Universidade Federal de Ouro Preto
Escola de Minas
Engenharia de Controle e Automação

Gabriel Ferreira

Aplicação do Método SeqTest em Amostradores MCMC

Ouro Preto
2026

Gabriel Ferreira

Aplicação do Método SeqTest em Amostradores MCMC

Trabalho apresentado ao Colegiado do Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para a obtenção do Grau de Engenheiro(a) de Controle e Automação.

Orientador: Prof. Dr. Ivair Ramos Silva
Coorientador: Profa. Dra. Adrielle de Carvalho Santana

Ouro Preto
2026

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
ESCOLA DE MINAS
DEPARTAMENTO DE ENGENHARIA CONTROLE E AUTOMACAO

FOLHA DE APROVAÇÃO

Gabriel Ferreira

Aplicação do Método SeqTest em Amostradores MCMC

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Engenheiro de Controle e Automação.

Aprovada em 28 de julho de 2026

Membros da banca

Dr. Ivair Ramos Silva - Orientador - DECOM - Universidade Federal de Ouro Preto
Dra. Adrielle de Carvalho Santana - Coorientadora - DECAT - Universidade Federal de Ouro Preto
Me. João Carlos Vilela de Castro - Convidado - DECAT - Universidade Federal de Ouro Preto
Dr. Agnaldo José da Rocha Reis - Convidado - DECAT - Universidade Federal de Ouro Preto

Ivair Ramos Silva, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 29/07/2026

Documento assinado eletronicamente por **Ivair Ramos Silva**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 28/07/2026, às 18:47, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **1145993** e o código CRC **83162E03**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591533 - www.ufop.br

Agradecimentos

Primeiramente, gostaria de expressar minha mais sincera gratidão aos meus pais, **William e Marilane**, por todo o amor, dedicação e incentivo que me permitiram chegar até aqui. Agradeço também à **Katty** e meu irmão **Heitor**, que sempre estiveram ao meu lado, oferecendo conselhos, apoio e motivação ao longo desta caminhada.

Manifesto igualmente meu reconhecimento a todo o corpo docente da **UFOP**, em especial ao **DECAT**, pela excelência no ensino e por proporcionarem um ambiente acadêmico que contribuiu não apenas para meu crescimento intelectual, mas também pessoal.

Aos meus amigos e irmãos da **República Pulgatório**, fica meu agradecimento pela parceria, pelas conversas inspiradoras, pela convivência que tornou essa trajetória muito mais enriquecedora, e que estiveram presentes em todos os momentos, oferecendo amizade e tornando cada etapa dessa jornada mais leve e significativa.

A todos, meu mais sincero e profundo obrigado.

Duvidar de tudo ou crer em tudo são duas soluções igualmente cômodas, porque ambas nos dispensam de refletir.

— Henri Poincaré, *A Ciência e a Hipótese* (1902).

Resumo

Métodos de Monte Carlo via Cadeias de Markov (MCMC), como o amostrador de Gibbs e o algoritmo Metropolis-Hastings, são amplamente utilizados na inferência Bayesiana para gerar amostras de distribuições *a posteriori* analiticamente intratáveis. Um problema central na utilização desses métodos é determinar o momento a partir do qual a cadeia efetivamente convergiu para a distribuição-alvo, uma vez que amostras anteriores à convergência produzem estimativas enviesadas. Os diagnósticos de convergência existentes na literatura, como os testes de Geweke, Heidelberger-Welch e o procedimento de Raftery-Lewis, apresentam limitações significativas: formulam a hipótese nula como estacionaridade da cadeia, não controlam formalmente o erro acumulado em aplicações repetidas e dependem de aproximações assintóticas. O presente trabalho aplica o método SeqTest, um teste sequencial não-paramétrico exato, a diferentes cenários de aplicação envolvendo amostradores MCMC. O SeqTest inverte a formulação das hipóteses, adotando como hipótese nula que a cadeia não convergiu, e utiliza funções de gasto alpha para controlar o erro tipo I de forma exata por meio da distribuição Binomial. Foram implementadas em linguagem R três aplicações: estimação de prevalência de SPAM com classificador imperfeito via Gibbs Sampler, estimação de frequência alélica via Metropolis-Hastings e estimação de taxa de fraude em transações financeiras via Gibbs Sampler. Os resultados evidenciaram que o SeqTest discrimina cadeias bem e mal misturadas, acompanhando a qualidade de mistura esperada teoricamente, enquanto os diagnósticos clássicos de Geweke e Heidelberger-Welch certificaram convergência de forma indiscriminada, inclusive em cadeias comprovadamente não-convergentes.

Palavras-chave: Monte Carlo via Cadeias de Markov. Teste sequencial. Convergência. Amostrador de Gibbs. Metropolis-Hastings. Função de gasto alpha.

Abstract

Markov Chain Monte Carlo (MCMC) methods, such as the Gibbs sampler and the Metropolis-Hastings algorithm, are widely used in Bayesian inference to generate samples from analytically intractable posterior distributions. A central problem in the use of these methods is determining when the chain has effectively converged to the target distribution, since pre-convergence samples yield biased estimates. Existing convergence diagnostics, such as the Geweke test, the Heidelberger-Welch test and the Raftery-Lewis procedure, have significant limitations: they formulate the null hypothesis as stationarity, do not formally control the accumulated error in repeated applications and rely on asymptotic approximations. This work applies the SeqTest method, an exact non-parametric sequential test, to different application scenarios involving MCMC samplers. The SeqTest reverses the hypothesis formulation by adopting as null hypothesis that the chain has not converged, and employs alpha spending functions to control the type I error exactly through the Binomial distribution. Three applications were implemented in R: SPAM prevalence estimation with an imperfect classifier via Gibbs Sampler, allele frequency estimation via Metropolis-Hastings and fraud rate estimation in financial transactions via Gibbs Sampler. The results showed that the SeqTest discriminates between well- and poorly-mixed chains, tracking the mixing quality expected on theoretical grounds, whereas the classical Geweke and Heidelberger-Welch diagnostics certified convergence indiscriminately, including in demonstrably non-convergent chains.

Keywords: Markov Chain Monte Carlo. Sequential test. Convergence. Gibbs sampler. Metropolis-Hastings. Alpha spending function.

Lista de figuras

Figura 1 – Cenário favorável de SPAM ($\eta = 0{,}99$, $\theta = 0{,}95$), primeira réplica. Acima: *traceplot* da prevalência ψ. Abaixo: monitoramento da estatística S_j frente ao limiar de sinalização. 50
Figura 2 – Cenário difícil de SPAM ($\eta = 0{,}75$, $\theta = 0{,}80$), primeira réplica. Acima: *traceplot* de ψ, com aparência estável. Abaixo: monitoramento de S_j, que acompanha o limiar sem folga. 51
Figura 3 – Cenário de detector moderno ($\eta = 0{,}95$, $\theta = 0{,}995$), primeira réplica. Acima: *traceplot* de ψ. Abaixo: monitoramento de S_j, com sinalização confortavelmente dentro da janela. 52
Figura 4 – Cenário de detector legado ($\eta = 0{,}80$, $\theta = 0{,}95$), primeira réplica. Acima: *traceplot* de ψ. Abaixo: monitoramento de S_j, em regime limítrofe. . . . 53
Figura 5 – HWE com $\sigma = 0{,}05$ (calibração adequada), primeira réplica. Acima: *traceplot* de p, com mistura densa em torno de 0,60. Abaixo: monitoramento de S_j. 55
Figura 6 – HWE com $\sigma = 0{,}005$ (passos curtos), primeira réplica. Acima: *traceplot* de p, com excursões lentas de baixa frequência. Abaixo: monitoramento de S_j, muito abaixo do limiar. 56
Figura 7 – HWE com $\sigma = 0{,}5$ (passos grandes), primeira réplica. Acima: *traceplot* de p, com patamares de estagnação e saltos abruptos. Abaixo: monitoramento de S_j, distante do limiar. 57
Figura 8 – Percentual de cadeias declaradas convergentes por cada método, em cada cenário, sobre 50 réplicas independentes. 58

Lista de tabelas

Tabela 1 – Comparação entre os diagnósticos de convergência. 27
Tabela 2 – Parâmetros de calibração do SeqTest e valores adotados. 33
Tabela 3 – Configuração do teste sequencial na função `AnalyzeSetUp.Binomial`. . 41
Tabela 4 – Parâmetros dos quatro cenários gerados por amostrador de Gibbs. . . . 43
Tabela 5 – Contagens genotípicas utilizadas na aplicação de frequência alélica. . . 44
Tabela 6 – Cenários da aplicação de frequência alélica. 45
Tabela 7 – Controles de validação da implementação, sobre 50 réplicas independentes. 48
Tabela 8 – Resultados dos diagnósticos para a aplicação de SPAM, sobre 50 réplicas por cenário. 49
Tabela 9 – Resultados dos diagnósticos para a aplicação de Fraude, sobre 50 réplicas por cenário. 51
Tabela 10 – Resultados dos diagnósticos para a aplicação HWE, sobre 50 réplicas por cenário. 54
Tabela 11 – Panorama comparativo: proporção de cadeias declaradas convergentes por cada método, sobre 50 réplicas independentes por cenário. 58
Tabela 12 – Relação entre o denominador de identificabilidade e a decisão de cada método, nos quatro cenários gerados por amostrador de Gibbs. 59
Tabela 13 – Estimativas pontuais medianas entre as 50 réplicas, confrontadas com os valores teóricos. 60

Lista de abreviaturas e siglas

MCMC	Monte Carlo via Cadeias de Markov
MH	Metropolis-Hastings
HWE	Equilíbrio de Hardy-Weinberg
i.i.d.	Independente e identicamente distribuído
SPAM	Mensagens eletrônicas não solicitadas
SeqTest	Teste sequencial de convergência

Lista de símbolos

α	Nível de significância
β	Probabilidade de erro tipo II
η	Sensibilidade do classificador
θ	Especificidade do classificador
ψ	Prevalência real
τ	Taxa observada de positivos
ϕ	Parâmetro de calibração do SeqTest
ρ	Parâmetro da função de gasto alpha
π	Distribuição estacionária de uma Cadeia de Markov
N	Comprimento máximo de vigilância da cadeia
M	Número total de testes sequenciais
k	Número de estatísticas por teste no SeqTest
w	Tamanho da janela de comparação menos um
c	Parâmetro de margem do SeqTest
S_j	Estatística de contagem acumulada no teste j
$Q_{j,l}$	Estatística de posição relativa
$B_{j,l}$	Variável binária derivada de $Q_{j,l}$
cv_j	Limiar de sinalização para o teste j
$A(j)$	Função de gasto alpha avaliada no teste j

Sumário

1 INTRODUÇÃO . . . 14
1.1 Justificativas e Relevância . . . 15
1.2 Objetivos . . . 16
1.3 Metodologia . . . 16
1.4 Organização e estrutura . . . 17

2 REVISÃO DE LITERATURA . . . 19
2.1 Fundamentos de Inferência Bayesiana . . . 19
2.2 Geração de Números Pseudo-Aleatórios . . . 20
2.3 Métodos de Monte Carlo via Cadeias de Markov . . . 21
2.3.1 Cadeias de Markov: conceitos básicos . . . 21
2.3.2 O Amostrador de Gibbs . . . 22
2.3.3 O Algoritmo Metropolis-Hastings . . . 23
2.4 O Problema da Convergência em MCMC . . . 24
2.4.1 Conceitos fundamentais de teste de hipóteses . . . 24
2.4.2 Teste de Geweke . . . 25
2.4.3 Teste de Heidelberger-Welch . . . 26
2.4.4 Procedimento de Raftery-Lewis . . . 26
2.4.5 Limitações comuns . . . 27
2.5 O Método SeqTest . . . 27
2.5.1 Formulação das hipóteses . . . 27
2.5.2 Construção das estatísticas . . . 28
2.5.2.1 Etapa 1: divisão da cadeia em blocos . . . 28
2.5.2.2 Etapa 2: estatísticas de posição relativa . . . 28
2.5.2.3 Etapa 3: binarização . . . 29
2.5.2.4 Etapa 4: estatística de contagem . . . 30
2.5.3 Funções de gasto α . . . 31
2.5.4 Limiar de sinalização . . . 32
2.5.5 Convergência em escala . . . 32
2.5.6 Parâmetros de calibração . . . 33
2.5.7 Resumo do procedimento . . . 33
2.6 Aplicações dos Amostradores MCMC . . . 34
2.6.1 Modelo de Prevalência com Classificador Imperfeito . . . 34
2.6.1.1 O Gibbs Sampler para o modelo de prevalência . . . 35
2.6.2 Aplicação 1: Classificação de E-mails SPAM . . . 36

2.6.3 Aplicação 2: Detecção de Fraude em Cartão de Crédito . . . 36
2.6.4 Aplicação 3: Estimação de Frequência Alélica . . . 36

3 DESENVOLVIMENTO . . . 38
3.1 Ferramentas Utilizadas . . . 38
3.2 Implementação do Gibbs Sampler . . . 38
3.3 Implementação do Metropolis-Hastings . . . 39
3.3.1 Taxa de aceitação como indicador de calibração . . . 40
3.4 Implementação do SeqTest . . . 40
3.4.1 Função de binarização . . . 40
3.4.2 Configuração do teste sequencial . . . 41
3.4.3 Monitoramento sequencial e cache dos limiares . . . 42
3.4.4 Aplicação dos diagnósticos clássicos . . . 43
3.5 Cenários das Aplicações de Gibbs . . . 43
3.6 Cenários da Aplicação de Frequência Alélica . . . 44
3.6.1 Dados genotípicos . . . 44
3.6.2 Cenários de calibração . . . 45
3.7 Procedimento de Comparação com Diagnósticos Clássicos . . . 45
3.7.1 Protocolo experimental . . . 45
3.7.1.1 Sementes aleatórias . . . 46
3.7.1.2 Controles de validação . . . 46
3.7.1.3 Custo computacional . . . 46
3.7.2 Critérios de comparação . . . 47
3.7.3 Apresentação dos resultados . . . 47

4 RESULTADOS . . . 48
4.1 Validação da implementação . . . 48
4.2 Aplicação: Classificação de SPAM . . . 49
4.2.1 Cenário favorável . . . 49
4.2.2 Cenário difícil . . . 50
4.3 Aplicação: Detecção de Fraude . . . 51
4.3.1 Cenário moderno . . . 52
4.3.2 Cenário legado . . . 52
4.4 Aplicação: Equilíbrio de Hardy-Weinberg . . . 53
4.4.1 Cenário com $\sigma = 0{,}05$ (calibração adequada) . . . 54
4.4.2 Cenário com $\sigma = 0{,}005$ (passos curtos) . . . 55
4.4.3 Cenário com $\sigma = 0{,}5$ (passos grandes) . . . 56
4.5 Análise Comparativa . . . 57
4.5.1 Discriminação entre cadeias boas e ruins . . . 58
4.5.2 O comportamento do teste de Geweke . . . 59

4.5.3 A insuficiência das estimativas pontuais 60
4.5.4 Síntese 61

5 CONSIDERAÇÕES FINAIS 62
5.1 Principais resultados 62
5.2 Implicações práticas 63
5.3 Limitações 63
5.4 Trabalhos futuros 64
5.5 Perspectivas para a Engenharia de Controle e Automação 64

Referências 66

APÊNDICE A – CÓDIGO-FONTE EM R 69
A.1 Configuração 69
A.2 Amostradores MCMC 69
A.3 Implementação do SeqTest 71
A.4 Diagnósticos Clássicos 72
A.5 Execução do Experimento 73

ANEXO A – DECLARAÇÃO DE USO DE IAGEN NO PROCESSO DE ESCRITA CIENTÍFICA 76

1 Introdução

A inferência Bayesiana consolidou-se, nas últimas décadas, como uma das abordagens mais versáteis para a análise estatística de dados em diversas áreas do conhecimento, como economia, biologia, genética, astrofísica e engenharia (GELMAN *et al.*, 2013). Diferentemente da abordagem frequentista, a perspectiva Bayesiana trata os parâmetros de um modelo estatístico como variáveis aleatórias, permitindo incorporar conhecimento prévio por meio de distribuições *a priori* e atualizar esse conhecimento à luz dos dados observados por meio do Teorema de Bayes (ROBERT; CASELLA, 2004).

Na prática, a aplicação do Teorema de Bayes exige o cálculo da distribuição *a posteriori* do parâmetro de interesse, o que frequentemente envolve integrais analiticamente intratáveis, especialmente em modelos com múltiplos parâmetros ou estruturas hierárquicas. Para contornar essa dificuldade, os métodos de Monte Carlo via Cadeias de Markov (MCMC) tornaram-se ferramentas computacionais indispensáveis na estatística Bayesiana (KASS *et al.*, 1998). Os dois amostradores MCMC mais utilizados são o amostrador de Gibbs, introduzido por Geman e Geman (1984) e popularizado por Casella e George (1992), e o algoritmo Metropolis-Hastings, proposto originalmente por Metropolis *et al.* (1953) e generalizado por Hastings (1970).

O princípio fundamental dos métodos MCMC consiste em construir uma Cadeia de Markov cuja distribuição estacionária é exatamente a distribuição *a posteriori* desejada. Ao executar a cadeia por um número suficiente de iterações, os valores gerados passam a se comportar como amostras da distribuição-alvo, permitindo estimar quantidades de interesse como médias, variâncias e intervalos de credibilidade *a posteriori* (ROBERT; CASELLA, 2004). Contudo, a cadeia requer um período inicial, denominado *burn-in*, em que os valores produzidos ainda são influenciados pelo ponto de partida e pelas correlações inerentes ao processo markoviano, não sendo representativos da distribuição-alvo.

Um dos principais problemas na utilização de métodos MCMC, portanto, é determinar a partir de qual momento a cadeia efetivamente convergiu para a distribuição estacionária. A identificação correta desse ponto é essencial, porque se forem utilizados valores anteriores à convergência na inferência, as estimativas resultantes estarão enviesadas e os intervalos de credibilidade serão incorretos. Apesar de sua importância, não existe uma regra geral que determine, *a priori*, o número de iterações necessárias para que a convergência seja alcançada (COWLES; CARLIN, 1996).

Para enfrentar esse problema, diversos métodos de diagnóstico de convergência foram propostos ao longo das últimas décadas. Entre os mais amplamente utilizados estão o teste de Geweke (1992), baseado na comparação das médias de dois trechos da cadeia;

o teste de Heidelberger e Welch (1983), que emprega a estatística de Cramer-von Mises para avaliar a estacionaridade; e o procedimento de Raftery e Lewis (1992), que estima o número de iterações necessárias com base em probabilidades de transição. Uma revisão comparativa abrangente desses e de outros métodos foi realizada por Cowles e Carlin (1996), e mais recentemente por Roy (2020).

Ainda que consolidados na literatura e amplamente utilizados, esses métodos apresentam limitações significativas. Em primeiro lugar, todos formulam a hipótese nula como "a cadeia é estacionária", de modo que a não rejeição dessa hipótese é interpretada como evidência de convergência, quando, na realidade, pode representar apenas ausência de poder estatístico para detectar a não-estacionaridade. Em segundo lugar, na prática esses testes são aplicados repetidamente até que a hipótese nula não seja rejeitada, o que configura um problema de testes múltiplos sem controle formal do erro acumulado, aumentando substancialmente a probabilidade de concluir erroneamente pela convergência. Por fim, os métodos de Geweke e Heidelberger-Welch são fundamentados em resultados assintóticos, cujo desempenho real pode divergir do nominal dependendo dos parâmetros de calibração e do tamanho da cadeia (SILVA *et al.*, 2026).

Diante dessas limitações, Silva *et al.* (2026) propuseram o SeqTest, um teste sequencial não-paramétrico exato para sinalizar a convergência de amostradores em geral. A inovação fundamental do método reside na inversão da formulação das hipóteses: a hipótese nula passa a ser que a cadeia ainda não convergiu, e a convergência só é sinalizada quando há evidência estatística positiva em seu favor. O controle do erro tipo I é realizado de forma exata, por meio de funções de gasto alpha, e a distribuição da estatística de teste sob a hipótese nula é binomial, assim eliminando a dependência de aproximações assintóticas. Além disso, o método permite calcular formalmente o número mínimo de iterações necessárias para atingir um poder estatístico desejado.

1.1 Justificativas e Relevância

A motivação para o presente trabalho parte de duas constatações. A primeira é de natureza teórica: como discutido na seção anterior, os diagnósticos de convergência existentes na literatura apresentam limitações estruturais que comprometem sua confiabilidade, particularmente em cenários em que a convergência é lenta ou inexistente. O SeqTest, proposto por Silva *et al.* (2026), oferece uma solução para essas limitações, com garantias exatas de controle de erro. A segunda constatação é de natureza prática: apesar de sua contribuição teórica, o método foi demonstrado no artigo original com um conjunto restrito de aplicações, havendo espaço para investigar seu comportamento em cenários adicionais e em diferentes problemas.

A relevância deste trabalho reside na aplicação do SeqTest a problemas práticos

que utilizam amostradores MCMC, tanto o Gibbs Sampler quanto o Metropolis-Hastings, e em cenários com diferentes níveis de dificuldade de convergência. Ao comparar sistematicamente o SeqTest com os diagnósticos clássicos de Geweke, Heidelberger-Welch e Raftery-Lewis em cada cenário, busca-se evidenciar as situações nas quais o método proposto oferece ganhos concretos em termos de diagnóstico correto, facilitando sua adoção por pesquisadores e praticantes que utilizam métodos MCMC em suas análises.

1.2 Objetivos

Geral

Aplicar o método SeqTest de teste sequencial de convergência, proposto por Silva *et al.* (2026), a amostradores de Monte Carlo via Cadeias de Markov em diferentes cenários de aplicação, avaliando seu desempenho em comparação com os diagnósticos clássicos de convergência.

Específicos

- Estudar e descrever os fundamentos teóricos dos métodos de Monte Carlo via Cadeias de Markov, incluindo o amostrador de Gibbs e o algoritmo Metropolis-Hastings, bem como os diagnósticos de convergência existentes na literatura;
- Compreender e descrever em profundidade o método SeqTest, incluindo a construção das estatísticas de teste, as funções de gasto alpha, os limiares de sinalização e o cálculo do número de iterações em função do poder estatístico desejado;
- Implementar o SeqTest em linguagem R, utilizando o pacote Sequential (SILVA; MARO; KULLDORFF, 2021), para aplicações envolvendo o modelo Bayesiano de estimação de prevalência com classificadores imperfeitos e o modelo Beta-Binomial com o algoritmo Metropolis-Hastings;
- Comparar os resultados obtidos pelo SeqTest com os diagnósticos de Geweke, Heidelberger-Welch e Raftery-Lewis, utilizando o pacote coda (PLUMMER et al., 2006), em cenários com diferentes níveis de dificuldade de convergência;
- Discutir as vantagens, limitações e condições de aplicabilidade do SeqTest nos cenários investigados.

