

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Nova classe de modelos paramétricos para análise de sistemas reparáveis

Tito Lívio da Cunha Lopes

Tese de Doutorado do Programa Interinstitucional de Pós-Graduação em Estatística (PIPGEs)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Tito Lívio da Cunha Lopes

Nova classe de modelos paramétricos para análise de sistemas reparáveis

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP e ao Departamento de Estatística – DEs-UFSCar, como parte dos requisitos para obtenção do título de Doutor em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística. *VERSÃO REVISADA*

Área de Concentração: Estatística

Orientadora: Profa. Dra. Vera Lúcia Damasceno Tomazella

USP – São Carlos
Dezembro de 2022

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

L864n Lopes, Tito Lívio da Cunha
 Nova classe de modelos paramétricos para análise
de sistemas reparáveis / Tito Lívio da Cunha Lopes;
orientadora Vera Lúcia Damasceno Tomazella. -- São
Carlos, 2022.
 161 p.

 Tese (Doutorado - Programa Interinstitucional de
Pós-graduação em Estatística) -- Instituto de Ciências
Matemáticas e de Computação, Universidade de São
Paulo, 2022.

 1. Pontos de mudança. 2. Processo de lei de
potência. 3. Processo Poisson não homogêneo. 4.
Reparo imperfeito. I. Tomazella, Vera Lúcia
Damasceno, orient. II. Título.

Tito Lívio da Cunha Lopes

New class of parametric models for analysis of repairable systems

Doctoral dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP and to the Department of Statistics – DEs-UFSCar, in partial fulfillment of the requirements for the degree of the Doctorate Interagency Program Graduate in Statistics. *FINAL VERSION*

Concentration Area: Statistics

Advisor: Profa. Dra. Vera Lúcia
 Damasceno Tomazella

**USP – São Carlos
December 2022**



UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências Exatas e de Tecnologia
Programa Interinstitucional de Pós-Graduação em Estatística

Folha de Aprovação

Defesa de Tese de Doutorado do candidato Tito Lívio da Cunha Lopes, realizada em 13/01/2023.

Comissão Julgadora:

Profa. Dra. Vera Lucia Damasceno Tomazella (UFSCar)

Prof. Dr. Marcelo Bourguignon Pereira (UFRN)

Prof. Dr. Pedro Luiz Ramos (PUC-Chile)

Prof. Dr. Eder Angelo Milani (UFG)

Prof. Dr. Afrânio Márcio Corrêa Vieira (UFSCar)

O Relatório de Defesa assinado pelos membros da Comissão Julgadora encontra-se arquivado junto ao Programa Interinstitucional de Pós-Graduação em Estatística.

*Dedico à minha querida esposa Bruna Cristian e à nossa pequena flor
cuja a essência ainda perfuma nossas vidas.*

AGRADECIMENTOS

Sou muito grato a Deus por todas as oportunidades que me concedeu, por sua graça e infinita misericórdia. Sem Ele, certamente não chegaria até aqui.

Agradeço à minha querida esposa Bruna pelo companheirismo, dedicação, orações e carinho que são fundamentais em todos os projetos.

Agradeço à minha família, em especial à minha querida mãe Antônia por todo seu cuidado, incentivo, pelas palavras de apoio e orações, que mesmo diante das limitações sempre acreditou na educação; aos meus irmãos Jorge Gaitam, Gabriela e Tales pelos incentivos e apoios e ao meu pai Jorge Lopes.

Agradeço à minha sogra Zilmar e aos avós da minha esposa, Sra. Francisca (in memoriam) e Sr. Francisco por todos os diferentes tipos de apoio que me deram desde a graduação.

Sou grato ao Pr. Gonçalves (in memoriam) que mesmo com dificuldades de locomoção, me levou à vários lugares que eu sozinho não poderia ir.

Agradeço a todos os colegas do TRF5 pelo apoio e incentivo, em especial à Laryssa, que prontamente me liberou para licença para capacitação.

Agradeço à minha orientadora, professora Vera L. D. Tomazella, por sua orientação, conhecimento, ensinamentos e incentivo durante a preparação desta tese.

Agradeço a todos os servidores, colegas e professores da UFPI, UFRN, UFSCar e da USP pela amizade e conhecimentos compartilhados.

Por fim, agradeço aos pesquisadores que abriram o caminho teórico que me possibilitou trilhar e desenvolver este trabalho.

*“Não a nós, Senhor, nenhuma glória para nós,
mas sim ao teu nome,
por teu amor e por tua fidelidade!”
(Salmos 115:1)*

RESUMO

LOPES, T. L. C. **Nova classe de modelos paramétricos para análise de sistemas reparáveis**. 2022. 165 p. Tese (Doutorado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

A classe de modelos *Arithmetic Reduction of Age* (ARA) de [Doyen e Gaudoin \(2004\)](#) tem sido amplamente usada em análise de sistemas reparáveis, cujo o efeito do reparo é expresso por uma redução aritmética da idade. No entanto, a classe pressupõe estado de degradação do sistema, essa condição implica em $\beta > 1$ no Processo de Lei de Potência (PLP). Porém, existem casos em que o sistema melhora (PLP com $\beta < 1$) até certo tempo. Já sua intensidade após os reparos permanece paralela à intensidade inicial, conseqüentemente, deixa de capturar outras formas de intensidade de falha. Diante dessas limitações, propomos o modelo ARA_1 modificado ($ARAM_1$), que possibilita modelar sistemas em processo de renovação ou degradação e também propomos um novo processo PLP generalizado (PLPG), baseado em pontos de mudança. A partir do PLPG é possível derivar os principais modelos com pontos de mudança e com reparo imperfeito. Novos modelos são propostos a partir do PLPG, que denominamos de reparos completamente imperfeitos (RCI) e reparos parcialmente imperfeitos $[RPI(p)]$. Outra vantagem dessa abordagem, é que permite que a intensidade após os reparos não permaneça paralela à intensidade inicial, ampliando suas aplicações no mundo real. Por fim, propomos uma nova reparametrização do PLP com truncamento por tempo para incorporá-lo aos novos modelos e assim obter uma melhor interpretação dos seus parâmetros. Os estimadores do modelo proposto foram obtidos usando o método da máxima verossimilhanças. Avaliamos a performance dos estimadores dos parâmetros através de simulações de Monte Carlo (MC). Para fins de ilustração consideramos os tempos de falha reais nas aplicações. Os modelos propostos indicaram superioridade a outros modelos da literatura, exemplificando a importância das novas abordagens.

Palavras-chave: Pontos de mudança, Processo de lei de potência, Processo Poisson não homogêneo, Reparo imperfeito.

ABSTRACT

LOPES, T. L. C. **New class of parametric models for analysis of repairable systems**. 2022. 165 p. Tese (Doutorado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

The *Arithmetic Reduction of Age* (ARA) model class from [Doyen e Gaudoin \(2004\)](#) has been widely used in the analysis of repairable systems, whose repair effect is expressed by an arithmetic age reduction. However, the class presupposes a state of degradation of the system, this condition implies $\beta > 1$ in the Power Law Process (PLP). However, there are cases in which the system improves (PLP with $\beta < 1$) up to a certain time. Its intensity after repairs remains parallel to the initial intensity, consequently, it fails to capture other forms of failure intensity. Given these limitations, we propose the modified ARA₁ model (ARAM₁), which makes it possible to model systems in the process of renovation or degradation, and we also propose a new generalized PLP process (PLPG), based on change points. From the PLPG it is possible to derive the main models with change points and with imperfect repair. New models are proposed from the PLPG, which we call completely imperfect repairs (RCI) and partially imperfect repairs (RPI(p)). Another advantage of this approach is that it allows the intensity after repairs not to remain parallel to the initial intensity, expanding its applications in the real world. Finally, we propose a new PLP reparameterization with time truncation to incorporate it into new models and thus obtain a better interpretation of its parameters. The estimators of the proposed model were obtained using the maximum likelihood method. We evaluated the performance of the parameter estimators through Monte Carlo (MC) simulations. For illustration purposes we consider actual failure times in applications. The proposed models indicated superiority to other models in the literature, which illustrates the importance of the new approaches.

Keywords: Change Points, Power Law Process, Non-homogeneous Poisson process, Imperfect Repair.

LISTA DE ILUSTRAÇÕES

Figura 1 – Ilustração de uma variável aleatória $N(t)$ de um processo de contagem.	32
Figura 2 – Função de intensidade do PLP para $\eta = 1$ e $\beta = 0,8, 1$ e $1,3$	35
Figura 3 – Exemplo de função de intensidade $\lambda_{RM}(t) = 0,4743 \times t^{0,5}$	36
Figura 4 – Exemplo de função de intensidade $\lambda_{RP}(t)$	38
Figura 5 – Exemplo de função de intensidade $\lambda_{RI}(t)$	39
Figura 6 – Exemplo de função de intensidade $\lambda_{ARA_1}(t H_{t-})$, com $\beta = 1,1, \theta = 0,8,$ $\mu_T = 4$ e $T = 30$	41
Figura 7 – Parametrização (P3) $\lambda(t) = (\mu_T/T^\beta)\beta t^{\beta-1}$ com $\beta = 1,5$ e 2	49
Figura 8 – Parametrização $\lambda(t) = (\mu_T/T)\beta t^{\beta-1}$ com $\beta = 1,5$ e 2	49
Figura 9 – Função de intensidade do ARAM ₁ para diferentes valores de θ com 1) $t > T_{N_t-} = 20, \mu_T = 80, \beta = 1,4$ e $T = 100$ e 2) $t > T_{N_t-} = 6, \mu_T = 80,$ $\beta = 0,9$ e $T = 15$	52
Figura 10 – Função de intensidade ARAM ₁ para diferentes tempos de reparo, em dias, com 1) $\mu_T = 80, \beta = 1,1$ e $T = 200$ e 2) $\mu_T = 80, \beta = 0,6$ e $T = 80$	53
Figura 11 – Simulação de $\lambda_{ARAM_1}(t H_{t-})$ para diferentes valores de β, θ e μ_T com $T = 60$	55
Figura 12 – Simulação de $\lambda_{ARAM_1}(t H_{t-})$ para diferentes valores de β, μ_T e θ com $T = 60$	55
Figura 13 – Cenário 1: resultado do viés e MSE da simulação para o modelo ARAM ₁ com $T = 500$	61
Figura 14 – Cenário 2: resultado do viés e MSE da simulação para o modelo ARAM ₁ com $T = 50$	61
Figura 15 – Total de falhas acumuladas observadas nas nove colheitadeiras até $T = 200$	63
Figura 16 – Dados Reais (DR) de acúmulo de falhas pela expectativa de falhas modelada (MA) pelo ARAM ₁ para as nove colheitadeiras até $T = 200$	65
Figura 17 – Intensidade ARAM ₁ estimada para as nove colheitadeiras até $T = 200$	65
Figura 18 – Função de intensidade do PLPG para os três cenários propostos.	70
Figura 19 – Gráficos de $\lambda_{PLPG}(t)$ como função de intensidade (HPP) com p pontos de mudança.	81
Figura 20 – $\lambda_{PLPG}(t)$ como função de intensidade de um Processo Poisson Não Homogê- neo (NHPP) com p pontos de mudança.	82
Figura 21 – Gráficos de $\lambda_{PLPG}(t)$ com um ponto de mudança ($p = 1$).	83
Figura 22 – Gráficos de $\lambda_{PLPG}(t)$ com dois pontos de mudança ($p = 2$).	83
Figura 23 – Cenário 1: resultado do viés e MSE da simulação para PLPG com $\tau_1 = 15$ e $T = 30$	91

Figura 24 – Cenário 2: resultado do viés e MSE da simulação para PLPG com $\tau_1 = 15$ e $T = 30$	91
Figura 25 – Total de falhas acumuladas de um exaustor para valor.	93
Figura 26 – Gráficos do 1) número de falhas observadas (DR) versus o número esperado de falhas (MA) do modelo RPI(p) ajustado e 2) função de intensidade estimada do modelo RPI(p).	95
Figura 27 – Total de falhas acumuladas no sistema de ar-condicionado do Boeing 720 n° 7908.	96
Figura 28 – Gráficos do 1) número de falhas observadas (DR) versus o número esperado de falhas (MA) do modelo RPI(p) ajustado e 2) função de intensidade estimada do modelo RPI(p).	98

LISTA DE TABELAS

Tabela 1 – Resultado da simulação para os dois cenários propostos, caso composto por múltiplos sistemas.	60
Tabela 2 – Estimativas dos parâmetros do modelo ARAM ₁ na situação multiparamétrica, arredondadas em duas casas decimais, com $k = 9$	63
Tabela 3 – Estimativas dos parâmetros do modelo ARAM ₁ na situação composta, arredondadas em duas casas decimais.	64
Tabela 4 – Resumo dos casos particulares do PLPG.	80
Tabela 5 – Resultado da simulação para os dois cenários propostos, caso composto por múltiplos sistemas.	91
Tabela 6 – Conjunto de tempos de falha de um exaustor para vapor em dias.	92
Tabela 7 – Resultado do $\log \hat{L}$ e AIC dos modelos utilizados.	94
Tabela 8 – Parâmetros do modelo RPI(p) usado para modelar os tempos de falha do exaustor para vapor.	94
Tabela 9 – Conjunto de tempos de falha no sistema de ar-condicionado do Boeing 720 n° 7908.	95
Tabela 10 – Resultado do $\log \hat{L}$ e AIC dos modelos utilizados para os tempos de falha no sistema de ar-condicionado do Boeing 720 n° 7908,	97
Tabela 11 – Parâmetros do modelo RPI(p) usado para modelar os tempos de falha no sistema de ar-condicionado do Boeing 720 n° 7908.	97

LISTA DE ABREVIATURAS E SIGLAS

ABAO	Tão ruim quanto velho, do inglês <i>as bad as old</i>
AIC	Critério de Informação de Akaike, do inglês <i>Akaike Information Criterion</i>
ARA	Redução Aritmética de Idade, do inglês <i>Arithmetic Reduction of Age</i>
ARAM	ARA modificado
ARI	Redução Aritmética de Intensidade, do inglês <i>Arithmetic Reduction of Intensity</i>
CI	Intervalo de confiança, do inglês <i>Confidence interval</i>
DR	Dados reais
GRA	Redução Geométrica da Intensidade, do inglês <i>Geometric Reduction of Age</i>
GRI	Redução Geométrica de Intensidade, do inglês <i>Geometric Reduction of Intensity</i>
HPP	Processo Poisson Homogêneo, do inglês <i>Homogeneous Poisson Process</i>
MA	Modelo ajustado
MC	Monte Carlo
MCP	Múltiplos pontos de mudança, do inglês <i>Multiple Change Points</i>
MSE	Erro quadrático médio, do inglês <i>Mean Squared Error</i>
NHPP	Processo Poisson Não Homogêneo, do inglês <i>Non-Homogeneous Poisson Process</i>
PLP	Processo de Lei de Potência, do inglês <i>Power-Law Process</i>
PLPCP	Processo de Lei de Potência com ponto de mudança, do inglês <i>Power-Law Process with Change Point</i>
PLPG	Processo de Lei de Potência Generalizado
RCI	Reparo Completamente Imperfeito
RI	Reparo imperfeito
RM	Reparo mínimo
RP	Reparo perfeito
RPI	Reparo Parcialmente Imperfeito
SCP	Um ponto de mudança, do inglês <i>Single Change Points</i>
SE	Erro padrão, do inglês <i>Standard error</i>
XARA	Redução aritmética estendida da idade, do inglês <i>Extended arithmetic reduction of age</i>

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Revisão Bibliográfica	23
1.2	Contribuição da Tese	27
1.3	Objetivos	28
1.4	Organização da Tese	29
2	FUNDAMENTAÇÃO TEÓRICA	31
2.1	Processo de contagem	31
2.2	Processo de Poisson	32
2.3	Modelos de Sistemas Reparáveis	35
2.3.1	<i>Modelos de Reparo Mínimo (RM)</i>	36
2.3.2	<i>Modelos de Reparo Perfeito (RP)</i>	37
2.3.3	<i>Modelos de Reparo imperfeito (RI)</i>	38
2.4	Classe de Modelos ARA e ARI	39
2.4.1	<i>Classe ARA</i>	39
2.4.2	<i>Classe ARI</i>	41
2.4.3	<i>Modelos GRA_1 e GRI_1</i>	42
2.4.4	<i>Modelos XARA</i>	43
2.4.5	<i>Modelos com pontos de mudança</i>	44
3	ARA ₁ MODIFICADO COM PLP PARA MÚLTIPLOS SISTEMAS REPARÁVEIS	45
3.1	Introdução	45
3.2	Modelo ARA ₁ considerando (P3) e limitações	47
3.3	Modelo ARAM ₁ considerando parametrização (P4)	50
3.4	Método de Estimação	56
3.5	Estudo de simulação	58
3.6	Aplicação a dados reais	61
3.7	Considerações finais	65
4	PROCESSO LEI DE POTÊNCIA GENERALIZADO PARA SISTEMAS REPARÁVEIS	67
4.1	Introdução	67
4.2	O PLP generalizado (PLPG)	68

4.2.1	<i>Interpretação dos parâmetros do modelo PLPG</i>	71
4.2.2	<i>Casos Particulares</i>	75
4.3	Modelo PLPG para sistemas reparáveis	76
4.3.1	<i>Novos modelos para reparos imperfeitos com PLPG</i>	77
4.3.2	<i>Comportamento gráfico da função $\lambda_{PLPG}(t)$</i>	80
4.4	Estimação dos parâmetros	84
4.4.1	<i>Pontos de mudança conhecidos</i>	84
4.4.2	<i>Pontos de mudança observáveis</i>	86
4.5	Estudo de Simulação	89
4.6	Aplicação	92
4.6.1	<i>Exaustor para vapor</i>	92
4.6.2	<i>Sistema de ar-condicionado do Boeing 720 nº 7908</i>	95
4.7	Considerações finais	98
5	CONCLUSÃO E PROPOSTAS FUTURAS	101
	REFERÊNCIAS	103
	APÊNDICE A PRINCIPAIS CÓDIGOS IMPLEMENTADOS NO R	107
A.1	ARAM ₁	107
A.1.1	<i>Pacotes e funções gerais</i>	107
A.1.2	<i>Funções de intensidade e intensidade acumulada</i>	108
A.1.3	<i>simulação de tempos de falha</i>	109
A.1.4	<i>Função para aplicação</i>	111
A.1.5	<i>Estudo de Simulação</i>	119
A.1.6	<i>Aplicação em dados reais</i>	126
A.2	PLPG	147
A.2.1	<i>Pacotes e funções gerais</i>	147
A.2.2	<i>Funções de intensidade e intensidade acumulada</i>	148
A.2.3	<i>Simulação de tempos de falha</i>	150
A.2.4	<i>Função para aplicação</i>	152
A.2.5	<i>Estudo de Simulação</i>	158
A.2.6	<i>Aplicação em dados reais</i>	160

INTRODUÇÃO

1.1 Revisão Bibliográfica

Diante da concorrência intensa, com um mercado com clientes cada vez mais exigentes, empresas buscam oferecer sistemas¹ mais duráveis, capazes de manter seu desempenho por mais tempo com custo reduzido, ou seja, buscam oferecer sistemas mais confiáveis, equacionando o custo de produção. [Pham \(2006\)](#) exemplifica a importância da confiabilidade em equipamentos de monitoramento de pacientes que sofrem com insuficiência cardíaca, aplicações em controle de tráfego, reatores nucleares, em aeronaves, em segurança automotiva etc. Uma das primeiras referências no estudo da teoria da confiabilidade, [Barlow e Proschan \(1996\)](#), colocam que a análise de confiabilidade surgiu com as demandas da tecnologia moderna e particularmente das experiências na Segunda Guerra Mundial com sistemas militares complexos. Para os autores, análise de confiabilidade consiste em um conjunto de métodos matemáticos direcionados para a solução de problemas que envolvem a probabilidade de um sistema funcionar apropriadamente por um determinado período de tempo. Mesmo após a ocorrência de falhas, a análise de confiabilidade do sistema pode ser retomada, em uma área de confiabilidade denominada de análise de sistemas reparáveis. Nesse contexto, pesquisas em análise de sistemas reparáveis vem buscando compreender o comportamento dos tempos de falhas de sistemas com o desenvolvimento de modelos minimamente capazes de responder sobre estimativa da taxa de envelhecimento do sistema, assim como os efeitos dos reparos realizados. Conforme [Ascher e Feingold \(1984\)](#), um sistema é dito reparável se, após uma falha, sua função requerida puder ser satisfatoriamente retomada através de reparo sem a necessidade de substituir todos os componentes do sistema.

O Processo de Lei de Potência (PLP) é um modelo utilizado frequentemente na modelagem de tempos de falha de sistemas reparáveis, proposto por [Crow \(1974\)](#), também conhecido na literatura como processo Weibull. O PLP é um famoso caso particular do Processo Poisson Não

¹ Definido como conjunto de componentes interconectados que desempenham determinadas funções, como máquinas, equipamentos industriais, eletrônicos, etc.

Homogêneo (NHPP) , que ganhou destaque pela flexibilidade de sua função de intensidade e seus parâmetros terem boas interpretações, tornando-o relevante na literatura em modelagem de sistemas reparáveis. Nos últimos cinco anos, o PLP vem sendo explorado em diversos aspectos por dezenas de artigos científicos. Podemos destacar, por exemplo, os mais recentes. Wang, Lu e Xiao (2018) propuseram um método de bootstrap paramétrico para construir intervalos de confiança altamente precisos para os parâmetros do PLP. No trabalho de Louzada *et al.* (2019) desenvolvem um novo modelo de PLP sob riscos competitivos. Almeida *et al.* (2020) propuseram modelagem para múltiplos sistemas reparáveis sob a presença de riscos competitivos dependentes em que a intensidades de falha é um PLP com fator de fragilidade. Zheng *et al.* (2020) utilizam o PLP para descrever o processo de falha de turbinas eólicas. D'Andrea *et al.* (2021) utilizam abordagem Bayesiana para obter prioris objetivas para os parâmetros do PLP de múltiplos sistemas reparáveis idênticos. Já Syamsundar, Naikan e Wu (2021) introduzem novos modelos para sistemas reparáveis, com redução aritmética estendida da idade (XARA) e o PLP conta como uma das bases para desenvolvimento de casos particulares e principalmente em aplicações. Mais recentemente, Al-Douri, El-Halwagi e Groth (2022) aplicam o PLP no estudo de tempos de falhas em sistemas de produção de propano, e teste para o parâmetro de escala do PLP com dados incompletos de tempo de falha é encontrado no estudo de Chumnaul e Sepehrifar (2022).

Uma suposição para uso da função de intensidade do PLP é que o sistema deve voltar ao estágio imediatamente anterior ao reparo, ou seja, tão ruim quanto velho (ABAO - *as bad as old*). Esse tipo de reparo é conhecido na literatura como Reparo Mínimo (RM). Claramente, essa suposição não é válida para todas as situações, podem existir casos em que os sistemas se tornaram melhores do que antes da falha (ou até tão bom quanto um novo) ou pioraram a partir do reparo e, portanto, não é adequado o uso do PLP para modelar os tempos de falhas desses sistemas. Evidenciando uma limitação do PLP. Muitos modelos foram propostos objetivando capturar esse efeito do reparo na função de intensidade de falha. Brown e Proschan (1983) introduziram o conceito de reparo imperfeito (RI), em que o sistema fica tão bom quanto um novo com probabilidade p e tão ruim quanto antes com probabilidade $1 - p$. Kijima, Morimura e Suzuki (1988) propuseram o processo de renovação generalizado por meio do mecanismo da idade virtual em sistemas em estado de degradação, que é uma idade, não real, que o sistema assume a partir da alteração da idade cronológica (real) provocada por reparos, a cada reparo é calculada uma nova idade virtual com base na nova idade cronológica, isso provoca efeitos na diminuição (idade virtual menor que a idade real) ou aumento (idade virtual maior que a idade real) da função de intensidade de falha. O modelo de Kijima, Morimura e Suzuki (1988) tem como caso especial o modelo de Brown-Proschan. Conforme observa Lindqvist (2006), nesses modelos com idade virtual de Kijima, Morimura e Suzuki (1988), o tipo de reparo (mínimo ou perfeito) deve ser relatado para cada ação de manutenção corretiva. Em aplicações reais, no entanto, informações sobre o tipo de reparo raramente estão disponíveis.

Um importante trabalho publicado por Doyen e Gaudoin (2004), duas novas classes

de modelos de reparo imperfeito são propostas, ARA - *Arithmetic Reduction of Age* e ARI - *Arithmetic Reduction of Intensity*, para os quais o efeito do reparo é caracterizado pelo parâmetro ρ que induz mudança na intensidade da falha, de forma que: (i) se $\rho < 0$ o reparo é prejudicial; (ii) se $\rho = 0$ o reparo é mínimo; (iii) se $0 < \rho < 1$ o reparo é eficiente; e (iv) se $\rho = 1$ o reparo é ótimo. Diferentemente do modelo de Kijima, Morimura e Suzuki (1988) que é necessário observar o tipo de reparo, no trabalho de Doyen e Gaudoin (2004) o tipo de reparo é parametrizado. Neste trabalho não há proposta explícita de extensão do PLP para se adequar aos efeitos provocados pelos reparos. Contudo, indiretamente, geram novas extensões. Por exemplo, no modelo ARA com memória um (ARA₁), propuseram uma redução da idade cronológica com base no último tempo de falha como consequência do efeito do reparo, essa redução pode ser interpretada como um truncamento do PLP provocado por tal efeito.

Os modelos ARA têm sido amplamente utilizados, como é o caso dos trabalhos de Pan e Rigdon (2009) que estudaram métodos Bayesianos para o modelo ARA₁ com PLP, de Toledo *et al.* (2016) que propuseram uma política ótima de manutenção periódica em sistemas sob reparo imperfeito modelando com a classe ARA, de Said e Taghipour (2016) que aplicam o modelo em sistemas submetidos a manutenções corretivas e preventivas de diversos tipos e de Doyen, Gaudoin e Syamsundar (2017) que, como base na classe ARA, propuseram novos modelos baseados em redução geométrica da idade e intensidade de falha. Os trabalhos evidenciam bons ajustes para muitos conjuntos de dados reais de manutenção. Contudo, as classes ARA e ARI de Doyen e Gaudoin (2004) não abrangem todas as possibilidades do comportamento da função de intensidade de falha dos sistemas, uma vez que as funções são encaixáveis, no horizonte para o ARA e verticalmente para o ARI e limitando-se a aplicações em sistemas que se deteriorando ao longo do tempo. Finkelstein e Cha (2021) observam que uma das desvantagens do modelo ARA é que a intensidade após os reparos permanece paralela à intensidade inicial, o que pode limitar suas aplicações no mundo real. Eles sugeriram que o modelo de redução da intensidade da falha onde a taxa de falha é diminuída proporcionalmente pode ser uma alternativa melhor.

Nesse sentido, recentemente pesquisadores desenvolveram diversos modelos de processo de falha com funções de intensidade não encaixáveis. Doyen, Gaudoin e Syamsundar (2017) propuseram os modelos de Redução Geométrica da Idade (GRA) ou Redução Geométrica da Intensidade (GRI), que utilizam uma redução geométrica de idade ou intensidade, abrindo a possibilidade de considerar comportamentos no processo de falha que não podem ser abordados pelos modelos ARA e ARI, como uma desaceleração mais forte do desgaste. Doyen e Gaudoin (2021) concordam com Finkelstein e Cha (2021) sobre as limitações do modelo ARA, mas afirmam que o modelo tem uma boa interpretação de rejuvenescimento, embora não real, mas útil e fácil de interpretar, conceituar e entender em campo. Eles também afirmam que, embora os modelos geométricos possam superar a desvantagem acima, eles podem ter outras desvantagens e podem não ser modelos melhores do que o modelo ARA. Já Syamsundar, Naikan e Wu (2021) inovam com a introdução de um fator de reparo geométrico, que acelera ou diminui a idade virtual ou a intensidade formando o que os autores denominam modelos de redução aritmética estendida

da idade (XARA). Esses modelos possuem quatro parâmetros e combinam as características dos modelos de reparo imperfeito linear e geométrico para superar as desvantagens desses modelos, isso permite que os modelos XARA sejam capazes de acomodar outras naturezas de intensidades de falha. Uma desvantagem dos modelos ARA/ARI, GRA/GRI e XARA é que em todos os tempos de falha há efeito do reparo (negativo, positivo ou nulo) sobre a função de intensidade. Não consideram que somente em alguns tempos houve mudanças na função de intensidade provocada pelo reparo.

Tratamos até aqui sobre o efeito provocado pelo reparo sobre a função de intensidade de falha de um sistema reparável, que é uma mudança provocada diretamente no sistema e os tempos em que esses eventos ocorrem são observados. Além deste efeito, a função de intensidade de um sistema reparável também pode ser impactada por outras mudanças, por exemplos, mudanças climáticas e ambientais, cujo tempo da mudança pode ser desconhecido ou conhecido, modificando o comportamento do processo e segmentando sua função de intensidade de falha. Na literatura, essa alteração do processo a partir de determinado ponto é denominado ponto de mudança (*Change-point*), parametrizado por τ (conhecido ou desconhecido), que, quando desconhecido, é frequentemente tratado um parâmetro especial no processo de estimação. Em sua revisão sobre métodos de estimação com ponto de mudança, Zacks (1982) a descreve como o ponto em que há alteração nas leis de distribuição das variáveis aleatórias envolvidas no processo. Processos com essa condição são bastantes explorados na literatura em diversas áreas, uma vez que é natural que processos mudem de parametrização por conta de perturbações no processo.

Matthews e Farewell (1982) consideram um modelo com um ponto de mudança dividindo a taxa de risco em duas taxas constantes e aplicando a dados reais de tratamento da leucemia. Yao (1986) apresenta o estimador de máxima verossimilhança para τ com restrição e Loader (1991) propõe inferência para o ponto de mudança, derivando o teste de razão de verossimilhança e obtendo as regiões de confiança aproximadas para o parâmetro τ , ambos trabalhos consideram a modelagem dada por Matthews e Farewell (1982). Já Zacks (1984), apresenta uma nova família de distribuição denominada Exponential-Weibull com um ponto de mudança, derivando suas principais propriedades e desenvolveram métodos para estimar τ . A taxa de risco da Exponential-Weibull é constante até o ponto de mudança e, após, não crescente baseada na função de intensidade do PLP mais uma constante. Não há salto/queda da taxa de risco, ou seja, não há ponto de descontinuidade em τ na taxa de risco. Modelos que possibilitam pontos de descontinuidade na taxa de risco em τ promovem mais generalização aos modelos, dado que possibilitam mais diversidade na modelagem. Boukai (1987) apresenta uma reparametrização do modelo Exponential-Weibull, em que a taxa de risco tem a estrutura funcional da intensidade do PLP, sem adição de uma constante, até o ponto de mudança e, após, constante. Também sem salto/queda na taxa de risco provocada pela mudança.

Um NHPP antes e depois do ponto de mudança, denominado Log-Linear, foi proposto por Loader (1992), que tem como caso particular o modelo de Matthews e Farewell (1982) e o

tradicional processo Log-Linear. Nessa proposta, há introdução de um parâmetro de salto/queda dado por δ , que pode ser interpretado como impacto da mudança na descontinuidade da função de intensidade em τ , contudo não estudou o PLP nessa condição. Richardson e Basu (2004) apresentam um novo modelo que é uma extensão do modelo PLP, denominado modelo de ponto de mudança do processo de lei de potência (PLPCP). O modelo descreve os tempos de falha de tipos particulares de sistemas reparáveis que experimentam uma mudança única em sua função de intensidade de falha. PLPCP é similar ao modelo proposto por Boukai (1987), isto é, PLP até τ e constante após o ponto de mudança, porém, no PLPCP consideram τ conhecido e permite salto/queda da função de intensidade em τ . Nessa mesma linha, Syamsundar e Naikan (2008) propuseram uma extensão do PLPCP, com múltiplos pontos de mudanças e as funções de intensidade podem assumir várias formas, como caso particular, os autores usam o PLP para cada segmentos de tempo formado entre os pontos de mudança, também obtêm as estimativas dos pontos de mudanças τ_i , em que i é o i -ésimo ponto de mudança. Nesta modelagem, os pontos de mudança rejuvenescem a intensidade de falha, já que a mesma é nula em τ_i . Os modelos propostos pelos autores são aplicados a diversos dados obtidos em um ambiente industrial. Contudo, limitam-se ao efeito perfeito do ponto de mudança, não há parametrização para quantificar o efeito da mudança.

Os efeitos do reparo e do ponto de mudança sobre a função de intensidade de um sistema estão conectados, uma vez que um reparo pode gerar mudança no processo cujo impacto tem efeito de redução do número médio de falhas ou no aumento médio delas, contudo os modelos atuais tratam essas situações em processos distintos no PLP, que é o processo motivacional deste trabalho. Nas propostas que parametrizam o efeito do reparo na modelagem, ou seja, os modelos de reparo imperfeito, e usam como base o PLP na função de intensidade de falha, todos tempos de falhas impactam no processo, como no trabalho de Doyen e Gaudoin (2004), não consideram a possibilidade de que algumas ou apenas uma falha modifiquem o processo, como acontece nos processos com ponto de mudança. Por outro lado, na literatura relacionada a pontos de mudanças, é muito comum propostas de modelos com um ponto de mudança e nos múltiplos pontos de mudança há sempre uma queda na função de intensidade, como no trabalho de Syamsundar e Naikan (2008), assim não há proposta na literatura para quantificar esse efeito da mudança sobre a função de intensidade como acontece em modelos de reparo imperfeito.

1.2 Contribuição da Tese

Diante da revisão bibliográfica apresentada e apesar do avanço em análise de sistemas reparáveis nos últimos anos, ainda existem lacunas. Como relatado, o PLP limita-se a modelagem de sistemas de reparo mínimo, não há explicitamente um PLP mais geral, cuja a parametrização permita modelar outras situações de reparo, como reparo imperfeito. Existem modificações do PLP, mas para tratar situações específicas como é o caso do ARA, amplamente utilizado, que há redução da idade cronológica para obtenção de uma situação intermediária, entre reparo mínimo

e perfeito. Outras limitações observadas, especificamente no modelo ARA, é que está definido para sistemas em deterioração, não permitindo, por tanto, capturar a informação do estado do sistema para aqueles que possivelmente estavam melhorando entre falhas até um certo tempo e também que a intensidade após os reparos permanece paralela à intensidade inicial, limitando suas aplicações no mundo real. Evidenciamos que uma desvantagem dos modelos de reparo imperfeito, como ARA/ARI, GRA/GRI e XARA, é que em todos os tempos de falha há efeito do reparo (negativo, positivo ou nulo) sobre a função de intensidade, não consideram na modelagem que somente alguns reparos impactam na função de intensidade de falha do sistema. Em contra partida, os modelos com pontos de mudança não parametrizam o efeito provocado por eles, pois é intuitivo imaginar situações cujo impacto da mudança seja intermediária, entre uma mudança perfeita e uma mudança mínima, como acontece nos modelos de reparo imperfeito.

Neste trabalho propomos novos modelos objetivando suprir essas lacunas apresentadas. Vamos apresentar uma generalização do modelo ARA_1 com PLP, em que adicionamos o caso em que o sistema reparável possui intensidade decrescente, ou seja, o caso em que o sistema está melhorando até um determinado tempo, denominamos essa nova modelagem de ARA_1 modificado ($ARAM_1$). Também propomos uma nova reparametrização do PLP com truncamento por tempo para incorporá-lo ao novo modelo e assim preservar a interpretação original dos parâmetros do PLP proposto por Crow (1974). Um novo processo PLP generalizado (PLPG) é proposto, baseado em pontos de mudança, que podem ser interpretados como parâmetros conhecidos, desconhecidos e como os tempos observados de falha. No PLPG é possível considerar o impacto de todos ou alguns tipos de reparo no processo ou também tratar os pontos de mudanças com saltos ou quedas e quantificar seu efeito sobre função de intensidade, lhe concedendo mais flexibilidade nas modelagens. A partir do PLPG é possível derivar os principais modelos com ponto de mudança, assim como os principais modelos de reparo imperfeito. São casos particulares do PLPG: PLP investigado por Crow (1974), HPP com um ponto de mudança postulado por West e Ogden (1997), PLPCP de Richardson e Basu (2004), o PLP com múltiplos pontos de mudança de Syamsundar e Naikan (2008), modelos para reparo mínimo, reparo perfeito, modelo ARA_1 proposto por Doyen e Gaudoin (2004), modelo ARA_1 modificado ($ARAM_1$). Novos modelos são propostos a partir do PLPG, que denominamos de reparos completamente imperfeitos (RCI) e reparos parcialmente imperfeitos [RPI(p)]. Logo, o PLPG é uma nova classe de modelos para sistemas reparáveis. Outra vantagem dessa abordagem, é que permite que a intensidade após os reparos não permaneça paralela à intensidade inicial, ampliando suas aplicações no mundo real.

1.3 Objetivos

O objetivo principal desta tese consiste em desenvolver modelos gerais para múltiplos sistemas reparáveis, baseados no PLP e na sua generalização. Dessa maneira, enumeramos os seguintes objetivos específicos:

- propor um novo modelo para múltiplos sistemas reparáveis a partir da modificação do ARA e baseados em uma nova parametrização do PLP;
- propor uma nova classe de modelos, denominado PLP generalizado (PLPG), baseada em pontos de mudança, em que são derivados os principais modelos com ponto de mudança, assim como os principais modelos de reparo imperfeito;
- desenvolver método de estimação dos parâmetros envolvidos nos modelos propostos e seleção dos pontos de mudança;
- avaliar as propriedades assintóticas dos estimadores sugeridos por meio de estudos de simulação;
- aplicar os novos modelos para avaliar a qualidade do ajuste frente aos principais modelos da literatura;
- implementar, na linguagem de programação R, os pontos supracitados para utilização em análise de dados de sistemas reparáveis.

1.4 Organização da Tese

O trabalho está dividido em cinco capítulos e organizado seguinte forma:

- No Capítulo 1, uma revisão bibliográfica foi apresentada, o problema e as motivações da pesquisa foram introduzidas, as novidades destacadas e as principais contribuições do trabalho apresentadas.
- Apresentamos, no Capítulo 2, a fundamentação teórica que serve de base para o desenvolvimento da nova classe proposta de modelos. Conceituamos análise de sistemas reparáveis e processo de contagem, descrevemos os tipos de processos Poisson e introduzimos os principais modelos para sistemas reparáveis disponíveis na literatura.
- No Capítulo 3 apresentamos o modelo ARA_1 modificado ($ARAM_1$) com PLP e o método de estimação de seus parâmetros. Em seguida, avaliamos a performance dos estimadores dos parâmetros através do estudo de simulação de Monte Carlo (MC). Também apresentamos uma aplicação do modelo $ARAM_1$ em dados reais de tempos de falha, em dias, de lâmina *Chopper* localizada no sistema de corte de cana-de-açúcar em nove colheitadeiras.
- Propomos, no Capítulo 4, um novo processo PLP, denominado de PLP generalizado (PLPG), baseado em pontos de mudança, assim como o método de estimação de seus parâmetros e seleção desses pontos. Também avaliamos a performance dos estimadores dos parâmetros através do estudo de simulação de MC. Apresentamos duas aplicações do PLPG em dados reais de tempos de falha para ilustrar a performance frente aos demais

modelos para sistemas reparáveis, o primeiro conjunto de tempos de falha utilizado para modelagem é o das falhas de um exaustor para vapor, com tempo de operação em número de dias e o segundo conjunto de tempos de falha utilizado para modelagem é o das falhas do sistema de ar-condicionado de um avião a jato Boeing 720 n° série 7908, com tempo de funcionamento em horas.

- O trabalho será concluído no Capítulo 5 com as considerações finais e propostas futuras.

FUNDAMENTAÇÃO TEÓRICA

Apresentaremos neste capítulo a fundamentação teórica, que servirá de base para o desenvolvimento dos novos modelos propostos.

2.1 Processo de contagem

Antes de definirmos Processo de contagem, vamos abordar um conceito mais geral de processo, denominado processo estocástico. [Ross \(1996\)](#) e [Rigdon e Basu \(2000\)](#) se aprofundam nos conceitos que serão apresentados a seguir.

Definição 1. Um processo estocástico $\mathbf{X} = \{X(t), t \in T\}$ é um conjunto de variáveis aleatórias. Para cada t no conjunto de índices T , $X(t)$ é uma variável aleatória.

Uma situação muito abordada é considerar o índice $t \in T$ como tempo transcorrido discreto ou contínuo, portanto, ordenado, e $X(t)$ uma variável no tempo t . Uma série temporal, por exemplo, é a realização de um processo estocástico. Um caso particular de processo estocástico é o processo de contagem, definido a seguir.

Definição 2. Um processo estocástico $\{N(t), t \geq 0\}$ é dito ser um processo de contagem, se $N(t)$ representar o número total de eventos que ocorreram até o tempo t . Um processo de contagem $N(t)$ deve satisfazer as seguintes condições:

- (i) $N(t) \geq 0$;
- (ii) $N(t)$ é um número inteiro;
- (iii) Se $s < t$, então $N(s) \leq N(t)$.
- (iv) Para $s < t$, $N(t) - N(s)$ é igual ao número de eventos ocorridos no intervalo $(s, t]$, que denotaremos por $N(s, t]$.

Na Figura 1, ilustramos graficamente uma variável $N(t)$ de um processo de contagem, em que os eventos de interesse ocorreram em t_1, t_2, t_3 e t_4 . Note que $N(t) \geq 0, N(t_1) < N(t_2) < N(t_3) < N(t_4)$ e, por exemplo, o número de eventos ocorridos no intervalo $(t_1, t_4]$ é $N(t_1, t_4] = 3$.

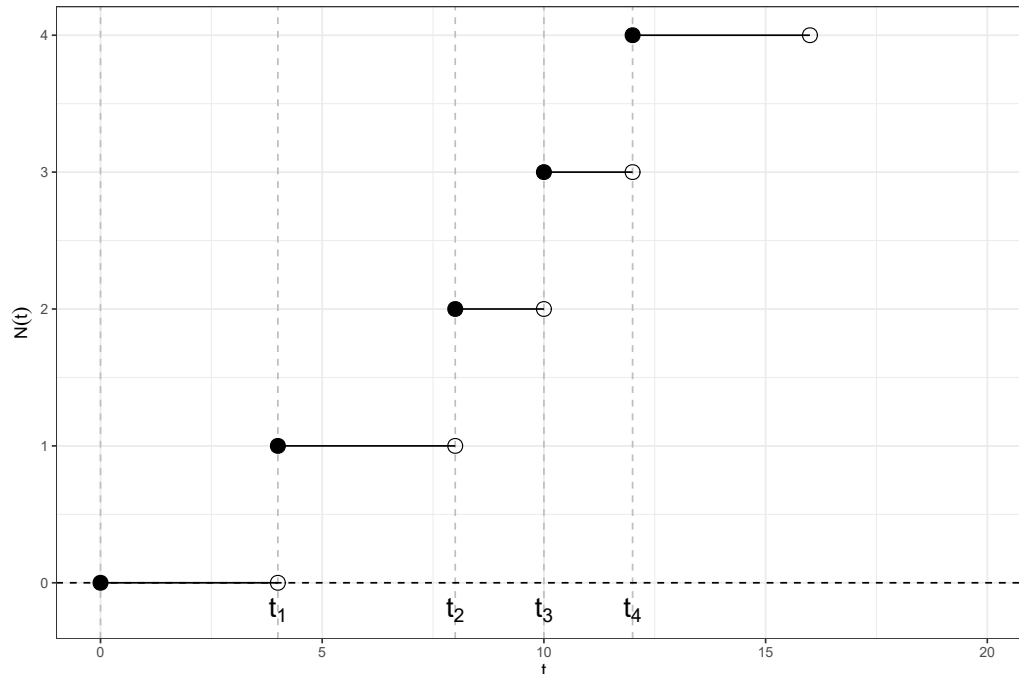


Figura 1 – Ilustração de uma variável aleatória $N(t)$ de um processo de contagem.

Definido processo de contagem, a função de intensidade $\lambda(t)$, Definição 4, é uma das funções mais importantes na modelagem de sistemas reparáveis. A função $\lambda(t)$ pode ser interpretado como taxa de falha instantânea em função do tempo t do processo. De acordo com o comportamento de $\lambda(t)$ é possível interpretar a situação do sistema, por exemplo, se $\lambda(t)$ resulta em uma função de intensidade estritamente crescente, então o sistema terá mais falhas com o passar do tempo, dado que taxa de falha instantânea está aumentando.

Definição 3. A função de intensidade de falha do processo de contagem é dada por

$$\lambda(t) = \lim_{\Delta t \rightarrow 0} \frac{P[N(t + \Delta t) - N(t) \geq 1 | H_{t-}]}{\Delta t},$$

em que H_{t-} é o histórico de todos os eventos ocorridos antes do tempo t .

Com isso, podemos definir o processo de Poisson, que é um caso particular do processo de contagem.

2.2 Processo de Poisson

Definição 4. Um processo de contagem $N(t)$ é dito Processo de Poisson se

- (i) $N(0) = 0$.
- (ii) Considere $a < b \leq c < d$, então as variáveis aleatórias $N(a, b]$ e $N(c, d]$ são independentes. Essa propriedade é chamada de incrementos independentes.
- (iii) Existe uma função λ , tal que

$$\lambda(t) = \lim_{\Delta t \rightarrow 0} \frac{P[N(t + \Delta t) - N(t) = 1]}{\Delta t},$$

A função λ é chamada de função de intensidade do processo de Poisson.

- (iv) Falhas simultâneas não ocorrem, ou seja,

$$\lim_{\Delta t \rightarrow 0} \frac{P[N(t + \Delta t) - N(t) \geq 2]}{\Delta t} = 0,$$

Como consequência da Definição 4, [Rigdon e Basu \(2000\)](#) demonstra o Teorema 1 partindo da função $P_n(t) = P[N(t) = n]$, utilizando limites e equações diferenciais. O Teorema 1 é particularmente elegante, dado que são utilizadas somente as condições de (i) a (iv) para concluir que o número de falhas em um intervalo de tempo é uma variável aleatória que segue uma distribuição Poisson.

Teorema 1. Se o processo de contagem $\{N(t), t \geq 0\}$ satisfizer as propriedades da Definição 4, então

$$P[N(t) = n] = \frac{1}{n!} \left(\int_0^t \lambda(x) dx \right)^n \exp \left(- \int_0^t \lambda(x) dx \right).$$

Para $n = 0, 1, 2, \dots$

O Teorema 1 nos garante que $N(t)$, satisfeitas as condições da Definição 4, terá distribuição Poisson, com média e variância iguais, dadas por

$$E[N(t)] = Var[N(t)] = \Lambda(t) = \int_0^t \lambda(x) dx,$$

em que $\Lambda(t)$ também é chamada, na literatura, de função de intensidade acumulada.

O Corolário 1 nos fornece, de posse de $\Lambda(t)$, a média de falhas do processo em um intervalo $(a, b]$. Os resultados apresentados no Teorema 1 e Corolário 1 serão o fundamento dos novos modelos que serão apresentados nos próximos capítulos. Para demonstração do Corolário 1 consultar o livro de [Rigdon e Basu \(2000\)](#).

Corolário 1. Para um processo de Poisson, a variável aleatória $N(a, b]$ tem distribuição Poisson com média

$$E[N(a, b)] = \int_a^b \lambda(x) dx = \Lambda(b) - \Lambda(a).$$

O processo de Poisson pode ser classificado em termos de função de intensidade de duas formas. Se a função de intensidade de um processo de Poisson for constante, isto é, $\lambda(t) = \lambda$, $t \geq 0$, dizemos que este é um Processo de Poisson Homogêneo (HPP) e os tempos entre as falhas são distribuídas de forma homogênea, entretanto, não é utilizado para modelar sistemas que estão se deteriorando ou melhorando com o tempo.

Por outro lado, se a função de intensidade $\lambda(t)$, $t \geq 0$, não for constante, dizemos que o processo é um Processo de Poisson Não Homogêneo (NHPP), neste caso, é possível modelar sistemas que estão se deteriorando ou melhorando.

