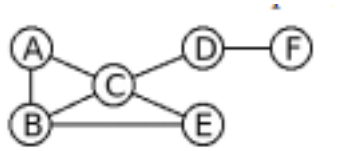
Homework 1: Key Network Properties, Graph Models and Gephi

Diana Egas up201604621, João Neto up201605883

28 de março de 2021

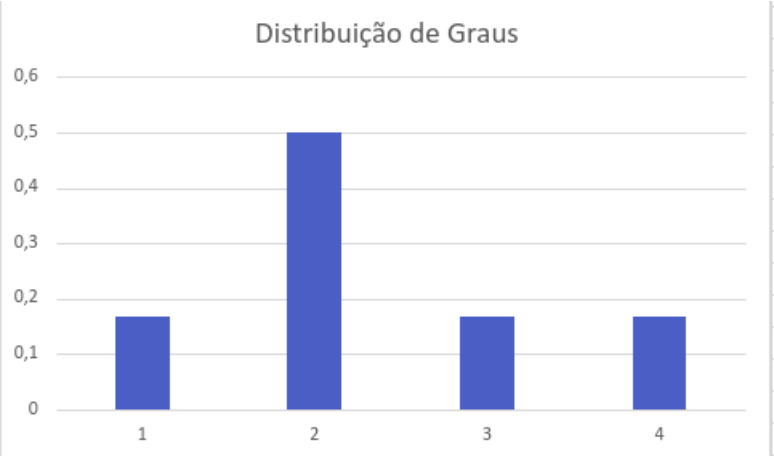
Exercise 1



1.a

Nó	Grau
A	2
B	3
C	4
D	2
E	2
F	1

Para normalizar os valores dados pelos graus dos nós, usa-se a notação $P(k)$ em que $P(k) = \frac{Nk}{N}$. Usaremos essa noção para construir o gráfico relativo à distribuição normalizada da distribuição dos graus dos nós.



1.b

Diâmetro: Distância máxima entre dois pares de nós num grafo.
De forma a tornar mais fácil o processo de cálculo do diâmetro, podemos ter em consideração que por se tratar de um grafo conexo e ao mesmo tempo completo, então o **diâmetro(G) > 1**. Reduzindo o conjunto de dados a considerar. Apresentamos os valores na tabela que se segue.

Nós	Distância
(A,E)	2
(A,D)	2
(A,F)	3
(B,D)	2
(B,F)	3

Concluímos que **diâmetro(G) = 3**

Comprimento médio do caminho: Número médio de passos relativamente aos caminhos mais curtos para todos os possíveis nós da rede. Medida de eficiência sobre a informação de uma rede.

$$\bar{h} = \frac{1}{2E_{max}} \sum_{i,j \neq i} h_{ij}$$

hij: distância de um nó i a j
Emax: número máximo de arestas $n(n-1)/2$
Para auxiliar na resolução do problema foi construída uma matriz de distâncias.

	A	B	C	D	E	F	soma
A	-	1	1	2	2	3	9
B	1	-	1	2	1	3	8
C	1	1	-	1	1	2	6
D	1	1	-	1	1	1	5
E	2	2	1	-	2	1	8
F	3	3	2	1	3	-	12

Com os valores dados pela coluna da soma temos que:

$\sum_{i,j \neq i} h_{ij} = (9 + 8 + 6 + 5 + 8 + 12) = 48$

Substituindo na fórmula:

$$\bar{h} = \frac{1}{2 \times (6-1)} \times 48 = 0,8$$
$$\bar{h} = 0,8$$

1.c

Local clustering coefficient: Porção de nós conectados.
$C_i = \frac{2e_i}{k_i(k_i - 1)}$

ei:número de arestas entre os nós de i
ki: grau do nó i

O número total de **nós** de G são **6**, escolhemos o nó com maior número de vizinhos para demonstrar este resultado, assumimos C, para o qual existem 4 vizinhos. Dessa forma, o número total de **conexões** é dado por $C_2^4 = \frac{4(4-1)}{2} = 6$.
Para o numerador, o número de arestas entre os vizinhos do nó e_i, é 2 . Assim o **resultado final** é dado por $C_i = \frac{2}{6}$