1.3 Metodologia

Este trabalho tem natureza computacional e aplicada, sendo baseado em simulação estatística. O desenvolvimento seguiu três etapas principais.

A primeira etapa consistiu em uma revisão da literatura sobre os fundamentos de inferência Bayesiana, métodos MCMC e diagnósticos de convergência, tendo como referência principal o artigo de Silva *et al.* (2026), complementado pelos trabalhos fundadores da área Metropolis *et al.* (1953), Hastings (1970), Geman e Geman (1984) e Casella e George (1992), e pelas revisões de diagnósticos de convergência de Cowles e Carlin (1996) e Roy (2020).

A segunda etapa envolveu a implementação dos amostradores MCMC para cada aplicação estudada, em linguagem R (R CORE TEAM, 2024). Para as aplicações que utilizam o modelo de estimação de prevalência com classificadores imperfeitos, em que a estrutura matemática segue o modelo proposto por Larangeira (2012) e empregado por Silva *et al.* (2026), foi implementado o amostrador de Gibbs com variáveis latentes. Para a aplicação que utiliza o modelo Beta-Binomial, foi implementado o algoritmo Metropolis-Hastings com proposta do tipo caminhada aleatória (*random walk*).

Para cada aplicação, foram definidos cenários com parâmetros realistas baseados em literatura específica da área, variando as condições de modo a produzir cadeias com convergência rápida (cenário favorável) e convergência lenta ou inexistente (cenário desafiador). O SeqTest foi aplicado a cada cadeia gerada utilizando a função Binario descrita no apêndice do artigo de referência e as funções AnalyzeSetUp.Binomial e Analyze.Binomial do pacote R Sequential (SILVA *et al.*, 2021). Os parâmetros de calibração do SeqTest foram definidos conforme recomendações do artigo original.

Os diagnósticos clássicos de Geweke, Heidelberger-Welch e Raftery-Lewis foram aplicados às mesmas cadeias por meio do pacote coda (PLUMMER *et al.*, 2006), permitindo a comparação direta dos resultados. A análise comparativa focou na capacidade de cada método de diagnosticar corretamente a convergência nos cenários favoráveis e, sobretudo, de identificar corretamente a não-convergência nos cenários desafiadores.

1.4 Organização e estrutura

O restante deste trabalho está organizado da seguinte forma.

O Capítulo 2 apresenta a revisão de literatura, abordando os fundamentos da geração de números pseudo-aleatórios, os métodos MCMC (amostrador de Gibbs e Metropolis-Hastings), os diagnósticos de convergência existentes e suas limitações, o método SeqTest em detalhes, e a descrição dos problemas de aplicação selecionados.

O Capítulo 3 descreve a implementação computacional de cada aplicação, incluindo os modelos Bayesianos empregados, os amostradores implementados, os parâmetros e cenários definidos, e o procedimento de aplicação do SeqTest e dos diagnósticos clássicos.

O Capítulo 4 apresenta os resultados obtidos para cada aplicação e cenário, incluindo

traceplots, decisões dos diagnósticos de convergência, e a análise comparativa entre o SeqTest e os métodos clássicos.

O Capítulo 5 apresenta as considerações finais, discutindo as conclusões do trabalho, suas limitações e sugestões para trabalhos futuros.

2 Revisão de literatura

Este capítulo apresenta os fundamentos teóricos necessários ao desenvolvimento do trabalho. São abordados os princípios da inferência Bayesiana e a origem de sua dificuldade computacional, os amostradores de Monte Carlo via Cadeias de Markov, os diagnósticos de convergência disponíveis na literatura, o método SeqTest e os problemas de aplicação investigados.

2.1 Fundamentos de Inferência Bayesiana

A inferência Bayesiana trata os parâmetros de um modelo como variáveis aleatórias, o que permite quantificar a incerteza sobre seus valores por meio de distribuições de probabilidade. Seu mecanismo central é o Teorema de Bayes, que estabelece como atualizar o conhecimento sobre um parâmetro θ à luz dos dados observados D, conforme a Equação (2.1):

$$p(\theta \mid D) = \frac{p(D \mid \theta) \cdot p(\theta)}{p(D)} \tag{2.1}$$

Nessa expressão, $p(\theta)$ é a distribuição *a priori*, que representa o conhecimento sobre θ antes de observar os dados; $p(D \mid \theta)$ é a *likelihood*, que expressa a probabilidade dos dados observados para cada valor possível de θ; $p(D) = \int p(D \mid \theta)p(\theta)\, d\theta$ é a evidência marginal, constante normalizadora que garante que a *posteriori* integre um; e $p(\theta \mid D)$ é a distribuição *a posteriori*, que representa o conhecimento atualizado após a observação dos dados.

Como a evidência marginal não depende de θ, na prática trabalha-se com a forma proporcional da Equação (2.2):

$$p(\theta \mid D) \propto p(D \mid \theta) \cdot p(\theta). \tag{2.2}$$

Essa forma é suficiente para os métodos MCMC, que requerem apenas a avaliação do produto entre *likelihood* e *prior*, sem necessidade de calcular $p(D)$.

Uma distribuição *a priori* é dita conjugada com respeito a uma dada *likelihood* quando a *posteriori* resultante pertence à mesma família da *prior*. Nesse caso, a atualização Bayesiana reduz-se à atualização dos parâmetros da distribuição. O par conjugado relevante para este trabalho é o Beta-Binomial: se a *prior* de uma proporção p é $\text{Beta}(\alpha_0, \beta_0)$ e os dados consistem em k sucessos em n tentativas independentes, a *posteriori* é dada pela Equação (2.3):

$$p \mid \text{dados} \sim \text{Beta}(\alpha_0 + k, \;\; \beta_0 + n - k). \tag{2.3}$$

Essa conjugação é utilizada diretamente nas aplicações de estimação de prevalência (Seções 2.6.2 e 2.6.3) e na aplicação de frequência alélica (Seção 2.6.4).

Embora o Teorema de Bayes forneça o arcabouço teórico completo, sua aplicação prática esbarra na intratabilidade da evidência marginal. Em modelos com múltiplos parâmetros, a integral $\int p(D \mid \theta)p(\theta)\, d\theta$ frequentemente não possui solução analítica, o que impossibilita o cálculo direto da *posteriori*. Essa dificuldade motivou o desenvolvimento dos métodos de Monte Carlo via Cadeias de Markov, apresentados na Seção 2.3.

2.2 Geração de Números Pseudo-Aleatórios

Os métodos de Monte Carlo requerem a geração de números aleatórios. Na prática, utilizam-se números pseudo-aleatórios, uma vez que a reprodutibilidade dos estudos computacionais precisa ser garantida (GENTLE, 2009). O ponto de partida é a obtenção de valores da distribuição Uniforme no intervalo $(0, 1)$, usualmente por algoritmos de congruência linear como o Fibonacci defasado e o Twister de Mersenne (GENTLE, 2009). A partir de valores uniformes, duas abordagens permitem gerar amostras de distribuições arbitrárias.

A primeira é o método da distribuição inversa. Seja X uma variável aleatória contínua com função de distribuição acumulada $F_X(x)$. Como a transformação $U = F_X(X)$ produz uma variável com distribuição Uniforme$(0, 1)$, se u é uma realização de U (um valor particular assumido por essa variável), então a Equação (2.4) fornece uma realização de X:

$$X = F_X^{-1}(u). \tag{2.4}$$

O método aplica-se a qualquer distribuição contínua cuja inversa da função de distribuição seja computável, o que constitui também sua principal limitação, pois F_X^{-1} não é conhecida analiticamente em distribuições mais complexas (ROBERT; CASELLA, 2004).

A segunda abordagem, o método de aceitação-rejeição, contorna essa restrição. Seja Y uma variável aleatória de distribuição conhecida e de fácil amostragem, com densidade f_Y cujo suporte contém o suporte de X, e seja f_X a densidade da variável-alvo. Se existe uma constante a que satisfaz a condição da Equação (2.5),

$$a \cdot f_Y(x) \geq f_X(x), \quad \forall\, x \in \mathbb{R}, \tag{2.5}$$

então o seguinte procedimento gera realizações de X: gera-se uma realização y da variável Y, cuja densidade é f_Y, e um valor u a partir de Uniforme$(0, 1)$; se $u \leq f_X(y)/(a \cdot f_Y(y))$, aceita-se y como realização de X; caso contrário, repete-se o passo anterior. A densidade f_Y é denominada majorante ou proposta, e a eficiência do método depende de sua proximidade com f_X: quanto mais próximas, menor a taxa de rejeição (ROBERT; CASELLA, 2004).

A limitação decisiva do método surge quando f_X não é plenamente computável, o que impede a determinação da constante a. Esse é precisamente o cenário da inferência Bayesiana, no qual o denominador da *posteriori* é uma integral intratável. É para essa classe de problemas que os métodos de Monte Carlo via Cadeias de Markov oferecem solução.

2.3 Métodos de Monte Carlo via Cadeias de Markov

2.3.1 Cadeias de Markov: conceitos básicos

Uma Cadeia de Markov é uma sequência de variáveis aleatórias $X_0, X_1, X_2, \ldots$ que satisfaz a propriedade de Markov: a distribuição do estado futuro depende exclusivamente do estado presente, sendo independente de toda a trajetória passada, conforme a Equação (2.6):

$$P(X_{n+1} = x \mid X_n, X_{n-1}, \ldots, X_0) = P(X_{n+1} = x \mid X_n). \tag{2.6}$$

As transições entre estados são governadas pelo *kernel* $K(x, x')$, que especifica a probabilidade (ou densidade, no caso contínuo) de transitar de x para x'. Em espaços de estados finitos, o *kernel* corresponde à matriz de transição P, cujo elemento P_{ij} representa a probabilidade de ir do estado i ao estado j (ROBERT; CASELLA, 2004).

Um amostrador é o procedimento que gera a sequência de valores da cadeia, isto é, o algoritmo responsável por produzir as amostras da distribuição de interesse. Para que uma Cadeia de Markov sirva como amostrador, isto é, como mecanismo gerador das amostras da distribuição de interesse, três propriedades são necessárias. A irredutibilidade garante que todos os estados se comunicam, de modo que qualquer estado pode ser alcançado a partir de qualquer outro em um número finito de passos. A aperiodicidade assegura que a cadeia não fique presa em ciclos de comprimento fixo. Uma cadeia simultaneamente irredutível e aperiódica é denominada ergódica.

O resultado central para os métodos MCMC é o Teorema Ergódico: se uma cadeia é ergódica, existe uma única distribuição estacionária π tal que, independentemente do estado inicial, a distribuição de X_n converge para π quando n tende ao infinito. Adicionalmente, para qualquer função integrável g, a média amostral ao longo da cadeia converge para a esperança de g sob π (GAMERMAN; LOPES, 2006), conforme a Equação (2.7):

$$\frac{1}{n}\sum_{i=1}^{n} g(X_i) \longrightarrow E_\pi[g(X)], \quad \text{quando } n \to \infty. \tag{2.7}$$

Uma condição suficiente para que π seja a distribuição estacionária é o balanço detalhado, apresentado na Equação (2.8):

$$\pi(x) \cdot K(x, x') = \pi(x') \cdot K(x', x), \quad \forall\, x, x'. \tag{2.8}$$

Essa condição tem papel central na construção do algoritmo Metropolis-Hastings (Seção 2.3.3).

A estratégia dos métodos MCMC consiste em construir uma cadeia ergódica cuja distribuição estacionária coincida com a distribuição de interesse. No contexto Bayesiano, essa distribuição é a *posteriori* $p(\theta \mid D)$. Executada a cadeia por tempo suficiente, os valores gerados podem ser utilizados como amostras aproximadas da *posteriori*, permitindo estimar médias, variâncias e intervalos de credibilidade (KASS *et al.*, 1998).

Cabe destacar uma consequência do Teorema Ergódico que será central na análise dos resultados deste trabalho. A convergência da média amostral é garantida assintoticamente para qualquer cadeia ergódica, independentemente da intensidade da autocorrelação entre iterações. O que a autocorrelação compromete não é a localização das estimativas, mas o tamanho efetivo da amostra e, por consequência, a quantificação da incerteza.

2.3.2 O Amostrador de Gibbs

O amostrador de Gibbs, proposto por Geman e Geman (1984) e popularizado por Casella e George (1992), aplica-se quando a distribuição conjunta de interesse possui condicionais completas de forma conhecida.

Considere um vetor de parâmetros $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$ do qual se deseja amostrar a *posteriori* conjunta $p(\theta_1, \ldots, \theta_k \mid D)$. O método requer que as condicionais completas $p(\theta_i \mid \boldsymbol{\theta}_{-i}, D)$ sejam conhecidas e que se possa gerar amostras de cada uma delas. Após inicializar $\theta_1^{(0)}, \ldots, \theta_k^{(0)}$, cada iteração t amostra sequencialmente conforme as Equações (2.9) a (2.11):

$$\theta_1^{(t)} \sim p(\theta_1 \mid \theta_2^{(t-1)}, \theta_3^{(t-1)}, \ldots, \theta_k^{(t-1)}, D) \tag{2.9}$$

$$\theta_2^{(t)} \sim p(\theta_2 \mid \theta_1^{(t)}, \theta_3^{(t-1)}, \ldots, \theta_k^{(t-1)}, D) \tag{2.10}$$

$$\vdots$$

$$\theta_k^{(t)} \sim p(\theta_k \mid \theta_1^{(t)}, \theta_2^{(t)}, \ldots, \theta_{k-1}^{(t)}, D) \tag{2.11}$$

Ao amostrar θ_i, utilizam-se sempre os valores mais recentes dos parâmetros já atualizados na iteração corrente, estratégia que melhora a eficiência da exploração do espaço paramétrico (CASELLA; GEORGE, 1992). Sob condições de regularidade, a sequência gerada constitui uma Cadeia de Markov ergódica cuja distribuição estacionária é a *posteriori* conjunta.

O amostrador de Gibbs é particularmente eficiente quando se utilizam *priors* conjugadas, pois nesse caso as condicionais completas pertencem a famílias conhecidas (GELMAN *et al.*, 2013). Essa é a situação do modelo de prevalência empregado nas duas primeiras aplicações deste trabalho (Seção 2.6.1).

2.3.3 O Algoritmo Metropolis-Hastings

O algoritmo Metropolis-Hastings, proposto por Metropolis *et al.* (1953) e generalizado por Hastings (1970), é o método MCMC mais geral. Diferentemente do Gibbs, dispensa o conhecimento das condicionais completas e exige apenas a avaliação da densidade *a posteriori* em qualquer ponto, a menos de uma constante de proporcionalidade.

O algoritmo utiliza uma distribuição proposta $q(\theta^* \mid \theta)$ para gerar candidatos a novos estados, aceitos ou rejeitados segundo uma razão que garante a convergência para a distribuição-alvo. Após inicializar $\theta^{(0)}$, cada iteração t executa os seguintes passos:

1. Gerar um candidato θ^* a partir de $q(\theta^* \mid \theta^{(t-1)})$.
2. Calcular a razão de aceitação A conforme a Equação (2.12).
3. Gerar $u \sim \text{Uniforme}(0, 1)$.
4. Se $u < \min(1, A)$, aceitar o candidato, fazendo $\theta^{(t)} = \theta^*$; caso contrário, manter $\theta^{(t)} = \theta^{(t-1)}$.

A razão de aceitação é dada pela Equação (2.12):

$$A = \frac{p(D \mid \theta^*) \cdot p(\theta^*) \cdot q(\theta^{(t-1)} \mid \theta^*)}{p(D \mid \theta^{(t-1)}) \cdot p(\theta^{(t-1)}) \cdot q(\theta^* \mid \theta^{(t-1)})} \tag{2.12}$$

A razão A é construída de modo a satisfazer o balanço detalhado (Equação (2.8)) com respeito à distribuição-alvo. Sua propriedade decisiva é que a constante normalizadora $p(D)$ cancela, por aparecer no numerador e no denominador, o que elimina a necessidade de calcular integrais intratáveis (ROBERT; CASELLA, 2004).

A variante empregada neste trabalho é o Metropolis-Hastings com caminhada aleatória, no qual a proposta é simétrica, $q(\theta^* \mid \theta) = \text{Normal}(\theta, \sigma^2_{\text{proposta}})$. Nesse caso, os termos da proposta também cancelam, e a razão simplifica-se para a Equação (2.13):

$$A = \frac{p(D \mid \theta^*) \cdot p(\theta^*)}{p(D \mid \theta^{(t-1)}) \cdot p(\theta^{(t-1)})}. \tag{2.13}$$

A escolha de σ_{proposta} determina a eficiência do amostrador, e sua influência é central nos experimentos deste trabalho. A adequação de σ_{proposta} é sempre relativa à dispersão da distribuição-alvo: um passo é considerado pequeno quando o desvio-padrão da proposta é muito inferior ao desvio-padrão da *posteriori*, e grande quando se aproxima ou excede a amplitude da região de alta densidade. Valores muito pequenos, nesse sentido, produzem alta taxa de aceitação, entendida como a proporção de propostas aceitas ao longo da cadeia, isto é, o número de aceitações dividido por $n-1$, porém com passos curtos e forte autocorrelação, fazendo a cadeia deslocar-se lentamente pelo espaço paramétrico. Valores muito grandes, por sua vez, projetam candidatos para fora dessa região de alta densidade, o que eleva a frequência de rejeições e deixa a cadeia estagnada no mesmo estado por longos períodos. A taxa de aceitação ótima é de aproximadamente 44% em problemas unidimensionais e 23% em dimensões maiores (ROBERT; CASELLA, 2004). Como a rejeição mantém a cadeia no estado corrente, o algoritmo é automaticamente aperiódico, o que, somado à liberdade de escolha da proposta, torna o método aplicável a uma ampla classe de problemas.

2.4 O Problema da Convergência em MCMC

Os métodos MCMC geram amostras da distribuição-alvo apenas depois que a cadeia atinge o regime estacionário. As iterações anteriores a esse ponto constituem o período de *burn-in* e devem ser descartadas. O problema central é que o número de iterações necessárias à convergência não é conhecido *a priori* e depende da complexidade do modelo, da escolha da distribuição proposta e do ponto de partida da cadeia (COWLES; CARLIN, 1996).

Além de ferramentas gráficas como *traceplots* e gráficos de autocorrelação, foram propostos testes formais que buscam decidir, a partir de propriedades estatísticas da cadeia, se ela já atingiu o regime estacionário. Note-se que esses testes respondem à pergunta sobre a convergência, e não determinam diretamente o número de iterações necessárias, com exceção do procedimento de Raftery-Lewis. Os três mais utilizados na literatura são descritos nesta seção. Para uma revisão abrangente, consulte Cowles e Carlin (1996) e Roy (2020).

2.4.1 Conceitos fundamentais de teste de hipóteses

Um teste de hipóteses é um procedimento para decidir entre duas afirmações mutuamente excludentes com base nos dados observados. A hipótese nula (H_0) é assumida

como verdadeira até que haja evidência suficiente para rejeitá-la e a hipótese alternativa (H_1) é aquela para a qual se busca evidência. No diagnóstico de convergência, essa escolha está longe de ser uma convenção sem consequências: adotar como (H_0) a afirmação de que a cadeia convergiu, ou a afirmação oposta, determina qual das duas conclusões passará a exigir evidência estatística para ser sustentada.

Dois tipos de erro podem ocorrer. O erro tipo I, ou falso positivo, consiste em rejeitar H_0 quando ela é verdadeira, e sua probabilidade máxima tolerada é o nível de significância α, tipicamente fixado em 0,05. O erro tipo II, ou falso negativo, consiste em não rejeitar H_0 quando H_1 é verdadeira, e sua probabilidade é denotada por β. O poder estatístico do teste é $1 - \beta$, isto é, a probabilidade de rejeitar corretamente H_0 quando H_1 é verdadeira.

A interpretação desses erros no contexto do diagnóstico de convergência é o que distingue as duas famílias de métodos tratadas neste trabalho. Sinalizar convergência em uma cadeia que não convergiu leva ao uso de amostras não representativas da *posteriori* e produz estimativas de incerteza enviesadas. Deixar de detectar a convergência de uma cadeia que já convergiu resulta apenas em desperdício computacional. O primeiro erro é, portanto, substancialmente mais grave que o segundo.

Ao adotar como hipótese nula a afirmação de que a cadeia não convergiu, o SeqTest coloca o erro mais grave sob controle formal do nível de significância α. Nos testes clássicos, cuja hipótese nula é a estacionaridade da cadeia, o erro controlado é o menos grave, ao passo que o erro perigoso permanece sem garantia. Essa inversão de formulação, e não uma diferença de sensibilidade, é a origem das discrepâncias observadas no Capítulo 4.

Uma última distinção é necessária. Em testes com tamanho amostral fixo, a decisão é tomada uma única vez, ao final da coleta. Em testes sequenciais, como o SeqTest, os dados são analisados à medida que são gerados e a decisão pode ocorrer em qualquer um dos M testes intermediários. Essa estratégia permite detectar a convergência mais cedo, mas exige mecanismos de controle do erro acumulado, tratados na Seção 2.5.3.

2.4.2 Teste de Geweke

O teste proposto por Geweke (1992) avalia a estacionaridade comparando as médias de duas porções da cadeia, tipicamente os primeiros 10% (parte A) e os últimos 50% (parte B). Sejam $\bar{X}_A$ e $\bar{X}_B$ as médias amostrais dessas partes, com tamanhos n_A e n_B. A estatística de teste é o Z-score da Equação (2.14):

$$Z = \frac{\bar{X}_A - \bar{X}_B}{\sqrt{\frac{\hat{\sigma}_A^2}{n_A} + \frac{\hat{\sigma}_B^2}{n_B}}} \tag{2.14}$$

onde $\hat{\sigma}_A^2$ e $\hat{\sigma}_B^2$ são estimadores das variâncias assintóticas. Sob a hipótese nula de estacionaridade, Z converge assintoticamente para uma Normal padrão.

