

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

**Inferência Bayesiana em modelos de volatilidade estocástica
na média utilizando o método de Monte Carlo Hamiltoniano
em variedade Riemanniana**

Bruno Estanislau Holtz

Dissertação de Mestrado do Programa Interinstitucional de
Pós-Graduação em Estatística (PIPGes)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Bruno Estanislau Holtz

Inferência Bayesiana em modelos de volatilidade
estocástica na média utilizando o método de Monte Carlo
Hamiltoniano em variedade Riemanniana

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

Área de Concentração: Estatística

Orientador: Prof. Dr. Ricardo Sandes Ehlers

USP – São Carlos
Março de 2024

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

E79i Estanislau Holtz, Bruno
Inferência Bayesiana em modelos de volatilidade
estocástica na média utilizando o método de Monte
Carlo Hamiltoniano em variedade Riemanniana / Bruno
Estanislau Holtz; orientador Ricardo Sandes
Ehlers. -- São Carlos, 2024.
97 p.

Dissertação (Mestrado - Programa
Interinstitucional de Pós-graduação em Estatística) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2024.

1. Séries Financeiras. 2. Modelo de Volatilidade
Estocástica na Média. 3. Monte Carlo Hamiltoniano.
4. Monte Carlo Hamiltoniano em Variedade
Riemanniana. 5. Distribuição Mistura de Escala
Normal. I. Sandes Ehlers, Ricardo, orient. II.
Título.

Bruno Estanislau Holtz

Bayesian inference in stochastic volatility in mean model
using Riemannian manifold Hamiltonian Monte Carlo
method

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

Concentration Area: Statistics

Advisor: Prof. Dr. Ricardo Sandes Ehlers

USP – São Carlos
March 2024



UNIVERSIDADE FEDERAL DE SÃO CARLOS

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

Folha de Aprovação

Defesa de Dissertação de Mestrado do candidato Bruno Estanislau Holtz, realizada em 21/02/2024.

Comissão Julgadora:

Prof. Dr. Ricardo Sandes Ehlers (USP)

Prof. Dr. Marcio Poletti Laurini (USP)

Prof. Dr. Carlos Antonio Abanto-Valle (UFRJ)

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

AGRADECIMENTOS

Gostaria de expressar meus agradecimentos a minha amiga e companheira Camila. Seu carinho e companheirismo foram a luz que iluminou o caminho percorrido neste trajeto.

Agradecer aos meus amigos de república, pela amizade, compreensão e apoio emocional. Nosso tempo compartilhado não apenas trouxe momentos de descontração, mas também serviu como um refúgio necessário durante os desafios acadêmicos.

Ao meu orientador, Professor Ricardo Sandes Ehlers, pela orientação, paciência e encorajamento ao longo deste processo. Sua disponibilidade para discussões e esclarecimentos foram cruciais para o desenvolvimento deste trabalho.

Ao Professor Carlos Antonio Abanto Valle, pelo generoso compartilhamento de seus códigos computacionais. Sua colaboração foi essencial para a realização deste estudo.

Ao corpo Docente do departamento de estatística da USP e UFSCAr, por todo conhecimento compartilhado durante o curso.

Aos funcionários do ICMC e PIPGEs, cuja dedicação e eficiência facilitaram as etapas burocráticas e administrativas, proporcionando um ambiente propício para a pesquisa e aprendizado.

Aos meus colegas de laboratório, cuja presença e colaboração tornaram o ambiente de pesquisa enriquecedor e estimulante. As discussões e trocas de experiências contribuíram significativamente para o meu crescimento acadêmico e profissional.

Aos responsáveis pelo funcionamento do cluster de alto desempenho EULER do projeto de pesquisa Centro de Ciências Matemáticas Aplicadas à Indústria (CeMEAI), financiados pela FAPESP (proc. 2013/07375-0), pela disponibilidade e suporte técnico, que possibilitaram a realização das simulações e análises computacionais necessárias para este trabalho.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001

RESUMO

HOLTZ, B. E. **Inferência Bayesiana em modelos de volatilidade estocástica na média utilizando o método de Monte Carlo Hamiltoniano em variedade Riemanniana.** 2024. 97 p. Dissertação (Mestrado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2024.

Este trabalho considera o modelo de volatilidade estocástica na média, no qual a distribuição condicional dos dados pertence a família mistura de escala normal para modelagem de séries financeiras. Esta classe de modelos é mais robusta por acomodar erros com caudas mais pesadas que a distribuição normal, visto que esta é uma característica marcante de séries financeiras. Para a estimativa dos parâmetros, propomos um algoritmo Bayesiano via cadeias de Markov, utilizando o método Monte Carlo Hamiltoniano (HMC) e sua variante, o método Monte Carlo Hamiltoniano em Variedade Riemanniana (RMHMC). O algoritmo foi implementado utilizando as bibliotecas `Rcpp` e `RcppArmadillo` disponíveis na linguagem R. Os critérios de informação recentemente desenvolvidos, *Watanabe Akaike Information Criterion* (WAIC) e *Leave-One-Out Cross-Validation* (LOO-CV) foram calculados para comparar o ajuste dos modelos, bem como o *Deviance Information Criterion* (DIC). Estudos de simulação foram realizados para ilustrar e avaliar o desempenho do método proposto. Por fim, realizamos aplicações a dados reais, fornecendo evidências empíricas de sua efetividade.

Palavras-chave: Séries Financeiras, Modelo de Volatilidade Estocástica na Média, Monte Carlo Hamiltoniano, Monte Carlo Hamiltoniano em Variedade Riemanniana, Distribuição Mistura de Escala Normal.

ABSTRACT

HOLTZ, B. E. **Bayesian inference in stochastic volatility in mean model using Riemannian manifold Hamiltonian Monte Carlo method**. 2024. 97 p. Dissertação (Mestrado em Estatística – Programa Interinstitucional de Pós-Graduação em Estatística) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2024.

This paper considers the stochastic volatility in mean model, where the conditional distribution of the data belongs to the mixed-scale normal family for modeling financial time series. This model class is more robust in accommodating errors with heavier tails than the normal distribution, a characteristic often observed in financial data. Parameter estimation is conducted through a Bayesian algorithm employing Markov Chain methods, specifically the Hamiltonian Monte Carlo (HMC) method and its variant, the Riemannian Manifold Hamiltonian Monte Carlo (RMHMC) method. The algorithm is implemented using the Rcpp and RcppArmadillo libraries in the R language. Recently developed information criteria, namely the Watanabe Akaike Information Criterion (WAIC) and leave-one-out cross-validation (LOO-CV), along with the deviance information criterion (DIC), are calculated to compare the model fits. Simulation studies are conducted to illustrate and evaluate the performance of the proposed method. Finally, we apply the developed methodology to real return series, providing empirical evidence of its effectiveness.

Keywords: Financial Series, Stochastic Volatility in Mean Model, Hamiltonian Monte Carlo, Riemannian Manifold Hamiltonian Monte Carlo, Scale Mixture of Normal.

LISTA DE ILUSTRAÇÕES

Figura 1	– Resultados estimados para os modelos SVM-t, SVM-S e SVM-VG. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.	59
Figura 2	– Resultados das volatilidades estimadas para SVM-S. A linha vermelha representa o verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.	60
Figura 3	– Resultados das volatilidades estimadas para SVM-t. A linha vermelha representa o verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.	61
Figura 4	– Resultados das volatilidades estimadas para SVM-VG. A linha vermelha representa o verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.	61
Figura 5	– Retornos compostos do índice IBOVESPA de 5 de janeiro de 1998 a 3 de outubro de 2005. O painel da esquerda mostra a série dos retorno, enquanto o painel da direita traz o histograma da série.	65
Figura 6	– Taxas de câmbio £/USD de 1 de janeiro de 2004 a 2 de junho de 2016. O painel da esquerda mostra a série das taxas, enquanto o painel da direita traz o histograma da série.	66
Figura 7	– Resultados estimados para os modelos SVM-N e SVM-t para os índices IBOVESPA. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.	69
Figura 8	– Resultados estimados para os modelos SVM-S e SVM-VG para os índices IBOVESPA. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.	70
Figura 9	– Resultados estimados para os modelos SVM-N e SVM-t para os retornos £/USD. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas a posteriori.	71

Figura 10 – Resultados estimados para os modelos SVM-S e SVM-VG para os retornos £/USD. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas a posteriori.	72
Figura 11 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (área sombreada representa a região com 95% de credibilidade) para SVM-N dos retornos IBOVESPA.	73
Figura 12 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-t dos retornos IBOVESPA.	74
Figura 13 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-S dos retornos IBOVESPA.	74
Figura 14 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-VG dos retornos IBOVESPA.	74
Figura 15 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (área sombreada representa a região com 95% de credibilidade) para SVM-N dos retornos £/USD.	75
Figura 16 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-t dos retornos £/USD.	75
Figura 17 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-S dos retornos £/USD.	75
Figura 18 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-VG dos retornos £/USD.	76
Figura 19 – Comparação das estimativas da variável de mistura λ_t para os modelos SVM-t, SVM-S e SVM-VG dos retornos IBOVESPA.	76
Figura 20 – Comparação das estimativas da variável de mistura λ_t para os modelos SVM-t, SVM-S e SVM-VG dos retornos £/USD.	76

LISTA DE ALGORITMOS

Algoritmo 1 – Metropolis-Hastings	31
Algoritmo 2 – Amostrador de Gibbs	32
Algoritmo 3 – Monte Carlo Hamiltoniano (HMC)	40
Algoritmo 4 – Monte Carlo Hamiltoniano em Variedade Riemanniana (RMHMC)	44
Algoritmo 5 – Amostrador SMN-SMN	47
Algoritmo 6 – svm_n.cpp	89
Algoritmo 7 – svm_t.cpp	91
Algoritmo 8 – svm_s.cpp	93
Algoritmo 9 – svm_vg.cpp	95

LISTA DE TABELAS

Tabela 1	– Resultados da estimação dos modelos SVM-t, SVM-S e SVM-VG. Foram reportados a média a posteriori, o intervalo de 95% de credibilidade, o fator IF e a estatística DC.	58
Tabela 2	– $Viés_{rel}$ e $REQM_{rel}$ das estimativas $\hat{\theta}$ dos modelos SVM-SMN, com $\phi = 0.95$ em que $T \in \{1000, 1500, 3000\}$	63
Tabela 3	– $Viés_{rel}$ e $REQM_{rel}$ das estimativas $\hat{\theta}$ dos modelos SVM-SMN, com $\phi = 0.98$ em que $T \in \{1000, 1500, 3000\}$	64
Tabela 4	– Estatísticas resumo para as série de retornos analisadas.	65
Tabela 5	– Resultados estimados para os retornos IBOVESPA. Primeira linha : Média a posteriori. Segunda linha: Intervalos de 95% de credibilidade. Terceira linha: Estatística de Geweke (DC). Quarta linha: Fator de ineficiência.	67
Tabela 6	– Resultados estimados para os retornos £/USD. Primeira linha : Média a posteriori. Segunda linha: Intervalos de 95% de credibilidade. Terceira linha: Estatística de Geweke (DC). Quarta linha: Fator de ineficiência.	68
Tabela 7	– Critérios de informação calculados.	77

SUMÁRIO

1	INTRODUÇÃO	21
2	METODOLOGIA	25
2.1	Séries de retornos	25
2.2	Família Mistura de Escala Normal	26
2.3	Abordagem Bayesiana	27
2.4	Método de Monte Carlo via Cadeias de Markov	28
2.4.1	<i>Cadeias de Markov</i>	28
2.4.2	<i>Algoritmo de Metropolis-Hastings</i>	30
2.4.3	<i>Amostrador de Gibbs</i>	31
2.5	Análise de Convergência	32
2.6	CrITÉRIOS para Seleção de Modelos	33
2.6.1	<i>Deviance Information Criterion</i>	33
2.6.2	<i>Watanabe Information Criterion</i>	34
2.6.3	<i>Leave-One-Out Cross-Validation</i>	35
3	MONTE CARLO HAMILTONIANO	37
3.1	Dinâmica Hamiltoniana	38
3.2	O método Monte Carlo Hamiltoniano	39
3.3	O método Monte Carlo Hamiltoniano em Variedade Riemanniana	41
4	MODELO DE VOLATILIDADE	45
4.1	Modelo de Volatilidade Estocástica na Média na Família Mistura de Escala Normal	46
4.2	Estimação dos parâmetros	46
4.3	Distribuições condicionais completas	48
5	ESTUDOS DE SIMULAÇÃO E APLICAÇÕES	57
5.1	Estudo de simulação 1	57
5.2	Estudo de simulação 2	62
5.3	Aplicação a dados reais	65
6	CONCLUSÕES	79

REFERÊNCIAS	81
APÊNDICE A DEMONSTRAÇÕES	85
APÊNDICE B CÓDIGOS COMPUTACIONAIS	89

INTRODUÇÃO

A análise de séries temporais compreende a investigação de fenômenos que se desenvolvem ao longo do tempo, onde as observações sequenciais apresentam interdependência. A importância dessa análise nas ciências aplicadas é destacada em várias áreas. Na epidemiologia, por exemplo, é frequente a modelagem do número de casos de uma doença em um determinado período. Nas ciências sociais, o acompanhamento das taxas de natalidade ou das intenções de voto durante períodos eleitorais desperta interesse. Na medicina, a mensuração da pressão arterial ao longo do tempo pode ser utilizada para avaliar a eficácia de um novo medicamento. No âmbito financeiro, é essencial quantificar os riscos de perdas ou ganhos, uma vez que grandes investimentos impactam a economia de um país. O enfoque deste trabalho concentrou-se em séries de retornos financeiros.

Na modelagem de fenômenos, é comum empregar modelos que capturem características essenciais para garantir resultados confiáveis. Ao analisar séries financeiras, modelos que assumem constância na variância dos dados (homoscedasticidade) frequentemente revelam-se inadequados (MORETTIN; TOLOI, 2006). Além disso, o enfoque nesse tipo de análise muitas vezes não se restringe apenas à previsão de taxas de retorno, mas também visa estimar a volatilidade condicional (volatilidade) ao longo do tempo (EHLERS, 2007). Para atender a esse propósito, destacam-se na literatura financeira modelos como o ARCH (*Autoregressive Conditional Heteroscedasticity*) (ENGLE, 1982), o GARCH (*Generalized ARCH*) (BOLLERS-LEV, 1986) e os modelos de Volatilidade Estocástica (em inglês *Stochastic Volatility - SV*), introduzidos por Taylor (1986).

Os modelos ARCH abordam a modelagem da volatilidade ao assumir que ela depende de maneira determinística dos quadrados dos erros passados da série. Já os modelos GARCH representam uma generalização dos modelos ARCH, permitindo que a volatilidade dependa tanto de uma combinação linear das volatilidades anteriores quanto do quadrado das observações passadas. Apesar de aparentarem maior complexidade, os modelos GARCH geralmente são

mais parcimoniosos do que os modelos ARCH, visto que introduzem menos parâmetros a serem estimados. No entanto, a identificação do modelo GARCH adequado para uma situação real não é uma tarefa trivial (MORETTIN, 2011).

A classe de modelos SV, por outro lado, considera que a volatilidade é influenciada por um processo estocástico. Isso decorre da ideia de que é sensato permitir que a volatilidade seja afetada por eventos inesperados, contrastando, portanto, com a abordagem determinística dos modelos GARCH. Além disso, o modelo SV surge como uma discretização aproximada de vários processos de difusão contínua ao longo do tempo. Na literatura financeira, é considerado uma generalização do modelo de Black-Scholes, permitindo a inclusão de efeitos de clusterização nos retornos (PITT; MALIK; DOUCET, 2014).

Apesar dos benefícios fornecidos pelos modelos SV, sua aplicação empírica tem sido limitada pela dificuldade na estimação dos parâmetros. Esta dificuldade é gerada, principalmente, pela difícil tarefa de calcular sua função de verossimilhança. Na literatura, diversos métodos foram implementados para lidar com esse desafio, incluindo o método dos momentos generalizado (MELINO; TURNBULL, 1990), o método de quasi-verossimilhança (RUIZ, 1994) e os métodos de Monte Carlo via Cadeia de Markov (em inglês, *Monte Carlo Markov Chains - MCMC*) (SHEPHARD, 1993; JACQUIER; POLSON; ROSSI, 1994). A principal vantagem dos métodos MCMC é que eles permitem realizar inferências simultâneas sobre os parâmetros e as volatilidades suavizadas (BROTO; RUIZ, 2004).

Recentemente, a relação entre retornos e volatilidade esperada tem sido explorada na literatura financeira. Teoricamente, há uma relação positiva entre retornos esperados e volatilidade, se os investidores forem avessos ao risco. Ou seja, investidores conservadores esperam um prêmio maior por títulos mais arriscados. Embora existam evidências de uma relação negativa entre retorno esperado e volatilidade, French, Schwert e Stambaugh (1987) interpretam essas evidências como indícios indiretos da relação positiva. Estes autores argumentam que choques imprevisíveis no processo de retorno, causados por boas ou más notícias, levam a maior volatilidade esperada. Se volatilidade e retornos esperados estiverem positivamente relacionados, e os fluxos de valores não forem afetados, o preço atual do índice de ações deve cair. O oposto acontece com pequenos choques, levando a um aumento recente no índice de ações. Essa teoria é conhecida como *feedback* de volatilidade.

Ao modelar séries financeiras utilizando o modelo SV, usualmente os resultados exigem um intensivo pré-processamento dos dados, a fim de evitar o problema de estimar a média e a variância simultaneamente. Para lidar com este problema, Koopman e Uspensky (2002) introduzem os modelos de Volatilidade Estocástica na Média (em inglês *Stochastic Volatility in Mean - SVM*) permitindo que a volatilidade não observável influencie a equação da média como uma variável explicativa. Eles utilizam o método de máxima verossimilhança simulada para estimação dos parâmetros e fornecem uma evidência empírica que o coeficiente da volatilidade na média captura informação sobre o efeito *feedback*. Abanto-Valle, Rodríguez e Garrafa-

Aragón (2021) desenvolvem um algoritmo utilizando o método Monte Carlo Hamiltoniano (em inglês *Hamiltonian Monte Carlo* - HMC) e sua variação em Variedade Riemanniana (em inglês *Riemannian Manifold Hamiltonian Monte Carlo* - RMHMC) para inferência dos parâmetros do modelo SVM, seguindo a abordagem do amostrador de Gibbs. Este algoritmo permite que os parâmetros da média e os da volatilidade sejam atualizados em blocos, para que então, as log-volatilidades sejam atualizadas de uma única vez.

Na formulação original dos modelos SV, assume-se que os erros observacionais seguem uma distribuição normal padrão. No entanto, diversos estudos empíricos indicam que esta formulação não é realista ao considerar a curtose observada nos retornos em inúmeras aplicações empíricas. Deste modo, Abanto-Valle, Migon e Lachos (2012) estendem o modelo de Koopman e Uspensky (2002) permitindo que os erros observacionais pertençam à família Mistura de Escala da Normal (em inglês *Scale Mixture of Normal* - SMN). Este modelo é referido como SVM-SMN.

Devido à natureza não linear das volatilidades no modelo SVM, a estimação dos parâmetros se torna uma tarefa desafiadora. Nesta dissertação, propõe-se um algoritmo computacionalmente intensivo do tipo MCMC, mais especificamente o método HMC e sua variação RMHMC, para o modelo SVM-SMN. Esta classe de modelos abrange diversos fatos estilizados presentes em séries de retornos, tais como conglomerados de volatilidade, caudas pesadas e o efeito *feedback*. A implementação dos códigos computacionais foi realizada utilizando pacotes que simplificam a integração entre as linguagens R e C++, permitindo unir a eficiência e flexibilidade da linguagem C++ ao mesmo tempo que desfrutamos do ambiente estatístico e analítico oferecido pelo R.

O restante desta dissertação está organizada da seguinte forma: O Capítulo 2 introduz a Família Mistura de Escala Normal e as séries de retornos. Ainda neste capítulo, são descritos de forma geral os conceitos fundamentais de inferência Bayesiana, métodos de simulação de MCMC e critérios para seleção de modelos. No Capítulo 3, apresenta-se o método de HMC e sua variação em Variedade Riemanniana. No Capítulo 4, é introduzido a classe de Modelos de Volatilidade de Média com erros pertencentes a família de Mistura de Escala da Normal, referida como SVM-SMN. No Capítulo 5, são apresentados os estudos de simulação para avaliar o método, bem como aplicações a dados reais. Finalmente, o Capítulo 6 apresenta conclusões e sugestões para trabalhos futuros.

METODOLOGIA

O propósito deste capítulo é oferecer uma visão abrangente dos princípios metodológicos antes de entrar nos detalhes específicos dos métodos utilizados. Para atingir os objetivos delineados, torna-se fundamental estabelecer um contexto metodológico amplo, proporcionando compreensão dos fundamentos e dos procedimentos gerais adotados nesta dissertação.

2.1 Séries de retornos

Em finanças, frequentemente deseja-se avaliar o risco de uma carteira de ativos financeiros utilizando a variação dos preços. Ou seja, o risco é calculado em termos da variação $\Delta P_t = P_t - P_{t-1}$, em que P_t é o preço do ativo no tempo t , usualmente um dia de negócio. A variação relativa dos preços

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}} = \frac{\Delta P_t}{P_{t-1}},$$

é dita ser o **retorno líquido simples** dos ativos no tempo t , enquanto

$$r_t = \ln\left(\frac{P_t}{P_{t-1}}\right) = \ln(1 + R_t) \quad (2.1)$$

é conhecido como **retorno composto continuamente**, em que $1 + R_t$ é o chamado **retorno bruto simples**.

Como podemos notar, existem várias definições de retorno. A definição (2.1) é frequentemente utilizada e referida simplesmente como "retorno". Segundo [Morettin \(2011\)](#), é aconselhável trabalhar com séries de retornos em vez de séries de preços, já que as primeiras são livres de escala e apresentam características estatísticas interessantes, como estacionariedade.

Os principais traços empíricos das séries de retornos, conhecidos como **fatos estilizados**, podem ser enumerados da seguinte forma:

1. retornos são geralmente não auto-correlacionados;
2. os quadrados dos retornos são autocorrelacionados, apresentando uma correlação de *lag* inicial pequena e depois uma queda lenta nas demais;
3. as volatilidades se apresentam agrupadas (*clusters*) ao longo do tempo;
4. a distribuição (incondicional) dos retornos apresenta caudas mais pesadas do que uma distribuição normal; além disso, a distribuição, embora aproximadamente simétrica, é em geral leptocúrtica;
5. existem séries não-lineares, no sentido que reagem diferentemente a choques grandes ou pequenos, ou ainda, a choques positivos ou negativos.

Maiores detalhes sobre séries de retornos podem ser encontrados em [Tsay \(2010\)](#) e [Morettin \(2011\)](#).