Average local clustering coefficient: Mede o grau com que os nós do grafo tendem a agrupar-se.
$C = \sum_{i=0}^{n} Ci$

Ci: Local clustering coefficient

Já sabemos que G tem 6 nós, ou seja, $N = 6$. Vamos agora para cada nó calcular o *local clustering coefficient*, os valores serão apresentados de seguida na tabela.

Nós	Ci
A	1
B	$\frac{1}{3}$
C	$\frac{2}{6}$
D	0
E	$\frac{1}{3}$
F	0

Resultado: $C = \frac{1}{6} \times (1 + \frac{1}{3} + \frac{2}{6} + \frac{1}{3}) = 0,33(3)$

1.d

Betweenness centrality (normalized): Número de caminhos mais curtos de um nó para quaisquer outros no grafo, mede a importância de cada nó.

$$C_B(i) = \sum_{j<k} \frac{gjk(i)}{gjk}$$

A fórmula normalizada é dada por:

$$C'_B(i) = \frac{CB(i)}{(n-1)\frac{(n-2)}{2}}$$

gjk:número de caminhos mais curtos que conectam j e k.

gjk(i): número em que nó i está

Para o nó B sabemos que o único caminho mais curto que passa por ele é : (A,E) então $C_B(B) = 1$
Em relação a C, os caminhos (A,D), (A,F),(B,D),(B,F),(E,D),(E,F), devem ser considerados e por isso $C_B(C) = 6$
(C,F), (A,F),(E,F),(B,F), devem ser os caminhos considerados quando temos em consideração o nó D dessa forma $C_B(D) = 4$.
Por fim, tanto A como F têm 0 como valor de centralidade.
Normalizando os valores apresentados, vem que $C'_B = \frac{11}{5 \times \frac{4}{2}} = \frac{11}{10}$

Closeness centrality (normalized): Trata-se da distância média de um vértice a todos os outros. É dado pela seguinte fórmula:
$C_C(i) = \frac{1}{\sum_{j=1}^{N}(i,j)}$
Utilizámos a matriz de distâncias presente em 1 b), recorrendo a esses dados conseguimos obter a *Closeness centrality* de cada nó.
$C_C(A) = \frac{1}{9};\ C_C(B) = \frac{1}{8};\ C_C(C) = \frac{1}{6};\ C_C(D) = \frac{1}{8};\ C_C(E) = \frac{1}{9};\ C_C(F) = \frac{1}{12}$

Normalizando estes valores através da fórmula:
$C'_C(i) = C_C(i) \times (n-1)$, obtemos:

$C'_C(A) = \frac{5}{9};\ C'_C(B) = \frac{5}{8};\ C_C(C) = \frac{5}{6};\ C'_C(D) = \frac{5}{8};\ C'_C(E) = \frac{5}{9};\ C'_C(F) = \frac{5}{12}$

Exercise 2

Tendo em conta que o *local clustering* se trata da fração de conexões aos vizinhos sobre todas as possíveis conexões entre eles, podemos apontar algumas limitações, como o facto de não relacionar os graus dos nós , não considerando por isso o seu peso, duas redes com a mesma topologia poderiam ser tratas de forma igual sem necessariamente serem. Então, o *average cluster coefficient* será uma métrica limitada. A métrica que dá maior peso a nós com maior grau é o *global clustering*, este modelo é baseado em transitividade, detetando a fração de tripletos, uma vez que não é calculado para cada nó, não é sensível a diferentes graus, neste caso atribui mais peso aos triângulos.

Exercise 3

Para este exercício foi utilizado a ferramenta de análise de redes, Gephi. A rede foi importada como um grafo dirigido, dependendo das alíneas, a estratégia de *merging* variou entre *sum* e *not merging*.

a) O número de aeroportos é representado pelo número de nós que são 3147 e o número de voos estão representados por arestas que são 66679. Estes valores são apresentados na janela de "contex"do gephi.

b) O número médio de voos, *outgoing*, pode ser calculado através da metade *average degre* que é 10.594. Este valor pode ser obtido dado que cada voo tem uma partida e uma chegada, que corresponde a uma aresta entre nós. O valor deverá ser metade do *average degree* que representa a média das partidas.

c) O diâmetro da rede é 13 e o *average path length* é 3.9688. Estes valores são obtidos através das funcionalidades de estatística.

d) O par de aeroportos com mais voos entre si corresponde ao aeroportos de "Chicago O'Hare International Airport "e "Hartsfield Jackson Atlanta International Airport "com 39 voos entre eles. Estes dados foram obtidos na janela "Data Laboratory"resultando os dois primeiros casos da lista ordenada por peso das arestas.

e) Os 3 aeroportos com voos para o maior número de aeroportos são: Frankfurt am Main Airport, Charles de Gaulle International Airport, Amsterdam Airport Schiphol. Estes dados foram obtidos através da divisão "Data Laboratory", do qual resultam os três primeiros casos da lista ordenada por "out-degree"dos nós.

f) Os 3 aeroportos com voos com maior *normalized betweenness centrality* são: Charles de Gaulle International Airport, Los Angeles International Airport e o Dubai International Airport. Estes dados foram obtidos com a mesma estratégia anterior mas ordenados por *normalized betweenness centrality* .

g) O "Ted Stevens Anchorage International Airport"é o 5ª(quinto) no *rankig* de *betweenness centrality* e o 1844º no *rankig* de *out-degree*. Um aeroporto com o mesmo comportamento será por exemplo o Faa'a International Airport com *outdegree* de 27 e *betweenness centrality* 164470.566329, consideramos que isto acontece porque apesar não haver muitos voos a sair do aeroporto, trata-se de um ponto intermédio, através do qual cruzam vários outros voos.

h) O top 3 de países com maior número de aeroportos são: United States of America, Canada e China. Estes dados foram obtidos através da seguinte sequência de ações nos painéis "Overview Panel - Appearance - Partition- Country".

i) O top 3 de companhias com maior número de voos são: Ryanair(FR), American Airlines(AA) e United Airlines(UA). Estes dados foram obtidos através da sequência de painéis "Overview Panel - Appearance- Partition - Airlines", aplicando a estratégia *no merge* na importação da rede.

j) Usando a *query* de *Inter-Edges*, selecionando USA obtivemos o número de voos, 10487.

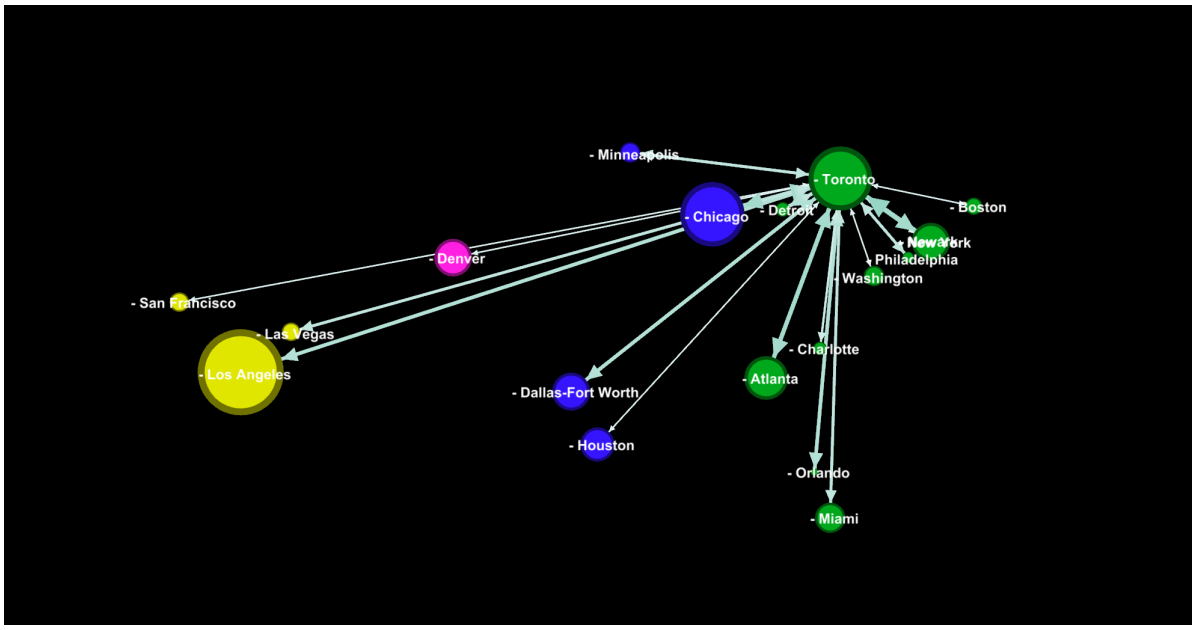
k) Foi usada a *query* para selecinar os aeroportos com "out-going degree"superiores a 50, concluímos que na China tem 21 aeroportos que tem pelo menos 50 outros aeroportos como destino.