Vamos evidenciar um caso particular de NHPP, amplamente utilizado na modelagem de sistemas reparáveis, que é o Processo de Lei de Potência (PLP), investigado inicialmente por [Crow \(1974\)](#). Um NHPP $\{N(t), t \geq 0\}$ é um PLP se sua função de intensidade e sua função de intensidade acumulada forem dadas, respectivamente, por

$$\lambda(t) = \alpha \beta t^{\beta-1} \text{ e } \Lambda(t) = \alpha t^{\beta}, \quad (2.1)$$

em que o parâmetro $\alpha > 0$ é de escala e $\beta > 0$ é o parâmetro de forma. O parâmetro α tem interpretação genérica, enquanto β representa a elasticidade do número esperado de falhas em relação ao tempo, então, se $\beta > 1$ significa que o sistema está deteriorando e se $\beta < 1$ o sistema está melhorando ao longo do tempo. Também é comum na literatura encontrar o PLP com a seguinte parametrização:

$$\lambda(t) = \frac{\beta}{\eta} \left(\frac{t}{\eta} \right)^{\beta-1} \text{ e } \Lambda(t) = \left(\frac{t}{\eta} \right)^{\beta}, \quad (2.2)$$

em que $\eta > 0$ é o parâmetro de escala, que é interpretado como o tempo durante o qual se espera exatamente uma falha, ou seja, $\Lambda(\eta) = 1$. Perceba que o PLP com a parametrização dada por (2.2) é uma reparametrização do dado em (2.1), com $\alpha = \eta^{-\beta}$.

A Figura 2 apresenta uma ilustração da função de intensidade do PLP com a parametrização (2.2). Para $\beta = 0,8$ temos $\lambda(t)$ decrescente, indicando um sistema que está melhorando após reparos, com $\beta = 1$ a intensidade de falha permanece constante, já para $\beta = 1,3$ a função $\lambda(t)$ cresce, indicando deterioração do sistema após reparos.

Os processos de contagem, em especial o processo de Poisson, são úteis na modelagem de tempos de falhas de sistemas reparáveis. Por exemplo, modelar a intensidade de falha de um processo de contagem de falhas de um equipamento ao longo do tempo, com tempo de reparo sendo ignorada. O processo de contagem Poisson com intensidade de falha dada por $\lambda(t)$ de um PLP, pode ser utilizado para este objetivo. Na próxima seção vamos tratar sobre os diferentes tipos de reparos e modelos para sistemas reparáveis.

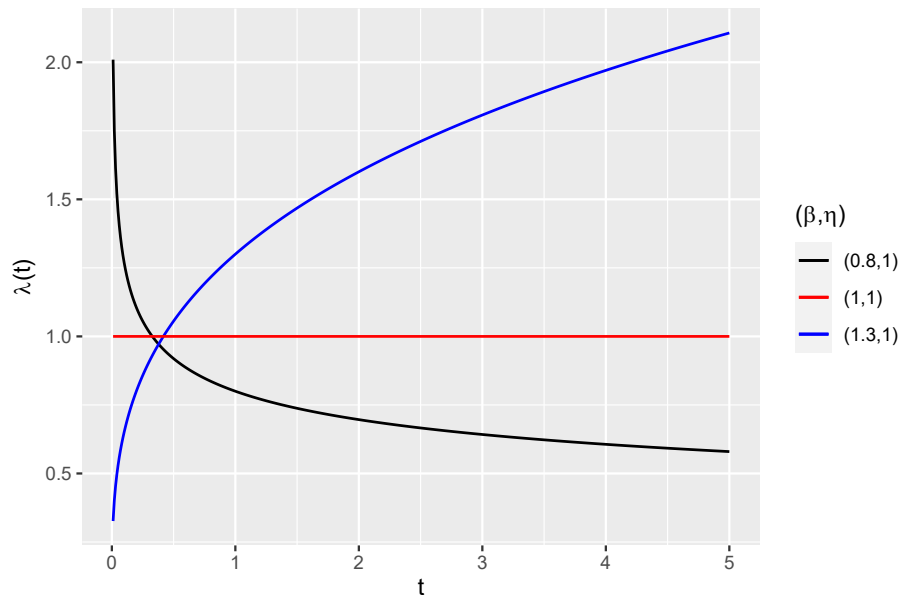


Figura 2 – Função de intensidade do PLP para $\eta = 1$ e $\beta = 0,8, 1$ e $1,3$.

2.3 Modelos de Sistemas Reparáveis

Conforme [Ascher e Feingold \(1984\)](#), um sistema é dito reparável se, após uma falha, sua função requerida puder ser satisfatoriamente retomada através de reparo sem a necessidade de substituir todos os componentes do sistema. Para modelar processos de falhas de um sistema reparável é necessário compreender as notações utilizadas na literatura, bem como diferenciar os tipos de reparos e os modelos disponíveis, assim como identificar suas vantagens e limitações.

Vamos iniciar pelo conceito de truncamento. Em sistemas reparáveis, os truncamentos põem fim às observações de falhas, que são de duas formas: (i) por tempo de truncamento, que limita o tempo de observação de falhas, ou (ii) por número de falhas observadas. Por exemplo, se o sistema for observado até um tempo predeterminado T , então o truncamento é por tempo, se for observado até um número n^* de falhas, então o truncamento é por falhas. No decorrer do trabalho vamos evidenciar mais o truncamento por tempo, visto que as novas modelagem serão baseadas nesse tipo de truncamento. Para as definições seguintes, consideremos um conjunto formado por k sistemas reparáveis, onde t_{ij} é o i -ésimo tempo de falha observado no j -ésimo sistema, com $i = 1, \dots, n_j$ e $j = 1, \dots, k$.

No truncamento por tempo, $0 < t_{1j} < \dots < t_{n_j j} < T$, em que $t_{n_j j}$ é o último tempo de falha observado do j -ésimo sistema. Note que, nesse caso, a quantidade n_j de falhas observadas do j -ésimo sistema é aleatória. No truncamento por falhas é definido um número máximo de falhas n_j a serem observadas, nesse caso apenas os tempos de falha são aleatórios. Para fins de modelagem, consideraremos que após a ocorrência da falha o sistema é reparado imediatamente e o tempo de reparo é insignificante. A seguir serão abordados os principais modelos para sistemas

reparáveis presentes na literatura: reparo mínimo (RM), perfeito (RP) e o imperfeito (RI).

2.3.1 Modelos de Reparo Mínimo (RM)

O RM é o tipo de reparo que deixa o sistema na mesma condição em que estava imediatamente anterior a falha, nem melhor e nem pior que antes. O RM é conhecido na literatura por *As Bad as Old* (ABAO), que pode ser traduzido livremente por tão ruim quanto antes da falha. O processo associado à ocorrência de falha em sistemas sob reparo mínimo pode ser um HPP ou, como é mais comum, por NHPP, em especial é adotado o PLP. Assim, adotando o PLP para modelar o processo de falhas de sistema sob reparo mínimo, então

$$\lambda_{RM}(t) = \alpha\beta t^{\beta-1}.$$

Para exemplificar, considere a seguinte função de intensidade $\lambda_{RM}(t) = 0,4743 \times t^{0,5}$ de um sistema em que são realizados reparos mínimos, ilustrando a situação de um sistema cuja a intensidade de falha é estritamente crescente, seguindo um PLP com $\alpha = 0,3162$ e $\beta = 1,5$. Percebemos na Figura 3 que as falhas vão ocorrendo sem pontos de descontinuidade em $\lambda_{RM}(t)$.

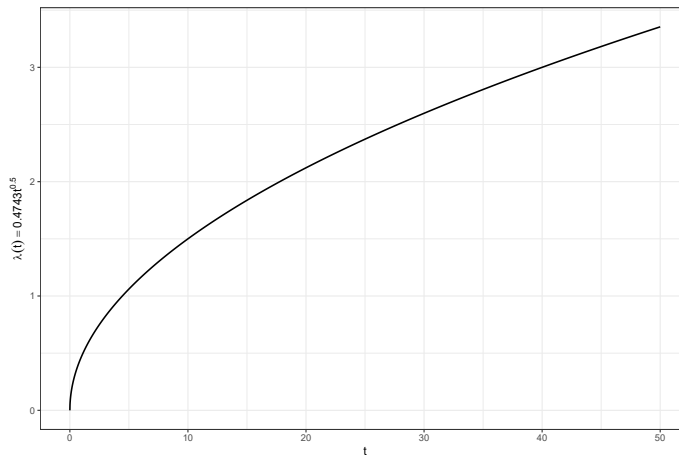


Figura 3 – Exemplo de função de intensidade $\lambda_{RM}(t) = 0,4743 \times t^{0,5}$.

Oliveira, Colosimo e Gilardoni (2012) propuseram uma parametrização do PLP em termos β , que é o parâmetro de elasticidade e μ_T como o número médio de falhas até um tempo de truncamento T , de modo que

$$\lambda_{RM}(t) = (\mu_T/T^\beta)\beta t^{\beta-1}, \mu_T > 0, \beta > 0, T > 0, t > 0,$$

dessa forma $E[N(T)] = \Lambda(T) = \mu_T$, desse modo, é bastante útil em aplicações que adotam truncamento por tempo, sendo uma parametrização com forte apelo interpretativo. Originalmente Oliveira, Colosimo e Gilardoni (2012) utilizam o parâmetro η em vez de μ_T .

A inferência estatística para o modelo de RM com PLP é geralmente baseada no estimador de máxima verossimilhança. Podemos escrever a função de verossimilhança, parametrizado de acordo com Oliveira, Colosimo e Gilardoni (2012), como

$$\begin{aligned} L_{RM}(\Theta|\mathbf{t}) &= \prod_{j=1}^k \left\{ \left(\prod_{i=1}^{n_j} \lambda_{RM}(t_{ij}) \right) e^{-\Lambda(T)} \right\} \\ &= \prod_{j=1}^k \left\{ \left(\prod_{i=1}^{n_j} (\mu_T/T^\beta) \beta t_{ij}^{\beta-1} \right) e^{-\mu_T} \right\}, \end{aligned}$$

em que $\Theta = (\mu_T, \beta)$ e $\mathbf{t} = (\mathbf{t}_1, \dots, \mathbf{t}_j, \dots, \mathbf{t}_k)$, com $\mathbf{t}_j = (t_{1j}, \dots, t_{n_j j})$.

2.3.2 Modelos de Reparo Perfeito (RP)

Nesta modelagem, é considerando que o sistema se torna tão bom quanto um novo, conhecido na literatura como *As Good As New* (AGAN). Não é coerente adotar o HPP para modelar sistemas com reparos perfeito, dado que após o reparo a função de intensidade seria nula se o mesmo fosse utilizado. Neste caso, o NHPP é o processo mais adequado para modelar o processo de falhas de sistemas com reparo perfeito, contudo uma suposição para uso da função de intensidade do PLP é que o sistema deve voltar ao estágio imediatamente anterior ao reparo, ou seja, tão ruim quanto velho (ABAO) e não é este o caso.

Nesta situação, adota-se um truncamento em t , de modo que o processo se inicia sempre a partir da última falha, dado que há impacto dos reparos sobre a função de intensidade de falha do PLP, que denominaremos de $\lambda_{RP}(t)$. Para ilustrar, considere uma função de intensidade dada por

$$\lambda_{RP}(t|t_1 = 10) = 0,4667 \times t^{0,4} \mathbf{I}(0 \leq t \leq t_1) + 0,4667 \times (t - t_1)^{0,4} \mathbf{I}(t > t_1).$$

de um sistema em que foi realizado um reparo perfeito em $t_1 = 10$, em que \mathbf{I} é uma função indicadora. A Figura 4 mostra o efeito do reparo perfeito sobre a função de intensidade, reduzindo-a ao ponto do sistema ficar tão bom quanto um novo. O ponto cheio representa a intensidade no instante da falha e, imediatamente após falha (após ponto aberto), inicia-se a nova intensidade. Fica claro que as falhas vão ocorrendo e interferindo em $\lambda_{RP}(t)$. Formalmente, a função de intensidade de falha do modelo de RP é dada por

$$\lambda_{RP}(t|H_{t-}) = \lambda(t - T_{N_{t-}}|H_{t-}),$$

em que $\lambda(\cdot)$ é a função de intensidade do NHPP até a primeira falha, N_{t-} é o número de falhas antes de t e $T_{N_{t-}}$ representa o tempo da última falha antes de t , e, portanto, $(t - T_{N_{t-}})$ é o tempo até t desde a última falha antes de t . Se considerarmos um PLP, podemos reescrever $\lambda_{RP}(t|H_{t-})$ por

$$\lambda_{RP}(t|H_{t-}) = (\mu_T/T^\beta) \beta (t - T_{N_{t-}})^{\beta-1}.$$

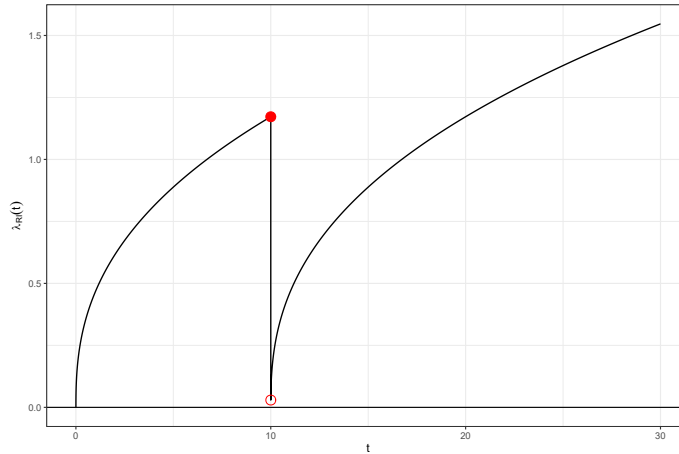


Figura 4 – Exemplo de função de intensidade $\lambda_{RP}(t)$.

Este processo NHPP com $\lambda_{RP}(t|H_{t-})$ também é dito ser renovável porque os intervalos entre tempos de ocorrência de falha, X_i , são independente e identicamente distribuídos (*i.i.d*), em que $X_i = T_i - T_{i-1}$ indica o tempo entre os eventos i e $i - 1$, com $T_0 = 0$ e $i = 1, 2, \dots$

Podemos escrever a função de verossimilhança para estimação dos parâmetros, parametrizado de acordo com [Oliveira, Colosimo e Gilardoni \(2012\)](#), como

$$\begin{aligned} L_{RP}(\Theta|\mathbf{t}) &= \prod_{j=i}^k \left\{ \left(\prod_{i=1}^{n_j} \lambda_{RP}(t_{ij}) \right) e^{-\Lambda^{(j)}(T)} \right\} \\ &= \prod_{j=i}^k \left\{ \left(\prod_{i=1}^{n_j} (\mu_T/T^\beta) \beta (t_{ij} - t_{(i-1)j})^{\beta-1} \right) e^{-\Lambda^{(j)}(T)} \right\}, \end{aligned}$$

em que $\Lambda^{(j)}(T) = \sum_{i=1}^{n_j+1} \int_{t_{(i-1)j}}^{t_{ij}} \lambda_{RP}(x|H_{x-}) dx$, com $t_{0j} = 0$ e $t_{(n_j+1)j} = T$. Perceba que se $t = t_{ij}$, então $T_{N_{t-}} = t_{(i-1)j}$. Logo, $\lambda_{RP}(t_{ij}|H_{t_{ij}-}) = (\mu_T/T^\beta) \beta (t_{ij} - t_{(i-1)j})^{\beta-1}$.

2.3.3 Modelos de Reparo imperfeito (RI)

A suposição de RM ou RP não são válidas para todas as situações, podem existir casos em que o sistema se torna melhor do que antes da falha mas não tão bom como um novo ou pode piorar após reparo e, portanto, não é adequado o uso do RM e RP. Para exemplificar o RI, considere uma função de intensidade dada por

$$\lambda_{RI}(t) = 0,4667 \times t^{0,4} \mathbf{I}(0 \leq t \leq 10) + 0,4667 \times (t - 8)^{0,4} \mathbf{I}(t > 10)$$

de um sistema em que foi realizado um reparo imperfeito em $t_1 = 10$. A Figura 5 mostra o efeito do reparo imperfeito sobre a função de intensidade, reduzindo-a no tempo em que houve o reparo, nesse caso o efeito foi intermediário, entre ABAO e AGAN.

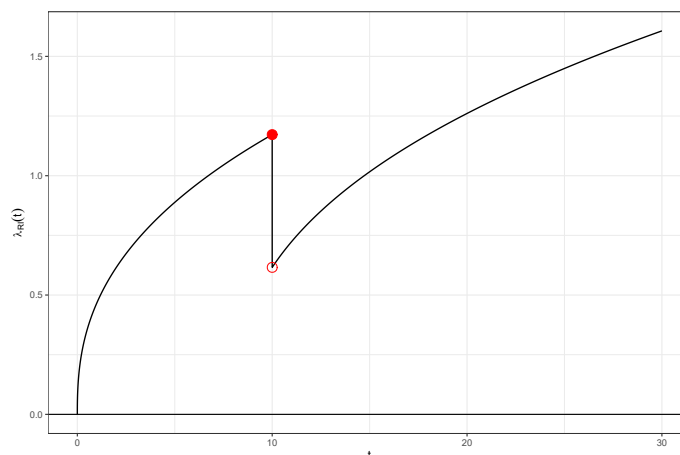


Figura 5 – Exemplo de função de intensidade $\lambda_{RI}(t)$.

Um importante trabalho publicado por [Doyen e Gaudoin \(2004\)](#), duas novas classes de modelos de reparo imperfeito são propostas, ARA - *Arithmetic Reduction of Age* e ARI - *Arithmetic Reduction of Intensity*, para os quais o efeito de reparo é caracterizado pelo parâmetro θ que induz mudança na intensidade da falha, de forma que: (i) se $\theta < 0$ o reparo é prejudicial; (ii) se $\theta = 0$ o reparo é mínimo; (iii) se $0 < \theta < 1$ o reparo é eficiente; e (iv) se $\theta = 1$ o reparo é ótimo. A vantagem é que a classe ARA e ARI podem ser adotadas para modelar sistemas de reparo mínimo e perfeito.

Diferentemente do modelo proposto por [Kijima, Morimura e Suzuki \(1988\)](#), que é necessário observar o tipo de reparo, entretanto, o tipo de reparo (mínimo ou perfeito) deve ser relatado para cada ação de reparo. Em aplicações reais, no entanto, informações sobre o tipo raramente estão disponíveis, conforme observa [Lindqvist \(2006\)](#). No trabalho de [Doyen e Gaudoin \(2004\)](#) o tipo de reparo é parametrizado. Na próxima seção vamos detalhar mais a classe de modelos ARA e ARI de [Doyen e Gaudoin \(2004\)](#).

2.4 Classe de Modelos ARA e ARI

Apresentaremos nesta seção a classe de modelos ARA e ARI, evidenciando os resultados da classe ARA, dada que o caso particular ARA_1 será explorado nos próximos capítulos.

2.4.1 Classe ARA

A classe ARA considera que o reparo rejuvenesce o sistema de tal forma que sua intensidade no tempo t é igual à intensidade inicial no tempo V_t , que é definida como a idade virtual do sistema dada por uma função linear $\phi(\theta, t, T_{N_t}, N_t)$, sendo θ o fator de reparo para redução ou aumento linear da idade em t . Exemplo: $V_t = t - \theta T_{N_t}$. O modelo ARA_m tem uma

intensidade de falha dada por

$$\lambda_{ARA_m}(t|H_{t-}) = \lambda(V_{tm}) = \lambda\left(t - \theta \sum_{p=0}^{\min(m-1, N_{t-}-1)} (1-\theta)^p T_{N_{t-}-p}\right),$$

em que $\lambda(\cdot)$ é a função de intensidade do NHPP até a primeira falha e que

$$V_{tm} = t - \theta \sum_{p=0}^{\min(m-1, N_{t-}-1)} (1-\theta)^p T_{N_{t-}-p}.$$

Dois importantes casos particulares da classe ARA são os modelos ARA_1 e ARA_∞ . Se $m = 1$ temos que a função de intensidade de falha para o modelo ARA_1 que é dada por

$$\lambda_{ARA_1}(t|H_{t-}) = \lambda(t - \theta T_{N_{t-}}),$$

por outro lado, na classe ARA_∞ assume-se que $m = N_{t-}$. Nesse caso, a função de intensidade de falha é dada por

$$\lambda_{ARA_\infty}(t|H_{t-}) = \lambda\left(t - \theta \sum_{p=0}^{N_{t-}-1} (1-\theta)^p T_{N_{t-}-p}\right).$$

Note que os modelos RM e RP são casos particulares do ARA_1 quando $\theta = 0$ e $\theta = 1$, respectivamente. Se considerarmos um PLP no modelo ARA_m , podemos reescrever a função de intensidade de falha como

$$\lambda_{ARA_m}(t|H_{t-}) = (\mu_T/T^\beta)\beta \left(t - \theta \sum_{p=0}^{\min(m-1, N_{t-}-1)} (1-\theta)^p T_{N_{t-}-p}\right)^{\beta-1}.$$

A Figura 6 exemplifica a dinâmica do modelo ARA_1 , com

$$\lambda_{ARA_1}(t|H_{t-}) = (\mu_T/T^\beta)\beta(t - \theta T_{N_{t-}})^{\beta-1},$$

cujos os efeitos dos reparos são notados pelas quedas pós falhas. Neste modelo, as curvas entre falhas se encaixam horizontalmente com a curva projetada até a primeira falha. Os pontos vermelhos indicam os tempos de falha.

Em especial, podemos escrever a função de verossimilhança para estimação dos parâmetros do ARA_1 como

$$L_{ARA_1}(\Theta|\mathbf{t}) = \prod_{j=i}^k \left\{ \left(\prod_{i=1}^{n_j} (\mu_T/T^\beta)\beta(t_{ij} - \theta t_{(i-1)j})^{\beta-1} \right) e^{-\Lambda^{(j)}(T)} \right\},$$

em que $\Lambda^{(j)}(T) = \sum_{i=1}^{n_j+1} \int_{t_{(i-1)j}}^{t_{ij}} \lambda_{ARA_1}(x|H_{x-})dx$, com $t_{0j} = 0$ e $t_{(n_j+1)j} = T$. Perceba que se $t = t_{ij}$, então $T_{N_{t-}} = t_{(i-1)j}$. Logo, $\lambda_{ARA_1}(t_{ij}|H_{t_{ij}-}) = (\mu_T/T^\beta)\beta(t_{ij} - \theta t_{(i-1)j})^{\beta-1}$.

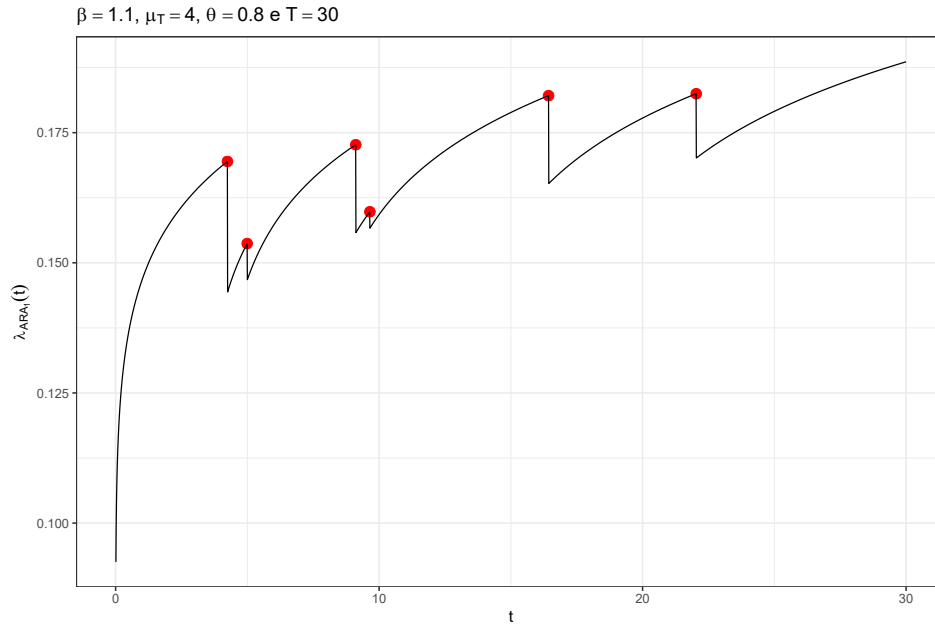


Figura 6 – Exemplo de função de intensidade $\lambda_{ARA_1}(t|H_{t-})$, com $\beta = 1,1$, $\theta = 0,8$, $\mu_T = 4$ e $T = 30$.

2.4.2 Classe ARI

A ideia básica da classe de modelos ARI é considerar que cada reparo reduz a intensidade de falha, dependendo do histórico de falhas do processo. No modelo ARI_m assume-se que o reparo reduz o incremento na intensidade da falha desde as últimas m falhas. Assim, a função de intensidade de falha para um modelo da classe ARI_m é definido por

$$\lambda_{ARI_m}(t|H_{t-}) = \lambda(t) - \theta \sum_{p=0}^{\min(m-1, N(t)-1)} (1-\theta)^p \lambda(T_{N_{t-}-p}),$$

em que $\lambda(t)$ é a função de intensidade até a primeira falha.

Analogamente à classe ARA, na classe ARI_m podemos considerar a existência de dois casos extremos, o ARI_1 e o ARI_∞ . Se $m = 1$ temos que a função de intensidade de falha para o modelo ARI_1 que é dada por

$$\lambda_{ARI_1}(t|H_{t-}) = \lambda(t) - \theta \lambda(T_{N_{t-}}).$$

Para o modelo ARI_∞ assume-se que cada reparo reduz a intensidade da falha, nesse caso, a função de intensidade de falha é dada por

$$\lambda_{ARI_\infty}(t|H_{t-}) = \lambda(t) - \theta \sum_{p=0}^{N_{t-}-1} (1-\theta)^p \lambda(T_{N_{t-}-p}).$$

Novamente, se considerarmos um PLP no modelo ARI_m , podemos reescrever a função de intensidade de falha como

$$\lambda_{ARI_m}(t|H_{t-}) = \frac{\mu_T \beta}{T^\beta} \left[t^{\beta-1} - \theta \sum_{p=0}^{\min(m-1, N(t)-1)} (1-\theta)^p T_{N_{t-}-p}^{\beta-1} \right].$$

2.4.3 Modelos GRA_1 e GRI_1

Doyen, Gaudoin e Syamsundar (2017) propuseram os modelos de Redução Geométrica da Idade (GRA) ou Redução Geométrica da Intensidade (GRI) com memória um, abrindo a possibilidade de considerar comportamentos no processo de falha que não podem ser abordados pelos modelos ARA e ARI, como uma desaceleração mais forte do desgaste. Os modelos com Redução Geométrica da Idade (GRA) e os modelos com Redução Geométrica da Intensidade (GRI) são dados por:

- GRA_1 - 1: $\lambda(t|H_{t-}) = \lambda(t[1-\rho]^{N_{t-}})$
- GRA_1 - 2: $\lambda(t|H_{t-}) = \lambda\left(\frac{t}{[1+T_{N_{t-}}]^\rho}\right)$
- GRA_1 - 3: $\lambda(t|H_{t-}) = \frac{1}{1+t} \lambda\left(\log(1+t) - \rho \log(1+T_{N_{t-}})\right)$
- GRA_1 - 4: $\lambda(t|H_{t-}) = \rho^{T_{N_{t-}}} \lambda(t - T_{N_{t-}})$
- GRI_1 - 1: $\lambda(t|H_{t-}) = \lambda(t) \left[\frac{\lambda(0)}{\lambda(T_{N_{t-}})} \right]^\rho$
- GRI_1 - 2: $\lambda(t|H_{t-}) = \lambda(t) \left[\frac{1+\lambda(0)}{1+\lambda(T_{N_{t-}})} \right]^\rho$
- GRI_1 - 3: $\lambda(t|H_{t-}) = \lambda(t) \left[\frac{\lambda(T_1) + \lambda(0)}{\lambda(T_1) + \lambda(T_{N_{t-}})} \right]^\rho$
- GRI_1 - 4: $\lambda(t|H_{t-}) = [1 + \lambda(t)] \left[\frac{1 + \lambda(0)}{1 + \lambda(T_{N_{t-}})} \right]^\rho - 1$
- GRI_1 - 5: $\lambda(t|H_{t-}) = \frac{1}{1+t} \left[\lambda(\log[1+t]) - \rho \lambda(\log[1+T_{N_{t-}}]) + \rho \lambda(0) \right]$

Em que $\lambda(\cdot)$ é a função de intensidade até a primeira falha. Os autores também denominam GRA_1 de redução geométrica da idade Calabria - Pulcini (GRA-CP). O modelo GRI_1 é equivalente ao modelo Calabria - Pulcini (CP) proposto por Calabria e Pulcini (1999). Os modelos GRA e GRI podem ser considerados como uma opção em detrimento aos Modelos ARA e ARI propostos por Doyen e Gaudoin (2004) e o Processo Geométrico (GP) proposto por Lam (1988). O modelo GP é dado por:

- GP: $\lambda(t|H_{t-}) = \rho^{N_{t-}} \lambda(\rho^{N_{t-}}(t - T_{N_{t-}}))$.

Doyen, Gaudoin e Syamsundar (2017) ressaltam que o modelo proposto por Finkelstein (2008) denominado *Geometric Failure Rate Reduction model* (GFRR), com intensidade de falha:

- GFRR: $\lambda(t|H_{t-}) = \rho^{N_{t-}} \lambda(t - T_{N_{t-}})$,

pode ser considerado como um caso especial de modelo de redução geométrica de intensidade, embora este modelo não pertença às categorias apresentadas acima.

2.4.4 Modelos XARA

Syamsundar, Naikan e Wu (2021) apresentaram o modelo XARA, *Extended arithmetic reduction of age models*, em português, modelos de redução aritmética estendida da idade. Nesse modelo, os autores inovam com a introdução de um fator de reparo geométrico, que acelera ou diminui a idade virtual ou a função de intensidade de falha. O fator de reparo geométrico é definido como uma função de i , ou seja, o fator de reparo geométrico ρ^i , $i = 1, 2, 3, \dots, n$ ou $\rho^{N_{t-}}$ é um fator para acelerar ou desacelerar a idade virtual ou intensidade do modelo ARA. A construção se baseia na combinação de outros modelos já existentes.

O modelo XARA se desmembra em dois modelos, denominados XARA-GRA e XARA-GRI. As características dos modelos ARA e GRA-CP/GP são combinadas para formar o XARA-GRA. Já para originar o modelo XARA-GRI, as características dos modelos ARA e CP/GFRR são combinadas. Portanto, esses novos modelos propostos combinam as características dos modelos de reparo imperfeito linear e geométrico para superar as desvantagens desses modelos, isso permite, segundo os autores, que os modelos XARA sejam capazes de acomodar outras naturezas de intensidades de falha.

Os modelos XARA-GRA e XARA-GRI tem as formas gerais dadas por, respectivamente:

$$\lambda(t|H_{t-}) = \rho_{2i} \lambda(\rho_{2i} V_t) \text{ e } \lambda(t|H_{t-}) = \rho_{2i} \lambda(V_t),$$

em que ρ_{2i} é o fator de reparo geométrico que acelera/desacelera a idade ou a intensidade e é uma função de i . V_t é a idade virtual e é um função $\phi(\rho_1, t, T_{N_{t-}}, N_{t-})$, sendo ρ_1 o fator de reparo para redução/aumento linear da idade ou intensidade. Casos especiais:

- XARA-GRA₁:

$$\lambda(t|H_{t-}) = \rho_2^{N_{t-}} \lambda(\rho_2^{N_{t-}}(t - \rho_1 T_{N_{t-}}))$$

- XARA-GRI₁:

$$\lambda(t|H_{t-}) = \rho_2^{N_{t-}} \lambda(t - \rho_1 T_{N_{t-}})$$

em que $\lambda(t)$ é a função de intensidade anterior à primeira falha, $\rho_{2i} = \rho_2^{N_{t-}}$ e $V_t = t - \rho_1 T_{N_{t-}}$.

2.4.5 Modelos com pontos de mudança

Um sistema reparável pode ser impactado por uma mudança no ambiente em que se encontra, modificando o comportamento do processo e segmentando sua função de intensidade de falha, tal como mudanças de temperatura, pressão, poeira, umidade, etc. [Syamsundar e Naikan \(2008\)](#) propuseram modelos com múltiplos pontos de mudanças na função de intensidade, por meio do qual é possível a obtenção das estimativas dos tempos em que ocorreram as alterações no processo de falha do sistema. Independente do tipo de reparo, um modelo com múltiplos pontos de mudanças é dado por:

$$\begin{aligned}\lambda(t|H_{t-}) &= \lambda_1(t), \text{ para } 0 \leq t \leq \tau_1 \\ &= \lambda_i(t - \tau_{i-1}), \text{ para } \tau_{i-1} \leq t \leq \tau_i \\ &= \lambda_p(t - \tau_p), \text{ para } \tau_p \leq t \leq T\end{aligned}$$

em que $i = 2, \dots, p$ que é o número de pontos de mudanças e T é o tempo de truncamento. Com $\lambda(t)$ sendo representado por um PLP, o modelo com múltiplos ponto de mudança fica:

$$\begin{aligned}\lambda(t|H_{t-}) &= \alpha_1 \beta_1 t^{\beta_1-1}, \text{ para } 0 \leq t \leq \tau_1 \\ &= \alpha_i \beta_i (t - \tau_{i-1})^{\beta_i-1}, \text{ para } \tau_{i-1} \leq t \leq \tau_i \\ &= \alpha_{p+1} \beta_{p+1} (t - \tau_p)^{\beta_{p+1}-1}, \text{ para } \tau_{p-1} \leq t \leq T,\end{aligned}$$

em que $i = 2, \dots, p$.

Vale ressaltar que $\lambda_i(\cdot)$ pode ser uma função de intensidade HPP, NHPP ou até mesmo uma função de intensidade de uma das classes de modelos ARA ou ARI. Seguem alguns casos, com um ponto τ de mudanças.

- Processo de Poisson Homogêneo (HPP)

$$\begin{aligned}\lambda(t|H_{t-}) &= \lambda_1, \text{ para } 0 \leq t \leq \tau \\ &= \lambda_2, \text{ para } \tau \leq t \leq T.\end{aligned}$$

- Processo de Poisson Não Homogêneo (NHPP)

$$\begin{aligned}\lambda(t|H_{t-}) &= \lambda_1(t), \text{ para } 0 \leq t \leq \tau \\ &= \lambda_2(t - \tau), \text{ para } \tau \leq t \leq T.\end{aligned}$$

ARA₁ MODIFICADO COM PLP PARA MÚLTIPLOS SISTEMAS REPARÁVEIS

3.1 Introdução

O PLP foi introduzido por [Crow \(1974\)](#) e tem sido largamente utilizado na modelagem de Processo Poisson Não Homogêneo (NHPP) por conta da estrutura matemática da sua função de intensidade ser bastante flexível e os parâmetros associados terem interpretações práticas relevantes. Em análise de sistemas reparáveis, o uso do PLP possibilita modelar sistemas em estado de deterioração, renovação ou até mesmo que permaneçam estáveis ao longo do tempo. A função de intensidade do PLP pode ser encontrada em diversas parametrizações na literatura, as mais comuns são:

- (P1) $\lambda(t) = \alpha\beta t^{\beta-1}$, $\alpha > 0$, $\beta > 0$, $t > 0$, [Crow \(1974\)](#);
- (P2) $\lambda(t) = (\beta/\eta^\beta)t^{\beta-1}$, $\eta > 0$, $\beta > 0$, $t > 0$, [Oliveira, Colosimo e Gilardoni \(2013\)](#);
- (P3) $\lambda(t) = (\mu_T\beta/T^\beta)t^{\beta-1}$, $\mu_T > 0$, $\beta > 0$, $T > 0$, $t > 0$, [Oliveira, Colosimo e Gilardoni \(2012\)](#).

Considerando o contexto de sistemas reparáveis, temos que para todas as parametrizações acima, o parâmetro β é de forma e sua interpretação está relacionada ao estado do sistema, isto é, valores de $\beta > 1$ indicam degradação/envelhecimento do sistema ao passar o tempo, uma vez que a função de intensidade de falha $\lambda(t)$ é estritamente crescente. Por outro lado, para $\beta < 1$, o sistema melhora com o tempo, pois a função de intensidade é estritamente decrescente. Além disso, quando $\beta = 1$ a função de intensidade é constante, ou seja, o sistema é estável em termos de $\lambda(t)$, portanto, o processo é Poisson homogêneo (HPP). A partir disso, temos que (P1) foi a proposta original de [Crow \(1974\)](#) e o parâmetro α tem interpretação genérica, representando

mudança escalar de $\lambda(t)$, quanto maior seu valor, maiores são os resultados obtidos da função de intensidade de falha.

Agora, seja $N(t)$ a variável aleatória de contagem do número de falhas até um tempo t , então o número médio de falhas de $N(t)$ é dado por $\Lambda(t) = E[N(t)] = \int_0^t \lambda(x) dx$. Considerando a parametrização (P2), o parâmetro η , também escalar, é interpretado como o tempo para o qual se espera exatamente uma falha, isto é, $\Lambda(\eta) = E[N(\eta)] = \int_0^\eta \lambda(x) dx = (\eta/\eta)^\beta = 1$, conforme Oliveira, Colosimo e Gilardoni (2013), que a utilizou para modelar múltiplos sistemas reparáveis heterogêneos. Por (P1) ser mais genérica, podemos reescrevendo (P1) como $\alpha = (1/\eta)^\beta$ e obtemos (P2).

Por fim, a função de intensidade dada em (P3) foi proposta Oliveira, Colosimo e Gilardoni (2012), com μ_T e β ortogonais, em que μ_T é o número médio de falhas até o tempo de truncamento T , uma vez que $\Lambda(T) = E[N(T)] = \mu_T$. Também pode ser obtida a partir da (P1), com $\alpha = \mu_T/T^\beta$. Para modelos em que as falhas serão observadas até um tempo limite predeterminado T , a parametrização (P3) é mais vantajosa do ponto de vista da interpretação, uma vez que agrega o próprio tempo de truncamento e a média até T , que é dada pelo parâmetro μ_T . A função de intensidade de falha acumulada de (P3) é dada por $\Lambda(t) = \mu_T(t/T)^\beta$.

O uso da função de intensidade de falha do PLP é restrito aos sistemas que tiveram reparos mínimos, isto é, voltam, em termos de $\lambda(t)$, ao estágio imediatamente anterior ao reparo, ou seja, tão ruim quanto velho (ABAO - *as bad as old*), tornando-o inadequado para situações em que o sistema retorna a um estado intermediário entre tão ruim quanto velho e tão bom quanto novo, comumente denominado de reparo imperfeito (RI). Nesse sentido, Doyen e Gaudoin (2004) propuseram uma classe de modelos de reparo imperfeito com PLP baseado em idade virtual do sistema, que é uma redução linear da idade cronológica devido aos reparos realizados. Os autores denominaram essa nova classe de *Arithmetic Reduction of Age* (ARA).

No modelo ARA_m , da classe ARA, o efeito do reparo realizado é medido através do parâmetro ρ , de forma que: (i) se $\rho < 0$ o reparo é prejudicial; (ii) se $\rho = 0$ o reparo é mínimo; (iii) se $0 < \rho < 1$ o reparo é eficiente; e (iv) se $\rho = 1$ o reparo é ótimo/perfeito. O valor m representa o número máximo de tempos de falha anteriores que estão presentes na intensidade da falha, exemplo, a função de intensidade do ARA com memória um (ARA_1) depende do último tempo de falha.

Contudo, há uma limitação, pois o modelo ARA é definido para $\beta > 1$ no Processo de Lei de Potência, não permitindo, por tanto, capturar a informação do estado do sistema que possivelmente esteja melhorando até um certo tempo T , ou seja, sistema com processo de falha em que a função de intensidade é decrescente, no PLP (P3), $\beta < 1$. Diante dessa limitação, propomos o modelo ARA_1 modificado ($ARAM_1$), que possibilita modelar sistemas em processo de melhoria ($\beta < 1$), de constância ($\beta = 1$) ou de degradação ($\beta > 1$).

O resto do capítulo segue da seguinte forma. Na Seção 3.2 é mostrado o Modelo ARA_1

com limitações do PLP (P3) e uma nova parametrização para o PLP será apresentada. Nas Seções 3.3 e 3.4, apresentamos, respectivamente, o modelo ARA_1 modificado ($ARAM_1$) com PLP e o método de estimação de seus parâmetros. Em seguida, na Seção 3.5, avaliamos a performance dos estimadores dos parâmetros através do estudo de simulação de Monte Carlo (MC). Na Seção 3.6 apresentamos uma aplicação do modelo $ARAM_1$ em dados reais de tempos de falha de lâmina *Chopper* localizada no sistema de corte de cana-de-açúcar em nove colheitadeiras. A Seção 3.7 resume as principais conclusões do capítulo.

3.2 Modelo ARA_1 considerando (P3) e limitações

Definição 5. O modelo de redução aritmética da idade com memória um, *Arithmetic Reduction of Age model* (ARA_1), com PLP proposto por Doyen e Gaudoin (2004) tem intensidade dada por

$$\lambda_{ARA_1}(t|H_{t-}) = \lambda(t - \theta T_{N_{t-}}) = \frac{\mu_T \beta}{T} \left(\frac{t - \theta T_{N_{t-}}}{T} \right)^{\beta-1},$$

em que $\beta \geq 1$, $\theta \leq 1$ é o efeito do reparo na redução da idade cronológica, $T_{N_{t-}}$ e H_{t-} são, respectivamente, o tempo e o histórico de falhas imediatamente anterior à t e $\lambda(\cdot)$ é dada por (P3).

Originalmente, a função de intensidade do modelo ARA_1 com PLP é parametrizada por (P1) com $\theta = \rho$, como nossa proposta é baseada em truncamento por tempo, logo é mais coerente o uso do PLP parametrizado por (P3).

Com relação à interpretação dos parâmetros do modelo ARA_1 , percebe-se que quanto maior θ , menor será $\lambda_{ARA_1}(t|H_{t-})$ e é desejável que atinja o seu máximo, ou seja, $\theta = 1$, significando que os reparos realizados no sistema foram perfeitos. Como já observado, o ARA_1 não está definido para $\beta < 1$, isto é, quando a função de intensidade é decrescente, pois, para exemplificar, se $\beta < 1$ e $\theta = 1$, o reparo seria classificado como reparo perfeito, contudo, essa situação implica em número de falhas maior se comparado ao sistema que recebeu reparo mínimo ($\theta = 0$), mantendo-se constante os demais parâmetros. De modo geral, se $\theta_1 < \theta_2$ e mantendo-se constantes os demais parâmetros, então $\lambda_{ARA_1}(t|H_{t-}; \theta_1) < \lambda_{ARA_1}(t|H_{t-}; \theta_2)$, pois $t - \theta_2 T_{N_{t-}} < t - \theta_1 T_{N_{t-}}$, como $\beta - 1 < 0$, então $[t - \theta_1 T_{N_{t-}}]^{\beta-1} < [t - \theta_2 T_{N_{t-}}]^{\beta-1}$, segue que $\lambda_{ARA_1}(t|H_{t-}, \theta_1) < \lambda_{ARA_1}(t|H_{t-}, \theta_2)$.

Portanto, de fato, não faz sentido definir o modelo ARA_1 para $\beta < 1$, uma vez que θ perde o sentido da interpretação, isto é, quanto θ mais próximo de um, pior é o efeito do reparo. No entanto, essa barreira limita o modelo ARA_1 , uma vez que não inclui sistema que possivelmente estava melhorando até certo tempo. Enfim, sem o devido cuidado, também pode-se incorrer em conclusões absurdas de interpretações a respeito do parâmetro característico do efeito do reparo.

Essa situação é contornável aplicando o mesmo princípio da redução da idade cronológica para sistemas em degradação ao longo do tempo, uma vez que, ao reduzir a idade, o sistema retorna ao estágio cuja função de intensidade é menor, já que $\beta > 1$. Em sistema que melhoram ou se renovam em termos de função de intensidade de falha, isto é, $\beta < 1$, a intensidade reduzirá com o aumento da idade virtual, indicando efeito positivo do reparo. Dessa maneira, uma solução para o problema é condicionando $\lambda_{ARA_1}(t|H_{t-})$ da seguinte maneira: $\lambda_{ARA_1}(t|H_{t-}) = \lambda(t - \theta T_{N_{t-}})$ para $\beta \geq 1$ e $\lambda_{ARA_1}(t|H_{t-}) = \lambda(t + \theta T_{N_{t-}})$ para $\beta < 1$, com $\lambda(t)$ dada por (P3), assim,

$$\lambda_{ARA_1}(t|H_{t-}) = \begin{cases} \frac{\mu_T \beta}{T} \left(\frac{t - \theta T_{N_{t-}}}{T} \right)^{\beta-1}, & \text{se } \beta \geq 1 \\ \frac{\mu_T \beta}{T} \left(\frac{t + \theta T_{N_{t-}}}{T} \right)^{\beta-1}, & \text{se } \beta < 1. \end{cases} \quad (3.1)$$

Dessa forma, para todo $\beta > 0$, com $\theta_1 < \theta_2$, temos $\lambda_{ARA_1}(t|H_{t-}; \theta_2) < \lambda_{ARA_1}(t|H_{t-}; \theta_1)$, uma vez que

$$\frac{d}{d\theta} \lambda_{ARA_1}(t|H_{t-}) = \begin{cases} \frac{-\mu_T \beta (\beta - 1) T_{N_{t-}}}{T^\beta} (t - \theta T_{N_{t-}})^{\beta-2} < 0, & \text{se } \beta > 1, \\ \frac{\mu_T \beta (\beta - 1) T_{N_{t-}}}{T^\beta} (t + \theta T_{N_{t-}})^{\beta-2} < 0, & \text{se } \beta < 1, \end{cases}$$

ou seja, como a primeira derivada de $\lambda_{ARA_1}(t|H_{t-})$ em relação ao parâmetro θ é negativa para todo θ , então $\lambda_{ARA_1}(t|H_{t-})$ é decrescente em termos de θ , logo, quanto mais próximo do reparo perfeito, menor é a função de intensidade de falha, solucionando o caso em que $\beta < 1$ no modelo ARA_1 .

Outro ponto a ser observado diz respeito à interpretação do parâmetro β do PLP com a parametrização dada por (P3). É intuitivo esperar que sistemas com valores de β maiores, mantendo-se constantes os demais parâmetros, tenham suas funções de intensidade de falha também maiores para todo t . Contudo, não é isso que ocorre quando utilizada $\lambda(t)$ com tal parametrização. Por exemplo, considere dois sistemas com β_1 e β_2 , tal que, $\beta_1 < \beta_2$. É intuitivo esperar, pela ordem de grandeza de β_1 e β_2 que $\lambda(t; \beta_1) < \lambda(t; \beta_2)$ para todo $t > 0$. Contudo,

$$\text{se } \beta_1 < \beta_2, \text{ então } \lambda(t; \beta_1) < \lambda(t; \beta_2) \text{ quando } t > T \left(\frac{\beta_1}{\beta_2} \right)^{1/(\beta_2 - \beta_1)},$$

com isso, β não pode ser utilizado como parâmetro indicador de força monótona da deterioração ou renovação do sistema em termos de $\lambda(t)$, somente indicador do estado do sistema. Por exemplo, para $\mu_T > 0$, $T = 50$, $\beta_1 = 1.5$ e $\beta_2 = 2.0$ então $\lambda(t; \beta_1) < \lambda(t; \beta_2)$ quando $t > 28.12$, ou seja, somente em 43.76% do tempo até o truncamento $T = 50$ (Figura 7).

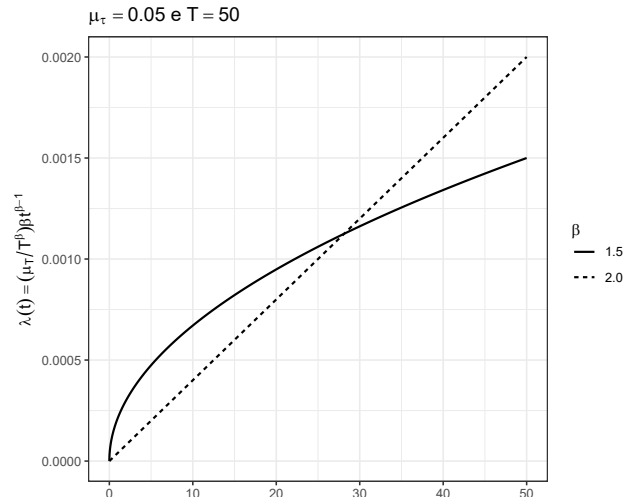


Figura 7 – Parametrização (P3) $\lambda(t) = (\mu_T/T^\beta)\beta t^{\beta-1}$ com $\beta = 1,5$ e 2 .

