Modelagem Matemática I
Escola de Matemática Aplicada
Fundação Getulio Vargas

Professor: Moacyr Alvim Horta Barbosa da Silva

Alunos: Bruna Fernanda Fistarol e Matheus Henrique Popst de Campos

Junho de 2017

1 Questão 1

Seja A a matriz de adjacência da rede em questão:

$$\begin{matrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{matrix}$$

(a) O número de amigos que cada aluno tem pode ser obtido multiplicando o vetor coluna $u = [1; 1; 1; 1; 1; 1]$ pela matriz de adjacência, obtendo outro vetor coluna $v = [2; 3; 3; 3; 3; 2]$, que indica que o aluno n possui $v(n, 1)$ amigos.

(b) O vetor v do item (a) contém o número de amigos que cada aluno tem. Somando seus elementos (para isso, pode-se usar a função "sum" do MatLab) e dividindo o número obtido por 2, obtemos o número de arestas dessa rede, que é 8.

(c) Proposição: A é a matriz de adjacência $n \times n$ de um grafo e A^k é uma matriz onde a_{ij} informa o número de caminhos que começam no vértice i e terminam no vértice j e possuem comprimento k. Quando $k = 1$, existe um único caminho que liga o vértice i ao vértice j, que é a própria aresta entre ambos. Para $k = 2$, temos que:

$$a_{ij} = a_{i1} \times a_{1j} + a_{i2} \times a_{2j} + \cdots + a_{in} \times a_{nj}$$

Note que se existir a aresta entre i e n e n e j, o produto entre eles será 1 e então somaremos a a_{ij} um possível caminho de comprimento 2 que vai do nó

i ao nó j. Agora, multiplicando a matriz obtida pela matriz de adjacência (ou sejan fazendo A^3), com um raciocínio análogo podemos concluir que a matriz resultante nos dará o número de caminhos de comprimento 3 que vai do nó i ao nó j. Já sabemos o significado de A^2, e novamente, fazendo a multiplicação, temos:

$$a_{ij} = a_{i1} \times a_{1j} + a_{i2} \times a_{2j} + \cdots + a_{in} \times a_{nj}$$

Se existe um certo número de caminhos de comprimento 2 que vai de i a n e existe uma aresta que liga n a j, então soma-se a A_{ij} um certo número de caminhos de comprimento 3 que vai de i a j.

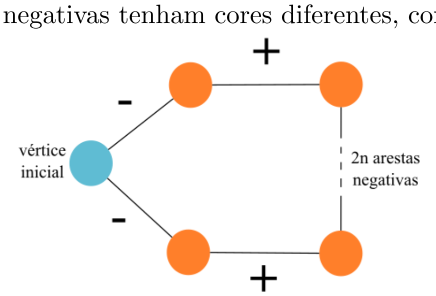
Se elevarmos uma matriz de adjacência M ao cubo, em cada elemento $M(n,n)$, encontraremos o número de caminhos de comprimento 3 que partem de n e terminam em n. Fazendo A^3, obtemos:

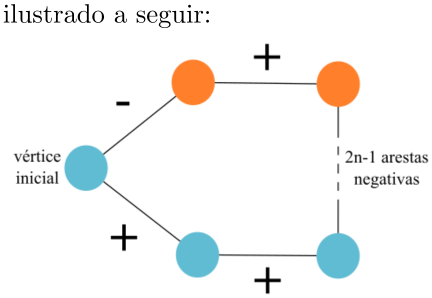
$$A^3 = \begin{matrix} 2 & 4 & 2 & 2 & 4 & 2 \\ 4 & 2 & 6 & 2 & 6 & 2 \\ 2 & 6 & 2 & 6 & 2 & 4 \\ 2 & 2 & 6 & 2 & 6 & 4 \\ 4 & 6 & 2 & 6 & 2 & 2 \\ 2 & 2 & 4 & 4 & 2 & 2 \end{matrix}$$

A diagonal principal dessa matriz nos diz que existem dois caminhos de comprimento 3, partindo e retornando a um mesmo nó. Note que cada um desses dois caminhos se referem ao mesmo triângulo, pois um triângulo pode ser percorrido no sentido horário ou anti-horário. Portanto, cada aluno está incluído em um único triângulo e possui um par de amigos que também são amigos entre si.