l) Há 24 voos entre Portugal e Brasil, conseguimos este valor através do filtro de voos.

m) O componente gigante contém 173 nós e 2119 voos, o nó com mais importância de *closeness centrality* é Londres, aeroporto London Stansted Airport. Estes dados foram obtidos com o menu de estatística, selecionando "Filters - Topology - Giant Component"depois combinado com o filtro de seleção de voos da companhia Rayanair.

n) Usando a ferramenta "Ego-Network", com partida em "Francisco de Sá Carneiro Porto Airport"é possível chegar com 1 voo a 61 aeroportos, com 2 voos a 755 aeroportos e com 3 voos a 2376 aeroportos.

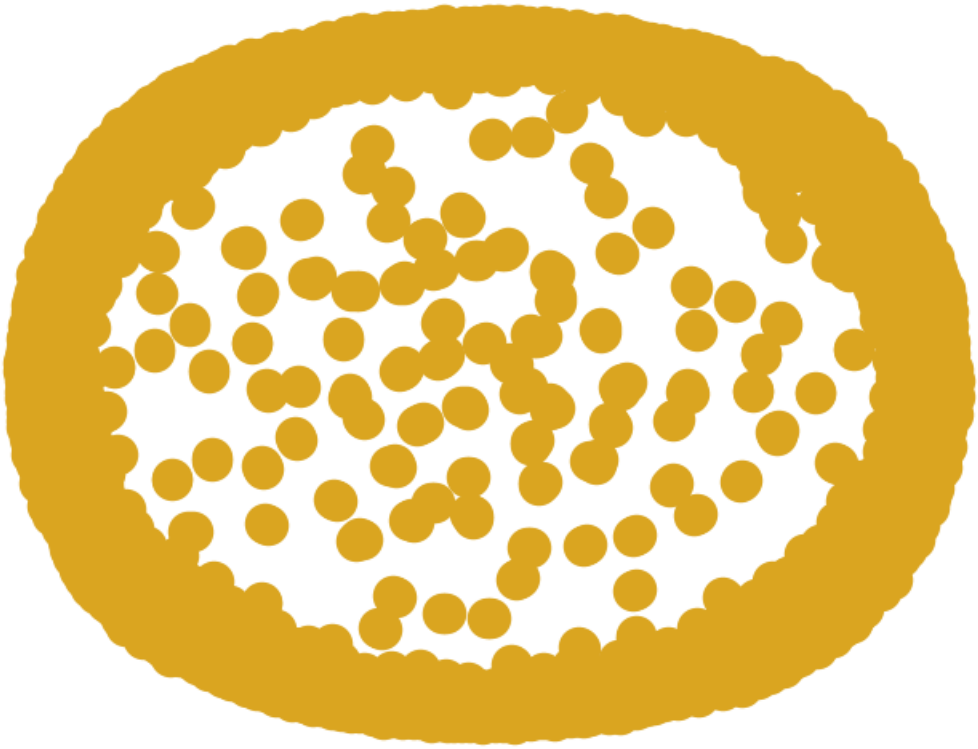
o) A figura seguinte mostra a rede entre os aeroportos de Canadá e América com mais de 100 destinos no mundo. O tamanho dos nós reflete a *betweenness centrality* global e as cores representadas por diferentes *time zones*.