2.2 Família Mistura de Escala Normal

A classe de distribuições Mistura de Escala da Normal tem sido usada em diversas aplicações estatísticas para inferência robusta, inclusive no contexto de séries financeiras. [Andrews e Mallows \(1974\)](#) mostram que a variável aleatória contínua $Y|\lambda$ pertence a família mistura de escala normal se ela puder ser escrita da forma

$$Y = \mu + k(\lambda)^{1/2}Z,$$

em que μ é o parâmetro de localização, $Z \sim N(0, \sigma^2)$ e $k(\lambda)$ é uma função positiva de λ . Aqui, λ é a variável de mistura e sua densidade é representada por $p(\lambda|v)$, que depende do parâmetro real $v > 0$. Escolhendo $k(\lambda) = 1/\lambda$, a densidade marginal de $Y|\lambda$ é expressa como

$$f(y|\mu, \sigma^2, v) = \int_0^\infty N(y|\mu, \lambda^{-1}\sigma^2)p(\lambda|v)d\lambda. \quad (2.2)$$

A equação (2.2) pode descrever uma ampla classe de distribuições contínuas e simétricas, desde que a densidade de mistura $p(\cdot|v)$ seja apropriada. Utilizaremos a notação $Y|\lambda \sim SMN(\mu, \sigma^2, \lambda)$ para referenciar variáveis pertencentes a esta família. Distribuições com caudas mais pesadas que a distribuição Normal podem ser consideradas, podendo incluir:

- *t-Student*: $Y \sim t_\nu(\mu, \sigma^2)$, com μ o parâmetro de locação, σ^2 o parâmetro de escala e $\nu > 0$ os graus de liberdade. Sua representação hierárquica é da forma:

$$Y|\mu, \sigma^2, \lambda \sim N(\mu, \lambda^{-1} \sigma^2), \quad \lambda|\nu \sim G(\nu/2, \nu/2),$$

em que $G(\cdot, \cdot)$ representa a distribuição Gama com densidade de probabilidade

$$p(\lambda|a, b) = \frac{b^a}{\Gamma(a)} \lambda^{a-1} \exp(-b\lambda), \quad a, b > 0.$$

- *Slash*: $Y \sim S(\mu, \sigma^2, \nu)$, com $\nu > 0$. Sua representação hierárquica é da forma:

$$Y|\mu, \sigma^2, \lambda \sim N(\mu, \lambda^{-1} \sigma^2), \quad \lambda|\nu \sim Be(\nu, 1),$$

em que $Be(\cdot, \cdot)$ representa a distribuição Beta.

- *Variância-Gama*: $Y \sim VG(\mu, \sigma^2, \nu)$, com $\nu > 0$. Sua representação hierárquica é da forma:

$$Y|\mu, \sigma^2, \lambda \sim N(\mu, \lambda^{-1} \sigma^2), \quad \lambda|\nu \sim IG(\nu/2, \nu/2),$$

em que $IG(\cdot, \cdot)$ representa a distribuição Gama Inversa (em inglês *Inverse Gamma* - IG). Esta distribuição também é conhecida como distribuição de Laplace Generalizada, distribuição de Bessel ou distribuição de McKay de tipo II (FISCHER; GAUNT; SARANTSEV, 2023).

As três distribuições mencionadas acima, englobam o caso normal quando $\nu \rightarrow \infty$. Notavelmente, tomando a distribuição degenerada $\lambda = 1$, recuperamos o caso em que Y segue uma distribuição Normal.

2.3 Abordagem Bayesiana

Em modelagem estatística, usualmente assumimos que os dados observados $(X_1, \dots, X_n) \in \Omega$ são provenientes de uma família de modelos estatísticos $p(x|\theta)$ parametrizada pelo vetor $\theta \in \Theta$ desconhecido e não observável. O conjunto Ω é dito ser o Espaço Amostral do experimento em estudo, enquanto o conjunto Θ é o Espaço Paramétrico do modelo adotado.

Antes de observarmos os dados, podemos considerar que a incerteza inicial acerca do vetor de interesse θ é representada probabilisticamente por sua distribuição a priori $p(\theta)$. Após observarmos $\mathbf{x} = (x_1, \dots, x_n)$, podemos atualizar nossa incerteza sobre θ dado \mathbf{x} , uma vez que

os dados contêm informação sobre $\boldsymbol{\theta}$. Formalmente, podemos calcular a posteriori $p(\boldsymbol{\theta}|\mathbf{x})$, que pode ser obtida via teorema de Bayes

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\Theta} p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}, \quad (2.3)$$

sendo $p(\mathbf{x}|\boldsymbol{\theta})$ a função de verossimilhança e $p(\mathbf{x})$ a distribuição preditiva de \mathbf{x} . Cabe notar que para \mathbf{x} fixado, a distribuição $p(\mathbf{x})$ atua como constante normalizadora. Assim, (2.3) pode ser reescrita como

$$p(\boldsymbol{\theta}|\mathbf{x}) \propto p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}),$$

uma vez que nosso interesse esta em $\boldsymbol{\theta}$.

Após a posteriori (2.3) ser computada, o procedimento de inferência sobre o vetor de parâmetros $\boldsymbol{\theta}$ pode ser realizado por meio do cálculo da integral

$$E_{\boldsymbol{\theta}|\mathbf{x}}[g(\boldsymbol{\theta})] = \int_{\Theta} g(\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{x})d\boldsymbol{\theta}, \quad (2.4)$$

para alguma função real $g : \Theta \rightarrow \mathbb{R}$.

2.4 Método de Monte Carlo via Cadeias de Markov

Em inúmeros problemas práticos, o cálculo da integral (2.4) é de complexa tratabilidade por não possuir solução analítica ou devido às elevadas dimensões que o vetor $\boldsymbol{\theta}$ pode assumir. Para lidar com tal dificuldade, o método de Monte Carlo via Cadeias de Markov (em inglês *Monte Carlo via Markov Chains* - MCMC), frequentemente é utilizado. Basicamente, a ideia do método MCMC consiste em construir uma cadeia de Markov capaz de simular amostras $\{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_S\}$ da distribuição de interesse $p(\boldsymbol{\theta}|\mathbf{x})$, e assim, aproximar a integral (2.4) por

$$\frac{1}{S} \sum_{i=1}^S g(\boldsymbol{\theta}_i) \xrightarrow{q.c.} E_{\boldsymbol{\theta}|\mathbf{x}}[g(\boldsymbol{\theta})],$$

segundo a Lei dos Grandes Números.

O restante desta seção é destinada a noções básicas sobre o método MCMC. Um tratamento mais aprofundado pode ser encontrado em [Robert e Casella \(2004\)](#) e [Gamerman e Lopes \(2006\)](#).

2.4.1 Cadeias de Markov

Definição 1 (Processo Estocástico). Um processo estocástico $\{X_t : t \in T\}$ é uma família de variáveis aleatórias indexadas por um conjunto T , podendo ser este discreto ou contínuo. Para

T discreto, o conjunto $S = \{s_0, s_1, \dots\}$ de todas as possíveis realizações de X_t , é chamado de **Espaço de Estados** do processo.

Neste presente trabalho, trataremos de processos estocásticos indexados por conjuntos enumeráveis, de modo que podemos assumir $T = \mathbb{N}$. Considere também que T representa instantes no tempo.

Definição 2 (Probabilidade de transição). Chama-se probabilidade de transição de um processo estocástico $\{X_t : t \in T\}$ a probabilidade do estado s_i ser atingido no instante t , dados os estados passados. Matematicamente,

$$p(X_t = s_i | X_{t-1} = s_j, \dots, X_0 = s_0).$$

Definição 3 (Cadeias de Markov). Seja $\{X_t : t \in T\}$ um processo estocástico. Dizemos que tal processo é uma Cadeia de Markov se sua probabilidade de transição pode ser escrita da forma

$$p(X_{t+1} = s_i | X_t = s_j, \dots, X_0 = s_0) = p(X_{t+1} = s_i | X_t = s_j),$$

para $\forall t \in T$. Ou seja, uma cadeia de Markov é um processo estocástico no qual sua probabilidade de transição em um instante t depende apenas do estado no tempo anterior.

Sob certas condições de regularidade, pode-se garantir que a distribuição de X_t converge para a distribuição estacionária (ou distribuição invariante), independente do valor inicial, X_0 , da cadeia. Para tanto, é necessário que a cadeia de Markov gerada seja, irredutível, reversível e aperiódica, que serão mencionadas a seguir.

A irredutibilidade exige que a probabilidade de transição da cadeia possa percorrer livremente todo o espaço de estado S . Isto é, $\forall s_i, s_j \in S, \exists t \in \mathbb{N}$ tal que

$$p(X_t = s_i | X_{t-1} = s_j) > 0.$$

A outra condição importante é a reversibilidade. Pode-se dizer que uma cadeia é reversível se $\forall s_i, s_j \in S$, temos que

$$p(s_i)p(X_{t+1} = s_i | X_t = s_j) = p(s_j)p(X_t = s_j | X_{t+1} = s_i), \quad \forall s_i, s_j \in S, \quad (2.5)$$

em que $p(s_i)$ é a distribuição estacionária do estado s_i . Em outras palavras, a equação (2.5) diz que se observarmos o processo no tempo reverso, a distribuição de probabilidade das transições entre os estados é a mesma que a distribuição no tempo original. Esta condição também é conhecida por *condição de balanço*.

Por fim, exige-se também que a cadeia seja aperiódica. Porém, primeiro precisamos definir o que é o período de um estado. Desta forma, considere $s \in S$ e defina o conjunto

$$\{n \in \mathbb{N} | p(X_{t+n} = s | X_t = s) > 0\}. \quad (2.6)$$

O maior divisor comum d_s do conjunto (2.6) é dito ser o período do estado s . Uma cadeia é aperiódica, quando todos os seus estados tem período unitário, ou seja, $d_s = 1, \forall s \in S$.

2.4.2 Algoritmo de Metropolis-Hastings

Suponha que desejamos realizar simulações do vetor aleatória $\boldsymbol{\theta}$ com distribuição de probabilidade $p(\boldsymbol{\theta})$ conhecida (a menos de uma constante) e que esta tarefa seja desafiadora. Suponha também que temos condições de gerar simulações de uma distribuição de probabilidade auxiliar $q(\cdot | \cdot)$ (distribuição proposta) mais facilmente. Neste caso podemos utilizar o método de Metropolis-Hasting (MH).

O algoritmo MH, que foi inicialmente proposto por [Metropolis *et al.* \(1953\)](#) e posteriormente estendido por [Hastings \(1970\)](#), consiste em gerar valores da distribuição proposta $q(\boldsymbol{\theta}^* | \boldsymbol{\theta})$, e assim, construir uma cadeia de Markov com distribuição invariante $p(\boldsymbol{\theta})$. [Chib e Greenberg \(1995\)](#) ilustram como o método satisfaz as condições de regularidade citadas na seção 2.4.1.

O algoritmo MH utiliza a probabilidade de transição $p_{MH}(\boldsymbol{\theta}, \boldsymbol{\theta}^*)$ definida por

$$p_{MH}(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = q(\boldsymbol{\theta}^* | \boldsymbol{\theta}) \alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*), \quad (2.7)$$

para propor movimentos do tipo $\boldsymbol{\theta} \rightarrow \boldsymbol{\theta}^*$. Estes movimentos são aceitos com probabilidade $\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*)$, em que

$$\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = \min \left\{ 1, \frac{p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta} | \boldsymbol{\theta}^*)}{p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta})} \right\}, \quad \forall \boldsymbol{\theta}, \boldsymbol{\theta}^* \in S.$$

Note que a cadeia gerada por (2.7) satisfaz a condição de balanço (2.5), uma vez que

$$\begin{aligned} p(\boldsymbol{\theta})p_{MH}(\boldsymbol{\theta}, \boldsymbol{\theta}^*) &= p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta})\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) \\ &= p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta}) \cdot \min \left\{ 1, \frac{p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta} | \boldsymbol{\theta}^*)}{p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta})} \right\} \\ &= \min \{ p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta}), p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta} | \boldsymbol{\theta}^*) \} \\ &= p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta} | \boldsymbol{\theta}^*) \cdot \min \left\{ 1, \frac{p(\boldsymbol{\theta})q(\boldsymbol{\theta}^* | \boldsymbol{\theta})}{p(\boldsymbol{\theta}^*)q(\boldsymbol{\theta} | \boldsymbol{\theta}^*)} \right\} \\ &= p(\boldsymbol{\theta}^*)p_{MH}(\boldsymbol{\theta}^*, \boldsymbol{\theta}). \end{aligned}$$

Desta forma, a cadeia gerada é reversível. Para que a cadeia seja irreduzível e aperiódica, basta que a função proposta $q(\cdot|\cdot)$ seja positiva no suporte de p (CHIB; GREENBERG, 1995).

O método MH é bastante genérico e pode ser implementado utilizando uma ampla classe de distribuições propostas $q(\cdot|\cdot)$. Contudo, essas vantagens não são isentas de custo em termos práticos. No capítulo 3 será introduzido uma variação do método MH capaz de mitigar algumas questões.

O algoritmo de MH pode ser descrito pelo Algoritmo 1.

Algoritmo 1 – Metropolis-Hastings

- 1: Tome $i = 0$ e inicialize $\boldsymbol{\theta}^{(i)}$
 - 2: Simule $\boldsymbol{\theta}^* \sim q(\boldsymbol{\theta}^* | \boldsymbol{\theta}^{(i-1)})$ e $u \sim U(0, 1)$
 - 3: **se** $u < \alpha(\boldsymbol{\theta}^{(i-1)}, \boldsymbol{\theta}^*)$ **então**
 - 4: $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^*$
 - 5: **senão**
 - 6: $\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)}$
 - 7: **fim se**
 - 8: Defina $i = i + 1$ e retorne ao passo 2 até atingir a convergência.
-

2.4.3 Amostrador de Gibbs

Um caso particular do método MH que é muito utilizado em simulações estocásticas é o chamado amostrador de Gibbs (em inglês *Gibbs Sampler* - GS). Este método foi introduzido por Geman e Geman (1984) no contexto de processamento de imagens e permaneceu desconhecido pela comunidade estatística até que Gelfand e Smith (1990) destacou seu potencial.

Suponha que realizar simulações de vetor aleatório $\boldsymbol{\theta} = (\theta_1, \dots, \theta_d)'$ com densidade de probabilidade conjunta $p(\boldsymbol{\theta})$ seja uma tarefa difícil, porém as condicionais completas sejam todas conhecidas e facilmente simuladas. Isto é, a distribuição univariada

$$p(\theta_i | \boldsymbol{\theta}_{-i}), \quad \text{em que } \boldsymbol{\theta}_{-i} = (\theta_1, \dots, \theta_{i-1}, \theta_{i+1}, \dots, \theta_d)',$$

tem forma conhecida para todo $i \in \{1, \dots, d\}$ e sabemos gerar simulações diretamente. Desta forma, podemos considerar o método MH tomando as condicionais completas $p(\theta_i | \boldsymbol{\theta}_{-i})$ como distribuição proposta para propor movimentos do tipo

$$\boldsymbol{\theta} = (\theta_1^{(t)}, \dots, \theta_i^{(t)}, \dots, \theta_d^{(t)}) \rightarrow \boldsymbol{\theta}^* = (\theta_1^{(t)}, \dots, \theta_i^{(t+1)}, \dots, \theta_d^{(t)}).$$

Neste caso, os movimentos serão aceitos com probabilidade $\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*)$, composta pela razão

$$\begin{aligned}
\frac{p(\boldsymbol{\theta}^*)p\left(\boldsymbol{\theta}_i^{(t)}|\boldsymbol{\theta}_{-i}^{(t)}\right)}{p(\boldsymbol{\theta})p\left(\boldsymbol{\theta}_i^{(t+1)}|\boldsymbol{\theta}_{-i}^{(t)}\right)} &= \frac{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t+1)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)p\left(\boldsymbol{\theta}_i^{(t)}|\boldsymbol{\theta}_{-i}^{(t)}\right)}{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)p\left(\boldsymbol{\theta}_i^{(t+1)}|\boldsymbol{\theta}_{-i}^{(t)}\right)} \\
&= \frac{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t+1)}, \dots, \boldsymbol{\theta}_d^{(t)}\right) \frac{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)}{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_{i-1}^{(t)}, \boldsymbol{\theta}_{i+1}^{(t)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)}}{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t)}, \dots, \boldsymbol{\theta}_d^{(t)}\right) \frac{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_i^{(t+1)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)}{p\left(\boldsymbol{\theta}_1^{(t)}, \dots, \boldsymbol{\theta}_{i-1}^{(t)}, \boldsymbol{\theta}_{i+1}^{(t)}, \dots, \boldsymbol{\theta}_d^{(t)}\right)}} \\
&= 1 \quad \Rightarrow \quad \alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = 1.
\end{aligned}$$

Portanto, os movimentos ao longo dos eixos coordenados serão sempre aceitos fazendo com que a cadeia gerada sempre esteja se movimentando.

Geman e Geman (1984) mostraram que sob certas condições, a distribuição das amostras simuladas $\{\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(M)}\}$ convergem para a distribuição conjunta de interesse $p(\boldsymbol{\theta})$, quando M é arbitrariamente grande. Vale ressaltar que, mesmo em dimensões elevadas, as simulações são realizadas por meio de distribuições condicionais univariadas, proporcionando assim uma eficiência computacional superior em comparação ao o método MH convencional.

O amostrador de Gibbs pode ser descrito pelo Algoritmo 2.

Algoritmo 2 – Amostrador de Gibbs

- 1: Tome $i = 0$ e inicialize $\boldsymbol{\theta}^{(0)} = \left(\boldsymbol{\theta}_1^{(0)}, \dots, \boldsymbol{\theta}_d^{(0)}\right)$
 - 2: Simule $\boldsymbol{\theta}_1^{(i)} \sim p\left(\boldsymbol{\theta}_1|\boldsymbol{\theta}_2^{(i-1)}, \boldsymbol{\theta}_3^{(i-1)}, \dots, \boldsymbol{\theta}_d^{(i-1)}\right)$
 - 3: Simule $\boldsymbol{\theta}_2^{(i)} \sim p\left(\boldsymbol{\theta}_2|\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_3^{(i-1)}, \dots, \boldsymbol{\theta}_d^{(i-1)}\right)$
 - 4: \vdots
 - 5: Simule $\boldsymbol{\theta}_j^{(i)} \sim p\left(\boldsymbol{\theta}_j|\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \dots, \boldsymbol{\theta}_{j-1}^{(i)}, \boldsymbol{\theta}_{j+1}^{(i-1)}, \dots, \boldsymbol{\theta}_d^{(i-1)}\right)$, para $j = 1, \dots, d$
 - 6: \vdots
 - 7: Simule $\boldsymbol{\theta}_d^{(i)} \sim p\left(\boldsymbol{\theta}_d|\boldsymbol{\theta}_1^{(i)}, \boldsymbol{\theta}_2^{(i)}, \dots, \boldsymbol{\theta}_{d-1}^{(i)}\right)$
 - 8: Defina $i = i + 1$ e retorne ao passo 2 até atingir a convergência.
-

2.5 Análise de Convergência

Os métodos MCMC desempenham um papel crucial em estatística Bayesiana e inferência estatística computacional. No entanto, avaliar a qualidade das amostras geradas pelos métodos é uma tarefa crucial, e a análise de convergência é um aspecto essencial desse processo.

Em teoria, os métodos MCMC garante que a distribuição dos valores gerados converge para a distribuição de interesse, p , quando o número de iteração i cresce para o infinito. Desta forma, surge de modo natural a questão: "A partir de qual iteração i teremos uma boa amostra da distribuição p ?". Em resposta a esta pergunta, toda uma área de pesquisa se ocupada em desenvolver técnicas de diagnóstico de convergência para métodos MCMC. Além de técnicas gráficas (plotagem do traço das cadeias, por exemplo) existem também testes de hipótese, dos quais podemos destacar o critério de Geweke (1992).

Como não temos garantia de que o valor inicial da cadeia segue a distribuição de interesse, uma prática comum é usar um período de aquecimento $B \in \mathbb{N}$, chamado de *burnin*. Desta forma, para obtermos uma amostra final de tamanho $S \in \mathbb{N}$, a cadeia é rodada $B + S$ iterações, sendo as B iniciais descartadas.

Por definição, as cadeias geradas são autocorrelacionadas. Para eliminar (ou, pelo menos, reduzir) os efeitos dessa autocorrelação, as amostras são salvas a cada k iterações, após a fase inicial de aquecimento B . Esse valor $k \in \mathbb{N}$ é comumente referido como *lag*.

Devido à natureza de passeio aleatório dos métodos descritos nas seções 2.4.2 e 2.4.3, frequentemente é necessário um considerável esforço computacional para que a cadeia atinja a convergência (NEAL *et al.*, 2011). No capítulo 3, será apresentado um método capaz de evitar esses passeios aleatórios e, assim, assegurar uma exploração mais eficiente das cadeias geradas.

2.6 Critérios para Seleção de Modelos

A tarefa de selecionar um modelo estatístico, dentre diversas opções disponíveis, é um problema de grande relevância em vários campos de estudo. Isso se deve, ao fato de que a escolha dos critérios de seleção depende do contexto específico e das características dos dados em estudo. De modo geral, é aconselhável considerar múltiplos critérios e realizar uma análise abrangente para garantir uma escolha confiável do modelo que melhor se ajusta a dinâmica dos dados analisados. Cada critério, por sua vez, oferece uma perspectiva única sobre o desempenho do modelo e sua combinação pode fornecer uma visão mais completa.

Nesta seção, serão descritos os critérios a serem empregados neste estudo, seguindo a metodologia de inferência Bayesiana.

2.6.1 Deviance Information Criterion

Uma popular escolha na literatura de inferência Bayesiana é o *Deviance Information Criterion* (DIC), introduzido por Spiegelhalter *et al.* (2002), definido como,

$$DIC = 2 \left(\log p(y|E(\boldsymbol{\theta}|y)) - E(\log p(y|\hat{\boldsymbol{\theta}})) \right).$$

Dado um conjunto de S simulações da distribuição a posteriori de $\boldsymbol{\theta}$, é possível aproximar facilmente o DIC por,

$$\widehat{DIC} = 2 \left(\log p \left(y \mid \frac{\sum_{j=1}^S \boldsymbol{\theta}_j}{S} \right) - \frac{\sum_{j=1}^S \log p(y \mid \boldsymbol{\theta}_j)}{S} \right).$$

Este critério, busca balancear a complexidade do modelo e sua capacidade de descrever os dados observados. Para uma leitura mais aprofundada, veja Spiegelhalter *et al.* (2002).

2.6.2 Watanabe Information Criterion

O critério de informação recentemente desenvolvido chamado *Watanabe Information Criterion* (WAIC) (WATANABE; OPPER, 2010), tem vantagem sobre DIC em termos do cálculo da média sobre a distribuição a posteriori e também não depender da teoria assintótica de Fisher, como apontado por Gelman *et al.* (1995).