Uma solução para aumentar o tempo t em que $\lambda(t; \beta_1) < \lambda(t; \beta_2)$, dado $\beta_1 < \beta_2$, é modificando $\lambda(t)$ de modo a obtermos $t > (\beta_1/\beta_2)^{1/(\beta_2-\beta_1)}$, sem o fator T , em que $(\beta_1/\beta_2)^{1/(\beta_2-\beta_1)} < 1$. Assim, propomos uma nova parametrização para o PLP, referenciaremos por (P4), dada pela seguinte função de intensidade:

$$\lambda(t) = \frac{\mu_T}{T} \beta t^{\beta-1}, \quad (3.2)$$

como consequência, a interpretação de μ_T passa a ser a média de um processo HPP ($\beta = 1$) até T . No mesmo exemplo, $\lambda(t; \beta_1) < \lambda(t; \beta_2)$ quando $t > 0,5625$, ou seja, em 98,87% do tempo até o truncamento $T = 50$ (Figura 8). Note que a função de intensidade com parametrização (P4), dada em (3.2), é um PLP com $\alpha = V_m = \mu_T/T$ na parametrização (P1), em que V_m é a velocidade instantânea de falha em t para um HPP.

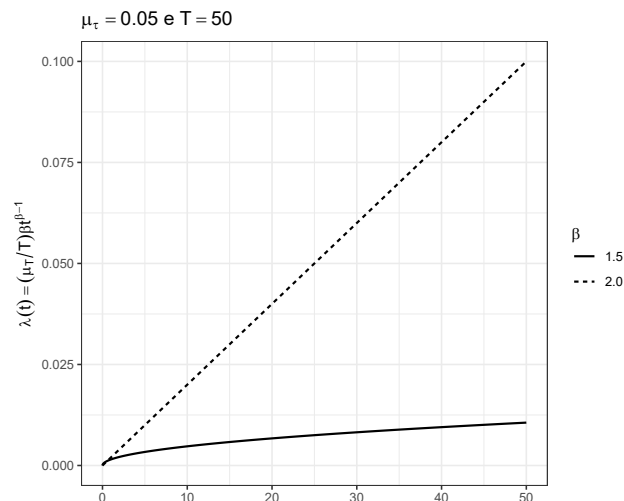


Figura 8 – Parametrização $\lambda(t) = (\mu_T/T)\beta t^{\beta-1}$ com $\beta = 1,5$ e 2 .

3.3 Modelo $ARAM_1$ considerando parametrização (P4)

O modelo $ARAM_1$ é proposto objetivando suprir limitações do ARA_1 , utilizado na modelagem de tempos de falha de sistemas com reparos imperfeitos, condicionando a função de intensidade de falha à situação do sistema para manter o sentido decrescente do efeito do reparo sobre a função de intensidade, interpretando β como quantificador da intensidade do estado do sistema, além de indicador da situação do sistema (estado de degradação, constante ou renovação). O efeito do reparo age na redução ou aumento da idade cronológica do sistema por meio de uma função $h(\theta)$. Assumiremos que o reparo é executado após cada falha e que o intervalo de tempos entre falha e reparo é positivo mas são tão pequenos quanto desejarmos.

Definição 6. Definiremos a função de intensidade $\lambda_{ARAM_1}(t|H_{t-})$ do modelo $ARAM_1$ como

$$\lambda_{ARAM_1}(t|H_{t-}) = \begin{cases} \frac{\mu_T \beta}{T} (t - h(\theta)T_{N_{t-}})^{\beta-1}, & \text{se } \beta \geq 1, \\ \frac{\mu_T \beta}{T} (t + h(\theta)T_{N_{t-}})^{\beta-1}, & \text{se } \beta < 1, \end{cases} \quad (3.3)$$

em que $T_{N_{t-}}$ e H_{t-} representam, respectivamente, o tempo e o histórico de falhas imediatamente anterior à t , $\mu_T > 0$, $\beta > 0$, $\theta \in [-1, 1]$ e $T > 0$.

Para padronização dos valores de θ no intervalo $[-1, 1]$ a função $h(\theta)$ é estritamente crescente é definida da seguinte maneira:

1) Para $\beta \geq 1$, $h(\theta) : (-1, 1] \rightarrow (-\infty, 1]$, tal que

$$h(\theta) = \begin{cases} \log\left(\frac{1+\theta}{1-\theta}\right)10^k, & \text{se } -1 < \theta \leq 0, \\ \theta, & \text{se } 0 \leq \theta \leq 1, \end{cases} \quad (3.4)$$

2) Para $\beta < 1$, $h(\theta) : [-1, 1) \rightarrow [-1, \infty)$, tal que

$$h(\theta) = \begin{cases} \theta, & \text{se } -1 \leq \theta \leq 0, \\ \log\left(\frac{1+\theta}{1-\theta}\right)10^k, & \text{se } 0 \leq \theta < 1, \end{cases}$$

em que $k = 1, 2, \dots$ é uma constante.

Outras funções $h(\theta)$ crescentes, relacionadas ao efeito do reparo, podem ser definidas. O fator 10^k em $h(\theta)$ calibra a intensidade do efeito negativo do reparo ($\beta \geq 1$ e $-1 < \theta \leq 0$) ou efeito positivo ($\beta < 1$ e $0 \leq \theta < 1$), o que permite intensificar na função de intensidade o efeito de $h(\theta)$, possibilitando melhores estimativas do parâmetro θ , vamos adotar neste trabalho $k = 4$,

contudo, simulações podem ser realizadas no processo de estimação dos parâmetros do ARAM₁ para identificar qual constante k considerar, por exemplo, a constante $k \in \mathbb{N}$ que maximiza a função de log-verossimilhança em uma amostra de tempos.

Em relação à interpretação do parâmetro β do modelo proposto, considere, por exemplo, dois sistemas tais que $1 < \beta_1 < \beta_2$, então, mantendo-se constantes os demais parâmetros,

$$\lambda_{ARAM_1}(t|H_{t-}; \beta_1) < \lambda_{ARAM_1}(t|H_{t-}; \beta_2),$$

para $t > T_{N_{t-}}$, se $T_{N_{t-}} - h(\theta)T_{N_{t-}} > 1$, pois $(\beta_1/\beta_2)^{1/(\beta_2-\beta_1)} < 1$. Dessa maneira, o parâmetro β é caracterizado como indicador do estado do sistema, assim como a força em que o sistema se deteriora ou melhora ao longo do tempo.

O modelo ARA₁ é um caso particular do ARAM₁, uma vez que para $\beta \geq 1$ e $h(\theta) = \theta \leq 1$ obtemos $\lambda_{ARA_1}(t|H_{t-}) = (\mu_T/T)\beta(t - \theta T_{N_{t-}})^{\beta-1}$ com parametrização (P4). No ARAM₁, em sua forma mais simples, com $\beta = 1$, o parâmetro escalar μ_T é interpretado como a média do processo HPP até T , uma vez que, nessas condições, a função de intensidade para todo $\beta > 0$ é dada por $\lambda_{ARAM_1}(t|H_{t-}) = \mu_T/T = V_m$ e $\Lambda_{ARAM_1}(t|H_{t-}) = (\mu_T/T)t = V_m t$, em que V_m é a velocidade instantânea de falha e, conseqüentemente, $\Lambda(T) = \mu_T$.

Percebe-se em (3.3) que quanto maior o número médio de falha μ_T mais elevada é a função de intensidade $\lambda_{ARA_1}(t|H_{t-})$. Já β é o parâmetro de elasticidade, de modo que para $\beta > 1$ significa que o sistema está deteriorando e para $\beta < 1$ o sistema está melhorando ao longo do tempo. O parâmetro θ representa o efeito do histórico dos reparos sobre o sistema. Quando os reparos realizados são mínimos, então $\theta = 0$, implicando em $h(0) = 0$, logo $\lambda_{ARAM_1}(t|H_{t-})$ é mais elevada comparativamente a $\theta > 0$. A situação não desejável é quando $-1 \leq \theta < 0$, implicando em um reparo prejudicial. Ou seja, quanto menor θ , menor será o valor de $h(\theta)$, implicando em uma aumento de $\lambda_{ARAM_1}(t|H_{t-})$, pois

$$\frac{d}{dh(\theta)}\lambda_{ARAM_1}(t|H_{t-}) = \begin{cases} \frac{-\mu_T\beta(\beta-1)T_{N_{t-}}}{T}(t-h(\theta)T_{N_{t-}})^{\beta-2} < 0, & \text{se } \beta > 1 \\ \frac{\mu_T\beta(\beta-1)T_{N_{t-}}}{T}(t+h(\theta)T_{N_{t-}})^{\beta-2} < 0, & \text{se } \beta < 1, \end{cases}$$

ou seja, como a primeira derivada de $\lambda_{ARAM_1}(t|H_{t-})$ em relação à $h(\theta)$ é negativa, então $\lambda_{ARAM_1}(t|H_{t-})$ é decrescente em termos de $h(\theta)$, logo, quanto menor θ (menor será $h(\theta)$), maior é a função de intensidade de falha. Por exemplo, para reparo mínimo, a função de intensidade dada em (3.3) fica para todo $\beta > 0$

$$\lambda_{ARAM_1}(t|H_{t-}, h(0) = 0) = \frac{\mu_T}{T}\beta t^{\beta-1} = V_m\beta t^{\beta-1}, \quad (3.5)$$

um reparo melhor que o mínimo, ou seja, $h(\theta) > 0$, implica que a intensidade de falha, dada em (3.3), é menor que a intensidade de falha de sistemas em que é realizado reparo mínimo, dada por (3.5), conseqüentemente, o número médio de falhas é menor, pois temos redução ($\beta \geq 1$) ou aumento ($\beta < 1$) da idade do sistema. Logo, no ARAM₁, o parâmetro θ é o indicador da

eficiência do reparo e $h(\theta)$ é a função do efeito de θ sobre $\lambda_{ARAM_1}(t|H_{t-})$. De modo geral, quando o reparo é melhor que o reparo mínimo ou que o prejudicial, $\lambda_{ARAM_1}(t|H_{t-}, h(\theta) > 0) < \lambda_{ARAM_1}(t|H_{t-}, h(\theta) \leq 0)$ para todo $t > 0$, isto é, a função de intensidade de falha reduz com o aumento da eficiência da manutenção corretiva, conforme exemplificado na Figura 9.

Os piores cenários ocorrem quando $\theta = 0,2$, tanto no caso 1), para $\beta \geq 1$, quanto no caso 2), quando $\beta < 1$, pois a função de intensidade de falha é mais elevada comparativamente a outros valores de θ maiores que 0,2. Em ambos os cenários, percebemos que quanto maior θ , menor fica a função de intensidade, uma vez que com o aumento de θ implicando em melhor reparo, conseqüentemente, redução de $\lambda_{ARAM_1}(t|H_{t-})$ (Figura 9).

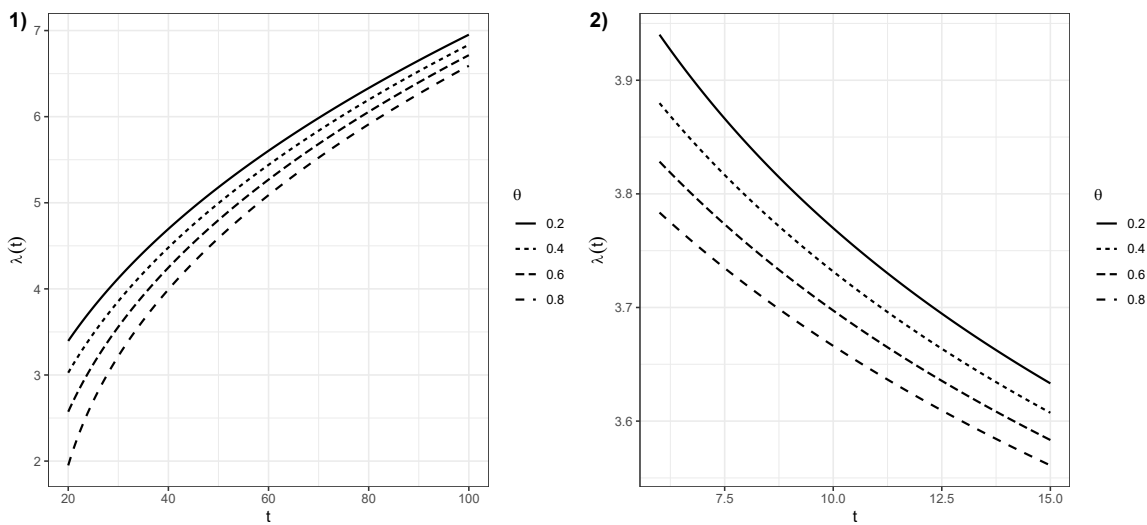


Figura 9 – Função de intensidade do $ARAM_1$ para diferentes valores de θ com 1) $t > T_{N_{t-}} = 20$, $\mu_T = 80$, $\beta = 1,4$ e $T = 100$ e 2) $t > T_{N_{t-}} = 6$, $\mu_T = 80$, $\beta = 0,9$ e $T = 15$.

No exemplo da Figura 10, após cada reparo ocorrido nos dias $t_1 = 20$, $t_2 = 40$, $t_3 = 60$ e $t_4 = 80$ (situação 1 com $\beta > 1$) e nos dias $t_1 = 10$, $t_2 = 20$, $t_3 = 30$ e $t_4 = 40$ (situação 2 com $\beta < 1$) a função de intensidade de falha apresenta melhora em ambos os cenários apresentadas, indicando o efeito do reparo quantificado como $h(\theta_0) = 0,8$, com $\theta_0 = h^{-1}(0,8)$, na redução da função de intensidade e conseqüentemente no número médio de falhas.

Situação 1), com $\beta > 1$ e $h(\theta_0) = 0,8$:

$$\lambda_{ARAM_1}(t) = \sum_{i=1}^5 \frac{80 \times 1,1}{200} (t - 0,8 \times t_{i-1})^{0,1} \mathbf{I}(t_{i-1} < t \leq t_i), \text{ em que } t_0 = 0 \text{ e } t_5 = T = 200.$$

Situação 2), com $\beta < 1$ e $h(\theta_0) = 0,8$:

$$\lambda_{ARAM_1}(t) = \sum_{i=1}^5 \frac{80 \times 0,6}{80} (t + 0,8 \times t_{i-1})^{-0,4} \mathbf{I}(t_{i-1} < t \leq t_i), \text{ em que } t_0 = 0 \text{ e } t_5 = T = 80.$$

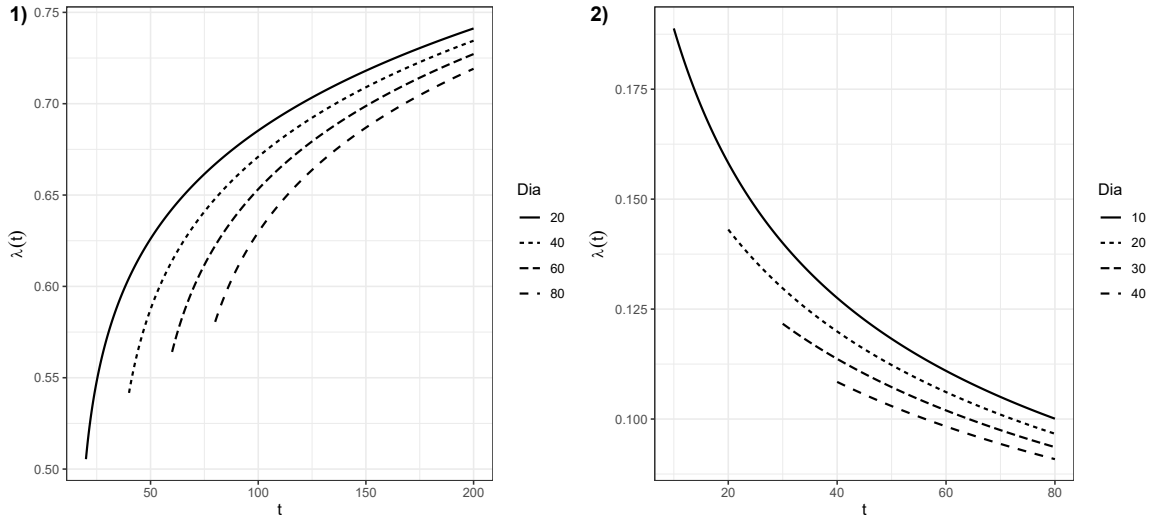


Figura 10 – Função de intensidade ARAM₁ para diferentes tempos de reparo, em dias, com 1) $\mu_T = 80$, $\beta = 1,1$ e $T = 200$ e 2) $\mu_T = 80$, $\beta = 0,6$ e $T = 80$.

A função de intensidade acumulada de falha da variável aleatória $N(0, t]$ dado H_{t-} para o modelo ARAM₁ é dada por

$$\begin{aligned}\Lambda_{ARAM_1}(t|H_{t-}) &= \int_0^t \lambda_{ARAM_1}(x|H_{x-})dx = \sum_{i=1}^{N_{t-}} \int_{T_{i-1}}^{T_i} \lambda_{ARAM_1}(x|H_{x-})dx + \int_{T_{N_{t-}}}^t \lambda_{ARAM_1}(x|H_{x-})dx \\ &= \sum_{i=1}^{N_{t-}} \Lambda_{ARAM_1}(T_i|T_{i-1}) + \Lambda_{ARAM_1}(t|T_{N_{t-}}),\end{aligned}$$

em que

$$\begin{aligned}\Lambda_{ARAM_1}(T_i|T_{i-1}) &= \int_{T_{i-1}}^{T_i} \lambda_{ARAM_1}(x|H_{x-})dx \\ &= \frac{\mu_T}{T} [(T_i - \text{sign}(\beta - 1)h(\theta)T_{i-1})^\beta - (T_{i-1} - \text{sign}(\beta - 1)h(\theta)T_{i-1})^\beta].\end{aligned}$$

e também que

$$\begin{aligned}\Lambda_{ARAM_1}(t|T_{N_{t-}}) &= \int_{T_{N_{t-}}}^t \lambda_{ARAM_1}(x|H_{x-})dx \\ &= \frac{\mu_T}{T} [(t - \text{sign}(\beta - 1)h(\theta)T_{N_{t-}})^\beta - (T_{N_{t-}} - \text{sign}(\beta - 1)h(\theta)T_{N_{t-}})^\beta],\end{aligned}$$

em que definimos $\text{sign}(\beta - 1) = 1$, caso $\beta \geq 1$ e $\text{sign}(\beta - 1) = -1$, caso $\beta < 1$. Assim, chegamos ao seguinte resultado

$$\begin{aligned}
\Lambda_{ARAM_1}(t|H_{t-}) &= \sum_{i=1}^{N_{t-}} \left\{ \frac{\mu_T}{T} \left[(T_i - h(\theta)T_{i-1})^\beta - (T_{i-1} - h(\theta)T_{i-1})^\beta \right] \right\} \\
&\quad + \frac{\mu_T}{T} \left[(t - h(\theta)T_{N_{t-}})^\beta - (T_{N_{t-}} - h(\theta)T_{N_{t-}})^\beta \right] \quad \text{se } (\beta \geq 1) \quad \text{e} \\
&\quad \sum_{i=1}^{N_{t-}} \left\{ \frac{\mu_T}{T} \left[(T_i + h(\theta)T_{i-1})^\beta - (T_{i-1} + h(\theta)T_{i-1})^\beta \right] \right\} \\
&\quad + \frac{\mu_T}{T} \left[(t + h(\theta)T_{N_{t-}})^\beta - (T_{N_{t-}} + h(\theta)T_{N_{t-}})^\beta \right] \quad \text{se } (\beta < 1). \quad (3.6)
\end{aligned}$$

A eficácia do reparo para o modelo $ARAM_1$ é interpretada da seguinte forma:

1) com $\theta = 0$, o reparo é mínimo e

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta t^{\beta-1}$$

2) com $\theta = 1$ e $\beta \geq 1$ o reparo é perfeito e

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta (t - T_{N_{t-}})^{\beta-1}$$

3) com $0 < \theta < 1$ e $\beta \geq 1$, o reparo é dito eficiente e a função é dada por

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta (t - \theta T_{N_{t-}})^{\beta-1}$$

4) com $-1 < \theta < 0$ e $\beta \geq 1$, o reparo é dito prejudicial e a função é dada por

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta (t - h(\theta)T_{N_{t-}})^{\beta-1}$$

5) com $0 < \theta < 1$ e $\beta < 1$ o reparo é eficiente e

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta (t + h(\theta)T_{N_{t-}})^{\beta-1}$$

6) com $-1 \leq \theta < 0$ e $\beta < 1$ o reparo é prejudicial e

$$\lambda_{ARAM_1}(t|H_{t-}) = \frac{\mu_T}{T} \beta (t + \theta T_{N_{t-}})^{\beta-1}.$$

Na Figura 11 apresentamos simulações de $\lambda_{ARAM_1}(t|H_{t-})$, em que os cenários 1) e 3) são reparos mínimos, eficiente no cenário 2) e perfeito no 4). Nos cenários em que os sistemas estão com intensidade de falha decrescente ($\beta < 1$) ou crescente ($\beta \geq 0$), podemos notar o efeito do reparo sobre a intensidade de falha, que é menor para os casos em que $\theta > 0$. O efeito do reparo é mais visível no cenário 4), em que simulamos reparos perfeitos em sistema com comportamento degradativo. Os pontos vermelhos nos gráficos representam o par ordenado $(T_i, \lambda_{ARAM_1}(T_i|H_{T_i-}))$ e o total desses pontos representam o número de falhas resultante da simulação. Notamos que

nos casos em que o efeito do reparo foi positivo, o número de falhas é menor, comparativamente aos cenários com os mesmos valores de β e μ_T . No cenário 3) em que $\theta = 0$, por exemplo, houve mais falhas que no cenário 4) em que foi simulado reparo perfeito ($\theta = 1$), em que os demais parâmetros são iguais para os dois cenários.

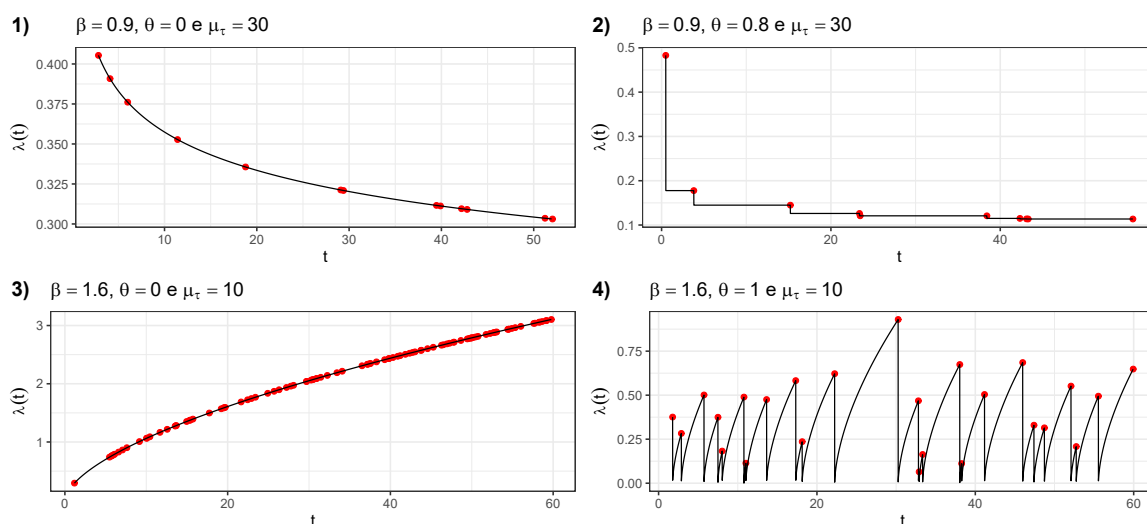


Figura 11 – Simulação de $\lambda_{ARAM_1}(t|H_{t-})$ para diferentes valores de β , θ e μ_T com $T = 60$.

Já na Figura 12, temos os gráficos das funções de intensidade simuladas para diferentes valores de β , θ e μ_T , cujos os sistemas estão melhorando nos cenários 1) e 2) e degradando nos cenários 3) e 4). Ressaltamos que o efeito do reparo é mais forte nas simulações em que foram considerados valores mais elevados de θ .

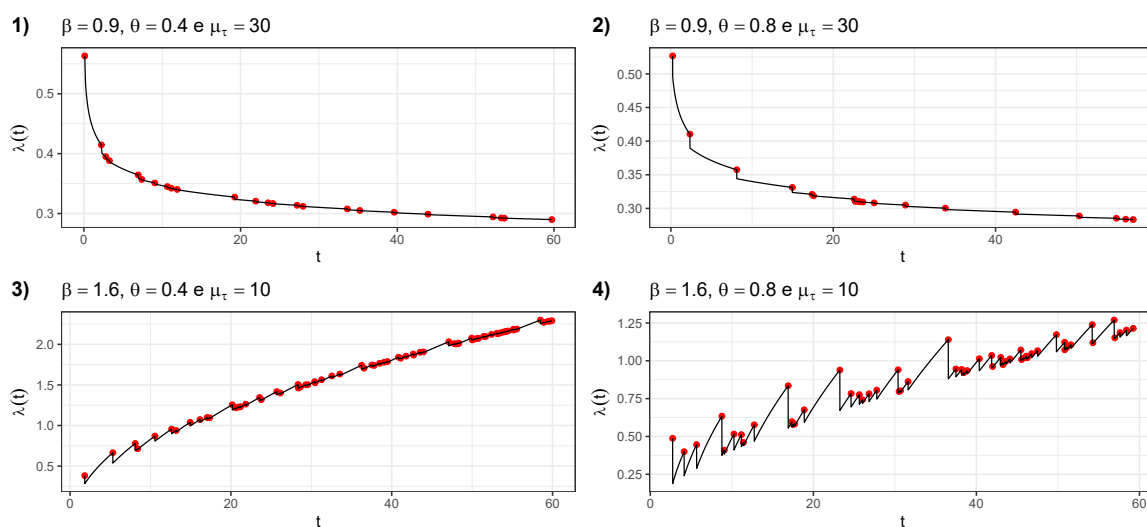


Figura 12 – Simulação de $\lambda_{ARAM_1}(t|H_{t-})$ para diferentes valores de β , μ_T e θ com $T = 60$.

3.4 Método de Estimação

Como estamos considerando que a parada das observações de falhas no sistema será por meio de um tempo predeterminado, denominado de tempo de truncamento e sabemos que esse tempo é conhecido pelo pesquisador e é dado pelo parâmetro conhecido T . Perceba que o número de falhas dada por $N(T)$ é aleatório. Para análise em dados reais, consideraremos que até T ocorreram n falhas nos tempos t_1, t_2, \dots, t_n . Objetivando estimar os parâmetros pelo método de máxima verossimilhança, a função de densidade conjunta de $(T_1, \dots, T_N, N(T))$ é dada por

$$\begin{aligned} L(\Theta|\mathbf{t}) &= f(t_1, \dots, t_n, n) = f(t_1, \dots, t_n)P(N(T) = n|t_1, \dots, t_n) \\ &= f(t_1)f(t_2|t_1) \cdots f(t_n|t_1, \dots, t_{n-1})P(N(T) = n|t_1, \dots, t_n) \\ &= f(t_1) \left[\prod_{i=1}^{n-1} f(t_{i+1}|t_i) \right] P(N(T) = n|t_n), \end{aligned} \quad (3.7)$$

em que $\Theta = (\beta, \theta, \mu_T)$ e $\mathbf{t} = (t_1, t_2, \dots, t_n)$. [Rigdon e Basu \(2000\)](#) demonstram que em um processo NHPP $f(t_{i+1}|t_i) = -\frac{d}{dt_{i+1}} \exp\left(-\int_{t_i}^{t_{i+1}} \lambda(x|H_{x-})dx\right)$, assim

$$\begin{aligned} f(t_{i+1}|t_i) &= -\frac{d}{dt_{i+1}} \exp\left(-\int_{t_i}^{t_{i+1}} \lambda_{ARAM_1}(x|H_{x-})dx\right) \\ &= -\frac{d}{dt_{i+1}} \exp\left(-\Lambda_{ARAM_1}(t_{i+1}|t_i)\right) \\ &= \lambda_{ARAM_1}(t_{i+1}|H_{t_{i+1}-}) \exp\left(-\Lambda_{ARAM_1}(t_{i+1}|t_i)\right), \end{aligned}$$

como consequência $f(t_1) = \lambda_{ARAM_1}(t_1|H_{t_1-}) \exp\left(-\Lambda_{ARAM_1}(t_1|t_0 = 0)\right)$, e, por fim, $P(N(T) = n|t_n) = P(N(t_n, T] = 0) = \exp\left(-\Lambda_{ARAM_1}(T|t_n)\right)$. Dessa forma, a Equação (3.7) fica

$$\begin{aligned} L(\Theta|\mathbf{t}) &= \lambda_{ARAM_1}(t_1|H_{t_1-}) \exp\left(-\Lambda_{ARAM_1}(t_1|t_0 = 0)\right) \\ &\times \left[\prod_{i=1}^{n-1} \lambda_{ARAM_1}(t_{i+1}|H_{t_{i+1}-}) \exp\left(-\Lambda_{ARAM_1}(t_{i+1}|t_i)\right) \right] \\ &\times \exp\left(-\Lambda_{ARAM_1}(T|t_n)\right) \\ &= \left[\prod_{i=1}^n \lambda_{ARAM_1}(t_i|H_{t_i-}) \right] \exp\left(-\sum_{i=1}^n \Lambda_{ARAM_1}(t_i|t_{i-1}) - \Lambda_{ARAM_1}(T|t_n)\right), \end{aligned}$$

em que $t_0 = 0$. Logo, a função de log-verossimilhança será dada por

$$\begin{aligned} \ell(\Theta|\mathbf{t}) &= \log\{L(\Theta|\mathbf{t})\} \\ &= \sum_{i=1}^n \log\{\lambda_{ARAM_1}(t_i|H_{t_i-})\} - \sum_{i=1}^n \Lambda_{ARAM_1}(t_i|t_{i-1}) - \Lambda_{ARAM_1}(T|t_n). \end{aligned} \quad (3.8)$$

Substituindo os resultados das expressões $\lambda_{ARAM_1}(t_i|H_{t_i-})$, $\Lambda_{ARAM_1}(t_i|t_{i-1})$ e $\Lambda_{ARAM_1}(T|t_n)$

na equação (3.8), obtemos

$$\begin{aligned} \ell(\Theta | \mathbf{t}) &= \sum_{i=1}^n \log \left\{ \frac{\mu_T \beta}{T} (t_i - \text{sign}(\beta - 1)h(\theta)t_{i-1})^{\beta-1} \right\} \\ &- \sum_{i=1}^n \frac{\mu_T}{T} \left[(t_i - \text{sign}(\beta - 1)h(\theta)t_{i-1})^\beta - (t_{i-1} - \text{sign}(\beta - 1)h(\theta)t_{i-1})^\beta \right] \\ &- \frac{\mu_T}{T} \left[(T - \text{sign}(\beta - 1)h(\theta)t_n)^\beta - (t_n - \text{sign}(\beta - 1)h(\theta)t_n)^\beta \right]. \end{aligned} \quad (3.9)$$

Portanto, o estimador de máxima verossimilhança $\hat{\Theta}_{mv}$ pode ser obtido computacionalmente, de modo que

$$\hat{\Theta}_{mv} = \underset{\Theta}{\text{argmax}} \{ \ell(\Theta | \mathbf{t}) \}.$$

Agora, para estimação dos parâmetros para múltiplos sistemas independentes e gerado de processos idênticos, fazemos

$$L(\Theta | \mathbf{t}^{(k)}) = \prod_{j=1}^k \left\{ \left[\prod_{i=1}^{n_j} \lambda_{ARAM_1}(t_{ij} | H_{t_{ij}-}) \right] \exp \left(- \sum_{i=1}^{n_j} \Lambda_{ARAM_1}(t_{ij} | t_{i(j-1)}) - \Lambda_{ARAM_1}(T | t_{n_j}) \right) \right\},$$

em que $\mathbf{t}^{(k)} = (\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_k)$ em que $\mathbf{t}_1 = (t_{11}, t_{21}, \dots, t_{n_11})$, $\mathbf{t}_2 = (t_{12}, t_{22}, \dots, t_{n_22})$, \dots , $\mathbf{t}_k = (t_{1k}, t_{2k}, \dots, t_{n_kk})$, k é o número de sistemas, t_{ij} é o tempo da i -ésima falha do j -ésimo sistema, n_j é o número de falhas do j -ésimo sistema e

$$\lambda_{ARAM_1}(t_{ij} | H_{t_{ij}-}) = \begin{cases} \frac{\mu_T \beta}{T} (t_{ij} - h(\theta)T_{N_{t_{ij}-}})^{\beta-1}, & \text{se } \beta \geq 1 \\ \frac{\mu_T \beta}{T} (t_{ij} + h(\theta)T_{N_{t_{ij}-}})^{\beta-1}, & \text{se } \beta < 1, \end{cases}$$

com $\Lambda_{ARAM_1}(t_{ij} | t_{i(j-1)}) = \int_{t_{i(j-1)}}^{t_{ij}} \lambda_{ARAM_1}(x | H_{x-}) dx$ e $\Lambda_{ARAM_1}(T | t_{n_j}) = \int_{t_{n_j}}^T \lambda_{ARAM_1}(x | H_{x-}) dx$.

Dessa forma, obtemos

$$\begin{aligned} \ell(\Theta | \mathbf{t}^{(k)}) &= \log \{ L(\Theta | \mathbf{t}^{(k)}) \} \\ &= \sum_{j=1}^k \left\{ \sum_{i=1}^{n_j} \log \{ \lambda_{ARAM_1}(t_{ij} | H_{t_{ij}-}) \} - \sum_{i=1}^{n_j} \Lambda_{ARAM_1}(t_{ij} | t_{i(j-1)}) - \Lambda_{ARAM_1}(T | t_{n_j}) \right\} \\ &= \sum_{j=1}^k \sum_{i=1}^{n_j} \log \{ \lambda_{ARAM_1}(t_{ij} | H_{t_{ij}-}) \} - \sum_{j=1}^k \sum_{i=1}^{n_j} \Lambda_{ARAM_1}(t_{ij} | t_{i(j-1)}) - \sum_{j=1}^k \Lambda_{ARAM_1}(T | t_{n_j}). \end{aligned}$$

Já considerando múltiplos parâmetros para sistemas reparáveis independentes, ou seja, considerando $\Theta^{(k)} = (\Theta_1, \Theta_2, \dots, \Theta_k)$, em que $\Theta_1 = (\beta_1, \theta_1, \mu_{T_1}^*)$, $\Theta_2 = (\beta_2, \theta_2, \mu_{T_2}^*)$, \dots , $\Theta_k = (\beta_k, \theta_k, \mu_{T_k}^*)$, chegamos a função de log-verossimilhança

$$\begin{aligned} \ell(\Theta^{(k)} | \mathbf{t}^{(k)}) &= \log \{ L(\Theta^{(k)} | \mathbf{t}^{(k)}) \} \\ &= \sum_{j=1}^k \left\{ \sum_{i=1}^{n_j} \log \{ \lambda_{ARAM_1}^{(j)}(t_{ij} | H_{t_{ij}-}) \} - \sum_{i=1}^{n_j} \Lambda_{ARAM_1}^{(j)}(t_{ij} | t_{i(j-1)}) - \Lambda_{ARAM_1}^{(j)}(T_j^* | t_{n_j}) \right\} \\ &= \sum_{j=1}^k \ell(\Theta_j | \mathbf{t}_j) \end{aligned} \quad (3.10)$$

em que T_1^*, \dots, T_k^* são os tempos de truncamento para os k sistemas, $\ell(\Theta_j | \mathbf{t}_j) = \sum_{i=1}^{n_j} \log \{ \lambda_{ARAM_1}^{(j)}(t_{ij} | H_{t_{ij}-}) \} - \sum_{i=1}^{n_j} \Lambda_{ARAM_1}^{(j)}(t_{ij} | t_{i(j-1)}) - \Lambda_{ARAM_1}^{(j)}(T_j^* | t_{n_j})$ e

$$\lambda_{ARAM_1}^{(j)}(t_{ij} | H_{t_{ij}-}) = \begin{cases} \frac{\mu_{j,T_j^*} \beta_j}{T_j^*} (t_{ij} - h(\theta_j) T_{N_{t_{ij}-}})^{\beta_j - 1}, & \text{se } \beta_j \geq 1 \\ \frac{\mu_{j,T_j^*} \beta_j}{T_j^*} (t_{ij} + h(\theta_j) T_{N_{t_{ij}-}})^{\beta_j - 1}, & \text{se } \beta_j < 1, \end{cases} \quad (3.11)$$

com

$$\Lambda_{ARAM_1}^{(j)}(t_{ij} | t_{i(j-1)}) = \int_{t_{i(j-1)}}^{t_{ij}} \lambda_{ARAM_1}^{(j)}(x | H_{x-}) dx \text{ e } \Lambda_{ARAM_1}^{(j)}(T_j^* | t_{n_j}) = \int_{t_{n_j}}^{T_j^*} \lambda_{ARAM_1}^{(j)}(x | H_{x-}) dx.$$

Portanto, para maximizar $\ell(\Theta^{(k)} | \mathbf{t}^{(k)})$ e obtermos as estimativas de máxima verossimilhança para $\Theta^{(k)}$, por (3.10), basta obter as estimativas de Θ_j que maximizam $\ell(\Theta_j | \mathbf{t}_j)$ para $j = 1, \dots, k$. Desse modo,

$$\begin{aligned} \hat{\Theta}_{mv}^{(k)} &= \operatorname{argmax}_{\Theta^{(k)}} \left\{ \ell(\Theta^{(k)} | \mathbf{t}^{(k)}) \right\} = \operatorname{argmax}_{\Theta^{(k)}} \left\{ \sum_{j=1}^k \ell(\Theta_j | \mathbf{t}_j) \right\} \\ &= \sum_{j=1}^k e_j \times \operatorname{argmax}_{\Theta_j} \left\{ \ell(\Theta_j | \mathbf{t}_j) \right\} \\ &= \sum_{j=1}^k e_j \times \Theta_{mv_j} = (\Theta_{mv_1}, \Theta_{mv_2}, \dots, \Theta_{mv_k}), \end{aligned} \quad (3.12)$$

em que $e_j = (\delta_1, \dots, \delta_i, \dots, \delta_k)$, com $\delta_i = 1$, se $i = j$, e $\delta_i = 0$, caso $i \neq j$, para $i = 1, \dots, k$.

Implementamos todas as etapas de estimação dos parâmetros do modelo ARAM₁ no ambiente R para programação estatística (R Core Team, 2021). As situações envolvendo as funções $\ell(\Theta | \mathbf{t})$, $\ell(\Theta | \mathbf{t}^{(k)})$ e $\ell(\Theta^{(k)} | \mathbf{t}^{(k)})$ serão denominadas neste trabalho, respectivamente, de simples (um sistema), composta (multissistema independentes e processo geradores de falha iguais) e multiparamétrica (multissistema independentes e processo geradores de falha diferentes).

3.5 Estudo de simulação

Objetivamos com o estudo de simulação avaliar a qualidade das estimativas pelo método de máxima verossimilhança em termos de viés (Bias) e erro quadrático médio (MSE). Esperamos obter viés próximo de zero e MSE decrescente com o aumento do tamanho amostral k , uma vez que são assintoticamente não viciados e eficientes. Utilizamos dois cenários para simulação, um em que sistemas estão se deteriorando (função de intensidade é crescente) e outro em que sistemas estão em processo de renovação (função de intensidade é decrescente), em termos de parametrização, temos cenário 1) modelo ARAM₁ com $\beta = 1, 1$, $\theta = 0, 3$ e $\mu_T = 10$ e cenário

2) uma função de intensidade decrescente com $\beta = 0,8$, $\theta = 0,6$ e $\mu_T = 50$. Consideramos para os dois cenários, respectivamente, tempo de truncamento $T = 500$ e $T = 50$.

Realizamos simulação de Monte Carlo com $m = 1000$ réplicas para os sistemas de tamanho $k = 5, 10, 30$ e 50 , considerando os dois cenários propostos. Para cada tamanho k foram obtidas $m = 1000$ réplicas das estimativas dos parâmetros pelo método de máxima verossimilhança, dessa forma, as réplicas formam o conjunto das estimativas e a partir delas foram calculadas as estimativas finais para o estudo de simulação. Abaixo o algoritmo utilizado para geração dos tempos de falha. Por exemplo, para obtermos uma amostra de estimativas de k sistemas, repetiremos o algoritmo abaixo $m = 1000$ vezes. Cada ciclo é um elemento do conjunto de estimativas.

- **Passo 1.** Estabelecer os valores iniciais dos parâmetros para o $ARAM_1$.
- **Passo 2.** Iniciar $t_0 = 0$ e $i = 0$.
- **Passo 3.** Gerar $x \sim F_i(x) = 1 - \exp\{-\Lambda_{ARAM_1}(t_i + x|t_i)\}$.
- **Passo 4.** Fazer $t_i \leftarrow t_{i-1} + x$ e $i \leftarrow i + 1$.
- **Passo 5.** Repetir o passo 3) e 4) enquanto $t_i < T$.

$F_i(x)$ é a função densidade acumulada da variável aleatória $X_i = T_{i+1} - T_i$, com $i = 1, \dots, n$, condicionada aos tempos de falha $T_i = t_i$ e $T_{i+1} = t_{i+1}$, assim,

$$\begin{aligned}
 F_i(x) &= P(X_i \leq x | T_i = t_i, T_{i+1} = t_{i+1}) \\
 &= P(N(t_i, t_i + x] \geq 1) = 1 - P(N(t_i, t_i + x] = 0) \\
 &= 1 - \exp\left\{-\int_{t_i}^{t_i+x} \lambda_{ARAM_1}(t|H_{t-}) dt\right\} \\
 &= 1 - \exp\{-\Lambda_{ARAM_1}(t_i + x|t_i)\}.
 \end{aligned}$$

Como $F_i(X)$ tem distribuição uniforme $U(0, 1)$, então, para obter um x pseudo-aleatório, gera-se $u_i \sim U(0, 1)$ e resolve-se x numericamente na equação $F_i(x) = u_i = 1 - \exp\{-\Lambda_{ARAM_1}(t_i + x|t_i)\}$ ou, equivalentemente, resolver numericamente a seguinte equação

$$\Lambda_{ARAM_1}(t_i + x|t_i) = -\log(1 - u_i).$$

Implementamos todas as etapas de simulação no ambiente R para programação estatística (R Core Team, 2021). Nas otimizações, tanto para obtenção das integrais numéricas quanto para maximizar a função de log-verossimilhança, usamos a função `optimr(.)` do pacote `optimr` (NASH; VARADHAN; GROTHENDIECK, 2019), com método L-BFGS-B que permite definir o espaço paramétrico na otimização. Para análise da qualidade das estimativas, utilizamos o MSE, que é uma medida frequentemente utilizada em estudos de simulação para quantificar a

qualidade das estimativas, assim como o viés (Bias) da estimativa. Por exemplo, o viés e o erro quadrático médio para as estimativas de θ_k são dados, respectivamente, por

$$Bias(\hat{\theta}_k) = \frac{\sum_{i=1}^m (\hat{\theta}_{ik} - \theta)}{m} \text{ e } MSE(\hat{\theta}_k) = \frac{\sum_{i=1}^m (\hat{\theta}_{ik} - \theta)^2}{m},$$

em que $\hat{\theta}_{ik}$ é a i -ésima estimativa para θ com total de sistemas igual a k . O mesmo processo se repete para as estimativas de β_k e μ_{kT_k} .

Na Tabela 1 apresentamos os valores do viés e MSE das estimativas pelo método de máxima verossimilhança para os dois cenários simulados. Nas Figura 13 e Figura 14 apresentamos os resultados gráficos da tendência do viés e MSE para os cenários 1 e 2, respectivamente, considerando os valores de $k = 5, 10, 30, 50$. Podemos notar que nos dois cenários as estimativas para os parâmetros β , μ_T e θ estão bastante próximos de seus respectivos valores predefinidos, uma vez que os vieses tendem para zero com o aumento de k , especialmente nos casos em que $k = 50$.

No cenário 1, percebemos que os vieses obtidos foram todos negativos, exceto para $\mu_T = 10$ com $k = 30$ e $k = 50$, e notamos uma tendência em direção ao zero, como desejado. Já no cenário 2, os vieses obtidos foram todos positivos, com decaimento em direção ao zero com o crescimento de k , exceto, por aleatoriedade, o viés do estimador de μ_T inicia com valor extremamente baixo. Analisando o MSE, podemos verificar visualmente na Figura 13 e na Figura 14, que que nos cenários apresentados, para as estimativas dos três parâmetros do modelo $ARAM_1$, quanto maior o tamanho da amostra k considerado, os valores do viés e MSE decrescem. Percebemos também que o MSE para as estimativas de θ nos dois cenários apresentados, decrescem lentamente em comparação às estimativas dos outros parâmetros.

Assim, concluímos do estudo de simulação que os estimadores de máxima verossimilhança para os parâmetros do modelo $ARAM_1$ possuem boas estimativas, que se aproximam de zero com aumento de k , em conformidade com a propriedade assintótica dos estimadores de máxima verossimilhança, o MSE, em quase todos os casos, reduz com o aumento de k , refletindo propriedades frequentistas esperadas e desejáveis.

Tabela 1 – Resultado da simulação para os dois cenários propostos, caso composto por múltiplos sistemas.

Cenário	k	$Bias(\hat{\beta})$	$MSE(\hat{\beta})$	$Bias(\hat{\theta})$	$MSE(\hat{\theta})$	$Bias(\hat{\mu}_T)$	$MSE(\hat{\mu}_T)$
$\beta = 1, 1, \theta = 0, 3 \text{ e } \mu_T = 10$							
1	5	-0,0453	0,0310	-0,191	0,231	-2,75	155,0
	10	-0,0249	0,0132	-0,103	0,189	-1,25	71,7
	30	-0,0176	0,0041	-0,050	0,150	0,07	12,0
	50	-0,0137	0,0023	-0,028	0,133	0,21	5,5
$\beta = 0, 8, \theta = 0, 6 \text{ e } \mu_T = 50$							
2	5	0,0650	0,0213	0,1180	0,2520	0,0060	575,0
	10	0,0385	0,0120	0,0707	0,2160	1,5800	276,0
	30	0,0138	0,0040	0,0276	0,1740	1,0600	122,0
	50	0,0062	0,0017	0,0027	0,1500	0,8130	75,5

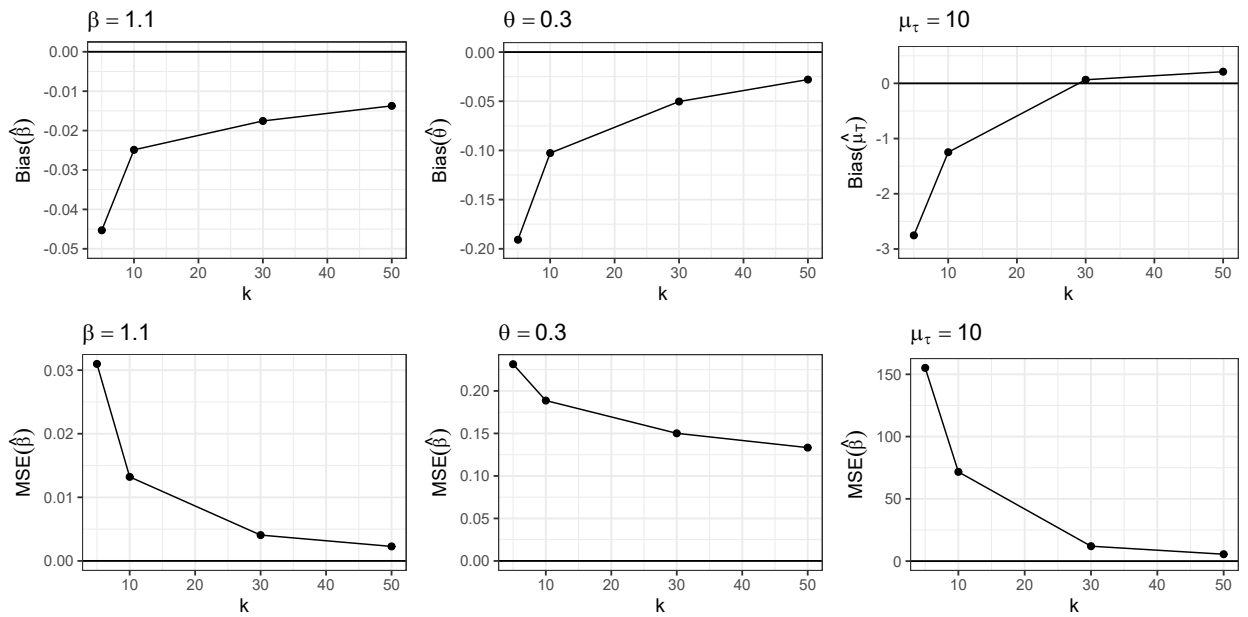


Figura 13 – Cenário 1: resultado do viés e MSE da simulação para o modelo ARAM_1 com $T = 500$.