A limitação do procedimento manifesta-se em seu uso corrente: quando H_0 é rejeitada, novos trechos da cadeia são gerados e o teste é reaplicado, repetindo-se até que H_0 não seja rejeitada. Essa prática configura um problema de testes múltiplos, no qual a probabilidade acumulada de concluir erroneamente pela convergência aumenta a cada repetição, sem controle formal (SILVA *et al.*, 2026).

2.4.3 Teste de Heidelberger-Welch

O teste de Heidelberger e Welch (1983) emprega a estatística de Cramer-von Mises para avaliar a estacionaridade de primeira ordem da segunda metade da cadeia, conforme a Equação (2.15):

$$T_n = \int_0^1 (R_n(t))^2 \, dt \tag{2.15}$$

onde $R_n(t)$ é um processo normalizado pela densidade espectral $D(0)$ avaliada na frequência zero, definido pela Equação (2.16):

$$R_n(t) = \frac{\gamma_{\lfloor nt \rfloor} - \lfloor nt \rfloor \, \bar{X}}{\sqrt{n \, D(0)}}, \quad 0 \leq t \leq 1 \tag{2.16}$$

com $\gamma_n = \sum_{i=1}^{n} X_i$ e $\bar{X} = \gamma_n / n$. Sob estacionaridade, T_n converge para a distribuição de uma ponte Browniana. À semelhança do teste de Geweke, o procedimento é repetido até que a hipótese de estacionaridade não seja rejeitada, incorrendo na mesma inflação do erro por testes múltiplos (COWLES; CARLIN, 1996).

2.4.4 Procedimento de Raftery-Lewis

O método de Raftery e Lewis (1992) não constitui um teste formal de estacionaridade. Seu objetivo é estimar a iteração r a partir da qual a cadeia se torna estacionária, com base nas probabilidades de transição de uma versão binarizada da cadeia.

Para um quantil q de interesse, define-se a função indicadora $H_i = I(X_i \leq x)$ e estimam-se as probabilidades de transição estacionárias pelo estimador da Equação (2.17):

$$\hat{P}_{l,u} = \frac{\#\{i : H_{si} = u, \; H_{s(i-1)} = l\}}{\#\{i : H_{s(i-1)} = l\}}, \quad l, \, u = 0, \, 1 \tag{2.17}$$

com $H_{si} = H_{1+(i-1)s}$. O valor de r é determinado como a iteração a partir da qual a diferença entre as probabilidades estimadas e um parâmetro de precisão ε definido pelo usuário torna-se suficientemente pequena (RAFTERY; LEWIS, 1992).

Duas limitações comprometem a confiabilidade do procedimento. O uso dos estimadores $\hat{P}_{l,u}$ no lugar dos valores verdadeiros implica que a cobertura nominal não é garantida. Além disso, o problema de testes múltiplos não é tratado formalmente quando o procedimento é aplicado a diferentes quantis (SILVA *et al.*, 2026).

2.4.5 Limitações comuns

Os três métodos compartilham uma limitação estrutural: formulam a hipótese nula como a estacionaridade da cadeia, de modo que a não rejeição dessa hipótese é interpretada como indicativo de convergência. A não rejeição, contudo, pode decorrer simplesmente de poder estatístico insuficiente, e não de convergência efetiva. Os testes de Geweke e Heidelberger-Welch dependem ainda de distribuições assintóticas, cujo desempenho pode divergir do nominal em cadeias finitas. Nenhum dos três oferece meio de calcular previamente o número de iterações necessárias para garantir um poder estatístico desejado (SILVA *et al.*, 2026). A Tabela 1 sintetiza essas características em comparação com o SeqTest.

Tabela 1 – Comparação entre os diagnósticos de convergência.

Característica	**Geweke**	**H-W**	**R-L**	**SeqTest**
Hipótese nula	Estacionária	Estacionária	Estimativa	Não convergiu
Tipo de teste	Assintótico	Assintótico	Não formal	Exato
Controle de α acumulado	Não	Não	N/A	Sim
Cálculo formal de N	Não	Não	Aproximado	Sim

Fonte: Elaborado pelo autor.

2.5 O Método SeqTest

Para superar as limitações descritas na seção anterior, Silva *et al.* (2026) propuseram o SeqTest, um teste sequencial não-paramétrico exato para sinalizar a convergência de amostradores.

2.5.1 Formulação das hipóteses

Seja N o comprimento máximo de vigilância da cadeia e $X_1, \ldots, X_N$ a sequência de valores gerados pelo amostrador. A sequência é considerada convergente se, para algum inteiro r com $1 \leq r \leq N$, os valores $X_r, X_{r+1}, \ldots, X_N$ são independentes e identicamente distribuídos com distribuição comum $F_X(x)$. As hipóteses do SeqTest são então formuladas pelas Equações (2.18) e (2.19):

$$H_0 \;:\; X_1, X_2, \ldots, X_N \text{ não convergiu para uma amostra aleatória.} \tag{2.18}$$

$$H_1 \;:\; X_r, X_{r+1}, \ldots, X_N \text{ é uma amostra aleatória para algum } r. \tag{2.19}$$

A hipótese nula afirma a não-convergência, e a convergência só é sinalizada mediante evidência estatística positiva, isto é, pela rejeição de H_0. Quando H_1 é verdadeira, o valor de r é desconhecido, mas uma estimativa é obtida como subproduto do procedimento. O método é não-paramétrico porque não assume forma alguma para $F_X(x)$: o teste baseia-se em comparações de *rank* entre valores da cadeia, o que o torna aplicável a qualquer distribuição-alvo contínua.

2.5.2 Construção das estatísticas

A ideia central do SeqTest é transformar a cadeia de valores contínuos em uma sequência de variáveis binárias, cuja soma acumulada pode ser comparada com limiares exatos calculados a partir da distribuição Binomial. O processo ocorre em quatro etapas.

2.5.2.1 Etapa 1: divisão da cadeia em blocos

A cadeia $X_1, \ldots, X_N$ é dividida em M blocos não-sobrepostos de tamanho igual, cada um contendo $k(w+1)$ valores consecutivos, em que k e w são parâmetros definidos pelo usuário e w é ímpar. O número de blocos, que corresponde ao número de testes sequenciais, é dado pela Equação (2.20):

$$M = \left\lfloor \frac{N}{k(w+1)} \right\rfloor . \tag{2.20}$$

Para o j-ésimo teste, extrai-se a sub-sequência correspondente ao j-ésimo bloco, cujos índices de início e fim na cadeia original são dados pelas Equações (2.21) e (2.22):

$$l_1 \;=\; k(w+1)(j-1)+1 \tag{2.21}$$

$$l_2 \;=\; k(w+1)j \tag{2.22}$$

Com os parâmetros adotados neste trabalho ($k = 20$ e $w = 49$), cada bloco consome $20 \times 50 = 1.000$ valores da cadeia. Uma cadeia de 80.000 iterações produz, portanto, $M = 80$ testes sequenciais.

2.5.2.2 Etapa 2: estatísticas de posição relativa

Dentro de cada bloco, verifica-se se os valores se comportam como uma amostra aleatória. A ideia baseia-se em posições relativas: se os valores são independentes e

identicamente distribuídos, a posição de qualquer valor em relação aos seus vizinhos deve ser imprevisível; havendo dependência ou tendência, as posições seguirão padrões sistemáticos.

Cada bloco de $k(w+1)$ valores é organizado em k janelas de $w+1$ valores. Para a l-ésima janela, toma-se o valor de referência $Y_{j,l}$ e compara-se com os w valores restantes, obtendo o *rank* normalizado da Equação (2.23):

$$V_{j,l} = \frac{1 + \sum_{m} I(Y_{j,l} \geq Y_{j,m})}{w+1} \tag{2.23}$$

onde $I(\cdot)$ é a função indicadora. Em termos simples, $V_{j,l}$ mede a fração de valores da janela que são menores ou iguais ao valor de referência. Aplica-se em seguida uma simetrização, de modo que a estatística não distinga entre valores extremamente grandes e extremamente pequenos, conforme a Equação (2.24):

$$Q_{j,l} = \min\left(V_{j,l},\ \ 1 - V_{j,l}\right). \tag{2.24}$$

Para tornar o cálculo concreto, considere uma janela com $w = 9$, portanto com dez valores. Suponha o valor de referência $Y_{j,l} = 5{,}2$ e os nove valores de comparação $\{3{,}1;\ 7{,}4;\ 4{,}8;\ 6{,}0;\ 2{,}5;\ 8{,}1;\ 5{,}5;\ 4{,}0;\ 6{,}3\}$. Quatro deles são menores ou iguais a 5,2, o que resulta nas Equações (2.25) e (2.26):

$$V_{j,l} = \frac{1+4}{10} = 0{,}5 \tag{2.25}$$

$$Q_{j,l} = \min(0{,}5;\ \ 0{,}5) = 0{,}5 \tag{2.26}$$

O valor obtido indica que a referência ocupa posição central na janela, comportamento compatível com uma amostra aleatória.

O contraste entre as hipóteses manifesta-se na distribuição de $Q_{j,l}$. Sob H_1, os valores são independentes e identicamente distribuídos, todas as posições relativas são igualmente prováveis e $Q_{j,l}$ segue distribuição uniforme discreta sobre o conjunto $\{0,\ 1/(w+1),\ \ldots,\ 1/2\}$. Sob H_0, a dependência entre valores consecutivos faz com que vizinhos tendam a ser similares, o que concentra $V_{j,l}$ em valores extremos e, consequentemente, $Q_{j,l}$ em valores próximos de zero.

2.5.2.3 Etapa 3: binarização

As estatísticas $Q_{j,l}$ são convertidas em variáveis binárias por meio de um limiar fixo, conforme a Equação (2.27):

$$B_{j,l} = I\left(Q_{j,l} \leq \phi/2\right) \tag{2.27}$$

onde ϕ é um parâmetro de calibração, adotado como $\phi = 0{,}5$. Quando Q é suficientemente pequeno, isto é, menor ou igual a 0,25, tem-se $B = 1$; caso contrário, $B = 0$. No exemplo anterior, $Q_{j,l} = 0{,}5$ excede o limiar e produz $B_{j,l} = 0$, sinal de comportamento compatível com amostra aleatória.

A utilidade da binarização reside nas distribuições resultantes. Sob H_1, demonstra-se que cada $B_{j,l}$ segue uma Bernoulli com o parâmetro da Equação (2.28):

$$\Pr(B_{j,l} = 1) = \phi + \frac{1}{w+1}. \tag{2.28}$$

Com os parâmetros deste trabalho ($\phi = 0{,}5$ e $w = 49$), essa probabilidade vale 0,52. Sob H_0, a concentração de Q em valores pequenos faz com que $B = 1$ ocorra com probabilidade maior, conforme a Equação (2.29):

$$\theta_{j,l} = \Pr(B_{j,l} = 1) > \phi + \frac{1}{w+1}. \tag{2.29}$$

Essa diferença de probabilidades é o que permite ao teste distinguir as duas situações: sob H_0 há mais uns, e sob H_1 há menos. Para estabelecer formalmente a fronteira entre elas, introduz-se um parâmetro de margem c, sujeito à restrição $c > 1/(w+1)$, e as hipóteses são reparametrizadas pelas Equações (2.30) e (2.31):

$$H_0 \;:\; \theta_{j,l} \geq (\phi + c) \quad \text{para algum } l = 1, \ldots, k \tag{2.30}$$

$$H_1 \;:\; \theta_{j,l} < (\phi + c) \quad \text{para cada } l = 1, \ldots, k \tag{2.31}$$

A restrição sobre c garante que a fronteira $\phi + c$ situe-se acima do valor esperado sob H_1. Com $w = 49$ e $c = 0{,}06$, a hipótese nula localiza-se em 0,56, acima dos 0,52 esperados sob convergência.

2.5.2.4 Etapa 4: estatística de contagem

O último passo acumula as variáveis binárias ao longo dos testes. A estatística monitorada, definida pela Equação (2.32), é o número de zeros acumulados até o teste j:

$$S_j = jk - \sum_{g=1}^{j} \sum_{l=1}^{k} B_{g,l} \tag{2.32}$$

onde jk é o total de variáveis B acumuladas e o somatório duplo conta quantas delas são iguais a um.

A lógica de decisão decorre do comportamento de B sob cada hipótese. Sob H_0, o valor $B = 1$ é mais frequente, há poucos zeros e S_j tende a ser pequeno. Sob H_1, ocorre o inverso, e S_j tende a ser grande. A convergência é sinalizada quando S_j atinge um valor suficientemente elevado, definido pelo limiar cv_j.

Para ilustrar com $k = 20$, suponha que nos três primeiros blocos os B's iguais a um tenham sido, respectivamente, 12, 11 e 10. A evolução de S_j é apresentada na Equação (2.33):

$$\begin{aligned} S_1 &= 1 \times 20 - 12 = 8 \\ S_2 &= 2 \times 20 - (12 + 11) = 17 \\ S_3 &= 3 \times 20 - (12 + 11 + 10) = 27 \end{aligned} \tag{2.33}$$

Uma propriedade central do método é que, no cenário extremo sob H_0 que maximiza a probabilidade de erro tipo I, com $\theta_{j,l} = \phi + c$ para todo j e l, a estatística S_j segue uma distribuição Binomial exata, conforme a Equação (2.34):

$$S_j \sim \text{Binomial}\left(jk,\ 1 - (\phi + c)\right). \tag{2.34}$$

É essa distribuição exata que separa o SeqTest dos métodos clássicos. Enquanto Geweke e Heidelberger-Welch dependem de aproximações assintóticas, o SeqTest calcula os limiares de sinalização a partir de uma distribuição conhecida, o que garante controle rigoroso do erro tipo I (SILVA *et al.*, 2026).

2.5.3 Funções de gasto α

A realização de múltiplas análises ao longo do tempo requer um mecanismo de controle da probabilidade acumulada de erro tipo I. As funções de gasto α, propostas por Jennison e Turnbull (2000), fornecem esse controle ao especificar a fração do nível de significância global consumida até cada teste.

Silva *et al.* (2026) adotam a família de gasto tipo potência, definida pela Equação (2.35):

$$A(j) = \alpha \left(\frac{jk}{N_e} \right)^{\rho} \tag{2.35}$$

onde α é o nível de significância global, jk é o número de observações binárias acumuladas até o teste j, $N_e = Mk$ é o total de observações binárias submetidas ao teste ao longo de toda a vigilância, e $\rho > 0$ controla a forma do gasto. Valores $\rho < 1$ produzem gasto côncavo, que aloca mais erro nos primeiros testes e favorece a detecção precoce; valores

$\rho > 1$ produzem gasto convexo, mais conservador no início e mais permissivo ao final (SILVA *et al.*, 2020; SILVA, 2018). Este trabalho adota $\rho = 2$. Ao término dos M testes, tem-se $jk = N_e$ e, portanto, $A(M) = \alpha$, de modo que o nível de significância é integralmente consumido.

2.5.4 Limiar de sinalização

Para um dado gasto α, o limiar de sinalização é uma sequência de valores críticos inteiros $cv_1, \ldots, cv_M$, e a regra de decisão consiste em rejeitar H_0 no primeiro j tal que $S_j \geq cv_j$. Os limiares são determinados de modo que a probabilidade acumulada de rejeição até o teste j, sob o cenário mais desfavorável de H_0, não exceda $A(j)$, conforme a Equação (2.36):

$$\Pr\left(\bigcup_{r=1}^{j}\{S_r \geq cv_r\} \;\middle|\; H_0\right) \leq A(j). \tag{2.36}$$

O cálculo é realizado por meio das probabilidades de absorção de um processo binomial com parâmetros jk e $1-(\phi+c)$, implementado no pacote R *Sequential* (SILVA *et al.*, 2021). Os limiares são obtidos de forma exata, sem aproximações assintóticas, o que garante o controle nominal do erro tipo I.

O comprimento da cadeia pode ser determinado previamente em função do poder estatístico desejado. Especificados o risco relativo alvo e o nível de significância, a função `SampleSize.Binomial` fornece o valor de M, e o comprimento N decorre da Equação (2.37):

$$N \geq M \cdot k \cdot (w+1). \tag{2.37}$$

Com $RR = 2$, $\alpha = 0{,}05$, poder de 0,80, $\phi = 0{,}5$, $c = 0{,}06$, $k = 20$ e $w = 49$, obtém-se $M = 80$ e, consequentemente, $N \geq 80.000$ (SILVA *et al.*, 2026). Esses são os valores adotados em todos os experimentos deste trabalho.

2.5.5 Convergência em escala

O teste descrito até aqui detecta convergência em localização e independência entre valores consecutivos. Essas propriedades não garantem convergência em escala: uma cadeia pode ter média constante e ainda apresentar variância que se altera sistematicamente ao longo das iterações. Um exemplo é a sequência $X_i \sim \text{Normal}(100,\ 10000/\sqrt{i})$, cuja média é sempre 100 mas cuja variância diminui progressivamente.

Silva *et al.* (2026) propõem uma extensão para esse caso. A cadeia original é dividida em sub-blocos consecutivos de tamanho k^*, calcula-se uma medida de escala

para cada sub-bloco, como a variância amostral, e aplica-se o SeqTest à nova cadeia formada por essas medidas. Se a variância da cadeia original é estacionária, os valores de escala serão aproximadamente independentes e o teste sinalizará; se a variância muda ao longo do tempo, apresentarão dependência e o teste não sinalizará. A convergência plena requer, portanto, a aplicação do SeqTest duas vezes, à cadeia original e à cadeia de escalas. Esta extensão não é empregada no presente trabalho, que se restringe à convergência em localização.

2.5.6 Parâmetros de calibração

O SeqTest requer a definição de sete parâmetros. A Tabela 2 apresenta cada um deles, com os valores adotados neste trabalho e as restrições aplicáveis.

Tabela 2 – Parâmetros de calibração do SeqTest e valores adotados.

Parâmetro	Significado	Valor adotado	Restrição
k	Estatísticas por teste	20	Inteiro positivo
w	Janela de comparação menos um	49	Ímpar
ϕ	Limiar de binarização	0,5	ϕk inteiro
c	Margem de convergência	0,06	$c > 1/(w+1)$
α	Nível de significância	0,05	$0 < \alpha < 1$
ρ	Forma do gasto α	2	$\rho > 0$
N	Comprimento da cadeia	80.000	Equação (2.37)

Fonte: Adaptado de Silva *et al.* (2026).

Duas restrições merecem atenção. O parâmetro w deve ser ímpar, e $w+1$ deve ser múltiplo de $1/\phi$; com $\phi = 0{,}5$, qualquer valor ímpar de w satisfaz essa condição. A restrição $c > 1/(w+1)$ vincula os dois parâmetros: para $w = 9$ seria necessário $c > 0{,}10$, ao passo que para $w = 49$ o valor $c = 0{,}06$ adotado por Silva *et al.* (2026) é adequado, pois $1/(w+1) = 0{,}02$.

O par de parâmetros w e c determina simultaneamente duas quantidades. A primeira é o tamanho do efeito a ser detectado, dado pela distância entre $\phi + c = 0{,}56$ sob H_0 e $\phi + 1/(w+1) = 0{,}52$ sob H_1, isto é, quatro pontos percentuais. A segunda é o custo em iterações, pois cada bloco consome $k(w+1)$ valores da cadeia para produzir apenas k observações binárias. Valores maiores de w produzem estatísticas $Q_{j,l}$ mais precisas, porém exigem cadeias mais longas para o mesmo número de testes.

2.5.7 Resumo do procedimento

O procedimento completo divide-se em duas fases. Na fase de planejamento, anterior à geração da cadeia, definem-se os parâmetros de calibração, calcula-se o número mínimo de iterações N para o poder desejado por meio da função `SampleSize.Binomial` e da

Equação (2.37), e configura-se o teste sequencial com a função `AnalyzeSetUp.Binomial`, que determina internamente os limiares $cv_1, \ldots, cv_M$.

Na fase de monitoramento, executada durante a geração da cadeia, repete-se o seguinte ciclo. Gera-se um bloco de $k(w+1)$ valores com o amostrador MCMC e transforma-se o bloco em k variáveis binárias. Atualiza-se a estatística acumulada S_j e compara-se com o limiar cv_j por meio da função `Analyze.Binomial`. Se $S_j \geq cv_j$, rejeita-se H_0 e conclui-se pela convergência, sendo o *burn-in* recomendado igual a $j \cdot k \cdot (w+1)$ iterações. Caso contrário, e havendo testes restantes, gera-se o próximo bloco. Esgotados os M testes sem que nenhum limiar seja atingido, não se rejeita H_0 e conclui-se que a cadeia não apresentou evidência de convergência dentro das N iterações.

Nessa última situação, o analista pode aumentar N ou revisar os parâmetros do amostrador, como o ponto de partida, a distribuição proposta ou a parametrização do modelo.

2.6 Aplicações dos Amostradores MCMC

Esta seção apresenta os problemas de aplicação selecionados. As duas primeiras utilizam o modelo de estimação de prevalência com classificadores imperfeitos e o amostrador de Gibbs; a terceira emprega o modelo Beta-Binomial e o algoritmo Metropolis-Hastings. Essa diversidade permite avaliar o SeqTest com amostradores distintos e em cenários com características diferentes de convergência.

2.6.1 Modelo de Prevalência com Classificador Imperfeito

As duas primeiras aplicações compartilham a mesma estrutura matemática: estimar a proporção real de itens pertencentes a uma categoria de interesse, dado que o classificador utilizado para detectá-los comete erros.

Seja S a variável indicadora do status real de um item, com $S = 1$ quando o item pertence à categoria de interesse, e seja R a classificação atribuída pelo sistema automático. O desempenho do classificador é caracterizado por dois parâmetros obtidos previamente em estudos de validação: a sensibilidade $\eta = P(R = 1 \mid S = 1)$, probabilidade de identificar corretamente um item positivo, e a especificidade $\theta = P(R = 0 \mid S = 0)$, probabilidade de identificar corretamente um item negativo.

O parâmetro de interesse é a prevalência real $\psi = P(S = 1)$, que não é diretamente observável. O que se observa é a taxa de classificações positivas do sistema, dada pela Equação (2.38):

$$\tau = P(R = 1) = \psi\eta + (1 - \psi)(1 - \theta) \tag{2.38}$$

na qual o primeiro termo corresponde aos verdadeiros positivos e o segundo aos falsos positivos. Invertendo essa relação, a prevalência real é recuperada pela Equação (2.39):

$$\psi = \frac{\tau + \theta - 1}{\eta + \theta - 1}. \tag{2.39}$$

A estimativa direta r/n, isto é, a proporção de itens marcados como positivos, não corresponde à prevalência real, pois ignora os erros do classificador. A correção por meio das Equações (2.38) e (2.39) é o que motiva a abordagem Bayesiana.

