PL 3

Cadeias de Markov

Nota: Adopte a a definição da matriz de transição (de estados) em que o elemento t_{ij} da matriz corresponde à probabilidade de transição do estado j para o estado i.

1. Considere a seguinte situação e responda às alíneas abaixo:

   Um aluno do primeiro ano de um curso de Engenharia tem todas as semanas 2 aulas Teórico-Práticas de uma Unidade Curricular X às 9:00, às quartas e sextas.

   Todos os dias que tem aulas desta UC, o aluno decide se vai à aula ou não da seguinte forma: Se tiver estado presente na aula anterior a probabilidade de ir à aula é 70%; se faltou à anterior, a probabilidade de ir é 80%.

   (a) Se estiver presente na aula de quarta numa determinada semana, qual a probabilidade de estar presente na aula de quarta da semana seguinte?

   Sugestão: Comece por definir a matriz de transição de estados e o vetor estado correspondentes.

   (b) Se não estiver presente na aula de quarta numa determinada semana, qual a probabilidade de estar presente na aula de quarta da semana seguinte?

   (c) Sabendo que esteve presente na primeira aula, qual a probabilidade de estar na última aula, assumindo que o semestre tem exactamente 15 semanas de aulas e não existem feriados?

   (d) Represente num gráfico a probabilidade de faltar a cada uma das 30 aulas, assumindo que a probabilidade de estar presente na primeira aula é de 85%.

2. Considere a seguinte "dança" de grupos: Divide-se uma turma em 3 grupos (A, B e C) no início do semestre e no final de cada aula efectuam-se os seguintes movimentos:

   - $1/3$ do grupo A vai para o grupo B e outro $1/3$ do grupo A vai para o grupo C;
   - $1/4$ do grupo B vai para A e $1/5$ de B vai para C
   - Metade do grupo C vai para o grupo B; a outra mantém-se no grupo C.

   (a) Crie em Matlab a matriz de transição de estados que representa as trocas entre grupos.

   Confirme que se trata de uma matriz estocástica.

   (b) Crie o vector relativo ao estado inicial considerando que no total temos 90 alunos, o grupo A tem o dobro da soma dos outros dois e os grupos B e C têm o mesmo número de alunos.

   (c) Quantos elementos integrarão cada grupo no fim da aula 30 considerando como estado inicial o definido na alínea anterior?

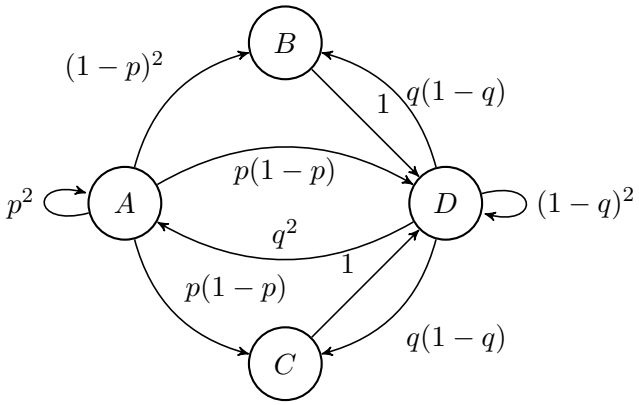
   (d) Quantos elementos integrarão cada grupo no fim da aula 30 considerando que inicialmente se distribuiram os 90 alunos equitativamente pelos 3 grupos?

3. Gere aleatoriamente uma matriz de transição de estados para uma cadeia de 20 estados (identificados de 1 a 20) recorrendo à função do Matlab *rand*. Com base nessa matriz:

(a) Confirme que a matriz de transição de estados é estocástica.

(b) Qual a probabilidade de o sistema, começando no estado 1, estar no estado 20 após 2 transições? E após 5? E após 10? E após 100? Apresente os resultados em percentagem e com 5 casas decimais. O que conclui?

4. Considere o seguinte diagrama representativo de uma Cadeia de Markov:



(a) Defina, em Matlab, a matriz de transição de estados T assumindo $p = 0,4$ e $q = 0,6$.

(b) Assuma que o sistema se encontra inicialmente no estado A. Qual a probabilidade de estar em cada estado ao fim de 5 transições? E de 10 transições? E de 100 transições? E de 200 transições?

(c) Determine as probabilidades limite de cada estado. Compare estes valores com os obtidos na alínea anterior. O que conclui?