O WAIC é definido por

$$WAIC = -2(lppd - p_{WAIC}),$$

em que

$$lppd = \sum_{i=1}^T \log p(y_i | y) = \sum_{i=1}^T \int_{\boldsymbol{\Theta}} p(y_i | \boldsymbol{\theta}) p(\boldsymbol{\theta} | y) d\boldsymbol{\theta},$$

é a log densidade preditiva pontual e

$$p_{WAIC} = \sum_{i=1}^T \text{Var}(\log p(y_i | \boldsymbol{\theta})),$$

é uma penalização para o número efetivo de parâmetros.

Podemos aproximar o critério WAIC utilizando simulações da posteriori. Então dadas S simulações, podemos calcular

$$\widehat{lppd} = \sum_{i=1}^T \log \left(\frac{1}{S} \sum_{j=1}^S p(y_i | \boldsymbol{\theta}^j) \right) \quad (2.8)$$

e

$$\widehat{p}_{WAIC} = \sum_{i=1}^T \widehat{\text{Var}}_S(\log p(y_i | \boldsymbol{\theta})),$$

em que $\widehat{\text{Var}}_S(a_i) = \frac{1}{S-1} \sum_{i=1}^S (a_i - \bar{a})^2$ é a variância amostral de cada ponto a_i sobre S simulações.

2.6.3 Leave-One-Out Cross-Validation

A estimativa Bayesiana *Approximate Leave-One-Out Cross-Validation*, chamada LOO-CV, é definida como:

$$lppd_{loo} = \sum_{i=1}^T \log p(y_i | y_{-i}) = \sum_{i=1}^T \log \int_{\Theta} p(y_i | \theta) p(\theta | y_{-i}) d\theta,$$

em que y_{-i} representa os dados excluindo o i -ésimo ponto. De acordo com [Gelfand e Smith \(1990\)](#), se os T pontos forem condicionalmente independentes, o critério LOO-CV pode ser calculado usando amostragem por importância com pesos

$$w_j = \frac{1}{p(y_i | \theta^j)} \approx \frac{p(\theta^j | y_{-i})}{p(\theta^j | y)}, j = 1, \dots, S.$$

Por outro lado, usar esses pesos diretamente pode levar a estimativas ruidosas com alta variância ou até mesmo infinitas nas caudas da distribuição. Para evitar isso, os pesos são estabilizados substituindo w_j por

$$\tilde{w}_j = \min(w_j, \sqrt{S\bar{w}}).$$

Entretanto, esse truncamento introduz um viés, que pode ser significativo em algumas situações. Uma abordagem mais precisa e confiável é proposta por [Vehtari, Gelman e Gabry \(2015\)](#), que utiliza pesos suavizados ao ajustar uma distribuição de Pareto generalizada para os pesos de importância. A estimativa LOO estabilizada, usando amostragem por importância suavizada de Pareto (PSIS), da log densidade preditiva esperada é dada por

$$\widehat{elpd}_{psis-loo} = \sum_{i=1}^T \log \left(\frac{\sum_{j=1}^S p(y_i | \theta^j) w_i^j}{\sum_{j=1}^S w_i^j} \right).$$

Conforme sugerido pela literatura sobre critérios de informação, a penalização para o número efetivo de parâmetros pode ser estimada pela diferença

$$\hat{p}_{psis-loo} = \widehat{lppd} - \widehat{elpd}_{psis-loo}.$$

Portanto, o LOO é dado por,

$$LOO = -2 \cdot \widehat{elpd}_{psis-loo},$$

em que \widehat{lppd} é calculado por (2.8).

Neste trabalho, os critérios WAIC e LOO-CV foram calculados utilizando o pacote `rstan` ([STAN DEVELOPMENT TEAM, 2022](#)), que implementa a biblioteca `loo` ([VEHTARI et al., 2022](#)).

MONTE CARLO HAMILTONIANO

Pouco tempo após a introdução do método proposto por [Metropolis *et al.* \(1953\)](#) para a simulação estocástica de partículas, [Alder e Wainwright \(1959\)](#) descreveram uma abordagem semelhante, porém determinística, baseada nas leis do movimento de Newton. Mais tarde, [Duane *et al.* \(1987\)](#) unificaram essas abordagens, dando origem ao método conhecido como "Híbrido Monte Carlo" (HMC). Posteriormente, [Neal \(1996\)](#) rebatiza este método como "Monte Carlo Hamiltoniano" em suas aplicações estatísticas.

Para ilustrar o funcionamento do método HMC, considere a simulação de realizações de uma variável aleatória contínua $\theta \in \mathbb{R}$ univariada, com densidade de probabilidade $p(\theta)$. Suponha que o gráfico da função $f(\theta) = -\log p(\theta)$ tem a forma de um sino com concavidade voltada para cima. Agora, imagine um disco movendo-se sem atrito sobre esse gráfico, em que a posição do disco no plano é representada por $(\theta, f(\theta))$. O disco parte da região de altura mínima do gráfico e move-se em direção a uma região mais elevada com velocidade constante $v = p/m$, em que m é a massa do disco e p é seu momento linear. Ao encontrar uma região inclinada, o momento linear permite que o disco continue o movimento, aumentando a energia potencial $U(\theta)$, enquanto a energia cinética $K(p)$ diminui até se anular. Nesse ponto, o disco interrompe seu movimento na direção original e inicia o retorno à região de altura mínima. Assim, a energia potencial diminui à medida que a energia cinética aumenta, mantendo constante a energia total do sistema. Esse processo de subida e descida se repete na outra direção.

Este exemplo hipotético ilustra como a dinâmica do sistema pode ser explorada para percorrer regiões em que a função $f(\theta)$ é mínima, ou seja, regiões de alta densidade de probabilidade de θ , já que $p(\theta)$ assume seus maiores valores. É importante destacar que, apesar de o disco visitar regiões de baixa densidade de probabilidade, a dinâmica do sistema o direciona para regiões de alta densidade, uma vez que o método utiliza o gradiente do logaritmo de $p(\theta)$.

3.1 Dinâmica Hamiltoniana

A dinâmica Hamiltoniana representa uma reformulação da mecânica clássica de Newton, oferecendo uma perspectiva mais aprofundada das estruturas inerentes a um sistema mecânico fechado e conservativo. Para uma leitura mais extensiva sobre este assunto, consulte [Leimkuhler e Reich \(2004\)](#).

Seja $\boldsymbol{\theta} \in \mathbb{R}^d$ a posição de uma partícula e $\mathbf{p} \in \mathbb{R}^d$ e seu momento linear. Neste caso, o Espaço de Estados do sistema é o conjunto de todos os vetores $(\boldsymbol{\theta}, \mathbf{p}) \in \mathbb{R}^d \times \mathbb{R}^d$.

Definição 4 (Função Hamiltoniana). Para um sistema mecânico fechado e conservativo, a função Hamiltoniana, $H : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$, definida por

$$H(\boldsymbol{\theta}, \mathbf{p}) = U(\boldsymbol{\theta}) + K(\mathbf{p}),$$

fornece a energia total do sistema $\forall (\boldsymbol{\theta}, \mathbf{p}) \in \mathbb{R}^d \times \mathbb{R}^d$. Usualmente, a energia cinética $K(\mathbf{p})$ é definida como

$$K(\mathbf{p}) = \frac{\mathbf{p}^t \mathbf{M}^{-1} \mathbf{p}}{2},$$

sendo \mathbf{M} a massa desta partícula.

Definição 5 (Equações do movimento). A dinâmica do sistema mecânico é totalmente determinada pelas soluções das seguintes equações diferenciais:

$$\frac{\partial \boldsymbol{\theta}}{\partial t} = \frac{\partial H}{\partial \mathbf{p}} = \nabla_{\mathbf{p}} K(\mathbf{p}) \quad (3.1)$$

e

$$\frac{\partial \mathbf{p}}{\partial t} = -\frac{\partial H}{\partial \boldsymbol{\theta}} = -\nabla_{\boldsymbol{\theta}} U(\boldsymbol{\theta}), \quad (3.2)$$

em que o estado $(\boldsymbol{\theta}(t), \mathbf{p}(t))$ esta parametrizado pelo tempo.

O conjunto de derivadas parciais (3.1) e (3.2) descrevem como as componentes $\boldsymbol{\theta}$ e \mathbf{p} evoluem no tempo ao definir uma bijeção T_s que leva o estado $(\boldsymbol{\theta}(t), \mathbf{p}(t))$ ao estado $(\boldsymbol{\theta}(t+s), \mathbf{p}(t+s))$, para qualquer intervalo de tempo de comprimento $s > 0$ ([NEAL et al., 2011](#)).

Em situações gerais, quando $\boldsymbol{\theta}$ e \mathbf{p} podem assumir altas dimensões, as soluções de (3.1) e (3.2) não são triviais, tornando necessário o uso de métodos numéricos. Um dos métodos mais conhecidos é o método de Euler, que, por meio da discretização dos tempos, calcula iterativamente uma trajetória aproximada do estado $(\boldsymbol{\theta}(t), \mathbf{p}(t))$ ao estado $(\boldsymbol{\theta}(t+s), \mathbf{p}(t+s))$, utilizando L passos de comprimento ε , de tal sorte que $s/\varepsilon = L$ seja um número inteiro.

Em sistemas conservativos, outra abordagem para o cálculo das soluções aproximadas para (3.1) e (3.2) é o método de Störmer-Verlet, também chamado de *leapfrog*. Este método é uma modificação do método de Euler e possui propriedades fundamentais na garantia da convergência das cadeias de Markov (LEIMKUHLER; REICH, 2004).

Para um instante t_0 e um pequeno passo de comprimento $\varepsilon > 0$, o método *leapfrog* pode ser descrito da seguinte maneira:

$$p(t_0 + \varepsilon/2) = p(t_0) - \frac{\varepsilon}{2} \nabla_{\boldsymbol{\theta}} U(\boldsymbol{\theta}(t_0)), \quad (3.3)$$

$$\boldsymbol{\theta}(t_0 + \varepsilon) = \boldsymbol{\theta}(t_0) - \varepsilon \mathbf{M}^{-1} p(t_0 + \varepsilon/2), \quad (3.4)$$

$$p(t_0 + \varepsilon) = p(t_0 + \varepsilon/2) - \frac{\varepsilon}{2} \nabla_{\boldsymbol{\theta}} U(\boldsymbol{\theta}(t_0 + \varepsilon)), \quad (3.5)$$

uma vez que $\nabla_{\mathbf{p}} K(\mathbf{p}) = \mathbf{M}^{-1} \mathbf{p}$.

3.2 O método Monte Carlo Hamiltoniano

A ideia fundamental por trás do HMC é introduzir conceitos da física Hamiltoniana na amostragem estatística. Ele utiliza a analogia com partículas em um campo de energia potencial para explorar eficientemente o espaço de parâmetros. O método é eficaz em superar limitações de algoritmos MCMC tradicionais, como o Metropolis-Hastings, descrito na seção 2.4.2, que podem ter dificuldades em explorar eficientemente espaços de alta dimensão. Uma leitura introdutória pode ser encontrada em Betancourt (2017).

Seja $\boldsymbol{\theta} \in R^d$ o vetor de parâmetros de interesse com densidade de probabilidade a posteriori $p(\boldsymbol{\theta}|\mathbf{x})$ e considere $\mathbf{p} \sim N_d(\mathbf{0}, \mathbf{M})$, uma variável auxiliar independente de $\boldsymbol{\theta}$ com matriz \mathbf{M} conhecida. A densidade conjunta de $(\boldsymbol{\theta}, \mathbf{p})$ é definida por

$$p(\boldsymbol{\theta}, \mathbf{p}) \propto \exp(-H(\boldsymbol{\theta}, \mathbf{p})),$$

em que a função de energia $H(\boldsymbol{\theta}, \mathbf{p})$ é dada por

$$H(\boldsymbol{\theta}, \mathbf{p}) = -\mathcal{L}(\boldsymbol{\theta}) + \frac{1}{2} \log \left\{ (2\pi)^d |\mathbf{M}| \right\} + \frac{1}{2} \mathbf{p}' \mathbf{M}^{-1} \mathbf{p}, \quad (3.6)$$

sendo $\mathcal{L}(\boldsymbol{\theta}) = \log p(\boldsymbol{\theta}|\mathbf{x})$ o logaritmo da densidade a posteriori de $\boldsymbol{\theta}$.

Neal *et al.* (2011) mostram como podemos construir uma cadeia de Markov com densidade invariante $p(\boldsymbol{\theta}|\mathbf{x})$ utilizando a dinâmica definida pela função (3.6). Para ver isso, suponha que a cadeia assuma os valores $(\boldsymbol{\theta}^{(i)}, \mathbf{p}^{(i)})$ na i -ésima iteração. Agora, o próximo candidato da

cadeia $(\boldsymbol{\theta}^*, \mathbf{p}^*)$ é proposto aplicando-se o método iterativo *leapfrog*, descrito pelas equações (3.3) - (3.5), utilizando L passos de tamanho ε definidos previamente pelo usuário. Devido a discretização do sistema de equações diferenciais, o método *leapfrog* deixa a função Hamiltoniana aproximadamente invariante, isto é,

$$H(\boldsymbol{\theta}^{(i)}, \mathbf{p}^{(i)}) - H(\boldsymbol{\theta}^*, \mathbf{p}^*) \approx 0.$$

Deste modo, a correção de Metropolis

$$\alpha(H^{(i)}, H^*) = \min \left\{ 1, \exp \left(H(\boldsymbol{\theta}^{(i)}, \mathbf{p}^{(i)}) - H(\boldsymbol{\theta}^*, \mathbf{p}^*) \right) \right\},$$

se faz necessária ao eliminar um eventual viés introduzido pela discretização (GIROLAMI; CALDERHEAD, 2011).

O método HMC pode ser sumarizado pelo algoritmo 3.

Algoritmo 3 – Monte Carlo Hamiltoniano (HMC)

- 1: Tome $i = 0$ e inicie $\boldsymbol{\theta}^{(i)}$
 - 2: Simule $\mathbf{p}^1 \sim N_d(0, M)$ e $u \sim U(0, 1)$
 - 3: $\boldsymbol{\theta}^1 = \boldsymbol{\theta}^{(i)}$ e calcule $H_1 = H(\boldsymbol{\theta}^1, \mathbf{p}^1)$
 - 4: **para** $n = 1, \dots, L-1$ **faça** (solução *leapfrog*)
 - 5: $\mathbf{p}^{n+\frac{1}{2}} = \mathbf{p}^n - \frac{\varepsilon}{2} \nabla_{\mathbf{p}} H(\boldsymbol{\theta}^n, \mathbf{p}^n)$
 - 6: $\boldsymbol{\theta}^{n+1} = \boldsymbol{\theta}^n + \varepsilon \nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}^n, \mathbf{p}^{n+\frac{1}{2}})$
 - 7: $\mathbf{p}^{n+1} = \mathbf{p}^{n+\frac{1}{2}} - \frac{\varepsilon}{2} \nabla_{\mathbf{p}} H(\boldsymbol{\theta}^{n+1}, \mathbf{p}^{n+\frac{1}{2}})$
 - 8: **fim para**
 - 9: Calcule $H_L = H(\boldsymbol{\theta}^L, \mathbf{p}^L)$ e $\alpha(H_1, H_L) = \min \{1, \exp(H_1 - H_L)\}$
 - 10: **se** $u < \alpha(H_1, H_L)$ **então**
 - 11: $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^L$
 - 12: **senão**
 - 13: $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$
 - 14: **fim se**
 - 15: Defina $i = i + 1$ e retorne ao passo 2 até atingir a convergência.
-

Podemos notar que o método HMC fornece uma vantagem prática por não exigir o uso de uma função proposta $q(\cdot)$, ao contrário do método MCMC tradicional descrito na seção 2.4.2. Outro benefício prático proporcionado pelo HMC é a capacidade de evitar passeios aleatórios

umentando o número de propostas aceitas, contanto que os parâmetros L e ε sejam escolhidos adequadamente. Dessa forma, a cadeia de Markov gerada percorre de modo mais eficiente regiões de alta densidade de probabilidade, resultando em uma convergência mais rápida.

No entanto, a tarefa de encontrar a melhor combinação para o par de parâmetros ε e L tem se mostrado mais desafiadora se comparado a outros métodos MCMC (NUGROHO; MORIMOTO, 2015). Se pudessemos simular a dinâmica perfeitamente, a taxa de aceitação seria igual à 1, pois

$$H(\boldsymbol{\theta}^{(i)}, \mathbf{p}^{(i)}) - H(\boldsymbol{\theta}^*, \mathbf{p}^*) = 0.$$

Ao utilizar aproximações, a taxa de aceitação fica dependente da magnitude de ε , uma vez que o erro total associado a cada proposta é da ordem de ε^2 (NEAL *et al.*, 2011). Desta forma, valores pequenos para ε parecem ser uma escolha adequada para otimizar a taxa de aceitação.

Contudo, se ε for pequeno demais, o número de passos L necessários para uma proposta distante será grande, causando um custo computacional elevado. Além disso, se L for muito grande o método HMC pode gerar trajetórias circulares resultando em propostas próximos do estado atual, e assim causando baixa eficiência. Por outro lado, se L for pequeno demais, as trajetórias resultantes não percorrerão grandes distâncias, gerando passeios aleatórios indesejados.

De maneira geral, a tarefa de encontrar a melhor escolha para o par de parâmetros ε e L pode exigir várias iterações de ajuste. Alguns analistas fixam o parâmetro ε enquanto otimizam o número de passos L , monitorando a taxa de aceitação. Também é possível escolher ε e L aleatoriamente dentro de um conjunto de possibilidades definidas pelo usuário (THOMAS; TU, 2021).

Outro parâmetro que necessita de ajustes é a matriz de covariâncias \mathbf{M} . A seleção desta matriz é uma parte crucial do HMC e pode ter um impacto significativo na eficácia do método. Infelizmente, há uma infinidade de escolhas possíveis, tornando impraticável explorar todas elas para encontrar a opção ideal dada uma distribuição alvo qualquer (BETANCOURT, 2017).

3.3 O método Monte Carlo Hamiltoniano em Variedade Riemanniana

Girolami e Calderhead (2011) apresentaram uma modificação do método HMC, explorando a geometria Riemanniana do espaço paramétrico de $\boldsymbol{\theta}$ para aprimorar a convergência e a mistura da cadeia Markoviana. Ao adaptar a matriz de covariância \mathbf{M} empregada no HMC, o método RMHMC leva em consideração a estrutura local da densidade conjunta de probabilidade.

A ideia fundamental reside na reformulação da função Hamiltoniana (3.6) como:

$$H(\boldsymbol{\theta}, \mathbf{p}) = -\mathcal{L}(\boldsymbol{\theta}) + \frac{1}{2} \log \left\{ (2\pi)^d |\mathbf{M}(\boldsymbol{\theta})| \right\} + \frac{1}{2} \mathbf{p}' \mathbf{M}(\boldsymbol{\theta})^{-1} \mathbf{p}.$$

Consequentemente, as equações Hamiltonianas para as variáveis de momento e posição são agora definidas por:

$$\frac{\partial \boldsymbol{\theta}}{\partial t} = \frac{\partial H}{\partial \mathbf{p}} = \mathbf{M}(\boldsymbol{\theta})^{-1} \mathbf{p},$$

e

$$\frac{\partial \mathbf{p}}{\partial t} = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) - \frac{1}{2} \text{tr} \left[\mathbf{M}(\boldsymbol{\theta})^{-1} \frac{\partial \mathbf{M}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] + \frac{1}{2} \mathbf{p}' \mathbf{M}(\boldsymbol{\theta})^{-1} \frac{\partial \mathbf{M}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \mathbf{M}(\boldsymbol{\theta})^{-1} \mathbf{p}. \quad (3.7)$$

A equação (3.7) envolve o cálculo de derivadas de segunda e terceira ordem, o que aumenta a complexidade computacional do algoritmo. Assim, soluções numéricas precisam ser adotadas, sendo o método *leapfrog* Generalizado a abordagem comumente utilizada. Para uma leitura mais aprofundada sobre o método, consulte [Leimkuhler e Reich \(2004\)](#).

Como ocorre com o HMC tradicional, essa estratégia de amostragem resulta em uma cadeia de Markov reversível e ergódica, atendendo à equação de balanço ([GIROLAMI; CALDERHEAD, 2011](#)). No entanto, não é necessário selecionar e ajustar manualmente a matriz \mathbf{M} , uma vez que ela é determinada automaticamente em cada etapa com base na geometria Riemanniana do espaço paramétrico.

Tendo em vista a perspectiva Bayesiana adotada neste trabalho, a matriz $\mathbf{M}(\boldsymbol{\theta})$ leva em consideração a probabilidade conjunta dos dados e o vetor de parâmetros, $p(\mathbf{y}, \boldsymbol{\theta})$. Desta forma, $\mathbf{M}(\boldsymbol{\theta})$ é definida da seguinte maneira:

$$\begin{aligned} \mathbf{M}(\boldsymbol{\theta}) &= -\mathbb{E}_{\mathbf{y}|\boldsymbol{\theta}} \left[\frac{\partial^2}{\partial \boldsymbol{\theta}^2} \log \{ p(\mathbf{y}, \boldsymbol{\theta}) \} \right] \\ &= -\mathbb{E}_{\mathbf{y}|\boldsymbol{\theta}} \left[\frac{\partial^2}{\partial \boldsymbol{\theta}^2} \log \{ p(\mathbf{y}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \} \right] \\ &= \mathbf{I}(\boldsymbol{\theta}) - \mathbf{H}_{\boldsymbol{\theta}}, \end{aligned} \quad (3.8)$$

em que $\mathbf{I}(\boldsymbol{\theta}) = -\mathbb{E}_{\mathbf{y}|\boldsymbol{\theta}} \left[\frac{\partial^2}{\partial \boldsymbol{\theta}^2} \mathcal{L}(\boldsymbol{\theta}) \right]$ é a matriz da informação de Fisher, sempre que o modelo $p(\mathbf{y}|\boldsymbol{\theta})$ pertence à família exponencial, e $\mathbf{H}_{\boldsymbol{\theta}}$ é o Hessiano do logaritmo da priori de $\boldsymbol{\theta}$.

No método RMHMC, a matriz $\mathbf{M}(\boldsymbol{\theta})$ é vista como um tensor métrico definido em uma variedade Riemanniana. Nesse contexto, é essencial que $\mathbf{M}(\boldsymbol{\theta})$ seja positiva definida para todos os valores de $\boldsymbol{\theta}$. Por definição, a matriz da informação de Fisher é positiva definida. Portanto,

para garantir a adequada definição de $\mathbf{M}(\boldsymbol{\theta})$, é suficiente exigir que $-\mathbf{H}_{\boldsymbol{\theta}}$ seja positiva definida, dado que a soma de matrizes positivas definidas resulta em uma matriz positiva definida (ver Apêndice A).