O denominador $\eta + \theta - 1$ da Equação (2.39) tem papel decisivo neste trabalho. Quando a sensibilidade e a especificidade são altas, esse denominador aproxima-se da unidade e as flutuações de τ são transmitidas a ψ com pouca amplificação. Quando são baixas, o denominador aproxima-se de zero e as mesmas flutuações são amplificadas, o que eleva a autocorrelação da cadeia. Trata-se, portanto, de um índice da qualidade de mistura esperada, conhecido antes de qualquer simulação, o que o torna adequado para avaliar a capacidade de discriminação dos métodos de diagnóstico.

2.6.1.1 O Gibbs Sampler para o modelo de prevalência

A estimação Bayesiana de ψ utiliza o amostrador de Gibbs com variáveis latentes, conforme proposto por Larangeira (2012) e empregado por Silva *et al.* (2026). Adota-se uma *prior* Beta(α_0, β_0) para τ, tipicamente não-informativa com $\alpha_0 = \beta_0 = 1$.

Quando o classificador marca r dos n itens como positivos, não se sabe quantos desses são realmente positivos e quantos foram marcados por engano. Da mesma forma, entre os $n - r$ itens não marcados pode haver positivos não detectados. Essa informação oculta é representada por duas variáveis latentes: X, o número de verdadeiros positivos entre os r itens marcados, e Y, o número de falsos negativos entre os $n - r$ itens não marcados. A cada iteração, valores plausíveis de X e Y são imputados com base no valor corrente de ψ, e a estimativa de τ é atualizada em seguida. As distribuições condicionais são dadas pelas Equações (2.40) a (2.42):

$$X \mid r, \psi \sim \text{Binomial}(r,\ p), \quad \text{onde } p = \frac{\psi\eta}{\tau} \tag{2.40}$$

$$Y \mid r, \psi \sim \text{Binomial}(n - r,\ q), \quad \text{onde } q = \frac{\psi(1-\eta)}{1-\tau} \tag{2.41}$$

$$\tau \mid X, Y \sim \text{Beta}(\alpha_0 + V,\ \beta_0 + n - V), \quad \text{onde } V = X + Y \tag{2.42}$$

e ψ é atualizado pela Equação (2.39).

2.6.2 Aplicação 1: Classificação de E-mails SPAM

A filtragem automática de SPAM é um problema clássico de classificação binária, no qual filtros de e-mail rotulam cada mensagem como SPAM ou legítima, com taxas de acerto imperfeitas. O objetivo é estimar a prevalência real de SPAM corrigindo os erros do filtro. Neste contexto, η é a fração de mensagens efetivamente maliciosas que o filtro detecta, e θ é a fração de mensagens legítimas que o filtro corretamente deixa passar. O modelo e o amostrador de Gibbs empregado são os descritos na Seção 2.6.1.

Esta aplicação adota o mesmo modelo utilizado por Silva *et al.* (2026), o que permite situar os resultados obtidos em relação à literatura. Dois cenários são construídos, contrastando um filtro de desempenho moderado com um filtro preciso. A variação de η e θ entre eles altera o denominador de identificabilidade e, com ele, a qualidade de mistura esperada da cadeia, sem que qualquer outro aspecto do modelo seja modificado. Os valores específicos são definidos na Seção 3.5.

2.6.3 Aplicação 2: Detecção de Fraude em Cartão de Crédito

A detecção automática de fraudes em transações financeiras é um problema de crescente relevância no setor bancário. Sistemas antifraude monitoram transações em tempo real e as classificam como legítimas ou suspeitas, com taxas de erro não desprezíveis (BOLTON; HAND, 2002). O modelo matemático e o amostrador de Gibbs são os mesmos da aplicação anterior, descritos na Seção 2.6.1, com S indicando se a transação é de fato fraudulenta e R indicando a classificação do sistema.

Essa aplicação complementa a de SPAM por dois motivos. A prevalência de fraude é tipicamente baixa, entre 1% e 5% das transações, o que reduz a razão sinal-ruído e concentra a *posteriori* em uma região estreita do espaço paramétrico. Além disso, as consequências de um diagnóstico incorreto de convergência são materiais: subestimar a taxa de fraude implica perdas financeiras, ao passo que superestimá-la resulta em bloqueios excessivos de transações legítimas (BOLTON; HAND, 2002).

2.6.4 Aplicação 3: Estimação de Frequência Alélica

A terceira aplicação emprega o algoritmo Metropolis-Hastings, o que permite avaliar o SeqTest com um amostrador de natureza distinta.

Em genética de populações, o equilíbrio de Hardy-Weinberg estabelece que, para um locus com dois alelos A e a, as frequências genotípicas são determinadas pela frequência alélica p: o genótipo AA ocorre com frequência p^2, o genótipo Aa com frequência $2p(1-p)$ e o genótipo aa com frequência $(1-p)^2$ (STEPHENS, 2017).

O problema consiste em estimar p a partir das contagens genotípicas observa-

das (n_{AA}, n_{Aa}, n_{aa}). Adotando-se a *prior* não-informativa Uniforme$(0, 1)$, equivalente a Beta$(1, 1)$, a *likelihood* é dada pela Equação (2.43):

$$L(p) \propto p^{2n_{AA}+n_{Aa}} \cdot (1-p)^{2n_{aa}+n_{Aa}} \tag{2.43}$$

e a *posteriori* resultante é apresentada na Equação (2.44):

$$p \mid \text{dados} \sim \text{Beta}(2n_{AA} + n_{Aa} + 1, \;\; 2n_{aa} + n_{Aa} + 1). \tag{2.44}$$

O fato de a *posteriori* possuir forma fechada confere a esta aplicação um papel particular no experimento. A média teórica é conhecida analiticamente, o que permite confrontar diretamente as estimativas produzidas por cada cadeia com o valor correto, independentemente de qualquer teste de convergência.

A amostragem utiliza o Metropolis-Hastings com caminhada aleatória, com proposta $q(p^* \mid p) = \text{Normal}(p, \sigma^2_{\text{proposta}})$. A eficiência do amostrador depende diretamente de σ_{proposta}, o que permite construir cenários com comportamentos de convergência conhecidos *a priori*. Valores pequenos produzem passos curtos e alta taxa de aceitação, fazendo a cadeia deslocar-se lentamente apesar de raramente rejeitar propostas. Valores grandes produzem rejeições frequentes, deixando a cadeia estagnada em patamares com saltos esporádicos. Entre esses extremos existe uma faixa de calibração adequada, na qual a exploração é eficiente.

Essa estrutura em que os extremos degradam a cadeia por mecanismos opostos, e cuja deficiência decorre da teoria do algoritmo e não do resultado de qualquer teste, torna a aplicação especialmente adequada para avaliar a ocorrência de falsos positivos nos diagnósticos de convergência. Os valores específicos de σ_{proposta} e as contagens genotípicas utilizadas são apresentados na Seção 3.6.

3 Desenvolvimento

Este capítulo descreve a implementação computacional dos amostradores MCMC e do método SeqTest apresentados no Capítulo 2, as escolhas de parâmetros que definem os sete cenários experimentais e o protocolo adotado na comparação com os diagnósticos clássicos.

Toda a implementação está reunida em um único *script* em R, reproduzido integralmente no Apêndice A. O texto que se segue descreve as decisões de implementação e as justifica, remetendo ao Apêndice para o código correspondente.

3.1 Ferramentas Utilizadas

A implementação foi realizada em linguagem R versão 4.4.0 (R CORE TEAM, 2024). Além das funções base, dois pacotes especializados foram empregados.

O pacote *Sequential* (SILVA *et al.*, 2021) implementa testes sequenciais exatos para dados binomiais e de Poisson, e foi desenvolvido pelos mesmos autores do SeqTest. Dele utilizam-se duas funções: `AnalyzeSetUp.Binomial`, que configura o teste sequencial e determina internamente os limiares de sinalização cv_j, e `Analyze.Binomial`, que compara a estatística acumulada S_j com o limiar a cada novo teste.

O pacote *coda* (PLUMMER *et al.*, 2006) implementa os diagnósticos clássicos por meio das funções `geweke.diag`, `heidel.diag` e `raftery.diag`, correspondentes aos testes descritos na Seção 2.4.

Os amostradores MCMC e a função de binarização do SeqTest foram implementados diretamente em R, sem dependência de pacotes adicionais, seguindo as formulações do Capítulo 2 e do artigo de Silva *et al.* (2026).

3.2 Implementação do Gibbs Sampler

O modelo de prevalência com classificador imperfeito (Seção 2.6.1) é utilizado nas aplicações de SPAM e Fraude. O objetivo é gerar amostras da *posteriori* da prevalência real ψ, dados os parâmetros do classificador (η e θ) e as contagens observadas: n itens analisados, dos quais r marcados como positivos. Como a *posteriori* de ψ não possui forma fechada simples, o amostrador recorre às variáveis latentes X e Y para viabilizar a amostragem iterativa. A função `gibbs_prevalencia`, apresentada na Seção A.2 do Apêndice, implementa esse procedimento.

Cada iteração executa quatro passos. Primeiro, a partir do valor corrente da prevalência, recompõe-se a taxa observada τ pela Equação (2.38) e calculam-se as probabilidades condicionais $p = \psi\eta/\tau$, que representa a chance de um item marcado ser realmente positivo, e $q = \psi(1-\eta)/(1-\tau)$, que representa a chance de um item não marcado ser um positivo não detectado. Em seguida, amostram-se as contagens latentes $X \sim \text{Binomial}(r,\ p)$ e $Y \sim \text{Binomial}(n-r,\ q)$, conforme as Equações (2.40) e (2.41).

A soma $V = X + Y$ é o número total de positivos verdadeiros imputado à amostra na iteração corrente. Como V é uma contagem de eventos governados pela prevalência, isto é, $V \sim \text{Binomial}(n, \psi)$, a conjugação Beta-Binomial fornece diretamente a *posteriori* completa de ψ, da qual se amostra o novo valor da cadeia, conforme a Equação (2.42). É esse valor, e não a taxa observada, que constitui o estado da cadeia e é armazenado para análise posterior.

Duas salvaguardas numéricas foram incorporadas. Os valores de ψ, τ, p e q são confinados ao intervalo $(0{,}001;\ 0{,}999)$ antes de alimentarem as funções de amostragem, o que evita erros decorrentes de argumentos fora do domínio válido. Essa precaução é mais relevante nos cenários com η e θ baixos, nos quais o denominador $\eta + \theta - 1$ é pequeno e amplifica as flutuações de τ. Adicionalmente, o caso degenerado $\eta + \theta = 1$, no qual ψ deixa de ser identificável, é tratado pela amostragem direta da *prior*. Esse caso não ocorre em nenhum dos cenários deste trabalho, nos quais o menor denominador é 0,550.

O valor inicial da cadeia é arbitrário, pois o amostrador converge para a *posteriori* independentemente do ponto de partida, desde que seja executado por tempo suficiente. A mesma função atende às aplicações de SPAM e Fraude, bastando alterar os valores de η, θ, n e r.

3.3 Implementação do Metropolis-Hastings

A estimação da frequência alélica sob o equilíbrio de Hardy-Weinberg (Seção 2.6.4) utiliza o Metropolis-Hastings com caminhada aleatória. Diferentemente do Gibbs, que amostra diretamente das condicionais completas, este algoritmo propõe candidatos e decide sobre sua aceitação a partir de uma razão de probabilidades. A função `mh_alelo` consta da Seção A.2 do Apêndice.

Cada iteração propõe um candidato somando ao estado corrente um incremento $\text{Normal}(0,\ \sigma^2_{\text{proposta}})$. Propostas fora do intervalo $(0, 1)$ são descartadas de imediato, o que equivale a atribuir probabilidade nula à *posteriori* fora do suporte da frequência alélica. A razão de aceitação é avaliada em escala logarítmica, pois a comparação $\log(u) < \log(A)$ é equivalente a $u < \min(1, A)$ e evita a perda de precisão que ocorreria ao exponenciar diretamente as *likelihoods*. Como a *prior* é $\text{Uniforme}(0, 1)$ e a proposta é simétrica, tanto o termo da *prior* quanto os termos $q(\cdot \mid \cdot)$ cancelam na razão, conforme a Equação (2.13),

e o cálculo reduz-se à diferença entre as log-*likelihoods* da Equação (2.43).

Os incrementos da proposta e os limiares de aceitação são gerados de forma vetorizada antes do laço. Essa organização é matematicamente equivalente à geração iterativa, pois Normal(μ, σ^2) equivale a μ somado a um incremento Normal$(0, \sigma^2)$, e reduz de modo relevante o tempo de execução, o que viabiliza as 350 réplicas do protocolo experimental.

3.3.1 Taxa de aceitação como indicador de calibração

A função registra a proporção de propostas aceitas, quantidade que serve como indicador direto da calibração do amostrador. Conforme discutido na Seção 2.3.3, a taxa ótima para problemas unidimensionais situa-se em torno de 44%, e valores moderados nessa vizinhança indicam boa exploração do espaço paramétrico.

Taxas muito altas, acima de 80%, indicam que σ_{proposta} é pequeno demais: as propostas são quase idênticas ao estado corrente e, embora quase sempre aceitas, produzem passos curtos e forte autocorrelação. Taxas muito baixas, abaixo de 15%, indicam o oposto: a maioria das propostas cai em regiões de baixa probabilidade e é rejeitada, deixando a cadeia estagnada. Nos experimentos deste trabalho, σ_{proposta} é variado intencionalmente para produzir cenários nos três regimes (Seção 3.6).

A *posteriori* verdadeira desta aplicação é conhecida analiticamente (Equação (2.44)), o que permite validar a implementação de forma independente do teste de convergência: se o amostrador está correto, a média amostral das cadeias deve coincidir com a média teórica da distribuição Beta. Essa comparação, apresentada no Capítulo 4, cumpre um papel adicional na análise, pois revela-se satisfeita mesmo em cadeias que os diagnósticos formais rejeitam, evidenciando que a concordância de estimativas pontuais não constitui, por si só, evidência de convergência.

3.4 Implementação do SeqTest

A aplicação do SeqTest envolve dois componentes: a função de binarização, que implementa as Etapas 2 e 3 da Seção 2.5, e o procedimento de monitoramento sequencial, que implementa a Etapa 4 e a comparação com os limiares. Ambos constam da Seção A.3 do Apêndice.

3.4.1 Função de binarização

A função `Binario`, proposta por Silva *et al.* (2026) e reimplementada neste trabalho a partir do código disponibilizado no apêndice do artigo original, recebe um bloco de $k(w+1)$ valores da cadeia e devolve k variáveis binárias.

A organização dos índices reserva as primeiras k posições do bloco aos valores de referência, e as kw posições restantes às janelas de comparação, de modo que a janela do i-ésimo valor de referência ocupa as posições $k + (i-1)w + 1$ a $k + iw$. Para cada referência, conta-se quantos valores de sua janela lhe são menores ou iguais, e a contagem é normalizada conforme a Equação (2.23). A soma de uma unidade ao numerador garante que o *rank* normalizado nunca seja nulo. A simetrização $Q = \min(V,\ 1 - V)$ faz com que tanto valores muito altos quanto muito baixos de V resultem em Q próximo de zero, o que permite capturar desvios da uniformidade em qualquer direção. Por fim, o limiar $\phi/2 = 0{,}25$ converte Q em variável binária.

Conforme a análise da Seção 2.5, sob H_1 espera-se $\Pr(B = 1) = \phi + 1/(w+1)$, o que corresponde a 52% de uns com os parâmetros adotados. Sob H_0, a dependência entre valores concentra Q em valores pequenos e eleva essa proporção. A margem $c = 0{,}06$ posiciona a hipótese nula em $\phi + c = 0{,}56$, acima do valor esperado sob convergência, e a restrição $c > 1/(w+1)$ é satisfeita, pois $0{,}06 > 0{,}02$.

3.4.2 Configuração do teste sequencial

O comprimento da cadeia foi fixado em 80.000 iterações, valor obtido na Seção 2.5 a partir da calibração recomendada por Silva *et al.* (2026), com risco relativo alvo $RR = 2$, nível de significância $\alpha = 0{,}05$ e poder de 0,80. Com $k = 20$ e $w = 49$, cada bloco consome $k(w+1) = 1.000$ valores da cadeia, o que resulta em $M = 80$ testes sequenciais e em um total de $M \times k = 1.600$ observações binárias submetidas ao teste. Esse valor é adotado em todos os cenários das três aplicações.

A função `AnalyzeSetUp.Binomial` configura o teste e determina internamente os limiares de sinalização. Seus argumentos, e os valores adotados, constam da Tabela 3.

Tabela 3 – Configuração do teste sequencial na função `AnalyzeSetUp.Binomial`.

Argumento	**Valor**	**Significado**
`N`	1.600	Total de eventos binomiais submetidos ao teste, $M \times k$. Não corresponde ao comprimento da cadeia.
`alpha`	0,05	Nível de significância global.
`zp`	1,2727	Razão $z = (1-p_0)/p_0$, com $p_0 = 1-(\phi+c) = 0{,}44$.
`M`	3	Número mínimo de casos acumulados para permitir sinalização. Não é o número de testes.
`AlphaSpendType`	`power-type`	Família da função de gasto de α, Equação (2.35).
`rho`	2	Parâmetro de forma do gasto.

Fonte: Elaborado pelo autor.

Dois argumentos exigem atenção, pois sua interpretação difere da notação adotada nas seções anteriores.

O argumento `N` não corresponde ao comprimento da cadeia MCMC. Trata-se do tamanho amostral máximo da vigilância medido em eventos, isto é, o total de observações binomiais submetidas ao teste ao longo de toda a análise. Como cada teste j contribui com exatamente k observações binárias, o valor a ser informado é

$$\texttt{N} = M \times k = 80 \times 20 = 1.600 \text{ eventos}, \tag{3.1}$$

e não às 80.000 iterações da cadeia. É esse total que serve de referência à função de gasto de α, a qual despende uma fração $\alpha \cdot (n/N)^{\rho}$ do nível nominal até o instante em que n eventos foram observados, de modo que o nível α seja integralmente consumido ao término dos M testes.

O argumento `M` é o número mínimo de casos acumulados a partir do qual uma sinalização é permitida, e não o número de testes. O número de testes decorre implicitamente do total de eventos e do tamanho de cada bloco. Cabe registrar ainda que, com `AlphaSpendType` igual a `power-type`, o pacote não utiliza o argumento `zp` na fase de planejamento, exigindo que a razão z seja informada na fase de monitoramento.

3.4.3 Monitoramento sequencial e cache dos limiares

Na fase de monitoramento, a cadeia é percorrida bloco a bloco. Cada bloco é transformado em k variáveis binárias pela função `Binario`, e as contagens resultantes são submetidas à função `Analyze.Binomial`, que atualiza internamente a estatística acumulada S_j (Equação (2.32)) e a compara com o limiar cv_j (Equação (2.36)). A coluna de decisão é localizada pelo nome, e não por índice fixo, uma vez que a posição das colunas pode variar entre versões do pacote.

A terminologia `cases` e `controls` provém do uso original do pacote em vigilância epidemiológica. No contexto do SeqTest, `cases` corresponde ao número de zeros, indicativos de comportamento compatível com amostra aleatória, e `controls` ao número de uns. Detectada a convergência no teste j, o ponto de convergência na cadeia original é $j \cdot k \cdot (w+1)$, e as iterações anteriores constituem o *burn-in* sugerido. Esgotados os M testes sem que nenhum limiar seja atingido, conclui-se que a cadeia não apresentou evidência suficiente de convergência dentro das iterações geradas.

A execução de 450 cadeias exigiria, nessa formulação direta, o mesmo número de configurações do teste sequencial, cada uma acompanhada de até 80 chamadas ao pacote. Adotou-se, por isso, uma reorganização do procedimento. Os limiares $cv_1, \ldots, cv_M$ dependem apenas dos parâmetros de calibração e da sequência de tamanhos amostrais dos testes, que é fixa por construção, com $k = 20$ eventos por teste, e não dependem dos dados observados. A função `obter_cv` explora essa propriedade: alimenta o pacote com uma sequência sintética de contagens que jamais produz sinalização, o que permite percorrer

os 80 testes e recolher o vetor completo de limiares. Esse vetor é então armazenado e reutilizado em todas as réplicas, nas quais a decisão reduz-se à comparação direta entre S_j e cv_j, sem novas chamadas ao pacote.

3.4.4 Aplicação dos diagnósticos clássicos

Para comparação, os três diagnósticos clássicos são aplicados às mesmas cadeias por meio do pacote *coda*, na função `diagnosticos_classicos` da Seção A.4 do Apêndice. O teste de Geweke devolve um Z-score, do qual se obtém o p-valor bilateral; valores superiores a 0,05 levam à não rejeição da estacionaridade, interpretada como indicativo de convergência, com as ressalvas discutidas na Seção 2.4.5. O teste de Heidelberger-Welch devolve o resultado do teste de estacionaridade, e o procedimento de Raftery-Lewis devolve o tamanho de cadeia recomendado. A mesma função registra ainda a média *a posteriori*, o intervalo de credibilidade de 95% e, nas cadeias de Metropolis-Hastings, a taxa de aceitação.

3.5 Cenários das Aplicações de Gibbs

Esta seção define os quatro cenários gerados pelo amostrador de Gibbs, dois para a aplicação de SPAM e dois para a de Fraude. Todos compartilham o mesmo modelo e a mesma função de amostragem, diferindo apenas nos valores de η, θ, n e r.

A Tabela 4 apresenta os cenários ordenados pelo denominador de identificabilidade $\eta + \theta - 1$, que, conforme discutido na Seção 2.6.1, indexa a qualidade de mistura esperada da cadeia. Em todos eles adota-se a *prior* não-informativa Beta$(1, 1)$ sobre a prevalência.

Tabela 4 – Parâmetros dos quatro cenários gerados por amostrador de Gibbs.

Aplicação	**Cenário**	η	θ	n	r	$\eta + \theta - 1$
SPAM	Difícil	0,75	0,80	500	200	0,550
Fraude	Legado	0,80	0,95	10.000	800	0,750
SPAM	Favorável	0,99	0,95	500	200	0,940
Fraude	Moderno	0,95	0,995	10.000	200	0,945

Fonte: Elaborado pelo autor.