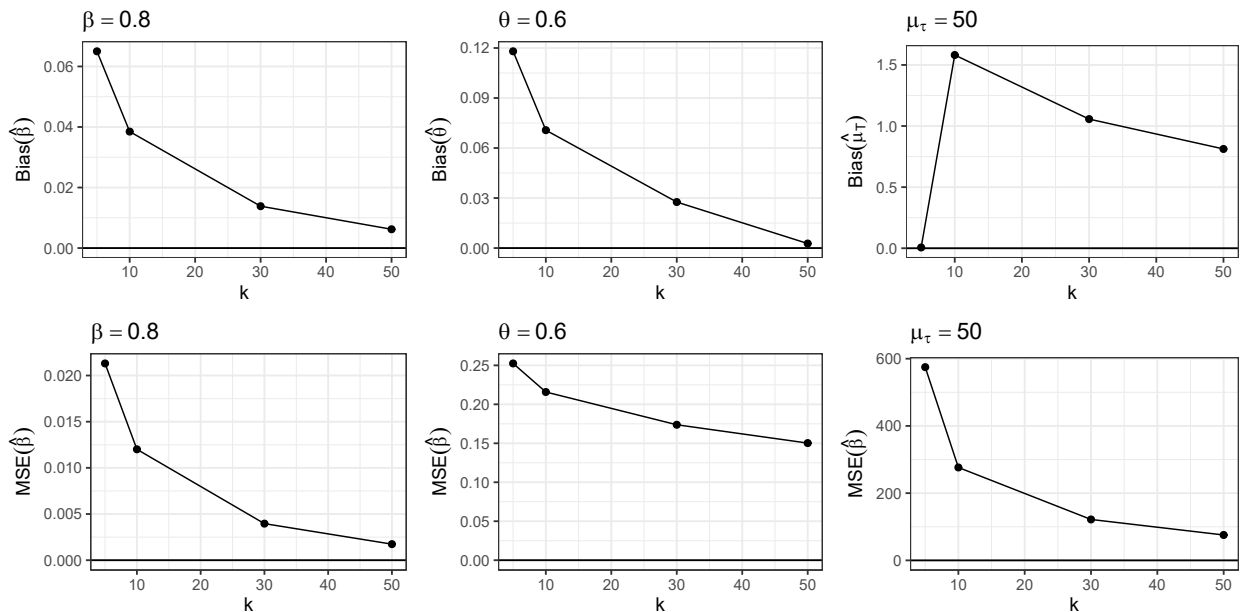


Figura 14 – Cenário 2: resultado do viés e MSE da simulação para o modelo ARAM_1 com $T = 50$.

3.6 Aplicação a dados reais

Para aplicação do novo modelo ARAM_1 , foram utilizados os tempos de falha, em dias, de lâmina *Chopper* localizada no sistema de corte de cana-de-açúcar em nove colheitadeiras. Os dados são oriundos do resultado da safra 2014/2015 de cana-de-açúcar nos estados de São Paulo e Paraná, Brasil. Durante o processo da colheita, a lâmina corta a cana em pedaços menores e,

em seguida, são levados para a usina. É certo que falhas nas lâminas *Chopper* provocam perdas financeiras, dessa forma, entender a situação das colheitadeiras assim como o impacto dos reparos tanto em níveis individuais quanto globalmente é de extrema relevância para continuidade das atividades, visto que é possível, por meio do modelo $ARAM_1$, identificar as que estão em estado de deterioração ou melhoria no decorrer das atividades, assim como a informação da qualidade dos reparos realizados.

Os parâmetros do modelo serão estimados por meio do método de máxima verossimilhança e para obtenção dos intervalos de confiança dos parâmetros estimados $\hat{\mu}_T$ e $\hat{\beta}$ serão utilizadas as estimativas dos erros padrões (SE, do inglês *standard error*), em que serão tomados os valores da diagonal principal resultante do inverso da matriz de informação observada. Usando esses valores, os intervalos de confiança de 95% (CI 95%) dos parâmetros estimados são obtidos da seguinte maneira:

$$\text{CI 95\% para } \hat{\beta} : (\hat{\beta} - 1,96\text{SE}(\hat{\beta}), \hat{\beta} + 1,96\text{SE}(\hat{\beta})),$$

em que $\text{SE}(\hat{\beta})$ é o erro padrão de $\hat{\beta}$, da mesma forma obtemos o CI 95% para $\hat{\mu}_T$.

Para o parâmetro θ abordaremos a estimativa pontual, dado que é limitado inferiormente e superiormente, dificultando a obtenção dos intervalos de confiança assintóticos, uma vez que frequentemente os limites do intervalo podem ficar fora da cobertura do espaço paramétrico de θ . Uma solução é a obtenção do intervalo de credibilidade para θ , abordagem Bayesiana, não tratada neste trabalho.

Na Figura 15 podemos notar que o total de falhas variam de $\min = 11$ a $\max = 19$ por sistema/colheitadeira. A observação dos tempos de falhas foi truncada, em cada sistema, em $T = 200$ dias (linha azul tracejada). A Tabela 2 fornece os valores das estimativas dos parâmetros $\hat{\Theta}^{(9)}$ para o caso multiparamétrico pela maximização da log-verossimilhança $\ell(\hat{\Theta}^{(9)} | \mathbf{t}^{(9)})$ dada em (3.10) para o modelo $ARAM_1$. Tais maximizações foram realizadas usando-se o *software* R. Notamos que os sistemas apresentam comportamentos diversos em termos de intensidade de falha.

Foi estimado que todos os sistemas estão em situação de degradação $\hat{\beta}_j > 1$, exceto o de nº 8, que possui $\hat{\beta}_8 = 0,87$, indicando uma melhora da colheitadeira em termos de falhas na lâmina *Chopper* ao longo do tempo até $T = 200$. No mesmo sistema, esperava-se, aproximadamente, 113 falhas ($\hat{\mu}_{T_j} = 112,69$) se não houvesse influência dos reparos próximos do perfeito ($\hat{\theta}_8 = 0,99$) e de seu estado de melhoria ($\hat{\beta}_8 < 1$), em que observou-se somente 13 falhas até $T = 200$. Já para a colheitadeira de cana-de-açúcar nº 1, apesar do número baixo, o processo possui o maior valor $\hat{\beta}_j$, igual a 2,42, com intervalo de confiança de 95% igual a (1,31, 3,52), portanto, o sistema está, significativamente, em estado de deterioração. Esperava-se menos de uma falha ao longo do tempo observado ($\hat{\mu}_{T_1} = 0,14$), caso o sistema não estivesse se deteriorando e sem contar com o efeito do reparo. A colheitadeira de cana-de-açúcar nº 2 é o caso em que os reparos realizados foram prejudiciais, uma vez que $\hat{\theta}_2 = -0,94$ (Tabela 2).

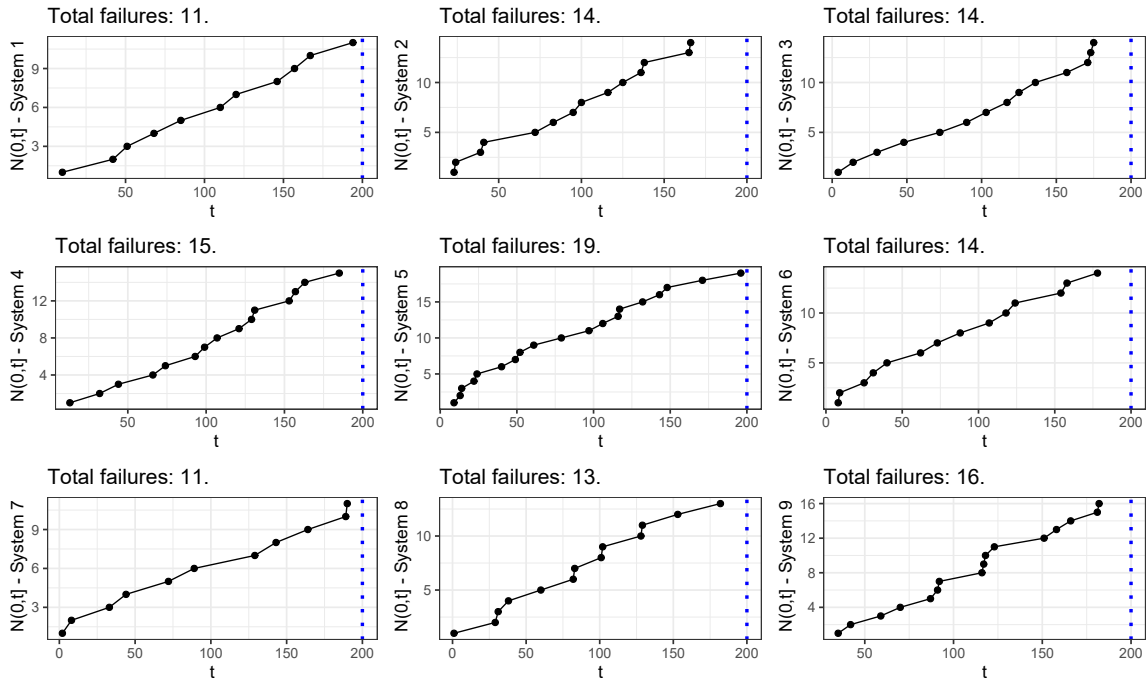


Figura 15 – Total de falhas acumuladas observadas nas nove colheitadeiras até $T = 200$.

Tabela 2 – Estimativas dos parâmetros do modelo $ARAM_1$ na situação multiparamétrica, arredondadas em duas casas decimais, com $k = 9$.

Sistema (j)	n_j	$\hat{\beta}_j$ (95% CI)	$\hat{\mu}_{T_j}$ (95% CI)	$\hat{\theta}_j$	$\ell(\hat{\Theta}_j \mathbf{t}_j)$
1	11	2,42 (1,31, 3,52)	0,14 (0,00, 0,65)	1,00	-37,98
2	14	1,03 (0,86, 1,21)	8,59 (0,00, 28,79)	-0,94	-51,11
3	14	2,01 (0,94, 3,08)	0,66 (0,00, 2,96)	0,98	-48,57
4	15	2,35 (1,10, 3,59)	0,26 (0,00, 1,33)	0,97	-49,90
5	19	1,45 (0,92, 1,97)	5,88 (0,00, 14,59)	1,00	-62,03
6	14	1,59 (0,90, 2,28)	2,58 (0,00, 7,91)	1,00	-49,39
7	11	1,48 (0,64, 2,31)	2,37 (0,00, 9,12)	0,99	-42,09
8	13	0,87 (0,71, 1,03)	112,69 (0,00, 411,89)	0,99	-47,43
9	16	1,65 (0,44, 2,86)	0,89 (0,00, 4,9)	0,65	-55,35
$\ell(\hat{\Theta}^{(9)} \mathbf{t}^{(9)}) = \sum_{j=1}^9 \ell(\hat{\Theta}_j \mathbf{t}_j)$					-443,8

Nas estimativas considerando o modelo $ARAM_1$ na situação composta, ou seja, considerando que os múltiplos sistemas possuem as mesmas funções de intensidade de falhas, percebemos que indicam que os sistemas estão, significativamente, se deteriorando, mesmo com reparos perfeitos realizados. A estimativa $\hat{\mu}_T = 5,57$ indica que esperava-se, aproximadamente, seis falhas ao longo do tempo observado, caso os sistemas não estivesse se deteriorando e sem

contar com os efeitos dos reparos. Já a média de falhas por sistema é 14,11, pouco mais que o dobro de $\hat{\mu}_T$ (Tabela 8). Em termos de log-verossimilhança estimada, o modelo $ARAM_1$ na situação multiparamétrica, por considerar a especificidade de cada sistema, se ajusta melhor aos tempos de falhas comparativamente ao modelo $ARAM_1$ na situação composta, uma vez que $\ell(\hat{\Theta}^{(9)} | \mathbf{t}^{(9)}) > \ell(\hat{\Theta} | \mathbf{t}^{(9)})$. Se considerássemos o ARA_1 para modelar o processo de falha do sistema 8, teríamos $\hat{\beta} = 1$ e $\mu_T = 13$, sem influenciaria de θ no processo, dado que $\hat{\beta} = 1$ e $\ell(\hat{\Theta}_8 | \mathbf{t}_8) = -48,53$, conseqüentemente, na situação multiparamétrica, obteríamos $\ell(\hat{\Theta}^{(9)} | \mathbf{t}^{(9)}) = -444,9$, dado que $\hat{\beta}_j \geq 1$, para $j \neq 8$. Portanto, o ARA_1 modificado é a melhor alternativa de ajuste em termos de $\ell(\hat{\Theta}^{(9)} | \mathbf{t}^{(9)})$. Dado que o objetivo, para esta aplicação, é analisar o comportamento de múltiplos sistemas em termos das suas situações e efeitos do reparo, não se considerou para análise da qualidade dos ajustes a penalização pelo quantitativo de parâmetros, como é o caso das medidas AIC e BIC.

Tabela 3 – Estimativas dos parâmetros do modelo $ARAM_1$ na situação composta, arredondadas em duas casas decimais.

k	n	n/k	$\hat{\beta}$ (95% CI)	$\hat{\mu}_T$ (95% CI)	$\hat{\theta}$	$\ell(\hat{\Theta} \mathbf{t}^{(9)})$
9	127	14,11	1,32 (1,13, 1,51)	5,57 (2,30, 8,84)	1,00	-457,36

A Figura 16 representa o número observado de falhas acumuladas (linha vermelha contínua, DR) e a intensidade cumulativa estimada (linha azul pontilhada, MA) do modelo $ARAM_1$ para os dados de tempos de falha de lâmina *Chopper* no sistema de corte das nove colheitadeiras. Vemos que o ajuste é razoavelmente bom, uma vez que as linhas estão próximas uma da outra. Por fim, na Figura 17 temos as funções de intensidade estimadas. O efeito do reparo nos sistemas com $\hat{\theta}_j$ iguais a um ou próximo de um, reduz a função de intensidade para zero ou próximo de zero imediatamente após falha, como vemos nos sistemas 1), 3), 4), 5), 6) e 7).

Como o efeito estimado do reparo para o sistema 2) foi negativo, então os reparos realizados foram prejudiciais, assim, a intensidade aumenta imediatamente após falha, como visto graficamente. Como foi estimado que o sistema 8) está em processo de renovação com efeito positivo do reparo, então a intensidade decresce imediatamente após falha. Já para sistema 9) foi estimado que está se deteriorando com $\hat{\theta}_9 = 0,65$, assim, a intensidade decresce, indicado efeito positivo dos reparos realizados (Figura 17).

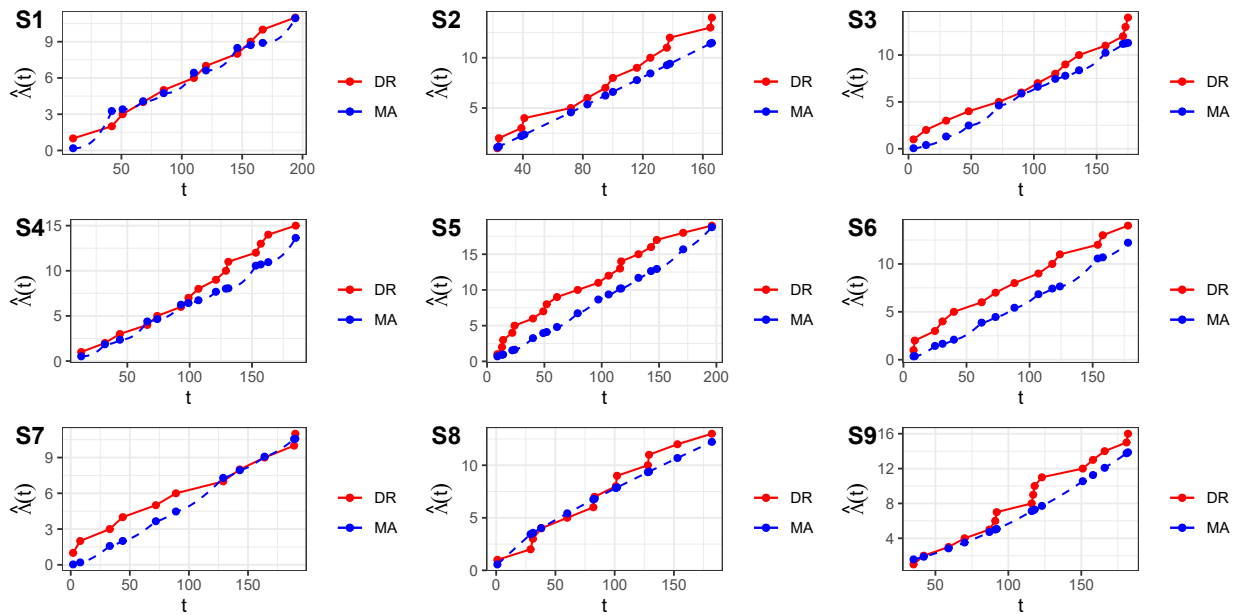


Figura 16 – Dados Reais (DR) de acúmulo de falhas pela expectativa de falhas modelada (MA) pelo $ARAM_1$ para as nove colheitadeiras até $T = 200$.

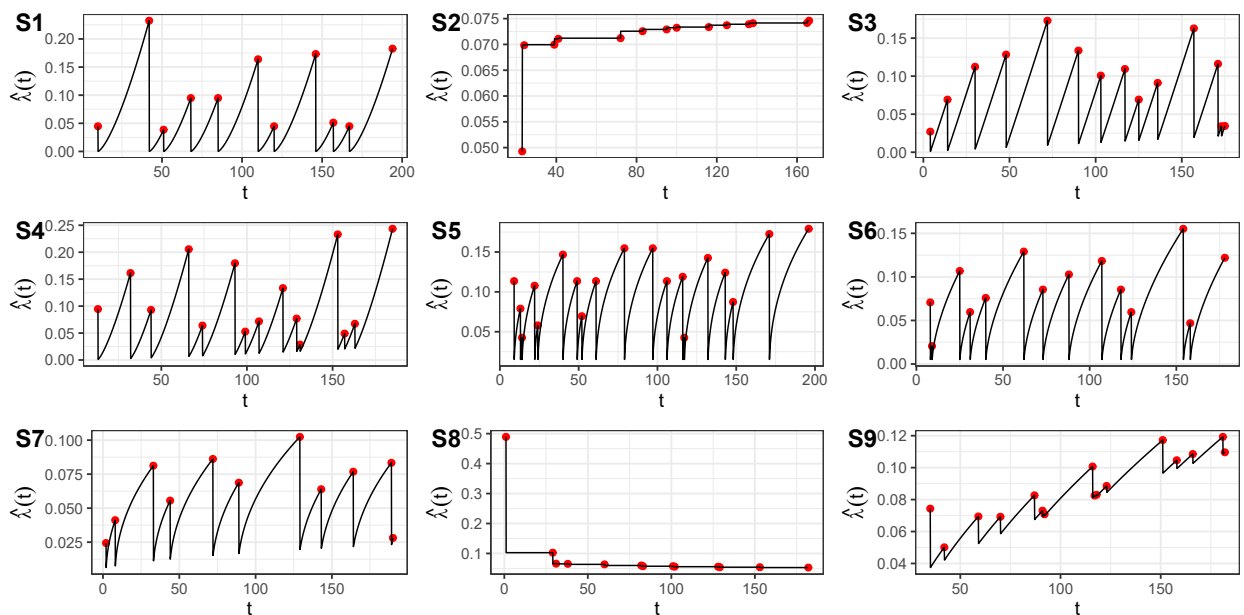


Figura 17 – Intensidade $ARAM_1$ estimada para as nove colheitadeiras até $T = 200$.

3.7 Considerações finais

Neste trabalho, apresentamos uma generalização do modelo ARA_1 com PLP, em que adicionamos a situação na qual o sistema reparável possui intensidade decrescente, ou seja, o caso cujo o sistema está melhorando com o tempo, denominamos essa nova modelagem de ARA_1 modificado ($ARAM_1$). Também propomos uma nova reparametrização do PLP como

truncamento por tempo para incorporá-lo ao novo modelo e assim preservar a interpretação original dos parâmetros do PLP. Para tal, consideramos que o efeito do reparo θ varia de -1 a 1 , quando $-1 \leq \theta < 0$ o reparo é considerado prejudicial, $0 < \theta < 1$ como eficiente e ótimo/perfeito quando $\theta = 1$.

Utilizamos o método clássico de estimação por máxima verossimilhança para obter as estimativas do modelo $ARAM_1$ nas simulações e aplicação. No estudo de simulação, concluímos, com os resultados obtidos nos dois cenários propostos, que as propriedades assintóticas dos estimadores de máxima verossimilhança estão de acordo com o esperado, assintoticamente não viciado e MSE decaindo para zero com o aumento do tamanho amostral.

Na aplicação em dados de tempos de falhas em nove colheitadeiras de cana-de-açúcar, concluímos que o novo modelo captura diversas situações não possível pelo tradicional ARA_1 , pois após os parâmetros do modelo $ARAM_1$ serem estimados, observamos, no caso multiparamétrico, que a colheitadeira n° 8 melhora com o tempo. Foi observado também um efeito prejudicial dos reparos na colheitadeira n° 2. Por fim, esperamos que o novo modelo proposto $ARAM_1$ forneça uma alternativa relevante aos habituais modelos para sistemas reparáveis existentes.

PROCESSO LEI DE POTÊNCIA GENERALIZADO PARA SISTEMAS REPARÁVEIS

4.1 Introdução

Neste capítulo propomos um PLP generalizado (PLPG) baseado em pontos de mudança, que podem ser tratados como tempos conhecidos ou desconhecidos em que houve mudanças externas ao sistema ou interpretá-los como os tempos de falha, em que consideramos o impacto de todos ou algumas reparos sobre função de intensidade de falha. Dessa maneira, o PLPG não se restringe a suposição de RM. Diferentemente do PLP, é possível considerar outras situações, como reparo perfeito, eficiente e prejudicial. A partir do PLPG é possível derivar os principais modelos com ponto de mudança, assim como os principais modelos para análise de sistemas com reparo imperfeito.

São casos particulares do PLPG: PLP investigado por [Crow \(1974\)](#), HPP com um ponto de mudança postulado por [West e Ogden \(1997\)](#), PLPCP de [Richardson e Basu \(2004\)](#), o PLP com múltiplos pontos de mudança de [Syamsundar e Naikan \(2008\)](#), modelos para reparo mínimo, reparo perfeito, modelo ARA_1 proposto por [Doyen e Gaudoin \(2004\)](#), modelo ARA_1 modificado ($ARAM_1$), dentro outras novas modelagens propostas denominadas de reparos completamente imperfeitos (RCI) e reparos parcialmente imperfeitos com p pontos de mudanças ($RPI(p)$).

A vantagem do uso do PLPG é que amplia as aplicações de modelos de sistemas reparáveis em dados reais de tempos de falha em um único modelo, abrangendo os principais modelos da literatura, além de considerar novas possibilidades, como RCI e $RPI(p)$, ampliando ainda mais sua utilização. Podemos destacar também que PLPG permite que a intensidade após os reparos não permaneça paralela à intensidade inicial, diferentemente do que ocorre com ARA_1 , que, entre duas falhas consecutivas, a intensidade da falha é horizontalmente paralela à

intensidade inicial.

O capítulo está organizado da seguinte forma. Nas Seções 4.2 e 4.3, apresentamos, respectivamente, o novo processo PLP generalizado (PLPG) e o método de estimação de seus parâmetros. Em seguida, na Seção 4.4, avaliamos a performance dos estimadores dos parâmetros através do estudo de simulação de Monte Carlo (MC). Na Seção 4.5 apresentamos duas aplicações do PLPG em dados reais de tempos de falha, o primeiro conjunto de tempos de falha utilizado para modelagem é o das falhas de um exaustor para vapor, com tempo de operação em número de dias e o segundo conjunto de tempos de falha utilizado para modelagem é o das falhas do sistema de ar-condicionado de um avião a jato Boeing 720 n° série 7908, com tempo de funcionamento em horas. A Seção 4.6 resume as principais conclusões do capítulo.

4.2 O PLP generalizado (PLPG)

Nesta seção vamos definir o PLPG, interpretar seus parâmetros e os efeitos dos pontos de mudança, derivar a função de intensidade acumulada até um certo tempo t , obter os casos particulares do novo processo, apresentar novos modelos para análise de sistemas reparáveis como consequência do PLPG e simular cenários com função de intensidade de falha do PLPG, além de representá-los graficamente.

Definição 7. O Processo Lei de Potência Generalizado (PLPG) é um Processo Poisson Não Homogêneo com função de intensidade de falha definida como

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i), \quad (4.1)$$

em que p é o número de mudanças no processo, $0 = \tau_0 < \tau_1 < \dots < \tau_p < \tau_{p+1} = T$, $T > 0$, $\mu_T > 0$, $\beta_{i-1} > 0$, $\theta_{i-1} \in \mathbb{R}$, $\text{sign}(\beta_{i-1} - 1) = 1$, caso $\beta_{i-1} \geq 1$ e $\text{sign}(\beta_{i-1} - 1) = -1$, caso $\beta_i < 1$ e $\mathbf{I}(\tau_{i-1} < t \leq \tau_i)$ é função indicadora, para $i = 1, \dots, p+1$. Caso $t \leq 0$, consideraremos $\lambda_{PLPG}(t) = 0$. As funções $g(\theta_{i-1})$ e $h(\theta_{i-1})$ deverão satisfazer as condições:

1. são contínuas e não decrescentes em θ_{i-1} ;
2. para $\beta_{i-1} \leq 1$, $h(\theta_{i-1}) \geq -1$ e para $\beta_{i-1} \geq 1$, $h(\theta_{i-1}) \leq 1$;
3. $g(\theta_{i-1}) \geq -1$, $g(0) = 0$ e $h(0) = 0$.

Considere o caso em que θ_{i-1} possua espaço paramétrico em $[-1, 1]$ e quando $h(\theta_{i-1})$ não for explicitada, vamos definir a função $h(\theta_{i-1})$, estritamente crescente, como:

- 1) Para $\beta_{i-1} \geq 1$, $h(\theta_{i-1}) : (-1, 1] \rightarrow (-\infty, 1]$, tal que

$$h(\theta_{i-1}) = \begin{cases} \log\left(\frac{1+\theta_{i-1}}{1-\theta_{i-1}}\right) 10^k, & \text{se } -1 < \theta_{i-1} \leq 0, \\ \theta_{i-1}, & \text{se } 0 \leq \theta_{i-1} \leq 1, \end{cases}$$

2) Para $\beta_{i-1} < 1$, $h(\theta_{i-1}) : [-1, 1) \rightarrow [-1, \infty)$, tal que

$$h(\theta_{i-1}) = \begin{cases} \theta_{i-1}, & \text{se } -1 \leq \theta_{i-1} \leq 0, \\ \log\left(\frac{1+\theta_{i-1}}{1-\theta_{i-1}}\right) 10^k, & \text{se } 0 \leq \theta_{i-1} < 1, \end{cases}$$

em que $k = 1, 2, \dots$ é uma constante.

Por fim, quando a função $g(\theta_{i-1})$ não for explicitada, tomaremos $g(\theta_{i-1}) = \theta_{i-1}$. Contudo, $g(\theta_{i-1})$ e $h(\theta_{i-1})$ podem ser definidas de outras formas, por exemplo, podemos tomar $g(\theta_{i-1})$ como

$$g(\theta_{i-1}) = \begin{cases} \frac{1 - e^{|\theta_{i-1}|}}{e - 1}, & \text{se } -1 \leq \theta_{i-1} \leq 0, \\ \frac{e^{\theta_{i-1}} - 1}{e - 1}, & \text{se } 0 \leq \theta_{i-1} \leq 1, \end{cases}$$

que implica em efeitos mais suaves de θ_{i-1} sobre $g(\theta_{i-1})$. Inclusive, as funções $g(\theta_{i-1})$ e $h(\theta_{i-1})$ podem ser nulas, ou seja, $g(\theta_{i-1}) = 0$ e $h(\theta_{i-1}) = 0$ para todo θ_{i-1} .

Para ilustrar o comportamento do PLPG em relação aos seus parâmetros, vamos considerar três cenários: 1) PLPG sem pontos de mudança; 2) PLPG com um ponto de mudança; e PLPG com dois pontos de mudança.

1. PLPG sem ponto de mudança ($p = 0$) com $\beta_0 = 1.4$, $\theta_0 = 0.5$, $\mu_T = 10$ e $T = 30$:

$$\begin{aligned} \lambda_{PLPG}(t) &= \sum_{i=1}^{0+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1} - 1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i), \\ &= \frac{\mu_T \beta_0}{T + g(\theta_0) \tau_0} [t - \text{sign}(\beta_0 - 1) h(\theta_0) \tau_0]^{\beta_0 - 1} \mathbf{I}(\tau_0 < t \leq \tau_1) \\ &= \frac{10 \times 1.4}{30 + 0.5 \times 0} [t - \text{sign}(1.4 - 1) \times 0.5 \times 0]^{1.4 - 1} \mathbf{I}(0 < t \leq 30) \\ &= \frac{14t^{0.4}}{30} \mathbf{I}(0 < t \leq 30). \end{aligned}$$

2. PLPG com um ponto de mudança ($p = 1$), em que $\beta_0 = 1.4$, $\theta_0 = 0.5$, $\beta_1 = 1$, $\theta_1 = 0.5$, $\mu_T = 10$, $T = 30$ e $\tau_1 = 10$:

$$\begin{aligned}\lambda_{PLPG}(t) &= \sum_{i=1}^{1+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1)h(\theta_{i-1})\tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i), \\ &= \frac{\mu_T \beta_0}{T + g(\theta_0) \tau_0} [t - \text{sign}(\beta_0 - 1)h(\theta_0)\tau_0]^{\beta_0-1} \mathbf{I}(\tau_0 < t \leq \tau_1) + \\ &+ \frac{\mu_T \beta_1}{T + g(\theta_1) \tau_1} [t - \text{sign}(\beta_1 - 1)h(\theta_1)\tau_1]^{\beta_1-1} \mathbf{I}(\tau_1 < t \leq \tau_2). \\ &= \frac{14t^{0.4}}{30} \mathbf{I}(0 < t \leq 10) + \frac{10}{35} \mathbf{I}(10 < t \leq 30).\end{aligned}$$

3. PLPG com dois pontos de mudança ($p = 2$), em que $\beta_0 = 1.4$, $\theta_0 = 0.5$, $\beta_1 = 1$, $\theta_1 = 0.5$, $\beta_2 = 0.9$, $\theta_2 = -0.9$, $\mu_T = 10$, $T = 30$, $\tau_1 = 10$ e $\tau_2 = 20$.

$$\begin{aligned}\lambda_{PLPG}(t) &= \sum_{i=1}^{2+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1)h(\theta_{i-1})\tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i), \\ &= \frac{\mu_T \beta_0}{T + g(\theta_0) \tau_0} [t - \text{sign}(\beta_0 - 1)h(\theta_0)\tau_0]^{\beta_0-1} \mathbf{I}(\tau_0 < t \leq \tau_1) + \\ &+ \frac{\mu_T \beta_1}{T + g(\theta_1) \tau_1} [t - \text{sign}(\beta_1 - 1)h(\theta_1)\tau_1]^{\beta_1-1} \mathbf{I}(\tau_1 < t \leq \tau_2) + \\ &+ \frac{\mu_T \beta_2}{T + g(\theta_2) \tau_2} [t - \text{sign}(\beta_2 - 1)h(\theta_2)\tau_2]^{\beta_2-1} \mathbf{I}(\tau_2 < t \leq \tau_3) \\ &= \frac{14t^{0.4}}{30} \mathbf{I}(0 < t \leq 10) + \frac{10}{35} \mathbf{I}(10 < t \leq 20) + \frac{9(t-18)^{-0.1}}{12} \mathbf{I}(20 < t \leq 30).\end{aligned}$$

Na Figura 18 obtemos as representações gráficas dos três cenários do PLPG, em que vemos os efeitos dos pontos de mudança sobre a função de intensidade $\lambda_{PLPG}(t)$. Os pontos cheios representam a intensidade de falha no instante da mudança e imediatamente após os pontos abertos inicia-se a intensidade mudada.

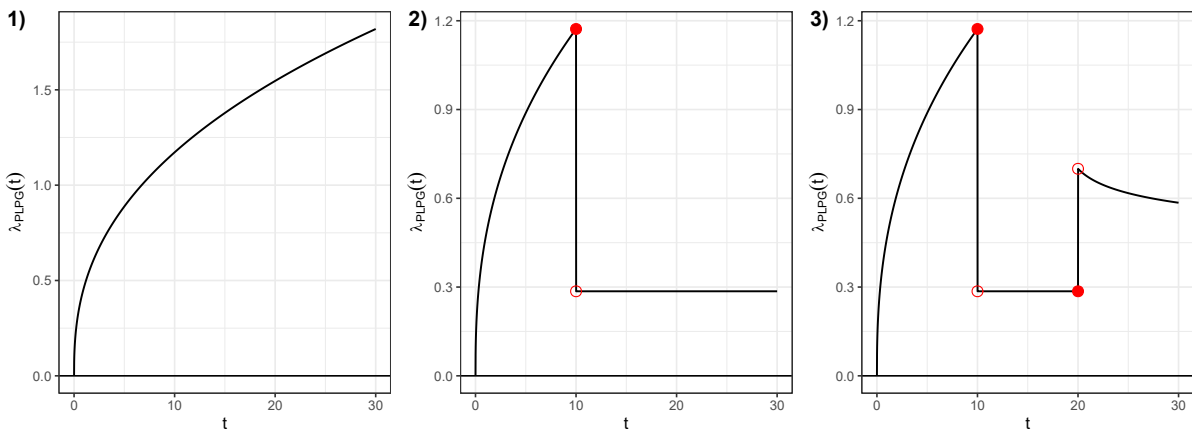


Figura 18 – Função de intensidade do PLPG para os três cenários propostos.

4.2.1 Interpretação dos parâmetros do modelo PLPG

O Processo de Lei de Potência Generalizado (PLPG), cuja função de intensidade $\lambda_{PLPG}(t)$ é dada na Definição 4.1, captura as alterações no processo por meio dos tempos em que ocorrem as mudanças, dada por τ_1, \dots, τ_p , que podem ser interpretados como parâmetros conhecidos, desconhecidos ou como tempos observáveis de falhas, cujos efeitos em $\lambda_{PLPG}(t)$ são dados por θ_{i-1} através das funções $h(\theta_{i-1})$ e $g(\theta_{i-1})$.

A partir da Definição 4.1 podemos concluir que o efeito da mudança na intensidade de falha ocorrerá imediatamente após o tempo de mudança. Por exemplo, quando τ_{i-1} for interpretado como tempo observável de falha, o intervalo de tempo entre τ_{i-1} e o reparo é tão pequeno quanto quisermos, contudo não é considerado nulo, uma vez que primeiro ocorre a falha e posteriormente o reparo, isto é, não ocorrem simultaneamente.

O parâmetro predefinido T , tempo máximo de t , trunca o tempo de observações dos eventos do PLPG, logo, neste trabalho daremos ênfase ao estudo do PLPG com truncamento por tempo T . No entanto, os casos de truncamento por falha também podem ser estudados com a seguinte reparametrização $\alpha_{i-1} = \mu_T / (T + g(\theta_{i-1})\tau_{i-1})$, contudo $\lambda_{PLPG}(t)$ não dependeria mais de $g(\theta_{i-1})$.

Como definido, $h(\theta_{i-1})$ é estritamente crescente em relação ao parâmetro θ_{i-1} , isto implica que aumento em θ_{i-1} provoca redução em $\lambda_{PLPG}(t)$ por meio da função $h(\theta_{i-1})$. A função de intensidade de falha não depende do ponto de mudança τ_{i-1} quando $\theta_{i-1} = 0$, contudo, valores negativos impactam em aumento de $\lambda_{PLPG}(t)$ e valores positivos acarretam redução, mantendo-se os demais parâmetros constantes. O parâmetro β_{i-1} é de forma, caracteriza a monotonicidade do PLPG entre os tempos de mudança. A função $\lambda_{PLPG}(t)$ decresce se $\beta_{i-1} < 1$, é constante se $\beta_{i-1} = 1$ e cresce para $\beta_{i-1} > 1$, quando $\tau_{i-1} < t \leq \tau_i$.

Ressaltamos que $h(\theta_{i-1})$ é condicionada ao valor de β_{i-1} , se $\beta_{i-1} > 1$ então $\lambda_{PLPG}(t)$ é crescente e $\text{sign}(\beta_{i-1} - 1) = 1$, desse modo, para $0 \leq \theta_{i-1} \leq 1$ faz com que o efeito $h(\theta_{i-1})$ seja de redução de $\lambda_{PLPG}(t)$. Na mesma condição, para $-1 < \theta_{i-1} \leq 0$, o efeito é de aumento de $\lambda_{PLPG}(t)$. Se $\beta_{i-1} < 1$ então $\lambda_{PLPG}(t)$ é decrescente e $\text{sign}(\beta_{i-1} - 1) = -1$, assim, segue que, para $0 \leq \theta_{i-1} \leq 1$, o efeito $h(\theta_{i-1})$ é de redução de $\lambda_{PLPG}(t)$. Na mesma condição, para $-1 < \theta_{i-1} \leq 0$, o efeito é de aumento de $\lambda_{PLPG}(t)$.

A função $g(\theta_{i-1})$, não nula, influencia na escala do PLPG junto aos parâmetros μ_T e T . Como $-1 \leq g(\theta_{i-1}) \leq 1$, então $T + g(\theta_{i-1})\tau_{i-1} > 0$, para todo θ_{i-1} . Se $-1 \leq g(\theta_{i-1}) < 0$, então provoca aumento da razão $\mu_T \beta_{i-1} / (T + g(\theta_{i-1})\tau_{i-1})$ a medida que θ_{i-1} se aproxima de -1 e, conseqüentemente, impacta no aumento de $\lambda_{PLPG}(t)$. De modo análogo, se θ_{i-1} se aproxima de 1 , então $g(\theta_{i-1})$ também se aproxima de 1 , impactando na redução de $\lambda_{PLPG}(t)$. Dessa maneira, $g(\theta_{i-1})$ é a função do efeito do ponto de mudança sobre a escala de $\lambda_{PLPG}(t)$, quanto maior θ_{i-1} , maior é o efeito da função $g(\theta_{i-1})$, portanto, menor é a escala do PLPG. Já μ_T é parâmetro não indexado pela $i - 1$ -ésima mudança, capturando o estado geral do processo, estendendo ou

reduzindo a escala do PLPG com seu aumento ou decrescimento, respectivamente. O parâmetro μ_T representa a média do PLPG sem a influência da situação do sistema ($\beta_{i-1} = 1$) e dos efeitos dos reparos ($\theta_{i-1} = 0$), para todo i , assim

$$\begin{aligned}\Lambda_{PLPG}(T) &= \int_0^T \lambda_{PLPG}(x) dx = \int_0^T \sum_{i=1}^{p+1} \frac{\mu_T}{T} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\ &= \frac{\mu_T}{T} \int_0^T \sum_{i=1}^{p+1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx = \frac{\mu_T}{T} \int_0^T dx = \mu_T\end{aligned}$$

uma vez que $\sum_{i=1}^{p+1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) = 1$. Nessa situação, $\lambda_{PLPG}(t)$ pode ser interpretada como velocidade instantânea de falha, pois

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T}{T} \mathbf{I}(\tau_{i-1} < t \leq \tau_i) = \frac{\mu_T}{T} \mathbf{I}(0 < t \leq T),$$

em que μ_T/T é a velocidade instantânea de falha em t , dado que o número esperado de falhas até t é dado por $\Lambda_{PLPG}(t) = \int_0^t \lambda_{PLPG}(x) dx = (\mu_T/T)t$.

O fator 10^k em $h(\theta_{i-1})$, para $k = 1, 2, \dots$, calibra a intensidade do efeito negativo do ponto de mudança ($\beta_{i-1} \geq 1$ e $-1 < \theta_{i-1} \leq 0$) ou efeito positivo ($\beta_{i-1} < 1$ e $0 \leq \theta_{i-1} < 1$), o que permite ressaltar seu efeito na função de intensidade, neste trabalho adotaremos $k = 4$, mas simulações podem ser realizadas no processo de estimação dos parâmetros do PLPG, como estratégia para selecionar qual constante k considerar, por exemplo, a constante $k \in \mathbb{N}$ que maximiza a função de log-verossimilhança em uma amostra de tempos.

Considere agora que $\beta_{i-1} = \beta$ e $\theta_{i-1} = \theta$ para todo $i = 1, \dots, p+1$. Assim, θ é responsável pela descontinuidade de $\lambda_{PLPG}(t)$ nos pontos de mudança, uma vez que, para $0 < \theta \leq 1$, $\lim_{t \rightarrow \tau_{i-1}^+} \lambda_{PLPG}(t) < \lim_{t \rightarrow \tau_{i-1}^-} \lambda_{PLPG}(t)$ e para $-1 \leq \theta < 0$, $\lim_{t \rightarrow \tau_{i-1}^+} \lambda_{PLPG}(t) > \lim_{t \rightarrow \tau_{i-1}^-} \lambda_{PLPG}(t)$, contudo, $\lambda_{PLPG}(t)$ é contínua no tempo de mudança τ_{i-1} quando $\theta = 0$, dessa forma, $\lim_{t \rightarrow \tau_{i-1}^+} \lambda_{PLPG}(t) = \lim_{t \rightarrow \tau_{i-1}^-} \lambda_{PLPG}(t) = \lambda_{PLPG}(\tau_{i-1})$, ou seja, não há efeito dos pontos de mudança em $\lambda_{PLPG}(t)$ quando $\theta = 0$.

Em resumo, vimos que o parâmetro β_{i-1} representa a situação do sistema, para $\beta_{i-1} > 1$ significa que o sistema está deteriorando, para $\beta_{i-1} < 1$ o sistema está melhorando ao longo do tempo e $\beta_{i-1} = 1$ significa que o sistema se mantém constante com o tempo. O parâmetro θ_{i-1} é interpretado como o efeito do reparo no sistema, $h(\theta_{i-1})$ é interpretada como a função de redução ou aumento da idade do sistema por conta do efeito θ_{i-1} e $g(\theta_{i-1})$ é a função de aceleração ou desaceleração do número médio de falhas até T . Por fim, μ_T é interpretado como a média de falhas até T , sem a influência da situação do sistema e dos efeitos dos reparos por meio de $g(\theta_{i-1})$.

A função de intensidade acumulada do Processo Poisson Não Homogêneo com intensidade PLPG é dada por

$$\begin{aligned}
\Lambda_{PLPG}(t) &= \int_0^t \lambda_{PLPG}(x) dx \\
&= \int_0^t \sum_{i=1}^{p+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\
&= \int_0^t \sum_{i=1}^{k(t)} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\
&\quad + \int_0^t \sum_{i=k(t)+1}^{p+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\
&= \sum_{i=1}^{k(t)} \int_0^t \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\
&= \sum_{i=1}^{k(t)-1} \int_0^t \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < x \leq \tau_i) dx \\
&\quad + \int_{\tau_{k(t)-1}}^t \frac{\mu_T \beta_{k(t)-1}}{T + g(\theta_{k(t)-1}) \tau_{k(t)-1}} [x - \text{sign}(\beta_{k(t)-1} - 1) h(\theta_{k(t)-1}) \tau_{k(t)-1}]^{\beta_{k(t)-1}-1} dx \\
&= \sum_{i=1}^{k(t)-1} \int_{\tau_{i-1}}^{\tau_i} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} dx \\
&\quad + \int_{\tau_{k(t)-1}}^t \frac{\mu_T \beta_{k(t)-1}}{T + g(\theta_{k(t)-1}) \tau_{k(t)-1}} [x - \text{sign}(\beta_{k(t)-1} - 1) h(\theta_{k(t)-1}) \tau_{k(t)-1}]^{\beta_{k(t)-1}-1} dx \\
&= \sum_{i=1}^{k(t)-1} \Lambda_{PLPG}(\tau_i | \tau_{i-1}) + \Lambda_{PLPG}(t | \tau_{k(t)-1}),
\end{aligned}$$

em que $k(t) = \{j; \tau_{j-1} < t \leq \tau_j, j = 1, \dots, p+1\}$ e que

$$\begin{aligned}
\Lambda_{PLPG}(\tau_i | \tau_{i-1}) &= \int_{\tau_{i-1}}^{\tau_i} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} dx \\
&= \frac{\mu_T [(\tau_i - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}} - (\tau_{i-1} - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}}]}{T + g(\theta_{i-1}) \tau_{i-1}},
\end{aligned}$$

e também

$$\begin{aligned}
\Lambda_{PLPG}(t | \tau_{k(t)-1}) &= \int_{\tau_{k(t)-1}}^t \frac{\mu_T \beta_{i-1}}{T + g(\theta_i) \tau_{k(t)-1}} [x - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{k(t)-1}]^{\beta_{i-1}-1} dx \\
&= \frac{\mu_T [(t - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{k(t)-1})^{\beta_{i-1}} - (\tau_{k(t)-1} - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{k(t)-1})^{\beta_{i-1}}]}{T + g(\theta_{i-1}) \tau_{k(t)-1}}.
\end{aligned}$$

Assim, chegamos ao seguinte resultado

$$\begin{aligned}
\Lambda_{PLPG}(t) &= \sum_{i=1}^{k(t)-1} \left[\frac{\mu_T [(\tau_i - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}} - (\tau_{i-1} - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}}]}{T + g(\theta_{i-1}) \tau_{i-1}} \right] \\
&\quad + \frac{\mu_T [(t - \text{sign}(\beta_{k(t)-1} - 1) h(\theta_{k(t)-1}) \tau_{k(t)-1})^{\beta_{k(t)-1}} - (\tau_{k(t)-1} - \text{sign}(\beta_{k(t)-1} - 1) h(\theta_{k(t)-1}) \tau_{k(t)-1})^{\beta_{k(t)-1}}]}{T + g(\theta_{k(t)-1}) \tau_{k(t)-1}}.
\end{aligned}$$

Como caso particular, temos que a média do processo PLPG no intervalo $(0, T]$ que é dada por

$$\begin{aligned}
\Lambda_{PLPG}(T) &= \sum_{i=1}^p \left[\frac{\mu_T [(\tau_i - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}} - (\tau_{i-1} - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1})^{\beta_{i-1}}]}{T + g(\theta_{i-1}) \tau_{i-1}} \right] \\
&\quad + \frac{\mu_T [(T - \text{sign}(\beta_p - 1) h(\theta_p) \tau_p)^{\beta_p} - (\tau_p - \text{sign}(\beta_p - 1) h(\theta_p) \tau_p)^{\beta_p}]}{T + g(\theta_p) \tau_p}. \tag{4.2}
\end{aligned}$$

uma vez que $k(T) = p + 1$, pois $k(T) = \{j; \tau_{j-1} < T \leq \tau_j, j = 1, \dots, p + 1\}$, como foi definido que $T = \tau_{p+1}$, então $\tau_p < T \leq \tau_{p+1}$, logo $k(T) = p + 1$.