O método HMC funciona de modo mais eficiente quando os parâmetros não têm restrições, ou seja, quando eles podem variar em toda a extensão dos números reais. Deste modo, frequentemente é empregado alguma reparametrização, caso o espaço paramétrico seja limitado. Por exemplo, pode-se utilizar a reparametrização $\gamma = \log(\sigma)$ quando o parâmetro original $\sigma > 0$ sempre assume valores positivos. É possível mostrar que a matriz $\mathbf{I}(\boldsymbol{\theta})$ é invariante sob reparametrização. Em outras palavras, se $\mathbf{I}(\boldsymbol{\theta})$ é positiva definida, então $\mathbf{I}(\boldsymbol{\theta}(\phi))$ também será para qualquer reparametrização $\boldsymbol{\theta}(\phi)$ (AMARI; NAGAOKA, 2000).

Portanto, para assegurar o adequado funcionamento do método, é essencial garantir que o Hessiano transformado, $-\mathbf{H}_{\boldsymbol{\theta}(\phi)}$, seja positivo definido. Considere $\boldsymbol{\theta} \in \mathbb{R}^d$ como o vetor de interesse com a priori $p(\boldsymbol{\theta})$. Suponha que as componentes de $\boldsymbol{\theta}$ sejam independentes, ou seja,

$$p(\boldsymbol{\theta}) = \prod_{i=1}^d p_i(\theta_i).$$

Desta forma, o Hessiano do logaritmo da priori é expresso como

$$-\mathbf{H}_{\boldsymbol{\theta}} = \left[-\frac{\partial^2}{\partial \theta_j \partial \theta_i} \sum_{i=1}^d \log p_i(\theta_i) \right]_{(ij)} = -\text{diag} \left(\frac{\partial^2}{\partial^2 \theta_1} \log p_1(\theta_1), \dots, \frac{\partial^2}{\partial^2 \theta_d} \log p_d(\theta_d) \right). \quad (3.9)$$

A caracterização de uma matriz positiva definida pode ser feita pela seguinte proposição:

Proposição 1 (Critério de Sylvester). Seja \mathbf{A} uma matriz simétrica $n \times n$ e considere \mathbf{A}_r a matriz formada pelo canto superior esquerdo $r \times r$ de \mathbf{A} . Então a matriz \mathbf{A} é positiva definida se, e somente se, todas as matrizes $\{\mathbf{A}_r\}_{r=1}^n$ possuem determinante positivo.

Demonstração: A demonstração desta proposição pode ser encontrada em Seber e Lee (2012).

Pela igualdade (3.9), podemos notar que $-\mathbf{H}_{\boldsymbol{\theta}}$ é uma matriz diagonal. Dessa forma, a sequência dos determinantes das matrizes \mathbf{H}_r (como definidas na proposição anterior) é dada da seguinte maneira:

$$\det(\mathbf{H}_r) = \prod_{i=1}^r \left\{ -\frac{\partial^2}{\partial^2 \theta_i} \log p_i(\theta_i) \right\}, \text{ para } r = 1, \dots, d.$$

Pela Proposição 1, temos que a matriz $-\mathbf{H}$ é positiva definida sempre que os elementos da sua diagonal forem positivos. Em outras palavras, isso ocorre se as prioris $p_i(\theta_i)$ forem log-côncavas. Esta observação é valiosa ao examinarmos o Hessiano das prioris reparametrizadas $-\mathbf{H}_{\boldsymbol{\theta}(\phi)}$.

Por fim, o método RMHMC pode ser resumido pelo algoritmo 4.

Algoritmo 4 – Monte Carlo Hamiltoniano em Variedade Riemanniana (RMHMC)

- 1: Tome $i = 0$ e inicie $\boldsymbol{\theta}^{(i)}$
 - 2: Simule $\mathbf{p}^1 \sim N_d\left(0, M\left(\boldsymbol{\theta}^{(i)}\right)\right)$ e $u \sim U(0, 1)$
 - 3: Calcule $\boldsymbol{\theta}^1 = \boldsymbol{\theta}^{(i)}$ e $H_1 = H\left(\boldsymbol{\theta}^1, \mathbf{p}^1\right)$
 - 4: % Passos *Leapfrog* Generalizado
 - 5: **para** $n = 1, \dots, L-1$ **faça**
 - 6: $\hat{\mathbf{p}}^0 = \mathbf{p}^n$
 - 7: % fixo é o número de pontos fixos
 - 8: **para** $i = 1, \dots, \text{fixo}$ **faça**
 - 9: $\hat{\mathbf{p}}^i = \mathbf{p}^n - \frac{\varepsilon}{2} \nabla_{\boldsymbol{\theta}} H\left(\boldsymbol{\theta}^n, \hat{\mathbf{p}}^{i-1}\right)$
 - 10: **fim para**
 - 11: $\mathbf{p}^{n+\frac{1}{2}} = \hat{\mathbf{p}}^{fixp}$
 - 12: $\hat{\boldsymbol{\theta}}^0 = \boldsymbol{\theta}^n$
 - 13: **para** $i = 1, \dots, \text{fixo}$ **faça**
 - 14: $\hat{\boldsymbol{\theta}}^i = \boldsymbol{\theta}^n + \frac{\varepsilon}{2} \nabla_{\mathbf{p}} H\left(\boldsymbol{\theta}^n, \mathbf{p}^{n+\frac{1}{2}}\right) + \frac{\varepsilon}{2} \nabla_{\mathbf{p}} H\left(\hat{\boldsymbol{\theta}}^{i-1}, \mathbf{p}^{n+\frac{1}{2}}\right)$
 - 15: **fim para**
 - 16: $\boldsymbol{\theta}^{n+1} = \hat{\boldsymbol{\theta}}^{fixp}$
 - 17: $\mathbf{p}^{n+1} = \mathbf{p}^{n+\frac{1}{2}} - \frac{\varepsilon}{2} \nabla_{\boldsymbol{\theta}} H\left(\boldsymbol{\theta}^{n+1}, \mathbf{p}^{n+\frac{1}{2}}\right)$
 - 18: **fim para**
 - 19: Calcule $H_L = H\left(\boldsymbol{\theta}^L, \mathbf{p}^L\right)$ e $\alpha(H_1, H_L) = \min\{1, \exp(H_1 - H_L)\}$
 - 20: **se** $u < \alpha(H_1, H_L)$ **então**
 - 21: $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^L$
 - 22: **senão**
 - 23: $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$
 - 24: **fim se**
 - 25: Defina $i = i + 1$ e retorne ao passo 2 até atingir convergência
-

MODELO DE VOLATILIDADE

Em análise de séries financeiras é bem conhecido que a variância condicional (volatilidade) não é constante no tempo. Sabemos também que estas séries apresentam longos períodos com alta ou baixa volatilidade, esta característica é conhecida como agrupamento das volatilidades. Para capturar a característica de autocorrelação da variância condicional em séries financeiras, duas classes de modelos tem sido amplamente desenvolvidas: os modelos GARCH e os modelos de volatilidade estocástica (SV).

Os modelos GARCH podem ser definidos da seguinte maneira: para $p, q \in \mathbb{N}$, o modelo GARCH(p, q) é dado por

$$y_t = \sqrt{\sigma_t^2} \varepsilon_t, \quad \varepsilon_t \sim N(0, 1), \quad (4.1)$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2, \quad (4.2)$$

em que os erros observacionais ε_t são independentes, os parâmetros $\alpha_0 > 0$, $\alpha_i \geq 0$, para $i = 1, \dots, p$, os parâmetros $\beta_j \geq 0$, para $j = 1, \dots, q$ e $\sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1$. O modelo ARCH(p), é obtido tomando $\beta_j = 0$, para $j = 1, \dots, q$ na equação (4.2) e, por conseguinte, ignorando a restrição $\sum_{i=1}^p \alpha_i < 1$. Como podemos notar pela equação (4.1), os modelos GARCH definem a variância condicional σ_t^2 como uma combinação linear das volatilidades passadas adicionada aos quadrados de observações anteriores.

Por outro lado, o modelo SV, originalmente introduzidos por [Taylor \(1986\)](#) assume que a volatilidade depende de um processo estocástico, uma vez que é razoável permitir que a volatilidade sofra influência de eventos inesperados. Além disso, comparado com os modelos GARCH, eles capturam de um modo mais apropriado as principais características empíricas observadas em séries de retornos diários ([CARNERO; PENA; RUIZ, 2003](#)). Maiores detalhes sobre estes modelos podem ser encontrados em [Morettin e Tolo \(2006\)](#).

Na formulação original, assume-se que os erros observacionais dos modelos SV seguem uma distribuição normal padrão. No entanto, diversos estudos empíricos indicam que esta formulação não é realista ao considerar a curtose observada nos retornos em inúmeras aplicações empíricas (veja seção 2.1). A seguir, apresentaremos uma classe de modelos mais robusta por relaxar a suposição de normalidade.

4.1 Modelo de Volatilidade Estocástica na Média na Família Mistura de Escala Normal

Modelo de Volatilidade Estocástica na Média na Família Mistura de Escala Normal (SVM-SMN) é definido por

$$y_t = b_0 + b_1 y_{t-1} + b_2 \exp(h_t) + \exp(h_t/2) \lambda_t^{-1/2} \varepsilon_t, \quad \varepsilon_t \sim N(0, 1), \quad (4.3)$$

$$h_{t+1} = \mu + \phi(h_t - \mu) + \sigma_h \eta_t, \quad \eta_t \sim N(0, 1), \quad (4.4)$$

$$h_1 \sim N\left(\mu, \frac{\sigma_h^2}{1 - \phi^2}\right), \quad (4.5)$$

$$\lambda_t \sim p(\lambda_t | \nu), \quad (4.6)$$

em que y_t é o retorno composto e h_t a log-volatilidade no tempo t . Os parâmetros μ e σ representam a média e o desvio padrão do processo definido pela equação (4.4), respectivamente. Para que este processo seja estacionário é assumido que, o módulo do parâmetro de persistência, ϕ , seja estritamente menor que 1, ou seja, $|\phi| < 1$. As inovações ε_t e η_t são considerados serialmente e mutuamente independentes. O parâmetro b_0 representa a média da série, b_1 a autocorrelação de primeira ordem dos retornos e b_2 mede a relação dos retornos com as volatilidades esperadas e inesperadas, bem como o efeito *feedback*.

Conforme descrito na seção 2.2, a classe de modelos SVM-SMN inclui distribuições com caudas mais pesadas que a normal, dependendo da seleção adequada da distribuição de mistura $p(\lambda_t | \nu)$. Modelos com erros t-Student (SVM-t), erros Slash (SVM-S) e erros de variância-gama (SVM-VG), são obtidos tomando $\lambda_t \sim G(\nu/2, \nu/2)$, $\lambda_t \sim Be(\nu, 1)$ e $\lambda_t \sim IG(\nu/2, \nu/2)$, respectivamente. O caso normal (SVM-N) é obtido tomando a distribuição degenerada, $\lambda_t = 1$.

4.2 Estimação dos parâmetros

Seja $\theta = (b_0, b_1, b_2, \mu, \phi, \sigma, \nu)'$ o vetor de parâmetros do modelo SVM-SMN, $\mathbf{h}_{1:T} = (h_1, \dots, h_T)'$ o vetor de log-volatilidades, $\boldsymbol{\lambda}_{1:T}$ as variáveis de mistura e $\mathbf{y}_{0:T} = (y_0, \dots, y_T)'$ a informação disponível até o tempo T . A distribuição a posterior conjunta dos parâmetros e variáveis latentes pode ser escrita da forma:

$$\begin{aligned}
p(\mathbf{h}_{1:T}, \boldsymbol{\lambda}_{1:T}, \boldsymbol{\theta} | \mathbf{y}_{0:T}) &\propto p(\mathbf{y}_{1:T} | \mathbf{y}_0, \mathbf{h}_{1:T}, \boldsymbol{\lambda}_{1:T}, \boldsymbol{\theta}) p(\mathbf{h}_{1:T} | \boldsymbol{\theta}) p(\boldsymbol{\lambda}_{1:T} | \boldsymbol{\theta}) p(\boldsymbol{\theta}) \\
&= \prod_{t=1}^T p(y_t | y_{t-1}, \beta_0, \beta_1, \beta_2, h_t, \lambda_t) p(\lambda_t | \mathbf{v}) \times \\
&\quad \times p(h_1 | \mu, \phi, \sigma^2) \prod_{t=2}^T p(h_t | h_{t-1}, \mu, \phi, \sigma^2) p(\boldsymbol{\theta}), \tag{4.7}
\end{aligned}$$

em que $p(y_t | y_{t-1}, \beta_0, \beta_1, \beta_2, h_t, \lambda_t)$, $p(h_1 | \mu, \phi, \sigma^2)$ e $p(h_t | h_{t-1}, \mu, \phi, \sigma^2)$ são dadas por (4.3), (4.4) e (4.5), respectivamente. $p(\lambda_t | \mathbf{v})$ é a densidade da variável mistura de mistura e $p(\boldsymbol{\theta})$ é a priori do modelo.

Para realizar uma análise Bayesiana na estimação dos parâmetros do modelo SVM-SMN, geramos amostras aleatórias da distribuição a posteriori (4.7) utilizando os métodos MCMC descritos no capítulo 3, dado que essa distribuição conjunta não possui forma conhecida. O esquema de amostragem é descrito pelo algoritmo 5.

Algoritmo 5 – Amostrador SMN-SMN

- 1: Tome $i = 0$ e inicialize $\mathbf{h}_{1:T}^{(0)}, \boldsymbol{\lambda}_{1:T}^{(0)}, \boldsymbol{\theta}^{(0)}$
 - 2: Simule $(\mu, \phi, \sigma)^{(i+1)} \sim p(\mu, \phi, \sigma | \mathbf{h}_{1:T}^{(i)})$
 - 3: Simule $(b_0, b_1, b_2)^{(i+1)} \sim p(b_0, b_1, b_2 | \mathbf{h}_{1:T}^{(i)}, \boldsymbol{\lambda}_{1:T}^{(i)}, \mathbf{y}_{0:T})$
 - 4: Simule $\mathbf{h}_{1:T}^{(i+1)} \sim p(\mathbf{h}_{1:T} | (\mu, \phi, \sigma)^{(i+1)}, (b_0, b_1, b_2)^{(i+1)}, \boldsymbol{\lambda}_{1:T}^{(i)}, \mathbf{y}_{0:T})$
 - 5: Simule $\boldsymbol{\lambda}_{1:T}^{(i+1)} \sim p(\boldsymbol{\lambda}_{1:T} | (b_0, b_1, b_2)^{(i+1)}, \mathbf{h}_{1:T}^{(i+1)}, \mathbf{v}^{(i)}, \mathbf{y}_{0:T})$
 - 6: Simule $\mathbf{v}^{(i+1)} \sim p(\mathbf{v} | \boldsymbol{\lambda}_{1:T}^{(i+1)})$
 - 7: Defina $i = i + 1$ e retorne ao passo 2 até atingir a convergência.
-

Para completar a especificação do modelo sob o paradigma Bayesiano, precisamos definir a priori do vetor $\boldsymbol{\theta}$. De modo análogo à [Abanto-Valle, Rodríguez e Garrafa-Aragón \(2021\)](#), tomamos a priori independente

$$p(\boldsymbol{\theta}) = p(b_0)p(b_1)p(b_2)p(\mu)p(\phi)p(\sigma)p(\mathbf{v}),$$

sendo

$$b_0 \sim N(0, 10), \quad \frac{b_1 + 1}{2} \sim Be(5, 1.5), \quad b_2 \sim N(0, 10),$$

para os parâmetros da média, e

$$\mu \sim N(0, 10), \quad \frac{\phi + 1}{2} \sim Be(20, 1.5), \quad \sigma^2 \sim IG(2.5, 0.025),$$

para os parâmetros da log-volatilidade.

A especificação da priori para o parâmetro ν varia de acordo com o modelo específico em consideração, isto é, depende se estamos analisando o modelo SVM-t, SVM-S ou o modelo SVM-VG. A seguir, esses casos serão abordados em maior detalhe.

4.3 Distribuições condicionais completas

Distribuição condicional completa de $(b_0, b_1, b_2)'$:

Como $|\beta_1| < 1$, utilizamos as transformações: $\delta = \text{arctanh}(\beta_1)$ para garantir que os parâmetros variem na reta. Seja $\boldsymbol{\theta}_1 = (b_0, \delta, b_2)'$, como a distribuição conjunta $p(\mathbf{y}_{0:T} | \boldsymbol{\theta}_1)$ pertence a família exponencial (mostrar), a matriz de informação de Fisher pode ser expressa da forma

$$\mathbf{I}(\boldsymbol{\theta}_1) = -\mathbf{E}_{\mathbf{y} | \boldsymbol{\theta}_1} \left[\frac{\partial^2}{\partial \boldsymbol{\theta}_1^2} \log \{p(\mathbf{y}_{0:T} | \boldsymbol{\theta}_1)\} \right].$$

Seja $\mathcal{L}(\boldsymbol{\theta}_1) = \log \{p(\mathbf{y}_{0:T} | \boldsymbol{\theta}_1) p(\boldsymbol{\theta}_1)\}$, logo

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}_1) &= k - \frac{1}{2} \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} [y_t - \beta_0 - \beta_1 y_{t-1} - \beta_2 e^{h_t}]^2 - \frac{1}{2\sigma_{\beta_0}^2} (\beta_0 - \mu_{\beta_0})^2 \\ &\quad + (a_{\beta_1} - 1) \log(1 + \beta_1) + (b_{\beta_1} - 1) \log(1 - \beta_1) - \frac{1}{2\sigma_{\beta_2}^2} (\beta_2 - \mu_{\beta_2})^2 \\ &\quad + \log(1 - \beta_1^2), \text{ em que } k \text{ é constante.} \end{aligned}$$

Sendo $\beta_1 = \tanh(\delta)$, então $\frac{d\beta_1}{d\delta} = 1 - \beta_1^2$. Assim, o gradiente é dado por

$$\nabla_{\boldsymbol{\theta}_1} \mathcal{L}(\boldsymbol{\theta}_1) = \begin{bmatrix} \frac{\partial}{\partial \beta_0} \mathcal{L}(\boldsymbol{\theta}_1) \\ \frac{\partial}{\partial \delta} \mathcal{L}(\boldsymbol{\theta}_1) \\ \frac{\partial}{\partial \beta_2} \mathcal{L}(\boldsymbol{\theta}_1) \end{bmatrix},$$

em que

$$\frac{\partial}{\partial \beta_0} \mathcal{L}(\boldsymbol{\theta}_1) = \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} [y_t - \beta_0 - \beta_1 y_{t-1} - \beta_2 e^{h_t}] - \frac{1}{\sigma_{\beta_0}^2} (\beta_0 - \mu_{\beta_0}),$$

$$\begin{aligned} \frac{\partial}{\partial \delta} \mathcal{L}(\boldsymbol{\theta}_1) &= (1 - \beta_1^2) \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} [y_t - \beta_0 - \beta_1 y_{t-1} - \beta_2 e^{h_t}] y_{t-1} \\ &\quad + (a_{\beta_1} - 1)(1 - \beta_1) - (b_{\beta_1} - 1)(1 + \beta_1) - 2\beta_1, \end{aligned}$$

$$\frac{\partial}{\partial \beta_2} \mathcal{L}(\boldsymbol{\theta}_1) = \sum_{t=1}^T \lambda_t [y_t - \beta_0 - \beta_1 y_{t-1} - \beta_2 e^{h_t}] - \frac{1}{\sigma_{\beta_1}} (\beta_2 - \mu_{\beta_2})$$

O tensor métrico $\mathbf{M}(\boldsymbol{\theta}_1)$ é dado por:

$$\mathbf{M}(\boldsymbol{\theta}_1) = \mathbf{I}(\boldsymbol{\theta}_1) - \mathbf{H}_{\boldsymbol{\theta}_1},$$

em que

$$\mathbf{I}(\boldsymbol{\theta}_1) = \begin{bmatrix} \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} & (1 - \beta_1^2) \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1} & \sum_{t=1}^T \lambda_t \\ (1 - \beta_1^2) \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1} & (1 - \beta_1^2)^2 \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1}^2 & \sum_{t=1}^T \lambda_t y_{t-1}^2 \\ \sum_{t=1}^T \lambda_t & \sum_{t=1}^T \lambda_t y_{t-1}^2 & \sum_{t=1}^T \lambda_t e^{h_t} \end{bmatrix},$$

é a matriz de informação de Fisher e

$$-\mathbf{H}_{\boldsymbol{\theta}_1} = \begin{bmatrix} \frac{1}{\sigma_{\beta_0}^2} & 0 & 0 \\ 0 & (a_{\beta_1} + b_{\beta_1})(1 - \beta_1^2) & 0 \\ 0 & 0 & \frac{1}{\sigma_{\beta_2}^2} \end{bmatrix}, \quad (4.8)$$

é o Hessiano do logaritmo das prioris. As derivadas parciais do tensor métrico são da forma:

$$\frac{\partial}{\partial \beta_0} \mathbf{M}(\boldsymbol{\theta}_1) = \frac{\partial}{\partial \beta_2} \mathbf{M}(\boldsymbol{\theta}_1) = \mathbf{0}_{3 \times 3},$$

$$\frac{\partial}{\partial \delta} \mathbf{M}(\boldsymbol{\theta}_1) = -2\beta_1(1 - \beta_1^2) \begin{bmatrix} 0 & \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1} & 0 \\ \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1} & 2(1 - \beta_1^2) \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} y_{t-1}^2 + (a_{\beta_1} + b_{\beta_1}) & \sum_{t=1}^T \lambda_t y_{t-1} \\ 0 & \sum_{t=1}^T \lambda_t y_{t-1} & 0 \end{bmatrix}.$$

Distribuição condicional completa de $(\mu, \phi, \sigma)'$:

Como $|\phi| < 1$ e $\sigma > 0$, utilizamos as transformações: $\omega = \text{arctanh}(\phi)$ e $\gamma = \log(\sigma)$ para garantir que os parâmetros variem na reta. Seja $\boldsymbol{\theta}_2 = (\mu, \omega, \gamma)'$, como a distribuição conjunta

$p(\mathbf{h}_{1:T}|\boldsymbol{\theta}_2)$ pertence a família exponencial (mostrar), a matriz de informação de Fisher pode ser expressa da forma

$$\mathbf{I}(\boldsymbol{\theta}_2) = -\mathbf{E}_{\mathbf{h}|\boldsymbol{\theta}_2} \left[\frac{\partial^2}{\partial \boldsymbol{\theta}_2^2} \log \{p(\mathbf{h}_{1:T}|\boldsymbol{\theta}_2)\} \right].$$

Seja $\mathcal{L}(\boldsymbol{\theta}_2) = \log \{p(\mathbf{h}_{1:T}|\boldsymbol{\theta}_2)p(\boldsymbol{\theta}_2)\}$, logo