Os dois cenários de SPAM contrastam um filtro de desempenho moderado com um filtro preciso. No cenário difícil, o filtro detecta 75% das mensagens maliciosas e classifica corretamente 80% das legítimas, o que produz o menor denominador do conjunto, 0,550. Esse valor amplifica as flutuações de τ na transformação para ψ, eleva a autocorrelação da cadeia e torna a convergência mais lenta. Com $\tau = r/n = 0{,}40$, a prevalência esperada é $\psi \approx 0{,}364$. No cenário favorável, os valores $\eta = 0{,}99$ e $\theta = 0{,}95$ elevam o denominador a 0,940 e conduzem a $\psi \approx 0{,}372$.

Os dois cenários de Fraude transportam o mesmo modelo para um regime de prevalência baixa. Estudos do setor bancário indicam que a fraude atinge tipicamente entre 1% e 5% das transações (BOLTON; HAND, 2002), o que concentra a *posteriori* em uma região estreita do espaço paramétrico e reduz a razão sinal-ruído. O detector legado, baseado em regras estáticas, apresenta denominador 0,750 e conduz a $\psi \approx 0{,}040$, ao passo que o detector moderno, baseado em aprendizado de máquina, atinge 0,945 e conduz a $\psi \approx 0{,}016$. A baixa taxa de falsos positivos do detector moderno é relevante em aplicações financeiras, nas quais bloqueios excessivos prejudicam a operação.

O conjunto dos quatro cenários cobre, portanto, uma faixa ampla do denominador de identificabilidade, de 0,550 a 0,945, sob dois regimes distintos de prevalência. Espera-se que a qualidade de mistura acompanhe essa ordenação, e é justamente essa expectativa, derivada da estrutura do modelo e independente de qualquer teste, que servirá de critério para avaliar a capacidade de discriminação dos métodos de diagnóstico. Não se antecipa aqui um veredito determinístico para cada cenário: como a decisão do teste depende da realização particular da cadeia, a avaliação é conduzida sobre múltiplas réplicas independentes, conforme o protocolo da Seção 3.7.1.

3.6 Cenários da Aplicação de Frequência Alélica

Diferentemente das aplicações de Gibbs, nas quais os parâmetros do modelo variam, aqui os dados permanecem fixos e a variação experimental incide sobre o parâmetro de calibração do amostrador, σ_{proposta}. Essa característica confere à aplicação um papel particular: a qualidade de mistura de cada cenário decorre da teoria do próprio algoritmo, de forma independente de qualquer teste de convergência, o que permite avaliar se cada método de diagnóstico recupera uma ordenação já conhecida.

3.6.1 Dados genotípicos

As contagens utilizadas são hipotéticas, similares às empregadas em estudos de genética de populações (STEPHENS, 2017), e constam da Tabela 5.

Tabela 5 – Contagens genotípicas utilizadas na aplicação de frequência alélica.

Genótipo	**Contagem**
AA	50
Aa	21
aa	29
Total	**100**

Fonte: Adaptado de Stephens (2017).

Pela Equação (2.44), a *posteriori* verdadeira é Beta$(122, 80)$, cuja média vale $122/202 \approx 0{,}604$. Esse valor serve de referência para validar o amostrador e, no Capítulo 4, para confrontar as estimativas pontuais com as decisões dos métodos de diagnóstico.

3.6.2 Cenários de calibração

Três valores de σ_{proposta} produzem os três regimes de comportamento do Metropolis-Hastings, apresentados na Tabela 6.

Tabela 6 – Cenários da aplicação de frequência alélica.

Cenário	$\sigma_{\textbf{proposta}}$	**Taxa de aceitação**	**Comportamento esperado**
1	0,005	Muito alta (acima de 90%)	Cadeia "rasteja"
2	0,05	Moderada (40% a 60%)	Exploração eficiente
3	0,5	Muito baixa (abaixo de 10%)	Frequentes rejeições

Fonte: Elaborado pelo autor.

Com $\sigma_{\text{proposta}} = 0{,}005$, os passos são extremamente curtos e quase todas as propostas são aceitas, precisamente porque quase nada muda. A cadeia desloca-se lentamente pelo espaço paramétrico, a autocorrelação entre valores consecutivos é elevada e muitas iterações seriam necessárias para explorar o suporte da *posteriori*. Espera-se que o SeqTest não sinalize convergência dentro das 80.000 iterações.

Com $\sigma_{\text{proposta}} = 0{,}05$, a taxa de aceitação situa-se na faixa em que o amostrador opera de forma eficiente, próxima do ótimo teórico de 44% para problemas unidimensionais (ROBERT; CASELLA, 2004). Este é o regime de melhor mistura entre os três, e espera-se que o SeqTest sinalize convergência na maioria das réplicas.

Com $\sigma_{\text{proposta}} = 0{,}5$, os passos são excessivamente grandes e a maioria das propostas cai em regiões de baixíssima probabilidade, sendo rejeitada. A cadeia permanece imóvel por longos períodos, com saltos esporádicos entre patamares. Espera-se novamente que o SeqTest não sinalize convergência.

Os Cenários 1 e 3 constituem patologias reconhecidas do algoritmo, cuja deficiência de mistura decorre da teoria e não do resultado de qualquer teste. São, por isso, o material mais adequado para quantificar a ocorrência de falsos positivos nos diagnósticos de convergência.

3.7 Procedimento de Comparação com Diagnósticos Clássicos

3.7.1 Protocolo experimental

A decisão do SeqTest depende da realização particular da cadeia analisada. Avaliar cada cenário a partir de uma única cadeia produziria um veredito binário sem medida

alguma de variabilidade e, portanto, sem base para afirmar que a diferença observada entre dois cenários é sistemática, e não fruto do acaso. Adota-se, por essa razão, um protocolo baseado em réplicas independentes.

Para cada um dos sete cenários definidos nas Seções 3.5 e 3.6, geram-se $R = 50$ cadeias independentes com o amostrador apropriado, cada uma com 80.000 iterações. O SeqTest é aplicado a cada réplica, registrando-se o teste j da sinalização ou a sua ausência, e os três diagnósticos clássicos são aplicados à mesma réplica. Consolidam-se, por cenário, a proporção de réplicas em que cada método declarou convergência, bem como a mediana e a amplitude do teste de sinalização. Ao todo, $7 \times 50 = 350$ cadeias são geradas e analisadas, o que permite reportar taxas de certificação em lugar de decisões isoladas. O laço completo consta da Seção A.5 do Apêndice.

3.7.1.1 Sementes aleatórias

Não se fixa uma semente por cenário. Fixá-la garantiria reprodutibilidade, mas restringiria cada cenário a uma única realização, que é exatamente o que as réplicas pretendem evitar, e abriria margem à suspeita de que a semente foi escolhida em função do resultado obtido. Adota-se, em vez disso, uma semente-mestra sorteada a partir do relógio do sistema no início da execução e registrada no arquivo de log. As 350 cadeias são, assim, todas distintas e nenhuma foi selecionada pelo autor, ao passo que a execução permanece integralmente reproduzível a partir da semente registrada.

3.7.1.2 Controles de validação

Um teste que jamais sinalizasse produziria, nos cenários de má mistura, resultados aparentemente idênticos aos de um teste corretamente calibrado. As duas situações não podem ser distinguidas examinando-se apenas os cenários de interesse. Para separá-las, o SeqTest é aplicado, com 50 réplicas cada, a dois conjuntos de referência.

O controle positivo é uma amostra independente e identicamente distribuída extraída exatamente da *posteriori* Beta$(122, 80)$, gerada pela função `rbeta`. Tratando-se de independência perfeita, nenhuma cadeia MCMC pode superá-la: um teste bem calibrado deve sinalizá-la em todas as réplicas, e o teste mediano de sinalização define um piso empírico de desempenho, contra o qual os cenários reais podem ser comparados de forma quantitativa. O controle negativo é um processo autorregressivo de primeira ordem com coeficiente 0,99, cuja dependência serial é extrema e que não deve ser sinalizado em réplica alguma por um teste com controle adequado do erro tipo I.

3.7.1.3 Custo computacional

Graças ao armazenamento do vetor de limiares descrito na Seção 3.4, cada réplica requer apenas a geração da cadeia, sua binarização e a comparação de S_j com o limiar,

sem novas chamadas ao pacote *Sequential*. As 450 cadeias do experimento, somando 36 milhões de iterações MCMC, são processadas em poucos minutos, o que torna o protocolo de réplicas viável em um computador pessoal.

3.7.2 Critérios de comparação

A comparação organiza-se em torno de quatro critérios:

O primeiro é a calibração da implementação. Antes de qualquer comparação, verifica-se que o SeqTest sinaliza convergência no controle positivo e não a sinaliza no controle negativo. Sem essa verificação, nenhuma conclusão sobre os cenários de estudo seria interpretável, pois um teste inoperante e um teste rigoroso produziriam resultados idênticos nos cenários de má mistura.

O segundo, e central, é a capacidade de discriminação. Avalia-se se a taxa de certificação de cada método acompanha a qualidade efetiva da mistura. Nos cenários de Gibbs, a qualidade é indexada pelo denominador $\eta + \theta - 1$ (Tabela 4); nos de Metropolis-Hastings, pela calibração de σ_{proposta} (Tabela 6). Um método útil deve certificar mais frequentemente as cadeias de melhor mistura; um método cego produzirá taxas semelhantes em todos os cenários, independentemente da qualidade.

O terceiro é a incidência de falsos positivos em cadeias patológicas. Nos regimes mal calibrados do Metropolis-Hastings, a deficiência de mistura é demonstrável de forma independente do teste, tanto pela teoria do algoritmo quanto pelas taxas de aceitação observadas. Quantifica-se, nesses casos, a proporção de réplicas em que cada método declara convergência incorretamente.

O quarto é a robustez à escolha de parâmetros. Os parâmetros do SeqTest são fixados conforme as recomendações de Silva *et al.* (2026) e mantidos constantes em todos os cenários e réplicas, e os diagnósticos clássicos são utilizados com os valores-padrão do pacote *coda*. Essa uniformidade elimina a possibilidade de ajuste *post hoc* em favor de qualquer método.

3.7.3 Apresentação dos resultados

Os resultados, apresentados no Capítulo 4, organizam-se em três níveis. Tabelas-resumo comparam, para cada cenário, a proporção de réplicas em que cada método declarou convergência, acompanhada da mediana do teste de sinalização e do tamanho amostral sugerido por Raftery-Lewis. Gráficos diagnósticos exibem, para cada cenário, o *traceplot* da cadeia e a trajetória da estatística S_j frente ao limiar, tornando explícito o momento da decisão. Por fim, a análise textual discute os casos em que o SeqTest e os métodos clássicos divergem, que constituem a principal contribuição empírica do trabalho.

4 Resultados

Este capítulo apresenta os resultados da aplicação do SeqTest e dos diagnósticos clássicos aos sete cenários das três aplicações. Conforme o protocolo da Seção 3.7.1, cada cenário foi avaliado sobre 50 cadeias independentes, totalizando 350 cadeias. Essa escolha permite reportar taxas de certificação em lugar de decisões isoladas e, com isso, distinguir diferenças sistemáticas entre cenários de meras flutuações amostrais, distinção que um experimento de cadeia única não comporta.

A Seção 4.1 apresenta os controles de validação da implementação, cujo exame precede necessariamente a interpretação dos demais resultados. As seções seguintes tratam de cada aplicação, e a Seção 4.5 consolida a comparação entre os métodos.

4.1 Validação da implementação

Um teste de convergência que jamais sinalizasse produziria, nos cenários de má mistura, resultados aparentemente idênticos aos de um teste corretamente calibrado. As duas situações não podem ser distinguidas examinando-se apenas os cenários de interesse. A implementação foi, portanto, submetida a dois controles de validação, cada um com 50 réplicas, cujos resultados constam da Tabela 7.

Tabela 7 – Controles de validação da implementação, sobre 50 réplicas independentes.

Controle	**Sinalizou**	**Teste mediano**	**Comportamento esperado**
Positivo: i.i.d. Beta$(122, 80)$	50/50 (100%)	25	sinalizar sempre
Negativo: AR(1) com $\rho = 0{,}99$	0/50 (0%)	n/a	nunca sinalizar

Fonte: Elaborado pelo autor.

O controle positivo consiste em uma amostra independente e identicamente distribuída extraída exatamente da *posteriori* Beta$(122, 80)$ da aplicação de frequência alélica. Tratando-se de independência perfeita, nenhuma cadeia MCMC pode superá-la. O SeqTest sinalizou convergência nas 50 réplicas, com teste mediano igual a 25 dos 80 disponíveis, o que corresponde à iteração 25.000 de 80.000, e amplitude de 7 a 56. Esse valor constitui um piso empírico de desempenho: sob a calibração adotada, nenhuma cadeia MCMC deve sinalizar sistematicamente antes do teste 25, e a proximidade a esse piso passa a ser uma medida direta da qualidade da mistura.

O controle negativo é um processo autorregressivo de primeira ordem com coeficiente 0,99, cuja dependência serial é extrema. Ele não foi sinalizado em nenhuma das 50 réplicas,

o que confirma que o teste não certifica cadeias fortemente dependentes e que o controle do erro tipo I opera conforme o esperado.

A amplitude observada no controle positivo merece registro. Mesmo sob independência perfeita, o teste de sinalização variou entre 7 e 56, um fator superior a sete. Essa variabilidade é invisível em um experimento de cadeia única e é precisamente o que justifica o protocolo de réplicas. Ela também recomenda cautela na interpretação de qualquer sinalização isolada, cuja posição depende tanto da qualidade da cadeia quanto do acaso.

4.2 Aplicação: Classificação de SPAM

Os dois cenários da aplicação de SPAM foram avaliados sobre 50 cadeias independentes cada, geradas pelo amostrador de Gibbs com 80.000 iterações. A Tabela 8 consolida as decisões dos métodos.

Tabela 8 – Resultados dos diagnósticos para a aplicação de SPAM, sobre 50 réplicas por cenário.

Cenário	**SeqTest**	**Teste med.**	**Geweke**	**Heidel-Welch**	**R-L ($N_{\mathbf{sug}}$)**
Difícil (η=0,75)	30/50 (60%)	56,5	49/50 (98%)	50/50 (100%)	13.803
Favorável (η=0,99)	50/50 (100%)	27,5	41/50 (82%)	50/50 (100%)	3.986

Fonte: Elaborado pelo autor. Valores medianos entre as réplicas.

4.2.1 Cenário favorável

O classificador de alta qualidade ($\eta = 0{,}99$, $\theta = 0{,}95$) produz cadeias com boa mistura. O denominador de identificabilidade $\eta + \theta - 1 = 0{,}94$, próximo da unidade, mantém as flutuações de τ pouco amplificadas na transformação para ψ (Equação (2.39)), o que se traduz em baixa autocorrelação entre iterações sucessivas.

O SeqTest sinalizou convergência nas 50 réplicas, com teste mediano 27,5, correspondente à iteração 27.500 de 80.000. Esse valor é próximo do piso empírico de 25 estabelecido pelo controle positivo (Seção 4.1), o que indica que a cadeia se comporta, para os fins do teste, de modo quase indistinguível de uma amostra independente. A prevalência estimada foi $\hat{\psi} = 0{,}373$, contra o valor teórico de 0,372.

A Figura 1 apresenta o *traceplot* e o monitoramento de S_j para a primeira réplica. A trajetória de S_j cresce de forma sustentada e ultrapassa o limiar bem antes do fim da janela de vigilância. Os diagnósticos clássicos também indicaram convergência na maioria das réplicas, embora o teste de Geweke tenha rejeitado a convergência em 9 das 50 cadeias (18%), a maior taxa de rejeição de todo o experimento, justamente no cenário de melhor mistura. Esse ponto é retomado na Seção 4.5.

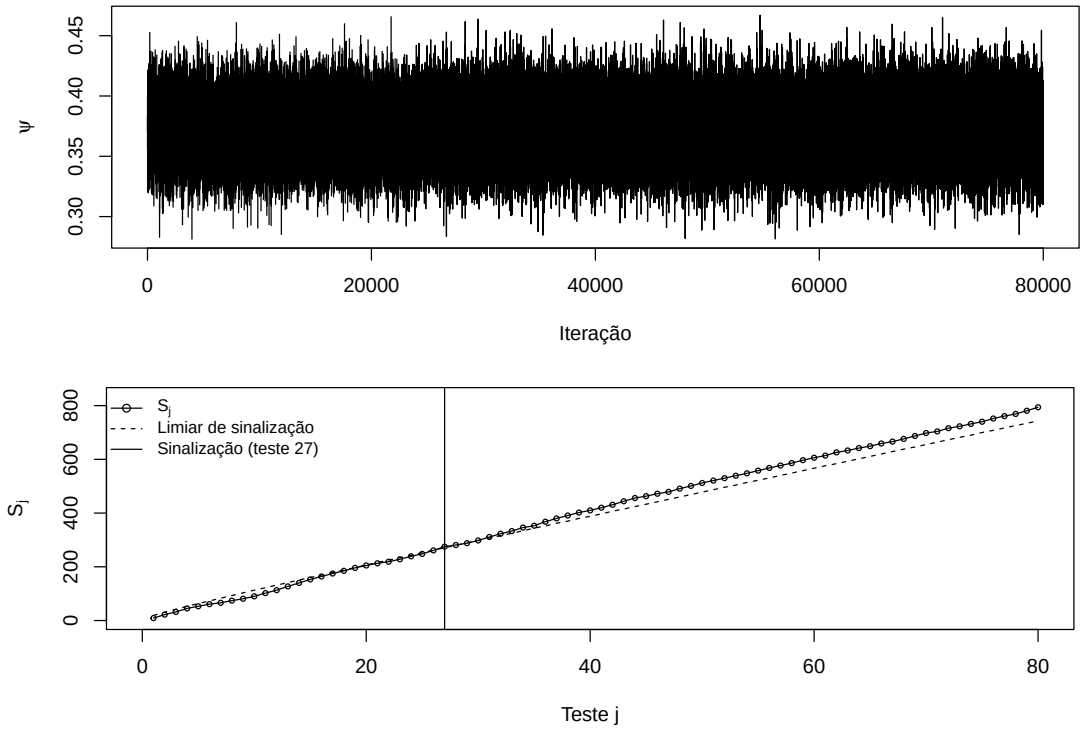

Figura 1 – Cenário favorável de SPAM ($\eta = 0{,}99$, $\theta = 0{,}95$), primeira réplica. Acima: *traceplot* da prevalência ψ. Abaixo: monitoramento da estatística S_j frente ao limiar de sinalização.

4.2.2 Cenário difícil

O classificador moderado ($\eta = 0{,}75$, $\theta = 0{,}80$) produz o menor denominador de identificabilidade entre os cenários de Gibbs, $\eta + \theta - 1 = 0{,}55$. Esse valor amplifica as flutuações de τ na transformação para ψ, elevando a autocorrelação e reduzindo o tamanho efetivo da amostra.

O SeqTest sinalizou convergência em apenas 30 das 50 réplicas (60%) e, quando o fez, tardiamente: teste mediano 56,5, o mais alto entre todos os cenários sinalizados. Os diagnósticos clássicos, em contraste, certificaram convergência em praticamente todas as réplicas, com Geweke em 49/50 e Heidelberger-Welch em 50/50.

A leitura correta desse resultado não se reduz a um veredito binário. A distribuição de S_{80} entre as réplicas tem mediana 742,5, praticamente coincidente com o limiar $cv_{80} = 744$: as cadeias deste cenário situam-se sobre a fronteira de decisão. O SeqTest não afirma que a cadeia é inadequada, tampouco que é adequada. Informa, com precisão, que se trata de um caso limítrofe, que exige cerca de 57.000 iterações quando a sinalização ocorre e que, em 40% das realizações, não é certificado nem com 80.000. Nenhum diagnóstico clássico transmite essa informação.

A Figura 2 evidencia a origem da divergência. O *traceplot* exibe uma cadeia

aparentemente estável, oscilando em torno de $\hat{\psi} = 0{,}364$, valor próximo ao 0,373 do cenário favorável e essencialmente idêntico ao teórico. É precisamente essa estabilidade superficial que os testes baseados em médias parciais interpretam como convergência. A patologia reside na estrutura de dependência, e é o crescimento comparativamente lento de S_j que a revela.

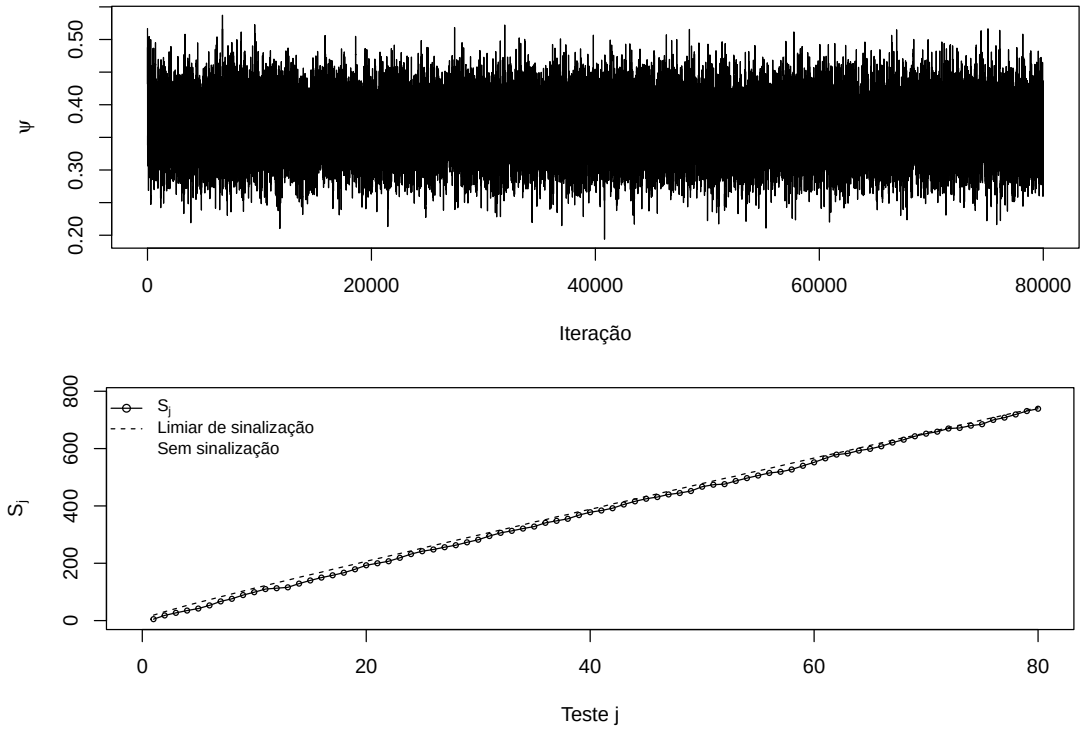

Figura 2 – Cenário difícil de SPAM ($\eta = 0{,}75$, $\theta = 0{,}80$), primeira réplica. Acima: *traceplot* de ψ, com aparência estável. Abaixo: monitoramento de S_j, que acompanha o limiar sem folga.

