabilidade define a centralidade do nó. O *PageRank* é definido por

$$\pi^T = \pi^T \mathbf{G}, \tag{2.6}$$

sendo G a matriz Google definida por

$$\mathbf{G} = \kappa\left(\mathbf{P} + \frac{ae^T}{N}\right) + \frac{1-\kappa}{N} uu^T, \tag{2.7}$$

a é um vetor binário em que a_i é igual a 1 se a_i não tem nenhuma aresta de saída e 0 caso contrário, u é um vetor de 1's de tamanho N, $P_{ij} = \frac{A_{ij}}{k_i}$ é uma matriz de probabilidade de transição e κ é uma constante e $\kappa = 0.85$ no algoritmo original. π é o auto-vetor dominante de G cuja i-ésima entrada corresponde ao *PageRank* do vértice i e $\sum_i \pi_i = 1$.

2.1.2 Medidas Globais

Nesta seção são apresentadas medidas que descrevem a rede completa, em oposição á seção anterior que caracterizava os nós individualmente. Primeiramente deve-se citar as medidas descritivas básicas como o número de vértices que compões a rede, denominado por $|V|$ e o número de arestas $|E|$ que nos permite definir a densidade da rede a densidade da rede

$$\mathbf{D} = \frac{2|E|}{|V|(|V|-1)}, \tag{2.8}$$

esta medida indica se a rede é densa, com número de arestas próximo ao máximo, ou esparso, com apenas algumas arestas.

Estatísticas descritivas podem ser derivadas de características particulares dos vértices. A partir do grau determinamos o grau médio da rede, a variância e sua distribuição. O mesmo ocorre em relação ao coeficiente de aglomeração, usamos a média como atributo da rede. Pelos caminhos mínimos l obtemos a média e seu valor máximo $l_{max} = \max_{i,j} l_{ij}$ que definimos como o diâmetro da rede.

Transitividade T, assim como o coeficiente de aglomeração, está relacionado com a formação de triângulos na rede, porém a transitividade é uma medida que descreve a rede globalmente. É dada pela razão entre o número de triângulos observados e o número possíveis triângulos. A maioria das redes reais possuem alta transitividade e diâmetro pequeno.

Assortatividade, ou *Assortativity*, quantifica se nós que apresentam determinada característica tendem a se conectar com outros nós que apresentam a mesma característica ou se tendem a se conectar com nós diferentes. Podemos considerar, por exemplo, o grau. Ao sortear aleatoriamente uma aresta na rede, analisamos a probabilidade de conectar um nó de grau k_i com outro de grau k_j. Com isso, podemos classificar a rede em assortativa, dissortativa ou neutra. No primeiro caso, os *hubs*, nó de alto grau, tendem a se conectar com outros *hubs*, no segundo caso, *hubs* se evitam, se conectando com nós de baixo grau. Em redes neutras as conexões são aleatórias.

2.2 Modelos

2.2.1 Modelo Aleatório de Erdös-Renyi (ER)

Proposto pelos matemático Paul Erdös e Alfred Rényi (ERDÖS; RÉNYI, 1959) em 1959, o modelo ER assume que a probabilidade da formação de uma aresta é a mesma para qualquer par de nós. Este modelo não é considerado representativo de redes reais mas pode ser usado como modelo nulo pra efeito comparativo.

Para gerar redes segundo este modelo, escolhemos o número N de nós, o número de arestas m ou a probabilidade de conexão $p = m/(N-1)$. A seguir selecionamos os vértices dois a dois e criamos a conexão com probabilidade p. Descrevemos, portanto, um processo de Bernoulli que gera uma rede com grau médio $\langle k \rangle = p(N-1)$ e a distribuição do grau é uma Binomial que tende a uma Poisson quando N é grande, ou seja

$$P(K = k) = e^{-\langle k \rangle} \frac{\langle k \rangle^k}{k!}. \tag{2.9}$$

2.2.2 Modelo de pequeno mundo (WS)

Observando a estrutura de redes reais, Duncan J. Watts e Steven Strogatz (WATTS; STROGATZ, 1998) perceberam que triângulos se formavam com mais frequência do que em

redes aleatórias e, em 1998, propuseram um modelo alternativo ao aleatório ao qual deram o nome de *small world* em referência ao experimento de Stanley Milgram. Neste modelo, assim como no experimento de Milgram, o caminho médio entre dois elementos da rede é curto e os triângulos se formam mais frequentemente.

A construção de uma rede segundo este modelo se dá a partir de uma rede regular com N vértices ligados a κ vizinhos, em seguida, cada aresta é aleatoriamente reconectada a um novo nó com uma probabilidade fixa p, introduzindo aleatoriedade na rede. Portanto a rede de Watts-Strogatz, dependendo do p escolhido, fica entre a rede regular e a completamente aleatória.