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}_2) &= k + \frac{1}{2} \log(1 - \phi^2) - \frac{1 - \phi^2}{2\sigma^2} (h_1 - \mu)^2 - \frac{1}{2\sigma^2} \sum_{t=1}^{T-1} [h_{t+1} - \mu - \phi(h_t - \mu)]^2 \\ &\quad - T \log(\sigma) + (a_\phi - 1) \log(1 + \phi) + (b_\phi - 1) \log(1 - \phi) - 2(a_\sigma + 1) \log(\sigma) \\ &\quad - \frac{b_\sigma}{\sigma^2} - \frac{1}{2\sigma_\mu^2} (\mu - \mu_0)^2 + \log(1 - \phi^2) + \log(\sigma), \text{ em que } k \text{ é constante.} \end{aligned}$$

Sendo $\phi = \tanh(\omega)$ e $\sigma = \exp(\gamma)$, temos que $\frac{d\phi}{d\omega} = 1 - \phi^2$ e $\frac{d\sigma}{d\gamma} = \sigma$. Assim, o gradiente é dado por

$$\nabla_{\boldsymbol{\theta}_2} \mathcal{L}(\boldsymbol{\theta}_2) = \begin{bmatrix} \frac{\partial}{\partial \mu} \mathcal{L}(\boldsymbol{\theta}_2) \\ \frac{\partial}{\partial \omega} \mathcal{L}(\boldsymbol{\theta}_2) \\ \frac{\partial}{\partial \gamma} \mathcal{L}(\boldsymbol{\theta}_2) \end{bmatrix},$$

em que

$$\frac{\partial}{\partial \mu} \mathcal{L}(\boldsymbol{\theta}_2) = \frac{1 - \phi^2}{\sigma^2} (h_1 - \mu) + \frac{1 - \phi}{\sigma^2} \sum_{t=1}^{T-1} [h_{t+1} - \mu - \phi(h_t - \mu)] - \frac{1}{\sigma_\mu^2} (\mu - \mu_0),$$

$$\begin{aligned} \frac{\partial}{\partial \omega} \mathcal{L}(\boldsymbol{\theta}_2) &= -\phi + \frac{\phi(1 - \phi^2)}{\sigma^2} (h_1 - \mu)^2 + \frac{1 - \phi^2}{\sigma^2} \sum_{t=1}^{T-1} [h_{t+1} - \mu - \phi(h_t - \mu)] (h_t - \mu) \\ &\quad + (a_\phi - 1)(1 - \phi) + (b_\phi - 1)(1 + \phi) - 2\phi, \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial \gamma} \mathcal{L}(\boldsymbol{\theta}_2) &= -T + \frac{1 - \phi^2}{\sigma^2} (h_1 - \mu)^2 + \frac{1}{\sigma^2} \sum_{t=1}^{T-1} [h_{t+1} - \mu - \phi(h_t - \mu)]^2 \\ &\quad - 2(a_\sigma + 1) + 2\frac{b_\sigma}{\sigma^2} + 1. \end{aligned}$$

O tensor métrico $\mathbf{M}(\boldsymbol{\theta}_2)$ é dado por:

$$\mathbf{M}(\boldsymbol{\theta}_2) = \mathbf{I}(\boldsymbol{\theta}_2) - \mathbf{H}_{\boldsymbol{\theta}_2},$$

em que

$$\mathbf{I}(\boldsymbol{\theta}_2) = \begin{bmatrix} \frac{(1-\phi)^2(T-1)}{\sigma^2} + \frac{1-\phi^2}{\sigma^2} & 0 & 0 \\ 0 & 2\phi^2 + (T-1)(1-\phi^2) & 2\phi \\ 0 & 2\phi & 2T \end{bmatrix},$$

é a matriz de informação de Fisher e

$$-\mathbf{H}_{\boldsymbol{\theta}_2} = \begin{bmatrix} \frac{1}{\sigma_\mu^2} & 0 & 0 \\ 0 & (a_\phi + b_\phi) & 0 \\ 0 & 0 & \frac{4b_\sigma}{\sigma^2} \end{bmatrix}, \quad (4.9)$$

é o Hessiano do logaritmo das prioris. As derivadas parciais do tensor métrico são da forma:

$$\frac{\partial}{\partial \mu} \mathbf{M}(\boldsymbol{\theta}_2) = \mathbf{0}_{3 \times 3}$$

$$\frac{\partial}{\partial \omega} \mathbf{M}(\boldsymbol{\theta}_2) = \begin{bmatrix} \frac{-2(1-\phi^2)}{\sigma^2} [(1-\phi)(T-1) + \phi] & 0 & 0 \\ 0 & 2\phi(1-\phi^2)[2 - (T-1) - a_\phi - b_\phi] & 2(1-\phi^2) \\ 0 & 2(1-\phi^2) & 0 \end{bmatrix}$$

$$\frac{\partial}{\partial \gamma} \mathbf{M}(\boldsymbol{\theta}_2) = \begin{bmatrix} \frac{-2}{\sigma^2} [(1-\phi)^2(T-1) + 1 - \phi^2] & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{-8b_\sigma}{\sigma^2} \end{bmatrix}.$$

Podemos notar que as Hessianas dos logaritmos das prioris (4.8) e (4.9) apresentam todos os elementos da diagonal principal como positivos. Assim, de acordo com a Proposição 1 (critério de Sylvester), essas Hessianas definem métricas, como em (3.8).

Distribuição condicional completa de $\mathbf{h}_{1:T}$:

Seja $\mathcal{L}(\mathbf{h}_{1:T})$ dado por

$$\begin{aligned} \mathcal{L}(\mathbf{h}_{1:T}) = & k - \frac{1}{2} \sum_{t=1}^T h_t - \frac{1}{2\sigma^2} \sum_{t=1}^{T-1} [h_{t+1} - \mu - \phi(h_t - \mu)]^2 - \frac{1-\phi^2}{2\sigma^2} (h_1 - \mu)^2 \\ & - \frac{1}{2} \sum_{t=1}^T \frac{\lambda_t}{e^{h_t}} [y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t}]^2, \text{ em que } k \text{ é constante.} \end{aligned}$$

Logo o gradiente é definido da forma

$$\nabla_{\mathbf{h}_{1:T}} \mathcal{L}(\mathbf{h}_{1:T}) = \begin{bmatrix} \frac{\partial}{\partial h_1} \mathcal{L}(\mathbf{h}_{1:T}) \\ \vdots \\ \frac{\partial}{\partial h_t} \mathcal{L}(\mathbf{h}_{1:T}) \\ \vdots \\ \frac{\partial}{\partial h_T} \mathcal{L}(\mathbf{h}_{1:T}) \end{bmatrix},$$

em que

$$\begin{aligned} \frac{\partial}{\partial h_1} \mathcal{L}(\mathbf{h}_{1:T}) &= -\frac{1}{2} + \frac{\lambda_1}{2e^{h_1}} (y_1 - b_0 - b_1 y_0 - b_2 e^{h_1})^2 + b_2 \lambda_1 (y_1 - b_0 - b_1 y_0 - b_2 e^{h_1}) \\ &\quad + \frac{\phi}{\sigma^2} (h_2 - \mu - \phi(h_1 - \mu)) - \frac{1 - \phi^2}{\sigma^2} (h_1 - \mu), \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial h_t} \mathcal{L}(\mathbf{h}_{1:T}) &= -\frac{1}{2} + \frac{\lambda_t}{2e^{h_t}} (y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t})^2 + b_2 \lambda_t (y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t}) \\ &\quad + \frac{\phi}{\sigma^2} (h_{t+1} - \mu - \phi(h_t - \mu)) - \frac{1}{\sigma^2} (h_t - \mu - \phi(h_{t-1} - \mu)), \end{aligned}$$

para $1 < t < T$,

$$\begin{aligned} \frac{\partial}{\partial h_T} \mathcal{L}(\mathbf{h}_{1:T}) &= -\frac{1}{2} + \frac{\lambda_T}{2e^{h_T}} (y_T - b_0 - b_1 y_{T-1} - b_2 e^{h_T})^2 + b_2 \lambda_T (y_T - b_0 - b_1 y_{T-1} - b_2 e^{h_T}) \\ &\quad - \frac{1}{\sigma^2} (h_T - \mu - \phi(h_{T-1} - \mu)). \end{aligned}$$

Ao considerar o método RMHMC para amostar as volatilidades $\mathbf{h}_{1:T}$, é necessário derivar a expressão para a matriz métrica $\mathbf{M}(\mathbf{h}_{1:T})$ e suas respectivas derivadas parciais, como feito anteriormente. A formulação da matriz métrica é representada por $\mathbf{M}(\mathbf{h}_{1:T}) = \frac{1}{2}\mathbf{I} + b_2^2\mathbf{G} + \mathbf{C}$, em que \mathbf{I} denota a matriz identidade $T \times T$, $\mathbf{G} = \text{diag}\{\lambda_1 \cdot e^{h_1}, \dots, \lambda_T \cdot e^{h_T}\}$, e \mathbf{C} é uma matriz tridiagonal. Nesta matriz, os elementos da diagonal são $(1 + \phi^2)/\sigma^2$, exceto pelo primeiro e último elemento, que são $1/\sigma^2$. Os elementos da diagonal superior e inferior são ambos $-\phi/\sigma^2$. Como podemos notar, a matriz métrica depende de $\mathbf{h}_{1:T}$, resultando na necessidade de empregar o método *leapfrog* generalizado a cada nova iteração ao propor um novo candidato, aumentando assim o custo computacional. Além disso, o cálculo da inversa $\mathbf{M}(\mathbf{h}_{1:T})^{-1}$ contribui para elevar ainda mais este custo, especialmente quando o tamanho amostral, T , é consideravelmente grande. Para mitigar esta questão, consideramos $\mathbf{M}(\mathbf{h}_{1:T}) = \mathbf{I}d_T$ ($\mathbf{I}d_T$ é a matriz identidade $T \times T$) com o objetivo de utilizarmos o método HMC. Mais adiante, na seção 5.1, mostramos que esta escolha pode conduzir a resultados satisfatórios.

Distribuição condicional completa de $\lambda_{1:T}$:

- SVM-t

Sendo $\lambda_t | \nu \sim G\left(\frac{\nu}{2}, \frac{\nu}{2}\right)$, a distribuição condicional completa de λ_t é dada por

$$\lambda_t | h_t, \boldsymbol{\theta}, y_t, y_{t-1} \sim G\left(\frac{\nu+1}{2}, \frac{\{y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t}\}^2 e^{-h_t} + \nu}{2}\right), \text{ para } t = 1, \dots, T.$$

- SVM-S

Sendo $\lambda_t | \nu \sim Be(\nu, 1)$, a distribuição condicional completa de λ_t é dada por

$$\lambda_t | h_t, \boldsymbol{\theta}, y_t, y_{t-1} \sim G_{(\lambda_t < 1)}\left(\nu + \frac{1}{2}, \frac{1}{2}\{y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t}\}^2 e^{-h_t}\right), \text{ para } t = 1, \dots, T,$$

em que $G_{(\lambda_t < 1)}$ é a distribuição gama truncada a direita.

Para garantir estabilidade numérica, optamos por implementar o método de simulação de uma variável aleatória gama truncada, ao invés de utilizarmos o pacote `truncdist` (NADARAJAH; KOTZ, 2006), disponibilizado em R (R Core Team, 2022). A abordagem utilizada pelos autores, se beneficia do fato de que a função quantil `qgamma` (implementada em R) é a inversa da função acumulada da distribuição Gama, possibilitando a aplicação do método da função inversa. Desta forma, simulações de uma distribuição Gama truncada podem ser obtidas facilmente (veja Proposição 7 no Apêndice A). No entanto, ao considerarmos a distribuição $G_{(X < 1)}(a, b)$ quando o hiperparâmetro b é próximo de zero, a constante normalizadora necessária para simular uma realização dessa variável truncada pode ter um denominador numericamente nulo, resultando em erro computacional. Uma maneira de contornar esse problema é considerar estes valores na escala logarítmica. Maiores detalhes desta implementação podem ser obtidos no Algoritmo 8.

- SVM-VG

Sendo $\lambda_t | \nu \sim IG\left(\frac{\nu}{2}, \frac{\nu}{2}\right)$, a distribuição condicional completa de λ_t é dada por

$$\lambda_t | h_t, \boldsymbol{\theta}, y_t, y_{t-1} \sim GIG\left(\frac{1-\nu}{2}, \{y_t - b_0 - b_1 y_{t-1} - b_2 e^{h_t}\}^2 e^{-h_t}, \nu\right), \text{ para } t = 1, \dots, T,$$

em que GIG representa a distribuição Normal Inversa Generalizada, e sua simulação é realizada por meio do pacote `GIGrvg`, desenvolvido por Leydold, Hormann e Leydold (2017).

Distribuição condicional completa de v :

Como mencionado anteriormente, a especificação da priori para o parâmetro v depende do membro da classe SVM-SMN. Para o modelo SVM-S, adotamos $v \sim G(0.08, 0.04)$, conforme realizado por [Abanto-Valle e Garrafa-Aragón \(2019\)](#). Esta escolha resulta em uma posteriori completa com forma conhecida, dada por

$$v|\lambda_{1:T} \sim G_{v>1} \left(T + a_v, b_v - \sum_{t=1}^T \log \lambda_t \right),$$

que é uma distribuição gama truncada à esquerda.

Considerando agora os modelos SVM-t e SVM-VG, não encontramos uma escolha adequada para a priori de v de modo que esta tenha condicional completa com forma conhecida, como o modelo SVM-S. Portanto, optamos por empregar o método RMHMC, conforme descrito na seção 3.3, para realizar simulações para esses modelos. Além disso, seguindo a abordagem de [Abanto-Valle *et al.* \(2010\)](#) e [Abanto-Valle e Garrafa-Aragón \(2019\)](#), especificamos v variando em intervalos limitados. Para SVM-t, consideramos $v \in (2, 40)$, enquanto que para o modelo SVM-VG, v varia no intervalo $(0, 40)$.

Por razão de eficiência, consideramos a transformação

$$\xi(v) = \frac{1}{b} \tanh^{-1} \left(\frac{v-c}{a} \right), \quad (4.10)$$

em que $v \in (i, s)$ varia em um intervalo limitado genérico, $2a = s - i$, $2b = \alpha$ e $2c = s + i$. Note que a transformação $\xi(v)$ leva o intervalo limitado (i, s) , na reta real \mathbb{R} , uma vez que

$$i < v < s \Rightarrow -1 < \left(\frac{v-c}{a} \right) < 1,$$

e a imagem da função $\tanh^{-1}(x)$, definida em $(-1, 1)$, é a reta \mathbb{R} . Cabe destacar, que a transformação 4.10 depende dos limite inferior i e superior s dos intervalos definidos. Deste modo, foi tomado $i = 2$ para SVM-t, e $i = 0$ para o modelo SVM-VG. O limite superior $s = 40$ foi definido para ambos os modelos.

Para que a condição 3.8 seja atendida no caso univariado, é suficiente que a priori do parâmetro transformado, ξ , seja log-côncava. Desta forma, tomamos

$$\xi \sim \mathcal{N} \left(\mu_\xi, \sigma_\xi^2 \right),$$

que é uma distribuição log-côncava. Os hiperparâmetros μ_ξ , σ_ξ e α foram escolhidos de maneira a garantir que a distribuição a priori $p(v)$ tenha média aproximada de $\mathbf{E}(v) \approx 12$ e variância de $\text{Var}(v) \approx 70$ no caso do SVM-t. No caso do modelo SVM-VG, os valores são ajustados para

alcançar uma média de $\mathbf{E}(\mathbf{v}) \approx 4$ e variância de $\text{Var}(\mathbf{v}) \approx 25$. Valores similares são encontrados nas prioris selecionadas por [Abanto-Valle e Garrafa-Aragón \(2019\)](#) para os modelos SVM-t e SVM-VG.

A obtenção dos momentos de \mathbf{v} é uma tarefa desafiadora devido à dificuldade de encontrar uma solução analítica. Para ver isso, note que a inversa da transformação 4.10 é da forma

$$\mathbf{v}(\xi) = a \cdot \tanh(b\xi) + c, \quad (4.11)$$

e o cálculo de $E(\mathbf{v})$ e $\text{Var}(\mathbf{v})$ envolve solucionar a integral

$$\frac{1}{\sqrt{2\pi}\sigma_\xi} \int_{-\infty}^{\infty} \tanh(b\xi) \exp\left(-\frac{1}{2\sigma_\xi^2}(\xi - \mu_\xi)^2\right) d\xi,$$

que é de difícil manipulação analítica. Assim, após realizarmos $S = 10000$ simulações de ξ a partir da distribuição $N(\mu_\xi, \sigma_\xi^2)$, estimamos $E(\mathbf{v})$ e $\text{Var}(\mathbf{v})$ por meio da média e variância amostral, respectivamente, da amostra $\{\mathbf{v}_1, \dots, \mathbf{v}_S\}$, em que $\mathbf{v}_i = \mathbf{v}(\xi_i)$ e $i = 1, \dots, S$.

Finalmente, as log-posteriori $\mathcal{L}(\xi) = \sum_{t=1}^T \log p(\lambda_t | \mathbf{v}(\xi))$ e suas derivadas $\nabla_\xi \mathcal{L}(\xi)$, são dadas, respectivamente, por

$$\mathcal{L}(\xi) = k + \frac{T\mathbf{v}(\xi)}{2} \log\left(\frac{\mathbf{v}(\xi)}{2}\right) - T \log \Gamma\left(\frac{\mathbf{v}(\xi)}{2}\right) + \frac{\mathbf{v}(\xi)}{2} \sum_{t=1}^T (\log \lambda_t - \lambda_t) - \frac{1}{2\sigma_\xi^2} (\xi - \mu_\xi)^2$$

e

$$\nabla_\xi \mathcal{L}(\xi) = \frac{d\mathbf{v}}{d\xi} \left(\frac{T}{2} \log\left(\frac{\mathbf{v}}{2}\right) - \frac{T}{2} \psi^{(0)}\left(\frac{\mathbf{v}}{2}\right) + \frac{T}{2} + \frac{1}{2} \sum_{t=1}^T (\log \lambda_t - \lambda_t) \right) - \frac{1}{\sigma_\xi^2} (\xi - \mu_\xi),$$

para o modelo SVM-t, enquanto que para o modelo SVM-VG, temos

$$\mathcal{L}(\xi) = k + \frac{T\mathbf{v}(\xi)}{2} \log\left(\frac{\mathbf{v}(\xi)}{2}\right) - T \log \Gamma\left(\frac{\mathbf{v}(\xi)}{2}\right) - \frac{\mathbf{v}(\xi)}{2} \sum_{t=1}^T (\log \lambda_t - \lambda_t^{-1}) - \frac{1}{2\sigma_\xi^2} (\xi - \mu_\xi)^2$$

e

$$\nabla_\xi \mathcal{L}(\xi) = \frac{d\mathbf{v}}{d\xi} \left(\frac{T}{2} \log\left(\frac{\mathbf{v}}{2}\right) - \frac{T}{2} \psi^{(0)}\left(\frac{\mathbf{v}}{2}\right) + \frac{T}{2} - \frac{1}{2} \sum_{t=1}^T (\log \lambda_t - \lambda_t^{-1}) \right) - \frac{1}{\sigma_\xi^2} (\xi - \mu_\xi).$$

Nestas expressões, $\Gamma(\cdot)$ representa a função gama, $\psi^{(n)}(\cdot) = \frac{d^{n+1}}{dx^{n+1}} \log \Gamma(\cdot)$ a função poligama de ordem n e $\frac{d\mathbf{v}}{d\xi}$ é a derivada da transformação inversa (4.11).

O tensor métrico $M(\xi)$, para ambos os modelos SVM-t e SVM-VG, fica definido por

$$M(\xi) = - \left(\frac{d\mathbf{v}}{d\xi} \right)^2 T \left(\frac{1}{2\mathbf{v}(\xi)} - \frac{1}{4} \psi^{(1)}\left(\frac{\mathbf{v}(\xi)}{2}\right) \right),$$

de modo que sua derivada é da forma

$$\begin{aligned} \frac{\partial}{\partial \xi} M(\xi) = & -2 \left(\frac{dv}{d\xi} \right) \left(\frac{d^2v}{d\xi^2} \right) T \left(\frac{1}{2v(\xi)} - \frac{1}{4} \psi^{(1)} \left(\frac{v(\xi)}{2} \right) \right) \\ & + \left(\frac{dv}{d\xi} \right)^3 T \left(\frac{1}{2v(\xi)} + \frac{1}{8} \psi^{(2)} \left(\frac{v(\xi)}{2} \right) \right). \end{aligned}$$

ESTUDOS DE SIMULAÇÃO E APLICAÇÕES

Neste capítulo, são apresentados os resultados obtidos nas simulações e nas aplicações a dados reais do modelos introduzidos no capítulo 4.

A implementação dos códigos computacionais foi realizada utilizando os pacotes Rcpp e RcppArmadillo, os quais são extensões que simplificam a integração entre as linguagens R e C++. Para uma compreensão mais detalhada sobre esses pacotes, consulte [Eddelbuettel e François \(2011\)](#).

Os cálculos computacionais foram realizados no cluster de alto desempenho Euler, que é mantido pelo Centro de Ciências Matemáticas Aplicadas à Indústria (CeMEAI). Maiores informações sobre o cluster podem ser obtidas no endereço [<https://euler.cemeai.icmc.usp.br/>](https://euler.cemeai.icmc.usp.br/).

5.1 Estudo de simulação 1

Este estudo de simulação busca ilustrar o método de estimação proposto. Desta maneira, foram geradas $T = 3000$ observações artificiais dos modelos SVM-t, SVM-S e SVM-VG, assumindo $(b_0, b_1, b_2)' = (0.10, 0.03, -0.10)'$, para os parâmetros da média e $(\mu, \phi, \sigma)' = (0.10, 0.98, 0.15)'$, para os parâmetros da volatilidade. O parâmetro v foi fixado de acordo com a especificação do modelo. Para SVM-t e SVM-VG, tomamos $v = 10$, enquanto que para o modelo SVM-S, v foi fixado sendo igual a 2. Estes valores são frequentemente encontrados na literatura.

Os verdadeiros valores foram utilizados como valores iniciais para as cadeias MCMC e as priori utilizadas são descritas na seção 4.2. Com base nos dados artificiais, foram conduzidas 100000 iterações MCMC, sendo as 20000 iniciais descartadas como *burn-in*. Além disso, a fim de diminuir as autocorrelações entre os valores consecutivos, foram registrados uma em cada vinte amostras das restantes, resultando em uma amostra de 4000 simulações.