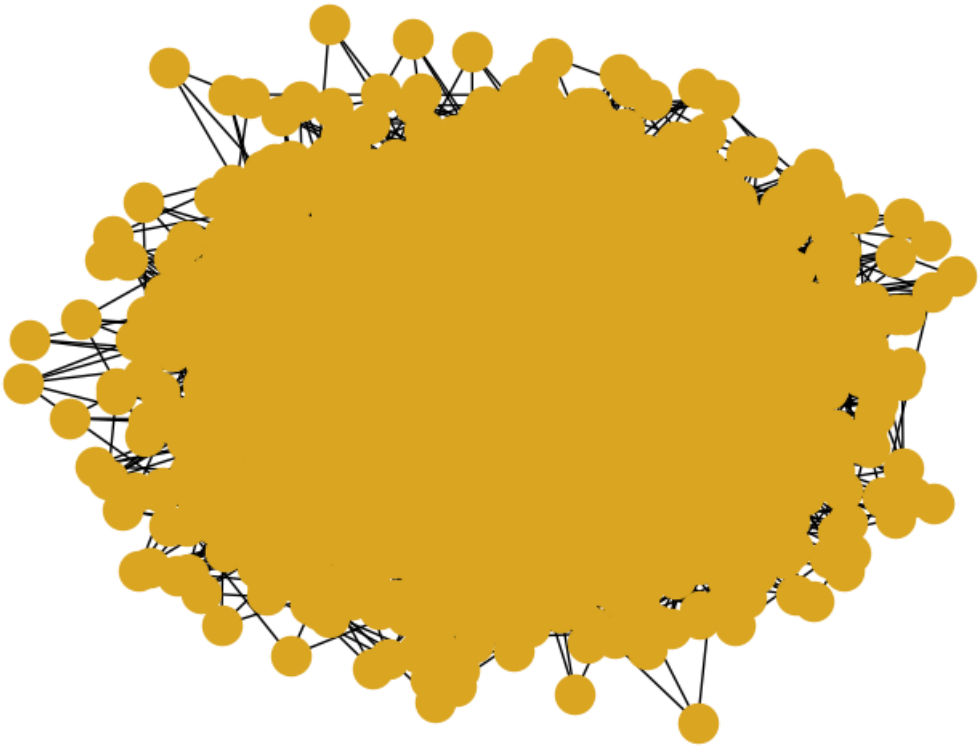
Rede resultante da pergunta 3 o)

Exercise 4

Para este exercício foi usada a linguagem python, o código encontra-se no zip com o nome randomNetwork.py.
Os grafos das redes construídas são apresentados de seguida, sendo que estão também incluídos no zip.