2.2.3 Modelo Livre de escala (BA)

Albert-László Barabási e Réka Albert estavam estudando o Teia Mundial (*World Wild Web*) quando notaram uma característica ainda não representada pelos modelos de redes propostos até então. Eles verificaram que a distribuição do grau na web não é aleatória, mas de livre-escala (*scale-free*), da forma $P(k) \propto k^{-\gamma}$ (BARABASI; ALBERT, 1999), onde alguns nós são muito conectados e a maioria possui poucas conexões.

Para simular uma rede seguindo este modelo, definimos um conjunto com N_0 vértices totalmente conectados e a cada passo adicionamos um novo vértice com m arestas, $m \leq N_0$, que se conectam aos nó já presentes na rede com probabilidade

$$P_{i \to j}(t+1) = \frac{k_j(t)}{\sum_u k_u(t)}, \tag{2.10}$$

sendo $k_j(t)$ o grau do vértice j no tempo t. A este processo se dá o nome de Ligação Preferencial, pois a probabilidade de uma ligação ser formada é proporcional ao grau.

2.2.4 Modelo de Blocos Estocásticos (SBM)

Em redes reais podemos observar indivíduos se organizando em forma de comunidades. Em redes sociais, por exemplo, pessoas ligadas por relações de amizades se reúnem compartilhando afinidades, se aproximam e criam laços com aqueles que são semelhantes a elas mesmas. No modelo da rede, observamos que existem conjuntos de vértices onde a densidade de arestas é maior dentro do conjunto do que entre conjuntos, dessa forma, definimos uma comunidade na rede.

Os modelos de blocos são amplamente aplicados para a e simulação de redes com estrutura de comunidades e para estudo dos agrupamentos (*clusters*) nas redes. O modelos de blocos estocástico gera grafos aleatórios com k comunidades. Definimos $N = (n_1, n_2, ..., n_k)$ um vetor de dimensão k sendo n_i o tamanho de comunidade i, e a matriz P de dimensão $k \times k$ onde $p_i j$ é a probabilidade de conexão entre as comunidades i e j e $p_i i$ é a probabilidade de formação de arestas entre membros da comunidade i.

CAPÍTULO

3

PREVISÃO DE ARESTAS

O problema de previsão de aresta parte de um momento t onde temos uma fotografia da rede. Nela observamos arestas formadas e podemos calcular métricas sobre arestas e vértices. Queremos a partir disso entender qual aresta se formará no tempo $t+1$. Para todas as análises consideramos todos os possíveis pares de arestas. A seguir serão descritos alguns métodos para previsão.

3.1 Medidas de Similaridade

Uma abordagem para a previsão de links assume que dois nós estão mais propensos a interagir se forem semelhantes. Similaridade em redes parte da hipótese de que dois nós são semelhantes se estão próximos de acordo com uma determinada distância. Essa abordagem define uma função s(x, y) que atribui uma pontuação para cada par de nós x e y. Calculamos a pontuação para cada par de nós e avaliamos as conexões mais prováveis como sendo aquelas entre pares com as maiores pontuações. A seguir são apresentadas diversas medidas de similaridade encontradas na literatura.

Vizinhos Comuns (CN). Esta medida de similaridade assume que se dois indivíduos de uma rede tem muitos vizinhos em comum, é muito provável que também estejam conectados, o que foi comprovado por (NEWMAN, 2001), apesar de simples, o método é utilizado por como base para várias outras medidas. A similaridade, definida por (LIBEN-NOWELL; KLEINBERG, 2003), segue a equação

$$s(i,j) = |\Gamma_i \cap \Gamma_j|. \tag{3.1}$$

Índice de Adamic-Adar (AA). Definida por (ADAMIC; ADAR, 2003), esta medida de similaridade estende a ideia de vizinhos comuns penalizando nós com grau mais alto. A intuição por trás dessa ideia é que a probabilidade de um amigo de uma pessoa popular ser selecionado

para ser apresentado para outro amigo é menor do que no caso de uma pessoa menos popular. A equação é dada por

$$s(i,j) = \sum_{k \in \Gamma_i \cap \Gamma_j} \frac{1}{\log |\Gamma_k|}. \tag{3.2}$$

Índice de Alocação de Recursos (RA). Motivada pelo modelo de transmissão de recursos em redes complexas (ZHOU; Lü; ZHANG, 2009), em que dois nós i e j não conectados transmitem recursos através de vizinhos. Cada vizinho de i recebe uma unidade de recurso e distribui igualmente entre seus vizinhos, a quantidade de recurso que j recebe nesse processo é a medida de similaridade entre i e j. A função de similaridade é dada por

$$s(i,j) = \sum_{k \in \Gamma_i \cap \Gamma_j} \frac{1}{|\Gamma_k|}. \tag{3.3}$$

Índice de Ligação Preferencial (PA). A ideia por trás do modelo de (BARABASI; ALBERT, 1999) define está função de similaridade. A probabilidade de dois nós formarem uma conexão é proporcional ao grau dos mesmos. É o conceito do *"the rich get richer"* que baseou o modelo de livre escala de Barabási-Albert e que representa bem as redes reais. A similaridade entre i e j é calculada

$$s(i,j) = |\Gamma_i||\Gamma_j|. \tag{3.4}$$

Coeficiente de Jaccard (JC). Função de similaridade clássica proposta por Paul Jaccard em 1901 na área de recuperação de informação (JACCARD, 1901), é dada pela razão entre vizinhos comuns de dois nós e o número de possíveis vizinhos comuns, ou seja

$$s(i,j) = \frac{|\Gamma_i \cap \Gamma_j|}{|\Gamma_i \cup \Gamma_j|}. \tag{3.5}$$