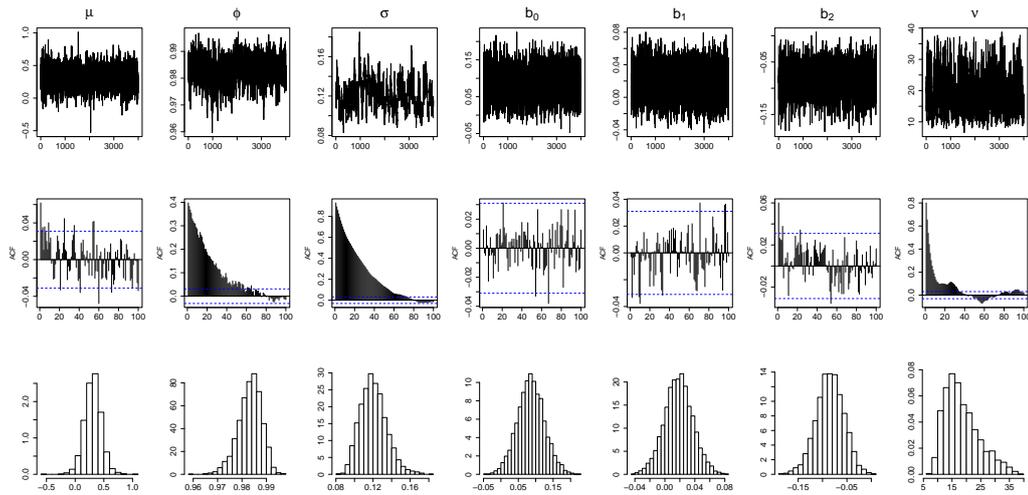
Para avaliar a eficiência das amostras, foi considerado o fator de ineficiência ¹ (IF), definido por $IF = 1 + 2\sum_{s=1}^{\infty} \rho_s$, em que ρ_s é a autocorrelação amostral no lag s . O fator de ineficiência avalia o esquema de amostragem pelo aumento da variância da amostra a posteriori quando comparada com a de amostras independentes. Ou seja, quanto mais próximo o valor de IF estiver de 1, mais eficiente é a amostra. Com as 4000 amostras consideradas, foram calculados a média a posteriori, o intervalo de credibilidade de 95%, o fator IF e a estatística Diagnóstico de Convergência (DC) (GEWEKE, 1992). Os resultados obtidos estão resumidos a seguir.

Tabela 1 – Resultados da estimação dos modelos SVM-t, SVM-S e SVM-VG. Foram reportados a média a posteriori, o intervalo de 95% de credibilidade, o fator IF e a estatística DC.

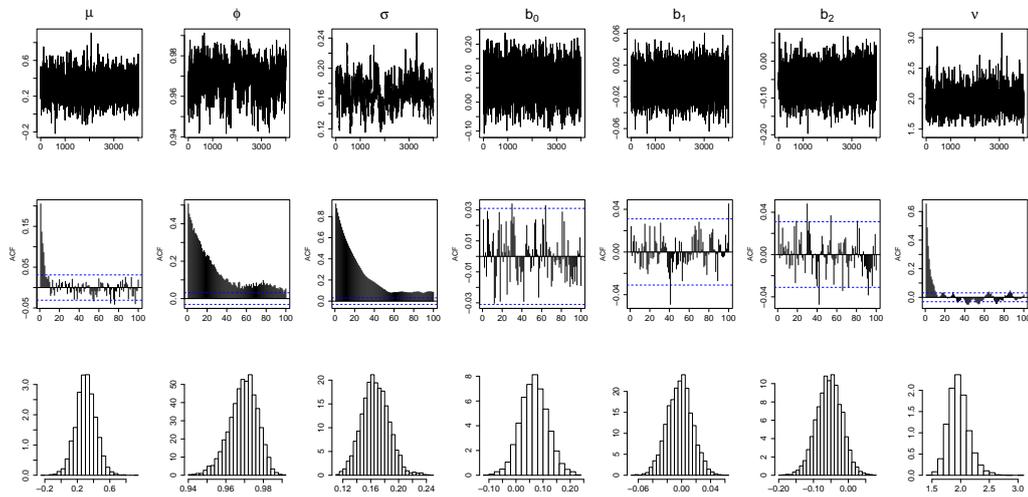
Modelo	Parâmetro	Real	Média	Intervalo 95%	IF	DC
SVM-t	b_0	0.10	0.0864	(0.0103, 0.1631)	1.00	1.60
	b_1	0.03	0.0183	(-0.0197, 0.0562)	0.93	1.47
	b_2	-0.10	-0.0848	(-0.1424, -0.0319)	1.38	-1.45
	μ	0.10	0.3085	(0.0179, 0.5953)	1.62	1.58
	ϕ	0.98	0.9833	(0.9727, 0.9915)	23.00	0.54
	σ	0.15	0.1205	(0.0965, 0.1510)	48.19	-0.54
	ν	10.0	18.4606	(9.9540, 33.0000)	9.69	0.30
SVM-S	b_0	0.10	0.0671	(-0.0328, 0.1694)	1.05	0.68
	b_1	0.03	-0.0025	(-0.0367, 0.0311)	1.05	-1.03
	b_2	-0.10	-0.0550	(-0.1273, 0.0147)	1.08	-0.85
	μ	0.10	0.3054	(0.0607, 0.5461)	2.30	0.49
	ϕ	0.98	0.9693	(0.9528, 0.9827)	24.75	-0.32
	σ	0.15	0.1668	(0.1296, 0.2055)	41.62	-0.15
	ν	2.0	1.9663	(1.6604, 2.3581)	5.88	-0.49
SVM-VG	b_0	0.10	0.0892	(0.0195, 0.1619)	1.00	0.14
	b_1	0.03	0.0083	(-0.0276, 0.0443)	1.00	-1.58
	b_2	-0.10	-0.0882	(-0.1434, -0.0353)	1.00	-0.54
	μ	0.10	0.2911	(-0.0018, 0.5911)	1.00	-1.06
	ϕ	0.98	0.9841	(0.9727, 0.9925)	31.35	-1.49
	σ	0.15	0.1133	(0.0891, 0.1511)	74.34	1.00
	ν	10.0	10.8449	(5.8644, 20.4653)	6.93	1.23

Pela Tabela 1, notamos que os intervalos de 95% de credibilidade contém os verdadeiros valores para todos os parâmetros dos modelos considerados. Os intervalos de credibilidade para os parâmetro b_1 , incluem o zero. É importante notar que o valor original de b_1 é bem pequeno.

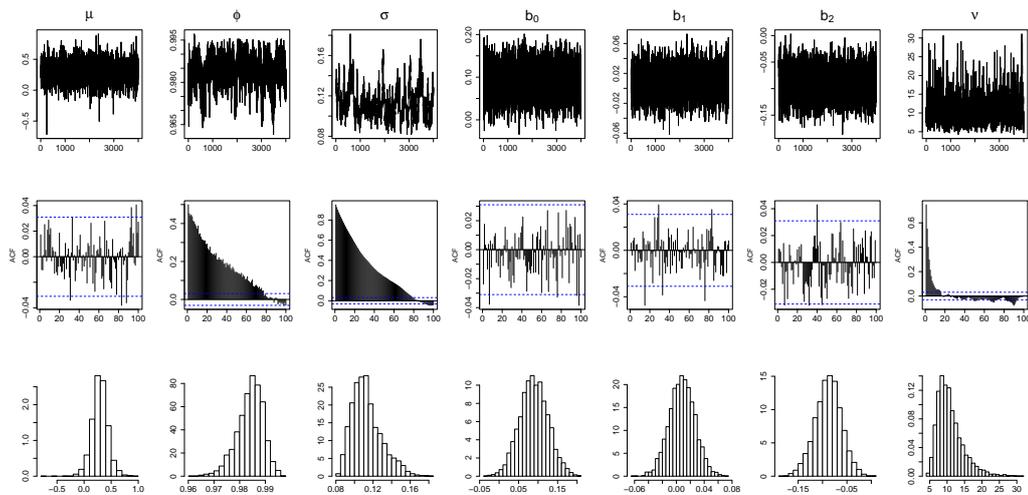
¹ O fator IF foi calculado com auxílio do pacote coda (PLUMMER *et al.*, 2015).



(a) Resultados estimados para o modelo SVM-t.



(b) Resultados estimados para o modelo SVM-S.



(c) Resultados estimados para o modelo SVM-VG.

Figura 1 – Resultados estimados para os modelos SVM-t, SVM-S e SVM-VG. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.

O mesmo ocorre com os parâmetros b_0 e b_2 no modelo SVM-S. Observamos, ainda, que o intervalo de credibilidade para o parâmetro μ no modelo SVM-VG também inclui o zero, no entanto, o limite inferior desse parâmetro é praticamente nulo. De acordo com a estatística DC, a hipótese nula de estacionariedade das sequências de 4000 amostras podem ser aceitas com um nível de significância de 5%. Isso se deve ao fato de que todos os valores de DC estão contidos no intervalo $(-1.96, 1.96)$, para todos os parâmetros. Ao analisar a Tabela 1, observa-se que os fatores de ineficiência dos modelos considerados estão próximos de 1 para a maioria dos parâmetros, com exceção de ϕ , σ e ν .

A Figura 1 exibe os traços das cadeias geradas para os parâmetros dos modelos em estudo, bem como as autocorrelações e histogramas. Pode-se notar uma baixa autocorrelação para os parâmetros μ , b_0 , b_1 e b_2 , em todos os modelos. Apesar de uma elevada autocorrelação inicial para os parâmetros ν , podemos observar um rápido decaimento para todos os modelos. Por outro lado, os parâmetros ϕ e σ apresentam altas autocorrelações, sugerindo uma baixa qualidade nas amostras geradas, o que é confirmado pelo elevado fator de ineficiência (IF), conforme ilustrado na Tabela 1.

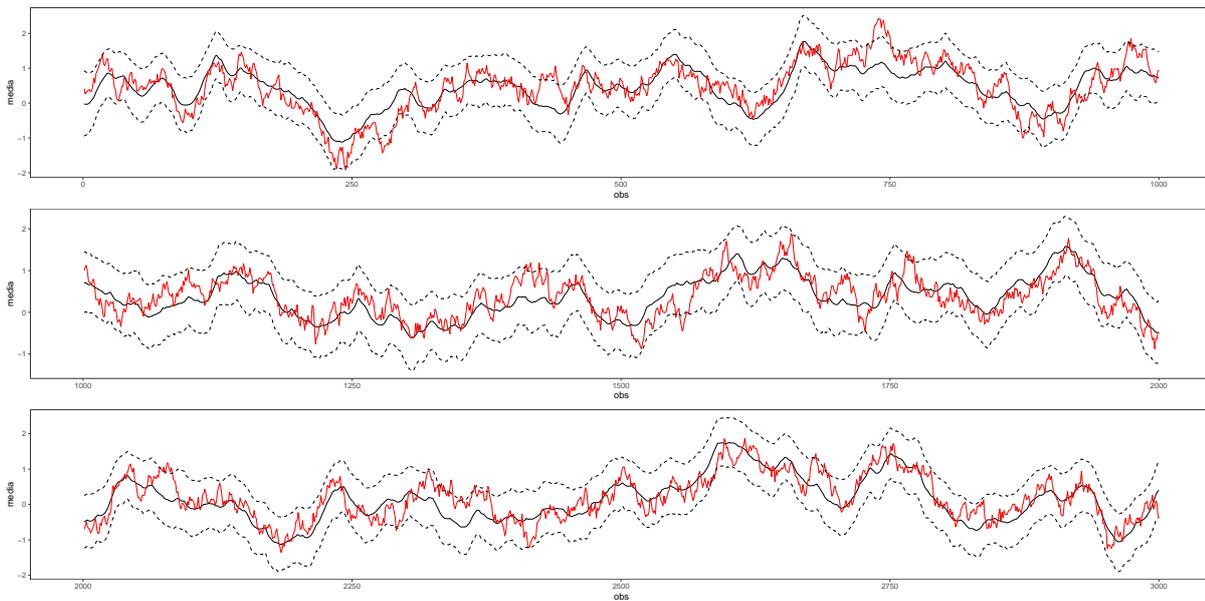


Figura 2 – Resultados das volatilidades estimadas para SVM-S. A linha vermelha representa a verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.

As Figuras 3, 2 e 4 apresentam as volatilidades suavizadas estimadas de $\exp(h_t/2)$ para os modelos SVM-t, SVM-S e SVM-VG em comparação com as verdadeiras volatilidades simuladas. As estimativas das volatilidades suavizadas são obtidas por $\frac{1}{S} \sum_{i=1}^S \exp(h_t^{(i)}/2)$, em que $h_t^{(i)}$ é o valor de h_t para a i -ésima iteração do algoritmo 5, e S é o número total de simulações.

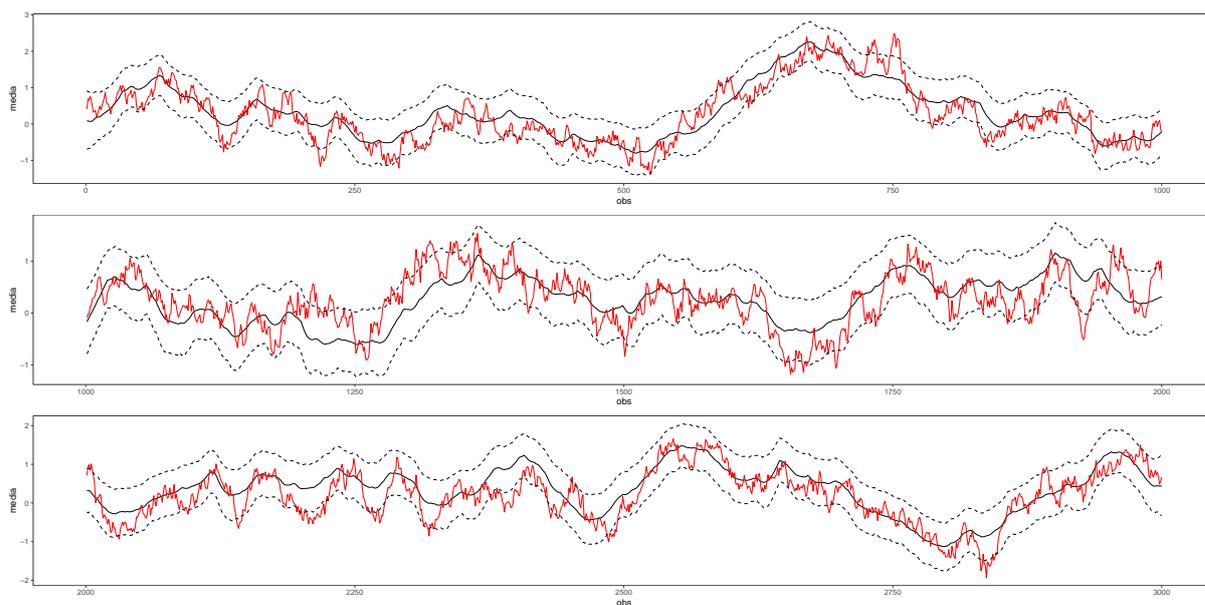


Figura 3 – Resultados das volatilidades estimadas para SVM-t. A linha vermelha representa o verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.

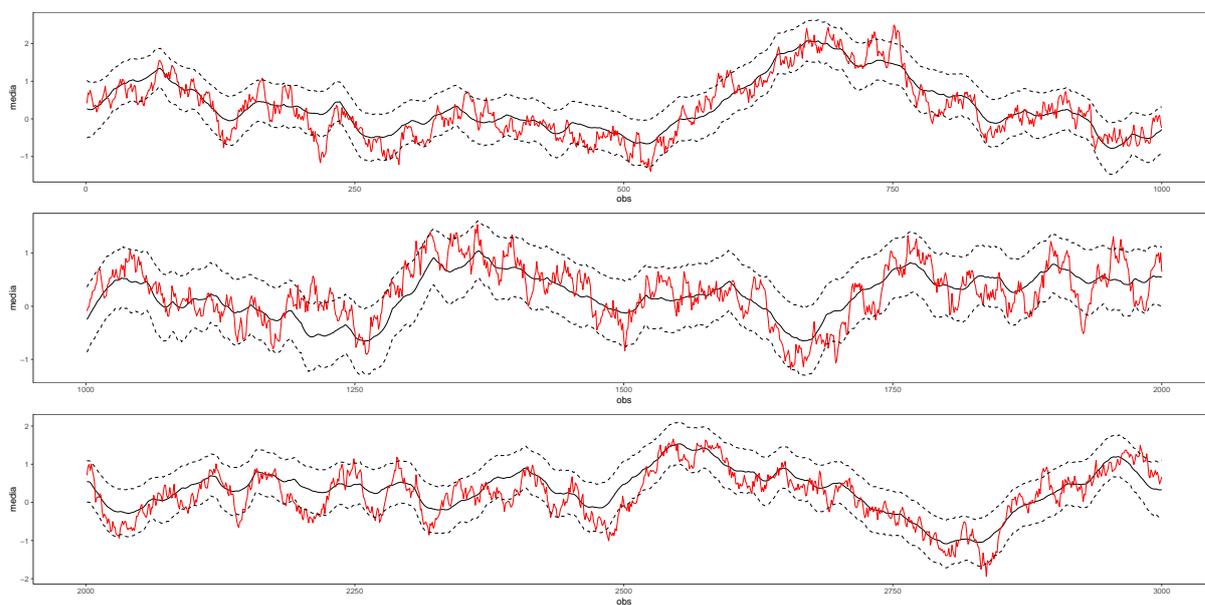


Figura 4 – Resultados das volatilidades estimadas para SVM-VG. A linha vermelha representa o verdadeira volatilidade simulada. A linha preta sólida representa a média a posteriori estimada, enquanto as linhas tracejadas representam a região de 95% de credibilidade.

Conforme observado, os valores médios das volatilidades estimadas (representados pela linha preta sólida) mantêm uma proximidade notável com os valores reais (indicados pela linha vermelha) em todos os modelos, e as áreas de 95% de credibilidade (representadas por linhas tracejadas) englobam as volatilidades verdadeiras na maior parte do tempo.

5.2 Estudo de simulação 2

Este estudo de simulação busca avaliar o desempenho do algoritmo proposto ao estimar os parâmetros dos modelos SVM-SMN. Para tanto, foram geradas $m = 200$ réplicas com diferentes tamanhos amostrais $T \in \{1000, 1500, 3000\}$ dos modelos SVM-t, SVM-S e SVM-VG, considerando os cenários: $(b_0, b_1, b_2)' = (0.10, 0.03, -0.10)'$, $(\mu, \sigma)' = (0.10, 0.15)$ e $\phi \in \{0.95, 0.98\}$. O parâmetro ν foi fixado de acordo com a especificação do modelo. Para SVM-t e SVM-VG, tomamos $\nu = 10$, enquanto que para o modelo SVM-S, ν foi fixado sendo igual à 2. Valores similares foram utilizados por [Girolami e Calderhead \(2011\)](#), [Leão, Abanto-Valle e Chen \(2017\)](#), [Abanto-Valle et al. \(2017\)](#) e [Zevallos, Gasco e Ehlers \(2017\)](#) em seus estudos de simulação.

Para avaliar o desempenho do método, foi considerado o viés relativo (viés_{rel}) e a raiz quadrada do erro quadrático médio relativo (REQM_{rel}), definidos da seguinte forma:

$$\text{viés}_{rel}(\hat{\theta}) = \frac{1}{m} \sum_{i=1}^m \left(\frac{\hat{\theta}^{(i)} - \theta}{\theta} \right) \quad \text{e} \quad \text{REQM}_{rel}(\hat{\theta}) = \sqrt{\frac{1}{m} \sum_{i=1}^m \left(\frac{\hat{\theta}^{(i)} - \theta}{\theta} \right)^2},$$

em que θ é o vetor de parâmetros original e $\hat{\theta}^{(i)}$ representa a média a posteriori estimada na i -ésima réplica, com $i = 1, \dots, m$. Pode-se mostrar (veja Apêndice A) que

$$\text{EQM}(\hat{\theta}) = \text{Var}(\hat{\theta}) + \text{viés}(\hat{\theta})^2, \quad (5.1)$$

$$\text{viés}_{rel}(\hat{\theta}) = \frac{1}{\theta} \text{viés}(\hat{\theta}) \quad \text{e} \quad (5.2)$$

$$\text{REQM}_{rel}(\hat{\theta}) = \frac{1}{|\theta|} \text{REQM}(\hat{\theta}). \quad (5.3)$$

Assim, através das equações (5.2) e (5.3), notamos que valores próximos de zero para as quantidades viés_{rel} e REQM_{rel} implicam em valores de viés e REQM também próximos de zero. Essa proximidade, por sua vez, indica uma boa precisão das estimativas de θ , pois a variabilidade do estimador $\hat{\theta}$ é baixa, como evidenciado pela equação (5.1).

Os verdadeiros valores dos parâmetros foram utilizados como valores iniciais para as cadeias MCMC e as priors utilizadas são descritas na seção 4.3. Com base nas simulações realizadas, foram conduzidas 20000 iterações, sendo as 10000 iniciais descartadas como *burn-in*, como feito por [Zevallos, Gasco e Ehlers \(2017\)](#). Além disso, a fim de diminuir as autocorrelações entre os valores consecutivos, foram registrados uma em cada dez amostras das restantes, resultando em uma amostra final de 1000 simulações.

Os resultados obtidos estão resumidos nas tabelas a seguir. Para cada modelo SVM-SMN, a primeira e segunda linha das tabelas apresentam, respectivamente, as medidas viés_{rel} e REQM_{rel} de cada parâmetro para diferentes tamanhos amostrais T .

Tabela 2 – Viés_{rel} e REQM_{rel} das estimativas $\hat{\theta}$ dos modelos SVM-SMN, com $\phi = 0.95$ em que $T \in \{1000, 1500, 3000\}$.

T	Modelo		b_0	b_1	b_2	μ	ϕ	σ	ν
1000	SVM-N	viés_{rel}	-0.583	-0.099	-0.563	0.139	0.004	-0.245	-
		REQM_{rel}	2.374	1.091	2.251	1.151	0.032	0.256	-
	SVM-t	viés_{rel}	0.345	-0.029	0.299	0.034	-0.011	-0.148	0.215
		REQM_{rel}	1.781	1.105	1.657	1.222	0.036	0.222	0.454
	SVM-S	viés_{rel}	0.348	-0.053	0.425	-0.219	-0.015	-0.160	0.023
		REQM_{rel}	2.028	1.059	1.989	1.319	0.039	0.237	0.200
	SVM-VG	viés_{rel}	0.680	-0.160	0.614	-0.088	-0.014	-0.157	0.127
		REQM_{rel}	2.370	1.159	2.181	1.052	0.040	0.224	0.388
1500	SVM-N	viés_{rel}	-0.143	-0.080	-0.135	0.090	0.010	-0.222	-
		REQM_{rel}	0.906	0.892	0.776	0.895	0.015	0.235	-
	SVM-t	viés_{rel}	0.223	-0.090	0.187	0.112	-0.003	-0.109	0.234
		REQM_{rel}	1.184	0.902	1.045	0.975	0.022	0.225	0.459
	SVM-S	viés_{rel}	0.180	-0.022	0.207	-0.188	-0.004	-0.121	0.023
		REQM_{rel}	1.439	0.898	1.372	1.105	0.022	0.230	0.168
	SVM-VG	viés_{rel}	0.302	-0.076	0.283	-0.026	-0.006	-0.105	0.134
		REQM_{rel}	1.806	0.962	1.617	0.914	0.027	0.207	0.358
3000	SVM-N	viés_{rel}	0.005	-0.028	-0.009	0.114	0.008	-0.166	-
		REQM_{rel}	0.558	0.615	0.475	0.680	0.013	0.190	-
	SVM-t	viés_{rel}	0.072	-0.033	0.040	0.116	-0.002	-0.049	0.233
		REQM_{rel}	0.695	0.603	0.608	0.723	0.019	0.197	0.406
	SVM-S	viés_{rel}	0.079	-0.053	0.083	-0.095	0.001	-0.064	0.007
		REQM_{rel}	0.953	0.606	0.886	0.829	0.017	0.210	0.104
	SVM-VG	viés_{rel}	0.113	0.043	0.084	0.003	0.001	-0.088	0.072
		REQM_{rel}	0.685	0.660	0.606	0.649	0.014	0.194	0.258

A Tabela 2 mostra os resultados obtidos com as simulações considerando $\phi = 0.95$. Podemos notar que os valores do viés_{rel} estão próximos de zero, indicando que na média as estimativas obtidas para a média a posteriori dos parâmetros está próxima dos valores pré-fixados nas simulações. Além disso, notamos que os valores do REQM_{rel} diminuem à medida que o

tamanho amostral, T , aumenta, indicando que as estimativas tornam-se mais precisas com a disponibilidade de mais informações.