2 Questão 2

É sabido que a rede é estruturalmente estável se todo ciclo nessa rede possui um número par de arestas negativas. E tendo um número par de arestas negativas em um ciclo, é possivel o colorir o ciclo de modo que os vértices ligados por arestas de sinal positivo sejam da mesma cor, e vértices ligados por arestas negativas tenham cores diferentes, como ilustrado a seguir:





3 Questão 3

Abaixo o programa em Matlab onde n é o número de nós e A a matriz adjacência. O programa irá sortear um número c que é um nó e procurá-lo na primeira coluna da lista de arestas. Toda vez que encontrar, ele vai fazer o teste de verificação e parar a execução se o teste falhar. Este procedimento acontece até que todos os nós estejam pintados e todas as arestas estejam pintadas.

```
function colorir(n, A)
tic
%n is knot
%A is a matrix n:3
Tamanho=size(A);
if Tamanho(2)~=3
    disp ‘Arrume a matriz!‘
    return
end

B=zeros(1, n);
%matrix with the knots

C=zeros(1, Tamanho(1));

B(1)=1;

while prod(B)==0 || prod(C)==0
    %logical test that ensures that knots knots are painted
    c=randi(n, 1)
    for i=(1:Tamanho(1)) %test a random number
    if A(i, 1)==c %if there is any knot c in the first collum
        if B(c)~=0 %is i painted?
            if B(A(i, 2))==0 %is the other knot not painted?
                B(A(i, 2))=B(c)*A(i, 3);
                if C(i)==0
                C(i)=1;
                end
            else if B(A(i, 2))==B(c)*A(i, 3)
                    if C(i)==0
                        C(i)=1;
                    end
                    %ok
                else disp ‘Nestavel‘
                    return
                end
            end
        end
```

```
        end
    end
end
disp ‘Estavel‘
toc
```

O primeiro grafo é não estável. O segundo, é estável.