Índice de Salton (SA). Relacionada ao Coeficiente de Jaccard, é também conhecido como similaridade do cosseno (SALTON; MCGILL, 1986). A similaridade é definida pela seguinte equação

$$s(i,j) = \frac{|\Gamma_i \cap \Gamma_j|}{\sqrt{|\Gamma_i||\Gamma_j|}}. \tag{3.6}$$

Índice de Sørensen (SO). Amplamente utilizada em redes ecológicas, desenvolvida pelo botânico Thorvald Sørensen (SØRENSEN, 1948), é dada pela equação

$$s(i,j) = \frac{2|\Gamma_i \cap \Gamma_j|}{|\Gamma_i| + |\Gamma_j|}. \tag{3.7}$$

Índice de Hub *Promovido (HP)* Advinda dos estudos de redes metabólicas em (RAVASZ *et al.*, 2002), esta medida de similaridade promove a ligação entre *hubs* e nós de menor grau. É definida por

$$s(i,j) = \frac{|\Gamma_i \cap \Gamma_j|}{min(|\Gamma_i|, |\Gamma_j|)}. \tag{3.8}$$

Índice de Hub *Rebaixado (HD)* Análoga ao Índice de *Hub* Promovido, porém favorece a formação de arestas entre vértices de rau semelhante. *Hubs* se ligam preferencialmente a *Hubs* e vértices de grau baixo se ligam entre si. A função de similaridade é dada por

$$s(i,j) = \frac{|\Gamma_i \cap \Gamma_j|}{max(|\Gamma_i|, |\Gamma_j|)}. \tag{3.9}$$

Índice Local de Leich-Holme-Newman (LL) Este índice atribui maior similaridade aos pares de nós que tem mais vizinhos em comum em relação ao número esperado de vizinhos em comum. Definido em (LEICHT; HOLME; NEWMAN, 2006) é calculado por

$$s(i,j) = \frac{|\Gamma_i \cap \Gamma_j|}{|\Gamma_i||\Gamma_j|}. \tag{3.10}$$

Índice de Atração Individual (IA) Esta medida considera que dois vértices tem maior probabilidade de estarem conectados se seus vizinhos em comum também estão conectados entre si (Dong *et al.*, 2011). A função de similaridade em sua versão de menor complexidade computacional é dada por

$$s(i,j) = \sum_{k \in \Gamma_i \cap \Gamma_j} \frac{|e_{\Gamma_i \cap \Gamma_j}| + 2}{|\Gamma_k||\Gamma_i \cap \Gamma_j|}, \tag{3.11}$$

onde $|e_{\Gamma_i \cap \Gamma_j}|$ é o número de arestas que conectam os vizinhos em comum de i e j.

CAR-Based Index. Pensando em comunidades locais (CANNISTRACI; ALANIS-LOBATO; RAVASI, 2013), a similaridade atribuída à dois nós aumenta quando seus vizinhos em comum são fortemente interconectados. Neste trabalho utilizamos as seguintes variações:

- CAR-CN. CAR-Based Index baseado na medida de vizinhos comuns (CN).

$$s(i,j) = \sum_{k \in \Gamma_i \cap \Gamma_j} 1 + \frac{|\Gamma_i \cap \Gamma_j \cap \Gamma_k|}{2}. \tag{3.12}$$

- CAR-RA. A Variação do CAR-Based Index utilizando a medida de alocação de recursos (RA) é dada por

$$s(i,j) = \sum_{k \in \Gamma_i \cap \Gamma_j} \frac{|\Gamma_i \cap \Gamma_j \cap \Gamma_k|}{|\Gamma_k|}. \tag{3.13}$$

Caminho Mínimo Negado (NSP). O caminho mínimo negado (LIBEN-NOWELL, 2005), assume que a similaridade entre dois vértices é maior quanto menos for o caminho entre eles, ou seja

$$s(i,j) = -|d_{ij}|. \tag{3.14}$$

3.2 *Embeddings*

Toda tarefa de predição necessita a definição de um conjunto de atributos pelos quais podemos descrever os objetos nos quais queremos aplicar a predição. Nas seções anteriores foram apresentados diversas métricas que podem ser utilizadas como atributos ao descrever um vértice em uma rede, criando um conjunto de dados estruturados onde podemos aplicar a maioria dos métodos de aprendizado de máquina.