Tabela 3 – $Viés_{rel}$ e $REQM_{rel}$ das estimativas $\hat{\theta}$ dos modelos SVM-SMN, com $\phi = 0.98$ em que $T \in \{1000, 1500, 3000\}$.

T	Modelo		b_0	b_1	b_2	μ	ϕ	σ	ν
1000	SVM-N	$viés_{rel}$	-0.023	-0.010	-0.035	0.155	-0.001	-0.166	-
		$REQM_{rel}$	0.667	1.063	0.575	2.425	0.007	0.187	-
	SVM-t	$viés_{rel}$	0.099	0.028	0.050	0.220	-0.004	-0.060	0.263
		$REQM_{rel}$	0.759	1.076	0.685	2.369	0.012	0.167	0.476
	SVM-S	$viés_{rel}$	0.136	-0.005	0.143	-0.124	-0.007	0.003	0.059
		$REQM_{rel}$	0.832	1.055	0.775	2.459	0.014	0.182	0.235
	SVM-VG	$viés_{rel}$	0.106	-0.072	0.104	-0.320	-0.005	-0.055	0.236
		$REQM_{rel}$	0.662	1.128	0.626	2.347	0.014	0.173	0.439
1500	SVM-N	$viés_{rel}$	-0.045	-0.038	-0.059	0.053	0.002	-0.151	-
		$REQM_{rel}$	0.501	0.887	0.408	1.984	0.006	0.174	-
	SVM-t	$viés_{rel}$	0.068	-0.072	0.029	0.297	-0.003	-0.029	0.297
		$REQM_{rel}$	0.561	0.886	0.449	1.945	0.009	0.160	0.492
	SVM-S	$viés_{rel}$	0.030	0.047	0.062	-0.239	-0.003	-0.036	0.052
		$REQM_{rel}$	0.615	0.902	0.584	2.016	0.009	0.174	0.223
	SVM-VG	$viés_{rel}$	0.016	-0.025	0.033	-0.186	-0.002	-0.038	0.199
		$REQM_{rel}$	0.471	0.947	0.443	2.039	0.009	0.151	0.392
3000	SVM-N	$viés_{rel}$	0.004	-0.018	-0.008	0.135	0.002	-0.099	-
		$REQM_{rel}$	0.302	0.620	0.242	1.510	0.005	0.132	-
	SVM-t	$viés_{rel}$	0.039	-0.031	0.004	0.188	-0.001	-0.022	0.256
		$REQM_{rel}$	0.355	0.601	0.299	1.428	0.007	0.125	0.419
	SVM-S	$viés_{rel}$	0.032	-0.033	0.027	-0.106	-0.001	-0.021	0.015
		$REQM_{rel}$	0.434	0.606	0.400	1.479	0.007	0.141	0.105
	SVM-VG	$viés_{rel}$	0.028	0.063	0.013	-0.130	-0.001	-0.032	0.129
		$REQM_{rel}$	0.323	0.669	0.290	1.441	0.006	0.130	0.307

Os resultados obtidos ao considerar $\phi = 0.98$ estão resumidos na Tabela 3. Observa-se que as medidas $viés_{rel}$ e $REQM_{rel}$ demonstram um comportamento semelhante à configuração anterior. Em outras palavras, o $viés_{rel}$ dos parâmetros se mantém próximo de zero, enquanto o $REQM_{rel}$ diminui ao passo que mais informação se torna disponível.

5.3 Aplicação a dados reais

Nesta seção a ilustração do método é realizada utilizando séries financeiras reais. Ajustamos os modelos SVM-SMN apresentados no capítulo 4 aos índices IBOVESPA e as taxas de câmbio £/USD². O período de análise para os índices IBOVESPA abrange de 5 de janeiro de 1998 a 3 de outubro de 2005, resultando em um total de $T = 1917$ observações. Este mesmo conjunto de dados foi utilizado por [Abanto-Valle, Migon e Lachos \(2012\)](#). Para as taxas de câmbio £/USD, tomamos o período que abrange de 1 de janeiro de 2004 a 2 de junho de 2016, resultando em um total de $T = 3227$ observações.

Para estas análises foi considerado o retorno composto,

$$y_t = 100 \times \log \left(\frac{P_t}{P_{t-1}} \right),$$

em que P_t é preço de fechamento do dia t . A Tabela 4 fornece algumas estatísticas descritivas para as série de retornos compostos.

Tabela 4 – Estatísticas resumo para as série de retornos analisadas.

	T	Média	Desvio p.	Mínimo	Máximo	Assimetria	Curtose
IBOVESPA	1917	0.06	2.34	-17.21	28.83	0.83	19.21
£/USD	3227	-0.01	0.59	-3.87	3.48	-0.31	5.86

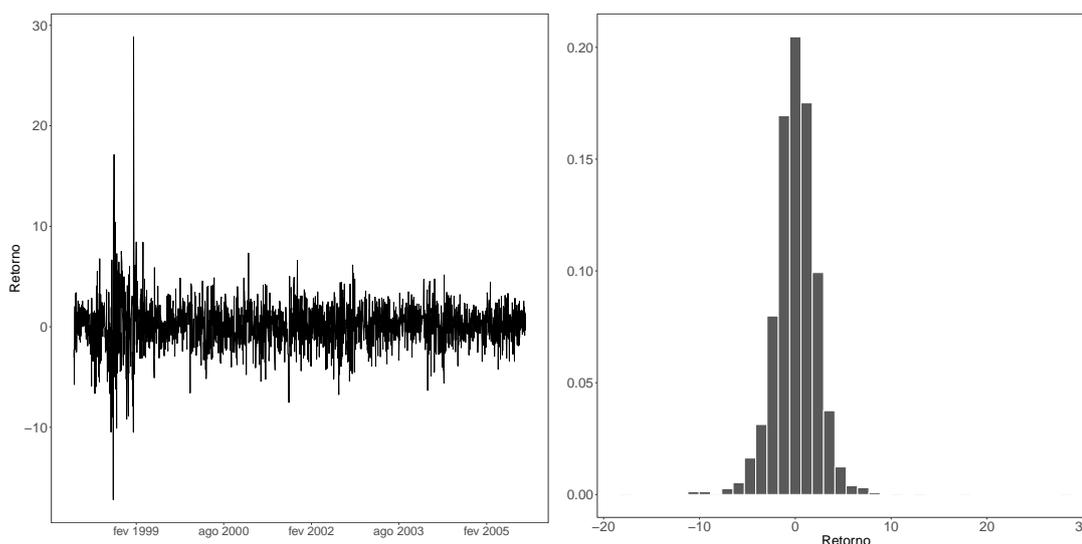


Figura 5 – Retornos compostos do índice IBOVESPA de 5 de janeiro de 1998 a 3 de outubro de 2005. O painel da esquerda mostra a série dos retorno, enquanto o painel da direita traz o histograma da série.

² As séries IBOVESPA e £/USD estão disponíveis no site: <https://finance.yahoo.com>.

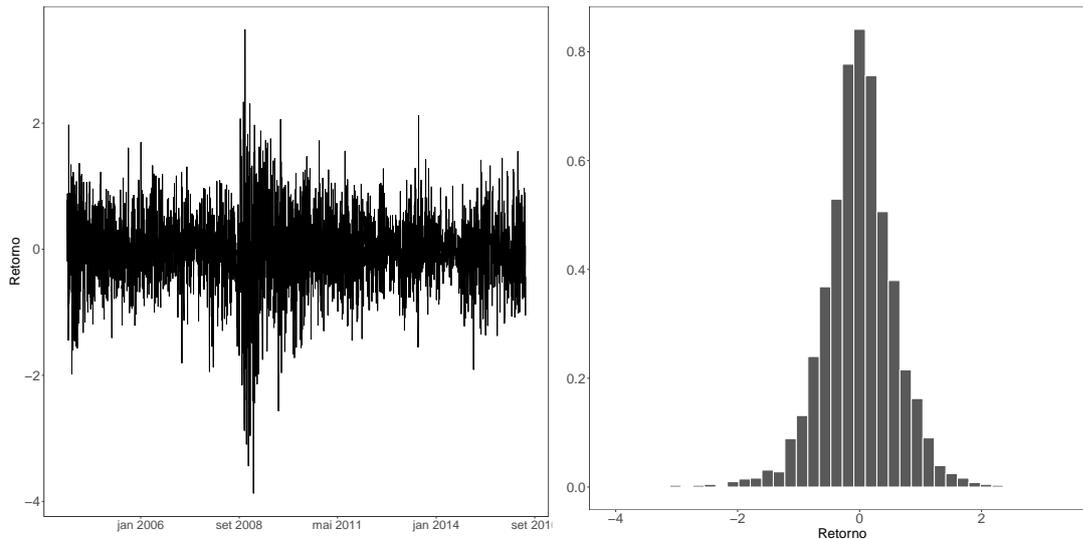


Figura 6 – Taxas de câmbio £/USD de 1 de janeiro de 2004 a 2 de junho de 2016. O painel da esquerda mostra a série das taxas, enquanto o painel da direita traz o histograma da série.

Pode-se notar pela Figura 5 que os dados IBOVESPA apresentam uma leve assimetria positiva (0.58) e uma grande amplitude (mín. = -17.21 e máx. = 28.83). Por outro lado, a Figura 6 mostra que os dados £/USD apresentam uma leve assimetria negativa (-0.31) e uma tímida amplitude (mín. = -3.87 e máx. = 3.48). Ambas as séries exibem uma curtose amostral maior que 3, sugerindo a presença de valores extremos. Assim, serão ajustados os modelos SVM-SMN conforme descritos no capítulo 4, utilizando os métodos inferenciais apresentados no capítulo 3.

Além disso, a seleção de modelos será feita segundo os critérios de informação introduzidos na seção 2.6. As distribuições a priori dos vetores de parâmetros (b_0, b_1, b_2) , (μ, ϕ, σ) e v , são descritas na seção 4.2. Os valores iniciais desses vetores foram configurados conforme o cenário adotado no primeiro estudo de simulação. Inicialmente, o vetor de log-volatilidades h_t foi simulado utilizando as equações 4.4 e 4.5. Por fim, os valores iniciais de λ_t foram gerados a partir das prioris (4.6).

Para todos os modelos, foram realizadas 100000 iterações MCMC, descartando-se as 20000 primeiras como *burn-in*. A fim de reduzir as autocorrelações entre iterações consecutivas, adotou-se saltos de tamanho 20, resultando em amostras de tamanho 4000 para cada modelo. Com base nessas amostras, foram calculadas a média a posteriori, os intervalos de 95% de credibilidade, as estatísticas de diagnóstico de convergência (DC) e o fator de ineficiência (IF) para os parâmetros.

As Tabelas 5 e 6 resumem os resultados encontrados. De acordo com as tabelas, a hipótese nula de que a sequência de 4000 amostras é estacionária é aceita com ao nível de 5% para todos os parâmetros e modelos, uma vez que as estatísticas DC estão contidas no intervalo $(-1.96, 1.96)$ (terceira linha). As estimativas para b_2 , que quantifica a relação entre retornos e

Tabela 5 – Resultados estimados para os retornos IBOVESPA. Primeira linha : Média a posteriori. Segunda linha: Intervalos de 95% de credibilidade. Terceira linha: Estatística de Geweke (DC). Quarta linha: Fator de ineficiência.

Parâmetro	SVM-N	SVM-t	SVM-S	SVM-VG
μ	1.292	1.2190	1.0252	1.3088
	(0.9010, 1.6726)	(0.7936, 1.6145)	(0.5974, 1.4506)	(0.8846, 1.7127)
	1.80	-0.30	-1.62	-1.06
	0.81	1.30	2.29	1.00
ϕ	0.9833	0.9831	0.9844	0.9835
	(0.9711, 0.9933)	(0.9690, 0.9938)	(0.9712, 0.9944)	(0.9704, 0.9944)
	-0.02	-0.16	-0.22	-0.36
	9.69	18.07	14.52	15.18
σ	0.1257	0.1230	0.1153	0.1222
	(0.0997, 0.1564)	(0.0902, 0.1660)	(0.0839, 0.1504)	(0.0896, 0.1541)
	-0.11	0.06	0.34	0.42
	27.96	47.96	40.78	34.47
b_0	0.2433	0.2802	0.2906	0.2770
	(0.0979, 0.3874)	(0.1274, 0.4396)	(0.1340, 0.4517)	(0.1197, 0.4356)
	0.22	0.15	1.17	0.48
	1.07	1.09	1.08	1.00
b_1	0.0295	0.0297	0.0328	0.0280
	(-0.0159, 0.0740)	(-0.0157, 0.0754)	(-0.0137, 0.0782)	(-0.0203, 0.0727)
	0.67	-0.46	-0.35	-0.40
	1.00	1.00	1.05	1.00
b_2	-0.0369	-0.0515	-0.0672	-0.0456
	(-0.0739, -0.0023)	(-0.0946, -0.0104)	(-0.1250, -0.0163)	(-0.0838, -0.0079)
	0.17	-0.56	-1.12	-0.40
	1.00	1.13	2.28	1.00
ν	-	24.7378	4.4579	21.8134
	-	(13.5034, 35.3943)	(2.8325, 8.4038)	(9.4299, 32.9362)
	-	-0.55	-1.58	-0.88
	-	4.51	35.50	9.45

volatilidade, assim como o efeito de *feedback*, apresentaram médias negativas. Além disso, para a serie IBOVESPA, observamos que esse parâmetro é estatisticamente significativo nos modelos, uma vez que os intervalos de credibilidade não incluem o zero. Estes resultados vão de encontro

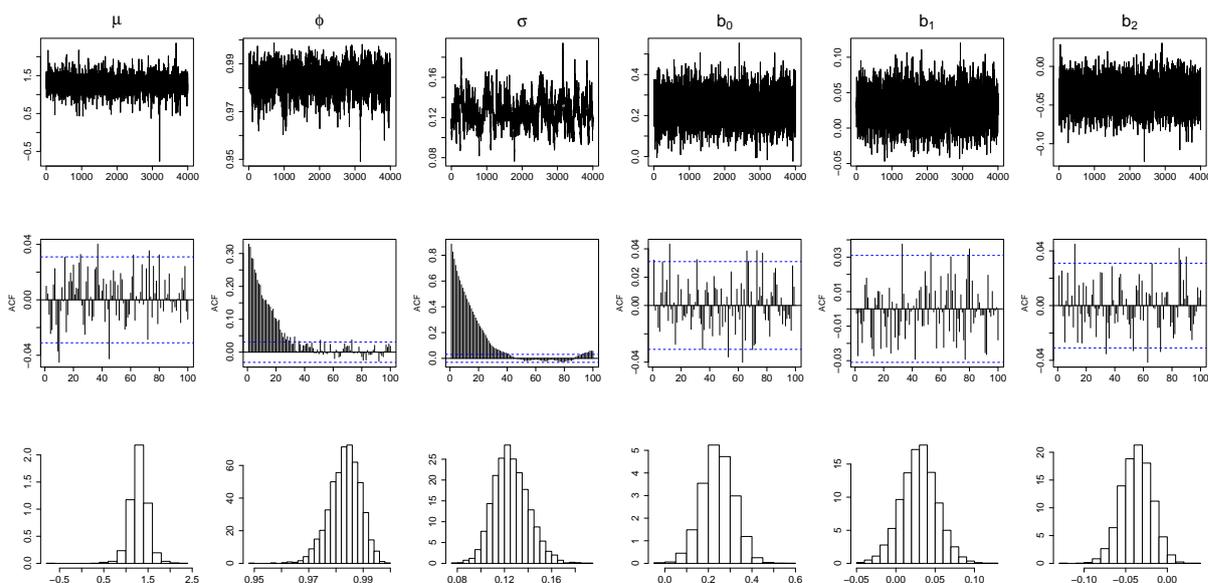
Tabela 6 – Resultados estimados para os retornos £/USD. Primeira linha : Média a posteriori. Segunda linha: Intervalos de 95% de credibilidade. Terceira linha: Estatística de Geweke (DC). Quarta linha: Fator de ineficiência.

Parâmetro	SVM-N	SVM-t	SVM-S	SVM-VG
μ	-1.2616	-1.3664	-1.5447	-1.2317
	(1.6278, -0.8418)	(-1.8151, -0.8940)	(-2.0050, -1.0607)	(-1.6825, -0.7237)
	-0.43	-0.65	0.06	0.0430
	1.00	1.31	3.65	1.00
ϕ	0.9910	0.9931	0.9928	0.9936
	(0.9848, 0.9962)	(0.9874, 0.9977)	(0.9873, 0.9974)	(0.9886, 0.9978)
	0.98	0.71	0.00	0.07
	9.20	21.79	12.82	14.67
σ	0.0877	0.0763	0.0783	0.0733
	(0.0730, 0.1056)	(0.0617, 0.0967)	(0.0632, 0.0960)	(0.0568, 0.0897)
	-1.09	-0.56	-0.28	0.14
	39.30	73.72	58.09	58.82
b_0	0.0189	0.0151	0.0160	0.0147
	(-0.0099, 0.0487)	(-0.0155, 0.0443)	(-0.0143, 0.0444)	(-0.0159, 0.0445)
	0.96	-1.52	0.15	-1.09
	1.00	1.00	1.07	1.00
b_1	-0.0151	-0.0132	-0.0130	-0.0129
	(-0.0501, 0.0197)	(-0.0490, 0.0225)	(-0.0464, 0.0217)	(-0.0467, 0.0223)
	0.99	-0.19	0.79	-0.79
	1.10	1.00	1.00	1.00
b_2	-0.0718	-0.0656	-0.0833	-0.0563
	(-0.1787, 0.0288)	(-0.1865, 0.0536)	(-0.2233, 0.0568)	(-0.1602, 0.0458)
	-0.60	-0.55	-0.29	1.42
	1.00	1.00	1.08	1.08
ν	-	18.4769	4.0813	13.1403
	-	(10.1413, 32.5596)	(2.7286, 7.1001)	(6.7678, 25.7170)
	-	0.22	0.10	-0.95
	-	11.38	36.97	8.92

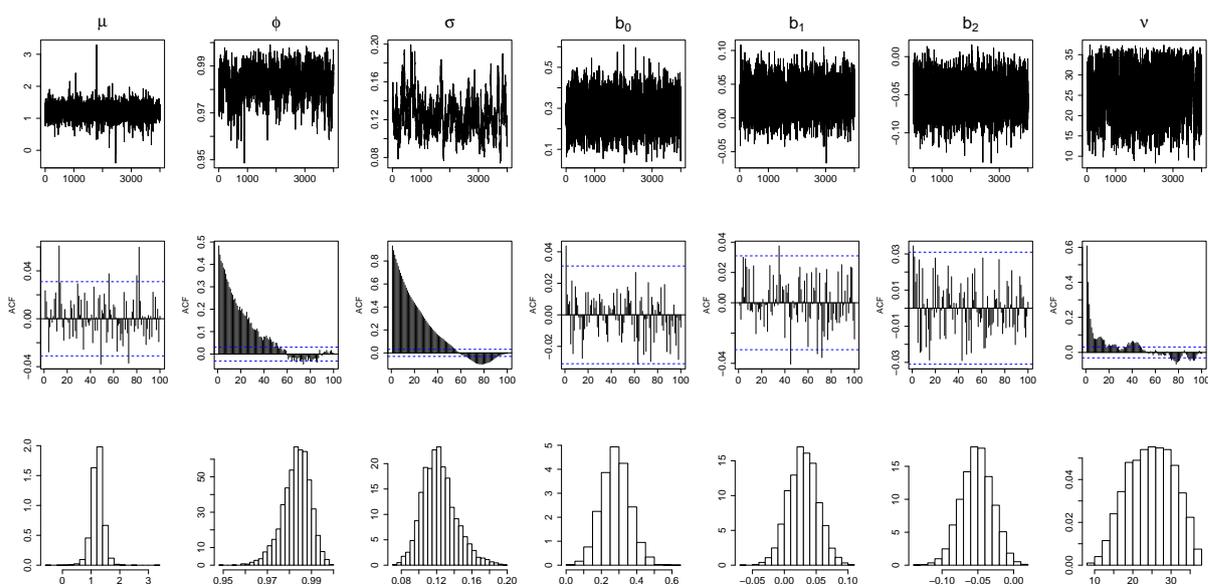
aos resultados encontrados na literatura. Contudo, para a série £/USD o mesmo não ocorre, já que os intervalos de credibilidade incluem o zero. Os intervalos de credibilidade encontrados para b_1 incluem o zero, desta forma este parâmetro pode não ser significativo. Observamos que

em todos os casos da aplicação IBOVESPA, as estimativas para b_0 são positivas e significativas estatisticamente. Porém, na aplicação da série £/USD, os intervalos incluem o zero.

A magnitude de v mede a intensidade das caudas pesadas nos modelos SVM-t, SVM-S e SVM-VG. Os intervalos de credibilidade deste parâmetro apresentam alta amplitude, sugerindo uma menor precisão nas estimativas. Este indício é corroborado pelos valores elevados do fator de ineficiência (IF) deste parâmetros para todos os modelos em ambas aplicações.