n=2000 p= 0.0001



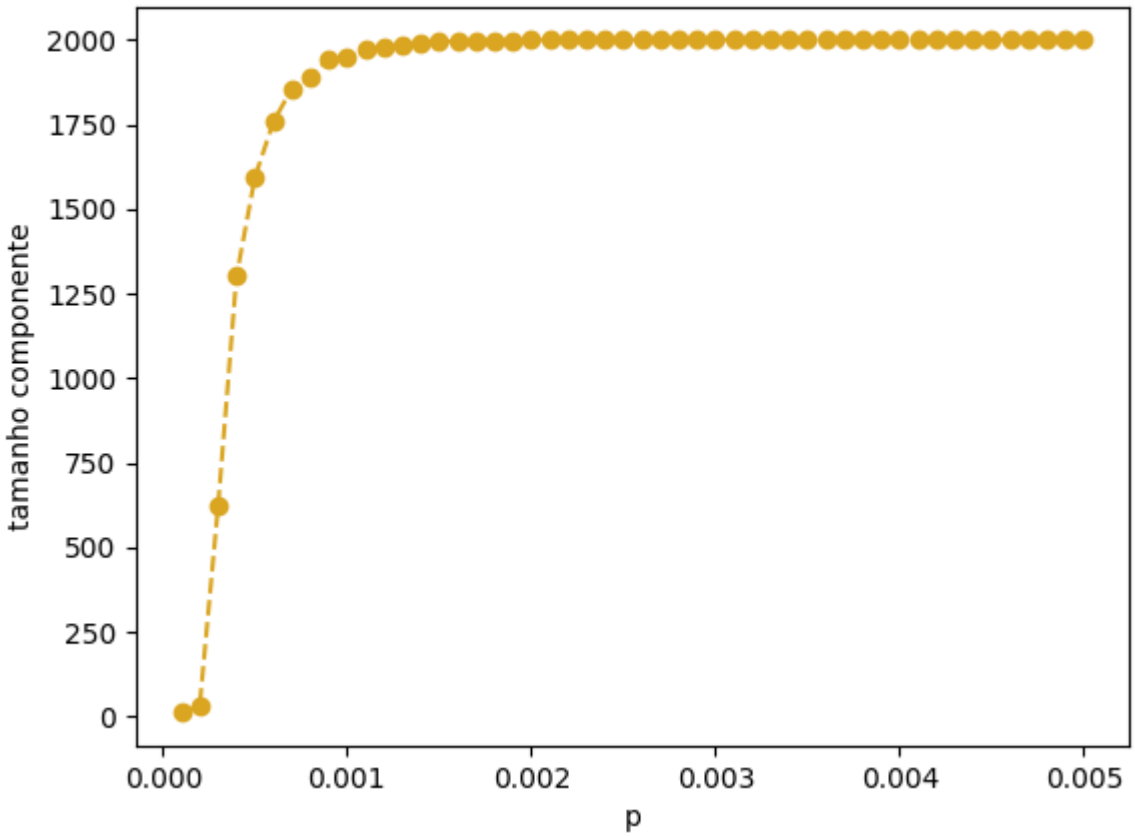
n=2000 p= 0.005

Exercise 5

Para a resolução do problema, usámos uma função da biblioteca networkx, que calcula a *Giant component* fazendo depois *print* do tamanho da maior. Para random1.txt a maior componente é **4** e para random2.txt a maior componente é **2000**. O código encontra-se no zip com o nome giantComp.py

Exercise 6

O código usado encontra-se no zip com o nome someRandomNetwork.py. O gráfico com os valore encontra-se a baixo.