Todo processo de definição e cálculo desses atributos pode consumir muito tempo. Querendo evitar este processo, nessa seção apresentamos a técnica de *Node Embedding*, cujo objetivo é criar automaticamente uma representação dos nós, mapeando-os como um vetor de d dimensões e definindo as coordenadas de forma que algo da rede seja preservado. O aprendizado de características deve ser eficiente, independente da definição de métricas e resultar em uma estrutura na qual possa ser aplicados algoritmos de classificação, predição e agrupamento.

A tarefa do *Node Embedding* é criar uma representação dos nós em um espaço de baixa dimensão codificando informações sobre a rede de forma que a similaridade entre dois vértices seja preservada no mapeamento. A similaridade na rede na rede pode ter diversos significados, como foi visto na seção anterior podemos definir a função de similaridade de várias formas, aqui consideraremos similares os vértices com alta probabilidade de estarem presentes em uma curta caminhada aleatória pela rede. Caminhadas aleatórias incorporam informações tanto locais quanto globais da rede e são mais eficientes que muitas das medidas de similaridade pois não precisam ser calculadas para todos os pares vértices, apenas dos que aparecem no caminho.

Portanto, seja z_u o mapeamento do vértice u no espaço d-dimensional, queremos definir z_u tal que

$$P(v|u) \approx z_u^\top z_v, \tag{3.15}$$

sendo $P(v|u)$ a probabilidade de uma caminhada aleatória iniciada no vértice u passar pelo vértice v e $z_u^\top z_v$ o produto escalar entre os vetores z_u e z_v que mede a similaridade no espaço do mapeamento. Para isto simulamos caminhadas aleatórias de tamanho fixo usando alguma estratégia R e começando de cada nó da rede. Para cada vértice u definimos $N_R(u)$, o conjunto de

vértices visitados na caminhada aleatória iniciada em u. Determinamos z_u minimizando a função

$$\mathscr{L} = \sum_{u \in V} \sum_{v \in N_R(u)} -\log(P(v|z_u)), \tag{3.16}$$

parametrizamos $P(v|z_u)$ pela função softmax

$$P(v|z_u) = \frac{exp(z_u^\top z_v)}{\sum_{n \in V} exp(z_u^\top z_n)}, \tag{3.17}$$

pois dessa forma garantimos que o valor se aproximará de 1 quanto mais similares forem z_u e z_v. Como estratégia para a simulação da caminhada aleatória, utilizaremos o método Node2Vec que será descrito a seguir.

3.2.1 Node2Vec

O objetivo do Node2Vec (GROVER; LESKOVEC, 2016) é mapear vértices com vizinhanças similares de forma que se mantenham similares no mapeamento. A ideia por trás do algoritmo é que a precisamos de uma noção flexível de vizinhança para que possamos identificar diferentes tipos de similaridade. Por exemplo, na figura **??** observamos que os vértices u e v_1 participam da mesma comunidade altamente interconectada, por outro lado, o vértice v_6 assume o mesmo papel de centro de uma comunidade que o nó u, queremos ser capazes de captar ambas as similaridades.

Diante disso Node2Vec propõe uma estratégia de caminhada aleatória enviesada de segunda ordem como estratégia R para calcular $N_R(u)$. Com ela podemos equilibrar a caminhada entre uma exploração local e global. A exploração local se dá por uma busca em largura e a global por busca em profundidade. A caminhada é dita de segunda ordem porque tem memória de um passo, lembra qual o nó anterior. Para atingir o equilíbrio entre exploração global e local, são definidos dois parâmetros, p e q, $\frac{1}{p}$ é a probabilidade de retornar ao nó anterior e $1q$ determina as probabilidades de aumentar a distância do nó anterior. Quando determinamos p com valor baixo, aumentamos a probabilidade de busca em largura, da mesma forma quando q é pequeno estimulamos a busca profundidade.

3.3 Classificação

3.3.1 Random Forest

Árvore de decisão é um método não paramétrico aqui usado para classificação. Consiste em um grafo acíclico e direcionado cujos vértices podem ser folhas, que representam as classes, ou nós de decisão que representam um teste condicional. Baseada na estratégia de dividir para conquistar, uma árvore de decisão é construída seguindo uma estratégia gulosa onde o algoritmo

decide, baseado em uma heurística, o atributo preditivo que será usado como nó de decisão. A heurística utilizada é o teste de impureza de Gini,

$$Gini(t) = 1 - \sum_i p_i^2, \tag{3.18}$$

que determina a impureza do nó t pela a probabilidade p_i de chegar na classe i. O nó que oferecer a maior redução da impureza é escolhido. Resumidamente é decidida qual a característica vai permitir a melhor divisão das observações de maneira que os grupos resultantes sejam o mais diferente possíveis uns dos outros e os membros de cada grupo sejam similares.

Random Forest é um coleção de árvores de decisão. Cada árvore decide uma classe para determinada observação, a classe com mais votos é a classe escolhida pelo *Random Forest*.

