



Universidade Federal de São Carlos
MBA em Machine Learning in Production
Turma C - Time B
Disciplina: Algoritmos e Estruturas de Dados no Big Data

Atividade: ESBD1_2

Representação e Manipulação de Redes

Alunos:

João Felipe Moura
Paulo Daniel Nobre Ferreira
Ricardo José Campelo Arruda Júnior
Rodolfo Arruda
Valter Alberto Melgarejo Martins
Victor Hugo

2.1. Situação Problema da Unidade 2

Graus de Separação em Redes Sociais: considere o grafo de uma rede social, em que cada perfil corresponde a um vértice, e existe uma aresta entre dois vértices apenas se os perfis correspondentes são amigos. Queremos saber qual o grau de separação médio entre dois perfis da rede. Isto é, saber quantas arestas tem, em média, um caminho mais curto que conecta dois perfis. Para tanto, elaboramos o seguinte experimento.

Temos uma função de construção que recebe como entrada um # (número) de vértices, um # de arestas, e que usa esses parâmetros para gerar aleatoriamente o grafo de uma rede. Também temos uma função de testes que recebe o grafo gerado anteriormente, realiza uma bateria de 100 testes, calcula a média dos resultados dos testes, e devolve o grau de separação médio encontrado. Em cada uma das 100 iterações a função de testes sorteia dois perfis (vértices) da rede e chama uma função de busca para calcular o grau de separação dos vértices sorteados.

No entanto, essa função de busca não está implementada. *Sua missão é implementá-la, realizar os testes para cada par (# de vértices, # de arestas) indicados na tabela a seguir, e preencher cada célula da tabela com o valor da média dos graus de separação encontrado.* Atente que, na tabela estão indicados os # de arestas por vértice, mas a função de construção recebe o número total. Com a tabela preenchida, *analise brevemente como cresce o grau de separação médio em função do # de vértices e de arestas do grafo.*

Tabela auxiliar para calcular o número total de arestas.

	# Total de arestas em cada cenário		
# Vértices (n)	5	raíz (n)	n / 5
100	500	1000	2000
1000	5000	31623	200000
10000	50000	1000000	20000000
100000	500000	31622777	2000000000

	# Médio de Arestas por Vértice		
# Vértices (n)	5	raíz (n)	n / 5
100	2,23	1,83	1,58
1000	3,25	1,94	1,59
10000	4,25	2,0	1,7
100000	5,18	2,01	Não finaliza

Código com a função de busca implementada:

https://github.com/rodolfoarruda/MLP-TurmaC/blob/main/ESBD1_2.py

Outra implementação, utilizando a biblioteca networkx

https://github.com/paulodnobre/mlp-c/blob/main/ESBD1_2.py

https://github.com/joaofmoura/MLP-ITIUFSCar/blob/main/ESBD1_2.ipynb

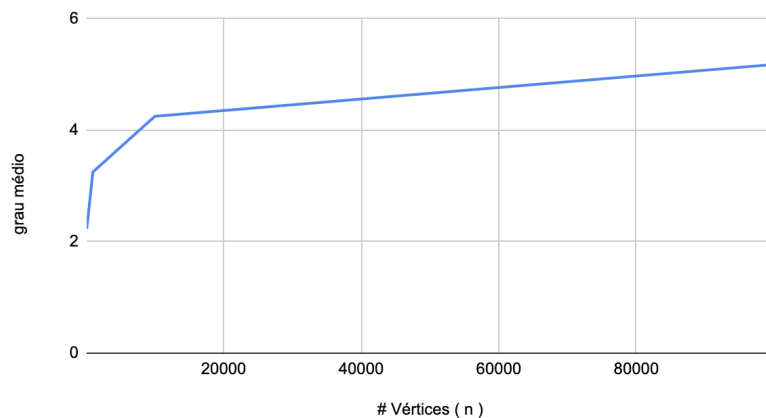
Planilha com os resultados obtidos usando o código acima:

https://docs.google.com/spreadsheets/d/1KwpjOjshJiyEr6vjAWcYwVqi_9IMCf9uKESBJL_LRSmk/edit?usp=sharing

Pergunta: Com a tabela preenchida, *analise brevemente como cresce o grau de separação médio em função do # de vértices e de arestas do grafo.*

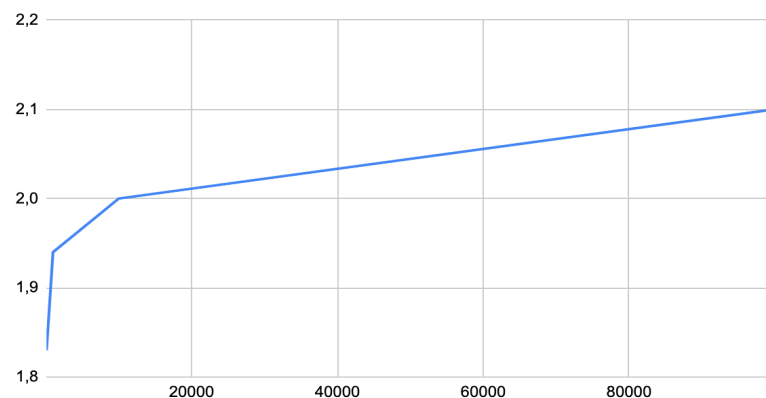
A primeira coluna reflete um grafo esparsos em que o número de arestas acompanha linearmente o número de vértices. Com isso, o grau de separação médio em função do # de vértices segundo o comportamento do gráfico indica ser logarítmico.

Grau de separação médio em função do # de vértices



A coluna do meio reflete uma situação intermediária em que o número de arestas cresce a uma proporção $n \cdot \sqrt{n}$ com relação ao número de vértices n . Com isso, o grau de separação médio em função do # de vértices segundo o comportamento do gráfico indica ser logaritmo também.

Grau de separação médio em função do # de vértices



A última coluna é um grafo denso em que as arestas crescem proporcionalmente a n ao quadrado com relação ao número de vértices. Não foi possível estimar o último ponto pela complexidade envolvida!

Além disso, é possível perceber que mantendo o número de nós fixo, a relação entre o grau de separação médio, em função do número médio de arestas por vértices é inversamente proporcional. Ou seja, à medida que aumentamos o número de arestas em um grafo o caminho médio diminui.

Realizados os testes anteriores, temos mais um desafio. Considere agora a versão alternante do grau de separação (distância) entre dois perfis da rede. Na versão alternante só são considerados caminhos em que sucedendo um perfil masculino vem um perfil feminino e vice-versa. Faça uma nova versão da sua função de busca para considerar apenas caminhos alternantes, refaça os testes usando essa nova função e preencha a próxima tabela com os graus médios de separação encontrados. *Como essa versão do problema afetou os graus de separação?*

	# Médio de Arestas por Vértice		
# Vértices (n)	5	raíz (n)	n / 5
100	2,52	1,89	1,51
1000	4,0	2,09	1,69
10000	6,02	2,08	1,51
100000	Retorna None	2,09	Não finaliza

Esta alteração causa um aumento no grau médio de separação para o caso em que a rede é esparsa, devido ao fato de que as restrições de alternâncias torna mais restrito os caminhos a serem percorridos. Para as redes das colunas 2 e 3, não é perceptível mudanças significativas nesta métrica uma vez que elas são redes já consideradas com uma densidade alta de conexões.

Código com a função de busca implementada:

https://github.com/rodolfoarruda/MLP-TurmaC/blob/main/ESBD1_2_bonus.py