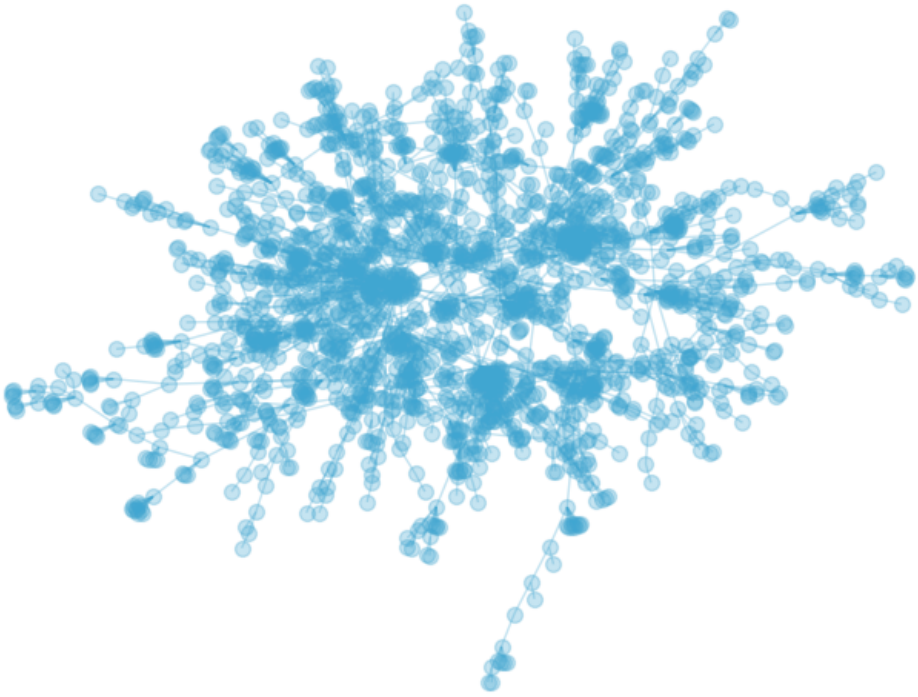
Tendo em consideração o Modelo Erdos-Renyi, em que cada aresta se liga, com probabilidade p, formando um par de nós, podemos interpretar o gráfico resultante. É expectável que à medida que p aumenta o tamanho da componente maior se aproxime do número total de vértices, visto que existe uma maior probabilidade de todas as arestas se ligarem entre si.

Podemos verificar um crescimento exponencial nos valores menores de p sendo que depois se torna quase constante, estabilizando quando o valor da componente é 2000. Esta "forma"apresentada no gráfico justifica-se pelo facto de que chegando a um ponto em que $(n-1)p$ este será o grau médio de um vértice, seguindo uma distribuição binomial. Trata-se de uma função logarítmica, $O(log(n))$ em que n, neste caso é fixo 2000 nós, sendo por isso a maior componente formada por todos os nós da rede.

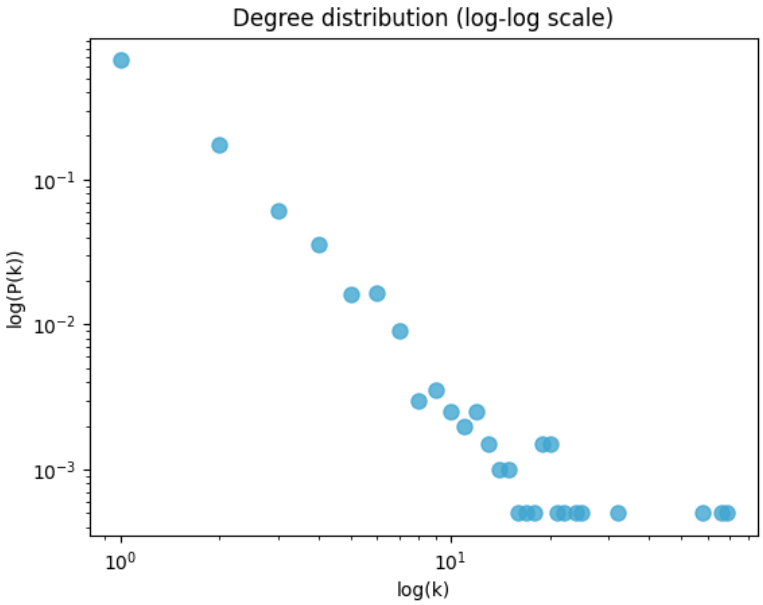
Exercise 7

Para resolução do exercício 7 e 8 foi usada a linguagem python, o código encontra-se no zip com o nome *ba_HW.py*. As seguintes imagens seguem-se anexadas no ficheiro zip.