3.3.2 Gradient Boosting

Gradient Boosting é baseado no algoritmo AdaBoost que começa treinando uma árvore de decisão em que cada observação recebe o mesmo peso. Depois da avaliação de desempenho da árvore os pesos são reavaliados de forma que observações difíceis de classificar tenham pesos maiores e as fáceis tenham pesos menores e uma segunda árvore é treinada. O modelo agora usa ambas as árvores para determinar a classe, e calcula uma nova árvore com os pesos ajustados pela avaliação dos resíduos do modelo anterior, o processo segue até determinado número de árvores ou até quando não há melhora significativa. A predição do modelo final é uma soma ponderada das predições dos modelos anteriores.

A principal diferença entre Gradient Boosting e o AdaBoost está na maneira como identificam a deficiência das classificações. Enquanto o Adaboost se baseia em observações com pesos maiores, Gradient Boosting usa o gradiente de uma função de perda que indica quão bem os coeficientes estão ajustados. A função perda é dependente do tipo de problema.

3.3.3 Multilayer Perceptron

Multilayer Perceptron é uma rede neural formada por um conjunto de neurônios artificiais. Um neurônio recebe como entrada um vetor x e resulta em um escalar y, seguindo o modelo $y = f(wx+b)$, sendo w um vetor de pesos da mesma dimensão de x e b é um escalar denominado vício. A função f é chamada de função de ativação e pode assumir diversas formas como a linear $f(u) = u$, sigmoid $f(u) = 1/(1+e^{-u})$ e degrau $f(u) = sgn(u)$ sendo sgn a função sinal. As duas últimas são principalmente utilizadas para a classificação binária.

As camadas do Multilayer Perceptron consiste em uma camada de entrada, camadas intermediárias com neurônios artificiais conectados entre si e a camada de saída. Os nerônios artificiais trabalhando em conjunto permitem resolver problemas que não são linearmente separáveis.

CAPÍTULO

4

RESULTADOS E DISCUSSÃO

4.1 Base de Dados

Nossa base de dados é composta por seis redes artificiais gerada a partir dos modelos apresentados no capítulo 2. As redes simuladas tem 600 vértices cada e variam os demais parâmetros. A citar

- Rede ER com probabilidade de conexão $p = 0.1$.
- Rede WS, referenciada por WS1, que se aproxima de uma rede regular com probabilidade de religação $p = 0.1$.
- Rede WS, referenciada por WS2, com probabilidade de religação $p = 0.4$.
- Rede BA, com grau médio $\langle k \rangle = 20$.
- Rede SBM, referenciada por SBM1, com 3 comunidades e probabilidade de ligação dentro da comunidade $p_{in} = 0.3$ e entre comunidade $p_{out} = 0.05$.
- Rede SBM, referenciada por SBM2, com 3 comunidades e probabilidade de ligação dentro da comunidade $p_{in} = 0.3$ e entre comunidade $p_{out} = 0.15$.

A tabela 1 resume as características globais das redes. As redes SBM apresentam maior densidade de conexões e maior grau médio, sendo que a maior variabilidade do grau fica com as redes BA. As redes apresentadas são dissortativas, apresentando tendência de conexão entre *hubs* e nós de baixo grau, com exceção de rede ER que é neutra. As redes com maior tendência a formação de triângulos WS1, SBM1 e SBM2.

	|E|	|V|	D	$\langle k \rangle$	k_{min}	k_{max}	r	$\langle cc \rangle$	T	$\langle l \rangle$	l_{max}
ER	17,724	600	0.0986	59.08	39	87	0.0000	0.0991	0.0992	1.9038	3
WS1	1,200	600	0.0067	4	2	7	-0.0522	0.3735	0.3522	7.7712	14
WS2	1,200	600	0.0067	4	2	9	-0.0972	0.1187	0.1017	5.4108	10
BA	5,900	600	0.0328	19.67	10	134	-0.0456	0.0836	0.0755	2.4113	4
SBM1	24,053	600	0.1339	80.18	55	108	-0.0048	0.1997	0.1994	1.8671	3
SBM2	35,998	600	0.2003	119.99	90	149	-0.0117	0.2067	0.2066	1.7997	2

Tabela 1 – Descrição das características dos modelos de redes gerados. Em ordem, número de arestas, número de vértices, densidade, grau médio, grau mínimo, grau máximo, assortatividade, média de coeficiente de aglomeração, transitividade, média de menores caminhos e diâmetro.

4.2 Medidas de Similaridade

Para avaliação do desempenho de cada medida de similaridade nas diferentes redes, foi conduzida uma validação cruzada com 5 dobras. A cada definição do conjunto de treino as medidas foram recalculadas para todos os pares de vértices. O valor associado ao par de vértices é então ordenado de forma que aqueles com maiores valores sejam escolhidos como potenciais conexões. A tabela 2 indica a precisão obtida em cada rede por cada medida de similaridade. Notamos que as medidas apresentam melhor desempenho em redes BA e SBM. Comparamos os resultados com a seleção aleatória de pares de vértices. A tabela 3 mostra a AUC, calculada pela equação

$$AUC = \frac{n' + 0.5n''}{n}, \tag{4.1}$$

sendo n o número de vezes que selecionamos um par de vértices aleatório para analisar, n' é o número de vezes que uma aresta que existe é associada ao maior valor em comparação com uma aresta faltante aleatóriamente selecionada, e n" é o número de vezes que esse valor é igual. Os resultados na tabela demonstram que a predição por medidas de similaridada não é muito melhor do que a predição puramente aleatória.