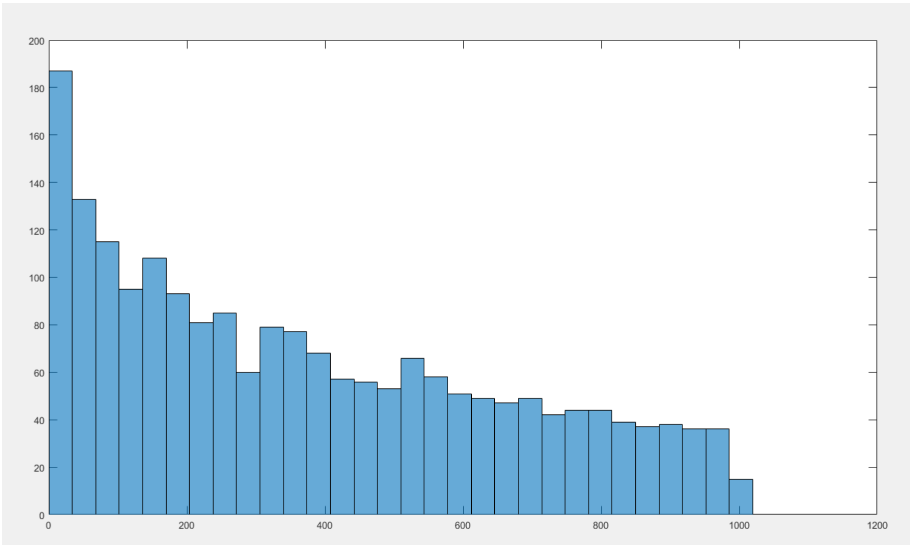
4 Questão 4

(a) O código abaixo cria uma lista de nós e arestas, indicando, na segunda coluna da lista, com qual nó já existente se liga um novo nó adicionado. Com base nessa lista, cria a matriz de adjacência usada para plotar o grafo de anexação uniforme. A base da criação dessa matriz é uma matriz de zeros, e os índices dos elementos da matriz que possuem peso 1 são os próprios valores da lista de nós e arestas.

```
function x=grafouni(n_nos)
x = zeros(n_nos-1, 2);

for t=1:(n_nos-1)
    x(t,1) = t+1;
    x(t,2) = randi([1, t]);
end
adj=zeros(n_nos,n_nos);
for i=1:n_nos-1
adj((x(i,1)),(x(i,2)))=1;
adj((x(i,2)),(x(i,1)))=1;
end
plot(graph(adj))
```

Abaixo, podemos conferir o histograma da distribuição de arestas:



O ajuste do modelo exponencial está na planilha ”Modelo.xlsx”.

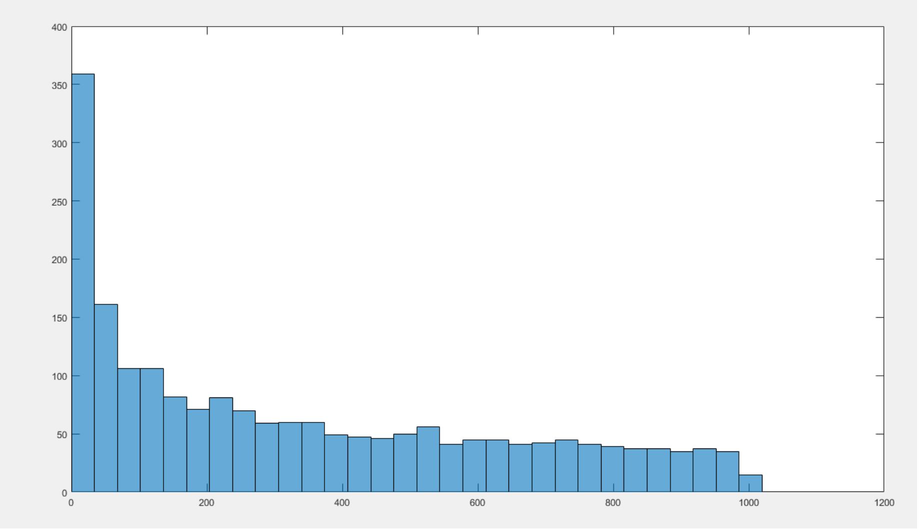
(b) O código abaixo cria uma anexação preferencial de n nós e nos retorna A, uma matriz adjadência e enquanto executa, ele plota o grafo para que vejamos como o grafo evolui.

```
function [A,B]=anexacao(n)
A(1, 2)=1;
A(2, 1)=1;
B=[1 2];
for i=3:n
    c=randi(size(B),1);
    A(i, B(c))=1;
    A(B(c), i)=1;
    B(size(B)+1)=i;
    B(size(B)+1)=B(c);
    plot(graph(A))
    pause(0.05)
end
```

Primeiramente, a opção por não criar a matriz de zeros de A é para deixar mais limpa a visualização da evolução do grafo. O código começa ao criar a aresta na matriz de adjacência $(1, 2)$ e então coloca no vetor B os dois pontos que agora tem grau 1 para que possamos realizar o sorteio. De modo geral, o grau do nó n aparecerá n vezes no vetor B.

Então temos $B(c)$ que foi escolhido de maneira aleatória em B e que será o ponto onde o novo nó i se ligará. Após adicionarmos a matriz adjacência esta nova informação (e consequentemente, uma nova linha e coluna), plota-se, para ver o programa em execução.