4.3 Aplicação: Detecção de Fraude

Os dois cenários de detecção de fraude foram avaliados sobre 50 cadeias independentes cada, com $n = 10.000$ transações e 80.000 iterações. A Tabela 9 apresenta os resultados.

Tabela 9 – Resultados dos diagnósticos para a aplicação de Fraude, sobre 50 réplicas por cenário.

Cenário	**SeqTest**	**Teste med.**	**Geweke**	**Heidel-Welch**	**R-L ($N_{\mathbf{sug}}$)**
Legado (η=0,80; r=800)	31/50 (62%)	47,0	47/50 (94%)	50/50 (100%)	13.872
Moderno (η=0,95; r=200)	50/50 (100%)	29,5	48/50 (96%)	50/50 (100%)	4.249

Fonte: Elaborado pelo autor. Valores medianos entre as réplicas.

4.3.1 Cenário moderno

O detector moderno ($\eta = 0{,}95$, $\theta = 0{,}995$) foi certificado pelo SeqTest em todas as 50 réplicas, com teste mediano 29,5, novamente próximo ao piso empírico de 25. A prevalência estimada foi $\hat{\psi} = 0{,}016$, contra 0,0159 teórico, o que corresponde a 1,6% de fraude.

O resultado é, à primeira vista, contraintuitivo, pois o detector moderno concentra a *posteriori* em uma região muito mais estreita que a do legado, o que poderia sugerir maior dificuldade de exploração. O que prevalece, contudo, é a identificabilidade: o denominador $\eta + \theta - 1 = 0{,}945$, próximo da unidade, quase não amplifica as flutuações de τ, produzindo cadeias de baixa autocorrelação. A Figura 3 ilustra esse comportamento, com um *traceplot* que ocupa uma faixa estreita em torno de 0,016 e uma trajetória de S_j que cruza o limiar com folga.

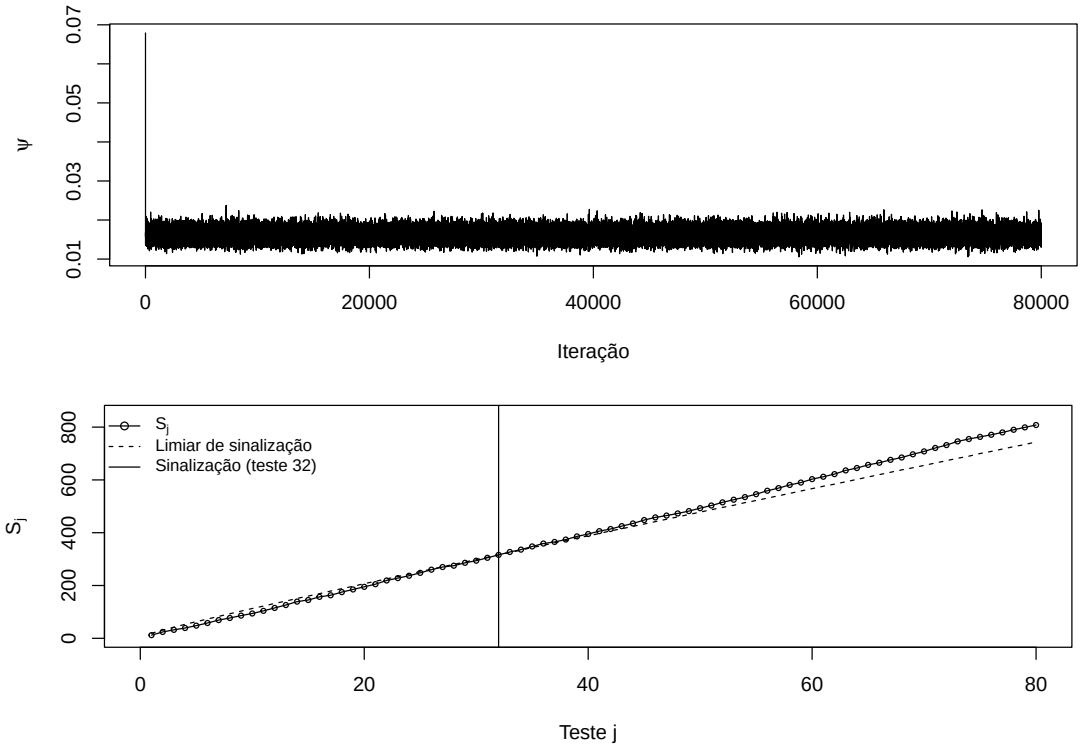

Figura 3 – Cenário de detector moderno ($\eta = 0{,}95$, $\theta = 0{,}995$), primeira réplica. Acima: *traceplot* de ψ. Abaixo: monitoramento de S_j, com sinalização confortavelmente dentro da janela.

4.3.2 Cenário legado

O detector legado ($\eta = 0{,}80$, $\theta = 0{,}95$) apresenta denominador $\eta + \theta - 1 = 0{,}75$, intermediário entre o do cenário moderno (0,945) e o do cenário difícil de SPAM (0,55). O SeqTest o certificou em 31 das 50 réplicas (62%), com teste mediano 47, comportamento igualmente intermediário, exatamente como a teoria de identificabilidade antecipa. A

prevalência estimada foi $\hat{\psi} = 0{,}040$, coincidente com o valor teórico.

Os diagnósticos clássicos, mais uma vez, não distinguem os dois detectores: Geweke certifica 94% das cadeias legadas e 96% das modernas, e Heidelberger-Welch certifica 100% de ambas. O SeqTest, por sua vez, os separa de modo inequívoco (62% contra 100%) e ainda quantifica a diferença de eficiência (47 contra 29,5 testes medianos).

A distribuição de S_{80} tem mediana 745,5, ligeiramente acima do limiar $cv_{80} = 744$, o que caracteriza mais um caso de fronteira, ainda que um pouco mais favorável que o do SPAM difícil. A Figura 4 mostra um *traceplot* com oscilações regulares em faixa estreita, visualmente compatível com boa exploração da *posteriori*, e uma trajetória de S_j que acompanha o limiar de perto.

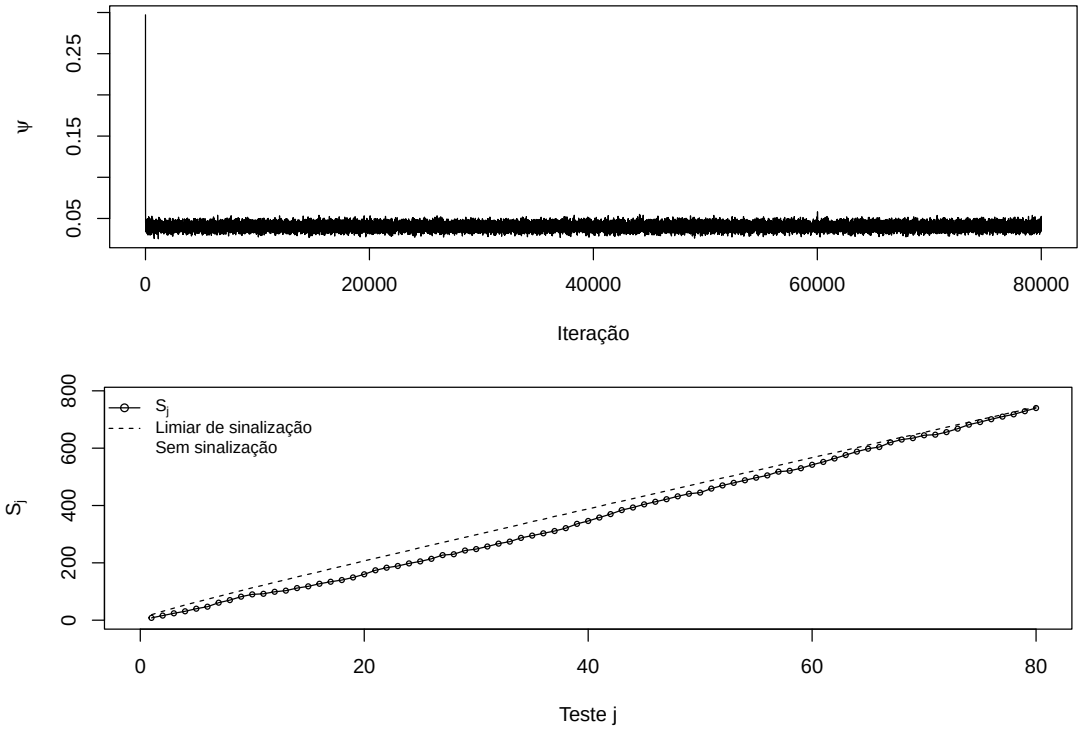

Figura 4 – Cenário de detector legado ($\eta = 0{,}80$, $\theta = 0{,}95$), primeira réplica. Acima: *traceplot* de ψ. Abaixo: monitoramento de S_j, em regime limítrofe.

4.4 Aplicação: Equilíbrio de Hardy-Weinberg

Os três cenários da aplicação de frequência alélica foram avaliados sobre 50 cadeias independentes cada, geradas pelo algoritmo Metropolis-Hastings com 80.000 iterações, variando o desvio-padrão da distribuição proposta. A Tabela 10 resume os resultados.

Tabela 10 – Resultados dos diagnósticos para a aplicação HWE, sobre 50 réplicas por cenário.

Cenário	**Aceit.**	**SeqTest**	**Teste med.**	**Geweke**	**H-W**	**R-L (N_{sug})**
$\sigma = 0{,}005$	95,4%	0/50 (0%)	n/a	47/50 (94%)	50/50 (100%)	214.722
$\sigma = 0{,}05$	60,0%	41/50 (82%)	34,0	48/50 (96%)	50/50 (100%)	15.814
$\sigma = 0{,}5$	8,7%	0/50 (0%)	n/a	46/50 (92%)	50/50 (100%)	55.181

Fonte: Elaborado pelo autor. Valores medianos entre as réplicas.

Esta aplicação oferece a evidência mais nítida do trabalho, pois a qualidade da mistura pode ser estabelecida de forma independente do teste. A teoria do Metropolis-Hastings prevê que valores extremos de σ_{proposta} degradam a cadeia por mecanismos opostos e que existe uma faixa intermediária de calibração adequada. O SeqTest recupera exatamente esse comportamento; os diagnósticos clássicos, não.

4.4.1 Cenário com $\sigma = 0{,}05$ (calibração adequada)

Com taxa de aceitação mediana de 60,0%, este é o regime de melhor mistura entre os três. A média *a posteriori* obtida foi $\hat{p} = 0{,}604$, coincidente com a média teórica da distribuição Beta$(122, 80)$ derivada na Seção 3.6.

O SeqTest certificou convergência em 41 das 50 réplicas (82%), com teste mediano 34. O *traceplot* da Figura 5 exibe o aspecto característico de uma cadeia bem-comportada, densamente distribuída em torno de 0,60.

A taxa de 82%, inferior aos 100% dos dois melhores cenários de Gibbs, reflete uma propriedade estrutural do algoritmo, e não uma deficiência do teste. Mesmo bem calibrada, uma caminhada aleatória Metropolis preserva autocorrelação de curto alcance por construção, pois cada estado depende diretamente do anterior por meio do mecanismo de aceitação e rejeição. O SeqTest capta essa diferença de natureza entre os amostradores, que os diagnósticos clássicos não registram.

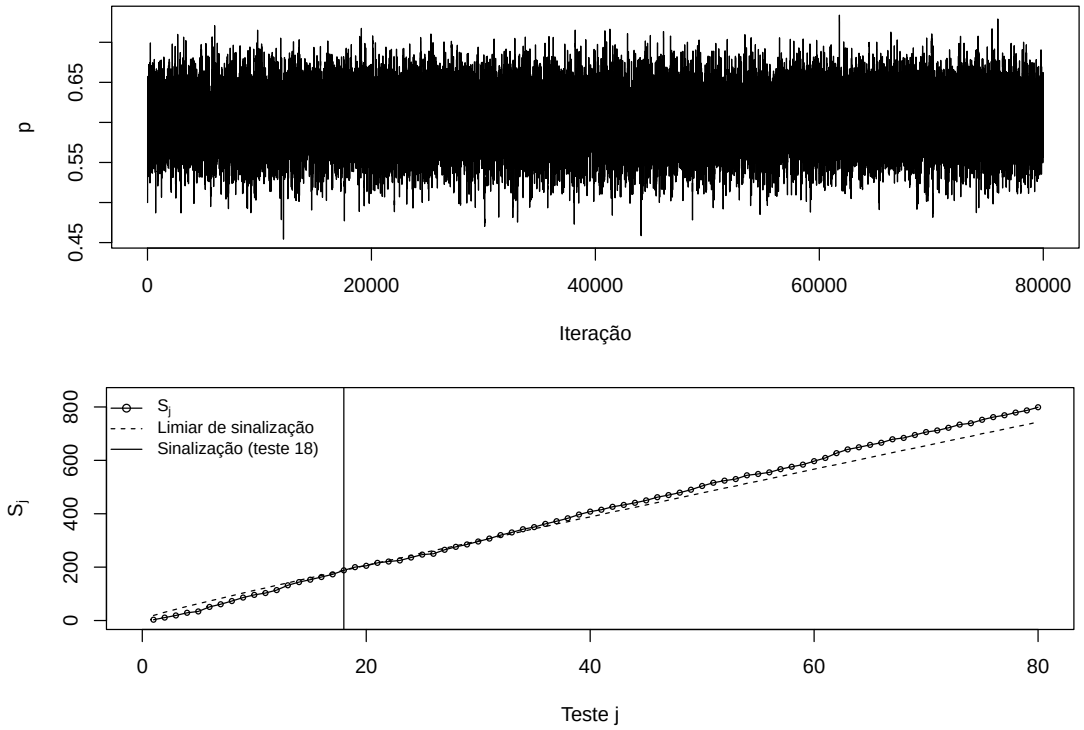

Figura 5 – HWE com $\sigma = 0{,}05$ (calibração adequada), primeira réplica. Acima: *traceplot* de p, com mistura densa em torno de 0,60. Abaixo: monitoramento de S_j.

4.4.2 Cenário com $\sigma = 0{,}005$ (passos curtos)

Com desvio-padrão cem vezes menor que o do cenário adequado, as propostas são quase idênticas ao estado corrente. A taxa de aceitação mediana de 95,4% reflete essa proximidade: quase tudo é aceito, precisamente porque quase nada muda. A cadeia avança lentamente pelo espaço paramétrico, fenômeno conhecido como *random walk* lento.

O SeqTest não sinalizou convergência em nenhuma das 50 réplicas. Geweke certificou 47 (94%) e Heidelberger-Welch, 50 (100%). Trata-se do exemplo mais contundente de falso positivo dos diagnósticos clássicos em todo o experimento.

O mecanismo de falha é transparente. Como a cadeia permanece aproximadamente na mesma região durante toda a execução, as médias de porções distintas tendem a coincidir, e o teste de Geweke conclui, equivocadamente, pela convergência. O teste de Heidelberger-Welch, por avaliar estacionaridade, também nada detecta, uma vez que a cadeia é localmente estacionária mesmo sem haver percorrido o suporte da *posteriori*. A distância até o limiar não deixa margem à ambiguidade: a mediana de S_{80} é 269, contra um limiar de 744, pouco mais de um terço do necessário.

O procedimento de Raftery-Lewis merece registro. Sugeriu 214.722 iterações, quase o triplo das 80.000 executadas, indicação independente de que a cadeia é insuficiente e que se alinha à abstenção do SeqTest.

A Figura 6 torna o diagnóstico visual: a cadeia descreve excursões lentas e suaves, perceptíveis como ondulações de baixa frequência, e a trajetória de S_j cresce vagarosamente, mantendo-se muito distante do limiar.

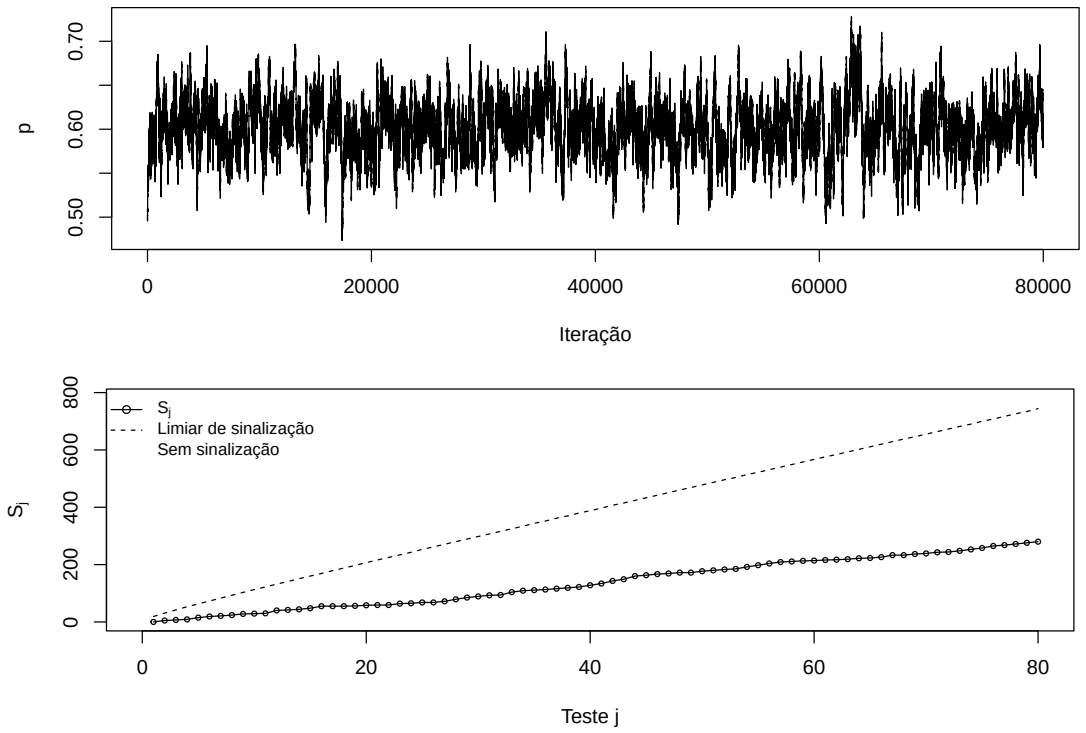

Figura 6 – HWE com $\sigma = 0{,}005$ (passos curtos), primeira réplica. Acima: *traceplot* de p, com excursões lentas de baixa frequência. Abaixo: monitoramento de S_j, muito abaixo do limiar.

4.4.3 Cenário com $\sigma = 0{,}5$ (passos grandes)

No extremo oposto, propostas amplas caem frequentemente fora da região de alta probabilidade da *posteriori*. A taxa de aceitação mediana de 8,7% indica que mais de 91% das propostas são rejeitadas, o que mantém a cadeia estagnada no mesmo estado por longos períodos, com saltos esporádicos entre patamares.

Novamente, o SeqTest não sinalizou convergência em nenhuma das 50 réplicas, enquanto Geweke certificou 46 (92%) e Heidelberger-Welch, 50 (100%). A mediana de S_{80} foi 523, contra o limiar de 744.

O mecanismo de falha difere do cenário anterior. Lá a cadeia movia-se continuamente, ainda que devagar; aqui ela permanece imóvel na maior parte do tempo, produzindo longas repetições do mesmo valor, que geram uma aparência de estacionaridade suficiente para enganar os testes baseados em médias parciais.

Merece atenção o comportamento do Raftery-Lewis neste cenário. O procedimento sugeriu 55.181 iterações, valor inferior às 80.000 executadas. Pela regra usual de leitura,

segundo a qual a cadeia é suficiente quando $N_{\text{sug}} < N_{\text{executado}}$, ele emitiria sinal verde para uma cadeia comprovadamente estagnada. Isso demonstra que nem mesmo o Raftery-Lewis, apesar de haver alertado corretamente no cenário de passos curtos, constitui salvaguarda confiável.

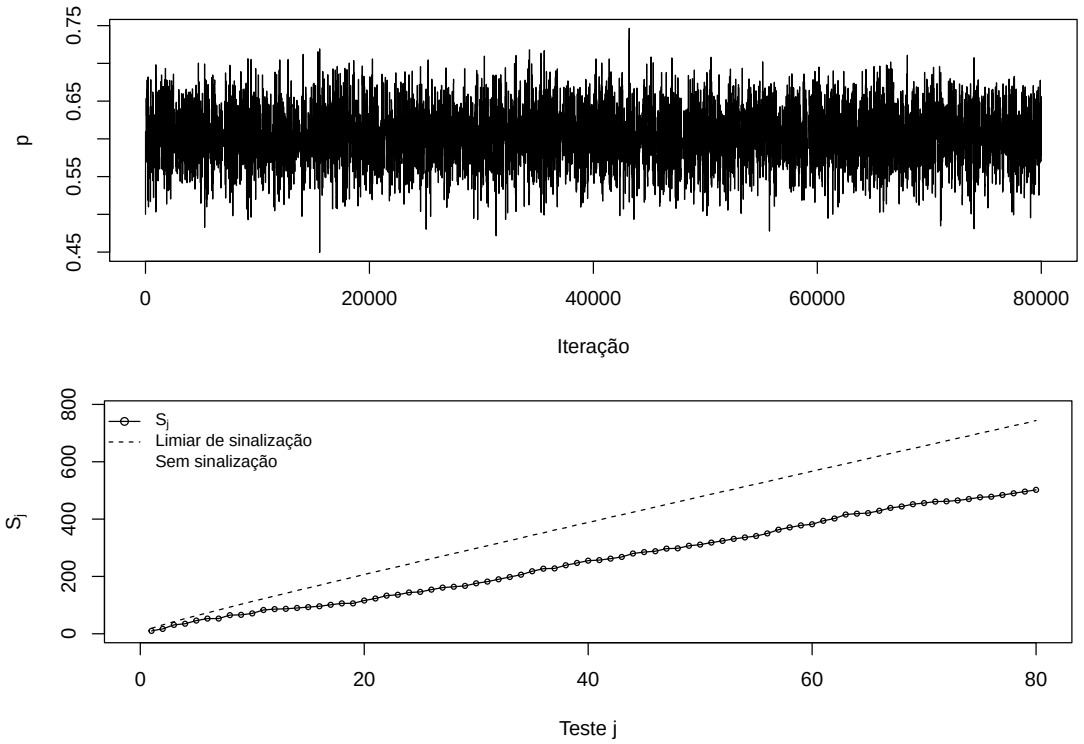

Figura 7 – HWE com $\sigma = 0{,}5$ (passos grandes), primeira réplica. Acima: *traceplot* de p, com patamares de estagnação e saltos abruptos. Abaixo: monitoramento de S_j, distante do limiar.