	ER	WS1	WS2	BA	SBM1	SBM2
AA	0.2037	0.4967	0.4954	0.2037	0.2153	0.2153
CC	0.2141	0.4963	0.4954	0.2141	0.2301	0.2301
CN	0.2129	0.4963	0.4954	0.2129	0.2222	0.2222
CR	0.3203	**0.5000**	**0.5000**	0.3203	0.2700	0.2700
HD	0.2780	0.4967	0.4954	0.2780	0.3579	0.3579
HP	0.3881	0.4967	0.4954	0.3881	0.3682	0.3682
IA	0.1963	0.4967	0.4954	0.1963	0.2096	0.2096
JC	0.2959	0.4967	0.4954	0.2959	0.3604	0.3604
LL	**0.4596**	0.4967	0.4954	**0.4596**	**0.4913**	**0.4913**
NSP	0.3204	0.1688	0.1842	0.3204	0.3060	0.3060
PA	0.1129	0.1600	0.1413	0.1129	0.0686	0.0686
RA	0.1999	0.4967	0.4954	0.1999	0.2119	0.2119
SA	0.3164	0.4967	0.4954	0.3164	0.3608	0.3608
SO	0.2959	0.4967	0.4954	0.2959	0.3604	0.3604

Tabela 3 – Média da AUC das 5 iterações da validação cruzada aplicada de cada medida de similaridade em cada modelo de rede.

	ER	WS1	WS2	BA	SBM1	SBM2
AA	0.0002	0.0000	0.0000	0.0025	0.0000	0.0000
CC	0.0006	0.0000	0.0000	0.0063	0.0000	0.0000
CN	0.0005	0.0000	0.0000	0.0042	0.0000	0.0000
CR	0.0006	0.0000	0.0000	0.0032	0.0000	0.0000
HD	0.0201	0.0042	0.0008	0.0312	0.0394	0.0394
HP	0.0499	0.0033	0.0008	0.0134	0.0475	0.0475
IA	0.0001	0.0000	0.0000	0.0029	0.0000	0.0000
JC	0.0218	0.0042	0.0008	0.0327	0.0462	0.0462
LL	0.0762	0.0042	0.0008	**0.0407**	**0.1930**	**0.1930**
NSP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
PA	0.0000	0.0000	0.0000	0.0042	0.0000	0.0000
RA	0.0001	0.0000	0.0000	0.0027	0.0000	0.0000
SA	0.0235	0.0033	0.0008	0.0305	0.0471	0.0471
SO	0.0218	0.0042	0.0008	0.0327	0.0462	0.0462
Aleatório	**0.0999**	**0.0067**	**0.0075**	0.0341	0.1326	0.1326

Tabela 2 – Média da precisão das 5 iterações da validação cruzada aplicada de cada medida de similaridade em cada modelo de rede.

4.3 Classificação

Para aplicação de métodos de classificação definimos todos os pares de vértices possíveis no grafo. Aqueles que possuem uma arestas são classificados com o valor 1, aqueles que não possuem com o valor 0. Devemos notar que as redes são esparsas e temos muito mais pares de nós com a classe 0. É importante executar o balanceamento das classes, selecionando apenas um subconjunto de pares de vértices da classe 0. Com o conjunto de treino e teste definidos,

	Random Forest	GBoost	MLP
Acurácia	0.5800	0.5833	-
AUC	0.6149	0.6172	-

Tabela 4 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede ER.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.4364	0.4047	0.8332	0.5448	0.4997
GBoost	0.4542	0.4329	0.8314	0.5694	0.4969
MLP	-	-	-	-	-

Tabela 5 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede ER.

	Random Forest	GBoost	MLP
Acurácia	0.6389	0.6496	-
AUC	0.7079	0.7069	-

Tabela 6 – Resultados obtidos no conjunto de treino com Embeddings na rede ER.

começamos a caracterização dos pares de vértices no conjunto de treino. Os resultados para cada rede estão expostos nas tabelas a seguir. A utilização de embeddings se demonstrou mais eficaz em redes simuladas e o Multilayer Perceptron desempenhou melhor na maioria dos casos.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.3361	0.2524	0.8375	0.3879	0.5037
GBoost	0.3731	0.3096	0.8335	0.4515	0.5001
MLP	-	-	-	-	-

Tabela 7 – Resultados obtidos no conjunto de teste com Embeddings na rede ER.

	Random Forest	GBoost	MLP
Acurácia	0.8375	0.835	-
AUC	0.9173	0.9228	-

Tabela 8 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede WS1.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.418	0.3333	0.9132	0.4884	0.5875
GBoost	0.2666	0.1483	0.8396	0.2521	0.5033
MLP	-	-	-	-	-

Tabela 9 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede WS1.