Considere agora que os efeitos dos pontos de mudança e as situações do sistema sejam constantes e iguais à θ e β , respectivamente. Assim, a classificação do efeito dos pontos de mudança será interpretado de acordo com o valor de θ , cujas expressões da função de intensidade do PLPG varia de acordo com o estado do sistema, expresso por β , logo temos:

- 1) com $\theta = 0$ e $\beta > 0$, o efeito é mínimo para sistema em estado de degradação, constância ou melhoria em que obtemos

$$\begin{aligned}\lambda_{PLPG}(t) &= \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T} t^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i) \\ &= \frac{\mu_T \beta}{T} t^{\beta-1} \mathbf{I}(0 < t \leq T).\end{aligned}$$

- 2) com $\theta = 1$ e $\beta \geq 1$, o efeito é perfeito em sistema em estado de degradação ou constância e temos que

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T + \tau_{i-1}} [t - \tau_{i-1}]^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i).$$

- 3) com $0 < \theta < 1$ e $\beta \geq 1$, o efeito é dito eficiente em sistema em estado de degradação ou constância e a função é dada por

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T + \theta \tau_{i-1}} [t - \theta \tau_{i-1}]^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i).$$

- 4) com $-1 < \theta < 0$ e $\beta \geq 1$, o efeito é prejudicial em sistema em estado de degradação ou constância com

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T + \theta \tau_{i-1}} [t - h(\theta) \tau_{i-1}]^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i).$$

- 5) com $0 < \theta < 1$ e $\beta < 1$, o efeito é eficiente em sistema em estado de melhoria e

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T + \theta \tau_{i-1}} [t + h(\theta) \tau_{i-1}]^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i).$$

- 6) com $-1 \leq \theta < 0$ e $\beta < 1$ o efeito é prejudicial em sistema em estado de melhoria e

$$\lambda_{PLPG}(t) = \sum_{i=1}^{p+1} \frac{\mu_T \beta}{T + \theta \tau_{i-1}} [t + \theta \tau_{i-1}]^{\beta-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i)$$

Através do PLPG é possível derivar vários modelos para análise de sistemas reparáveis, a seguir, vamos obter seus casos particulares, considerando como os pontos de mudança como parâmetros desconhecidos, conhecidos ou considerando como tempos observados em que ocorreram as falhas no sistema. Dessa maneira, o PLPG pode ser visto como uma classe de modelos paramétricos para análise de sistemas reparáveis.

4.2.2 Casos Particulares

Primeiro vamos considerar os casos particulares em que os pontos de mudança são interpretados como parâmetros desconhecidos ou conhecidos.

- **Processo de Lei de Potência - PLP (P4)**

O PLP investigado por [Crow \(1974\)](#) é o caso particular mais simples do PLPG, uma vez que no PLP não há pontos de mudança, então (4.1), para $p = 0$, fica

$$\begin{aligned}\lambda_{PLP}(t) &= \sum_{i=1}^1 \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1) h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i) \\ &= \frac{\mu_T \beta_0}{T + g(\theta_0) \tau_0} [t - \text{sign}(\beta_0 - 1) h(\theta_0) \tau_0]^{\beta_0-1} \mathbf{I}(\tau_0 < t \leq \tau_1) \\ &= \frac{\mu_T \beta_0 t^{\beta_0-1}}{T} \mathbf{I}(0 < t \leq T),\end{aligned}$$

uma vez que $\tau_0 = 0$, $\tau_{p+1} = \tau_1 = T$ e $\lambda_{PLPG}(t) = 0$ para $t \leq 0$.

- **HPP com um ponto de mudança**

Para o PLPG com um único ponto de mudança ($p = 1$) temos que sua função de intensidade de falha é dada por

$$\lambda_{PLPG}(t) = \begin{cases} \frac{\mu_T \beta_0}{T} t^{\beta_0-1}, & \text{se } 0 < t \leq \tau_1, \\ \frac{\mu_T \beta_1}{T + g(\theta_1) \tau_1} [t - \text{sign}(\beta_1 - 1) h(\theta_1) \tau_1]^{\beta_1-1}, & \text{se } \tau_1 < t \leq T, \end{cases} \quad (4.3)$$

como consequência do caso para $p = 1$ temos um HPP com um ponto de mudança postulado por [West e Ogden \(1997\)](#):

$$\lambda_{WP}(t) = \begin{cases} \frac{\mu_T}{T}, & \text{se } 0 < t \leq \tau_1, \\ \frac{\mu_T}{T + g(\theta_1) \tau_1}, & \text{se } \tau_1 < t \leq T, \end{cases}$$

fazendo $\beta_0 = \beta_1 = 1$ em (4.3), com parâmetro desconhecido τ_1 . Originalmente, $\lambda_{WP}(t)$ é dada em função de dois parâmetros, θ_0^* e θ_1^* , em que $\mu_T/T = \theta_0^*$ e $\mu_T/(T + g(\theta_1) \tau_1) = \theta_1^*$.

- **Processo Lei de Potência com um ponto de mudança - PLPCP**

Já o processo o PLP com ponto de mudança (PLPCP) de [Richardson e Basu \(2004\)](#) pode ser obtido com $\beta_1 = 1$ em (4.3), assim

$$\lambda_{RB}(t) = \begin{cases} \frac{\mu_T \beta_0}{T} t^{\beta_0 - 1}, & \text{se } 0 < t \leq \tau_1, \\ \frac{\mu_T}{T + g(\theta_1) \tau_1}, & \text{se } \tau_1 < t \leq T, \end{cases}$$

em que τ_1 é conhecido. Originalmente, [Richardson e Basu \(2004\)](#) utilizam λ e γ na parametrização da função de intensidade, em que $\mu_T/T = \lambda$ e $\mu_T/(T + g(\theta_1)\tau_1) = \gamma$.

- **Processo Lei de Potência com múltiplos pontos de mudança**

O PLP com múltiplos pontos de mudança de [Syamsundar e Naikan \(2008\)](#) é obtido fazendo $\beta_{i-1} \geq 1$ e $\theta_{i-1} = 1$ em (4.1), isto é,

$$\lambda_{SN}(t) = \sum_{i=1}^{p+1} [\mu_T/(T + \tau_{i-1})] \beta_{i-1} [t - \tau_{i-1}]^{\beta_{i-1} - 1} I(\tau_{i-1} < t \leq \tau_i),$$

em que τ_{i-1} são parâmetros desconhecidos e, originalmente, [Syamsundar e Naikan \(2008\)](#) utilizam o parâmetro α_{i-1} na modelagem da função de intensidade, tal que $[\mu_T/(T + \tau_{i-1})] = \alpha_{i-1}$.

4.3 Modelo PLPG para sistemas reparáveis

Agora, considerando que os tempos de mudança são os tempos observados em que ocorreram as falhas, isto é, $\tau_i = t_i$, para todo $i = 1 \dots, p$ e $p = n$, em que n é o número de falhas até T e t_i é o tempo observado da i -ésima falha, logo, nessa situação, τ_i não é interpretado como parâmetro e sim como valor observável dos tempos de falha.

Assim, considerando que o sistema permanece no mesmo estado (renovação ou melhoria) e que o efeito do reparo seja o mesmo, isto é, $\beta_i = \beta$ e $\theta_i = \theta$, então teremos os seguintes casos particulares do PLPG:

- **Processo de Poisson Homogêneo (HPP)**

Caso mais simples do PLPG, para obtenção do HPP fazemos $\theta = 0$ e $\beta = 1$, obtemos, portanto:

$$\lambda_C(t) = \frac{\mu_T}{T} I(0 < t \leq T),$$

indicando que a velocidade instantânea de falha é constante ao longo do tempo t .

- **Modelo de Reparo mínimo (RM)**

A parametrização do PLPG para obtenção do modelo para sistemas reparáveis com reparo mínimo (RM), ou seja, não há redução da intensidade de falha provocado pelo efeito do reparo θ , é dada com $\theta = 0$, obtemos, portanto:

$$\begin{aligned}\lambda_{RM}(t) &= \sum_{i=1}^{n+1} \frac{\mu_T}{T} \beta t^{\beta-1} \mathbf{I}(t_{i-1} < t \leq t_i), \\ &= \frac{\mu_T}{T} \beta t^{\beta-1} \mathbf{I}(0 < t \leq T),\end{aligned}$$

em que $\beta \geq 1$, $t_0 = 0$, $t_{n+1} = T$ e $\sum_{i=1}^{n+1} \mathbf{I}(t_{i-1} < t \leq t_i) = 1 = \mathbf{I}(0 < t \leq T)$. Perceba que a função de intensidade do modelo RM é a intensidade do PLP ($\lambda_{PLP}(t)$), isso significa que se o efeito $\theta_i = 0$ em todos os pontos de mudança do PLPG, então é equivalente a não ter pontos de mudança em termos de função de intensidade, ou seja, $p = 0$, uma vez que todos os efeitos são nulos.

- **Modelo de Reparo Perfeito (RP)**

Para $\theta = 1$ e definindo $g(\theta) = 0$ para todo θ no PLPG, obtemos o modelo para sistemas reparáveis com reparo perfeito (RP), assim:

$$\lambda_{RP}(t|H_{t-}) = \sum_{i=1}^{n+1} \frac{\mu_T}{T} \beta (t - t_{i-1})^{\beta-1} \mathbf{I}(t_{i-1} < t \leq t_i),$$

em que $\beta \geq 1$, $t_0 = 0$, $t_{n+1} = T$ e H_{t-} é o histórico de falhas imediatamente anterior à t .

- **Modelo de Reparo Imperfeito (ARA₁)**

A parametrização do PLPG para obtenção do modelo ARA₁ é dada com $0 < \theta < 1$ e definimos $g(\theta) = 0$ para todo θ , e assim obtemos:

$$\lambda_{ARA_1}(t|H_{t-}) = \sum_{i=1}^{n+1} \frac{\mu_T}{T} \beta (t - \theta t_{i-1})^{\beta-1} \mathbf{I}(t_{i-1} < t \leq t_i),$$

em que $\beta \geq 1$, $t_0 = 0$, $t_{n+1} = T$.

4.3.1 Novos modelos para reparos imperfeitos com PLPG

O modelo ARA₁, proposto por [Doyen e Gaudoin \(2004\)](#), tem sido amplamente utilizado na literatura. Contudo, se restringe a sistema que se desgasta continuamente entre falhas, portanto, o modelo é restrito à $\beta \geq 1$ no PLP, não considerando, portanto, sistemas que se renovam entre falhas, ou seja, com função de intensidade decrescente, no PLP, $0 < \beta < 1$. O uso do PLPG soluciona essa limitação, proporcionando mais possibilidades de modelagem, como é o caso do ARA₁ modificado (ARAM₁), definido abaixo:

- **Modelo ARA1 modificado (ARAM₁)**

Já a parametrização do PLPG para se chegar ao modelo ARAM₁ é dada por $\beta > 0$ e $-1 \leq \theta \leq 1$, assim, obtemos:

$$\lambda_{ARAM_1}(t|H_{t-}) = \sum_{i=1}^{n+1} \frac{\mu_T}{T} \beta [t - \text{sign}(\beta - 1)h(\theta)t_{i-1}]^{\beta-1} \mathbf{I}(t_{i-1} < t \leq t_i),$$

em que $\beta > 1$, $t_0 = 0$, $t_{n+1} = T$ e definindo $g(\theta) = 0$ para todo θ .

Porém, os modelos apresentados acima ainda sofrem com a desvantagem de que a intensidade após os reparos permanece paralela à intensidade inicial, segundo [Doyen e Gaudoin \(2004\)](#) o ARA tem intensidade horizontalmente paralela. O PLPG resolve essa limitação do paralelismo, uma vez que agrega à intensidade de falha $\lambda_{PLPG}(t)$ a aceleração ou desaceleração do número médio de falha a cada tempo de falha por meio da função $g(\theta)$, de acordo com o efeito θ , tornando-a mais geral e ampliando a modelagem para dados de tempos de falha.

- **Modelo de reparos completamente imperfeitos (RCI)**

Denominaremos o modelo de reparos completamente imperfeitos (RCI), cuja função de intensidade é dada por

$$\lambda_{RCI}(t|H_{t-}) = \sum_{i=1}^{n+1} \frac{\mu_T}{T + g(\theta)t_{i-1}} \beta [t - \text{sign}(\beta - 1)h(\theta)t_{i-1}]^{\beta-1} \mathbf{I}(t_{i-1} < t \leq t_i),$$

em que $\beta > 0$, $-1 \leq \theta \leq 1$, $t_0 = 0$, $t_{n+1} = T$, caso $g(\theta) = 0$ para todo θ , então o RCI dependerá de θ por meio de $h(\theta)$ e recai no ARAM₁.

- **Reparo Completamente Imperfeito em função de $g(\cdot)$ - (RCI_g)**

Se $h(\theta) = 0$ para todo θ , então o RCI dependerá de θ por meio da $g(\theta)$, e assim obtemos mais um caso particular, denominado de RCI_g dada por

$$\lambda_{RCI_g}(t|H_{t-}) = \sum_{i=1}^{n+1} \frac{\mu_T \beta t^{\beta-1}}{T + g(\theta)t_{i-1}} \mathbf{I}(t_{i-1} < t \leq t_i).$$

Nesse modelo, o impacto dos efeitos do reparo é somente sobre a velocidade de falha $\mu_T/(T + g(\theta)t_{i-1})$ da função de intensidade $\lambda_{RCI_g}(t|H_{t-})$.

Interpretamos nos novos modelos para sistemas reparáveis apresentados acima que os pontos de mudanças são todos os tempos de falha, contudo, podemos interpretar que alguns pontos em que houve reparos foram de mudança em relação à última intensidade de falha, e os demais não promovem mudança, dessa maneira, obtemos novos modelos para análise de sistemas reparáveis com uso do PLPG. O caso mais simples é dada em (4.3), porém τ_1 é o tempo de falha que modificou a função de intensidade com reparo.

Por exemplo, supondo que a função possua o mesmo parâmetro de forma, isto é, $\beta_0 = \beta_1 = \beta$, assim (4.3), ficaria

$$\lambda_{RPI}(t|H_{t-}) = \begin{cases} \frac{\mu_T \beta}{T} t^{\beta-1}, & \text{se } 0 < t \leq \tau_1, \\ \frac{\mu_T \beta}{T + g(\theta) \tau_1} [t - \text{sign}(\beta - 1)h(\theta) \tau_1]^{\beta-1}, & \text{se } \tau_1 < t \leq T, \end{cases} \quad (4.4)$$

em que $\tau_1 \in \{t_1, \dots, t_n\}$.

Podemos considerar o modelo (4.4) como um modelo de reparo parcialmente imperfeito com um ponto de mudança, vamos chamar de RPI(1), pois somente um reparo provocou mudança na intensidade de falha com efeito θ . Uma generalização do modelo (4.4) para p pontos de mudança será expressa abaixo.

- **Modelos de Reparos Parcialmente Imperfeitos com p pontos de mudança - RPI(p)**

A função de intensidade do modelo RPI(p) é dada por

$$\lambda_{RPI}(t|H_{t-}) = \sum_{i=1}^{p+1} \frac{\mu_T \beta_{i-1}}{T + g(\theta_{i-1}) \tau_{i-1}} [t - \text{sign}(\beta_{i-1} - 1)h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i),$$

em que RPI(p) será a nomenclatura dada para reparo parcialmente imperfeito com p pontos de mudança, com $p \leq n$, e $\{\tau_1, \dots, \tau_p\} \subset \{t_1, \dots, t_n\}$ são os tempos em que ocorreram os reparos e provocaram mudanças na função de intensidade. Perceba que alguns ou todos os tempos de falha são pontos de mudança na função de intensidade. O modelo RPI(p) é uma aplicação do PLPG na modelagem de sistemas reparáveis em que os pontos de mudanças τ_i são tempos observáveis de falha que impactam em $\lambda_{RPI}(t|H_{t-})$, portanto, nesta interpretação, τ_i são selecionados e não parâmetros a ser estimados.

Os casos especiais do RPI(p) são dados quando RPI(p) depende somente de $h(\theta_i)$ que denominaremos de RPI _{h} (p) e quando o RPI(p) depende somente $g(\theta_i)$ que denominaremos de RPI _{g} (p). As funções de intensidade $\lambda_{RPI_h}(t|H_{t-})$ e $\lambda_{RPI_g}(t|H_{t-})$ são dadas por, respectivamente,

- **RPI(p) em função de $h(\cdot)$ - RPI _{h} (p)**

$$\lambda_{RPI_h}(t|H_{t-}) = \sum_{i=1}^{p+1} \frac{\mu_T \beta_{i-1}}{T} [t - \text{sign}(\beta_{i-1} - 1)h(\theta_{i-1}) \tau_{i-1}]^{\beta_{i-1}-1} \mathbf{I}(\tau_{i-1} < t \leq \tau_i).$$

- **RPI(p) em função de $g(\cdot)$ - RPI _{g} (p)**

$$\lambda_{RPI_g}(t|H_{t-}) = \sum_{i=1}^{p+1} \frac{\mu_T \beta_{i-1} t^{\beta_{i-1}-1}}{T + g(\theta_{i-1}) \tau_{i-1}} \mathbf{I}(\tau_{i-1} < t \leq \tau_i),$$

em que $\{\tau_1, \dots, \tau_p\} \subset \{t_1, \dots, t_n\}$ com $p \leq n$.

Na Tabela 4 apresentamos um resumo dos casos particulares da nova classe de modelos PLPG apresentado nesta seção. Perceba que o modelo RPI(p) é o modelo RCI quando consideramos $p = n$.

Tabela 4 – Resumo dos casos particulares do PLPG.

p	β_{i-1}	θ_{i-1}	$h(\theta_{i-1})$	$g(\theta_{i-1})$	$\lambda(t)$
$p = 0$	$\beta_0 = 1$	-	-	-	λ_C
$p = 0$	$\beta_0 > 0$	-	-	-	$\lambda_{PLP}(t)$
$p = 1$	$\beta_0 = \beta_1 = 1$	$-1 \leq \theta_1 = \theta \leq 1$	-	$g(\theta_1) \geq -1$	$\lambda_{WP}(t)$
$p = 1$	$\beta_0 > 0, \beta_1 = 1$	$-1 \leq \theta_1 = \theta \leq 1$	-	$g(\theta_1) \geq -1$	$\lambda_{RB}(t)$
$p = 0, 1, \dots$	$\beta_{i-1} \geq 1$	$\theta_{i-1} = 1$	$h(\theta_{i-1}) = 1$	$g(\theta_{i-1}) = 1$	$\lambda_{SN}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta > 0$	$\theta_{i-1} = 0$	$h(\theta_{i-1}) = 0$	$g(\theta_{i-1}) = 0$	$\lambda_{RM}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta \geq 1$	$\theta_{i-1} = 1$	$h(\theta_{i-1}) = 1$	$g(\theta_{i-1}) = 0, \forall \theta_{i-1}$	$\lambda_{RP}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta \geq 1$	$0 \leq \theta_{i-1} = \theta \leq 1$	$h(\theta) = \theta$	$g(\theta) = 0, \forall \theta$	$\lambda_{ARA_1}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta > 0$	$-1 \leq \theta_{i-1} = \theta \leq 1$	$h(\theta) \in \mathbb{R}$	$g(\theta) = 0, \forall \theta$	$\lambda_{ARAM_1}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta > 0$	$-1 \leq \theta_{i-1} = \theta \leq 1$	$h(\theta) \in \mathbb{R}$	$g(\theta) = \theta$	$\lambda_{RCI}(t)$
$p = n = 1, 2, \dots$	$\beta_{i-1} = \beta > 0$	$-1 \leq \theta_{i-1} = \theta \leq 1$	$h(\theta) = 0, \forall \theta$	$g(\theta) = \theta$	$\lambda_{RCI_g}(t)$
$p = 0, 1, \dots, n$	$\beta_{i-1} > 0$	$-1 \leq \theta_{i-1} \leq 1$	$h(\theta_{i-1}) \in \mathbb{R}$	$g(\theta_{i-1}) \geq -1$	$\lambda_{RPI}(t)$
$p = 0, 1, \dots, n$	$\beta_{i-1} > 0$	$-1 \leq \theta_{i-1} \leq 1$	$h(\theta_{i-1}) \in \mathbb{R}$	$g(\theta_{i-1}) = 0, \forall \theta_{i-1}$	$\lambda_{RPI_h}(t)$
$p = 0, 1, \dots, n$	$\beta_{i-1} > 0$	$-1 \leq \theta_{i-1} \leq 1$	$h(\theta_{i-1}) = 0, \forall \theta_{i-1}$	$g(\theta_{i-1}) \geq -1$	$\lambda_{RPI_g}(t)$

4.3.2 Comportamento gráfico da função $\lambda_{PLPG}(t)$

Foi mostrado na subseção (4.3) que o PLPG tem como casos particulares os principais modelos para sistemas reparáveis da literatura. Apresentaremos nessa subseção o comportamento do PLPG graficamente para alguns casos particulares, por meio da função de intensidade de falha dada em (4.1) para tempos de falha simulados. Os pontos vermelhos nos gráficos representam o par ordenado $(t_i, \lambda_{PLPG}(t_i))$ e o total desses pontos representam o número de falhas simuladas. Queremos mostrar que após ocorrência da mudança no sistema, seja por reparo, melhorias, manutenção preventiva, mudanças externas, etc, a função de intensidade de falhas cai quando $\theta > 0$ e sobe para $\theta < 0$ e também que não são encaixáveis quando $g(\theta) \neq 0$, que é mostrado, por exemplo, na Figura 21.

Na Figura 19 apresentamos a função de intensidade de um Processo Poisson Homogêneo

(HPP) com p pontos de mudança, que é o caso particular mais simples do PLPG, em que $\beta_i = 1$, ou seja, $\lambda_{PLPG}(t)$ é constante entre os tempos de falha. Assim, os seguintes cenários foram estudados para este caso particular:

- Caso 1: temos o gráfico da $\lambda_{PLPG}(t)$ sem ponto de mudança ($p = 0$ e $\tau_0 = 0$) que recai na λ_C . Nesse exemplo não há saltos, mantêm-se constante até o tempo de truncamento $T = 30$. Quando $p = 1$ temos o caso HPP com um ponto de mudança, $\lambda_{PLPG}(t)$ é caracterizada por duas constantes até T .
- Caso 2: o efeito do ponto de mudança é positivo, $\theta = 0,5$, logo há uma queda da $\lambda_{PLPG}(t)$ a partir do ponto de mudança $\tau = 15$.
- Caso 3: temos que $\theta = -0,5$, há um salto da função de intensidade. Inicia-se em um patamar mais baixo e por conta da mudança prejudicial ($\theta < 0$) a intensidade salta para um patamar mais alto, prejudicando o sistema e consequentemente aumento no número esperado de falhas até $T = 30$.
- Caso 4: Também exemplificamos um caso com dois pontos de mudança da $\lambda_{PLPG}(t)$ para o caso HPP, com $\theta = 0,9$, notamos que há duas reduções por conta do efeito positivo dos pontos de mudança.

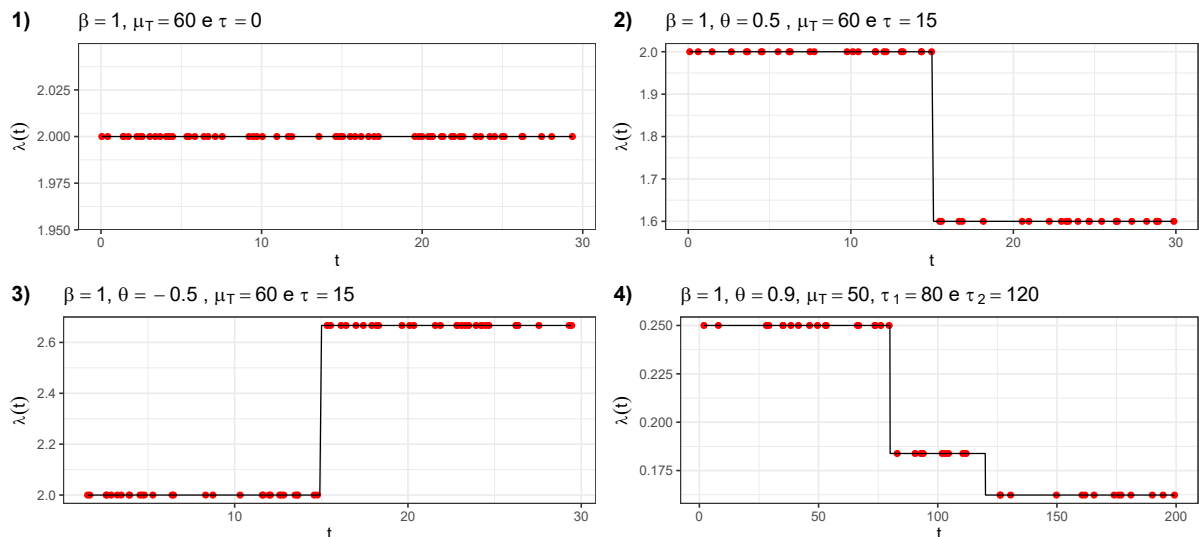


Figura 19 – Gráficos de $\lambda_{PLPG}(t)$ como função de intensidade (HPP) com p pontos de mudança.

Passando para os casos NHPP, na Figura 20 temos a função de intensidade $\lambda_{PLPG}(t)$ para a situação em que $\beta_i \neq 1$, no qual consideramos os casos sem ponto de mudança e com um ponto de mudança ($p = 1$). Nesta situação, também consideramos $\beta_{i-1} = \beta$ e $\theta_{i-1} = \theta$, assim, elaboramos os seguintes cenários:

- Nos cenários (1) e (3) temos casos sem pontos de mudança, em que a função $\lambda_{PLPG}(t)$ segue sem descontinuidade, característicos de sistemas com reparos mínimos, como em (1), em que $\beta = 1,4$, assim, $\lambda_{PLPG}(t)$ é crescente ao longo do tempo, caracterizando envelhecimento do sistema, o que acarreta aumento no número médio de falhas ao longo do tempo de truncamento $T = 30$. Para 3) $\beta = 0,9$, então $\lambda_{PLPG}(t)$ é decrescente, caracterizando melhoria do sistema e provocando redução no número médio de falhas até o tempo de truncamento $T = 200$.
- Nos cenários (2) e (4) temos o caso do efeito de um ponto de mudança sobre $\lambda_{PLPG}(t)$, no caso (2) a intensidade de falha é crescente, em $\tau = 15$ o sistema sofre uma mudança com efeito perfeito $\theta = 1$, de modo que $\lim_{t \rightarrow \tau^+} \lambda_{PLPG}(t) = 0$, quando $\tau = 15$, continuando crescente imediatamente após esse ponto. No caso (4), $\lambda_{PLPG}(t)$ é decrescente e no ponto de mudança igual a $\tau = 80$ tem efeito positivo, com $\theta = 0,9$, reduzindo ainda mais a função de intensidade de falha.

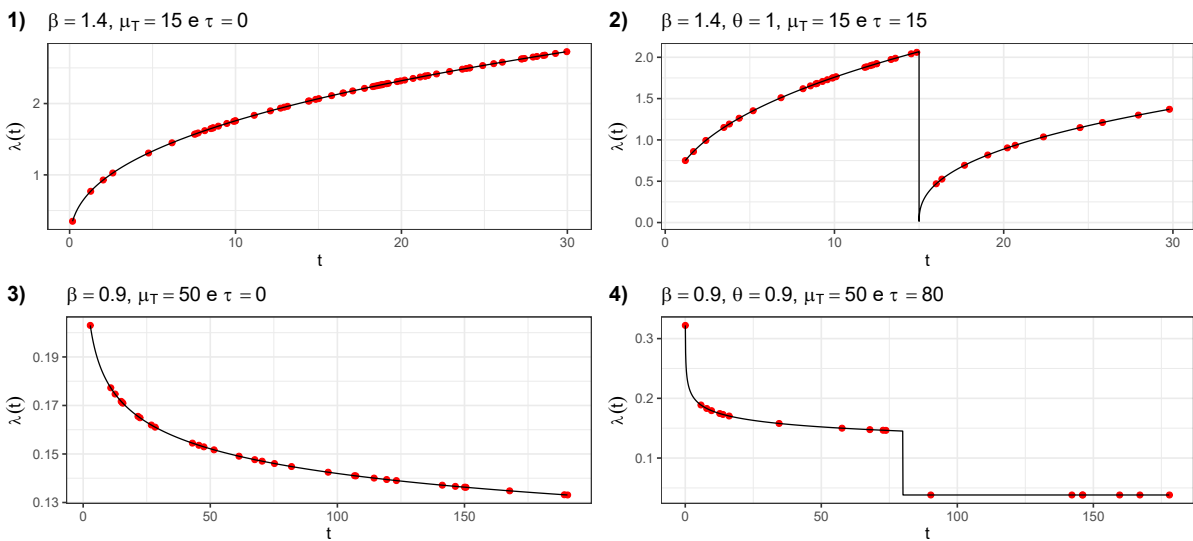


Figura 20 – $\lambda_{PLPG}(t)$ como função de intensidade de um Processo Poisson Não Homogêneo (NHPP) com p pontos de mudança.

Na Figura 21 apresentamos situações mais gerais para a função de intensidade $\lambda_{PLPG}(t)$ com um ponto de mudança ($p = 1$), também supondo $\beta_{i-1} = \beta$ e $\theta_{i-1} = \theta$, em que os efeitos de θ são apresentados nos seguintes cenários:

- Nos cenários (1) e (3) os efeitos são positivos, na situação de deterioração e renovação do sistema, respectivamente.
- Nos cenários (2) e (4) os efeitos são negativos, também diante da situação de deterioração e renovação, respectivamente.

Na Figura 22 temos a função de intensidade simulada de $\lambda_{PLPG}(t)$ com dois pontos de mudança ($p = 2$). Podemos observar os efeitos negativos e positivos das mudanças nas duas situações do sistema sobre $\lambda_{PLPG}(t)$, queda após os pontos de mudança para $\theta > 0$ e saltos após mudanças para $\theta < 0$. Perceba que os gráficos formado entre falhas não são horizontalmente encaixáveis.

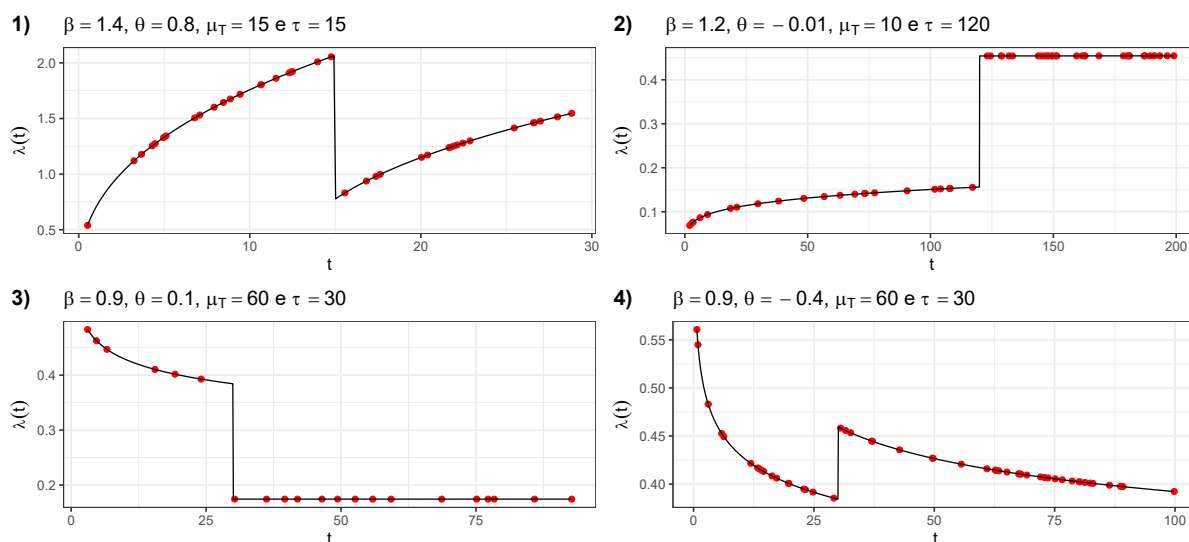


Figura 21 – Gráficos de $\lambda_{PLPG}(t)$ com um ponto de mudança ($p = 1$).

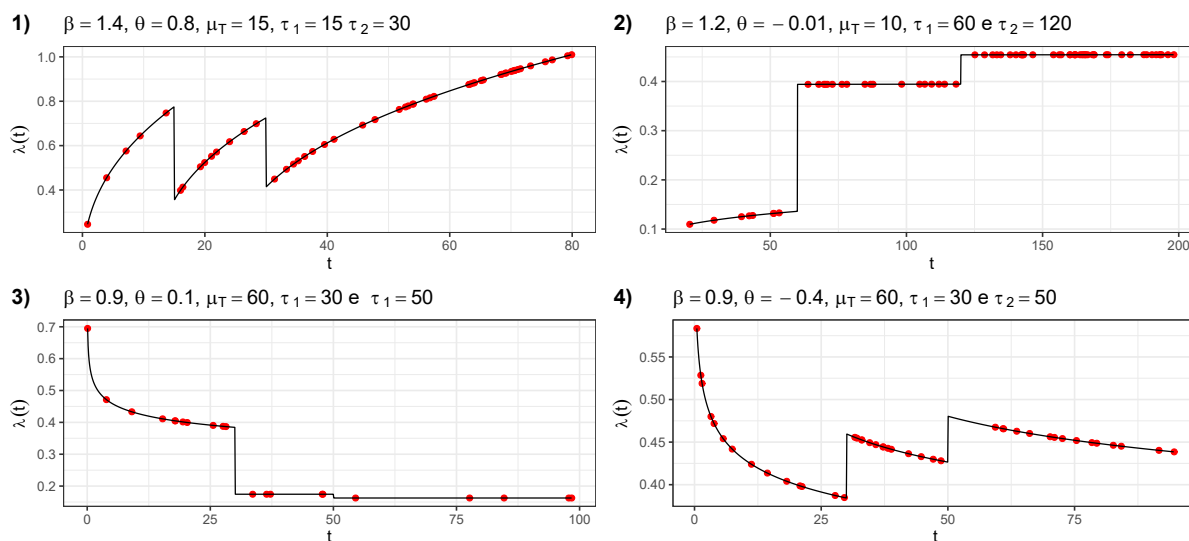


Figura 22 – Gráficos de $\lambda_{PLPG}(t)$ com dois pontos de mudança ($p = 2$).

4.4 Estimação dos parâmetros

A abordagem utilizada para estimação dos parâmetros do PLPG levará em consideração o contexto de sistemas reparáveis, em que os pontos de mudança serão interpretados como pontos conhecidos ou como pontos observados. O caso em que os parâmetros τ_i são desconhecidos será deixado como proposta de trabalho futura.

4.4.1 Pontos de mudança conhecidos

Se no intervalo de tempo $(0, T]$ forem observadas uma amostra de n tempos de falha de um sistema reparável, dados por (t_1, t_2, \dots, t_n) , em que $0 < t_1 < t_2 < \dots < t_n < T$, então a função de verossimilhança correspondente à amostra observada é dada por

$$L(\Theta|\mathbf{t}, \boldsymbol{\tau}) = f(t_1) \left[\prod_{i=1}^{n-1} f(t_{i+1}|t_i) \right] P[N(T) = n|t_n], \quad (4.5)$$

em que $\Theta = (\beta, \theta, \mu_T)$, $\boldsymbol{\tau} = \{\tau_1, \dots, \tau_p\}$ e $\mathbf{t} = (t_1, t_2, \dots, t_n)$. Em um NHPP com intensidade de falha dada por $\lambda(t|H_{t-})$, [Rigdon e Basu \(2000\)](#) demonstram que

$$f(t_i|t_{i-1}) = -\frac{d}{dt_i} \exp\left(-\int_{t_{i-1}}^{t_i} \lambda(x|H_{x-})dx\right),$$

desse modo, chegamos a

$$f(t_i|t_{i-1}) = \lambda_{PLPG}(t_i|H_{t_i-}) \exp\left(-\Lambda_{PLPG}(t_i|t_{i-1})\right),$$

em que $\Lambda_{PLPG}(t_i|t_{i-1}) = \int_{t_{i-1}}^{t_i} \lambda_{PLPG}(x|H_{x-})dx$ como consequência $f(t_1) = \lambda_{PLPG}(t_1|H_{t_1-}) \exp\left(-\Lambda_{PLPG}(t_1|t_0=0)\right)$ e $P(N(T) = n|t_n) = P(N(t_n, T] = 0) = \exp\left(-\Lambda_{PLPG}(T|t_n)\right)$. Dessa forma, a equação (4.5) é dada por

$$\begin{aligned} L(\Theta|\mathbf{t}, \boldsymbol{\tau}) &= \lambda_{PLPG}(t_1|H_{t_1-}) \exp\left(-\Lambda_{PLPG}(t_1|t_0=0)\right) \\ &\times \left[\prod_{i=1}^{n-1} \lambda_{PLPG}(t_{i+1}|H_{t_{i+1}-}) \exp\left(-\Lambda_{PLPG}(t_{i+1}|t_i)\right) \right] \\ &\times \exp\left(-\Lambda_{PLPG}(T|t_n)\right) \\ &= \left[\prod_{i=1}^n \lambda_{PLPG}(t_i|H_{t_i-}) \right] \exp\left(-\Lambda_{PLPG}(T)\right), \end{aligned}$$

em que $t_0 = 0$ e $\Lambda_{PLPG}(T) = \sum_{i=1}^n \Lambda_{PLPG}(t_i|t_{i-1}) + \Lambda_{PLPG}(T|t_n)$ é o número médio de falhas do processo até T . Utilizando o resultado obtido em (4.2), temos que

$$\begin{aligned} \Lambda_{PLPG}(T) &= \sum_{i=1}^p \left[\frac{\mu_T \left[(\tau_i - \text{sign}(\beta - 1)h(\theta)\tau_{i-1})^\beta - (\tau_{i-1} - \text{sign}(\beta - 1)h(\theta)\tau_{i-1})^\beta \right]}{T + g(\theta)\tau_{i-1}} \right] \\ &+ \frac{\mu_T \left[(T - \text{sign}(\beta - 1)h(\theta)\tau_p)^\beta - (\tau_p - \text{sign}(\beta - 1)h(\theta)\tau_p)^\beta \right]}{T + g(\theta)\tau_p}. \end{aligned}$$

Logo, a função de log-verossimilhança será dada por

$$\begin{aligned}\ell(\Theta | \mathbf{t}, \boldsymbol{\tau}) &= \log\{L(\Theta | \mathbf{t}, \boldsymbol{\tau})\} \\ &= \sum_{i=1}^n \log\{\lambda_{PLPG}(t_i | H_{t_i-})\} - \Lambda_{PLPG}(T).\end{aligned}\quad (4.6)$$

Substituindo os resultados das expressões $\lambda_{PLPG}(t_i | H_{t_i-})$ e $\Lambda_{PLPG}(T)$ na equação (4.6), obtemos

$$\begin{aligned}\ell(\Theta | \mathbf{t}, \boldsymbol{\tau}) &= \sum_{i=1}^n \log \left\{ \sum_{j=1}^{p+1} \frac{\mu_T \beta}{T + g(\theta) \tau_{j-1}} [t_i - \text{sign}(\beta - 1)h(\theta)\tau_{j-1}]^{\beta-1} \mathbf{j}(\tau_{j-1} < t_i \leq \tau_j) \right\} \\ &- \sum_{j=1}^p \left[\frac{\mu_T [(\tau_j - \text{sign}(\beta - 1)h(\theta)\tau_{j-1})^\beta - (\tau_{j-1} - \text{sign}(\beta - 1)h(\theta)\tau_{j-1})^\beta]}{T + g(\theta)\tau_{j-1}} \right] \\ &- \frac{\mu_T [(T - \text{sign}(\beta - 1)h(\theta)\tau_p)^\beta - (\tau_p - \text{sign}(\beta - 1)h(\theta)\tau_p)^\beta]}{T + g(\theta)\tau_p}.\end{aligned}$$

Portanto, o estimador de máxima verossimilhança $\hat{\Theta}_{mv}$ pode ser obtido computacionalmente, de modo que

$$\hat{\Theta}_{mv} = \underset{\Theta}{\text{argmax}} \{ \ell(\Theta | \mathbf{t}, \boldsymbol{\tau}) \}, \quad (4.7)$$

em que $\boldsymbol{\tau}$ são pontos de mudança conhecidos.

Agora, para estimação dos parâmetros para múltiplos sistemas independentes fazemos

$$L(\Theta | \mathbf{t}^{(k)}) = \prod_{l=1}^k \left\{ \left[\prod_{i=1}^{n_l} \lambda_{PLPG}(t_{il} | H_{t_{il}-}) \right] \exp \left(- \Lambda_{PLPG}^{(l)}(T) \right) \right\},$$

em que $\mathbf{t}^{(k)} = (\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_k)$ em que $\mathbf{t}_1 = (t_{11}, t_{21}, \dots, t_{n_11})$, $\mathbf{t}_2 = (t_{12}, t_{22}, \dots, t_{n_22})$, \dots , $\mathbf{t}_k = (t_{1k}, t_{2k}, \dots, t_{n_kk})$, k é o número de sistemas, t_{il} é o tempo da i -ésima falha do l -ésimo sistema, n_l é o número de falhas do l -ésimo sistema e

$$\lambda_{PLPG}(t_{il} | H_{t_{il}-}) = \sum_{j=1}^{p_l+1} \frac{\mu_T \beta}{T + g(\theta) \tau_{l(j-1)}} [t_{il} - \text{sign}(\beta - 1)h(\theta)\tau_{l(j-1)}]^{\beta-1} \mathbf{I}(\tau_{l(j-1)} < t_{il} \leq \tau_{lj}),$$

com

$$\begin{aligned}\Lambda_{PLPG}^{(l)}(T) &= \sum_{j=1}^{p_l} \left[\frac{\mu_T [(\tau_{lj} - \text{sign}(\beta - 1)h(\theta)\tau_{l(j-1)})^\beta - (\tau_{l(j-1)} - \text{sign}(\beta - 1)h(\theta)\tau_{l(j-1)})^\beta]}{T + g(\theta)\tau_{l(j-1)}} \right] \\ &+ \frac{\mu_T [(T - \text{sign}(\beta - 1)h(\theta)\tau_{lp_l})^\beta - (\tau_{lp_l} - \text{sign}(\beta - 1)h(\theta)\tau_{lp_l})^\beta]}{T + g(\theta)\tau_{lp_l}},\end{aligned}$$

em que p_l é o número de pontos de mudança do l -ésimo sistema.

Dessa forma, obtemos

$$\ell(\Theta | \mathbf{t}^{(k)}) = \log\{L(\Theta | \mathbf{t}^{(k)})\} = \sum_{l=1}^k \sum_{i=1}^{n_l} \log\{\lambda_{PLPG}(t_{il} | H_{t_{il}-})\} - \sum_{l=1}^k \Lambda_{PLPG}^{(l)}(T),$$

em que $\tau_{l(j-1)}$ é o $(j-1)$ -ésimo ponto de mudança do l -ésimo sistema.

Considerando que os tempos de mudança sejam comuns para todos os sistemas, as funções $\lambda_{PLPG}(t_{il} | H_{t_{il}-})$ e $\Lambda_{PLPG}^{(l)}(T)$ ficam reduzidas da seguinte maneira

$$\lambda_{PLPG}(t_{il} | H_{t_{il}-}) = \sum_{j=1}^{p+1} \frac{\mu_T \beta}{T + g(\theta) \tau_{j-1}} [t_{il} - \text{sign}(\beta - 1)h(\theta) \tau_{j-1}]^{\beta-1} \mathbf{I}(\tau_{j-1} < t_{il} \leq \tau_j),$$

com

$$\begin{aligned} \Lambda_{PLPG}^{(l)}(T) &= \Lambda_{PLPG}(T) = \sum_{j=1}^p \left[\frac{\mu_T [(\tau_j - \text{sign}(\beta - 1)h(\theta) \tau_{j-1})^\beta - (\tau_{j-1} - \text{sign}(\beta - 1)h(\theta) \tau_{j-1})^\beta]}{T + g(\theta) \tau_{j-1}} \right] \\ &+ \frac{\mu_T [(T - \text{sign}(\beta - 1)h(\theta) \tau_p)^\beta - (\tau_p - \text{sign}(\beta - 1)h(\theta) \tau_p)^\beta]}{T + g(\theta) \tau_p}, \end{aligned}$$

logo,

$$\ell(\Theta | \mathbf{t}^{(k)}) = \log\{L(\Theta | \mathbf{t}^{(k)})\} = \sum_{l=1}^k \sum_{i=1}^{n_l} \log\{\lambda_{PLPG}(t_{il} | H_{t_{il}-})\} - k \Lambda_{PLPG}(T).$$

Implementamos todas estas etapas de simulação no ambiente R para programação estatística (R Core Team, 2021). Nas otimizações, tanto para obtenção das integrais numéricas quanto para maximizar a função de log-verossimilhança, usamos a função `optimr` do pacote `optimr` (NASH; VARADHAN; GROTHENDIECK, 2019), com método L-BFGS-B que permite definir o espaço paramétrico na otimização.

4.4.2 Pontos de mudança observáveis

Nessa subseção, a abordagem utilizada para estimação dos parâmetros do PLPG trata os pontos de mudanças como tempos observados de falha, como notado anteriormente, quando os pontos de mudanças τ_i são tempos observáveis, então eles não são parâmetros a serem estimados, contudo, é necessário inserir, dentro dessa abordagem de estimação, um método para seleção dos pontos de mudança que serão interpretados como tempos de falha.

Nesse sentido, iremos propor uma metodologia para estimação dos parâmetros e seleção dos p pontos de mudança do modelo RPI(p), que é o caso particular mais apropriado neste contexto descrito, considerando que o efeito da mudança é sempre o mesmo e o sistema continua no mesmo estado, isto é, $\theta_{i-1} = \theta$ e $\beta_{i-1} = \beta$, para todo $i = 1, \dots, p+1$. Além de obter as estimativas dos parâmetros β , θ e μ_T pelo método de máxima verossimilhança, será proposta uma seleção dos pontos de mudança baseada também na maximização da função de verossimilhança, entretanto, penalizada pela soma do número de parâmetros estimados com o número de pontos

de mudanças do modelo proposto, vamos definir esse critério como critério de informação de Akaike com p pontos de mudanças (AIC_p), dado por $AIC_p = -2\log\hat{L} + 2(3 + p)$, em que \hat{L} é o valor da verossimilhança estimada.

Assim, considerando um sistema, o primeiro passo é obter a função de verossimilhança do modelo RPI(p), que é dada por (4.7). Entretanto, obter $\hat{\Theta}_{mv}$ só é possível se dado $\boldsymbol{\tau} \subset \{t_1, \dots, t_n\}$, o que motiva a seleção dos pontos $\tau_i \in \boldsymbol{\tau}$. Então, seja $\tau_{(i)}$ o i -ésimo ponto de mudança selecionado para compor $\boldsymbol{\tau}$. Note que não há relação de ordem entre $\tau_{(1)}, \dots, \tau_{(p)}$, por exemplo, o ponto $\tau_{(p)}$ pode ser o τ_1 , uma vez que o menor ponto de mudança τ_1 pode ser selecionado por último na proposta de seleção. Posto isso, o método objetiva estimar os parâmetros do modelo e obter $\boldsymbol{\tau}$ com o menor $AIC_p = -2\ell(\hat{\Theta}_{mv} | \mathbf{t}, \boldsymbol{\tau}) + 2(3 + p)$, em que $p = 0, \dots, n$.

O método se inicia com $p = 0$ e calculamos o $AIC_0 = -2\ell(\hat{\Theta}_{mv} | \mathbf{t}, \tau_0 = 0) + 2(3 + 0)$, neste primeiro momento não há pontos de mudança, portanto o RPI(0) é uma PLP. Em seguida, para $p = 1$, vamos obter o valor de AIC_1 com $\tau_{(1)}$ selecionado dentre os tempos $\{t_1, \dots, t_n\}$ de modo que

$$\tau_{(1)} = \underset{t \in \{t_1, \dots, t_n\}}{\operatorname{argmax}} \{ \ell(\hat{\Theta}_{mv} | \mathbf{t}, \boldsymbol{\tau} = \{t\}) \},$$

em que $\hat{\Theta}_{mv} = \underset{\Theta}{\operatorname{argmax}} \{ \ell(\Theta | \mathbf{t}, \boldsymbol{\tau} = \{t\}) \}$, assim, temos $AIC_1 = -2\ell(\hat{\Theta}_{mv} | \mathbf{t}, \boldsymbol{\tau} = \{\tau_{(1)}\}) + 2(3 + 1)$. Se $AIC_1 < AIC_0$, prosseguimos para $p = 2$, similarmente a etapa anterior, $\tau_{(2)}$ será selecionado dentre os tempos $\{t_1, \dots, t_n\} - \{\tau_{(1)}\}$ de modo

$$\tau_{(2)} = \underset{t \in \{t_1, \dots, t_n\} - \{\tau_{(1)}\}}{\operatorname{argmax}} \{ \ell(\hat{\Theta}_{mv} | \mathbf{t}, \boldsymbol{\tau} = \{\tau_{(1)}, t\}) \},$$

em que $\hat{\Theta}_{mv} = \underset{\Theta}{\operatorname{argmax}} \{ \ell(\Theta | \mathbf{t}, \boldsymbol{\tau} = \{\tau_{(1)}, t\}) \}$, em seguida calculamos $AIC_2 = -2\ell(\hat{\Theta}_{mv} | \mathbf{t}, \boldsymbol{\tau} = \{\tau_{(1)}, \tau_{(2)}\}) + 2(3 + 2)$. Se $AIC_2 < AIC_1$, prosseguimos para $p = 3$ e assim sucessivamente enquanto $AIC_p < AIC_{p-1}$ e $p \leq n$, ao final obtemos $\hat{\Theta}_{mv}$ e $\boldsymbol{\tau} = \{\tau_{(1)}, \dots, \tau_{(p)}\}$ conforme é apresentado no Algoritmo 1.