5. Considere que o tempo em cada dia é genericamente classificado num de 3 estados – sol, nuvens e chuva – e que o tempo num determinado dia apenas depende do tempo no dia anterior. Assuma que estamos no primeiro dia de janeiro e que as probabilidades de transição de estados são as da tabela seguinte.

dia n \ dia $n+1$ $\rightarrow$	sol	nuvens	chuva
sol	0,7	0,2	0,1
nuvens	0,2	0,3	0,5
chuva	0,3	0,3	0,4

(a) Defina, em Matlab, a correspondente matriz de transição.

(b) Qual a probabilidade de estar sol no segundo dia e no terceiro dia de janeiro quando o primeiro dia é de sol?

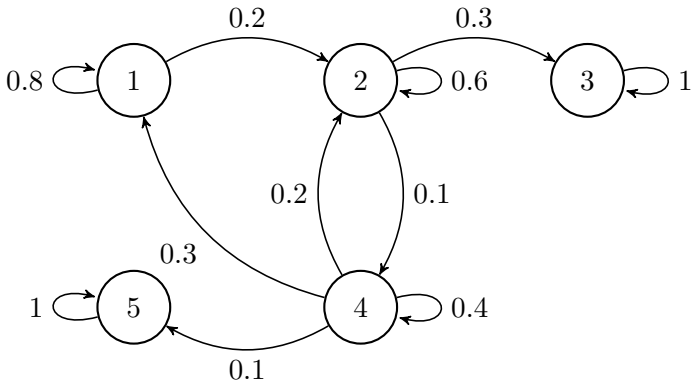
(c) Qual a probabilidade de não chover nem no segundo dia nem no terceiro dia de janeiro quando o primeiro dia é de sol?

(d) Assumindo que o primeiro dia é de sol, determine o número médio de dias de sol, de nuvens e de chuva que se espera ter em todo o mês de janeiro.

(e) Assumindo que o primeiro dia é de chuva, determine o número médio de dias de sol, de nuvens e de chuva que se espera ter em todo o mês de janeiro. Compare estes resultados com os da alínea anterior. O que conclui?

(f) Considere uma pessoa com reumatismo crónico que tem dores reumáticas com probabilidades de 10%, 30% e 50% quando os dias são de sol, de nuvens ou de chuva, respetivamente. Qual o número esperado de dias que a pessoa vai sofrer de dores reumáticas em janeiro quando o primeiro dia é de sol? E quando o primeiro dia é de chuva?

6. Considere a cadeia de Markov com o diagrama de transição de estados seguinte:



   (a) Defina em Matlab a matriz de transição de estados T, com T_{ij} sendo a probabilidade de ir do estado j para o estado i num único passo.

   (b) Faça um gráfico com a probabilidade de, começando no estado 1, estar no estado 2 ao fim de n passos, com n a variar de 1 até 100. Justifique o que observa.

   (c) Faça um gráfico com a probabilidade de, começando no estado 1, estar no estado 3 ao fim de n passos. Na mesma figura, faça um segundo gráfico com a probabilidade de, começando no estado 1, estar no estado 5 ao fim de n passos. Em ambos os casos, considere n a variar de 1 até 100. Justifique o que observa.

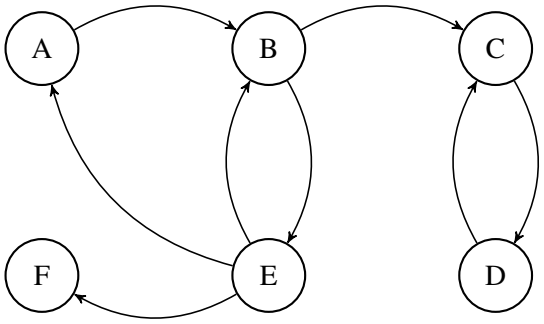
   (d) Determine a matriz Q.

   (e) Determine a matriz fundamental F.

   (f) Qual a média (valor esperado) do número de passos até à absorção começando no estado 1? E começando no estado 2? E se começando no estado 4?

   (g) Começando no estado 1, qual é a probabilidade de absorção do estado 3? E do estado 5? Verifique a coerência destes valores com o que observou na alínea 6c).

7. Considere o conjunto de páginas Web e respetivas hyperligações entre si dado pelo diagrama seguinte:



   (a) Usando a matriz H das hyperligações, obtenha a estimativa do pagerank de cada página ao fim de 10 iterações. Relembre que deve considerar (i) a mesma probabilidade de transição de cada página para todas as páginas seguintes possíveis e (ii) a probabilidade da página inicial deve ser igual para todas as páginas. Qual/quais a(s) página(s) com maior pagerank e qual o seu valor?