	Random Forest	GBoost	MLP
Acurácia	0.9674	0.9558	0.965
AUC	0.9915	0.9879	0.9745

Tabela 10 – Resultados obtidos no conjunto de treino com Embeddings na rede WS1.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.4986	0.4033	0.9877	0.5727	0.6891
GBoost	0.4916	0.4000	0.9756	0.5673	0.6749
MLP	0.8333	1.0	0.8333	0.9090	0.5000

Tabela 11 – Resultados obtidos no conjunto de teste com Embeddings na rede WS1.

	Random Forest	GBoost	MLP
Acurácia	0.7941	0.7958	-
AUC	0.8779	0.8825	-

Tabela 12 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede WS2.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.2833	0.1883	0.7957	0.3045	0.4733
GBoost	0.2888	0.1883	0.8188	0.3062	0.4899
MLP	-	-	-	-	-

Tabela 13 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede WS2.

	Random Forest	GBoost	MLP
Acurácia	0.9600	0.9349	0.9383
AUC	0.9867	0.9790	0.9749

Tabela 14 – Resultados obtidos no conjunto de treino com Embeddings na rede WS2.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.2930	0.1666	0.9174	0.2820	0.5458
GBoost	0.2972	0.1700	0.9272	0.2873	0.5516
MLP	0.8333	1.0	0.8333	0.9090	0.5000

Tabela 15 – Resultados obtidos no conjunto de teste com Embeddings na rede WS2.

	Random Forest	GBoost	MLP
Acurácia	0.7238	0.7179	0.7371
AUC	0.7977	0.7964	0.8171

Tabela 16 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede BA.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.5745	0.5467	0.9051	0.6817	0.6301
GBoost	0.5661	0.5345	0.9063	0.6724	0.6291
MLP	0.1666	0.0000	0.0000	0.0000	0.5000

Tabela 17 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede BA.

	Random Forest	GBoost	MLP
Acurácia	0.7584	0.7413	0.8700
AUC	0.8464	0.8168	0.9444

Tabela 18 – Resultados obtidos no conjunto de treino com Embeddings na rede BA.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.3539	0.2664	0.8646	0.4073	0.5289
GBoost	0.3768	0.2966	0.8697	0.4423	0.5372
MLP	0.3471	0.2677	0.8395	0.4060	0.5059

Tabela 19 – Resultados obtidos no conjunto de teste com Embeddings na rede BA.

	Random Forest	GBoost	MLP
Acurácia	0.6628	0.6658	-
AUC	0.7181	0.7224	-

Tabela 20 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede SBM1.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.6451	0.6424	0.9039	0.7510	0.6505
GBoost	0.6191	0.6102	0.9007	0.7275	0.6369
MLP	-	-	-	-	-

Tabela 21 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede SBM1.

	Random Forest	GBoost	MLP
Acurácia	0.7072	0.6850	0.7750
AUC	0.7756	0.7482	0.8592

Tabela 22 – Resultados obtidos no conjunto de treino com Embeddings na rede SBM1.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.5726	0.5330	0.9208	0.6751	0.6519
GBoost	0.5537	0.5177	0.9066	0.6591	0.6256
MLP	0.8333	1.0	0.8333	0.9090	0.5000

Tabela 23 – Resultados obtidos no conjunto de teste com Embeddings na rede SBM1.

	Random Forest	GBoost	MLP
Acurácia	0.5575	0.5585	-
AUC	0.5833	0.5847	-

Tabela 24 – Resultados obtidos no conjunto de treino com medidas de similaridade na rede SBM2.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.4784	0.4668	0.8342	0.5986	0.5015
GBoost	0.4749	0.4636	0.8319	0.5954	0.4975
MLP	-	-	-	-	-

Tabela 25 – Resultados obtidos no conjunto de teste com medidas de similaridade na rede SBM2.

	Random Forest	GBoost	MLP
Acurácia	0.5918	0.5852	-
AUC	0.6372	0.6210	-

Tabela 26 – Resultados obtidos no conjunto de treino com Embeddings na rede SBM2.

	Acurácia	Precisão	Recall	F1	AUC
Randon Forest	0.4262	0.3338	0.8426	0.4782	0.5110
GBoost	0.4262	0.3817	0.8446	0.5258	0.5152
MLP	-	-	-	-	-

Tabela 27 – Resultados obtidos no conjunto de teste com Embeddings na rede SBM2.

CAPÍTULO

5

CONCLUSÃO

A previsão de arestas em redes é uma tarefa desafiadora. Neste trabalho pudemos comparar diferentes medidas de similaridade usadas na previsão da arestas e avaliar o desempenho das previsões feitas a partir de medidas predefinidas em comparação com a caracterização automática dos *Embeddings*. Observamos que os métodos de classificação são muito mais efetivos que métodos que consideram apenas as medidas de similaridade e que, em particular no caso das redes simuladas, os Embedding apresentam melhor desempenho.

A estrutura da rede gerada interfere na previsão de arestas, sendo que aquelas que apresentam livre-escala ou estrutura de comunidades, aproximando-se mais de redes reais, apresentam melhor desempenho que aquelas cujas arestas são formadas aleatoriamente.

REFERÊNCIAS

ADAMIC, L. A.; ADAR, E. Friends and neighbors on the web. **Social Networks**, v. 25, n. 3, p. 211 – 230, 2003. ISSN 0378-8733. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0378873303000091>. Citado na página 29.

ALBERT, R. Emergence of scaling in random networks. **Science**, v. 286, n. 5439, p. 509–512, 1999. Citado na página 23.

BARABÁSI, A. The architecture of complexity. **IEEE Control Systems**, v. 27:4, p. 33–42, 2007. Citado na página 23.

BARABASI, A.-L.; ALBERT, R. Emergence of scaling in random networks. **Science**, v. 286, n. 5439, p. 509–512, 1999. Disponível em: <http://www.sciencemag.org/cgi/content/abstract/286/5439/509>. Citado nas páginas 28 e 30.

CANNISTRACI, C. V.; ALANIS-LOBATO, G.; RAVASI, T. From link-prediction in brain connectomes and protein interactomes to the local-community-paradigm in complex networks. **Scientific Reports**, Springer Science and Business Media LLC, v. 3, n. 1, apr 2013. Disponível em: <https://doi.org/10.1038%2Fsrep01613>. Citado na página 31.

Dong, Y.; Ke, Q.; Wang, B.; Wu, B. Link prediction based on local information. In: **2011 International Conference on Advances in Social Networks Analysis and Mining**. [S.l.: s.n.], 2011. p. 382–386. Citado na página 31.

ERDÖS, P.; RÉNYI, A. On random graphs, i. **Publicationes Mathematicae (Debrecen)**, v. 6, p. 290–297, 1959. Citado na página 27.

GROVER, A.; LESKOVEC, J. node2vec: Scalable feature learning for networks. **CoRR**, abs/1607.00653, 2016. Disponível em: <http://arxiv.org/abs/1607.00653>. Citado na página 33.

JACCARD, P. Etude de la distribution florale dans une portion des alpes et du jura. **Bulletin de la Societe Vaudoise des Sciences Naturelles**, v. 37, p. 547–579, 01 1901. Citado na página 30.

LEICHT, E. A.; HOLME, P.; NEWMAN, M. E. J. Vertex similarity in networks. **Phys. Rev. E**, American Physical Society, v. 73, p. 026120, Feb 2006. Disponível em: <https://link.aps.org/doi/10.1103/PhysRevE.73.026120>. Citado na página 31.

LIBEN-NOWELL, D. **An Algorithmic Approach to Social Networks**. Tese (Doutorado), USA, 2005. AAI0808764. Citado na página 32.

LIBEN-NOWELL, D.; KLEINBERG, J. The link prediction problem for social networks. In: **Proceedings of the Twelfth International Conference on Information and Knowledge Management**. New York, NY, USA: Association for Computing Machinery, 2003. (CIKM '03), p. 556–559. ISBN 1581137230. Disponível em: <https://doi.org/10.1145/956863.956972>. Citado na página 29.

NEWMAN, M. Newman, m.e.j.: Clustering and preferential attachment in growing networks. phys. rev. e 64, 025102. **Physical review. E, Statistical, nonlinear, and soft matter physics**, v. 64, p. 025102, 09 2001. Citado na página 29.

RAVASZ, E.; SOMERA, A.; MONGRU, D.; OLTVAI, Z.; BARABÁSI, A. Hierarchical organization of modularity in metabolic networks. **Science**, American Association for the Advancement of Science, v. 297, n. 5586, p. 1551–1555, ago. 2002. ISSN 0036-8075. Copyright: Copyright 2008 Elsevier B.V., All rights reserved. Citado na página 31.

SALTON, G.; MCGILL, M. J. **Introduction to Modern Information Retrieval**. USA: McGraw-Hill, Inc., 1986. ISBN 0070544840. Citado na página 30.

SØRENSEN, T. **A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and Ist Application to Analyses of the Vegetation on Danish Commons**. Munksgaard, 1948. (Det Kongelige Danske Videnskabernes Selskab). Disponível em: <https://books.google.com.br/books?id=2cdDmwEACAAJ>. Citado na página 30.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of 'small-world' networks. **Nature**, v. 393, n. 6684, p. 440–442, jun. 1998. ISSN 0028-0836. Disponível em: <http://dx.doi.org/10.1038/30918>. Citado nas páginas 23 e 27.

ZHOU, T.; Lü, L.; ZHANG, Y.-C. Predicting missing links via local information. **The European Physical Journal B - Condensed Matter and Complex Systems**, v. 71, p. 623–630, 10 2009. Citado na página 30.


ICMC
USP
SÃO CARLOS