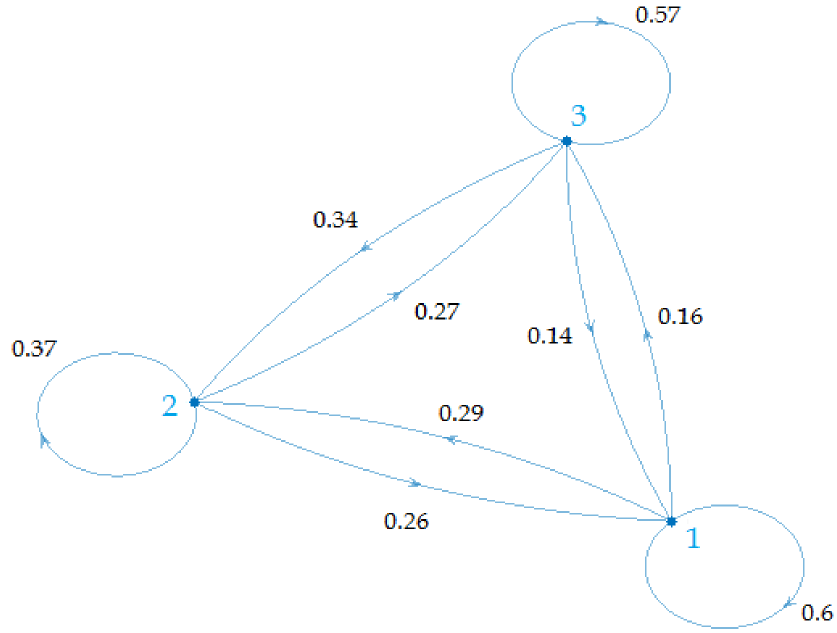
Para termos uma ideia de como é a distribuição, basta que usemos a função histogram(B). Um histograma possível é o abaixo:



O ajuste da lei de potência está na planilha "Modelo.xlsx".

5 Questão 5

(a) Segue abaixo a representação gráfica desse processo de Markov.