   (b) Identifique a "spider trap"e o "dead-end"contidos neste conjunto de páginas.

   (c) Altere a matriz H para resolver apenas o problema do "dead-end"e recalcule o pagerank de cada página novamente em 10 iterações.

   (d) Resolva agora ambos os problemas e recalcule o pagerank de cada página novamente em 10 iterações (assuma $\beta = 0,8$).

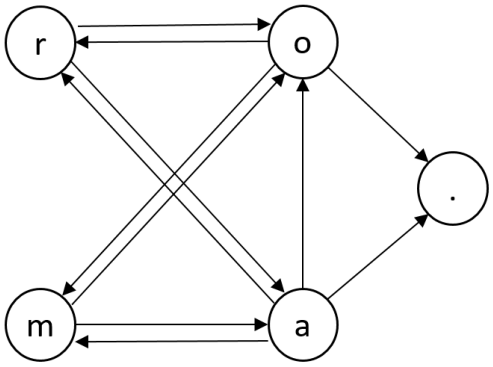
   (e) Calcule agora o pagerank de cada página considerando um número mínimo de iterações que garanta que nenhum valor muda mais do que 10^{-4} em 2 iterações consecutivas. Quantos iterações são necessárias? Compare os valores de pagerank obtidos com os da alínea anterior. O que conclui?

3.1 Secção para avaliação [1]

Considere a geração aleatória de palavras em português com base em cadeias de Markov. Para avaliar a eficiência de um gerador de palavras aleatórias, considere 2 parâmetros: (1) o número de palavras diferentes geradas e (2) a probabilidade de uma palavra gerada ser uma palavra portuguesa válida. Quanto mais altos esses dois parâmetros, mais eficiente é o gerador de palavras aleatórias.

Nas experiências seguintes, considere a geração aleatória de palavras portuguesas compostas apenas pelas letras 'a', 'm', 'o' e 'r'. Considere também o arquivo `wordlist-preao-20201103.txt` (disponível em `https://natura.di.uminho.pt/download/sources/Dictionaries/wordlists/`) com uma lista de quase 1 milhão de palavras portuguesas válidas, organizado em 1 palavra por linha.

1. **(Peso de avaliação = 60%)** Suponha que o gerador de palavras aleatórias é baseado na cadeia de Markov representada pelo seguinte diagrama de transição de estados, onde a probabilidade de transição de cada estado é a mesma para todos os estados possíveis seguintes (o estado '.' indica que a letra anterior é a última letra da palavra):



e a probabilidade da primeira letra é a mesma para todas as letras.

   (a) Defina, no Matlab, a matriz de transição de estados T e simule a geração de uma palavra aleatória (em Anexo, tem uma proposta de uma função `crawl` que pode usar para implementar a simulação).

   (b) Simule a geração de 10^5 palavras aleatórias para estimar a lista de palavras geradas e a probabilidade de cada palavra. Quantas palavras diferentes foram geradas? Apresente as 5 palavras com as probabilidades estimadas mais altas e respectivos valores de probabilidade.

   (c) Determine as probabilidades teóricas das 5 palavras apresentadas na questão anterior. Compare os valores teóricos com os valores estimados anteriores. O que conclui?

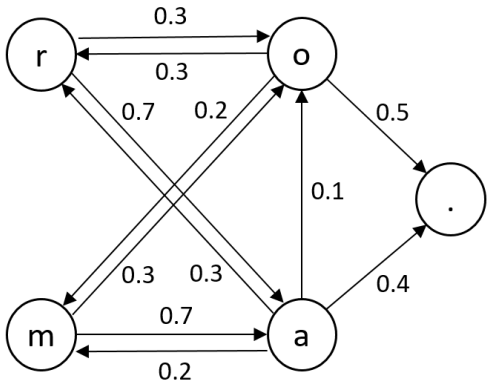
   (d) Importe (do arquivo `wordlist-preao-20201103.txt`) a lista de palavras em português para um "cell array". Com essa lista e os resultados da questão b), estime a probabilidade do gerador de palavras aleatórias gerar uma palavra válida em português.

   (e) Altere o gerador de palavras aleatórias para considerar um novo parâmetro de entrada n que representa o tamanho máximo da palavra (em número de letras) das palavras geradas (ou seja, o gerador de palavras pára se atingir o estado '.' ou se já tiver gerado n letras).

   (f) Para $n = 8, 6$ e 4, simule a geração de 10^5 palavras aleatórias para estimar o número de palavras diferentes geradas e a probabilidade de uma palavra gerada ser uma palavra válida em português. Compare esses resultados entre eles e com os resultados de 1b) e 1d). O que conclui? Explique as suas conclusões!