Algoritmo 1 – Algoritmo para estimar Θ pelo método de máxima verossimilhança (MV) e seleção de τ pelo critério AIC_p para um sistema reparável.

Input :

Amostra de tempos de falha truncadas em T , dada por $\mathbf{t} = (t_1, t_2, \dots, t_n)$, em que $t_i < t_j < T$ para $i < j$.

Valor da função de verossimilhança estimado do PLP ajustado, ou seja, para $p = 0$, dada por \hat{L}_{PLP} .

$\Theta = (\beta, \theta, \mu_T)$.

$\tau = \{\tau_0 = 0\}$.

Output : estimativa $\hat{\Theta}_{mv} = (\hat{\beta}, \hat{\theta}, \hat{\mu}_T)$ por MV e os valores selecionados de τ pelo critério AIC, dada por $\tau = \{\tau_{(1)}, \dots, \tau_{(p)}\}$.

$AIC_0 \leftarrow -2\log(\hat{L}_{PLP}) + 2 \times 3$

$p \leftarrow 1$

for $j \leftarrow 1$ **to** n **do**

$t \leftarrow t_j$

$\hat{\Theta}_{mv} \leftarrow \underset{\Theta}{\operatorname{argmax}} \{ \ell(\Theta | \mathbf{t}, \tau = \{t\}) \}$

$\hat{L}_j \leftarrow L(\hat{\Theta}_{mv} | \mathbf{t}, \tau)$

end

$\hat{L} \leftarrow \max\{\hat{L}_1, \dots, \hat{L}_n\}$

$AIC_1 \leftarrow -2\log(\hat{L}) + 2(3 + 1)$

$\tau_{(1)} \leftarrow \{t_j; \hat{L}_j = \hat{L}, j = 1, \dots, n\}$

while ($AIC_p < AIC_{p-1}$ and $p \leq n$) **do**

$p \leftarrow p + 1$

for $j \leftarrow 1$ **to** n **do**

if $t_j \notin \{t_{(1)}, \dots, t_{(p-1)}\}$ **then**

$t \leftarrow t_j$

$\hat{\Theta}_{mv} \leftarrow \underset{\Theta}{\operatorname{argmax}} \{ \ell(\Theta | \mathbf{t}, \tau) \}$ com $\tau = \{\tau_{(1)}, \dots, \tau_{(p-1)}, t\}$

$\hat{L}_j \leftarrow L(\hat{\Theta}_{mv} | \mathbf{t}, \tau)$

else

$j \leftarrow j$

end

end

$\hat{L} \leftarrow \max\{\hat{L}_j\}$

$AIC_p \leftarrow -2\log(\hat{L}) + 2(3 + p)$

$\tau_{(p)} \leftarrow \{t_j; \hat{L}_j = \hat{L}\}$

end

return

$p \leftarrow p - 1$

$\tau = \{\tau_{(1)}, \dots, \tau_{(p)}\}$

$\hat{\Theta}_{mv} \leftarrow \underset{\Theta}{\operatorname{argmax}} \{ \ell(\Theta | \mathbf{t}, \tau) \}$

4.5 Estudo de Simulação

Para efeitos de estudo de simulação será considerado um ponto de mudança e conhecidos, comuns para todos os sistemas. O objetivo é avaliar a qualidade das estimativas dos parâmetros do PLPG pelo método de máxima verossimilhança dada por (4.8) em termos de viés (Bias) e erro quadrático médio (MSE). Esperamos obter viés próximo de zero e MSE decrescente com o aumento do tamanho amostral k . Utilizamos dois cenários para simulação, um em que sistemas estão se deteriorando (função de intensidade é crescente) todos com um ponto de mudança, e outro cenário em que sistemas estão melhorando (função de intensidade é decrescente), também com um ponto de mudança. Em termos de parametrização, temos cenário 1) modelo PLPG com $\beta = 1.4$, $\theta = 0.8$, $\mu_T = 5$ e $\tau = 15$ e cenário 2) uma função de intensidade decrescente com $\beta = 0.8$, $\theta = 0.8$, $\mu_T = 50$ e $\tau = 15$. Consideramos $T = 30$ para os dois cenários.

Realizamos simulação de Monte Carlo com $m = 1000$ réplicas para os sistemas de tamanho $k = 5, 10, 20$ e 30 , considerando os dois cenários propostos. Para cada tamanho k foram obtidas $m = 1000$ réplicas das estimativas dos parâmetros pelo método de máxima verossimilhança. Abaixo o algoritmo utilizado para geração dos tempos de falha.

Algoritmo 2 – Gerador dos tempos de falha a partir do PLPG.

Input : $p = 1$, τ , $\Theta = (\beta, \theta, \mu_T)$, T e k .

Output : $\mathbf{t}^{(k)} = \{\mathbf{t}_1, \dots, \mathbf{t}_k\}$.

$t_{0l} = 0$, para $l = 1, \dots, k$.

for $l \leftarrow 1$ **to** k **do**

$i \leftarrow 0$

while $t_{il} < T$ **do**

$x \sim F_{t_{il}}(x) = 1 - \exp\{-\Lambda_{PLPG}(t_{il} + x|t_{il})\}$.

$t_{(i+1)l} \leftarrow t_{il} + x$ e $i \leftarrow i + 1$

end

$n_l = i$

$\mathbf{t}_l = (t_{1l}, \dots, t_{n_l})$

end

return

$\mathbf{t}^{(k)} = \{\mathbf{t}_1, \dots, \mathbf{t}_k\}$

$F_{t_{il}}(x)$ é a função densidade acumulada da variável aleatória $X_{il} = T_{(i+1)l} - T_{il}$, com $i = 1, \dots, n_l$, condicionada aos tempos de falha $T_{il} = t_{il}$ e $T_{(i+1)l} = t_{(i+1)l}$, assim, chegamos a

$$F_{t_{il}}(x) = 1 - \exp\{-\Lambda_{PLPG}(t_{il} + x|t_{il})\},$$

em que $\Lambda_{PLPG}(t_{il} + x|t_{il}) = \int_{t_{il}}^{t_{il}+x} \lambda_{PLPG}(t|H_{x-})dt$. Como $F_{t_{il}}(X)$ tem distribuição uniforme $U(0, 1)$, então, para obter um x pseudo-aleatório, gera-se $u_{il} \sim U(0, 1)$ e resolve-se x numericamente na equação $F_{t_{il}}(x) = u_{il} = 1 - \exp\{-\Lambda_{PLPG}(t_{il} + x|t_{il})\}$ ou, equivalentemente, resolver

numericamente a seguinte equação

$$\Lambda_{PLPG}(t_{il} + x|t_{il}) = -\log(1 - u_{il}).$$

Implementamos todas estas etapas de simulação no ambiente R para programação estatística (R Core Team, 2021). Nas otimizações para obtenção das integrais numéricas usamos a função `optimr(·)` do pacote `optimr` elaborado por Nash, Varadhan e Grothendieck (2019), com método L-BFGS-B que permite definir os limites das variáveis na otimização.

Para análise da qualidade das estimativas, utilizamos o erro quadrático médio (MSE), que é uma medida frequentemente utilizada em estudos de simulação para quantificar a qualidade das estimativas, assim como o viés (Bias) da estimativa. Por exemplo, o viés e erro quadrático médio para as estimativas de θ , respectivamente, com total de sistema igual a k :

$$Bias(\hat{\theta}_k) = \frac{\sum_{i=1}^m (\hat{\theta}_{ik} - \theta)}{m} \text{ e } MSE(\hat{\theta}_k) = \frac{\sum_{i=1}^m (\hat{\theta}_{ik} - \theta)^2}{m},$$

em que $\hat{\theta}_{ik}$ é a i -ésima estimativa para θ com total de sistemas igual a k . O mesmo processo se repete para as estimativas de β e μ_T .

Na Tabela 5 apresentamos os valores do viés e MSE das estimativas pelo método de máxima verossimilhança para os dois cenários simulados. Nas Figura 13 e Figura 14 apresentamos os resultados gráficos da tendência do viés e MSE para os cenários 1 e 2, respectivamente, considerando os valores de $k = 5, 10, 20, 30$.

Nos dois cenários, as estimativas obtidas pelo método de máxima verossimilhança para os parâmetros β , μ_T e θ estão bastante próximos de seus respectivos valores predefinidos, uma vez que os vieses tentem para zero com o aumento de k , atingindo os menores valores para $k = 30$.

Com relação ao MSE, podemos verificar de modo geral na Figura 23 e na Figura 24 que, nos cenários apresentados, para as estimativas dos três parâmetros do PLPG, quanto maior o tamanho da amostra k considerado, os valores do MSE decrescem para zero.

Podemos destacar no cenário 1, que os vieses obtido para β e $\mu_T = 10$ foram negativos, exceto para θ , e observamos para os três parâmetros uma aproximação dos vieses em direção ao zero com o aumento de k , como desejado. Já no cenário 2, o viés e o MSE de θ se aproximam de zero com lentidão, comparativamente ao parâmetro β , por exemplo, do mesmo cenário.

Concluimos do estudo de simulação que as estimativas de máxima verossimilhança para os parâmetros do PLPG com um ponto de mudança conhecido possuem boas estimativas, com vieses e MSE tendendo a zero com aumento de k , em conformidade com a propriedade assintótica dos estimadores de máxima verossimilhança, refletindo propriedades frequentistas esperadas e desejáveis.

Tabela 5 – Resultado da simulação para os dois cenários propostos, caso composto por múltiplos sistemas.

Cenário	k	$Bias(\hat{\beta})$	$MSE(\hat{\beta})$	$Bias(\hat{\theta})$	$MSE(\hat{\theta})$	$Bias(\hat{\mu}_T)$	$MSE(\hat{\mu}_T)$
$\beta = 1,4, \theta = 0,8 \text{ e } \mu_T = 5$							
1	5	-0,0217	0,0379	0,0509	0,0555	-0,328	7,77
	10	-0,0089	0,0197	0,0268	0,0269	-0,219	4,30
	20	-0,0019	0,0092	0,0061	0,0137	-0,182	2,04
	30	-0,0025	0,0023	0,0019	0,0083	-0,095	1,30
$\beta = 0,8, \theta = 0,8 \text{ e } \mu_T = 50$							
2	5	-0,01260	0,00840	-0,143	0,187	2,48	224,0
	10	-0,01440	0,00342	-0,146	0,166	2,39	103,0
	20	-0,00801	0,00190	-0,116	0,132	1,27	50,6
	30	-0,00651	0,00141	-0,117	0,123	1,03	35,5

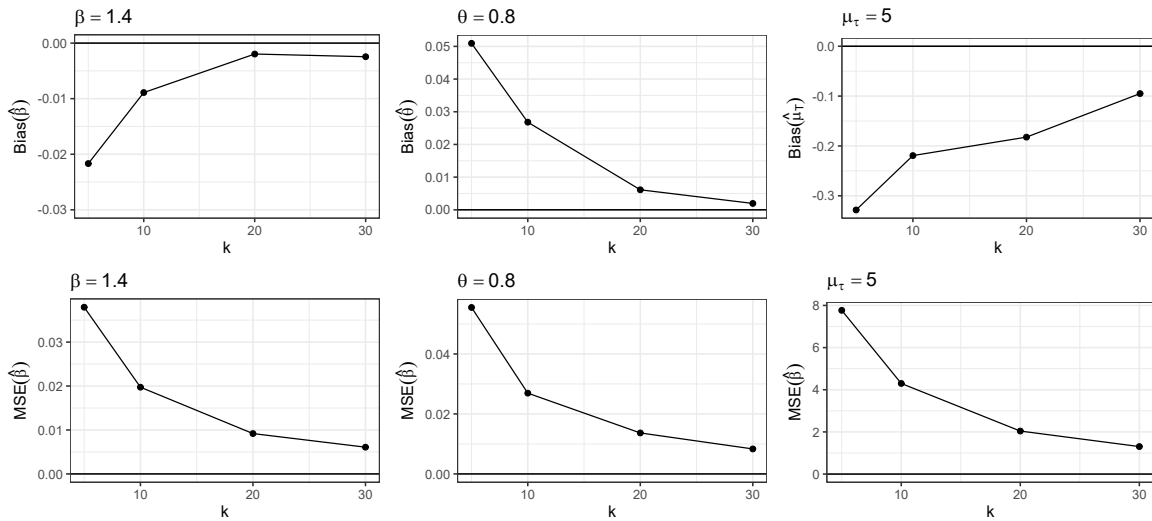


Figura 23 – Cenário 1: resultado do viés e MSE da simulação para PLPG com $\tau_1 = 15$ e $T = 30$.

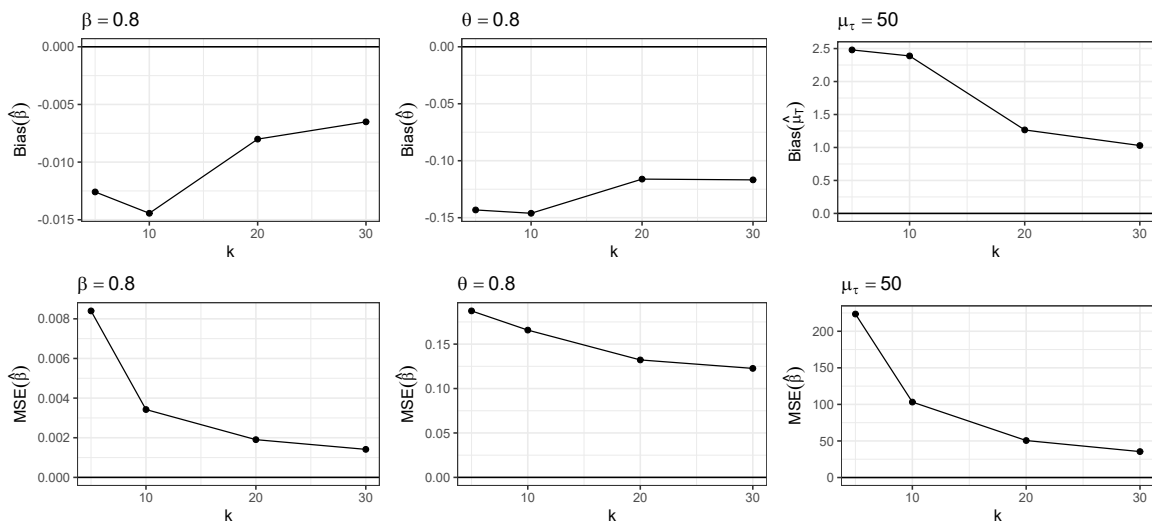


Figura 24 – Cenário 2: resultado do viés e MSE da simulação para PLPG com $\tau_1 = 15$ e $T = 30$.

4.6 Aplicação

Para ilustrar a importância da abordagem proposta, vamos utilizar duas amostras reais de tempos de falha. Será utilizado o critério de informação de Akaike para indicar o modelo com melhor ajuste e mais parcimonioso, dado por $AIC = -2\log\hat{L} + 2m$, em que $\log\hat{L}$ é log natural da verossimilhança máxima estimada e m é total de parâmetros estimados do modelo. O objetivo da aplicação é comparar o PLPG com o PLP e outros modelos da literatura que utilizam $\lambda(t)$ do PLP como base para o desenvolvimento da modelagem da função de intensidade.

Para obtenção dos intervalos de confiança dos parâmetros estimados $\hat{\mu}_T$ e $\hat{\beta}$ serão utilizadas as estimativas dos seus erros padrões (SE, do inglês *standard error*), em que serão tomados os valores da diagonal principal resultante do inverso da matriz de informação observada. Usando esses valores, os intervalos de confiança de 95% (CI 95%) dos parâmetros estimativas são obtidos por:

$$\text{CI 95\% para } \hat{\beta} : (\hat{\beta} - 1,96\text{SE}(\hat{\beta}), \hat{\beta} + 1,96\text{SE}(\hat{\beta})),$$

em que $\text{SE}(\hat{\beta})$ é o erro padrão de $\hat{\beta}$, da mesma forma obtemos o CI 95% para $\hat{\mu}_T$.

Para o parâmetro θ abordaremos a estimativa pontual, dado que é limitado inferiormente e superiormente, dificultando a obtenção dos intervalos de confiança assintóticos, uma vez que frequentemente os limites do intervalo podem ficar fora da cobertura do espaço paramétrico de θ . Uma solução é a obtenção do intervalo de credibilidade para θ , abordagem Bayesiana, não tratada neste trabalho.

4.6.1 Exaustor para vapor

O primeiro conjunto de tempos de falha utilizado para modelagem é o das falhas de um exaustor para vapor, com tempo de operação em número de dias. Este conjunto de dados foram analisados por [Syamsundar, Naikan e Wu \(2021\)](#) e [Syamsundar e Naikan \(2008\)](#). Na pesquisa, os autores [Syamsundar, Naikan e Wu \(2021\)](#) identificaram que o melhor modelo para o conjunto de dados é o XARA₁₅-GRA com linha de base logarítmica linear ($\log\hat{L} = -64.4$), contudo o melhor modelo com PLP como linha de base foi o XARA_∞ - GRA - PLP com $\log\hat{L} = -66.5$. Já [Syamsundar e Naikan \(2008\)](#) concluíram que o modelo segmentado HPP com dois pontos de mudança (HPP MCP, $\log\hat{L} = -64.211$), $\hat{\tau}_1 = 128$ e $\hat{\tau}_2 = 244$ dias, se ajusta melhor ao conjunto de dados em comparação com os modelos HPP sem pontos de mudança ($\log\hat{L} = -70.733$), NHPP ($\log\hat{L} = -70.665$) e ARI₁ ($\log\hat{L} = -70.662$). O conjunto de dados é composto por 17 registros de tempos de falha, em um período de pouco mais de um ano, conforme apresentado na Tabela 6.

Tabela 6 – Conjunto de tempos de falha de um exaustor para vapor em dias.

27	92	128	133	140	151	153	158	169
28	183	191	212	234	244	300	401	

Na Figura 25 observamos no gráfico de falhas acumuladas que há, subjetivamente, mudanças na curvatura provocada por mudanças no exaustor, seja no ambiente em que se encontra ou provocado pelos reparos realizados no sistema. Vamos considerar nas aplicações que as mudanças sejam por conta das ações de manutenção corretiva, isto é, por conta das ações de reparo. Pra efeitos práticos, estabeleceremos o tempo de truncamento $T = 401$.

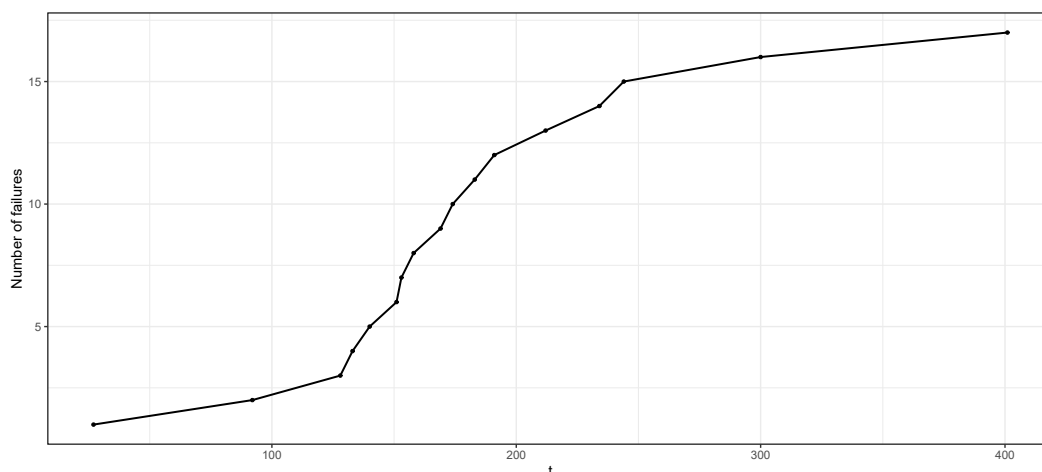


Figura 25 – Total de falhas acumuladas de um exaustor para valor.

Os modelos $RPI(p)$ e RCI com $\theta_i = \theta$ e $\beta_i = \beta$, que são casos particulares do PLPG, e os principais modelos propostos na literatura de sistemas reparáveis com função de intensidade baseados em PLP foram ajustados ao conjunto de dados de tempos de falha do exaustor para vapor. Os valores do log da função de verossimilhança estimada ($\log \hat{L}$) para o conjunto de dados proposto, o total m de parâmetros estimados do modelo e os valores de AIC são dados na Tabela 7. Os modelos tradicionais como HPP e NHPP (PLP) obtiveram um dos menores valores de $\log \hat{L}$, resultado similar ao observado nos modelos $ARAM_1$ e ARI_1 , porém, por terem três parâmetros, os valores do AIC para estes modelos são maiores comparativamente aos modelos HPP e NHPP. Por outro lado, os melhores modelos em termos de AIC são o modelo HPP MCP de Syamsundar e Naikan (2008), o modelo $XARA_\infty$ - GRA - PLP proposto por Syamsundar, Naikan e Wu (2021) e o modelo $RPI(p)$ proposto neste trabalho. Nesse sentido, dentre os três melhores, o modelo $RPI(p)$ obteve menor AIC , ou seja, se destacando com melhor ajuste dentre os principais modelos propostos na literatura, indicando uma superioridade do modelo $RPI(p)$.

Os parâmetros do modelo $RPI(p)$ estimado através do Algoritmo 1 estão dispostos na Tabela 8. Podemos observar que o intervalo de confiança estimado para o parâmetro $\hat{\beta} = 2.0627$ se mantém acima da unidade, sugerindo envelhecimento significativo do sistema ao longo do tempo observado. Os tempos de falha selecionados como pontos de mudança pelo método foram $\tau_1 = 244$ e $\tau_2 = 300$, com $\hat{\theta} = 1$, indicando reparo perfeito nesses pontos de mudança selecionados. O valor estimado de $\hat{\mu}_T$ indica o impacto negativo da situação de deterioração do sistema no número médio de falhas, uma vez que, se o sistema mantivesse a intensidade de falha

Tabela 7 – Resultado do $\log \hat{L}$ e AIC dos modelos utilizados.

Modelo	$\log \hat{L}$	m	AIC
HPP	-70,73	1	142,00
HPP MCP	-64,21	5	138,40
NHPP (PLP)	-70,66	2	145,32
ARAM ₁	-70,65	3	147,30
ARI ₁	-70,66	3	147,32
XARA _∞ - GRA - PLP	-66,50	4	141,00
RPI(p)	-63,75	3	133,50
RCI	-70,39	3	146,78

constante com $\hat{\mu}_T = 0,0723$ ao longo dos 401 dias observados, o número de falhas esperado seria menor que uma falha em 401 dias, pois a média de falhas do HPP seria $\hat{\mu}_T = 0,0723 < 1$, sem influência do efeito da mudança e da situação de deterioração do sistema.

Tabela 8 – Parâmetros do modelo RPI(p) usado para modelar os tempos de falha do exaustor para vapor.

n	p	$\hat{\beta}$ (CI 95%)	$\hat{\mu}_T$ (CI 95%)	$\hat{\theta}$	τ_1	τ_2	$\log \hat{L}$	AIC
17	2	2,0627 (1,2751, 2,8500)	0,0723 (0, 0,3790)	1	244	300	-63,75	133,50

Os gráficos do número de falhas observadas (dados reais - DR) versus o número esperado de falhas (modelo ajustado - MA) do modelo RPI(p) ajustado aos dados de falhas do exaustor para vapor e sua função de intensidade estimada são mostrados na Figura 26. Podemos ver graficamente que o modelo RPI(p) oferece um bom ajuste para o cumulativo de falhas observadas. Como esperado, a função de intensidade apresenta queda nos pontos de mudança $\tau_1 = 244$ e $\tau_2 = 300$ uma vez que o efeito do reparo foi perfeito ($\hat{\theta} = 1$) nesses tempos de falha. Por outro lado, podemos considerar que os demais tempos não caracterizam mudanças na intensidade de falha provocada pelos reparos. Diferentemente do RCI que considera o efeito do θ em todos os tempos de falha, porém, não obteve um bom ajuste, como observado na análise do AIC da Tabela 7.

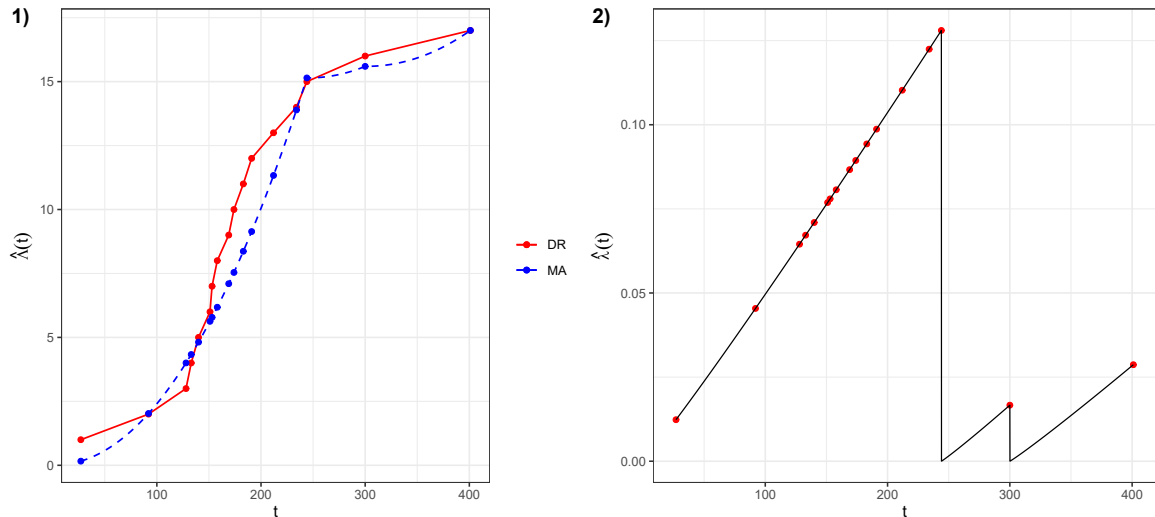


Figura 26 – Gráficos do 1) número de falhas observadas (DR) versus o número esperado de falhas (MA) do modelo $RPI(p)$ ajustado e 2) função de intensidade estimada do modelo $RPI(p)$.

4.6.2 Sistema de ar-condicionado do Boeing 720 nº 7908

O segundo conjunto de tempos de falha utilizado para modelagem é o das falhas do sistema de ar-condicionado de um avião a jato Boeing 720 nº série 7908, com tempo de funcionamento em horas. Este conjunto de dados é bastante utilizado na literatura, inicialmente com [Proschan \(1963\)](#) para investigar a taxa de falha desse sistema, posteriormente analisado por [Cox e Lewis \(1966\)](#) e [Akman e Raftery \(1986\)](#). Recentemente esses dados foram tratados no trabalho de [Syamsundar e Naikan \(2008\)](#), em que os autores concluíram que o modelo segmentado HPP com três pontos de mudança (HPP MCP, $\log \hat{L} = -119,26$), com $\hat{\tau}_1 = 413$, $\hat{\tau}_2 = 696$ e $\hat{\tau}_3 = 1851$ dias, se ajusta melhor ao conjunto de dados em comparação com os modelos HPP sem pontos de mudança ($\log \hat{L} = -127,91$), NHPP ($\log \hat{L} = -124,95$), ARI_1 ($\log \hat{L} = -124,95$), HPP com um ponto de mudança (HPP SCP, $\log \hat{L} = -123,01$) e NHPP com um ponto de mudança (NHPP SCP, $\log \hat{L} = -118,46$). Foram registrados 23 tempos de falha, conforme apresentado na Tabela 9.

Tabela 9 – Conjunto de tempos de falha no sistema de ar-condicionado do Boeing 720 nº 7908.

413	427	485	522	622	687	696	865	1312
1496	1532	1733	1851	1885	1916	1934	1952	2019
2076	2138	2145	2167	2201				

Similarmente à primeira base de dados, na Figura 27, notamos, em uma análise subjetiva, que o gráfico de falhas acumuladas apresentam inflexão em algum ponto, provocando mudanças na curvatura do gráfico do total de falhas acumuladas do sistema, seja por mudanças provocadas

no ambiente em que se encontra ou provocado pelos reparos realizados no mesmo ao longo do tempo. Para efeitos práticos, estabeleceremos o tempo de truncamento $T = 2201$.

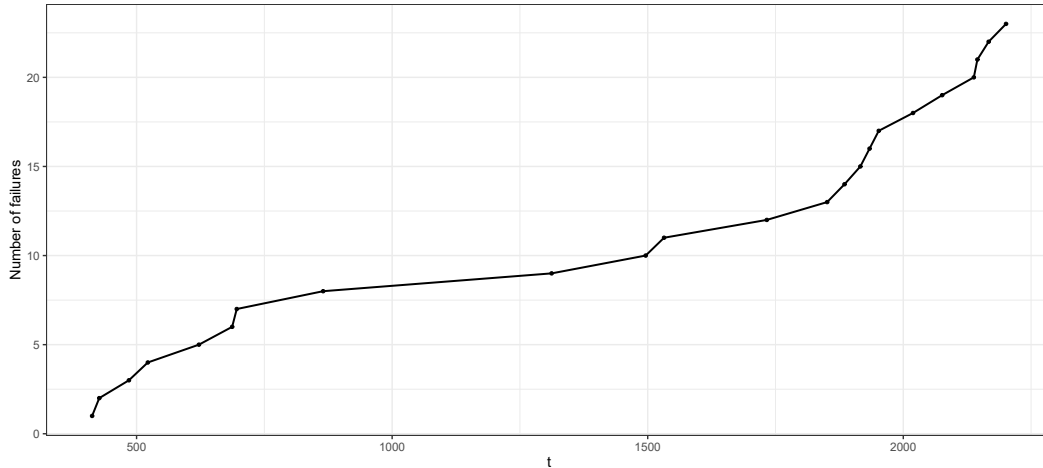


Figura 27 – Total de falhas acumuladas no sistema de ar-condicionado do Boeing 720 nº 7908.

Os modelos $RPI(p)$ e RCI e os principais modelos propostos na literatura de sistemas reparáveis com função de intensidade baseados no PLP foram ajustados ao conjunto de dados de tempos de falha no sistema de ar-condicionado do Boeing 720 nº 7908. Os valores do log da função de verossimilhança estimada ($\log \hat{L}$) para o conjunto de dados proposto, o total m de parâmetros estimados do modelo e os valores de AIC são dados na Tabela 10. Os modelos tradicionais como HPP, NHPP (PLP), ARI_1 , $ARAM_1$, HPP SCP e o novo modelo proposto PLPG - RCI obtiveram os menores valores de $\log \hat{L}$ e maiores valores de AIC . Por outro lado, os melhores modelos em termos de AIC são os modelos NHPP SCP de [Syamsundar e Naikan \(2008\)](#) e $RPI(p)$ proposto. Nesse sentido, dentre os dois melhores, o modelo $RPI(p)$ obteve menor AIC , com valor de 244,00 contra $AIC = 246,92$ do NHPP SCP, indicando uma superioridade do modelo $RPI(p)$ em comparação aos outros modelos da literatura abordados.

Os parâmetros do modelo $RPI(p)$ estimado através do Algoritmo 1 estão dispostos na Tabela 11. Podemos observar que o intervalo de confiança estimado para o parâmetro $\hat{\beta} = 2,2343$ se mantém acima da unidade, sugerindo envelhecimento significativo do sistema ao longo do tempo observado. O tempo de falha selecionado como ponto de mudança pelo modelo foi $\tau_1 = 865$, com $\hat{\theta} = 0,9999$, indicando reparo próximo do perfeito nesse ponto de mudança selecionado. O valor estimado de $\hat{\mu}_T$ indica o impacto negativo da situação de deterioração do sistema no número médio de falhas, uma vez que, se o sistema mantivesse a intensidade de falha constante com $\hat{\mu}_T = 0,0047$ ao longo do tempo observado, o número de falhas esperado seria menor que uma falha em $T = 2201$ dias, pois a média de falhas do HPP seria $\hat{\mu}_T = 0,0047 < 1$, desconsiderando a influência do efeito da mudança e a situação do sistema.

Tabela 10 – Resultado do $\log \hat{L}$ e AIC dos modelos utilizados para os tempos de falha no sistema de ar-condicionado do Boeing 720 n° 7908,

Modelo	$\log \hat{L}$	m	AIC
HPP	-127,91	1	257,82
HPP SCP	-123,01	3	252,02
HPP MCP	-119,26	7	252,52
NHPP (PLP)	-124,95	2	253,90
NHPP SCP	-118,46	5	246,92
ARAM ₁	-124,80	3	255,60
ARI ₁	-124,95	3	255,90
RPI(p)	-119,00	3	244,00
RCI	-124,80	3	255,60

Tabela 11 – Parâmetros do modelo RPI(p) usado para modelar os tempos de falha no sistema de ar-condicionado do Boeing 720 n° 7908.

n	p	$\hat{\beta}$ (CI 95%)	$\hat{\mu}_T$ (CI 95%)	$\hat{\theta}$	τ_1	$\log \hat{L}$	AIC
23	1	2,2343 (1,4148, 3,0538)	0,0047 (0, 0,03307)	0,9999	865	-119,00	244,00

Os gráficos do número de falhas observadas (dados reais - DR) versus o número esperado de falhas (modelo ajustado - MA) do modelo RPI(p) ajustado aos dados de falhas do exaustor para vapor e sua função de intensidade estimada são mostrados na Figura 28. Podemos ver graficamente que o modelo RPI(p) oferece um bom ajuste para o cumulativo de falhas observadas. Como esperado, a função de intensidade apresenta queda no ponto de mudança $\tau_1 = 865$ uma vez que o efeito do reparo foi próximo do perfeito ($\hat{\theta} = 0,9999$). Por outro lado, podemos considerar que os demais tempos não caracterizam mudanças na intensidade de falha provocada pelos reparos. Diferentemente do RCI que considera o efeito do θ em todos os tempos de falha, porém não obteve um bom ajuste, como observado na análise do AIC da Tabela 10.

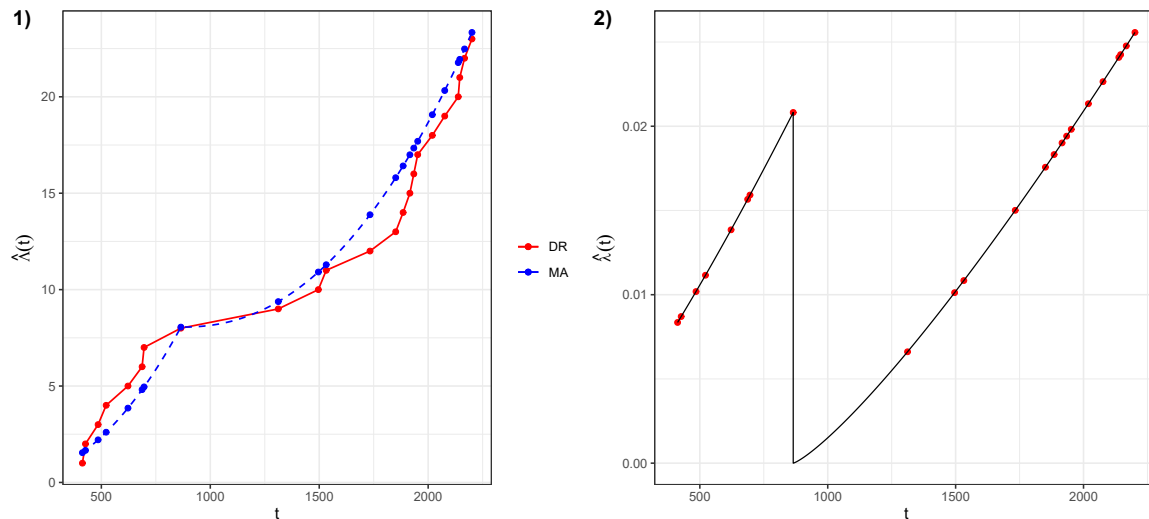


Figura 28 – Gráficos do 1) número de falhas observadas (DR) versus o número esperado de falhas (MA) do modelo $RPI(p)$ ajustado e 2) função de intensidade estimada do modelo $RPI(p)$.

4.7 Considerações finais

Neste capítulo, propomos um novo Processo Poisson Não Homogêneo que generaliza o Processo de Lei de Potência, baseados em pontos de mudança, que chamamos de PLPG. Frisamos que os pontos de mudanças podem ser interpretados como parâmetros conhecidos, desconhecidos e como tempos observáveis de falha.

A partir do PLPG é possível obter, como casos particulares, os principais modelos para sistemas reparáveis, como o próprio PLP, vários modelos baseados em pontos de mudanças, modelos de reparos perfeitos, imperfeitos como ARA_1 e $ARAM_1$, também são adicionados novos modelos baseados em todos ou partes dos tempos de falhas, que denominamos de Modelos de Reparo Completamente Imperfeitos - RCI e modelos de Reparo Parcialmente Imperfeitos com p pontos de mudança - $RPI(p)$, respectivamente. O PLPG permitindo que a intensidade após os reparos não permaneça paralela à intensidade inicial. Essa nova alternativa de modelagem de sistemas reparáveis com PLPG amplia as aplicações em dados real de tempos de falha.

Propomos um método de estimação por máxima verossimilhança para obter as estimativas dos parâmetros do PLPG e seleção dos pontos de mudança quando são interpretados como tempos de falha observáveis. Na simulação consideramos os pontos de mudança como parâmetros conhecidos e na aplicação realizamos a seleção dos pontos de mudança.

Com os resultados obtidos no estudo de simulação, considerando os dois cenários propostos, concluímos que as propriedades assintóticas dos estimadores de máxima verossimilhança estão de acordo com o esperado: assintoticamente não viciado e com MSE se aproximando de zero com o aumento do tamanho amostral k .

Nas duas aplicações propostas, para ilustrar o potencial do PLPG, concluímos que o

novo modelo $RPI(p)$ obteve o menor valor de AIC, se mostrando uma alternativa potencial. Por fim, esperamos que o novo processo PLPG seja uma alternativa relevante aos principais modelos existentes para sistemas reparáveis.

CONCLUSÃO E PROPOSTAS FUTURAS

Desenvolvemos nesta tese uma generalização do modelo ARA_1 ($ARAM_1$) com PLP em que adicionamos o caso cujo o sistema reparável possui intensidade decrescente, também apresentamos o PLP generalizado (PLPG), baseados em pontos de mudança, que podem ser interpretados como parâmetros conhecidos, desconhecidos e como tempos observáveis de falha.

No modelo ARA_1 com PLP, propomos uma nova reparametrização do PLP como truncamento por tempo para incorporá-lo ao novo modelo e assim preservar a interpretação original dos parâmetros do PLP. Consideramos que o efeito do reparo θ varia de -1 a 1 , quando $-1 \leq \theta < 0$ o reparo é considerado prejudicial, $0 < \theta < 1$ como eficiente e quando $\theta = 1$ o reparo é perfeito.

No novo processo PLP generalizado (PLPG) é possível considerar o impacto de todos ou algumas tipos de reparos no processo ou também tratar os pontos de mudanças com saltos ou quedas e, principalmente, quantificar seu efeito sobre função de intensidade, lhe concedendo mais flexibilidade nas modelagens. A partir do PLPG é possível derivar diversos modelos para processos de contagem com ponto de mudança e modelos para análise de sistemas reparáveis.

Foram obtidas as log-verossimilhança correspondente a uma amostra observada de tempos de falha, objetivando estimar seus parâmetros. Em especial, foi proposto para o PLPG um algoritmo de seleção dos pontos de mudança quando são interpretados como tempos de falha observáveis.

No estudo de simulação, concluímos, a partir dos cenários estabelecidos, que os estimadores propostos para os novos modelos apresentaram boas estimativas, o MSE reduz com o aumento da amostra k e as estimativas ficam próximas dos verdadeiros valores dos parâmetros preestabelecidos, refletindo boas propriedades.

Já nas aplicações com dados reais de tempos de falha, notamos que os modelos se ajustaram bem aos dados, mostrando que são fortes concorrentes para modelagem em sistemas reparáveis. A partir do $ARAM_1$ foi possível captura situação de renovação do sistema até o

tempo de truncamento T , melhorando o ajuste do modelo aos dados. Concluímos que o modelo de Reparo Parcialmente Imperfeitos com p pontos de mudança - RPI(p), caso particular do PLPG, obteve o menor valor de AIC, se mostrando como uma alternativa potencial.

Como não é possível esgotar nesta tese todas as propriedades, avaliações, melhorias, aplicações e métodos de estimação dos novos modelos propostos, vamos apontar algumas propostas de trabalhos futuros:

- Ampliar a memória do modelo ARAM₁, em que a intensidade de falha em t é influenciada pelo acúmulo de m tempos de falhas anteriores à t com certo efeito θ .
- Estudar o PLPG com parametrização truncada por falhas.
- Avaliar a influência da constante k em $h(\theta)$ na qualidade das estimativas dos parâmetros do PLPG por meio de estudo de simulação.
- Realizar um estudo de simulação mais abrangente, com diversidade de cenários, para análise detalhada das propriedades assintóticas dos estimadores propostos para o PLPG.
- Utilizar abordagem Bayesiana para estimar os parâmetros do PLPG e, principalmente, obter os intervalos de credibilidade dos parâmetros, em especial, o intervalo para o parâmetro limitado θ .
- Propor método de estimação para o PLPG cujos parâmetros β_i e θ_i , para $i = 0, \dots, p$, são distintos para cada ponto de mudança e avaliar a qualidade por meio de estudos de simulação.
- Estudar o modelo PLPG com pontos de mudança como parâmetros desconhecidos, propondo métodos de estimação para seus parâmetros.
- Realizar estudo aprofundado dos casos particulares do PLPG.
- Incorporar os novos modelos para promover política ótima de manutenção para modelos de sistemas reparáveis.

REFERÊNCIAS

AKMAN, V. E.; RAFTERY, A. E. Bayes factors for non-homogeneous poisson processes with vague prior information. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 48, n. 3, p. 322–329, 1986. Citado na página 95.

AL-DOURI, A.; EL-HALWAGI, M. M.; GROTH, K. M. Emergency shutdowns of propylene production plants: Root cause analysis and availability modeling. **Journal of Loss Prevention in the Process Industries**, Elsevier, p. 104921, 2022. Citado na página 24.

ALMEIDA, M. P.; PAIXÃO, R. S.; RAMOS, P. L.; TOMAZELLA, V.; LOUZADA, F.; EHLERS, R. S. Bayesian non-parametric frailty model for dependent competing risks in a repairable systems framework. **Reliability Engineering & System Safety**, Elsevier, v. 204, p. 107145, 2020. Citado na página 24.

ASCHER, H.; FEINGOLD, H. Repairable systems reliability: Modelling, inference, misconceptions and their causes. **Marcel Dekker Inc**, 1984. Citado nas páginas 23 e 35.

BARLOW, R. E.; PROSCHAN, F. **Mathematical theory of reliability**. [S.l.]: SIAM, 1996. Citado na página 23.

BOUKAI, B. Bayes sequential procedure for estimation and for determination of burn in time in a hazard rate model with an unknown change point parameter. **Sequential Analysis**, Taylor & Francis, v. 6, n. 1, p. 37–53, 1987. Citado nas páginas 26 e 27.

BROWN, M.; PROSCHAN, F. Imperfect repair. **Journal of Applied Probability**, Cambridge University Press, v. 20, n. 4, p. 851–859, 1983. Citado na página 24.

CALABRIA, R.; PULCINI, G. Discontinuous point processes for the analysis of repairable units. **International Journal of Reliability, Quality and Safety Engineering**, World Scientific, v. 6, n. 04, p. 361–382, 1999. Citado na página 42.

CHUMNAUL, J.; SEPEHRIFAR, M. Improved test for the scale parameter of the power-law process with incomplete failure data. **Communications in Statistics-Theory and Methods**, Taylor & Francis, v. 51, n. 12, p. 4221–4234, 2022. Citado na página 24.

COX, D. R.; LEWIS, P. A. The statistical analysis of series of events. Springer, 1966. Citado na página 95.

CROW, L. Reliability analysis for complex repairable systems. **Reliability and Biometry**, ed. F. Proschan and RJ Serfling. SIAM, Philadelphia, 1974. Citado nas páginas 23, 28, 34, 45, 67 e 75.

DOYEN, L.; GAUDOIN, O. Classes of imperfect repair models based on reduction of failure intensity or virtual age. **Reliability Engineering & System Safety**, Elsevier, v. 84, n. 1, p. 45–56, 2004. Citado nas páginas 11, 13, 24, 25, 27, 28, 39, 42, 46, 47, 67, 77 e 78.

- _____. Discussion of “virtual age, is it real?”. **Applied Stochastic Models in Business and Industry**, v. 37, n. 1, p. 21–23, 2021. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/asmb.2589>>. Citado na página 25.
- DOYEN, L.; GAUDOIN, O.; SYAMSUNDAR, A. On geometric reduction of age or intensity models for imperfect maintenance. **Reliability Engineering & System Safety**, Elsevier, v. 168, p. 40–52, 2017. Citado nas páginas 25, 42 e 43.
- D’ANDREA, A. M.; TOMAZELLA, V. L.; ALJOHANI, H. M.; RAMOS, P. L.; ALMEIDA, M. P.; LOUZADA, F.; VERSSANI, B. A.; GAZON, A. B.; AFIFY, A. Z. Objective bayesian analysis for multiple repairable systems. **Plos one**, Public Library of Science San Francisco, CA USA, v. 16, n. 11, p. e0258581, 2021. Citado na página 24.
- FINKELSTEIN, M. **Failure rate modelling for reliability and risk**. [S.l.]: Springer Science & Business Media, 2008. Citado na página 43.
- FINKELSTEIN, M.; CHA, J. H. Virtual age, is it real? - discussing virtual age in reliability context. **Applied Stochastic Models in Business and Industry**, v. 37, n. 1, p. 3–16, 2021. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/asmb.2567>>. Citado na página 25.
- KIJIMA, M.; MORIMURA, H.; SUZUKI, Y. Periodical replacement problem without assuming minimal repair. **European Journal of Operational Research**, Elsevier, v. 37, n. 2, p. 194–203, 1988. Citado nas páginas 24, 25 e 39.
- LAM, Y. A note on the optimal replacement problem, *adv. appl. prob.* 1988. Citado na página 42.
- LINDQVIST, B. H. On the statistical modeling and analysis of repairable systems. **Statistical science**, Institute of Mathematical Statistics, v. 21, n. 4, p. 532–551, 2006. Citado nas páginas 24 e 39.
- LOADER, C. R. Inference for a hazard rate change point. **Biometrika**, Oxford University Press, v. 78, n. 4, p. 749–757, 1991. Citado na página 26.
- _____. A log-linear model for a poisson process change point. **The Annals of Statistics**, JSTOR, p. 1391–1411, 1992. Citado na página 26.
- LOUZADA, F.; CUMINATO, J. A.; RODRIGUEZ, O. M.; TOMAZELLA, V. L.; FERREIRA, P. H.; RAMOS, P. L.; NIAKI, S. R.; GONZATTO, O. A.; PERISSINI, I. C.; ALEGRIA, L. F. *et al.* A repairable system subjected to hierarchical competing risks: Modeling and applications. **IEEE Access**, IEEE, v. 7, p. 171707–171723, 2019. Citado na página 24.
- MATTHEWS, D. E.; FAREWELL, V. T. On testing for a constant hazard against a change-point alternative. **Biometrics**, JSTOR, p. 463–468, 1982. Citado na página 26.
- NASH, J. C.; VARADHAN, R.; GROTHENDIECK, G. **Package ‘optimr’: A Replacement and Extension of the ‘optim’ Function**. [S.l.]: CRAN, 2019. Citado nas páginas 59, 86 e 90.
- OLIVEIRA, M. D. D.; COLOSIMO, E. A.; GILARDONI, G. L. Power law selection model for repairable systems. **Communications in Statistics-Theory and Methods**, Taylor & Francis, v. 42, n. 4, p. 570–578, 2013. Citado nas páginas 45 e 46.

OLIVEIRA, M. D. de; COLOSIMO, E. A.; GILARDONI, G. L. Bayesian inference for power law processes with applications in repairable systems. **Journal of Statistical Planning and Inference**, Elsevier, v. 142, n. 5, p. 1151–1160, 2012. Citado nas páginas 36, 37, 38, 45 e 46.

PAN, R.; RIGDON, S. E. Bayes inference for general repairable systems. **Journal of Quality Technology**, Taylor & Francis, v. 41, n. 1, p. 82–94, 2009. Citado na página 25.

PHAM, H. Introduction. In: _____. **System Software Reliability**. London: Springer London, 2006. p. 1–7. ISBN 978-1-84628-295-9. Disponível em: <https://doi.org/10.1007/1-84628-295-0_1>. Citado na página 23.

PROSCHAN, F. Theoretical explanation of observed decreasing failure rate. **Technometrics**, Taylor & Francis, v. 5, n. 3, p. 375–383, 1963. Citado na página 95.

R Core Team. **R: A Language and Environment for Statistical Computing**. Vienna, Austria, 2021. Disponível em: <<https://www.R-project.org/>>. Citado nas páginas 58, 59, 86 e 90.

RICHARDSON, M.; BASU, A. Inferences on the parameters and system reliability for a failure-truncated power law process: A bayesian approach using a change-point. **International Journal of Reliability, Quality and Safety Engineering**, World Scientific, v. 11, n. 02, p. 175–185, 2004. Citado nas páginas 27, 28, 67 e 76.

RIGDON, S. E.; BASU, A. P. **Statistical methods for the reliability of repairable systems**. [S.l.]: Wiley New York, 2000. Citado nas páginas 31, 33, 56 e 84.

ROSS, S. **STOCHASTIC PROCESSES, 2ND ED**. Wiley India Pvt. Limited, 1996. (Wiley series in probability and statistics). ISBN 9788126517572. Disponível em: <<https://books.google.com.br/books?id=HVHqPgAACAAJ>>. Citado na página 31.

SAID, U.; TAGHIPOUR, S. Modeling failure and maintenance effects of a system subject to multiple preventive maintenance types. In: IEEE. **2016 Annual Reliability and Maintainability Symposium (RAMS)**. [S.l.], 2016. p. 1–7. Citado na página 25.

SYAMSUNDAR, A.; NAIKAN, V. Hierarchical segmented point process models with multiple change points for maintained systems. **International Journal of Reliability, Quality and Safety Engineering**, World Scientific, v. 15, n. 03, p. 261–304, 2008. Citado nas páginas 27, 28, 44, 67, 76, 92, 93, 95 e 96.

SYAMSUNDAR, A.; NAIKAN, V.; WU, S. Extended arithmetic reduction of age models for the failure process of a repairable system. **Reliability Engineering System Safety**, v. 215, p. 107875, 2021. ISSN 0951-8320. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S095183202100394X>>. Citado nas páginas 24, 25, 43, 92 e 93.

TOLEDO, M. L. G. d.; FREITAS, M. A.; COLOSIMO, E. A.; GILARDONI, G. L. Optimal periodic maintenance policy under imperfect repair: A case study on the engines of off-road vehicles. **IIE Transactions**, Taylor & Francis, v. 48, n. 8, p. 747–758, 2016. Citado na página 25.

WANG, Y.; LU, Z.; XIAO, S. Parametric bootstrap confidence interval method for the power law process with applications to multiple repairable systems. **IEEE Access**, IEEE, v. 6, p. 49157–49169, 2018. Citado na página 24.

WEST, W. R.; OGDEN, T. R. Continuous-time estimation of a change-point in a poisson process. **Journal of Statistical Computation and Simulation**, Taylor & Francis, v. 56, n. 4, p. 293–302, 1997. Citado nas páginas 28, 67 e 75.

YAO, Y.-C. Maximum likelihood estimation in hazard rate models with a change-point. **Communications in Statistics-Theory and Methods**, Taylor & Francis, v. 15, n. 8, p. 2455–2466, 1986. Citado na página 26.

ZACKS, S. Classical and bayesian approaches to the change-point problem: fixed sample and sequential procedures. **Statistique et analyse des données**, v. 7, n. 1, p. 48–81, 1982. Citado na página 26.

_____. Estimating the shift to wear-out of systems having exponential-weibull life distributions. **Operations research, INFORMS**, v. 32, n. 3, p. 741–749, 1984. Citado na página 26.

ZHENG, Y.; WEI, J.; ZHU, K.; DONG, B. Reliability analysis assessment of the wind turbines system under multi-dimensions. **Advanced Composites Letters**, SAGE Publications Sage UK: London, England, v. 29, p. 2633366X20966337, 2020. Citado na página 24.

PRINCIPAIS CÓDIGOS IMPLEMENTADOS NO R

A.1 ARAM₁

A.1.1 *Pacotes e funções gerais*

```
1: # Pacotes
2: library(tidyverse)
3: library(dplyr)
4: library(plyr)
5: library(data.table)
6: library(ggplot2)
7: library(latex2exp)
8: library(hrbrthemes)
9: library(cowplot)
10: library(optimr)
11: library(plotly)
12: library(optimx)
13: library(readxl)
14: library(Rmpfr)
15:
16: # Funções
17:
18: lgi<-function(x){if(x>0 & is.finite(x)){log(x)}else{0}}
19: lgi<-Vectorize(lgi)
20: lg<-function(x){apply(as.matrix(x),2,lgi)}
21:
22: logit_2<-function(x,b,k){
```

```

23:         if( (b<1 & x>=0) | (b>1 & x<=0) ){log((1+x)/(1-x)) *
          10^k}else{x}
24: }
25: logit_2<-Vectorize(logit_2)
26:
27: logit_3<-function(x,b,k){
28:     if(b>1 & x>0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
29:     if(b>1 & x<=0){r = x}
30:     if(b>0 & b<=1 & x>0){r = x}
31:     if(b>0 & b<=1 & x<=0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
32:     return(r)
33: }
34: logit_3<-Vectorize(logit_3)
35: # ARA1 modificado

```

A.1.2 Funções de intensidade e intensidade acumulada

```

1: # ARA1 modificado
2:
3: # Função de intensidade
4:
5: func_intensidade_aram<-function(para,t,t0,k=1,tal){
6:     beta<-para[1:k]
7:     mu_tau<-para[(k+1):(2*k)]
8:     theta1<-logit_2(para[(2*k+1):(3*k)],beta,k = 4)
9:     valor<-(mu_tau*beta*((t-sign(beta-1)*theta1*t0)^(
    beta-1)))/(tal))
10:     return(valor)
11: }
12:
13: # Função de intensidade parcial
14:
15: func_intensidade_aram_ac<-function(para,t,t0,k=1,tal){
16:     beta<-para[1:k]
17:     mu_tau<-para[(k+1):(2*k)]
18:     theta1<-logit_2(para[(2*k+1):(3*k)],beta,k =4)
19:     valor<-(mu_tau*(t-sign(beta-1)*theta1*t0)^beta)/(tal
    )
20:     return(valor)
21: }

```

A.1.3 *simulação de tempos de falha*

```

1: simula_aram<-function(trunc_time, k., beta, mu_tau, theta1){ #,
      theta=c(-0.8,-0.5,0.5,0.8,-0.8,-0.5,0.5,0.8)
2:           #options("scipen"=20,digits=20)
3:
4:           # Definições
5:           # k: total de sistemas observados na amostragem
6:           # trunc_time: Tempo de truncamento do estudo
7:
8:           # Matriz Tempo
9:
10:          matrix_time<-matrix(data=0, nrow = 10^6, ncol=k.)
11:          n_vec_time_max<-0
12:          falhas_sis<-vector()
13:
14:          # Simulando o tempo
15:
16:          for(j in 1:k.){
17:
18:              # Parametrização da simulação
19:
20:              parametros_sim<-c('Beta' = beta, 'Mu_tau' =
mu_tau, 'Theta 1' = theta1) #'Theta 2' = theta2)
21:              tempo_novo<-0
22:              tempo_acumulado<-0
23:              vetor_tempo<-0
24:              mu<-mu_tau
25:              tempo_primeira_falha<-2 # Chute inicial
26:              i<-1
27:              while(tempo_acumulado<=trunc_time){
28:                  options("scipen"=22,digits=22)
29:                  tempo=tempo_acumulado
30:                  u<-runif(1)
31:                  Fun<-function(x,y){r <- func_
intensidade_aram_ac(para=parametros_sim,t=x, t0 = y, k = k.,
      tal = trunc_time)[j];return(r)}
32:                  Fun<-Vectorize(Fun)
33:                  simula<-function(par){
34:                      z<-par[1]
35:                      resultado <- 10^8*
abs(Fun(x = tempo+z, y=tempo)-Fun(x = tempo, y = tempo))+log

```


A.1.4 Função para aplicação

```

1: aplicacao.aram<-function(dado_aplica = dado_aplica, tal,
  unidade = 1, method = 'eqm', format.model = 'row', grafico
  , um_sistema = FALSE, uma_trinca_geral = FALSE){
2:     # print('Organizando os dados')
3:     options("scipen"=20,digits=20)
4:     if(format.model == 'col'){
5:
6:         if(um_sistema){
7:             dado_aplica=matrix(c(dado_
aplica,1:length(dado_aplica)), ncol=2)
8:         }
9:
10:        dado_aplica <- reshape2::melt(data = dado
_aplica)
11:        dado_aplica$Var1<-NULL
12:        colnames(dado_aplica)<-c('Unidade','Tempo
')
13:        dado_aplica<-dado_aplica |>
14:            dplyr::filter(Tempo!=0)
15:        dado_aplica$Evento<-1
16:
17:        dado_aplica$Tn<-unlist(tapply(dado_aplica
$Tempo,dado_aplica$Unidade, function(x) dplyr::lag(x,n=1,
default = 0)))
18:
19:    }
20:
21:    if(method == 'mlp'){
22:        if(um_sistema){
23:            dado_aplica <-dado_aplica|>
24:                dplyr::filter(
Evento==1 & Unidade == 1)
25:        }else{
26:            dado_aplica <-dado_aplica|>
27:                dplyr::filter(
Evento==1)
28:        }
29:
30:        posicao_max_tempo<-as.matrix(tapply(dado_
aplica$Tn, dado_aplica$Unidade, length))

```

```

31:         k..<-max(dado_aplica$Unidade)
32:         #--
33:         #Matriz T
34:         df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) x*1)
35:         df_time<-as.data.frame(plyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
36:         df_time$.id<-NULL
37:         df_time<-t(df_time);colnames(df_time)<-as
.character(1:length(posicao_max_tempo))
38:         pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))
39:         matriz_tempo<-data.table::as.data.table(
df_time)
40:         matriz_tempo[is.na(matriz_tempo)]<-0
41:         #--
42:
43:         #--
44:         #Matriz Tn
45:         df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) dplyr::lag(x,n=1,default = 0))
46:         df_time<-as.data.frame(plyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
47:         df_time$.id<-NULL
48:         df_time<-t(df_time);colnames(df_time)<-as
.character(1:length(posicao_max_tempo))
49:         pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))
50:         lag_matrix_time<-data.table::as.data.
table(df_time)
51:         lag_matrix_time[is.na(lag_matrix_time)]<-
0
52:         #--
53:
54:         #Matriz Tn
55:         df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) dplyr::lag(x,n=2,default = 0))
56:         df_time<-as.data.frame(plyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
57:         df_time$.id<-NULL

```

```

58:             df_time<-t(df_time);colnames(df_time)<-as
               .character(1:length(posicao_max_tempo))
59:             pri_time<-as.data.frame(t(as.matrix(
               tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
               [1]}))))
60:             lag2_matrix_time<-data.table::as.data.
               table(df_time)
61:             lag2_matrix_time[is.na(lag2_matrix_time)]
               <-0
62:
63:             tempo_trunca<-tal
64:             tempo_final_sistemas<-apply(matriz_tempo,
               2, max)
65:             matriz_tempo<-list(matriz_tempo,lag_
               matrix_time)
66:
67:             log_L_PLP<-function(par, time_truca = as.
               vector(tempo_trunca) , matrix_time = matriz_tempo, k. = k.,
               len. = posicao_max_tempo){
68:                 log.loop<-0
69:                 penalidade_eqm.loop<-0
70:                 if(uma_trinca_geral==TRUE){k
               .=1}
71:                 NN<-1:max(posicao_max_tempo)
72:                 #time_1 = matrix_time[[3]]
73:                 time0 = matrix_time[[2]]
74:                 time1 = matrix_time[[1]]
75:
76:                 # Função do loop da
               intensidade
77:                 fxyz1<-function(x,y) {func_
               intensidade_aram(para=par, t = x, t0 = y, tal = time_truca,
               k = k.)}
78:                 #fxyz1<-Vectorize(fxyz1)
79:                 log.loop1 <- sum(lg(fxyz1(x=
               time1,y=time0))) #sum(lg(fxyz1(x=time1,y=time0)[,1]))
80:                 fxyz2<-function(x,y) {func_
               intensidade_aram_ac(para=par, t = x, t0 = y, tal = time_
               truca, k = k.)}
81:                 #fxyz2<-Vectorize(fxyz2)
82:                 log.loop2 <- sum(fxyz2(x=time1
               ,y=time0)) #sum(fxyz2(x=time1,y=time0)[,1])

```

```

83:             log.loop3 <- sum(fxyz2(x=time0
, y=time0))#sum(fxyz2(x=time0, y=time0)[,1])
84:
85:             # Valor da acumulada
86:             log.acumulada1<-sum((func_
intensidade_aram_ac(para=par, t=time_truca, t0 = tempo_final
_sistemas , tal=time_truca, k = k.)))
87:             log.acumulada2<-sum((func_
intensidade_aram_ac(para=par, t=tempo_final_sistemas, t0 =
tempo_final_sistemas , tal=time_truca, k = k.)))
88:
89:             l<-log.loop1-(log.loop2-log.
loop3) - (log.acumulada1-log.acumulada2)
90:
91:             return(-l)
92:             print(-l)
93:         }
94:         if(uma_trinca_geral==TRUE){k=1}else{k=k
..}
95:
96:         par0<-c(rep(0.5,k),rep(5,k),rep(0.8,k))#,
rep(0.3,k))
97:
98:         # estima_<-optimr::optimr(par=par0, fn=
log_L_PLP, hessian=TRUE, lower = c(rep(-100,k), rep(-100,k)
, rep(-100,k)), upper = c(rep(100,k), rep(100,k), rep(0,k)),
method="L-BFGS-B")#, upper = c(rep(Inf,2*k)), method="L-
BFGS-B")
99:
100:         estima_<-optimr::optimr(par=par0, fn=log_
L_PLP, hessian=TRUE, lower = c(rep(0.001,k), rep(0.001,k),
rep(0,k)), upper = c(rep(Inf,k), rep(Inf,k), rep
(1-0.1^10,k)), method="L-BFGS-B", control = list(all.methods=
TRUE,lmm = 10,factr=1e9,pgtol=1,maxit = 1000000) )#, upper
= c(rep(Inf,2*k)), method="L-BFGS-B")
101:         estima_2<-optimr::optimr(par=par0, fn=log
_L_PLP, hessian=TRUE, lower = c(rep(0.001,k), rep(0.001,k),
rep(-1,k)), upper = c(rep(1,k), rep(Inf,k), rep
(0,k)), method="L-BFGS-B", control = list(all.methods= TRUE,
lmm = 10,factr=1e9,pgtol=1,maxit = 1000000))#, upper = c(rep
(Inf,2*k)), method="L-BFGS-B")}

```

```

102:         if(is.na(estima_2$par[1])){estima_2$value
           = 10000}
103:         estima_3<-optimr::optimr(par=par0, fn=log
_L_PLP, hessian=TRUE, lower = c(rep(1+0.1^10,k), rep(0.001,
k), rep(-1+0.1^10,k)), upper = c(rep(Inf,k), rep(Inf,k), rep
(0,k)), method="L-BFGS-B", control = list(all.methods= TRUE,
lmm = 10, factr=1e9, pgtol=1, maxit = 1000000))#, upper = c(rep
(Inf,2*k)), method="L-BFGS-B")}
104:         if(is.na(estima_3$par[1])){estima_3$value
           = 10000}
105:         estima_1=estima_
106:
107:
108:         if( (estima_$value < estima_2$value) & (
estima_$value < estima_3$value) ){estima_ = estima_} }
109:         if( (estima_2$value < estima_$value) & (
estima_2$value < estima_3$value) ){estima_ = estima_2} }
110:         if( (estima_3$value < estima_$value) & (
estima_3$value < estima_2$value) ){estima_ = estima_3} }
111:
112:         parametros_estimados<-c(estima_$par)
113:         names(parametros_estimados)<-c(paste0('
beta',1:k),paste0('mu_tau',1:k),paste0('theta1',1:k))#,
paste0('theta2',1:k))
114:
115:         M.rm = estima_$hessian
116:         vcov.rm = solve(M.rm)
117:         sd_<-diag(sqrt(vcov.rm))
118:         valor_p=2*(1-pnorm(estima_$par/sd_))
119:         IC_95<-cbind(estima_$par,estima_$par-1.96
*sd_,estima_$par+1.96*sd_)
120:     }
121:
122:     if(grafico == 1){
123:         lista_de_graficos = list()
124:         for(unidade in 1:max(dado_aplica$Unidade)
){
125:             # Gráficos #
126:
127:             k.<-1
128:             dado_aplica_select <- dado_
aplica |>

```

```

129:                                     dplyr::filter(
      Unidade==unidade)
130:
131:                                     tempo<-dado_aplica_select$
      Tempo[order(dado_aplica_select$Tempo)]
132:
133:                                     dados<-data.frame(y=1:length(
      tempo), t=tempo, t0=dplyr::lag(tempo,n = 1, default = 0))
134:                                     if(uma_trinca_geral==TRUE){par
      _estimado=estima_$par}else{
135:                                     par_estimado<-
      estima_$par[c(unidade,max(dado_aplica$Unidade)+unidade,2*max
      (dado_aplica$Unidade)+unidade)]
136:                                     }
137:
138:                                     # sistema<-unidade
139:                                     vetor_tempo<-tempo
140:                                     trun=tal
141:                                     if(length(tempo_trunca)==1){
      unidade=1;sistema=1}
142:                                     Fun1.1<-function(x){
143:                                     r<-func_intensidade
      _aram_ac(para=par_estimado,t=x,
144:
      t0 = if(length(vetor_tempo[vetor_tempo<=x])==0){0}
      else{max(vetor_tempo[vetor_tempo<=x])},
145:
      k = k., tal = tempo_trunca[unidade]);
146:                                     return(r)
147:                                     }
148:
149:                                     Fun1.1<-Vectorize(Fun1.1)
150:
151:                                     Fun1.10<-function(x){
152:                                     r<-func_intensidade
      _aram_ac(para=par_estimado,t=x,
153:
      t0 = if(length(vetor_tempo[vetor_tempo<x])==0){0}
      else{max(vetor_tempo[vetor_tempo<x])},
154:
      k = k., tal = tempo_trunca[unidade]);
155:                                     return(r)

```

```

156:         }
157:
158:         Fun1.10<-Vectorize(Fun1.10)
159:
160:         Fun1.1.1<-function(x){
161:             t = vetor_tempo[
vetor_tempo<=x]
162:             t0 = dplyr::lag(
vetor_tempo[vetor_tempo<=x],n = 1, default = 0)
163:             r <- sum(Fun1.10(t)
-Fun1.1(t0)) + ifelse( !(x%in%vetor_tempo), (Fun1.10(x)-Fun1
.1(max(vetor_tempo[vetor_tempo<=x]))), 0) + ifelse( x>tempo_
trunca, (Fun1.10(x)-Fun1.1(max(vetor_tempo[vetor_tempo<=x]))
), 0)
164:             return(r)}
165:         Fun1.1.1<-Vectorize(Fun1.1.1)
166:
167:         Fun2.1<-function(x){
168:             r<-func_intensidade
_aram(para=par_estimado,t=x,
169:
             t0 = if(length(vetor_tempo[vetor_tempo<x])==0){0}else{
max(vetor_tempo[vetor_tempo<x])},
170:
             k = k., tal = tempo_trunca[unidade]);
171:             return(r)
172:         }
173:
174:         Fun2.1<-Vectorize(Fun2.1)
175:
176:
177:         x=c(seq(min(vetor_tempo[vetor_
tempo!=0]), max(vetor_tempo),0.1),vetor_tempo)
178:
179:         # Acumulada observada
180:
181:         df3<-data.frame(f2=1:length(
vetor_tempo),x=vetor_tempo,Color='DR')
182:
183:         if(length(tempo_trunca)!=1){
184:             df1.1<-data.frame(
f2=Fun1.1.1(vetor_tempo)[1,],x=vetor_tempo,Color='MA')

```



```

185:                                     df1.1x<-data.frame(
      f2=Fun1.1.1(x)[1,],x=x,Color='MA')
186:                                     }else{
187:                                     df1.1<-data.frame(
      f2=Fun1.1.1(vetor_tempo),x=vetor_tempo,Color='MA')
188:                                     df1.1x<-data.frame(
      f2=Fun1.1.1(x),x=x,Color='MA')
189:                                     }
190:                                     df5.1<-data.frame(f2=Fun2.1(
      vetor_tempo),x=vetor_tempo,Color='red')  # Função estimada
      nos pontos de tempo
191:                                     df5.1x<-data.frame(f2=Fun2.1(x
      ), x=x, Color='red')  # Função estimada na
      grade
192:
193:                                     cols2 <- c("DR"="red","MA"="
      blue")
194:
195:                                     # Intensidade Acumulada
196:                                     plot1.a<-ggplot(df3,aes(x,f2,
      color = Color))  +
197:                                     geom_line(data =
      df3, aes(x,f2,color = Color),size=0.55) + geom_point(data =
      df3, aes(x,f2,color = Color),size=1.5)+
198:                                     geom_point(data =
      df1.1, aes(x,f2,color = Color),size=1.5) + geom_line(data =
      df1.1x, aes(x,f2,color = Color), size=0.55,linetype = 2) +
199:                                     theme_bw() +
200:                                     #ggtitle(TeX(r'($\
      \hat{\beta} = 0.71$, $\hat{\theta} = 0.83$ e $\hat{\mu}_{\tau}
      = 29.99$)')) +
201:                                     xlab(TeX(r'($t$)'))
      + ylab(TeX(r'($\hat{\Lambda}(t)$)')) + scale_colour_manual
      (name = '',values=cols2)
202:                                     plot1.a
203:
204:                                     # Intensidade
205:                                     plot1.es <- ggplot(df5.1x,aes(
      x,f2,color = Color))  + geom_point(data = df5.1, aes(x,f2,
      color = Color),size=1.5) + geom_line(aes(color = 1), size
      =0.4)+ theme_bw() +

```

```

206:                                     scale_color_
      identity()+ xlab(TeX(r'($t$)')) + ylab(TeX(r'($\hat{\lambda}
      }(t)$)'))
207:                                     plot1.es
208:                                     options("scipen"=20,digits=4)
209:                                     plot_grid<-plot_grid(plotlist
= list(plot1.a,plot1.es),ncol=2,nrow = 1, labels = c('1'),'
2)')
210:
211:                                     lista_de_graficos[[unidade]] =
      plot_grid
212:                                     }
213:                                     return(list('parâmetros estimados' =
estima_, 'parâmetros estimados2' = list(estima_1,estima_2,
estima_3), 'Gráfico duplo' = lista_de_graficos,'Gráfico
duplo AC' = list(plot1.a),'Gráfico duplo 1' = list(plot1.es)
, 'intervalo de confiança' = IC_95)) #'Modelo Estimado -
Intesidade acumulada' = plot1.a, 'Intensidade' = plot1.es,
214:                                     }else{
215:                                     return(list('parâmetros estimados' =
estima_, 'intervalo de confiança' = IC_95))
216:                                     }
217: }

```

A.1.5 Estudo de Simulação

```

1: options("scipen"=20,digits=20)
2:
3: # Simulação I
4:
5: # 5 sistemas
6:
7: M=1000
8: res = matrix(0, nrow = M, ncol = 3)
9: time.i=Sys.time()
10: i=1#;k=1
11: while (dim(matrix(res[res[,3]!=0 ],,ncol=3))[1] != M) {
12:     k...=5
13:     b=rep(0.8,k...); m=rep(50,k...); t1.= rep(0.6,k...);
      tal.= 50

```

```

14:         dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
    =b, mu_tau=m, theta1= t1.)
15:         resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
    Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
    format.model = 'col', grafico=1, um_sistema = FALSE, uma_
    trinca_geral = TRUE)#,par_original = c(b,m,t))
16:
17:         #resul_sim
18:         res[i,]<-c(resul_sim$`parâmetros estimados`$par)
19:         time.f = Sys.time()
20:         if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
21:             res1 = as.data.frame(na.exclude(res))
22:             print(list('Matriz completa' = res[res
    [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
    valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
    estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
    !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
    !=1 &
23:         }
24:         i<-i+1
25: }
26:
27: write.csv2(as.data.frame(res),"resultado.resumo.0.8_50_0.6_T_50
    _5.csv")
28:
29: # 10 sistemas
30:
31: M=1000
32: res = matrix(0, nrow = M, ncol = 3)
33: time.i=Sys.time()
34: i=1#;k=1
35: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
36:     k...=10
37:     b=rep(0.8,k...); m=rep(50,k...); t1.= rep(0.6,k...);
    tal.= 50
38:     dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
    =b, mu_tau=m, theta1= t1.)
39:     resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
    Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
    format.model = 'col', grafico=1, um_sistema = FALSE, uma_
    trinca_geral = TRUE)#,par_original = c(b,m,t))
40:

```

```

41:         #resul_sim
42:         res[i,]<-c(resul_sim$`parâmetros estimados`$par)
43:         time.f = Sys.time()
44:         if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
45:             res1 = as.data.frame(na.exclude(res))
46:             print(list('Matriz completa' = res[res
[ ,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
!=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
!=1 &
47:         }
48:         i<-i+1
49: }
50:
51: write.csv2(as.data.frame(res),"resultado.resumo.0.8_50_0.6_T_50
_10.csv")
52:
53: # 30 sistemas
54:
55: M=1000
56: res = matrix(0, nrow = M, ncol = 3)
57: time.i=Sys.time()
58: i=1#;k=1
59: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
60:     k...=30
61:     b=rep(0.8,k...); m=rep(50,k...); t1.= rep(0.6,k...);
tal.= 50
62:     dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
=b, mu_tau=m, theta1= t1.)
63:     resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
format.model = 'col', grafico=1, um_sistema = FALSE, uma_
trinca_geral = TRUE)#,par_original = c(b,m,t))
64:
65:         #resul_sim
66:         res[i,]<-c(resul_sim$`parâmetros estimados`$par)
67:         time.f = Sys.time()
68:         if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
69:             res1 = as.data.frame(na.exclude(res))
70:             print(list('Matriz completa' = res[res
[ ,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '

```

```

        valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
        estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
        !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
        !=1 &
71:         }
72:         i<-i+1
73: }
74:
75: write.csv2(as.data.frame(res),"resultado.resumo.0.8_50_0.6_T_50
        _30.csv")
76:
77:
78: # 50 sistemas
79:
80: M=1000
81: res = matrix(0, nrow = M, ncol = 3)
82: time.i=Sys.time()
83: i=1#;k=1
84: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
85:         k...=50
86:         b=rep(0.8,k...); m=rep(50,k...); t1.= rep(0.6,k...);
        tal.= 50
87:         dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
        =b, mu_tau=m, theta1= t1.)
88:         resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
        Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
        format.model = 'col', grafico=1, um_sistema = FALSE, uma_
        trinca_geral = TRUE)#,par_original = c(b,m,t))
89:
90:         #resul_sim
91:         res[i,]<-c(resul_sim$`parâmetros estimados`$par)
92:         time.f = Sys.time()
93:         if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
94:                 res1 = as.data.frame(na.exclude(res))
95:                 print(list('Matriz completa' = res[res
        [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
        valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
        estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
        !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
        !=1 &
96:         }
97:         i<-i+1

```

```

98: }
99:
100: write.csv2(as.data.frame(res),"resultado.resumo.0.8_50_0.6_T_50
      _50.csv")
101:
102:
103:
104: # Simulação II
105:
106: # 5 sistemas
107:
108: M=1000
109: res = matrix(0, nrow = M, ncol = 3)
110: time.i=Sys.time()
111: i=1#;k=1
112: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
113:     k...=5
114:     b=rep(1.1,k...); m=rep(10,k...); t1.= rep(0.3,k...);
      tal.= 500
115:     dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
      =b, mu_tau=m, theta1= t1.)
116:     resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
      Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
      format.model = 'col', grafico=1, um_sistema = FALSE, uma_
      trinca_geral = TRUE)#,par_original = c(b,m,t))
117:
118:     #resul_sim
119:     res[i,]<-c(resul_sim$`parâmetros estimados`$par)
120:     time.f = Sys.time()
121:     if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
122:         res1 = as.data.frame(na.exclude(res))
123:         print(list('Matriz completa' = res[res
      [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
      valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
      estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
      !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
      !=1 &
124:     }
125:     i<-i+1
126: }
127:

```

```

128: write.csv2(as.data.frame(res),"resultado.resumo.1.1_10_0.3_T_
      500_5.csv")
129:
130: # 10 sistemas
131:
132: M=1000
133: res = matrix(0, nrow = M, ncol = 3)
134: time.i=Sys.time()
135: i=1#;k=1
136: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
137:     k...=10
138:     b=rep(1.1,k...); m=rep(10,k...); t1.= rep(0.3,k...);
      tal.= 500
139:     dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
      =b, mu_tau=m, theta1= t1.)
140:     resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
      Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
      format.model = 'col', grafico=1, um_sistema = FALSE, uma_
      trinca_geral = TRUE)#,par_original = c(b,m,t))
141:
142:     #resul_sim
143:     res[i,]<-c(resul_sim$`parâmetros estimados`$par)
144:     time.f = Sys.time()
145:     if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
146:         res1 = as.data.frame(na.exclude(res))
147:         print(list('Matriz completa' = res[res
      [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
      valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
      estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
      !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
      !=1 &
148:     }
149:     i<-i+1
150: }
151:
152: write.csv2(as.data.frame(res),"resultado.resumo.1.1_10_0.3_T_
      500_10.csv")
153:
154: # 30 sistemas
155:
156: M=1000
157: res = matrix(0, nrow = M, ncol = 3)

```

```

158: time.i=Sys.time()
159: i=1#;k=1
160: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
161:     k...=30
162:     b=rep(1.1,k...); m=rep(10,k...); t1.= rep(0.3,k...);
        tal.= 500
163:     dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
        =b, mu_tau=m, theta1= t1.)
164:     resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
        Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
        format.model = 'col', grafico=1, um_sistema = FALSE, uma_
        trinca_geral = TRUE)#,par_original = c(b,m,t))
165:
166:     #resul_sim
167:     res[i,]<-c(resul_sim$`parâmetros estimados`$par)
168:     time.f = Sys.time()
169:     if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
170:         res1 = as.data.frame(na.exclude(res))
171:         print(list('Matriz completa' = res[res
            [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
            valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
            estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
            !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
            !=1 &
172:     }
173:     i<-i+1
174: }
175:
176: write.csv2(as.data.frame(res),"resultado.resumo.1.1_10_0.3_T_
        500_30.csv")
177:
178:
179: # 50 sistemas
180:
181: M=1000
182: res = matrix(0, nrow = M, ncol = 3)
183: time.i=Sys.time()
184: i=1#;k=1
185: while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M) {
186:     k...=50
187:     b=rep(1.1,k...); m=rep(10,k...); t1.= rep(0.3,k...);
        tal.= 500

```



```

188:         dado_sim<-simula_aram(trunc_time=tal., k.=k..., beta
      =b, mu_tau=m, theta1= t1.)
189:         resul_sim<-aplicacao.aram(dado_aplica = dado_sim$`
      Matriz de tempo`, tal = tal., unidade = 1, method = 'mlp',
      format.model = 'col', grafico=1, um_sistema = FALSE, uma_
      trinca_geral = TRUE)#,par_original = c(b,m,t))
190:
191:         #resul_sim
192:         res[i,]<-c(resul_sim$`parâmetros estimados`$par)
193:         time.f = Sys.time()
194:         if(dim(matrix(res[res[,3]!=0 ,],ncol=3))[1]>=2){
195:             res1 = as.data.frame(na.exclude(res))
196:             print(list('Matriz completa' = res[res
      [,2]>0,], 'Estimativa' = colMeans(res1[res1[,2]>0,]), '
      valor do i' = i , 'time loop' =time.f-time.i, 'Média p/
      estimativa' = (time.f-time.i)/i )) # colMeans(res1[ res1[,2]
      !=0 & res1[,1]<2 & res1[,2]<100 ,]) res1[,3]!=0 & res1[,3]
      !=1 &
197:         }
198:         i<-i+1
199: }
200:
201: write.csv2(as.data.frame(res),"resultado.resumo.1.1_10_0.3_T_
      500_50.csv")

```

A.1.6 Aplicação em dados reais

```

1:
2:
3: # Função de intensidade
4:
5: func_intensidade_aram<-function(para,t,t0,k=1,tal){
6:     beta<-para[1:k]
7:     mu_tau<-para[(k+1):(2*k)]
8:     theta1<-logit_3(para[(2*k+1):(3*k)],beta,k = 4)
9:     valor<-(mu_tau*beta*((t-sign(beta-1)*theta1*t0)^(
      beta-1)))/(tal))
10:     return(valor)
11: }
12:
13: # Função de intensidade parcial

```

```

14:
15: func_intensidade_aram_ac<-function(para,t,t0,k=1,tal){
16:     beta<-para[1:k]
17:     mu_tau<-para[(k+1):(2*k)]
18:     theta1<-logit_3(para[(2*k+1):(3*k)],beta,k=4)
19:     valor<-(mu_tau*(t-sign(beta-1)*theta1*t0)^beta)/(tal
    )
20:     return(valor)
21: }
22:
23: aplicacao.aram<-function(dado_aplica = dado_aplica, tal,
    unidade = 1, method = 'eqm', format.model = 'row', grafico
    , um_sistema = FALSE, uma_trinca_geral = FALSE, limites_par
    = 1){
24:     # print('Organizando os dados')
25:     options("scipen"=20,digits=20)
26:     if(format.model == 'col'){
27:         # b=1.2;m=30;t=0.4
28:         # dado_sim<-simula_PLP_theta_eta_SAR_
    redu2(trunc_time=tal, k=1, beta=b, mu_tau=m, theta= t,
    controle_amostra=500)
29:         # dado_sim<-matrix(c(dado_sim$`Matriz de
    tempo`,1:length(dado_sim$`Matriz de tempo`)), ncol=2) ; dado
    _aplica = simula_plpg_n$`Matriz de tempo` ; dado_aplica =
    matriz_simula_plpg_n ; dado_aplica = dados.colheitadere ;
    dado_aplica =df.colhe$Tempo
30:         # par_original = c(b,m,t)
31:
32:         if(um_sistema){
33:             dado_aplica=matrix(c(dado_
    aplica,1:length(dado_aplica)), ncol=2)
34:         }
35:
36:         dado_aplica <- reshape2::melt(data = dado
    _aplica)
37:         dado_aplica$Var1<-NULL
38:         colnames(dado_aplica)<-c('Unidade','Tempo
    ')
39:         dado_aplica<-dado_aplica |>
40:             dplyr::filter(Tempo!=0)
41:         dado_aplica$Evento<-1
42:

```

```

43:         dado_aplica$Tn<-unlist(tapply(dado_aplica
$Tempo,dado_aplica$Unidade, function(x) dplyr::lag(x,n=1,
default = 0)))
44:
45:     }
46:
47:
48:     if(method == 'mlp'){
49:         # print('Realizando as estimativas - mlp
')
50:
51:         if(um_sistema){
52:             dado_aplica <-dado_aplica|>
53:                 dplyr::filter(
Evento==1 & Unidade == 1)
54:         }else{
55:             dado_aplica <-dado_aplica|>
56:                 dplyr::filter(
Evento==1)
57:         }
58:
59:         posicao_max_tempo<-as.matrix(tapply(dado_
aplica$Tn, dado_aplica$Unidade, length))
60:         k..<-max(dado_aplica$Unidade)
61:         #--
62:         #Matriz T
63:         df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) x*1)
64:         df_time<-as.data.frame(plyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
65:         df_time$.id<-NULL
66:         df_time<-t(df_time);colnames(df_time)<-as
.character(1:length(posicao_max_tempo))
67:         pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))
68:         matriz_tempo<-data.table::as.data.table(
df_time)
69:         matriz_tempo[is.na(matriz_tempo)]<-0
70:         #--
71:
72:         #--

```

```

73:             #Matriz Tn
74:             df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) dplyr::lag(x,n=1,default = 0))
75:             df_time<-as.data.frame(dplyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
76:             df_time$.id<-NULL
77:             df_time<-t(df_time);colnames(df_time)<-as
.character(1:length(posicao_max_tempo))
78:             pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))
79:             lag_matrix_time<-data.table::as.data.
table(df_time)
80:             lag_matrix_time[is.na(lag_matrix_time)]<-
0
81:             #--
82:
83:             #Matriz Tn
84:             df_time<-tapply(dado_aplica$Tempo,dado_
aplica$Unidade,function(x) dplyr::lag(x,n=2,default = 0))
85:             df_time<-as.data.frame(dplyr::ldply(df_
time,function(s){t(data.frame(unlist(s)))}))
86:             df_time$.id<-NULL
87:             df_time<-t(df_time);colnames(df_time)<-as
.character(1:length(posicao_max_tempo))
88:             pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))
89:             lag2_matrix_time<-data.table::as.data.
table(df_time)
90:             lag2_matrix_time[is.na(lag2_matrix_time)]
<-0
91:             #--
92:
93:             tempo_trunca<-tal
94:             tempo_final_sistemas<-apply(matriz_tempo,
2, max)
95:             #tempo_final_sistemas2<-apply(lag2_matrix
_time, 2, max)
96:             matriz_tempo<-list(matriz_tempo,lag_
matrix_time)
97:

```

```

98:
99:
100:         log_L_PLP<-function(par, time_truca = as.
vector(tempo_trunca) , matrix_time = matriz_tempo, k. = k.,
        len. = posicao_max_tempo){
101:             log.loop<-0
102:             penalidade_eqm.loop<-0
103:             if(uma_trinca_geral==TRUE){k
.=1}
104:             NN<-1:max(posicao_max_tempo)
105:             #time_1 = matrix_time[[3]]
106:             time0 = matrix_time[[2]]
107:             time1 = matrix_time[[1]]
108:
109:             # Função do loop da
intensidade
110:             fxyz1<-function(x,y) {func_
intensidade_aram(para=par, t = x, t0 = y, tal = time_truca,
k = k.)}
111:             #fxyz1<-Vectorize(fxyz1)
112:             log.loop1 <- sum(lg(fxyz1(x=
time1,y=time0))) #sum(lg(fxyz1(x=time1,y=time0)[,1]))
113:             fxyz2<-function(x,y) {func_
intensidade_aram_ac(para=par, t = x, t0 = y, tal = time_
truca, k = k.)}
114:             #fxyz2<-Vectorize(fxyz2)
115:             log.loop2 <- sum(fxyz2(x=time1
,y=time0))#sum(fxyz2(x=time1,y=time0)[,1])
116:             log.loop3 <- sum(fxyz2(x=time0
,y=time0))#sum(fxyz2(x=time0,y=time0)[,1])
117:
118:             # Valor da acumulada
119:             log.acumulada1<-sum((func_
intensidade_aram_ac(para=par, t=time_truca, t0 = tempo_final
_sistemas , tal=time_truca, k = k.)))
120:             log.acumulada2<-sum((func_
intensidade_aram_ac(para=par, t=tempo_final_sistemas, t0 =
tempo_final_sistemas , tal=time_truca, k = k.)))
121:
122:             l<-log.loop1-(log.loop2-log.
loop3) - (log.acumulada1-log.acumulada2)
123:

```

```

124:             return(-1)
125:             print(-1)
126:         }
127:         if(uma_trinca_geral==TRUE){k=1}else{k=k
    ..}
128:
129:         par0<-c(rep(0.5,k),rep(5,k),rep(0.4,k))#,
    rep(0.3,k))
130:
131:         # estima_<-optimr::optimr(par=par0, fn=
    log_L_PLP, hessian=TRUE, lower = c(rep(-100,k), rep(-100,k)
    , rep(-100,k)), upper = c(rep(100,k), rep(100,k), rep(0,k)),
    method="L-BFGS-B")#, upper = c(rep(Inf,2*k)), method="L-
    BFGS-B")
132:
133:         # Simulação
134:         # estima_<-optimr::optimr(par=par0, fn=
    log_L_PLP, hessian=TRUE, lower = c(rep(0.001,k), rep
    (0.001,k), rep(0,k)), upper = c(rep(Inf,k), rep(
    Inf,k), rep(1-0.1^10,k)), method="L-BFGS-B", control = list(
    all.methods= TRUE, lmm = 10, factr=1e9, pgtol=1, maxit =
    1000000) )#, upper = c(rep(Inf,2*k)), method="L-BFGS-B")
135:         # estima_2<-optimr::optimr(par=par0, fn=
    log_L_PLP, hessian=TRUE, lower = c(rep(0.001,k), rep(0.001,
    k), rep(-1,k)), upper = c(rep(1,k), rep(Inf,k),
    rep(0,k)), method="L-BFGS-B", control = list(all.methods=
    TRUE, lmm = 10, factr=1e9, pgtol=1, maxit = 1000000))#, upper =
    c(rep(Inf,2*k)), method="L-BFGS-B")}
136:         # if(is.na(estima_2$par[1])){estima_2$
    value = 10000}
137:         # estima_3<-optimr::optimr(par=par0, fn=
    log_L_PLP, hessian=TRUE, lower = c(rep(1+0.1^10,k), rep
    (0.001,k), rep(-1+0.1^10,k)), upper = c(rep(Inf,k), rep(Inf,
    k), rep(0,k)), method="L-BFGS-B", control = list(all.methods
    = TRUE, lmm = 10, factr=1e9, pgtol=1, maxit = 1000000))#, upper
    = c(rep(Inf,2*k)), method="L-BFGS-B")}
138:         # if(is.na(estima_3$par[1])){estima_3$
    value = 10000}
139:         # estima_1=estima_
140:
141:         # Aplicação
142:         if( limites_par == 1){

```

```

143:             estima_<- optimx::optimr(par0, log_L_PLP,
             hessian = TRUE, lower = c(rep(0.01,k), rep(0.01,k), rep(-
             Inf,k)), upper = c(rep(Inf,k), rep(Inf,k), rep(Inf,k)),
             method="L-BFGS-B")#control = list(all.methods = TRUE))#,
             upper = c(rep(Inf,2*k)), method="L-BFGS-B")
144:             }else{
145:             #para i=4 colheitadeira
146:             estima_<- optimx::optimr(par0, log_L_PLP,
             hessian = TRUE, lower = c(rep(1,k), rep(0.01,k), rep(-10^5,
             k)), upper = c(rep(10^5,k), rep(10^5,k), rep(10^5,k)),
             method="L-BFGS-B")#control = list(all.methods = TRUE))#,
             upper = c(rep(Inf,2*k)), method="L-BFGS-B")
147:             }#
148:             estima_1 = 0;estima_2=0;estima_3=0
149:             #
150:
151:             #
152:             # if( (estima_1$value < estima_2$value) &
             (estima_1$value < estima_3$value) ){estima_1 = estima_1}
153:             # if( (estima_2$value < estima_1$value) &
             (estima_2$value < estima_3$value) ){estima_1 = estima_2}
154:             # if( (estima_3$value < estima_1$value) &
             (estima_3$value < estima_2$value) ){estima_1 = estima_3}
155:
156:
157:             # estima_
158:             #
159:             parametros_estimados<-c(estima_1$par)
160:             names(parametros_estimados)<-c(paste0('
             beta',1:k),paste0('mu_tau',1:k),paste0('theta1',1:k))#,
             paste0('theta2',1:k))
161:
162:             M.rm = estima_1$hessian
163:             vcov.rm = solve(M.rm)
164:             sd_<-sqrt(diag(vcov.rm))
165:             valor_p=2*(1-pnorm(estima_1$par/sd_))
166:             IC_95<-cbind(estima_1$par,estima_1$par-1.96
             *sd_,estima_1$par+1.96*sd_)
167:             IC_95 =rbind(IC_95,2*(exp(IC_95[3,])/(exp
             (IC_95[3,])+1)-0.5))
168:             if(is.na(IC_95[4,3])){IC_95[4,3]=1}

```

```

169:         #colnames(IC_95)<-c("Estimativa","LI","LS
      ")
170:         #rownames(IC_95)<-c('beta','mu_tau','
      theta1')#,'theta2')
171:         #IC_95
172:     }
173:
174:     if(grafico == 1){
175:         lista_de_graficos = list()
176:         for(unidade in 1:max(dado_aplica$Unidade)
      ){
177:             # Gráficos #
178:             #print('Elaborando os gráficos
      ')
179:             k.<-1
180:             dado_aplica_select <- dado_
      aplica |>
181:                 dplyr::filter(
      Unidade==unidade)
182:
183:             tempo<-dado_aplica_select$
      Tempo[order(dado_aplica_select$Tempo)]
184:
185:             dados<-data.frame(y=1:length(
      tempo), t=tempo, t0=dplyr::lag(tempo,n = 1, default = 0))
186:             if(uma_trinca_geral==TRUE){par
      _estimado=estima_$par}else{
187:                 par_estimado<-
      estima_$par[c(unidade,max(dado_aplica$Unidade)+unidade,2*max
      (dado_aplica$Unidade)+unidade)]
188:             }
189:
190:             # sistema<-unidade
191:             vetor_tempo<-tempo
192:             trun=tal
193:             if(length(tempo_trunca)==1){
      unidade=1;sistema=1}
194:             Fun1.1<-function(x){
195:                 r<-func_intensidade
      _aram_ac(para=par_estimado,t=x,
196:
      t0 = if(length(vetor_tempo[vetor_tempo<=x])==0){0}

```



```

else{max(vetor_tempo[vetor_tempo<=x])},
197:
      k = k., tal = tempo_trunca[unidade]);
198:                                     return(r)
199:     }
200:
201:     Fun1.1<-Vectorize(Fun1.1)
202:
203:     Fun1.10<-function(x){
204:         r<-func_intensidade
_aram_ac(para=par_estimado,t=x,
205:
      t0 = if(length(vetor_tempo[vetor_tempo<x])==0){0}
else{max(vetor_tempo[vetor_tempo<x])},
206:
      k = k., tal = tempo_trunca[unidade]);
207:                                     return(r)
208:     }
209:
210:     Fun1.10<-Vectorize(Fun1.10)
211:
212:     Fun1.1.1<-function(x){
213:         t = vetor_tempo[
vetor_tempo<=x]
214:         t0 = dplyr::lag(
vetor_tempo[vetor_tempo<=x],n = 1, default = 0)
215:         r <- sum(Fun1.10(t)
-Fun1.1(t0)) + ifelse( !(x%in%vetor_tempo), (Fun1.10(x)-Fun1
.1(max(vetor_tempo[vetor_tempo<=x]))), 0) + ifelse( x>tempo_
trunca, (Fun1.10(x)-Fun1.1(max(vetor_tempo[vetor_tempo<=x]))
), 0)
216:                                     return(r)}
217:     Fun1.1.1<-Vectorize(Fun1.1.1)
218:
219:     Fun2.1<-function(x){
220:         r<-func_intensidade
_aram(para=par_estimado,t=x,
221:
      t0 = if(length(vetor_tempo[vetor_tempo<x])==0){0}else{
max(vetor_tempo[vetor_tempo<x])},
222:
      k = k., tal = tempo_trunca[unidade]);

```

```

223:                                     return(r)
224:     }
225:
226:     Fun2.1<-Vectorize(Fun2.1)
227:
228:
229:     x=c(seq(min(vetor_tempo[vetor_
tempo!=0]), max(vetor_tempo),0.1),vetor_tempo)
230:
231:     # Acumulada observada
232:
233:     df3<-data.frame(f2=1:length(
vetor_tempo),x=vetor_tempo,Color='DR')
234:
235:     if(length(tempo_trunca)!=1){
236:         df1.1<-data.frame(
f2=Fun1.1.1(vetor_tempo)[1,],x=vetor_tempo,Color='MA')
237:         df1.1x<-data.frame(
f2=Fun1.1.1(x)[1,],x=x,Color='MA')
238:     }else{
239:         df1.1<-data.frame(
f2=Fun1.1.1(vetor_tempo),x=vetor_tempo,Color='MA')
240:         df1.1x<-data.frame(
f2=Fun1.1.1(x),x=x,Color='MA')
241:     }
242:     df5.1<-data.frame(f2=Fun2.1(
vetor_tempo),x=vetor_tempo,Color='red') # Função estimada
nos pontos de tempo
243:     df5.1x<-data.frame(f2=Fun2.1(x
), x=x, Color='red') # Função estimada na
grade
244:
245:     cols2 <- c("DR"="red","MA"="
blue")
246:
247:     # Intensidade Acumulada
248:     plot1.a<-ggplot(df3,aes(x,f2,
color = Color)) +
249:         geom_line(data =
df3, aes(x,f2,color = Color),size=0.55) + geom_point(data =
df3, aes(x,f2,color = Color),size=1.5)+

```

```

250:                                     geom_point(data =
df1.1, aes(x,f2,color = Color),size=1.5) + geom_line(data =
df1.1x, aes(x,f2,color = Color), size=0.55,linetype = 2) +
251:                                     theme_bw() +
252:                                     #ggtitle(TeX(r'($\
hat{\beta} = 0.71$, $\hat{\theta} = 0.83$ e $\hat{\mu}_{\tau}
= 29.99$)')) +
253:                                     xlab(TeX(r'($t$)'))
+ ylab(TeX(r'($\hat{\Lambda}(t)$)')) + scale_colour_manual
(name = '',values=cols2)
254:                                     plot1.a
255:
256:                                     # Intensidade
257:                                     plot1.es <- ggplot(df5.1x,aes(
x,f2,color = Color)) + geom_point(data = df5.1, aes(x,f2,
color = Color),size=1.5) + geom_line(aes(color = 1), size
=0.4)+ theme_bw() +
258:                                     #
ggtitle(TeX(r'($\hat{\beta} = 0.71$, $\hat{\theta} =
0.83$ e $\hat{\mu}_{\tau} = 29.99$)')) +
259:                                     # xlab(TeX(r'($t$)
')) + ylab(TeX(r'($\hat{\lambda}(t)$)'))+
260:                                     scale_color_
identity()+ xlab(TeX(r'($t$)')) + ylab(TeX(r'($\hat{\lambda}
}(t)$)'))
261:                                     plot1.es
262:                                     options("scipen"=20,digits=4)
263:                                     plot_grid<-plot_grid(plotlist
= list(plot1.a,plot1.es),ncol=2,nrow = 1, labels = c('1'),'
2)'))
264:                                     #plot_grid<-plot_grid(plotlist
= list(plot1.a),ncol=1,nrow = 1, labels = c('1'))
265:
266:                                     lista_de_graficos[[unidade]] =
plot_grid
267:                                     }
268:                                     return(list('parâmetros estimados' =
estima_, 'parâmetros estimados2' = list(estima_1,estima_2,
estima_3), 'Gráfico duplo' = lista_de_graficos,'Gráfico
duplo AC' = list(plot1.a),'Gráfico duplo 1' = list(plot1.es)
, 'intervalo de confiança' = IC_95)) #'Modelo Estimado -
Intensidade acumulada' = plot1.a, 'Intensidade' = plot1.es,

```

```
269:         }else{
270:             return(list('parâmetros estimados' =
                estima_, 'intervalo de confiança' = IC_95))
271:         }
272: }
273:
274:
275:
276: # Copiar dados das colheitadeiras abaixo, sem copiar linhas
    vazias.
277:
278: Unidade Tempo Evento  Tn  GapTime FalhasCum
279: 1 10  1 0 10  0
280: 1 42  1 10 32  1
281: 1 51  1 42  9  2
282: 1 68  1 51 17  3
283: 1 85  1 68 17  4
284: 1 110 1 85 25  5
285: 1 120 1 110 10  6
286: 1 146 1 120 26  7
287: 1 157 1 146 11  8
288: 1 167 1 157 10  9
289: 1 194 1 167 27 10
290: 1 200 0 194 6 11
291: 2 23  1 0 23  0
292: 2 24  1 23  1  1
293: 2 39  1 24 15  2
294: 2 41  1 39  2  3
295: 2 72  1 41 31  4
296: 2 83  1 72 11  5
297: 2 95  1 83 12  6
298: 2 100 1 95  5  7
299: 2 116 1 100 16  8
300: 2 125 1 116 9  9
301: 2 136 1 125 11 10
302: 2 138 1 136 2 11
303: 2 165 1 138 27 12
304: 2 166 1 165 1 13
305: 2 200 0 166 34 14
306: 3 4  1 0 4  0
307: 3 14  1 4 10  1
308: 3 30  1 14 16  2
```

309: 3 48 1 30 18 3
310: 3 72 1 48 24 4
311: 3 90 1 72 18 5
312: 3 103 1 90 13 6
313: 3 117 1 103 14 7
314: 3 125 1 117 8 8
315: 3 136 1 125 11 9
316: 3 157 1 136 21 10
317: 3 171 1 157 14 11
318: 3 173 1 171 2 12
319: 3 175 1 173 2 13
320: 3 200 0 175 25 14
321: 4 13 1 0 13 0
322: 4 32 1 13 19 1
323: 4 44 1 32 12 2
324: 4 66 1 44 22 3
325: 4 74 1 66 8 4
326: 4 93 1 74 19 5
327: 4 99 1 93 6 6
328: 4 107 1 99 8 7
329: 4 121 1 107 14 8
330: 4 129 1 121 8 9
331: 4 131 1 129 2 10
332: 4 153 1 131 22 11
333: 4 157 1 153 4 12
334: 4 163 1 157 6 13
335: 4 185 1 163 22 14
336: 4 200 0 185 15 15
337: 5 9 1 0 9 0
338: 5 13 1 9 4 1
339: 5 14 1 13 1 2
340: 5 22 1 14 8 3
341: 5 24 1 22 2 4
342: 5 40 1 24 16 5
343: 5 49 1 40 9 6
344: 5 52 1 49 3 7
345: 5 61 1 52 9 8
346: 5 79 1 61 18 9
347: 5 97 1 79 18 10
348: 5 106 1 97 9 11
349: 5 116 1 106 10 12
350: 5 117 1 116 1 13

351: 5 132 1 117 15 14
352: 5 143 1 132 11 15
353: 5 148 1 143 5 16
354: 5 171 1 148 23 17
355: 5 196 1 171 25 18
356: 5 200 0 196 4 19
357: 6 8 1 0 8 0
358: 6 9 1 8 1 1
359: 6 25 1 9 16 2
360: 6 31 1 25 6 3
361: 6 40 1 31 9 4
362: 6 62 1 40 22 5
363: 6 73 1 62 11 6
364: 6 88 1 73 15 7
365: 6 107 1 88 19 8
366: 6 118 1 107 11 9
367: 6 124 1 118 6 10
368: 6 154 1 124 30 11
369: 6 158 1 154 4 12
370: 6 178 1 158 20 13
371: 6 200 0 178 22 14
372: 7 2 1 0 2 0
373: 7 8 1 2 6 1
374: 7 33 1 8 25 2
375: 7 44 1 33 11 3
376: 7 72 1 44 28 4
377: 7 89 1 72 17 5
378: 7 129 1 89 40 6
379: 7 143 1 129 14 7
380: 7 164 1 143 21 8
381: 7 189 1 164 25 9
382: 7 190 1 189 1 10
383: 7 200 0 190 10 11
384: 8 1 1 0 1 0
385: 8 29 1 1 28 1
386: 8 31 1 29 2 2
387: 8 38 1 31 7 3
388: 8 60 1 38 22 4
389: 8 82 1 60 22 5
390: 8 83 1 82 1 6
391: 8 101 1 83 18 7
392: 8 102 1 101 1 8

```
393: 8 128 1 102 26 9
394: 8 129 1 128 1 10
395: 8 153 1 129 24 11
396: 8 182 1 153 29 12
397: 8 200 0 182 18 13
398: 9 35 1 0 35 0
399: 9 42 1 35 7 1
400: 9 59 1 42 17 2
401: 9 70 1 59 11 3
402: 9 87 1 70 17 4
403: 9 91 1 87 4 5
404: 9 92 1 91 1 6
405: 9 116 1 92 24 7
406: 9 117 1 116 1 8
407: 9 118 1 117 1 9
408: 9 123 1 118 5 10
409: 9 151 1 123 28 11
410: 9 158 1 151 7 12
411: 9 166 1 158 8 13
412: 9 181 1 166 15 14
413: 9 182 1 181 1 15
414: 9 200 0 182 18 16
415:
416: # Rodar script só após copiar dados acima
417:
418: dados.colheitadera = read.table(file='clipboard', sep='\t',
  header = TRUE)
419: Tal = tapply(dados.colheitadera$Tempo, dados.colheitadera$
  Unidade, max)
420:
421: # Gráficos
422:
423: # Dotplot
424:
425: dados.colheitadera.dotplot = dados.colheitadera |> dplyr::
  filter(Evento==1)
426: dados.colheitadera.dotplot$Unidade = factor(dados.colheitadera.
  dotplot$Unidade)
427:
428: ggplot(dados.colheitadera.dotplot) +
429:   geom_point(aes(x = Tempo, y = Unidade),size=3.5) +
  theme_bw()
```

```
430:         geom_vline(xintercept = 200, linetype="dotted",
431:           color = "blue", size=1)+
432:           xlab('Time') + ylab('System')+scale_color_identity()
433: # Gráficos de linha
434:
435: # Sis1
436:
437: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==1 &
438:           Evento ==1)
439: total_falhas_i = nrow(df.colhe_i)
440: plot1<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
441:           size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw()
442:           +
443:           ggtitle(paste0('Total failures: ',total_falhas_i,',')) +
444:           geom_vline(xintercept = 200, linetype="dotted",
445:             color = "blue", size=1)+
446:             xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 1')+
447:             scale_color_identity()
448: plot1
449:
450: # Sis2
451:
452: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==2 &
453:           Evento ==1)
454: total_falhas_i = nrow(df.colhe_i)
455: plot2<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
456:           size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +
457:           ggtitle(paste0('Total failures: ',total_falhas_i,',')) +
458:           geom_vline(xintercept = 200, linetype="dotted",
459:             color = "blue", size=1)+
460:             xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 2')+
461:             scale_color_identity()
462: plot2
463:
464: # Sis3
465:
466: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==3 &
467:           Evento ==1)
468: total_falhas_i = nrow(df.colhe_i)
```



```
459: plot3<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(  
  size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +  
460:   ggtitle(paste0('Total failures: ',total_falhas_i, '.'  
  )) +  
461:   geom_vline(xintercept = 200, linetype="dotted",  
  color = "blue", size=1)+  
462:   xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 3')+  
  scale_color_identity()  
463: plot3  
464:  
465:  
466: # Sis4  
467:  
468: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==4 &  
  Evento ==1)  
469: total_falhas_i = nrow(df.colhe_i)  
470: plot4<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(  
  size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +  
471:   ggtitle(paste0('Total failures: ',total_falhas_i, '.'  
  )) +  
472:   geom_vline(xintercept = 200, linetype="dotted",  
  color = "blue", size=1)+  
473:   xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 4')+  
  scale_color_identity()  
474: plot4  
475:  
476:  
477: # Sis5  
478:  
479: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==5 &  
  Evento ==1)  
480: total_falhas_i = nrow(df.colhe_i)  
481: plot5<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(  
  size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +  
482:   ggtitle(paste0('Total failures: ',total_falhas_i, '.'  
  )) +  
483:   geom_vline(xintercept = 200, linetype="dotted",  
  color = "blue", size=1)+  
484:   xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 5')+  
  scale_color_identity()  
485: plot5  
486:
```

```
487:
488: # Sis6
489:
490: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==6 &
      Evento ==1)
491: total_falhas_i = nrow(df.colhe_i)
492: plot6<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
      size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +
493:       ggtitle(paste0('Total failures: ',total_falhas_i,',')) +
      )) +
494:       geom_vline(xintercept = 200, linetype="dotted",
      color = "blue", size=1)+
495:       xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 6')+
      scale_color_identity()
496: plot6
497:
498: # Sis7
499:
500: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==7 &
      Evento ==1)
501: total_falhas_i = nrow(df.colhe_i)
502: plot7<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
      size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +
503:       ggtitle(paste0('Total failures: ',total_falhas_i,',')) +
      )) +
504:       geom_vline(xintercept = 200, linetype="dotted",
      color = "blue", size=1)+
505:       xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 7')+
      scale_color_identity()
506: plot7
507:
508: # Sis8
509:
510: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==8 &
      Evento ==1)
511: total_falhas_i = nrow(df.colhe_i)
512: plot8<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
      size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +
513:       ggtitle(paste0('Total failures: ',total_falhas_i,',')) +
      )) +
514:       geom_vline(xintercept = 200, linetype="dotted",
      color = "blue", size=1)+
```

```

515:         xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 8')+
        scale_color_identity()
516: plot8
517:
518:
519: # Sis9
520:
521: df.colhe_i = dados.colheitadera |> dplyr::filter(Unidade ==1 &
        Evento ==1)
522: total_falhas_i = nrow(df.colhe_i)
523: plot9<-ggplot(df.colhe_i,aes(Tempo,FalhasCum+1)) + geom_point(
        size=1.5) + geom_line(aes(color = 1),size=0.4)+ theme_bw() +
524:         ggtitle(paste0('Total failures: ',total_falhas_i, '.'
        )) +
525:         geom_vline(xintercept = 200, linetype="dotted",
        color = "blue", size=1)+
526:         xlab(TeX(r'($t$)')) + ylab(' N(0,t] - System 9')+
        scale_color_identity()
527: plot9
528:
529:
530: plot_grid(plotlist = list(plot1,plot2,plot3,plot4,plot5,plot6,
        plot7,plot8,plot9),ncol=3,nrow = 3)#, labels = c('Sistema
        1','Sistema 2','Sistema 3','Sistema 4','Sistema 5','Sistema
        6','Sistema 7','Sistema 8','Sistema 9'))
531:
532:
533: # Estimar parâmetros
534:
535: lista_grafico=list()
536: lista_grafico_l=list()
537: lista_par=list()
538: lista_IC95=list()
539: estimativas = data.frame('beta' = rep(0,9),'muT' = rep(0,9), '
        theta' = rep(0,9), 'log' = rep(0,9))
540: log=0
541: for(i in c(1,2,3,5,6,7,8,9)){ # i=4
542:         df.colhe = dados.colheitadera |> dplyr::filter(
        Unidade ==i & Evento ==1)
543:         resultado_aplicao=aplicacao.aram(dado_aplica = df.
        colhe$Tempo, tal = as.vector(Tal)[i], unidade = 1, method =
        'mlp', format.model = 'col', grafico=1, um_sistema = TRUE,

```

```
      limites_par = 1)#,par_original = c(b,m,t))
544:      estimativas[i,] = c(resultado_aplicao$`parâmetros
      estimados`$par,resultado_aplicao$`parâmetros estimados`$
      value)
545:      lista_par[[i]]=resultado_aplicao$`parâmetros
      estimados`
546:      lista_grafico[i]=resultado_aplicao$`Gráfico duplo AC
      `
547:      lista_grafico_l[i]=resultado_aplicao$`Gráfico duplo
      l`
548:      lista_IC95[[i]]=resultado_aplicao$`intervalo de
      confiança`
549:      log=log+resultado_aplicao$`parâmetros estimados`$
      value
550:      print(lista_grafico[i])
551: }
552: log
553:
554:
555: # 0 limite para estimar os parâmetros do ARAM_1 para i=4, com
      limites_par = 0
556:
557: i=4
558:
559: df.colhe = dados.colheitadera |> dplyr::filter(Unidade ==i &
      Evento ==1)
560: resultado_aplicao=aplicacao.aram(dado_aplica = df.colhe$Tempo,
      tal = as.vector(Tal)[i], unidade = 1, method = 'mlp', format
      .model = 'col', grafico=1, um_sistema = TRUE, limites_par =
      0)#,par_original = c(b,m,t))
561: estimativas[i,] = c(resultado_aplicao$`parâmetros estimados`$
      par,resultado_aplicao$`parâmetros estimados`$value)
562: lista_par[[i]]=resultado_aplicao$`parâmetros estimados`
563: lista_grafico[i]=resultado_aplicao$`Gráfico duplo AC`
564: lista_grafico_l[i]=resultado_aplicao$`Gráfico duplo l`
565: lista_IC95[[i]]=resultado_aplicao$`intervalo de confiança`
566: log=-(log+resultado_aplicao$`parâmetros estimados`$value)
567:
568: log
569:
570: # Intervalos de confiança
571:
```

```
572:
573: padro_theta = vector()
574: for(i in 1:9){ # i=4
575:     padro_theta[i] = padroniza_theta(x = estimativas$
        theta[i], b = estimativas$beta[i],k=4)
576: }
577:
578: padro_thetaLI = vector()
579: for(i in 1:9){ # i=4
580:     padro_thetaLI[i] = padroniza_theta(x = lista_IC95[[i
        ]][3,2], b = estimativas$beta[i],k=4)
581: }
582:
583:
584: padro_thetaLS = vector()
585: for(i in 1:9){ # i=4
586:     padro_thetaLS[i] = padroniza_theta(x = lista_IC95[[i
        ]][3,3], b = estimativas$beta[i],k=4)
587: }
588:
589: IC95 = data.frame(estimativa = padro_theta, LI = padro_thetaLI,
        LS = padro_thetaLS)
590: options("scipen"=20, digits=2)
591: round(IC95,4)
592: lista_IC95
593: estimativas_gerais = cbind(estimativas,IC95)
594:
595:
596: # Gráficos
597:
598: # Acumulada ajustada
599: plot_grid(plotlist = lista_grafico,ncol=3,nrow = 3, labels = c(
        'S1','S2','S3','S4','S5','S6','S7','S8','S9'))
600: #Intensidade ajustada
601: plot_grid(plotlist = lista_grafico_1,ncol=3,nrow = 3, labels =
        c('S1','S2','S3','S4','S5','S6','S7','S8','S9'))
602:
603:
604: # Caso multissistema
605:
606: # Estimando em conjunto
```

```

607: resultado_aplicao.geral=aplicacao.aram(dado_aplica = dados.
      colheitadeira, tal = as.vector(Tal), unidade = 3, method = '
      mlp', format.model = 'row', grafico=0, um_sistema = FALSE,
      uma_trinca_geral = TRUE, limites_par = 1)#,par_original = c(b
      ,m,t))
608: resultado_aplicao.geral$`parâmetros estimados`
609: resultado_aplicao.geral
610:
611:
612: # estimador para theta e limites de confiança
613:
614: padroniza_theta(x = resultado_aplicao.geral$`parâmetros
      estimados`$par[3], b = resultado_aplicao.geral$`parâmetros
      estimados`$par[3],k=4)
615: padroniza_theta(x = resultado_aplicao.geral$`intervalo de
      confiança`[3,2], b = resultado_aplicao.geral$`parâmetros
      estimados`$par[3],k=4)
616: padroniza_theta(x = resultado_aplicao.geral$`intervalo de
      confiança`[3,3], b = resultado_aplicao.geral$`parâmetros
      estimados`$par[3],k=4)

```

A.2 PLPG

A.2.1 Pacotes e funções gerais

```

1: # Pacotes
2: library(dplyr)
3: library(plyr)
4: library(reshape2)
5: library(data.table)
6: library(optimr)
7: library(optimx)
8:
9: # Funções Gerais
10:
11: lgi<-function(x){if(x>0 & is.finite(x)){log(x)}else{0}}
12: lgi<-Vectorize(lgi)
13: lg<-function(x){apply(as.matrix(x),2,lgi)}
14:
15: logit_2<-function(x,b,k){
16:           x = mpfr(x, precBits = 40)

```

```

17:         if( (b<1 & x>=0) | (b>1 & x<=0) ){r=log((1+x)/(1-x))
        * 10^k}else{r=x}
18:         return(as.numeric(r))
19: }
20: logit_2<-Vectorize(logit_2)
21:
22: logit_3<-function(x,b,k){
23:     if(b>1 & x>0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
24:     if(b>1 & x<=0){r = x}
25:     if(b>0 & b<=1 & x>0){r = x}
26:     if(b>0 & b<=1 & x<=0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
27:     return(r)
28: }
29: logit_3<-Vectorize(logit_3)
30:
31: padroniza_theta<-function(x,b,k){
32:     x = mpfr(x, precBits = 20)
33:     if(b>1 & x>0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
34:     if(b>1 & x<=0){r = 2*(exp(10^(-k)*x)/(exp(10^(-k)*x)
+1)-0.5)}
35:     if(b>0 & b<=1 & x>0){r = 2*(exp(10^(-k)*x)/(exp
(10^(-k)*x)+1)-0.5)}
36:     if(b>0 & b<=1 & x<=0){r = 2*(exp(x)/(exp(x)+1)-0.5)}
37:     return(as.numeric(r))
38: }
39: padroniza_theta<-Vectorize(padroniza_theta)

```

A.2.2 Funções de intensidade e intensidade acumulada

```

1: # Funções de Intensidade e acumulada do PLPG
2:
3: # Função de intensidade
4:
5: func_intensidade_plpg_p<-function(para,t,tau,k=1,tal,pd=FALSE){
6:     if(pd==TRUE){
7:         # Parametros distintos
8:         p.par<-length(tau)+1}else{
9:         # Parametros iguais
10:         p.par<-1}
11:     p<-length(tau)

```

```

12:         if(tau[1]==0){tau<-c(0,tal);p<-0}else{tau<-c(0,tau,
    tal)}
13:         tal = tal[1]
14:         ind.t<-rep(0,p+2); posi<-c(1:(p+2))[t<=tau][1]-1;
    ind.t[posi]<-1
15:         mu_t.<-para[1]
16:         beta.<-para[2:(1+p.par)]
17:         theta.<-para[(2+p.par):(1+2*p.par)]
18:         h_theta = logit_2(x = theta., b = beta., k = 4)
19:         g_theta = theta.
20:         if(p==0){valor<-(mu_t./tal)*beta.*t^(beta.-1)}else{
    valor<-(mu_t./(tal+g_theta*tau))*beta.*(t-sign(beta.-1)*h_
    theta*tau)^(beta.-1)*ind.t}
21:         valor[is.na(valor)]<-0; valor<-sum(valor)
22:         return(matrix(valor,nrow=1))
23: }
24:
25: # Função de intensidade acumulada
26:
27: func_intensidade_plpg_ac_p<-function(para,t,tau,k=1,tal,pd=
    FALSE){
28:     if(pd==TRUE){
29:         # Parametros distintos
30:         p.par<-length(tau)+1}else{
31:         # Parametros iguais
32:         p.par<-1}
33:     p<-length(tau)
34:     if(tau[1]==0){tau<-c(0,tal);p<-0}else{tau<-c(0,tau,
    tal)}
35:     ind.t<-rep(0,p+2); posi<-c(1:(p+2))[t<=tau][1]-1
36:     mu_t.<-para[1]
37:     beta.<-para[2:(1+p.par)]
38:     theta.<-para[(2+p.par):(1+2*p.par)]
39:     if(posi==0|p==0){
40:         valor = (mu_t./tal)*t^(beta.)
41:     }else{
42:         ind.t[posi]<-1
43:         h_theta = logit_2(x = theta., b = beta.,
    k = 4)
44:         g_theta = theta.
45:         tau_i = tau[1:posi]

```



```

46:         tau_i_1 = dplyr::lag(tau_i, n=1, default
= 0)
47:         valor0<-(mu_t./((tal+g_theta*tau_i_1))*(
tau_i-sign(beta.-1)*h_theta*tau_i_1)^(beta.)-
48:         (mu_t./((tal+g_theta*tau_i_1))*
(tau_i_1-sign(beta.-1)*h_theta*tau_i_1)^(beta.)
49:         valor0[is.na(valor0)]<-0; valor0<-sum(
valor0)
50:         valor1<-(mu_t./((tal+g_theta*tau[posit]))*(
t-sign(beta.-1)*h_theta*tau[posit]))^(beta.) -
51:         (mu_t./((tal+g_theta*tau[posit])
)*(tau[posit]-sign(beta.-1)*h_theta*tau[posit]))^(beta.)
52:         valor1[is.na(valor1)]<-0; valor<-sum(
valor1) ; valor = valor0+valor1
53:     }
54:     return(matrix(valor,nrow=1))
55: }

```

A.2.3 Simulação de tempos de falha

```

1: # Simulação
2:
3: simula_plpg_p<-function(trunc_time, tau_, k., mu_t, beta_,
theta_){
4:     # Nota: funciona para sistemas com parametros iguais
5:     matrix_time<-matrix(data=0, nrow = 10^6, ncol=k.)
6:     n_vec_time_max<-0
7:     falhas_sis<-vector()
8:     for(j in 1:k.){
9:         parametros_sim<-c('Mu_T'= mu_t, 'Beta' =
beta_, 'Theta' = theta_)#'Theta 2' = theta2)
10:         tempo_novo<-0
11:         tempo_acumulado<-0
12:         vetor_tempo<-0
13:         tempo_esperado_primeira_falha<- 1#exp(
delta) + ( (trunc_time+exp(delta))/exp(alpha) )^(1/exp(beta)
)
14:         i<-1
15:         #w<-1
16:         #tau_valor<-vector()
17:         while(tempo_acumulado<trunc_time){

```

```

18:             options("scipen"=22,digits=22)
19:             tempo=tempo_acumulado
20:             u<-runif(1)
21:             #if(w<=length(tau_)){if(i==tau
_ [w]){tau_valor[w] = tempo_acumulado; w<-w+1}}
22:             Fun<-function(x){r <- func_
intensidade_plpg_ac_p(para=parametros_sim,t=x, tau = tau_, k
= k., tal = trunc_time, pd = FALSE)[1];return(r)} # j=1
23:             #Fun<-Vectorize(Fun)
24:             simula<-function(par){
25:                 z<-par[1]
26:                 resultado <- 10^8*
abs(Fun(x = tempo+z)-Fun(x = tempo)+log(1-u))
27:                 return(resultado)#;
print(z)
28:             }
29:             otimizacao<-try(optimr::optimr
(par = 2, fn = simula, lower = 10^-20, upper = trunc_time-
tempo, method="L-BFGS-B"), silent=TRUE)
30:             if(class(otimizacao)!="try-
error"){
31:                 tempo_novo<-
otimizacao$par
32:                 tempo_primeira_
falha<-tempo_novo
33:                 tempo_acumulado<-
tempo_acumulado+tempo_novo
34:                 vetor_tempo[i]<-
tempo_acumulado
35:                 rm(u)
36:                 i<-i+1 #; print(
list(i,tempo_acumulado)) #; print(vetor_tempo)#;print(list(i
,posi))
37:                 }#; print(list(i,tempo_
acumulado)) #; print(vetor_tempo)#;print(list(i,posi))
38:                 if(is.na(tempo_acumulado)){
tempo_acumulado <- tempo; i<-i-1}
39:             }
40:             n_vec_time<-length(vetor_tempo[vetor_
tempo<trunc_time])
41:             matrix_time[1:n_vec_time,j] <- vetor_
tempo[vetor_tempo<trunc_time]

```

```

42:             falhas_sis[j]<-n_vec_time
43:             n_vec_time_max<-max(n_vec_time_max,n_vec_
time)
44:         }
45:         matrix_time<-matrix_time[1:n_vec_time_max,]
46:         matrix_time[is.na(matrix_time)]<-0
47:         time_simulation<-list('Matriztempo' = matrix_time,'
Parametros' = parametros_sim,'Truncamentotempo' = trunc_time
,'Vetormax' = n_vec_time_max, 'Falhassis' = falhas_sis)
48:         return(time_simulation)
49: }

```

A.2.4 Função para aplicação

```

1: # Aplicação
2:
3: aplicacao.plpg_p<-function(dado_aplica = dado_aplica, tau_, tal
, unidade = 1, method = 'eqm', format.model = 'row',
grafico, um_sistema = FALSE, uma_trinca_geral = FALSE){
4:     options("scipen"=20,digits=20)
5:     if(format.model == 'col'){
6:         if(um_sistema){
7:             dado_aplica=matrix(c(dado_
aplica,1:length(dado_aplica)), ncol=2)
8:         }
9:         dado_aplica <- reshape2::melt(data = dado
_aplica)
10:         dado_aplica$Var1<-NULL
11:         colnames(dado_aplica)<-c('Unidade','Tempo
')
12:         dado_aplica<-dado_aplica |>
13:             dplyr::filter(Tempo!=0)
14:         dado_aplica$Evento<-1
15:         dado_aplica$Tn<-unlist(tapply(dado_aplica
$Tempo,dado_aplica$Unidade, function(x) dplyr::lag(x,n=1,
default = 0)))
16:     }
17:     if(method == 'mlp'){
18:         if(um_sistema){
19:             dado_aplica <-dado_aplica|>

```



```

44:             df_time<-t(df_time); colnames(df_time)<-as
               .character(1:length(posicao_max_tempo))
45:             pri_time<-as.data.frame(t(as.matrix(
tapply(dado_aplica$Tempo,dado_aplica$Unidade,function(x){x
[1]}))))))
46:             lag2_matrix_time<-data.table::as.data.
table(df_time)
47:             lag2_matrix_time[is.na(lag2_matrix_time)]
<-0
48:             tempo_trunca<-tal
49:             tempo_inicial_sistemas<-apply(matriz_
tempo, 2, min); tempo_inicial_sistemas<-min(tempo_inicial_
sistemas[tempo_inicial_sistemas!=0])
50:             tempo_final_sistemas2<-apply(lag2_matrix_
time, 2, max)
51:             matriz_tempo<-list(matriz_tempo, lag_
matrix_time)
52:
53:             log_L_PLP<-function(par, time_truca = as.
vector(tempo_trunca) , matrix_time = matriz_tempo, k. = k.,
len. = posicao_max_tempo){
54:                 if(uma_trinca_geral==TRUE){k
.=1}
55:                 NN<-1:max(posicao_max_tempo)
56:                 time1 = matrix_time[[1]]
57:                 fxyz1<-function(x) {func_
intensidade_plpg_p(para=par, t = x, tau = tau_,k = k.,tal =
time_truca)}
58:                 fxyz1<-Vectorize(fxyz1)
59:                 log.loop1 <- sum(lg(apply(
time1, 2, fxyz1)))#, na.rm = TRUE)
60:                 fxyz2<-function(x) {(func_
intensidade_plpg_ac_p(para=par, t=x, tau = tau_, k = k., tal
= time_truca))}
61:                 fxyz2<-Vectorize(fxyz2)
62:                 log.acumulada1<- k..*sum(fxyz2
(time_truca))
63:                 l<-log.loop1-log.acumulada1
64:                 return(-l)
65:             }
66:             if(uma_trinca_geral==TRUE){k=1}else{k=k
..}

```

```

67:         par0<-c(rep(5,k),rep(1.1,k),rep(0.5,k))
68:
69:         #Otimização
70:         estima_<-optimx::optimr(par=par0, fn=log_
_L_PLP, hessian=TRUE, lower = c(rep(0.0001,k), rep(0.01,k),
rep(-0.999999,k)), upper = c(rep(20000,k), rep(40,k), rep
(0.999999,k)), method="L-BFGS-B")
71:         #estima_<-optimx::optimr(par=par0, fn=log
_L_PLP, hessian=TRUE, lower = c(rep(0.0001,k), rep(0.01,k),
rep(-Inf,k)), upper = c(rep(20000,k), rep(40,k), rep(Inf,k))
, method="L-BFGS-B")
72:         parametros_estimados<-c(estima_$par)
73:         names(parametros_estimados)<-c(paste0('Mu
_T',1:k),paste0('Beta',1:k),paste0('Theta',1:k))
74:         M.rm = estima_$hessian
75:         vcov.rm = solve(M.rm)
76:         sd_<-diag(sqrt(vcov.rm))
77:         valor_p=2*(1-pnorm(estima_$par/sd_))
78:         IC_95<-cbind(estima_$par,estima_$par-1.96
*sd_,estima_$par+1.96*sd_)
79:
80:         colnames(IC_95)<-c("Estimativa","LI","LS"
)
81:         rownames(IC_95)<-c('Mu_T','Beta','Theta')
82:         IC_95
83:     }
84:
85:     if(grafico == 1){
86:         lista_de_graficos = list()
87:         for(unidade in 1:max(dado_aplica$Unidade)
){
88:             # Graficos #
89:             #print('Elaborando os graficos
')
90:             k.<-1
91:             dado_aplica_select <- dado_
aplica |>
92:                 dplyr::filter(
Unidade==unidade)
93:
94:             tempo<-dado_aplica_select$
Tempo[order(dado_aplica_select$Tempo)]

```

```

95:
96:         dados<-data.frame(y=1:length(
tempo), t=tempo, t0=dplyr::lag(tempo,n = 1, default = 0))
97:         if(uma_trinca_geral==TRUE){par
_estimado=estima_$par}else{
98:         par_estimado<-
estima_$par[c(unidade,max(dado_aplica$Unidade)+unidade,2*max
(dado_aplica$Unidade)+unidade)]
99:         }
100:
101:         # sistema<-unidade
102:         vetor_tempo<-tempo
103:         trun=tal
104:         if(max(dado_aplica$Unidade)
==1){unidade=1;sistema=1}
105:         Fun1.1<-function(x){
106:         r<-func_intensidade
_plpg_ac_p(para=par_estimado,t=x, tau = tau_,
107:
k = k., tal = tempo_trunca);
108:         return(r)
109:         }
110:
111:         Fun1.1<-Vectorize(Fun1.1)
112:
113:         Fun1.10<-function(x){
114:         r<-func_intensidade
_plpg_ac_p(para=par_estimado,t=x, tau = tau_,
115:
k = k., tal = tempo_trunca);
116:         return(r)
117:         }
118:
119:         Fun1.10<-Vectorize(Fun1.10)
120:
121:         Fun2.1<-function(x){
122:         r<-func_intensidade
_plpg_p(para=par_estimado,t=x, tau = tau_,
123:
k = k., tal = tempo_trunca);
124:         return(r)
125:         }

```

```

126:
127:         Fun2.1<-Vectorize(Fun2.1)
128:
129:
130:         x=c(seq(min(vetor_tempo[vetor_
tempo!=0]), max(vetor_tempo),0.1),vetor_tempo)
131:
132:
133:         df3<-data.frame(f2=1:length(
vetor_tempo),x=vetor_tempo,Color='DR')
134:
135:         if(length(tempo_trunca)!=1){
136:             df1.1<-data.frame(
f2=Fun1.10(vetor_tempo),x=vetor_tempo,Color='MA') #[1,]
137:             df1.1x<-data.frame(
f2=Fun1.10(x),x=x,Color='MA') # [1,]
138:         }else{
139:             df1.1<-data.frame(
f2=Fun1.10(vetor_tempo),x=vetor_tempo,Color='MA')
140:             df1.1x<-data.frame(
f2=Fun1.10(x),x=x,Color='MA')
141:         }
142:         df5.1<-data.frame(f2=Fun2.1(
vetor_tempo),x=vetor_tempo,Color='red')
143:         df5.1x<-data.frame(f2=Fun2.1(x
), x=x, Color='red')
144:
145:         cols2 <- c("DR"="red","MA"="
blue")
146:
147:         # Intensidade Acumulada
148:         plot1.a<-ggplot(df3,aes(x,f2,
color = Color)) +
149:             geom_line(data =
df3, aes(x,f2,color = Color),size=0.55) + geom_point(data =
df3, aes(x,f2,color = Color),size=1.5)+
150:             geom_point(data =
df1.1, aes(x,f2,color = Color),size=1.5) + geom_line(data =
df1.1x, aes(x,f2,color = Color), size=0.55,linetype = 2) +
151:             theme_bw() +
152:             xlab(TeX(r'($t$)'))
+ ylab(TeX(r'($\hat{\Lambda}(t)$)')) + scale_colour_manual

```



```

      (name = '', values=cols2)
153:         plot1.a
154:
155:         # Intensidade
156:         plot1.es <- ggplot(df5.1x, aes(
x,f2,color = Color)) + geom_point(data = df5.1, aes(x,f2,
color = Color),size=1.5) + geom_line(aes(color = 1), size
=0.4)+ theme_bw() +
157:         scale_color_
identity()+ xlab(TeX(r'($t$)')) + ylab(TeX(r'($\hat{\lambda
}(t)$)'))
158:         plot1.es
159:         options("scipen"=20,digits=4)
160:         plot_grid<-plot_grid(plotlist
= list(plot1.a,plot1.es),ncol=2,nrow = 1, labels = c('1'),'
2)'))
161:         lista_de_graficos[[unidade]] =
plot_grid
162:         }
163:         return(list('parametros estimados' =
estima_, 'Grafico duplo' = lista_de_graficos,'Grafico duplo
AC' = list(plot1.a),'Grafico duplo 1' = list(plot1.es), '
intervalo de confianca' = IC_95)) #'Modelo Estimado -
Intesidade acumulada' = plot1.a, 'Intensidade' = plot1.es,
164:         }else{
165:         return(list('parametros estimados' =
estima_, 'intervalo de confianca' = IC_95))
166:         }
167: }

```

A.2.5 Estudo de Simulação

```

1: # Estudo de Simulação
2:
3: estudo.simula.plpg.p= function( M=1000, k..., tau, m. , b., t
., tal.){
4:     res = matrix(0, nrow = M, ncol = 3)
5:     format.model. = 'col'
6:     time.i=Sys.time()
7:     i=1 ; if( k...==1){one.sys =TRUE}else{one.sys =
FALSE}

```

```

8:         while (dim(matrix(res[res[,3]!=0 ,],ncol=3))[1] != M
9:         ) {
10:             m = rep(m.,1) ; b = rep(b.,1); t = rep
11:             (t.,1)
12:             r_simula_plpg = simula_plpg_p(trunc_time=
13:             tal., tau_ = tau , k.=k..., mu_t = m, beta_ =b, theta_ = t)
14:             resul_sim<-try(aplicacao.plpg_p(dado_
15:             aplica = r_simula_plpg$`Matriztempo`, tau_ = tau, tal = as.
16:             vector(tal.), unidade = 1, method = 'mlp', format.model = '
17:             col', grafico=0, um_sistema = one.sys, uma_trinca_geral =
18:             TRUE), silent=T)
19:             if(class(resul_sim)!="try-error"){
20:                 res[i,]<-c(resul_sim$`
21:                 parametros estimados`$par)
22:                 time.f = Sys.time()
23:                 if(dim(matrix(res[res[,3]!=0
24:                 ,],ncol=3))[1]>=2){
25:                     res1 = as.data.
26:                     frame(na.exclude(res))
27:                     Res = list('Matriz
28:                     completa' = res[res[,2]>0,], 'Estimativa' = colMeans(res1[
29:                     res1[,2]>0,]), ' valor do i' = i , 'time loop' =time.f-time.
30:                     i, 'Media p/ estimativa' = (time.f-time.i)/i , 'Total de
31:                     falhas da ultima simulacao' = r_simula_plpg$`Falhassis`)
32:                     if(i %% 1 == 0) {
33:                         print(Res)} # colMeans(res1[ res1[,2]!=0 & res1[,1]<2 & res1
34:                         [,2]<100 ,]) res1[,3]!=0 & res1[,3]!=1 &
35:
36:                     }
37:                     if(is.na(res[i,1])){i<-i}else{
38:                         i<-i+1}
39:                     }else{i<-i}
40:                 }
41:             return(Res)
42:         }
43: }
44: # Cenário I
45:
46: for(nsis in c(5, 10, 20, 30)){
47:     resultado = estudo.simula.plpg.p(M=1000, k... = nsis
48:     , tau = 15 , m. = 05, b. = 1.4, t. = 0.8, tal.= 30)

```

```

32:         write.csv2(resultado$`Matriz completa`,paste0('
           resultado_simulacao_',nsis, '.csv'))
33: }
34:
35: # Cenário II
36:
37: for(nsis in c(5, 10, 20, 30)){
38:     resultado = estudo.simula.plpg.p(M=1000, k... = nsis
           , tau = 15 , m. = 50, b. = 0.8, t. =0.8, tal.= 30)
39:     write.csv2(resultado$`Matriz completa`,paste0('
           resultado2_simulacao_',nsis, '.csv'))
40: }

```

A.2.6 Aplicação em dados reais

```

1: # Aplicação em dados reais
2:
3: # Aplicação I
4:
5: df_exhauster = c(27, 92, 128, 133, 140, 151, 153, 158, 169,
           174, 183, 191, 212, 234, 244, 300, 401)
6:
7:
8: df_graf = data.frame(x=df_exhauster,y=1:length(df_exhauster),
           group = rep(1,length(df_exhauster)))
9:
10: # Gráfico
11:
12: ggplot(data = df_graf, aes(x=x,y=y)) + geom_line(size=0.7) +
           geom_point(size=0.9)+
13:     labs(x = TeX(r'($ t $)') , y = "Number of failures"
           , linetype = TeX(r'($\theta$)') ) + theme_bw()
14:
15: r_simula_plpg = df_exhauster
16: tal. = max(r_simula_plpg)+0.01
17: one.sys=TRUE
18:
19: # Seleção do primeiro ponto de mudança
20:
21: time_logvero = data.frame(time = r_simula_plpg, logvero = rep
           (0, length(r_simula_plpg)))

```

```
22: i=1
23: for(tau_i in r_simula_plpg){
24:     resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_
      plpg, tau_ = tau_i, tal = as.vector(tal.), unidade = 1,
      method = 'mlp', format.model = 'col', grafico=0, um_sistema
      = one.sys, uma_trinca_geral = TRUE)#,par_original = c(b,m,t)
      )
25:     time_logvero[i,2] = resul_sim$`parametros estimados`
      $value
26:     i=i+1
27:     print(time_logvero)
28: }
29: # Ordenando por log
30: time_logvero$logvero = -time_logvero$logvero
31: time_logvero = time_logvero[order(time_logvero$logvero,
      decreasing = TRUE),]
32:
33: # Seleção do segundo ponto de mudança
34: tau_1 = time_logvero$time[1]
35: r_simula_plpg2 = r_simula_plpg[r_simula_plpg!=tau_1]
36: time_logvero2 = data.frame(time = r_simula_plpg2, logvero =
      rep(0, length(r_simula_plpg2)))
37: i=1
38: for(tau_i in r_simula_plpg2){
39:     tau_2 = c(tau_i,tau_1); tau_2 = tau_2[order(tau_2,
      decreasing = FALSE)]
40:     resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_
      plpg, tau_ = tau_2, tal = as.vector(tal.), unidade = 1,
      method = 'mlp', format.model = 'col', grafico=0, um_sistema
      = one.sys, uma_trinca_geral = TRUE)#,par_original = c(b,m,t)
      )
41:     time_logvero2[i,2] = resul_sim$`parametros estimados`
      $value
42:     i=i+1
43:     print(time_logvero2)
44: }
45:
46: # Ordenando por log
47: time_logvero2$logvero = -time_logvero2$logvero
48: time_logvero2 = time_logvero2[order(time_logvero2$logvero,
      decreasing = TRUE),]
49:
```

```

50: # Comparando os AICs
51: AIC1 = -2*time_logvero$logvero[1]+2*4
52: AIC2 = -2*time_logvero2$logvero[1]+2*5
53: AIC1;AIC2
54:
55: # Seleção do terceiro ponto de mudança
56: tau_2 = time_logvero2$time[1]
57: r_simula_plpg3 = r_simula_plpg[!(r_simula_plpg %in% c(tau_1,tau_2))]
58: time_logvero3 = data.frame(time = r_simula_plpg3, logvero = rep(0, length(r_simula_plpg3)))
59: i=1
60: for(tau_i in r_simula_plpg3){
61:     tau_3 = c(tau_i,tau_2,tau_1); tau_3 = tau_3[order(tau_3,decreasing = FALSE)]
62:     resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_plpg, tau_ = tau_3, tal = as.vector(tal.), unidade = 1, method = 'mlp', format.model = 'col', grafico=0, um_sistema = one.sys, uma_trinca_geral = TRUE)#,par_original = c(b,m,t)
63:     time_logvero3[i,2] = resul_sim$`parametros estimados`$value
64:     i=i+1
65:     print(time_logvero3)
66: }
67:
68: time_logvero3$logvero = -time_logvero3$logvero
69: time_logvero3 = time_logvero3[order(time_logvero3$logvero, decreasing = TRUE),]
70: AIC3 = -2*time_logvero3$logvero[1]+2*6
71: AIC1;AIC2;AIC3
72:
73: # Como AIC2>AIC3, então seleciona-se p=2, com pontos de mudança iguais a c(tau_1,tau_2)
74:
75: t1 = c(tau_1,tau_2)
76:
77: resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_plpg, tau_ = t1, tal = as.vector(tal.), unidade = 1, method = 'mlp', format.model = 'col', grafico=1, um_sistema = TRUE, uma_trinca_geral = TRUE)
78: resul_sim$`Grafico duplo`

```

```
79: resul_sim$`parametros estimados`$par
80:
81: # Aplicação II
82:
83:
84: df_boeing =c(413, 427, 485, 522, 622, 687, 696, 865, 1312,
      1496, 1532, 1733, 1851, 1885,
85:           1916, 1934, 1952, 2019, 2076, 2138, 2145, 2167,
      2201)
86:
87:
88: df_graf = data.frame(x=df_boeing,y=1:length(df_boeing),group =
      rep(1,length(df_boeing)))
89:
90: # Gráfico
91:
92: ggplot(data = df_graf, aes(x=x,y=y)) + geom_line(size=0.7) +
      geom_point(size=0.9)+
93:       labs(x = TeX(r'($ t $)'), y = "Number of failures"
      , linetype = TeX(r'($\theta$)') ) + theme_bw()
94:
95: r_simula_plpg = df_boeing
96: tal. = max(r_simula_plpg)+0.01
97: one.sys=TRUE
98:
99: # Seleção do primeiro ponto de mudança
100:
101: time_logvero = data.frame(time = r_simula_plpg, logvero = rep
      (0, length(r_simula_plpg)))
102: i=1
103: for(tau_i in r_simula_plpg){
104:       resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_
      plpg, tau_ = tau_i, tal = as.vector(tal.), unidade = 1,
      method = 'mlp', format.model = 'col', grafico=0, um_sistema
      = one.sys, uma_trinca_geral = TRUE)#,par_original = c(b,m,t)
      )
105:       time_logvero[i,2] = resul_sim$`parametros estimados`
      $value
106:       i=i+1
107:       print(time_logvero)
108: }
109: # Ordenando por log
```

```
110: time_logvero$logvero = -time_logvero$logvero
111: time_logvero = time_logvero[order(time_logvero$logvero,
    decreasing = TRUE),]
112:
113: # Seleção do segundo ponto de mudança
114: tau_1 = time_logvero$time[1]
115: r_simula_plpg2 = r_simula_plpg[r_simula_plpg!=tau_1]
116: time_logvero2 = data.frame(time = r_simula_plpg2, logvero =
    rep(0, length(r_simula_plpg2)))
117: i=1
118: for(tau_i in r_simula_plpg2){
119:     tau_2 = c(tau_i,tau_1); tau_2 = tau_2[order(tau_2,
    decreasing = FALSE)]
120:     resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_
    plpg, tau_ = tau_2, tal = as.vector(tal.), unidade = 1,
    method = 'mlp', format.model = 'col', grafico=0, um_sistema
    = one.sys, uma_trinca_geral = TRUE)#,par_original = c(b,m,t)
    )
121:     time_logvero2[i,2] = resul_sim$`parametros estimados
    `$value
122:     i=i+1
123:     print(time_logvero2)
124: }
125:
126: # Ordenando por log
127: time_logvero2$logvero = -time_logvero2$logvero
128: time_logvero2 = time_logvero2[order(time_logvero2$logvero,
    decreasing = TRUE),]
129:
130: # Comparando os AICs
131: AIC1 = -2*time_logvero$logvero[1]+2*4
132: AIC2 = -2*time_logvero2$logvero[1]+2*5
133: AIC1;AIC2
134:
135: # Como AIC2>AIC1, então seleciona-se p=1, com ponto de mudança
    igual a tau_1
136:
137: t1 = tau_1
138:
139: resul_sim<-aplicacao.plpg_p(dado_aplica = r_simula_plpg, tau_ =
    t1, tal = as.vector(tal.), unidade = 1, method = 'mlp',
    format.model = 'col', grafico=1, um_sistema = TRUE, uma_
```

```
    trinca_geral = TRUE)  
140: resul_sim$`Grafico duplo`  
141: resul_sim$`parametros estimados`$par
```
