RELATÓRIO TÉCNICO A3

ESTRUTURAS MATEMÁTICAS

ALUNOS:

Alisson Rayan | **RA:** 1272314418

Júlio César Souza | **RA:** 12723120855

Wesley Dantas | **RA:** 1272311443

DESCRIÇÃO

A **cadeia de Markov** é um modelo probabilístico utilizado para descrever sistemas que evoluem de um estado para outro ao longo do tempo, com a característica de que a probabilidade de transição depende apenas do estado atual. Neste projeto, implementamos uma simulação interativa que permite ao usuário definir:

- A quantidade de estados;
- A matriz de transição probabilística;
- O vetor de estado inicial;
- A quantidade de passos da simulação.

A aplicação foi projetada para rodar no terminal, preferencialmente no **cmd**, de forma clara e didática, permitindo análises visuais por meio de **gráficos** gerados ao final da simulação.

→ CÓDIGO FONTE

O código fonte do projeto, em Julia, pode ser encontrado pelo GitHub em: https://github.com/zhucssz/julia_base-a3-em, pois ocuparia muito espaço nesta documentação.

INSTALAÇÃO

Primeiramente, é necessário instalar a linguagem Julia, podendo ser encontrada no site oficial:

- https://julialang.org/downloads/

Posteriormente, é necessário instalar o pacote **Plots**, que foi usado para a criação dos gráficos, para demonstrar melhor o funcionamento do código. Para instalar o pacote **Plots**, basta abrir o REPL do Julia (aplicativo que aparece ao instalar a linguagem) e digitar as linhas:

```
using Pkg

Pkg.add("Plots")
```

→ EXECUÇÃO

Após a instalação, para executar o código basta seguir os passos:

- Salve o código em um arquivo chamado cadeia_markov.jl;
- No cmd, navegue até o diretório onde está o arquivo;
- Execute digitando “julia cadeia_markov.jl”, sem aspas.

Ao executar o programa, ele irá perguntar o número dos estados, o vetor inicial, a matriz de transição e o número de passos, que devem ser digitados no terminal como requisitado.

Após concluído, o programa irá gerar os resultados, assim como um gráfico, em png, na mesma pasta em que o código está sendo executado.

CASO ESTUDADO:

No presente estudo, utiliza-se uma cadeia de Markov para modelar a escolha sucessiva de pacotes de seguro por parte de clientes ao longo do tempo. A matriz de transição define as probabilidades de migração entre os estados A, B e C, representando diferentes pacotes, em função apenas da escolha anterior, o que caracteriza a propriedade de memória curta do processo de Markov. A evolução das probabilidades é obtida pela multiplicação do vetor de estado inicial pela matriz de transição elevada à potência correspondente ao número de períodos analisados. No exemplo, após quatro passos, observa-se uma distribuição de probabilidade estabilizada em torno de 41,2% para o pacote A, 17,0% para o B e 41,8% para o C. Este comportamento ilustra como cadeias de Markov podem ser aplicadas de forma eficaz na previsão do comportamento de consumidores em processos de decisão sequencial, oferecendo uma ferramenta poderosa para análises probabilísticas em ambientes dinâmicos.

https://github.com/zhucssz/julia_base-a3-em/blob/main/caso_estudado.pdf

APLICAÇÃO DO CASO

$\rightarrow$ Cálculo manual, passos 1 a 4 (Feito usando o aplicativo Mathcha.io):

$$P = \begin{bmatrix} 0,7 & 0,1 & 0,2 \\ 0,4 & 0,5 & 0,1 \\ 0,1 & 0,1 & 0,8 \end{bmatrix}$$

$$p\,(0) = [0,5 \;\; 0,3 \;\; 0,2]$$

$$p\,(1) = [0,5 \;\; 0,3 \;\; 0,2] \cdot \begin{bmatrix} 0,7 & 0,1 & 0,2 \\ 0,4 & 0,5 & 0,1 \\ 0,1 & 0,1 & 0,8 \end{bmatrix}$$

$$\begin{aligned} 1^\circ\ posição &= 0,5 \cdot 0,7 + 0,3 \cdot 0,4 + 0,2 \cdot 0,1 = 0,35 + 0,12 + 0,02 = 0,49 \\ 2^\circ\ posição &= 0,5 \cdot 0,1 + 0,3 \cdot 0,5 + 0,2 \cdot 0,1 = 0,05 + 0,15 + 0,02 = 0,22 \\ 3^\circ\ posição &= 0,5 \cdot 0,2 + 0,3 \cdot 0,1 + 0,2 \cdot 0,8 = 0,10 + 0,03 + 0,16 = 0,29 \end{aligned}$$

$$p\,(1) = [0,49 \;\; 0,22 \;\; 0,29]$$

$$p\,(2) = [0,49 \;\; 0,22 \;\; 0,29] \cdot \begin{bmatrix} 0,7 & 0,1 & 0,2 \\ 0,4 & 0,5 & 0,1 \\ 0,1 & 0,1 & 0,8 \end{bmatrix}$$

$$\begin{aligned} 1^\circ\ posição &= 0,49 \cdot 0,7 + 0,22 \cdot 0,4 + 0,29 \cdot 0,1 = 0,343 + 0,088 + 0,029 = 0,46 \\ 2^\circ\ posição &= 0,49 \cdot 0,1 + 0,22 \cdot 0,5 + 0,29 \cdot 0,1 = 0,049 + 0,11 + 0,029 = 0,188 \\ 3^\circ\ posição &= 0,49 \cdot 0,2 + 0,22 \cdot 0,1 + 0,29 \cdot 0,8 = 0,098 + 0,022 + 0,232 = 0,352 \end{aligned}$$

$$p\,(2) = [0,46 \;\; 0,188 \;\; 0,352]$$

$$p\,(3) = [0,46 \;\; 0,188 \;\; 0,352] \cdot \begin{bmatrix} 0,7 & 0,1 & 0,2 \\ 0,4 & 0,5 & 0,1 \\ 0,1 & 0,1 & 0,8 \end{bmatrix}$$

$$\begin{aligned} 1^\circ\ posição &= 0,46 \cdot 0,7 + 0,188 \cdot 0,4 + 0,352 \cdot 0,1 = 0,322 + 0,0752 + 0,0352 = 0,4324 \\ 2^\circ\ posição &= 0,46 \cdot 0,1 + 0,188 \cdot 0,5 + 0,352 \cdot 0,1 = 0,046 + 0,094 + 0,0352 = 0,1752 \\ 3^\circ\ posição &= 0,46 \cdot 0,2 + 0,188 \cdot 0,1 + 0,352 \cdot 0,8 = 0,092 + 0,0188 + 0,2816 = 0,3924 \end{aligned}$$

$$p\,(3) = [0,4324 \;\; 0,1752 \;\; 0,3924]$$

$$p\,(4) = [0,4324 \;\; 0,1752 \;\; 0,3924] \cdot \begin{bmatrix} 0,7 & 0,1 & 0,2 \\ 0,4 & 0,5 & 0,1 \\ 0,1 & 0,1 & 0,8 \end{bmatrix}$$

$$\begin{aligned} 1^\circ\ pos &= 0,4324 \cdot 0,7 + 0,1752 \cdot 0,4 + 0,3924 \cdot 0,1 = 0,30268 + 0,07008 + 0,03924 = 0,412 \\ 2^\circ\ pos &= 0,4324 \cdot 0,1 + 0,1752 \cdot 0,5 + 0,3924 \cdot 0,1 = 0,04324 + 0,0876 + 0,03924 = 0,17008 \\ 3^\circ\ pos &= 0,4324 \cdot 0,2 + 0,1752 \cdot 0,1 + 0,3924 \cdot 0,8 = 0,08648 + 0,01752 + 0,31392 = 0,41792 \end{aligned}$$

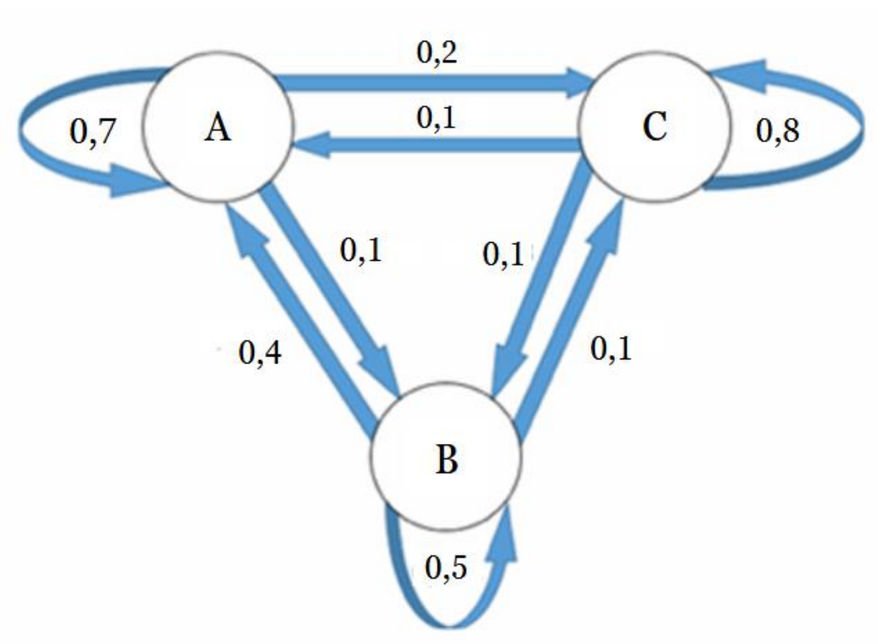
$$p\,(4) = [0,412 \;\; 0,17008 \;\; 0,41792]$$

→ Programa no terminal(cmd):

```
==[ Simulador cadeias de Markov ]==

[i] Quantidade de estados da cadeia: 3

[i] Digite a matriz de transição (linha por linha, separados por espaço):
Linha 1: 0.7 0.1 0.2
Linha 2: 0.4 0.5 0.1
Linha 3: 0.1 0.1 0.8

[i] Digite o vetor de estado inicial (separado por espaço): 0.5 0.3 0.2

[i] Quantos passos deseja simular? 4

[#] Estado inicial: [0.5, 0.3, 0.2]
[-] Após passo (1): [0.49, 0.22, 0.29]
[-] Após passo (2): [0.46, 0.188, 0.352]
[-] Após passo (3): [0.4324, 0.1752, 0.3924]
[-] Após passo (4): [0.412, 0.1701, 0.4179]
```

Como é possível observar, os resultados apresentados batem com os 4 passos do programa, comprovando que o cálculo foi realizado corretamente.

→ Diagrama de transição:



SOLUÇÃO APLICADA

A solução foi construída utilizando conceitos fundamentais de cadeias de Markov com foco didático e interativo. O sistema simula a evolução dos estados ao longo do tempo multiplicando iterativamente o vetor de estados pelo transposto da matriz de transição.

Foi implementado também um controle de entrada de dados que garante que a matriz de transição e o vetor inicial sejam válidos (com soma igual a 1 e consistência numérica).

A visualização dos resultados é feita com o pacote **Plots.jl**, gerando um gráfico com a evolução da distribuição de probabilidade de cada estado ao longo dos passos simulados.