[1] A execução desta secção será objeto de avaliação. Assim, deverá fazer um relatório (a submeter em PDF) o mais completo possível com as respostas às perguntas desta secção. O relatório deverá começar por identificar o ano letivo, a disciplina, a turma prática e os elementos do grupo (nome e No. Mec.) que realizou o trabalho. Sempre que precisar de desolvolver um código Matlab, deverá incluir no relatório o código devidamente comentado.

2. **(Peso de avaliação = 10%)** Altere a matriz de transição de estados T assumindo agora as probabilidades de transição definidas no diagrama seguinte:



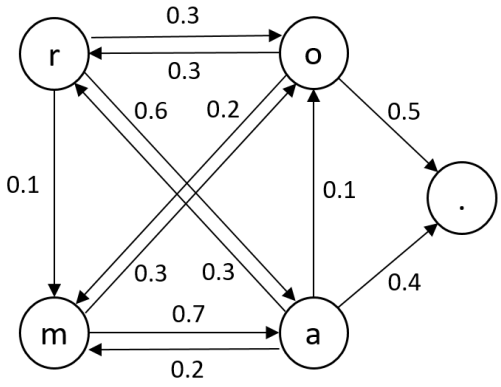
Assuma novamente que a probabilidade da primeira letra é a mesma para todas as letras. Para $n = \infty, 8, 6$ e 4, simule a geração de 10^5 palavras aleatórias para estimar o número de palavras diferentes geradas e a probabilidade de uma palavra gerada ser uma palavra válida em português.

Analise of resultados obtidos. Compare-os entre si e com os reaultados anteriores. O que conclui sobre a eficiência destes geradores de palavras aleatórias quando comparados com os geradores anteriores? Explique as suas conclusões!

3. **(Peso de avaliação = 10%)** Considere a matriz de transição de estados T da questão 2. Usando as palavras do ficheiro `wordlist-preao-20201103.txt` que apenas contenham as letras 'a', 'm', 'o' e 'r', estime a probabilidade de cada letra ser a primeira letra da palavra. Repita a questão 2 assumindo agora estas novas probabilidades para a primeira letra.

Analise of resultados obtidos. Compare-os entre si e com os resultados anteriores. O que conclui sobre a eficiência destes geradores de palavras aleatórias quando comparados com os geradores anteriores? Explique as suas conclusões!

4. **(Peso de avaliação = 10%)** Uma das limitações dos geradores anteriores é não conseguirem gerar palavras com a sequência 'rm' que existe em algumas palavras portuguesas (por exemplo, 'arma'). Considere agora um gerador de palavras aleatórias baseado na cadeia de Markov representada pelo diagrama de transição de estados seguinte:



Repita a questão 3 para este caso e analise of resultados obtidos. Compare-os entre si e com os resultados anteriores. O que conclui sobre a eficiência destes geradores de palavras aleatórias quando comparados com os geradores anteriores? Explique as suas conclusões!

5. **(Peso de avaliação = 10%)** Usando as palavras do ficheiro `wordlist-preao-20201103.txt` que apenas contenham as letras 'a', 'm', 'o' e 'r', estime as probabilidades de transição de estados da matriz T baseado nas sequências de 2 letras que aparecem nessas palavras (e nas últimas letras para estimar a transição para o estado '.'). Com esta matriz T estimada, repita questão 3.

Analise of resultados obtidos. Compare-os entre si e com os resultados anteriores. O que conclui sobre a eficiência destes geradores de palavras aleatórias quando comparados com os geradores anteriores? Explique as suas conclusões!

Anexo

```
% a state transition matrix example
H = [0.5 0.5 0 ;
     0.5 0.4 0 ;
     0   0.1 1 ];

 % how to use crawl()
state = crawl(H, 1, 3)

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

% Returning the next state
% Inputs:
% H - state transition matrix
% currentState - current state
function state = nextState(H, currentState)
     % find the probabilities of reaching all states starting at the current one:
     probVector = H(:,currentState)';  % probVector is a row vector
     n = length(probVector);  %n is the number of states
     % generate the next state randomly according to probabilities probVector:
     state = discrete_rnd(1:n, probVector);
end

% Generate randomly the next state.
% Inputs:
% states = vector with state values
% probVector = probability vector
function state = discrete_rnd(states, probVector)
     U=rand();
     i = 1 + sum(U > cumsum(probVector));
     state= states(i);
end
```