4.5 Análise Comparativa

A Tabela 11 consolida os resultados das 350 cadeias dos cenários de estudo, acrescidos das 100 réplicas dos controles de validação.

Tabela 11 – Panorama comparativo: proporção de cadeias declaradas convergentes por cada método, sobre 50 réplicas independentes por cenário.

Aplicação	**Cenário**	**SeqTest**	**Teste med.**	**Geweke**	**Heidel-Welch**
Controle positivo (i.i.d.)		50/50	25,0	n/a	n/a
Controle negativo (AR(1))		0/50	n/a	n/a	n/a
SPAM	Favorável (η=0,99)	50/50	27,5	41/50	50/50
Fraude	Moderno (η=0,95)	50/50	29,5	48/50	50/50
HWE	$\sigma = 0{,}05$	41/50	34,0	48/50	50/50
Fraude	Legado (η=0,80)	31/50	47,0	47/50	50/50
SPAM	Difícil (η=0,75)	30/50	56,5	49/50	50/50
HWE	$\sigma = 0{,}005$	0/50	n/a	47/50	50/50
HWE	$\sigma = 0{,}5$	0/50	n/a	46/50	50/50
Total (350 cadeias)		**202 (58%)**	n/a	**326 (93%)**	**350 (100%)**

Fonte: Elaborado pelo autor. Cenários ordenados pela taxa de certificação do SeqTest.

O contraste é imediato. Nas 350 cadeias analisadas, o teste de Heidelberger-Welch declarou convergência em todas, sem uma única exceção. O teste de Geweke declarou convergência em 326 (93%). O SeqTest certificou 202 (58%), distribuídas de modo fortemente desigual entre os cenários. A Figura 8 apresenta essa comparação graficamente.

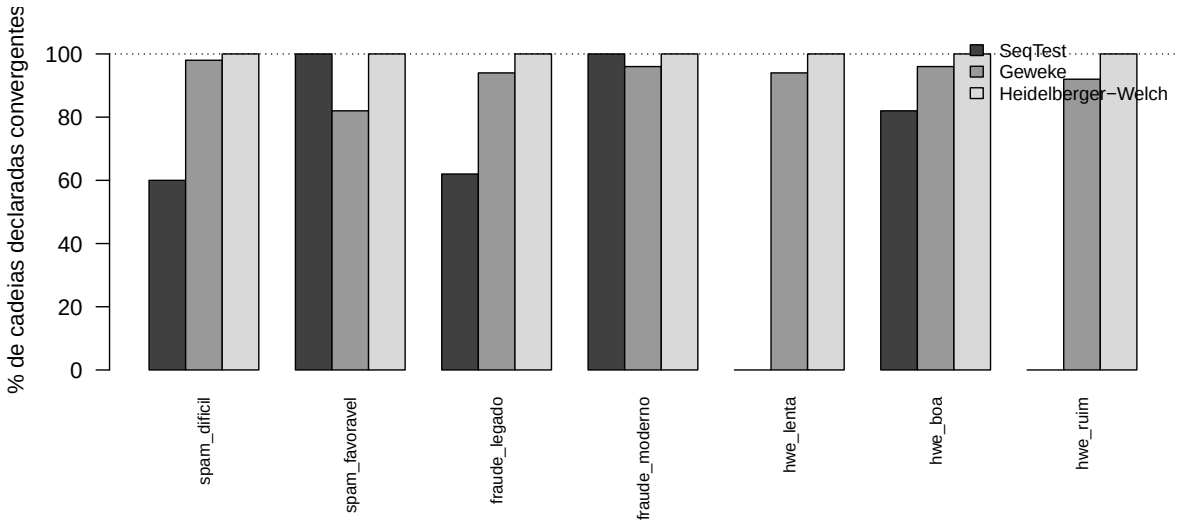

Figura 8 – Percentual de cadeias declaradas convergentes por cada método, em cada cenário, sobre 50 réplicas independentes.

4.5.1 Discriminação entre cadeias boas e ruins

O ponto central não é que o SeqTest certifique menos cadeias, pois um teste que jamais sinalizasse também o faria. O ponto é que a taxa de certificação do SeqTest

acompanha a qualidade efetiva da mistura, ao passo que as dos diagnósticos clássicos não acompanham nada.

Nos quatro cenários gerados pelo amostrador de Gibbs, a qualidade da mistura é indexada, por construção do modelo, pelo denominador de identificabilidade $\eta + \theta - 1$: quanto menor, maior a amplificação das flutuações de τ na transformação para ψ e, portanto, maior a autocorrelação. A Tabela 12 confronta esse índice com o desempenho dos métodos.

Tabela 12 – Relação entre o denominador de identificabilidade e a decisão de cada método, nos quatro cenários gerados por amostrador de Gibbs.

Cenário	$\eta + \theta - 1$	**SeqTest**	**Teste mediano**	**Geweke**	**Heidel-Welch**
SPAM difícil	0,550	30/50 (60%)	56,5	49/50 (98%)	50/50 (100%)
Fraude legado	0,750	31/50 (62%)	47,0	47/50 (94%)	50/50 (100%)
SPAM favorável	0,940	50/50 (100%)	27,5	41/50 (82%)	50/50 (100%)
Fraude moderno	0,945	50/50 (100%)	29,5	48/50 (96%)	50/50 (100%)

Fonte: Elaborado pelo autor.

A relação é monotônica em ambas as métricas. À medida que o denominador cresce, a taxa de certificação do SeqTest aumenta (60%, 62%, 100% e 100%) e o teste mediano de sinalização diminui (56,5, 47,0, 27,5 e 29,5). Trata-se de uma confirmação empírica direta do mecanismo teórico postulado na modelagem: o SeqTest recupera, a partir apenas da cadeia observada, a ordenação de qualidade que decorre da estrutura do modelo. A taxa de certificação do Geweke, exibida na mesma tabela, não apresenta padrão algum, e a do Heidelberger-Welch permanece em 100% nos quatro cenários, independentemente da qualidade da mistura.

Nos três cenários de Metropolis-Hastings, a qualidade é indexada pela calibração de σ_{proposta}, e a teoria prevê degradação nos dois extremos. O SeqTest recupera precisamente esse comportamento em U invertido: 0% de certificação com passos curtos, 82% com calibração adequada e 0% com passos grandes. Os diagnósticos clássicos produzem, nos mesmos três cenários, taxas de 94%, 96% e 92% (Geweke) e de 100% em todos eles (Heidelberger-Welch), isto é, nenhuma discriminação.

4.5.2 O comportamento do teste de Geweke

O exame das taxas de rejeição do teste de Geweke revela um comportamento ainda mais problemático que a simples ausência de discriminação. Ordenadas da menor para a maior, as taxas foram: 2% no SPAM difícil, 4% no fraude moderno e no HWE com $\sigma = 0{,}05$, 6% no fraude legado e no HWE com $\sigma = 0{,}005$, 8% no HWE com $\sigma = 0{,}5$ e 18% no SPAM favorável.

As taxas oscilam entre 2% e 18%, em torno do nível de significância nominal de 5%, e não guardam relação com a qualidade da cadeia. Mais grave, a maior taxa de rejeição ocorre justamente no cenário de melhor mistura, e a menor, em um dos piores. O teste de Geweke não está detectando falta de convergência, está rejeitando aproximadamente à taxa do seu erro tipo I, de forma essencialmente aleatória.

Esse achado tem explicação estrutural. Ao formular a estacionaridade como hipótese nula, o teste de Geweke atribui à convergência o benefício da dúvida, e só a rejeita mediante evidência forte em contrário. Como a evidência que ele examina, a diferença entre médias parciais, é justamente aquela que a autocorrelação não altera, o teste permanece cego ao problema. O SeqTest inverte a formulação, tomando a não-convergência como hipótese nula, de modo que a certificação exige evidência positiva de independência. É essa inversão, e não uma diferença de sensibilidade, que explica a disparidade de desempenho.

4.5.3 A insuficiência das estimativas pontuais

Um resultado transversal a todas as aplicações merece destaque. A Tabela 13 compara a média *a posteriori* estimada em cada cenário com o respectivo valor teórico.

Tabela 13 – Estimativas pontuais medianas entre as 50 réplicas, confrontadas com os valores teóricos.

Cenário	**Estimada**	**Teórica**	**Erro**	**SeqTest**
SPAM difícil	0,3644	0,3636	0,0009	30/50
SPAM favorável	0,3728	0,3723	0,0005	50/50
Fraude legado	0,0401	0,0400	0,0001	31/50
Fraude moderno	0,0162	0,0159	0,0003	50/50
HWE $\sigma = 0{,}005$	0,6038	0,6040	0,0002	0/50
HWE $\sigma = 0{,}05$	0,6040	0,6040	0,0000	41/50
HWE $\sigma = 0{,}5$	0,6040	0,6040	0,0000	0/50

Fonte: Elaborado pelo autor.

Em todos os sete cenários, inclusive nos dois em que o SeqTest não certificou uma única das 50 réplicas, a estimativa pontual coincide com o valor teórico até a terceira ou quarta casa decimal. A cadeia de passos curtos, que se desloca tão lentamente que não explora o suporte da *posteriori*, produz média amostral de 0,6038 contra valor teórico de 0,6040.

Essa constatação é a justificativa mais direta para a existência de métodos formais de diagnóstico. A média amostral de uma cadeia mal misturada converge para o valor correto pelo Teorema Ergódico. O que a autocorrelação compromete não é a localização da estimativa, mas o seu tamanho efetivo de amostra e, por consequência, toda a quantificação de incerteza, o que inclui intervalos de credibilidade, erros-padrão e testes derivados da *posteriori*. Uma cadeia com autocorrelação elevada fornece a resposta certa acompanhada

de uma medida de precisão indevidamente otimista, e nenhuma inspeção de médias ou de *traceplots* é capaz de revelá-lo.

4.5.4 Síntese

Os resultados são consistentes com as conclusões de Silva *et al.* (2026), que demonstraram, por meio de extenso estudo de simulação, que os diagnósticos clássicos podem exibir taxas de falso positivo superiores a 50% em condições adversas de mistura. Neste trabalho, sobre as 100 cadeias de Metropolis-Hastings mal calibrado, cuja deficiência é estabelecida independentemente pela teoria do algoritmo e confirmada pelas taxas de aceitação de 95,4% e 8,7%, o teste de Heidelberger-Welch declarou convergência em 100, e o de Geweke, em 93. O SeqTest, em nenhuma.

Em contrapartida, o SeqTest certificou integralmente os dois cenários de Gibbs bem condicionados, com teste mediano de sinalização (27,5 e 29,5) muito próximo do piso empírico estabelecido pelo controle positivo (25), comportando-se, para efeito do teste, de maneira quase indistinguível de uma amostra independente. Não se trata, portanto, de um teste excessivamente conservador, mas de um teste que discrimina.

5 Considerações finais

Este trabalho teve por objetivo aplicar o método SeqTest de teste sequencial de convergência, proposto por Silva *et al.* (2026), a amostradores de Monte Carlo via Cadeias de Markov em três aplicações distintas, comparando seu desempenho ao dos diagnósticos clássicos de Geweke, Heidelberger-Welch e Raftery-Lewis. As duas primeiras aplicações, classificação de SPAM e detecção de fraude, empregam o amostrador de Gibbs sobre o modelo de prevalência com classificador imperfeito; a terceira, estimação de frequência alélica, emprega o Metropolis-Hastings sobre o modelo Beta-Binomial.

O método foi implementado em linguagem R com o pacote *Sequential*, empregando os valores críticos exatos calculados pelas funções `AnalyzeSetUp.Binomial` e `Analyze.Binomial` e os parâmetros recomendados no artigo original ($k = 20$, $w = 49$, $\phi = 0{,}5$, $c = 0{,}06$, $\alpha = 0{,}05$, $\rho = 2$), o que resultou em $M = 80$ testes sequenciais sobre cadeias de 80.000 iterações. Cada um dos sete cenários foi avaliado sobre 50 cadeias independentes, totalizando 350 cadeias, acrescidas de 100 réplicas destinadas à validação da implementação.

5.1 Principais resultados

A adoção de controles de validação mostrou-se elemento essencial do protocolo, e não mera formalidade. Sem eles, um teste inoperante e um teste rigoroso produziriam resultados indistinguíveis nos cenários de má mistura. O controle positivo, uma amostra independente extraída exatamente da *posteriori*, foi sinalizado nas 50 réplicas, com teste mediano igual a 25; o controle negativo, um processo autorregressivo de coeficiente 0,99, não foi sinalizado em nenhuma. Além de atestar a calibração, o controle positivo estabeleceu um piso empírico de desempenho contra o qual os cenários reais puderam ser comparados de forma quantitativa.

O achado central é que os diagnósticos clássicos revelaram-se incapazes de discriminar cadeias bem e mal comportadas. O teste de Heidelberger-Welch declarou convergência nas 350 cadeias, sem uma única exceção. O teste de Geweke o fez em 326, com taxa de rejeição oscilando em torno de seu erro tipo I nominal e sem qualquer relação com a qualidade da mistura, chegando a rejeitar mais frequentemente a cadeia de melhor mistura de todo o experimento do que aquelas comprovadamente patológicas. O procedimento de Raftery-Lewis, embora tenha alertado corretamente no regime de passos curtos, emitiu sinal verde para a cadeia estagnada do regime de passos grandes, o que o desqualifica como salvaguarda isolada.

O SeqTest, por sua vez, produziu uma resposta graduada que acompanha monotonicamente a qualidade da mistura. Nos cenários de Gibbs, sua taxa de certificação e sua velocidade de sinalização ordenaram-se segundo o denominador de identificabilidade do modelo, o que significa recuperar, a partir apenas da cadeia observada, uma propriedade que decorre da estrutura teórica e é conhecida antes de qualquer simulação. Nos cenários de Metropolis-Hastings, recuperou o comportamento em U previsto pela teoria do algoritmo, certificando 82% das cadeias bem calibradas e nenhuma das 100 cadeias dos dois regimes patológicos. Essa capacidade de ordenar cenários por qualidade, e não apenas de emitir vereditos binários, constitui a principal contribuição empírica do trabalho.

Um resultado transversal reforça a necessidade de diagnósticos formais. Nos sete cenários, inclusive nos dois em que nenhuma das 50 réplicas foi certificada, as estimativas pontuais coincidiram com os valores teóricos até a terceira ou quarta casa decimal. A média amostral de uma cadeia mal misturada é correta, pois converge para o valor esperado pelo Teorema Ergódico. O que a autocorrelação compromete é o tamanho efetivo da amostra e, com ele, toda a quantificação de incerteza. Uma inspeção baseada em médias ou em *traceplots* é, portanto, estruturalmente incapaz de revelar o problema.

5.2 Implicações práticas

Os resultados sustentam uma recomendação clara ao praticante da inferência Bayesiana. Uma sinalização do SeqTest constitui evidência forte de convergência, com controle exato da probabilidade de falso alarme ao nível $\alpha = 0{,}05$. A ausência de sinalização deve ser lida como recomendação de prolongar a cadeia ou de reconfigurar o amostrador, e não como prova de não-convergência, dado que o poder do teste é finito, como evidencia a variabilidade observada mesmo sob independência perfeita.

Os diagnósticos clássicos, ao formularem a estacionaridade como hipótese nula e ao examinarem estatísticas que a autocorrelação não altera, tendem a certificar convergência com facilidade excessiva, e seu uso isolado como critério de decisão não é recomendável. Isso não os torna inúteis. O Raftery-Lewis, quando sugere um tamanho amostral muito superior ao executado, fornece um alerta valioso, e os *traceplots* mantêm sua utilidade na identificação de patologias grosseiras. O que os resultados demonstram é que essas ferramentas devem complementar, e não substituir, um teste formal com garantias probabilísticas explícitas.

5.3 Limitações

O trabalho apresenta limitações que convém explicitar. Primeiro, cada cenário foi avaliado a partir de réplicas de cadeias únicas, e não de múltiplas cadeias simultâneas com

pontos iniciais dispersos, o que impede o emprego do diagnóstico de Gelman-Rubin e a avaliação de convergência entre cadeias.

Segundo, os parâmetros de calibração w e c foram fixados nos valores recomendados no artigo original e não submetidos a análise de sensibilidade. Essa escolha tem consequências mensuráveis. Sob H_1 espera-se $\Pr(B = 1) = \phi + 1/(w + 1) = 0{,}52$, contra $\phi + c = 0{,}56$ sob H_0, de modo que o efeito a ser detectado é de apenas quatro pontos percentuais. O par de parâmetros w e c determina simultaneamente o tamanho desse efeito e o número de observações binomiais extraídas por iteração da cadeia, $1/(w+1)$, e governa conjuntamente o poder do teste e seu custo computacional. A variabilidade observada no controle positivo, com sinalização entre os testes 7 e 56 sob independência perfeita, é manifestação direta dessa configuração.

Terceiro, monitorou-se um único parâmetro por cadeia e apenas a convergência em localização. A extensão do método para convergência em escala, descrita na Seção 2.5, não foi aplicada.

5.4 Trabalhos futuros

Essas limitações sugerem direções naturais de continuidade. A mais promissora é um estudo de sensibilidade ao par de parâmetros w e c, respeitada a restrição $c > 1/(w+1)$, com o objetivo de mapear o compromisso entre poder estatístico e eficiência computacional, questão que o presente trabalho identificou mas não explorou.

Seria igualmente proveitoso aplicar a extensão do método para convergência em escala, de modo a diagnosticar não apenas a estabilização da localização, mas também a da variância das cadeias. Uma comparação sistemática com o diagnóstico de Gelman-Rubin, empregando múltiplas cadeias com pontos iniciais dispersos, permitiria contrastar o SeqTest com a principal alternativa formal disponível na prática corrente.

Por fim, a metodologia de validação aqui adotada, baseada em controles positivo e negativo avaliados sobre réplicas independentes, mostrou-se suficientemente informativa para ser recomendada como prática padrão na avaliação de qualquer diagnóstico de convergência, e não apenas do SeqTest. Sua aplicação sistemática aos diagnósticos clássicos, sobre um conjunto ampliado de cenários, contribuiria para delimitar com precisão as condições sob as quais cada método é confiável.

5.5 Perspectivas para a Engenharia de Controle e Automação

Como aluno de Engenharia de Controle e Automação, encerro apontando três direções em que o SeqTest, ou o princípio de teste sequencial exato que o sustenta, pode ser levado a problemas da área, independentemente de envolverem ou não amostradores

MCMC. O que torna o método atraente nesses contextos é a mesma propriedade explorada ao longo deste trabalho: decidir, sobre um fluxo de dados que chega ao longo do tempo, se ele se estabilizou em um regime estacionário, sob controle formal e exato da taxa de falso alarme.

A primeira direção é a detecção de falhas baseada em modelo. O sinal de resíduo, isto é, a diferença entre a saída medida e a predita, deve comportar-se como uma sequência estacionária e sem estrutura enquanto o sistema opera em condições normais, ao passo que a ocorrência de uma falha introduz correlação ou viés nesse resíduo. Como o SeqTest sinaliza o início de um regime estacionário, e não o seu abandono, ele não se presta a flagrar o instante em que a falha rompe esse regime; presta-se, sim, à decisão complementar de certificar que o resíduo retornou a um comportamento independente e identicamente distribuído após um distúrbio ou uma ação de acomodação de falha, condição para reabilitar com segurança a lógica de monitoramento em regime. Nessa decisão, o método insere-se na mesma família de ferramentas sequenciais do teste sequencial da razão de probabilidades (SPRT) e das cartas CUSUM, das quais se distingue pelo controle exato do erro tipo I por meio das funções de gasto alpha, propriedade especialmente valiosa quando alarmes falsos e falhas não detectadas têm custos fortemente assimétricos.

A segunda direção é o monitoramento de sensores por inferência (*soft sensors*). Esses sensores estimam uma variável de difícil medição, como composição ou uma propriedade de qualidade, a partir de variáveis de fácil aquisição, por meio de um modelo empírico identificado sobre dados de regime permanente e que se degrada ao longo do tempo em razão de desgaste, incrustação ou desvio do processo, exigindo recalibração periódica. Respeitada a direção do teste, o SeqTest atua em dois momentos. Na construção do modelo, pode automatizar a seleção dos trechos estacionários das bases históricas, tarefa hoje conduzida de forma manual ou heurística. Após uma recalibração, pode certificar, com controle exato de falso alarme, que o erro entre a estimativa do *soft sensor* e medições de referência esporádicas, como análises de laboratório, assentou em um regime estacionário e centrado, condição para que a estimativa volte a ser considerada confiável, sem depender das aproximações assintóticas que afetam os testes clássicos de deriva.

A terceira direção é a detecção de regime permanente. Diversas tarefas de automação industrial, como a reconciliação de dados, a otimização em tempo real e a avaliação de desempenho de malhas, pressupõem que o processo tenha atingido o estado estacionário, e decidir quando uma variável deixou o transitório e se estabilizou é, essencialmente, a mesma pergunta que o SeqTest responde para uma cadeia MCMC: a partir de qual instante o sinal passa a se comportar como uma amostra estacionária. Empregar o SeqTest como detector de regime permanente ofereceria um critério não-paramétrico com garantias probabilísticas explícitas, em substituição às heurísticas baseadas em janelas móveis e limiares empíricos usualmente adotadas na prática.

