

MPEI 2018-2019

# # 27

## Resolução de Alguns Exercícios

# Simulação de cadeias de Markov

Demonstração em Matlab

Com base no código anexo ao guião PL08

# Crawl.m

- `function state = crawl(H, first, last)`
- % random walk on the graph according to state transition matrix H
- % first = initial state, last = terminal or absorbing state
- % the sequence of states will be saved in the vector "state"
- % initially, the vector contains only the initial state
- 
- 
- `state = [first];`
- 
- %% keep moving from page to page
- % until page "last" is reached
- 
- `while (1)`
- `state(end+1) = nextState(H, state(end));`
- `if (state(end) == last) break % last reached`
- `end`
- `end`

# nextState

- `function state = nextState(H, currentState)`
- % given a transition matrix and the current state,
- % this function returns the next state
- % find the probabilities of reaching all pages starting at the current one
- % get the column corresponding to currentState
- `probVector = H(:,currentState);`
- 
- % N is the number of pages, that is, H is N x N
- `N = length(probVector);`
- % pick the next page randomly according to those probabilities
- `state = discrete_rnd(N, probVector);`
- `state = state(1);`

# Exemplo simples

```
T=zeros(4)
```

```
T(2,1)=1;
```

```
T(1,2)=0.5; T(4,2)=0.5;
```

```
T(2,3)=0.6; T(4,3)=0.4;
```

```
T(2,4)=0.6; T(3,4)=0.4;
```

```
T
```

```
from=1
```

```
to=3
```

```
state=crawl(T,from,to)
```

# PL08 exercício 6

```
% Confirmar resultados de tempos de absorção por simulação
```

```
t=sum(F) ; % para comparação
```

```
Ntry= 1000; to=4;
```

```
for from=1:3
```

```
    R= zeros(1,Ntry);
```

```
    for n= 1:Ntry,
```

```
        state = crawl(H, from, to);
```

```
        R(n)= length(state);
```

```
    end
```

```
m=mean(R);
```

```
fprintf(1, 'De %d para %d (absorvente):\n\tSIMUL (%d iters) = %.2f Teórico (usando F) = %.2f\n',from, to,  
Ntry,m, t(from));
```

```
end
```

# Resultado (exemplo)

$t =$

90.0000 85.0000 75.0000

De 1 para 4 (absorvente):

SIMUL (1000 iters) = 89.37 Teórico (usando F) = 90.00

De 2 para 4 (absorvente):

SIMUL (1000 iters) = 83.10 Teórico (usando F) = 85.00

De 3 para 4 (absorvente):

SIMUL (1000 iters) = 75.19 Teórico (usando F) = 75.00

# Exercícios de revisão

Cadeias de Markov

# Problema 1

- Implemente uma função Matlab chamada *markov\_estadoestacionario.m* que utilize o método das potências para calcular as probabilidades em estado estacionário para uma cadeia de Markov com N estados.
- Um dos parâmetros de entrada será a matriz de transição
  - Assumida como irredutível e aperiódica
- Outros parâmetros:
  - Vector estado inicial
  - Limiar para terminar o processo (máximo da diferença entre os vectores em duas iterações deverá ser inferior a esse valor)
- Inicialize o processo com um vector uniforme, isto é  $x = \frac{[1,1,1,\dots,1]}{N}$ , e limiar=1e-5 caso apenas seja fornecida a matriz de transição

# Problema 2

Considere a matriz T seguinte:

$T =$

$$\begin{matrix} 0 & 0.5000 & 0 & 0 \\ 1.0000 & 0 & 0.6000 & 0.6000 \\ 0 & 0 & 0 & 0.4000 \\ 0 & 0.5000 & 0.4000 & 0 \end{matrix}$$

- Use a função que criou no problema 1 para calcular o vector estado estacionário usando o vector inicial  $x = [0.25, 0.25, 0.25, 0.25]$
- Confirme o resultado calculando o vector estacionário por outro método
- Adicione à sua função a capacidade de mostrar num gráfico o valor do segundo elemento de  $x$  em função da iteração (que deverá ser o eixo do  $xx$ ) e repita o ponto anterior

# Problema 3

- Repita o problema anterior para outros vectores iniciais
  - Exemplos:  $[1,0,0,0]$  e  $[0,1,0,0]$  etc
- Continua a convergir para o mesmo vector ?

# Exercícios de mini-testes

# Exercício 1 (2015-2016)

- 8.0 **2)** Considere que tem um pequeno conjunto de páginas web identificadas pelas letras A a F com as seguintes ligações entre si no dia 1 de janeiro de 2016: a página A tem links para as páginas B a E; a página B tem links para as páginas B, D e F; pode chegar-se às páginas A e B através da página C; D apenas tem links para E e B; E tem links para A e B; F possui links para todas as outras páginas, excepto para ela própria.

Considerando que se pretende obter o pagerank das páginas e que se inicializa esse valor com um valor igual para todas as páginas e igual a  $1/6$ :

- 3.0 **2.a)** Qual o valor da estimativa do pagerank de cada página ao fim de três iterações do processo de cálculo?

Resposta: A \_\_\_\_\_ B \_\_\_\_\_

B \_\_\_\_\_ D \_\_\_\_\_

E \_\_\_\_\_ F \_\_\_\_\_

- 2.0** **2.b)** Represente num gráfico a evolução do valor do pagerank de cada uma das páginas em função da iteração? Deve utilizar um número de iterações suficiente para que os valores estabilizem.
- Código matlab/octave:

- 3.0** **2.c)** Qual a página com o maior valor do pagerank e qual o seu valor?

Resposta: Página? \_\_\_\_\_  
pagerank? \_\_\_\_\_

Código matlab/octave: