UEPB
UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS I
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE ESTATÍSTICA
CURSO DE BACHARELADO EM ESTATÍSTICA

MAYARA CRISTINA DE ALMEIDA SILVA

QUEUEPB: UMA PROPOSTA DE PACOTE PARA MODELAGEM DE FILAS MARKOVIANAS

CAMPINA GRANDE - PB
2021

MAYARA CRISTINA DE ALMEIDA SILVA

QUEUEPB: UMA PROPOSTA DE PACOTE PARA MODELAGEM DE FILAS MARKOVIANAS

Trabalho de Conclusão de Curso (Artigo) apresentado ao curso de Bacharelado em Estatística do Departamento de Estatística do Centro de Ciências e Tecnologia da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de bacharel em Estatística.

Orientador: Prof. Divanilda Maia Esteves
Coorientador: Prof. Gustavo Henrique Esteves

CAMPINA GRANDE - PB
2021

M929q Silva, Mayara Cristina de Almeida.
QUEUEPB [manuscrito] : uma proposta de pacote para modelagem de filas markovianas / Mayara Cristina de Almeida Silva. - 2021.
18 p.

Digitado.
Trabalho de Conclusão de Curso (Graduação em Estatística) - Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia , 2021.
"Orientação : Profa. Dra. Divanilda Maia Esteves , Departamento de Estatística - CCT."

1. Processos estocásticos. 2. Medidas de desempenho. 3. Variáveis aleatórias. 4. Filas markovianas. I. Título

21. ed. CDD 519.23

Elaborada por Lucas H. A. da Silva - CRB - 15/898 BC/UEPB

MAYARA CRISTINA DE ALMEIDA SILVA

QUEUEPB: UMA PROPOSTA DE PACOTE PARA MODELAGEM DE FILAS MARKOVIANAS

Trabalho de Conclusão de Curso (Artigo) apresentado ao curso de Bacharelado em Estatística do Departamento de Estatística do Centro de Ciências e Tecnologia da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de bacharel em Estatística.

Trabalho aprovado em 27 de SETEMBRO de 2021.

BANCA EXAMINADORA

Prof. Gustavo Henrique Esteves (Coorientador)
Universidade Estadual da Paraíba (UEPB)

Prof. Tiago Almeida de Oliveira
Universidade Estadual da Paraíba (UEPB)

Prof. Ednário Barbosa de Mendonça
Universidade Estadual da Paraíba (UEPB)

Este trabalho é todo dedicado aos meus pais, pois é graças ao seu esforço que hoje posso concluir o meu curso.

“As pequenas oportunidades são, frequentemente, o início de grandes empreendimentos”
(Demóstenes)

SUMÁRIO

QUEUEPB: UMA PROPOSTA DE PACOTE PARA MODELAGEM DE FILAS MARKOVIANAS

Mayara C. de A. Silva *
Divanilda Maia Esteves †
Gustavo Henrique Esteves‡

RESUMO

Em um sistema de filas, supomos que "usuários" chegam a um posto de atendimento buscando algum serviço. O tempo entre chegadas é uma variável aleatória e o tempo para realizar o serviço é outra variável aleatória. Devido a esse caráter aleatório é impossível garantir que términos de serviços coincidam exatamente com chegadas de usuários. Consequentemente, há vezes que o serviço completa sua tarefa com um usuário e não encontra mais ninguém disponível com quem trabalhar. Neste caso, ele fica ocioso até que o próximo usuário chegue. Mas também há vezes que um usuário chega e já encontra o serviço ocupado com alguma chegada anterior. Este usuário deve então aguardar a sua vez ou partir, isso dependerá da estrutura do sistema. O objetivo deste trabalho foi realizar um estudo teórico e computacional de teoria de filas, e propor um pacote para modelagem de filas. O pacote foi desenvolvido no *software* R, com uso da interface *RStudio*, os modelos considerados foram $M|M|1|\infty|FIFO$, $M|M|c|\infty|FIFO$, e a função para simulação do modelo $M|M|c$ com desistência. O mesmo `queuepb` foi desenvolvido com o sistema de controle de versionamento de código conhecido como *Git* e está disponível na plataforma *GitHub*[1]. O propósito da criação do pacote é auxiliar o processo de análise dos modelos de fila, permitindo avaliar a eficiência do sistema.

Palavras-chaves: Processos estocásticos. Simulação no R. Medidas de desempenho.

ABSTRACT

In a queuing system, we assume that "users" arrive at a service post looking for some service. The time between arrivals is a random variable and the time to service is another random variable. Due to this random character it is impossible to guarantee that service terminations exactly coincide with user arrivals. Consequently, there are times when the service completes its task with one user and doesn't find anyone else available to work with. In this case, it is idle until the next user arrives. But there are also times when a user arrives and finds the service already occupied with some previous arrival. This user must then wait their turn or leave, it will depend on the structure of the system. The objective of this work was to carry out a theoretical and computational study of queuing theory, and to propose a package for queuing modeling. The package was developed in *software* R, using the *RStudio* interface, the models considered were $M|M|1|\infty|FIFO$, $M|M|c|\infty|FIFO$, and the function for simulating the model $M|M|c$ with surrender. The same `queuepb` was developed with the code versioning control system known as *Git* and is available on the platform *GitHub*[2]. The purpose of creating the package is to assist the process of analyzing the queue models, allowing you to assess the efficiency of the system.

Keywords: Stochastic processes. R Simulation. Performance measures.

* Mayara C. de A. Silva, Depto Estatística, UEPB, Campina Grande, PB, mayara010897@gmail.com
† Prof. Divanilda Maia Esteves, Depto Estatística, UEPB, Campina Grande, PB, diana.maia@servidor.uepb.edu.br
‡ Prof. Gustavo Henrique Esteves, Depto Estatística, UEPB, Campina Grande, PB, gesteves@servidor.uepb.edu.br
1 <https://github.com/mayara124/queuepb>
2 <https://github.com/mayara124/queuepb>

1 INTRODUÇÃO

Passar a compra no caixa de supermercado, esperar atendimento médico, pagar uma conta em uma loteria são fatos comuns à vida da maioria das pessoas e que reforçam a importância do estudo teórico dos sistemas de filas. O objetivo principal desses estudos é avaliar o desempenho na espera e no atendimento, buscando equilibrar eficiência e custos, proporcionando satisfação ao cliente, sem onerar excessivamente quem oferece o serviço.

De forma simplificada, um sistema de filas envolve primeiramente "usuários" que chegam a um posto de atendimento, buscando serviço. Esses usuários, em geral, são pessoas, mas também podem ser máquinas/equipamentos em uma fila de conserto, processos em uma fila esperando julgamento ou até lotes de produtos a serem vendidos, por exemplo. O tempo entre duas novas entradas/chegadas consecutivas é uma variável aleatória e o tempo de realização do serviço é outra variável aleatória. Desta forma, muito frequentemente um usuário entra no sistema e outro usuário está sendo "atendido". Com isso, ele pode esperar para ser atendido, formando uma fila de espera, ou ir embora, como no caso de algumas ligações telefônicas. Outras vezes, o serviço é finalizado e não há mais ninguém esperando para ser atendido. Neste caso, o posto de atendimento fica ocioso até que o próximo usuário chegue.

Diversas características podem interferir no desempenho do sistema, de modo que seu funcionamento passa a ser função delas. Essas características podem ser classificadas em: forma dos atendimentos, forma das chegadas, disciplina da fila e estrutura do sistema.

Assim, o objetivo deste trabalho foi realizar um estudo teórico e computacional de teoria de filas e propor um pacote para modelagem de filas markovianas no *software* R (R Core Team, 2021).

A parte teórica foi orientada pela professora Divanilda Maia Esteves e a parte computacional contou com a colaboração do professor Gustavo Henrique Esteves.

2 METODOLOGIA

Neste capítulo serão apresentados alguns conceitos importantes que foram estudados acerca da teoria das filas. O estudo teórico foi baseado no livro (FOGLIATTI; MATTOS, 2007) visto que na literatura não há outras referências com bases consolidadas. Inicialmente serão apresentados os conceitos de processo estocástico e de processo de nascimento e morte, seguidos de alguns aspectos de um sistema de filas, como as medidas de desempenho e uma síntese dos modelos estudados.

2.1 Processo Estocástico

Um processo estocástico é uma sequência de variáveis aleatórias $\{X(t);\, t \in T\}$ que descreve a evolução de um processo ao longo do tempo. Em particular, diz-se que um processo estocástico é markoviano (de ordem 1) quando

$$P\left[X(t) \leq x \mid X(t_n) = x_n, X(t_{n-1}) = x_{n-1}, \ldots, X(t_0) = x_0\right] = P\left[X(t) \leq x \mid X(t_n) = x_n\right],$$

ou seja, para prever a posição do processo no futuro apenas a informação mais recente é relevante (ROSS, 2007).

Quando as variáveis $X(t)$ são discretas, então tem-se uma cadeia de Markov. Neste caso, para $s > 0$, a probabilidade

$$P_{x,y}(t, t+s) = P\left[X(t+s) = y \mid X(t) = x\right]$$

é chamada probabilidade de transição. Em particular, se $P_{x,y}(t,t+s)$ não depende de t, então a cadeia é homogênea no tempo e consequentemente a probabilidade de ir de um estado x para um estado y em uma unidade de tempo será denotada simplesmente por $P(x,y)$. Diz-se ainda que uma cadeia é irredutível quando todos os estados do processo podem ser alcançados pelos demais.

2.1.1 Processo de nascimento e morte

Uma cadeia de Markov homogênea, irredutível, de parâmetro contínuo, é denominada processo de nascimento e morte (*Birth-Death process*) se as únicas mudanças permitidas a partir de um determinado estado n do processo são para seus vizinhos imediatos, quando a transição ocorre para $n+1$ representa um nascimento e para o estado $n-1$ ocorre uma morte para todo $n>0$. Essas transições se processam com taxas:

$$P(n,n+1) = \lambda_n$$
$$P(n,n-1) = \mu_n.$$

Considerando que este processo assuma valores no conjunto $\{0,1,2,\dots\}$, para modelar sua evolução é preciso conhecer a probabilidade P_i de o sistema estar no estado i. A partir das taxas de transição e considerando-se que $\sum_{i\geq 1} P_i = 1$, obtém-se um sistema de equações cuja solução $(P_0, P_1, P_2 ...)$ é a distribuição limite dos estados do sistema e coincide com a distribuição do estado no regime estacionário. Neste caso,

$$P_0 = \frac{1}{1+\sum_{n\geq 1}\prod_{i=1}^{n}\frac{\lambda_{i-1}}{\mu_i}}$$
$$P_n = P_0\prod_{i=1}^{n}\frac{\lambda_{i-1}}{\mu_i} n \geq 1,$$

desde que $\rho_i = \frac{\lambda_{i-1}}{\mu_i} < 1$.

2.2 Sistemas de filas

A teoria de filas consiste na modelagem analítica de processos ou sistemas que resultam em espera e tem como objetivo determinar e avaliar quantidades que expressam sua produtividade/operacionalidade (FOGLIATTI; MATTOS, 2007).

Para descrever um sistema com fila, será utilizada a notação proposta por Kendall (1953), que é a forma $A|B|C|D|E$, onde A, B, C, D e E denotam, respectivamente, as distribuições dos tempos entres as chegadas sucessivas e de atendimentos, o número de postos de atendimento, a capacidade física do sistema e a disciplina de atendimento.

Quando o tempo entre chegadas é modelado de acordo com uma distribuição exponencial, o número de chegadas em t unidades de tempo tem distribuição Poisson com taxa λ_t e, consequentemente, as chegadas ocorrem de acordo com um processo de Poisson. Neste caso, o processo de chegadas ao sistema é markoviano (ROSS, 2007). O mesmo ocorre com o processo de atendimento. Assim, quando $A = M$ e/ou $B = M$ significa que as chegadas e/ou saídas são markovianas, respectivamente. Isso quer dizer, em outras palavras, que os tempos entre chegadas consecutivas e o tempo de atendimento são variáveis aleatórias exponencialmente distribuídas (ROSS, 2007).

Em geral, o número de postos de atendimentos será um número inteiro $C = k \geq 1$. Quando pessoas esperam em uma fila de caixa rápido com três caixas, por exemplo, $C = 3$. Já quando o sistema é de autoatendimento, como no preenchimento de um formulário online, usa-se $C = \infty$, o que significa que não há uma quantidade limitada de servidores.

A capacidade do sistema é um número inteiro $D \geq 0$. Quando $D = 0$, o sistema não comporta fila. É o caso, por exemplo, das ligações telefônicas. Quando $D = k > 0$, quer dizer que até k "pessoas" podem aguardar pelo atendimento na fila e as que ultrapassarem este limite serão "dispensadas". Na maioria dos casos, considera-se $D = \infty$, ou seja, que todos que chegarem ao sistema podem esperar na fila pelo atendimento.

A disciplina de atendimento (E) é um critério pré-estabelecido que regulamenta o modo como as "pessoas" serão atendidas quando o posto de atendimento fica disponível. Os casos geralmente considerados são:

- FIFO (*first in - first out*): os usuários são atendidos na ordem das chegadas.
- LIFO (*last in - first out*): o primeiro usuário a ser atendido é o que chegou por último.
- PRI (*priority service*): o atendimento dos usuários segue uma ou mais prioridades preestabelecidas pela gerência do sistema.
- SIRO (*service in random order*): o atendimento aos usuários segue uma ordem aleatória.

Muitas vezes, as posições D e E são omitidas e, nestes casos, considera-se que $D = \infty$ e $E = FIFO$. Ou seja, o sistema tem capacidade de fila ilimitado e quem entra primeiro no sistema é atendido primeiro.

2.2.1 Medidas de desempenho

As medidas de desempenho são aspectos considerados no intuito de avaliar a eficiência de um sistema com fila e são determinadas de acordo com a estrutura do sistema. Aqui serão consideradas as seguintes:

- Número médio de usuários na Fila (L_q) e no sistema (L).
- Tempo médio de espera de um usuário qualquer na fila (W_q).
- Tempo médio de permanência de um usuário qualquer no sistema (W).

Outras medidas (mais pormenorizadas) úteis:

- P_n é a probabilidade de existirem n clientes no sistema.
- Probabilidade de existirem no sistema k ou mais clientes, $P(n \geq k)$.
- Probabilidade de um usuário qualquer ter que aguardar mais do que um determinado tempo t na fila, $P(W_q > t)$
- $P(W_q = 0)$ é a probabilidade de o tempo de espera na fila ser zero.
- $P(W > t)$ é a probabilidade de o tempo gasto no sistema exceder t.

2.2.2 *Modelo M|M|1*

Este modelo é caracterizado por ter:

- tempos entre chegadas sucessivas e os tempos de atendimentos exponenciais, com parâmetros λ e μ, respectivamente;
- um único posto de atendimento;
- espaço reservado para a fila de espera sem limitação;
- ordem de acesso dos usuários ao serviço seguindo a ordem das chegadas dos mesmos ao sistema.

As taxas de chegadas (ingresso) ao sistema e de atendimento são constantes e dadas respectivamente por:

$$\lambda_n = \lambda, \quad \forall n \geq 0 \quad \text{e} \quad \mu_n = \mu, \quad \forall n \geq 1,$$

e o parâmetro ρ, denominado como a taxa de ocupação/utilização do sistema, será

$$\rho = \frac{\lambda}{\mu}.$$

As medidas de desempenho para este sistema podem ser visualizadas na Tabela 1 e são obtidas sob a condição de que $\rho < 1$.

Tabela 1 – Medidas de desempenho para o sistema de filas $M|M|1$.

Número médio de usuários no sistema	$L = \frac{\rho}{(1-\rho)}$
Número médio de usuários na fila	$L_q = \frac{\rho^2}{(1-\rho)}$
Tempo médio de espera na fila	$W_q = \frac{\rho}{\mu-\lambda}$
Tempo médio de permanência no sistema	$W = \frac{1}{\mu-\lambda}$
Probabilidade de que o sistema tenha $n \geq 0$ clientes	$P_n = \rho^n(1-\rho)$
Probabilidade de que o sistema esteja desocupado	$P_0 = 1 - \frac{\lambda}{\mu}$
Prob. de tempo de espera na fila ser maior do que $t > 0$	$W_q(t) = P(T_q > t) = \rho e^{-(\mu-\lambda)t}$
Probabilidade de haver mais do que k elementos no sistema	$P(n > k) = \rho^{k+1}$
Probabilidade de o tempo de espera na fila ser zero	$P(W_q = 0) = P_0$
Probabilidade de o tempo gasto no sistema exceder $t > 0$	$P(W > t) = e^{-(\mu-\lambda)t}$

2.2.3 *Modelo M|M|c*

Este modelo é caracterizado por ter:

- tempos entre chegadas sucessivas seguindo distribuição exponenciais de parâmetro λ;
- c servidores, cada um dos quais com tempos de atendimentos que seguem distribuições exponenciais, de parâmetro μ;
- espaço reservado para a fila de espera sem limitação;
- o usuário sendo atendido conforme ordem de chegada;

Neste contexto, as taxas de chegadas e de serviços são dadas por:

$$\lambda_n = \lambda, \quad \forall n \geq 0$$
$$\mu_n = \begin{cases} n\mu, & \text{se } 1 \leq n < c, \\ c\mu, & \text{se } n \geq c \end{cases}$$

e, definindo $r = \frac{\lambda}{\mu}$, a taxa de utilização do sistema é dada por

$$\rho = \frac{\lambda}{c\mu} = \frac{r}{c}.$$

Consequentemente, a probabilidade de se ter pelo menos n usuários no sistema é

$$P_n = \begin{cases} P_0 \cdot \dfrac{r^n}{r!}, & \text{se } 1 \leq n < c, \\ \\ \dfrac{r^n}{c!\, c^{n-c}}, & \text{se } n \geq c. \end{cases}$$

A probabilidade de que o sistema esteja desocupado é

$$P_0 = \left(\sum_{n=0}^{c-1} \frac{r^n}{n!} + \frac{c\, r^c}{c!\,(c-r)} \right)^{-1}.$$

O número médio de elementos na fila e o número médio de elementos no sistema são, respectivamente

$$L_q = P_0 \cdot \frac{c\, r^{c+1}}{c!\,(c-r)^2} \quad \text{e} \quad L = r + \left(\frac{r^{c+1c}}{c!(c-r)^2} \right) P_0.$$

Tempo médio de espera na fila e o tempo médio de permanência no sistema são, respectivamente

$$W_q = \frac{r\, c\mu}{(c-1)!(c\mu-\lambda)^2} \cdot P_0 \quad \text{e} \quad W = \frac{1}{\mu} + \left(\frac{r^c \mu}{(c-1)!(c\mu-\lambda)^2} \right) \cdot P_0.$$

Por fim, a função de distribuição acumulada do tempo de espera na fila é

$$W_q(t) = 1 - P_0 \frac{r^c}{c!(1-\rho)} e^{-(c\mu-\lambda)^t}.$$

2.2.4 Modelo $M_n|M|1|\infty|FIFO$

Este modelo se diferencia do modelo $M|M|1$ que foi estudado, pois as taxas de ingresso (λ_n) mudam de acordo com o estado do sistema. Essas mudanças podem ser expressas como funções que decrescem/crescem e representam o comportamento de usuários que desistem (ou não) a depender do número de usuários no sistema. Em geral, considerando

$$\alpha_n = \frac{1}{n+1}, \quad \forall n = 0, 1, 2, \ldots$$

dessa forma as taxas de chegadas são

$$\lambda_n = \frac{\lambda}{n+1}, \quad \forall n \geq 1 \text{ e } \lambda \geq 0 \text{ constante}$$

e as taxas de atendimento permanecem contantes, dadas por

$$\mu_n = \mu, \quad \forall n \geq 1.$$

Nesta configuração, as medidas de desempenho são diferentes daquelas usadas para o modelo $M|M|1$ e serão vistas a seguir.

A probabilidade do sistema estar desocupado é

$$P_0 = \left[1 + \sum_{n=1}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}\right]^{-1} = \left[\sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!}\right]^{-1} = e^{-\frac{\lambda}{\mu}}.$$

A probabilidade de que n usuários encontram-se no sistema.

$$P_n = \left(\frac{\lambda}{\mu}\right)^n \frac{e^{-\frac{\lambda}{\mu}}}{n!}.$$

Como neste modelo considera-se apenas um servidor, o coeficiente de utilização do sistema é dada por:

$$\rho = 1 - e^{-\frac{\lambda}{\mu}}$$

Além disso, o número médio de usuários no sistema e na fila são, respectivamente,

$$L = \frac{\lambda}{\mu} \quad \text{e} \quad L_q = \frac{\lambda}{\mu} - \rho$$

e os tempos médios de permanência no sistema e na fila são, respectivamente,

$$W = \frac{\lambda}{\mu^2\left(1 - e^{-\frac{\lambda}{\mu}}\right)} \quad \text{e} \quad W_q = \frac{\lambda - \mu\rho}{\mu^2\left(1 - e^{\frac{\lambda}{\mu}}\right)}.$$

2.3 Teste de aderência

Este teste de aderência avalia a concordância entre a distribuição observada da amostra e uma determinada distribuição teórica. Por exemplo, no processo entre chegadas obtemos uma distribuição para a taxa de chegada, estamos interessados em testar se ela é uma exponencial. Este tipo de teste, chamado de teste de Aderência, contém várias formas e veremos umas dela conhecida como o teste de Kolmogorov–Smirnov. Para isso utiliza-se a função distribuição acumulada observada, compara-se com a teórica, determina-se o ponto em que estas distribuições mais divergem, e testa-se se essa divergência é significativa ou não.

Método

Seja $F_0(x)$ uma distribuição teórica acumulada e $S_n(x)$ uma distribuição acumulada observada em uma amostra de n observações.

Encontra-se a seguir o maior valor das diferenças entre $F_0(x) e S_n(x)$, ou seja,

$$D = \text{máximo}|F_0(x) - S_n(x)|$$

Compara-se o valor observado com o valor crítico (Tábua E - (KITTEL, 1970)) bilateral. Na tabela associa-se o valor observado com o seu p-valor.

Para fazer o teste utilizamos a função ks.test já implementada no *software* R (R Core Team, 2021). No exemplo mencionado anteriormente queriamos testar se os tempos entre chegadas segue uma distribuição de exponencial$(\frac{1}{\lambda})$.

```
test.1 <- ks.test(data[,1],"pexp",1/mean(data[,1]))
p.value <- test.1$p.value
#alpha <- 0.05
if( p.value <= alpha){
stop("the data does not follow an exponential distribution")
```

Na função especificámos o conjunto de dados onde contém os tempos entre chegadas, o parâmetro 'pexp' representa a distribuição exponencial com taxa $1/\lambda$. Logo após extraímos o p-valor e observamos se é menor ou igual ao nosso alfa especificado pelo usuário, se a condição for aceita calculamos as medidas de desempenho caso contrário o processo é interrompido e uma mensagem informando que 'os dados não seguem uma distribuição exponencial'.

3 RESULTADOS E DISCUSSÃO

A modelagem de um sistema de filas busca conhecer seu funcionamento para otimizar tais sistemas, de forma a reduzir seus custos operacionais e melhorar seu desempenho. Sendo assim, surgiu a ideia de desenvolver funções no *software* R (R Core Team, 2021) que pudessem auxiliar esse processo e que sirvam de ponto de partida para a criação de um pacote de análise de sistemas de filas.

O estudo da teoria, bem como as primeiras funções foram desenvolvidas em um projeto de iniciação científica (SILVA, 2019). Os modelos considerados foram $M|M|1|\infty|FIFO$, $M|M|c|\infty|FIFO$, e a função para simulação do modelo $M|M|c$ com desistência, em colaboração para o pacote, implementada por Cordeiro (2020), considerando $c = 1$ na simulação, temos o modelo $M_n|M|1|\infty|FIFO$ (taxas de ingresso dependem dos estados do sistema).

Foi implementada uma função, `r.queue`, para simular os modelos de filas básicos, que são $M|M|1$ e $M|M|c$. Nessa função o usuário especifica o número de servidores, representado por c, as taxas médias de chegadas e serviços, λ e μ, respectivamente e o número de simulações que deseja realizar. Na sequência foram implementadas funções para calcular as medidas de desempenho, `fit.mm1` e `fit.mmc`, sendo que neste caso o usuário deve inerir dados reais ou simulados em forma matricial. Além das funções para cálculo das medidas de desempenho, foram implementadas funções para verificar se os tempos podem ser considerados exponenciais, o que caracteriza a fila markoviana. Caso os dados sejam realmente de uma fila markoviana, são calculadas medidas de desempenho do sistema. Foram ainda implementadas funções para calcular a probabilidade acumulada do tempo de espera na fila, `wqt.mm1` e `wqt.mmc`, para um certo tempo t, e a função de probabilidade acumulada do tempo de permanência no sistema, `wt.mm1` e `wt.mmc`.

O pacote foi desenvolvido no *software* R, com uso da interface *RStudio*. Foram utilizados os pacotes `devtools` (WICKHAM; HESTER; CHANG, 2021), que possibilita construir, conferir a documentação, checar e fazer a instalação das funções e `roxygen2` (WICKHAM et al., 2020), o qual simplifica a etapa de criação e gerenciamento da documentação das funções criadas.

A estrutura do pacote por padrão do R é composta por dois diretórios e dois arquivos de metadados:

- DESCRIPTION: um arquivo texto contendo as informações sobre o pacote, como título, finalidade do pacote, autor, licença, pacotes dependentes, versão do pacote.

- NAMESPACE: um arquivo texto que informa quais funções do pacote são exportadas e que estarão disponíveis para os usuários e quais funções precisam ser importadas de outros pacotes.

- R/: diretório contendo as funções criadas em *scripts* do R, isto é, arquivos com a extensão `*.R`.

- man/: diretório contendo a documentação das funções criadas, em arquivos com a extensão `*.Rd` que dão origem às páginas de ajuda do pacote no R.

O pacote `queuepb` foi desenvolvido com o sistema de controle de versionamento de código conhecido como *Git* e está disponível na plataforma *GitHub*[3]. Para a instalação e uso do pacote inicialmente usa-se:

```
> install.packages("devtools")
> devtools::install_github("mayara124/queuepb")
> library(queuepb)
```

Instalado o pacote, suponha que se deseja simular um sistema de fila com dois guichês de atendimento, no qual passaram 20 usuários. Considere que os tempos médios entre chegadas sucessivas e de atendimento são, respectivamente, 5 minutos e 4 minutos, o que implica que $\lambda = \frac{1}{5} = 0,2$ e $\mu = \frac{1}{4}$. Neste caso, o comando usado e a saída seriam:

```
> data <- r.queue(20,2,5,4)

             tdc        tcs        saida   guiche
 [1,]   1.220276   1.220276     4.191220     1
 [2,]   5.547534   5.547534     8.172413     2
 [3,]  14.423175  14.423175    25.152479     2
 [4,]  31.157026  31.157026    43.264419     2
 [5,]  34.032510  34.032510    38.895872     1
 [6,]  47.427701  47.427701    51.108365     2
 [7,]  51.777046  51.777046    52.449097     1
 [8,]  69.118305  69.118305    90.257281     2
 [9,]  71.044264  71.044264    71.126190     1
[10,]  80.495342  80.495342    80.744468     1
[11,]  81.610222  81.610222    81.853741     1
[12,]  94.558456  94.558456    99.693343     2
[13,]  99.592959  99.592959   113.814469     1
[14,] 105.793894 105.793894   105.905723     2
[15,] 106.295804 106.295804   108.082935     2
[16,] 109.684924 109.684924   109.799373     2
[17,] 121.775177 121.775177   127.572349     2
[18,] 127.764433 127.764433   129.886233     1
[19,] 128.570329 128.570329   132.230941     2
[20,] 132.880205 132.880205   139.369179     1
```

A função retorna uma matriz com quatro colunas, que representam, para cada usuário, quatro valores:

- **tdc:** tempo de chegada, ou seja, quantos minutos se passaram entre o momento que o sistema começou a funcionar e a chegada desse cliente;

- **tcs:** quantos minutos se passaram entre o momento que o sistema começou a funcionar e o instante de início do seu atendimento;

[3] <https://github.com/mayara124/queuepb>

- **saida:** quantos minutos se passaram entre o momento que o sistema começou a funcionar e o instante do término do seu atendimento;
- **guiche:** número do guichê no qual o cliente foi atendido.

Quando o cliente chega, ele "escolhe" aleatoriamente um dos guichês vazios para ir receber o serviço. Se todos os atendentes estiverem ocupados, ele esperará na fila e irá para o primeiro que ficar vazio. No exemplo, observe que os tempos de chegada (tdc) são iguais aos tempos de início do serviço (tds), o que indica que quando o cliente chegou, havia algum servidor desocupado. Podemos calcular as medidas de desempenho através do comando:

```
> fit.mmc(data,20,2,0.05)

Rate Arrival =  0.1505115

Rate Service =  0.2024281

System Probability is Empty  =  0.4579753

                            L        Lq        W           Wq
Performance Measures 1.040726 0.2971957 4.943341 0.003315554
```

Têm-se os valores estimados dos tempos de chegada e de atendimento, o número médio de usuários na Fila (L_q) e no sistema (L), tempo médio de espera de um usuário qualquer na fila (W_q), tempo médio de permanência de um usuário qualquer no sistema (W), e a probabilidade do sistema estar vazio.

Considerando agora um segundo exemplo onde ainda há dois guichês de atendimento e os 20 usuários, mas com a alteração dos tempos médios de chegadas sucessivas e de atendimento, considerando-se respectivamente 4 e 6 minutos, resultando $\lambda = \frac{1}{4}$ e $\mu = \frac{1}{6}$. Neste novo cenário obtém-se:

```
> data <- r.queue(20,2,4,6)

            tdc         tcs     saida   guiche
[1,]   0.6776948   0.6776948  33.10007      1
[2,]  10.8926457  10.8926457  14.66672      2
[3,]  11.1725944  14.6667204  53.34185      2
[4,]  11.9893372  33.1000681  45.05096      1
[5,]  16.4957337  45.0509631  45.05224      1
[6,]  23.6509832  45.0522356  50.97520      1
[7,]  24.0415223  50.9751985  57.45689      1
[8,]  24.0851359  53.3418497  53.88770      2
[9,]  31.8330805  53.8876969  72.29731      2
[10,] 32.9283134  57.4568858  64.48948      1
[11,] 34.4307142  64.4894759  65.77092      1
[12,] 35.5522403  65.7709164  67.77654      1
[13,] 40.0509948  67.7765441  69.75746      1
[14,] 50.9342009  69.7574621  76.68137      1
[15,] 61.5105860  72.2973145  76.91488      2
[16,] 68.9081237  76.6813680  79.19480      1
[17,] 74.6765615  76.9148804  80.74496      2
```

```
[18,] 75.4054733  79.1947995  82.22960      1
[19,] 77.5925854  80.7449628  81.10156      2
[20,] 81.9665498  81.9665498  82.55828      2
```

Diferentemente do exemplo anterior, agora há a formação de uma fila. Observe que os dois primeiros clientes encontram algum dos guichês vazio, mas o terceiro cliente chega e os dois guichês estão ocupados, pois seu tempo de chegada (tdc = 11.1725944 na terceira linha) é menor do que os tempos de saída do primeiro (saida = 33.10007 na primeira linha) e do segundo (saida = 14.66672 na segunda linha) usuários. Como o tempo de saída do segundo usuário é menor do que o do primeiro, o guichê 2 foi o primeiro a ser desocupado e o usuário três foi para lá. No período entre o tempo da chegada e o início do atendimento, o usuário espera na fila, como foi dito anteriormente.

As medidas de desempenho neste caso são

```
> fit.mmc(data,20,2,0.05)
Rate Arrival =  0.2229394

Rate Service =  0.1653172

System Probability is Empty  =  0.194545

                            L        Lq         W          Wq
Performance Measures 1.551035 0.2024796 6.049656 0.00067837
```

Em relação ao primeiro exemplo, a probabilidade de o sistema ficar ocioso foi de $0,4579753$ para $0,194545$. Isso aconteceu porque, apesar de o número de postos de atendimento ser o mesmo, o tempo médio entre chegadas diminuiu de 5 para 4 minutos e o tempo médio de serviço aumentou de 4 para 6 minutos.

As taxas de ocupação no primeiro e no segundo casos são, respectivamente,

$$\rho_1 = \frac{\frac{1}{5}}{2 \cdot \frac{1}{4}} = 0,4 \quad \text{e} \quad \rho_2 = \frac{\frac{1}{4}}{2 \cdot \frac{1}{6}} = 0,75.$$

Isso reforça a teoria de que quanto mais próximo de 1 estiver o ρ, "piores" serão as medidas de desempenho.

4 CONCLUSÃO

A realização deste trabalho possibilitou estudo teórico de aspectos centrais de alguns modelos de filas markovianas e a implementação das funções para o pacote não apenas consolidou a teoria vista, mas também foi um desafio no aspecto computacional. A implementação das medidas de desempenho é relativamente simples, mas a simulação de tais modelos envolve muitos fatores, como por exemplo a variabilidade dos servidores que podem estar livres para serem escolhidos pelo usuário que chega. Apesar da grande aplicabilidade da teoria de filas markovianas, em situações práticas há problemas como a dificuldade da coleta dos dados. Sendo assim, a possibilidade de simular uma fila foi fundamental para testar as demais funções. No caso do uso de um conjunto de dados reais, inicialmente será verificado se os tempos entre chegadas e de atendimento podem ser considerados exponencias, o que caracteriza a fila markoviana. Depois são estimadas as taxas λ e μ e calculadas as medidas de desempenho. Se a taxa de ocupação

$\rho > 1$ será dada uma mensagem indicando a impossibilidade do cálculo dessa medidas, pois o sistema neste caso não está em equilíbrio.

Deve-se ressaltar que não foi um estudo completo do conteúdo, dado que há vários outros aspectos que podem ser estudados através de outros modelos de filas, como por exemplo, o caso em que se considera que os tempos entre chegadas e/ou de atendimento não são exponencialmente distribuídos e aqueles casos em que há atendimento preferencial. Assim, o pacote está ainda - e provavelmente sempre estará - em desenvolvimento, para que outras situações sejam consideradas.

O propósito da criação do pacote é auxiliar o processo de análise dos modelos de fila, permitindo avaliar a eficiência do sistema, o que pode dar indícios de melhorias que podem ser feitas.

REFERÊNCIAS

CORDEIRO, D. d. S. *Estudo de simulação de um sistema na fila do tipo M/M/C com desistência.* 2020. 27 p. Monografia (Bacharel em Estatística), UEPB (Universidade Estadual da Paraíba), Campina Grande, Brasil. Citado na página 13.

FOGLIATTI, M. C.; MATTOS, N. M. C. *Teoria de filas*. [S.l.: s.n.], 2007. Citado 2 vezes nas páginas 7 e 8.

KENDALL, D. G. Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded markov chain. *The Annals of Mathematical Statistics*, JSTOR, p. 338–354, 1953. Citado na página 8.

KITTEL, E. *Siegel*. [S.l.]: Klinkhardt & Biermann, 1970. Citado na página 12.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2021. Disponível em: <https://www.R-project.org/>. Citado 3 vezes nas páginas 7, 12 e 13.

ROSS, S. *Introdução aos modelos de probabilidade. Nono. [Sl]*. 2007. Citado 2 vezes nas páginas 7 e 8.

SILVA, M. josé Lima da. Desafios e perspectivas da pesquisa técnico-científica na contemporanidade. In: *XXVI Encontro de Iniciação Científica – Campina Grande: UEPB*. Campina Grande: [s.n.], 2019. p. 74. Citado na página 13.

WICKHAM, H. et al. *roxygen2: In-Line Documentation for R*. [S.l.], 2020. R package version 7.1.1. Disponível em: <https://CRAN.R-project.org/package=roxygen2>. Citado na página 13.

WICKHAM, H.; HESTER, J.; CHANG, W. *devtools: Tools to Make Developing R Packages Easier*. [S.l.], 2021. R package version 2.4.1. Disponível em: <https://CRAN.R-project.org/package=devtools>. Citado na página 13.

AGRADECIMENTOS

A Deus por ter me dado saúde e força para superar as dificuldades.

Agradeço a minha mãe Eliene, que me deu apoio, incentivo nas horas difíceis, de desânimo e cansaço.

Ao meu pai Geraldo que, apesar das dificuldades, se manteve forte e me ajudou e que para mim, foi muito importante.

Obrigada a meus irmãos, Jéssica e Gerismar, pelo apoio que sempre me deram durante toda minha vida.

Ao meu noivo Bruno, por sempre me apoiar durante meu percurso acadêmico.

Aos meus amigos, agradeço por todo amor, força, incentivo.

Aos amigos da universidade Guilherme, Débora, Elizandra, Gilmar, Hiago, Samuel, Sóstenes, Willian, Matheus. Obrigada pelos inúmeros conselhos, motivação e puxões de orelha. As risadas que vocês compartilharam comigo nessa etapa tão desafiadora da vida acadêmica, também fizeram toda diferença.

Sou grata pela confiança depositada na proposta do projeto, a Diana e Gustavo por orientarem meu trabalho. Obrigada por me manter motivada durante todo o processo.

A todos os meus professores do curso de Bacharelado em Estatística da UEPB pela excelência da qualidade técnica de cada um.