Referências

BOLTON, Richard J.; HAND, David J. Statistical fraud detection: a review. *Statistical Science*, v. 17, n. 3, p. 235–255, 2002. DOI: https://doi.org/10.1214/ss/1042727940. Citado 3 vezes nas páginas 36, 44.

CASELLA, George; GEORGE, Edward I. Explaining the Gibbs sampler. *The American Statistician*, v. 46, n. 3, p. 167–174, 1992. DOI: https://doi.org/10.1080/00031305.1992.10475878. Citado 4 vezes nas páginas 14, 17, 22, 23.

COWLES, Mary Kathryn; CARLIN, Bradley P. Markov chain Monte Carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, v. 91, n. 434, p. 883–904, 1996. DOI: https://doi.org/10.1080/01621459.1996.10476956. Citado 6 vezes nas páginas 14, 15, 17, 24, 26.

GAMERMAN, Dani; LOPES, Hedibert F. *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*. 2. ed.: Chapman e Hall/CRC, 2006. ISBN 978-1584885870. Citado 1 vez na página 21.

GELMAN, Andrew *et al. Bayesian Data Analysis*. 3. ed.: Chapman e Hall/CRC, 2013. ISBN 978-1439840955. Citado 2 vezes nas páginas 14, 23.

GEMAN, Stuart; GEMAN, Donald. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6, n. 6, p. 721–741, 1984. DOI: https://doi.org/10.1109/TPAMI.1984.4767596. Citado 3 vezes nas páginas 14, 17, 22.

GENTLE, James E. *Computational Statistics*. New York: Springer, 2009. ISBN 978-0387981437. Citado 2 vezes na página 20.

GEWEKE, John. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. *In:* BERNARDO, José M. *et al.* (ed.). *Bayesian Statistics 4*. Oxford: Clarendon Press, 1992. p. 169–193. Citado 2 vezes nas páginas 14, 25.

HASTINGS, W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, v. 57, n. 1, p. 97–109, 1970. DOI: https://doi.org/10.1093/biomet/57.1.97. Citado 3 vezes nas páginas 14, 17, 23.

HEIDELBERGER, Philip; WELCH, Peter D. Simulation run length control in the presence of an initial transient. *Operations Research*, v. 31, n. 6, p. 1109–1144, 1983. DOI: https://doi.org/10.1287/opre.31.6.1109. Citado 2 vezes nas páginas 15, 26.

JENNISON, Christopher; TURNBULL, Bruce W. *Group Sequential Methods with Applications to Clinical Trials*. London: Chapman e Hall/CRC, 2000. ISBN 978-0849303166. Citado 1 vez na página 31.

KASS, Robert E. *et al.* Markov chain Monte Carlo in practice: a roundtable discussion. *The American Statistician*, v. 52, n. 2, p. 93–100, 1998. DOI: https://doi.org/10.1080/00031305.1998.10480547. Citado 2 vezes nas páginas 14, 22.

LARANGEIRA, Gustavo. *Three simple applications of Markov chains and Gibbs sampling*. 2012. Disponível em: https://rstudio-pubs-static.s3.amazonaws.com/279858_010f9da7c8d744988019397e3fe51cb2.html. Acesso em: 15 set. 2025. Citado 2 vezes nas páginas 17, 35.

METROPOLIS, Nicholas *et al.* Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, v. 21, n. 6, p. 1087–1092, 1953. DOI: https://doi.org/10.1063/1.1699114. Citado 3 vezes nas páginas 14, 17, 23.

PLUMMER, Martyn *et al.* coda: Convergence Diagnosis and Output Analysis for MCMC. *R News*, v. 6, n. 1, p. 7–11, 2006. Disponível em: https://cran.r-project.org/package=coda. Citado 2 vezes nas páginas 17, 38.

R CORE TEAM. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. 2024. Disponível em: https://www.R-project.org/. Citado 2 vezes nas páginas 17, 38.

RAFTERY, Adrian E.; LEWIS, Steven M. How many iterations in the Gibbs sampler? *In:* BERNARDO, José M. *et al.* (ed.). *Bayesian Statistics 4*. Oxford: Oxford University Press, 1992. p. 763–773. Citado 3 vezes nas páginas 15, 26.

ROBERT, Christian P.; CASELLA, George. *Monte Carlo Statistical Methods*. 2. ed. New York: Springer, 2004. ISBN 978-0387212395. Citado 8 vezes nas páginas 14, 20, 21, 23, 24, 45.

ROY, Vivekananda. Convergence diagnostics for Markov chain Monte Carlo. *Annual Review of Statistics and Its Application*, v. 7, p. 387–412, 2020. DOI: https://doi.org/10.1146/annurev-statistics-031219-041300. Citado 3 vezes nas páginas 15, 17, 24.

SILVA, Ivair R. Type error 1 probability spending for post-market drug and vaccine safety surveillance with binomial data. *Statistics in Medicine*, v. 37, n. 1, p. 107–118, 2018. DOI: https://doi.org/10.1002/sim.7504. Citado 1 vez na página 32.

SILVA, Ivair R.; KULLDORFF, Martin; YIH, Wei Katherine. Optimal alpha spending for sequential analysis with binomial data. *Journal of the Royal Statistical Society, Series B*, v. 82, n. 4, p. 1141–1164, 2020. DOI: https://doi.org/10.1111/rssb.12369. Citado 1 vez na página 32.

SILVA, Ivair R.; MARO, Julio; KULLDORFF, Martin. Exact sequential test for clinical trials and post-market drug and vaccine safety surveillance with Poisson and binary data. *Statistics in Medicine*, v. 40, n. 22, p. 4890–4913, 2021. DOI: https://doi.org/10.1002/sim.9100. Citado 3 vezes nas páginas 17, 32, 38.

SILVA, Ivair R.; NGASSI, Roger C. N.; MOREIRA, Gladston J. P. Exact non-parametric sequential convergence test for samplers. *Journal of Computational and Applied Mathematics*, v. 475, p. 117051, 2026. DOI: https://doi.org/10.1016/j.cam.2025.117051. Citado 24 vezes nas páginas 15–17, 26, 27, 31–33, 35, 36, 38, 40, 41, 47, 61, 62, 69.

STEPHENS, Matthew. *Simple Examples of Metropolis-Hastings Algorithm*. fiveMinuteStats. 2017. Disponível em: https://stephens999.github.io/fiveMinuteStats/MH-examples1.html. Citado 2 vezes nas páginas 36, 44.

APÊNDICE A – Código-Fonte em R

Este apêndice reproduz o *script* utilizado nas simulações. A execução requer os pacotes `Sequential` e `coda`. As 450 cadeias do experimento, somando 36 milhões de iterações MCMC, são processadas em poucos minutos em um computador pessoal.

Omitem-se as rotinas de geração de gráficos, de registro em arquivo de log e de escrita das tabelas em disco, que não influenciam os resultados e cuja reprodução ocuparia espaço sem acrescentar conteúdo metodológico. Preservam-se integralmente a configuração dos parâmetros, os amostradores, a implementação do SeqTest, os diagnósticos clássicos e o laço de execução.

A.1 Configuração

Os parâmetros de calibração seguem as recomendações de Silva *et al.* (2026) e permanecem constantes em todos os cenários e réplicas. A partir deles derivam-se o tamanho do bloco, o número de testes e o total de eventos binomiais submetidos ao teste sequencial.

```
library(Sequential)
library(coda)

R_REP      <- 50         # replicas por cenario
N_CADEIA   <- 80000      # iteracoes por cadeia
K          <- 20         # k  : estatisticas por teste
W          <- 49         # w  : janela de comparacao menos um
PHI        <- 0.5        # phi: limiar de binarizacao
CC         <- 0.06       # c  : margem de convergencia
ALPHA      <- 0.05       # nivel de significancia
RHO        <- 2          # forma do gasto de alfa
M_MIN      <- 3          # min. de casos para permitir sinal

BLOCO    <- K * (W + 1)           # 1000 iteracoes por teste
M_TOTAL  <- N_CADEIA %/% BLOCO    # 80 testes
N_EVENT  <- M_TOTAL * K           # 1600 eventos binomiais
```

A.2 Amostradores MCMC

A função `gibbs_prevalencia` implementa o amostrador de Gibbs com variáveis latentes para o modelo de prevalência (Seção 3.2). O estado da cadeia é a prevalência

ψ, cuja *posteriori* completa, dada a contagem latente de positivos verdadeiros V, é a distribuição Beta$(\alpha_0 + V,\ \beta_0 + n - V)$.

```
gibbs_prevalencia ← function(n_iter, eta, theta, n, r,
                             alpha0 = 1, beta0 = 1) {
  psi ← numeric(n_iter)
  tau ← 0.5
  denom ← eta + theta - 1

  for (t in 1:n_iter) {
    # prevalencia corrente (salvaguarda para o caso nao identificavel)
    prev ← if (abs(denom) < 1e-10) rbeta(1, alpha0, beta0)
           else (tau + theta - 1) / denom
    prev ← max(0.001, min(0.999, prev))

    # probabilidades condicionais das variaveis latentes
    tau_c ← max(0.001, min(0.999, prev*eta + (1-prev)*(1-theta)))
    p ← max(0.001, min(0.999, prev*eta / tau_c))
    q ← max(0.001, min(0.999, prev*(1-eta) / (1-tau_c)))

    # X ~ Binom(r, p) e Y ~ Binom(n-r, q); V = X + Y
    V ← rbinom(1, r, p) + rbinom(1, n - r, q)

    # posteriori conjugada da prevalencia
    psi[t] ← rbeta(1, alpha0 + V, beta0 + n - V)

    # taxa observada induzida, para a proxima iteracao
    tau ← psi[t]*eta + (1 - psi[t])*(1 - theta)
  }
  psi
}
```

A função `mh_alelo` implementa o Metropolis-Hastings com caminhada aleatória para a frequência alélica (Seção 3.3). Os incrementos da proposta e os limiares de aceitação são gerados de forma vetorizada antes do laço, o que é matematicamente equivalente à geração iterativa e reduz o tempo de execução.

```
mh_alelo ← function(n_iter, nAA, nAa, naa,
                    p0 = 0.5, sigma = 0.05) {
  p ← numeric(n_iter)
  p[1] ← p0
  a ← 2*nAA + nAa                  # expoentes da likelihood
  b ← 2*naa + nAa

  passos ← rnorm(n_iter, 0, sigma) # incrementos da proposta
  logu   ← log(runif(n_iter))      # limiares de aceitacao
  aceitos ← 0L

```

```
  for (t in 2:n_iter) {
    ant  <- p[t-1]
    prop <- ant + passos[t]
    if (prop <= 0 || prop >= 1) { p[t] <- ant; next }

    logA <- a*log(prop) + b*log(1 - prop) -
            (a*log(ant) + b*log(1 - ant))

    if (logu[t] < logA) { p[t] <- prop; aceitos <- aceitos + 1L }
    else                  p[t] <- ant
  }
  attr(p, "aceitacao") <- round(100 * aceitos / (n_iter - 1), 1)
  p
}
```

A.3 Implementação do SeqTest

A função `Binario` transforma um bloco de $k(w+1)$ valores da cadeia em k variáveis binárias, conforme as Etapas 2 e 3 descritas na Seção 2.5.

```
Binario <- function(x, phi = PHI, k = K, w = W) {
  y <- rep(0, k)
  for (i in 1:k) {
    ini <- k + (i - 1)*w + 1        # janela de comparacao
    fim <- k + (i - 1)*w + w
    V <- (sum(x[i] >= x[ini:fim]) + 1) / (w + 1)   # rank
    y[i] <- min(V, 1 - V)                          # simetrizacao
  }
  1 * (y <= phi / 2)                               # binarizacao
}
```

A função `obter_cv` configura o teste sequencial e recolhe o vetor completo de limiares. Como os limiares não dependem dos dados observados, alimenta-se o pacote com uma sequência sintética de contagens abaixo da fronteira de H_0, que jamais produz sinalização, o que permite percorrer os M testes até o fim. O vetor obtido é reutilizado em todas as réplicas.

```
obter_cv <- function(N_pacote) {
  pp <- 1 - (PHI + CC)          # 0,44 : p0 sob H0
  z  <- (1 - pp) / pp           # 1,2727

  nome  <- paste0("cv_", as.integer(runif(1) * 1e6))
  dir_t <- file.path(tempdir(), nome)
  dir.create(dir_t, recursive = TRUE, showWarnings = FALSE)

  AnalyzeSetUp.Binomial(name = nome, N = N_pacote, alpha = ALPHA,
```

```
                          zp = z, M = M_MIN,
                          AlphaSpendType = "power-type", rho = RHO,
                          title = nome, address = dir_t)

  cv <- rep(NA_real_, M_TOTAL)
  for (j in 1:M_TOTAL) {
    res <- Analyze.Binomial(name = nome, test = j, z = z,
                            cases = 8, controls = 12)
    col <- grep("^CV$|Critical", colnames(res), value = TRUE)[1]
    if (!is.na(col)) cv[j] <- as.numeric(res[j, col])
  }
  cv
}
```

As funções `calcula_Sj` e `ponto_sinal` percorrem a cadeia bloco a bloco, acumulam o número de zeros conforme a Equação (2.32) e localizam o primeiro teste em que a estatística atinge o limiar.

```
calcula_Sj <- function(cadeia) {
  S <- numeric(M_TOTAL); acc <- 0
  for (j in 1:M_TOTAL) {
    b <- cadeia[((j-1)*BLOCO + 1):(j*BLOCO)]
    acc <- acc + (K - sum(Binario(b)))   # zeros acumulados
    S[j] <- acc
  }
  S
}

ponto_sinal <- function(S, cv) {
  i <- which(S >= cv)
  if (!length(i)) NA_integer_ else min(i)
}
```

A.4 Diagnósticos Clássicos

A função `diagnosticos_classicos` aplica os três diagnósticos do pacote `coda` à mesma cadeia e registra ainda a média *a posteriori*, o intervalo de credibilidade e a taxa de aceitação, quando disponível.

```
diagnosticos_classicos <- function(cadeia) {
  mc <- as.mcmc(as.numeric(cadeia))

  p_g <- tryCatch({ z <- geweke.diag(mc)$z; 2*pnorm(-abs(z)) },
                  error = function(e) NA_real_)
  hw  <- tryCatch(heidel.diag(mc)[1, "stest"],
                  error = function(e) NA_real_)
```

```
  rl  <- tryCatch(raftery.diag(mc)$resmatrix[1, "N"],
                  error = function(e) NA_real_)

  acc <- attr(cadeia, "aceitacao")
  if (is.null(acc)) acc <- NA_real_
  ic <- as.numeric(quantile(cadeia, c(0.025, 0.975)))

  list(Geweke_p = round(p_g, 4),
       Geweke   = if (is.na(p_g)) NA
                  else if (p_g > 0.05) "Conv." else "Nao conv.",
       HW       = if (is.na(hw)) NA
                  else if (hw == 1) "Conv." else "Nao conv.",
       Raftery = rl,
       Media    = round(mean(cadeia), 4),
       IC_inf   = round(ic[1], 4),
       IC_sup   = round(ic[2], 4),
       Aceit    = acc)
}
```

A.5 Execução do Experimento

A execução inicia pelo sorteio da semente-mestra a partir do relógio do sistema, registrada em log para permitir a reprodução integral do experimento, e pelo cálculo do vetor de limiares.

```
seed_mestra <- as.integer(Sys.time()) %% .Machine$integer.max
set.seed(seed_mestra)

cv <- obter_cv(N_EVENT)      # calculado uma unica vez
```

Aplicam-se em seguida os dois controles de validação, cada um com 50 réplicas: uma amostra independente extraída exatamente da *posteriori* da aplicação de frequência alélica, e um processo autorregressivo de coeficiente 0,99.

```
# controle positivo: i.i.d. exata da posteriori Beta(122,80)
pos <- sapply(1:R_REP, function(r)
  ponto_sinal(calcula_Sj(rbeta(N_CADEIA, 122, 80)), cv))

# controle negativo: AR(1) com coeficiente 0,99
neg <- sapply(1:R_REP, function(r) {
  e <- rnorm(N_CADEIA); x <- numeric(N_CADEIA)
  for (i in 2:N_CADEIA) x[i] <- 0.99*x[i-1] + e[i]
  ponto_sinal(calcula_Sj(x), cv)
})

piso_iid <- median(pos, na.rm = TRUE)   # piso empirico
```

Os sete cenários são declarados em uma lista, na qual cada elemento contém a função geradora da cadeia e o valor teórico da média *a posteriori*, calculado pela Equação (2.39) nas aplicações de Gibbs.

```
cenarios ← list(
  list(tag = "spam_dificil",    app = "SPAM",
       sim = function() gibbs_prevalencia(N_CADEIA, 0.75, 0.80,
                                          n = 500,   r = 200)),
  list(tag = "spam_favoravel", app = "SPAM",
       sim = function() gibbs_prevalencia(N_CADEIA, 0.99, 0.95,
                                          n = 500,   r = 200)),
  list(tag = "fraude_legado",  app = "Fraude",
       sim = function() gibbs_prevalencia(N_CADEIA, 0.80, 0.95,
                                          n = 10000, r = 800)),
  list(tag = "fraude_moderno", app = "Fraude",
       sim = function() gibbs_prevalencia(N_CADEIA, 0.95, 0.995,
                                          n = 10000, r = 200)),
  list(tag = "hwe_lenta",      app = "HWE",
       sim = function() mh_alelo(N_CADEIA, 50, 21, 29, sigma = 0.005)),
  list(tag = "hwe_boa",        app = "HWE",
       sim = function() mh_alelo(N_CADEIA, 50, 21, 29, sigma = 0.05)),
  list(tag = "hwe_ruim",       app = "HWE",
       sim = function() mh_alelo(N_CADEIA, 50, 21, 29, sigma = 0.5)))
```

O laço principal percorre os sete cenários e, dentro de cada um, as 50 réplicas. Para cada cadeia gerada, aplicam-se o SeqTest e os três diagnósticos clássicos, e os resultados são acumulados em uma tabela.

```
det ← list()

for (i in seq_along(cenarios)) {
  cn ← cenarios[[i]]

  for (r in 1:R_REP) {
    cad ← cn$sim()                    # gera a cadeia
    S   ← calcula_Sj(cad)             # estatistica acumulada
    js  ← ponto_sinal(S, cv)          # teste da sinalizacao (ou NA)
    dg  ← diagnosticos_classicos(cad) # Geweke, H-W, Raftery-Lewis

    det[[length(det) + 1]] ← data.frame(
      Aplicacao     = cn$app,
      Tag           = cn$tag,
      Replica       = r,
      SeqTest       = if (is.na(js)) "Nao conv." else "Conv.",
      SeqTest_teste = js,
      Sj_final      = S[M_TOTAL],
      cv_final      = round(cv[M_TOTAL], 1),
      Geweke_p      = dg$Geweke_p,
```

```
      Geweke         = dg$Geweke,
      HeidelWelch    = dg$HW,
      Raftery_N      = dg$Raftery,
      Media          = dg$Media,
      Aceitacao      = dg$Aceit,
      stringsAsFactors = FALSE)

    rm(cad); gc(verbose = FALSE)
  }
}

detalhe <- do.call(rbind, det)   # 350 linhas
```

A consolidação por cenário calcula, a partir dessa tabela, a proporção de réplicas em que cada método declarou convergência, a mediana e a amplitude do teste de sinalização, e a mediana das demais estatísticas. Esses são os valores reportados nas tabelas do Capítulo 4.

```
resumo <- do.call(rbind, lapply(split(detalhe, detalhe$Tag),
  function(d) data.frame(
    Aplicacao      = d$Aplicacao[1],
    Tag            = d$Tag[1],
    Replicas       = nrow(d),
    SeqTest_conv   = sum(d$SeqTest == "Conv."),
    teste_mediano  = median(d$SeqTest_teste, na.rm = TRUE),
    teste_min      = min(d$SeqTest_teste, na.rm = TRUE),
    teste_max      = max(d$SeqTest_teste, na.rm = TRUE),
    Geweke_conv    = sum(d$Geweke      == "Conv.", na.rm = TRUE),
    HW_conv        = sum(d$HeidelWelch == "Conv.", na.rm = TRUE),
    Raftery_N_med  = round(median(d$Raftery_N, na.rm = TRUE)),
    Media_med      = round(median(d$Media, na.rm = TRUE), 4),
    Aceitacao_med  = round(median(d$Aceitacao, na.rm = TRUE), 1),
    stringsAsFactors = FALSE)))
```

ANEXO A – Declaração de uso de IAGen no processo de escrita científica

Durante a preparação deste documento, eu, Gabriel Ferreira, estudante de graduação do curso de Engenharia de Controle e Automação, declaro o uso da ferramenta Claude (Anthropic), baseada em modelos de linguagem de grande escala (LLM), para apoio à revisão textual, reescrita acadêmica, organização estrutural de capítulos, ajuste de coerência entre seções, tradução de trechos para o inglês e refinamento de resumos, objetivos e conclusões.

Após o uso desta ferramenta/modelo/serviço, a pessoa autora revisou e editou o conteúdo em conformidade com os princípios éticos para uso de IAGen e com os acordos estabelecidos com a pessoa orientadora da pesquisa. Dessa forma, assumo total responsabilidade pelo conteúdo da publicação.

Etapas em que a IAGen foi utilizada e exemplos de prompts adotados:

- Apoio à formulação, revisão e reescrita de elementos textuais da dissertação, incluindo resumo, palavras-chave e organização do texto, com foco em clareza, concisão e fluidez de leitura. Prompts representativos: “reescreva este parágrafo para esclarecer melhor o referencial dos valores de σ no algoritmo Metropolis-Hastings”; “revise o resumo e as palavras-chave”.

- Organização estrutural de capítulos e seções, incluindo o ajuste de transições e a formatação de tabelas e listas de elementos pré-textuais. Prompts representativos: “com base nos meus objetivos, o que eu ainda não contextualizei nos resultados”; “ajude a estruturar uma tabela comparativa entre os diagnósticos de convergência”.

- Apoio à sintaxe do código LaTeX e à adequação do texto às normas ABNT, com auxílio na verificação e formatação de citações e referências bibliográficas de acordo com o padrão normativo. Prompt representativo: “como citar este autor de forma indireta neste parágrafo?”.

- Apoio à leitura, interpretação e aproveitamento de trechos de artigos, livros e demais referências. Prompts representativos: “o que o autor quis dizer exatamente nesta parte?”; “me explique melhor como o autor chegou a este resultado”.