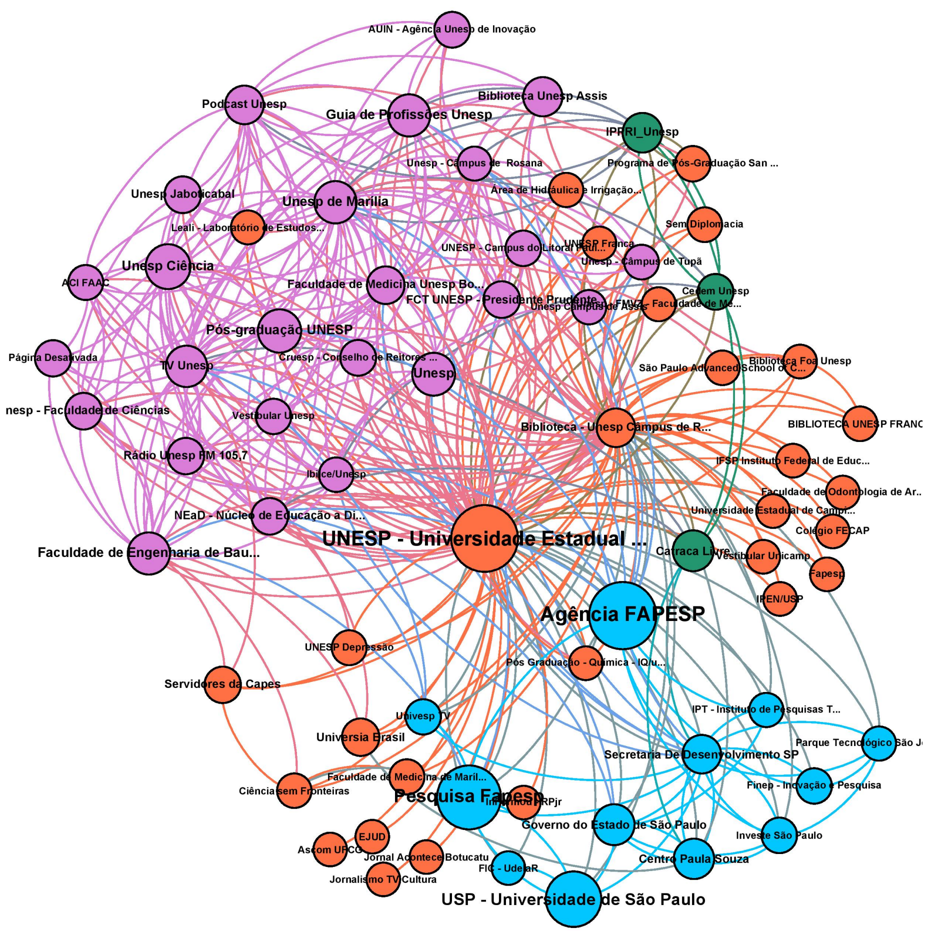


(b) Se o trabalhador é da categoria 2, então vamos multiplicar a matriz de probabilidades M por um vetor coluna $v = [0; 1; 0]$ (já que o trabalhador pertence à segunda categoria), obtendo como resultado o vetor coluna $v_1 = [0.2900; 0.3700; 0.3400]$, que diz que o filho do trabalhador tem 29% de chances de pertencer à categoria 1, 37% de chances de pertencer à categoria 2 e 34% de chances de pertencer à categoria 3. Multiplicando o vetor obtido por M, obtém-se o vetor $v_2 = [0.3357; 0.3041; 0.3602]$, que são as probabilidades de seu neto pertencer às categorias 1, 2 e 3, respectivamente, e fazendo $v_2 * M$, obtemos $v_3 = [0.3472; 0.2971; 0.3557]$, que são as probabilidades do bisneto.

(c) Para encontrar o estado estacionário, basta multiplicar o vetor v pela matriz M^k com um k suficientemente grande para observar que os valores do vetor obtido não se alteram mais. Dessa forma, obtemos como resultado o vetor $[0.3544; 0.2961; 0.3495]$, que nos diz que, em longo prazo, devemos observar cerca de $35,44\%$ dos trabalhadores na categoria 1, $29,61\%$ dos trabalhadores na categoria 2 e $34,95\%$ dos trabalhadores na categoria 3.

6 Questão 6

Foi escolhido no Facebook a página da UNESP. Optou-se por apenas páginas de aresta 1, pois de outra maneira, haveria uma quantidade exagerada de nós que impossibilitaria que qualquer informação fosse obtida visualmente.



O tamanho dos nós foi decidido por Page Rank e a cor dos nós e arestas, por modularidade. Mesmo com pouca informação, a modularidade funcionou relativamente bem, por exemplo, a Revista Pesquisa FAPESP e a própria FAPESP ficaram com a mesma modularidade. Da mesma forma, praticamente todas as páginas das faculdades e institutos da UNESP ficaram com a mesma modularidade.

Quando o ranqueamento por Page Rank foi feito, verificou-se que mesmo sendo a rede dos likes da UNESP, nesta rede a página da Agência FAPESP tem maior Page Rank, deixando à UNESP apenas o segundo lugar. Comparando com a realidade, talvez isso seja revelador pois mesmo dentro da rede da UNESP, as organizações "valorizem" mais a FAPESP, pois é quem oferece dinheiro para a pesquisa.

Ranking por Page Rank
Agência FAPESP
UNESP - Universidade Estadual Paulista
Pesquisa FAPESP
USP - Universidade de São Paulo
Unesp Ciência
Pós-graduação UNESP

Ranking por Grau
UNESP - Universidade Estadual Paulista
Biblioteca - Unesp Câmpus de Rosana
Unesp de Marília
TV Unesp
Podcast Unesp

Enquanto o ranqueamento por grau coloca algumas das principais instituições da Universidade no começo do ranking, não são as mais valorizadas. Isso pode ser um indício de que as pessoas curtem páginas como da TV e Podcast, mas não ligam muito para elas, sendo que elas valorizam mais coisas como a FAPESP, a USP e a pós-graduação.