Nota: O código que gerou esta rede foi inspirado e certas funções usadas pelo o trabalho seguinte: urlhttps://github.com/AlxndrMlk/Barabasi-Albert$_N etwork$

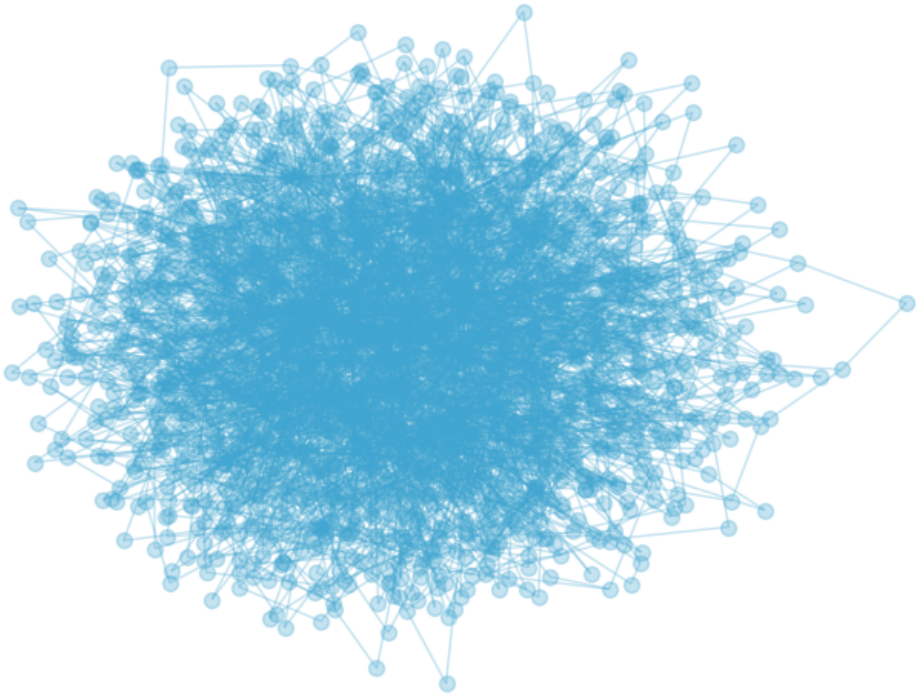


Rede gerada pelo algoritmos Barabási-Albert: ba1.txt

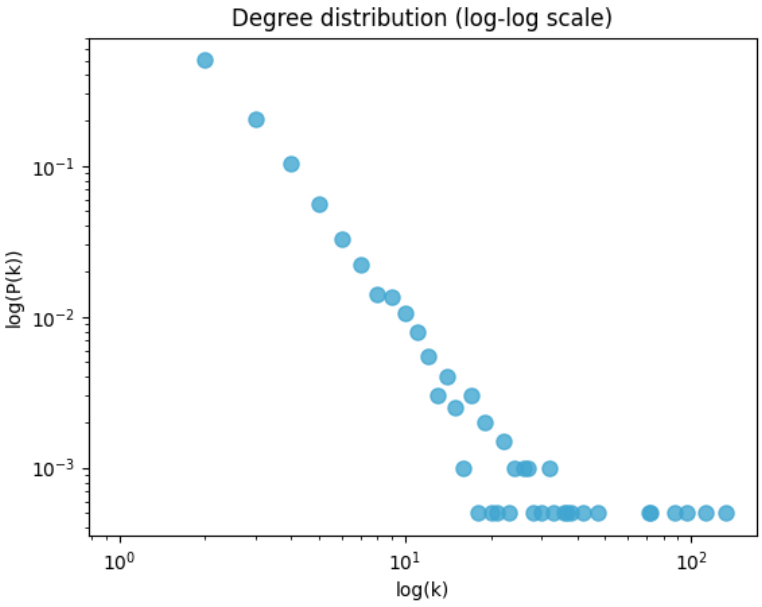


Distribuição de grau de ba1.txt

Exercise 8



Rede gerada pelo algoritmos Barabási-Albert: ba2.txt



Distribuição de grau de ba2.txt

Os ficheiros podem ser encontrados neste link:`https://github.com/JoaoNeto15/NetworkScience.git`