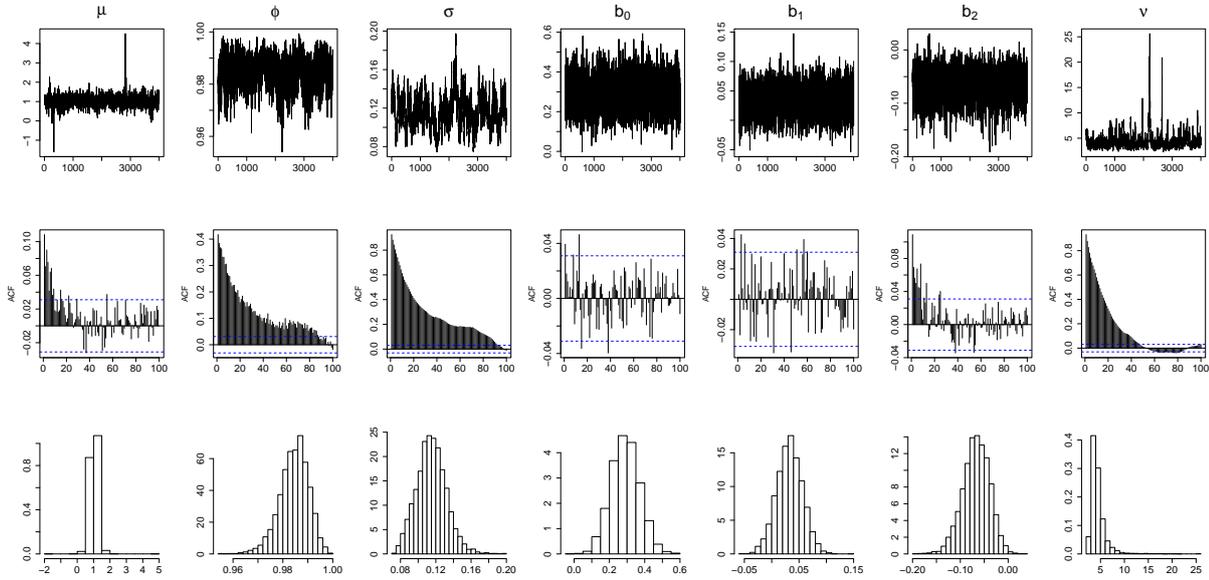
(a) Resultados estimados para o modelo SVM-N.



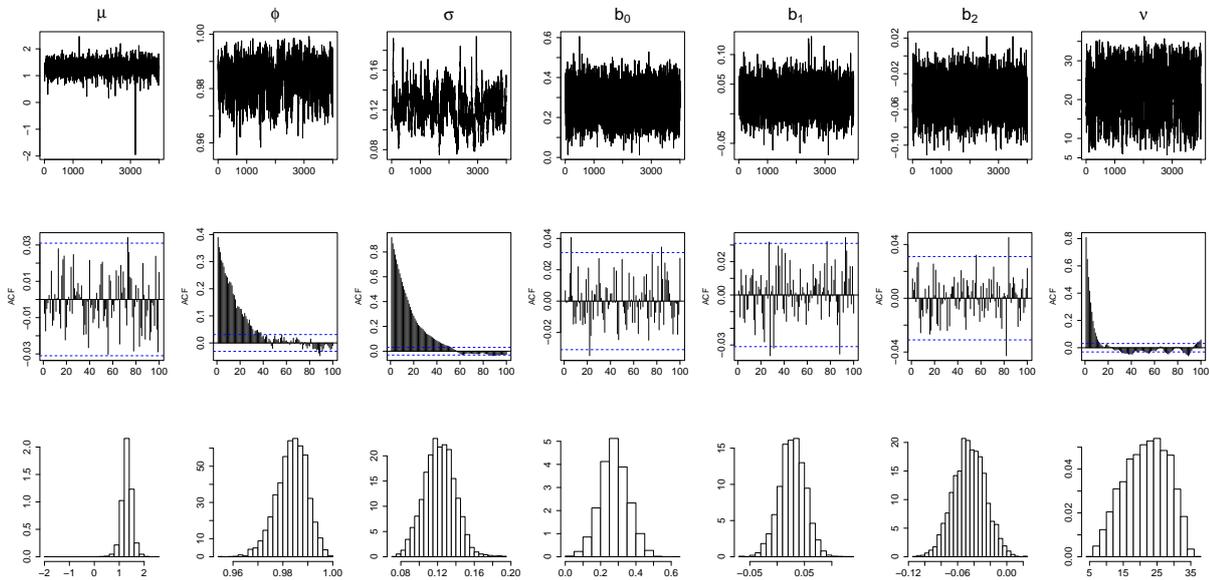
(b) Resultados estimados para o modelo SVM-t.

Figura 7 – Resultados estimados para os modelos SVM-N e SVM-t para os índices IBOVESPA. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.

As estimativas para μ , que representa o nível das log-volatilidades h_t , foram encontradas próximas de 1, para a série IBOVESPA, e próximas de -1, para a serie £/USD. Nas duas aplicações, este parâmetro é estatisticamente significativas em todos os modelos. Os valores estimados para ϕ são consideravelmente altos, indicando uma persistência significativa nas volatilidades.



(a) Resultados estimados para o modelo SVM-S.

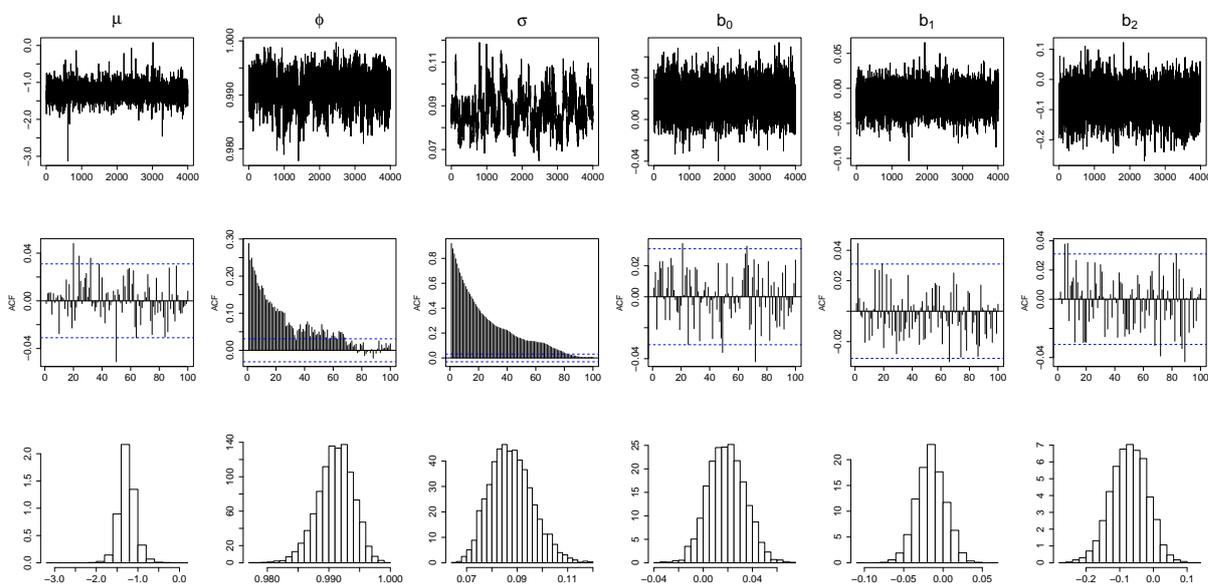


(b) Resultados estimados para o modelo SVM-VG.

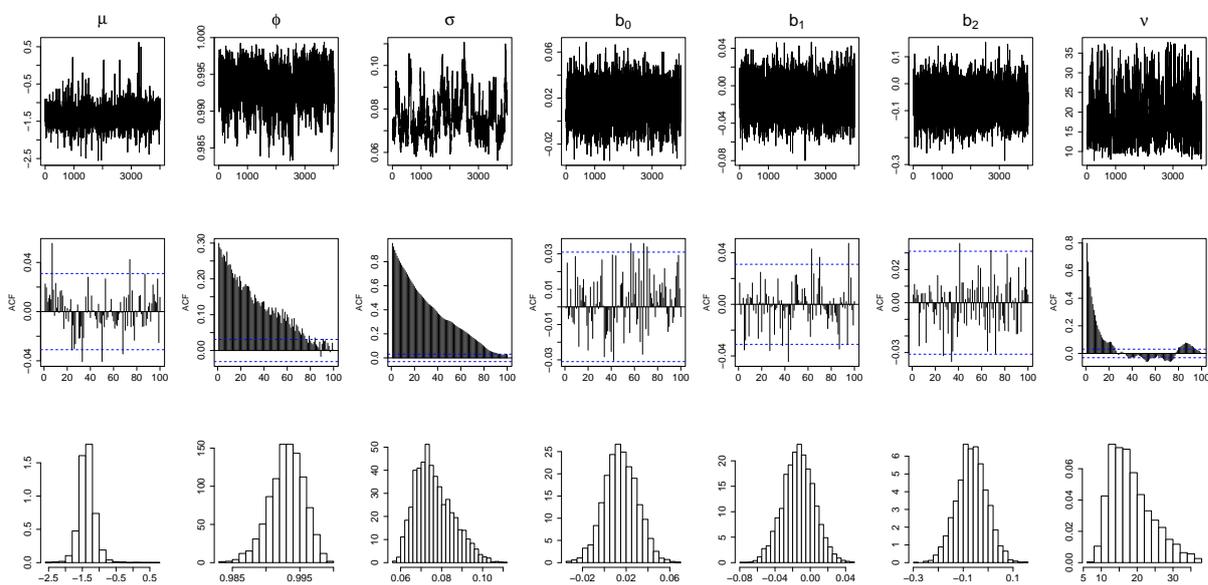
Figura 8 – Resultados estimados para os modelos SVM-S e SVM-VG para os índices IBOVESPA. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas.

É relevante observar que as médias a posteriori estimadas para σ são ligeiramente

menores nos modelos SVM-t, SVM-S e SVM-VG em comparação com o modelo SVM-N. Isso sugere que os modelos que incorporam caudas pesadas são capazes de explicar o excesso de curtose inflacionando os erros observacionais, o que, por sua vez, reduz a variabilidade do processo de volatilidade.



(a) Resultados estimados para o modelo SVM-N.

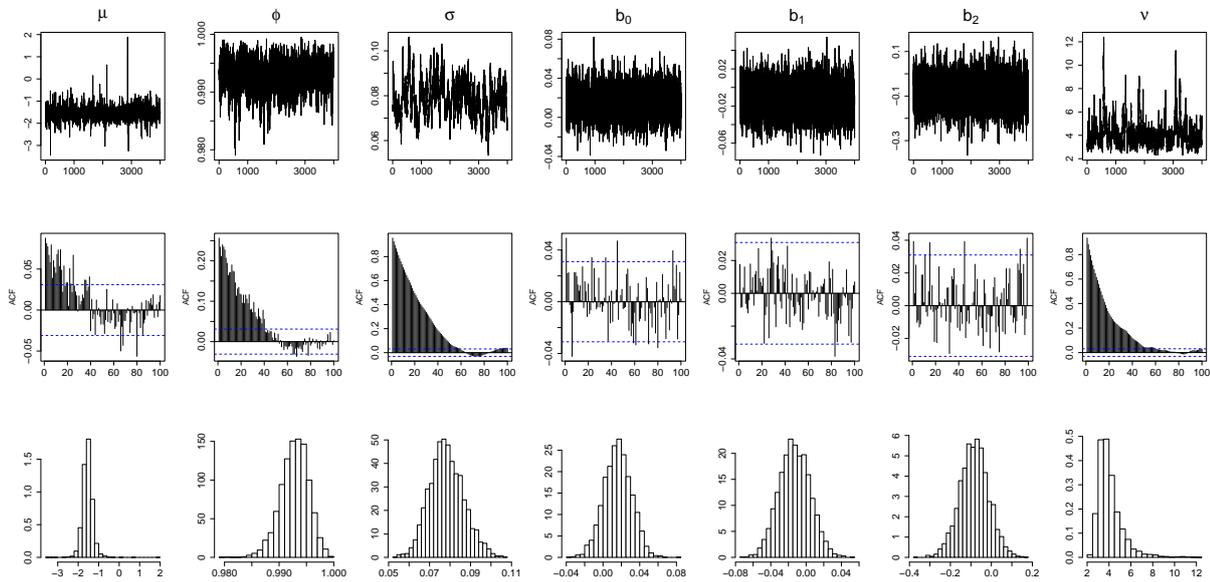


(b) Resultados estimados para o modelo SVM-t.

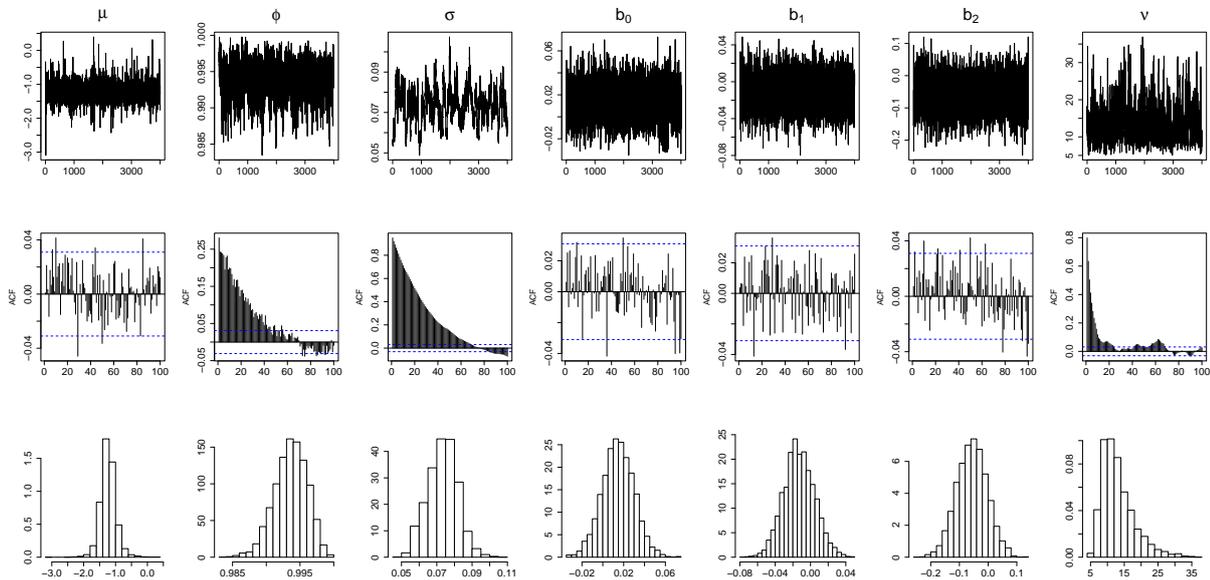
Figura 9 – Resultados estimados para os modelos SVM-N e SVM-t para os retornos £/USD. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas a posteriori.

As Figuras 7 e 8 mostram os resultados obtidos na aplicação IBOVESPA, enquanto as Figuras 9 e 10 mostram os resultados obtidos na aplicação £/USD. A Figura 7 mostra o gráfico

dos traços, autocorrelações e histogramas das distribuições marginais a posteriori simuladas para os modelos SVM-N e SVM-t. Podemos notar uma baixa estrutura de autocorrelação, exceto para os parâmetros ϕ e σ .



(a) Resultados estimados para o modelo SVM-S.



(b) Resultados estimados para o modelo SVM-VG.

Figura 10 – Resultados estimados para os modelos SVM-S e SVM-VG para os retornos £/USD. A primeira linha mostra o traço das cadeias geradas e a segunda linha traz as autocorrelações das amostras. A Terceira linha mostra os histogramas a posteriori.

De maneira semelhante, a Figura 8 apresenta os resultados MCMC para os modelos SVM-S e SVM-VG. Podemos notar um padrão similar aos modelos anteriores, com baixa autocorrelação nas estimativas, exceto para os parâmetros ϕ e σ . Cabe notar, que as autocorrelações

apresentam um rápido decaimento para o parâmetro ν em todos os modelos que capturam caudas pesadas.

A Figura 9 mostra o gráfico dos traços, autocorrelações e histogramas das distribuições marginais a posteriori simuladas para os modelos SVM-N e SVM-t. Podemos notar uma baixa estrutura de autocorrelação, exceto para os parâmetros ϕ e σ . De maneira semelhante, a Figura 10 apresenta os resultados MCMC para os modelos SVM-S e SVM-VG. Podemos notar um padrão similar aos modelos anteriores, com baixa autocorrelação nas estimativas, exceto para os parâmetros ϕ e σ . Cabe notar, que as autocorrelações apresentam um rápido decaimento para o parâmetro ν em todos os modelos que capturam caudas pesadas.

As Figuras 11-18, exibem as volatilidades suavizadas estimadas $\exp(\hat{h}_t/2)$ (representadas pela linha preta) sobrepostas aos retornos absolutos em porcentagem (linha cinza). As figuras incluem também a região de 95% de credibilidade (área sombreada) das estimativas. Observa-se que as volatilidades estimadas $\exp(\hat{h}_t/2)$ seguem de perto as séries de ambas aplicações. Além disso, percebe-se que os modelos SVM-t, SVM-S e SVM-VG exibem movimentos mais suavizados em comparação com o modelo SVM-N, sendo essa diferença especialmente evidente em períodos de retornos extremos.

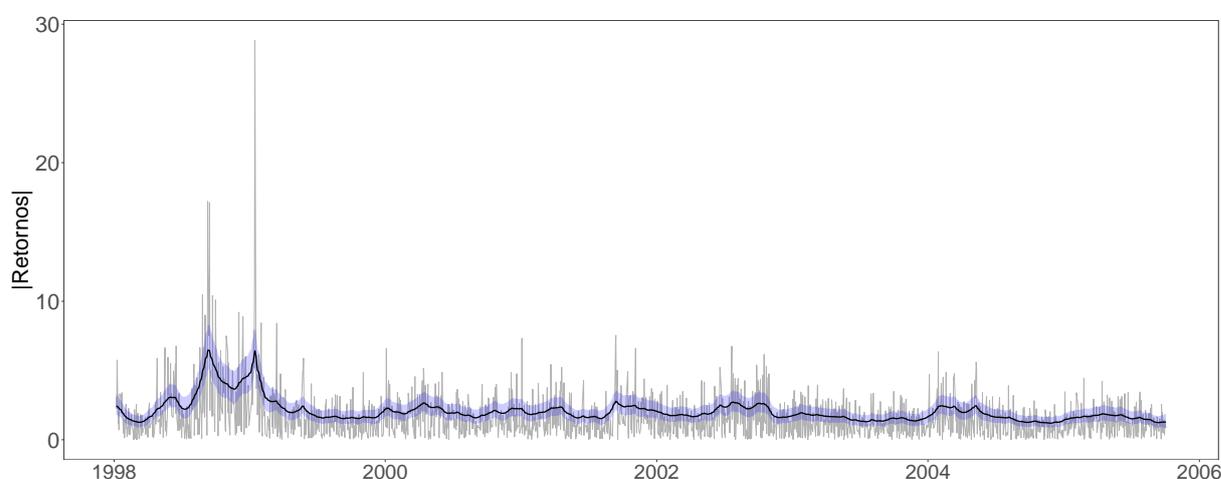


Figura 11 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (área sombreada representa a região com 95% de credibilidade) para SVM-N dos retornos IBOVESPA.

Os modelos que levam em consideração caudas pesadas conseguem lidar com possíveis *outliers* inflacionando a volatilidade $\exp(h_t/2)$ por meio da variável de mistura λ_t . Em outras palavras, a variância da inovação observacional é expressa como $\lambda_t^{-1}\exp(h_t/2)$. Portanto, pequenos valores para λ_t podem estar associados a observações extremas. Este fato pode ser observado pela Figuras 19 e 20, que mostram as médias a posteriori estimadas para as variáveis de mistura, λ_t , dos modelos SVM-t, SVM-S e SVM-VG.

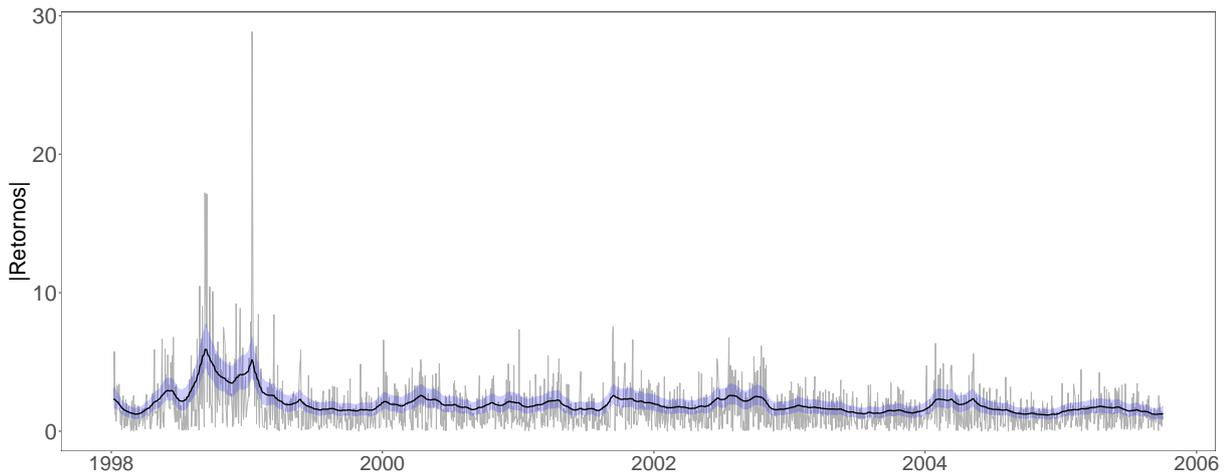


Figura 12 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-t dos retornos IBOVESPA.

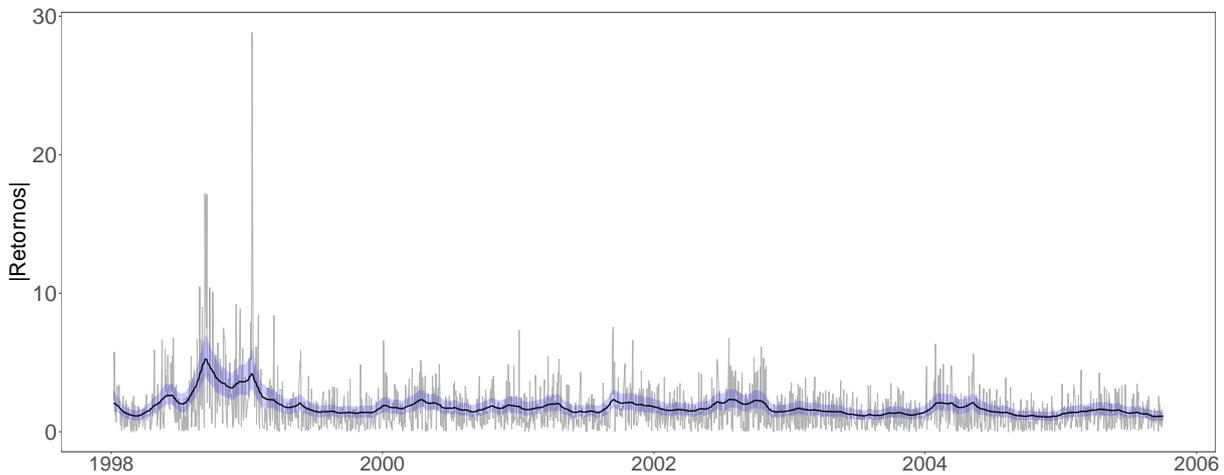


Figura 13 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-S dos retornos IBOVESPA.

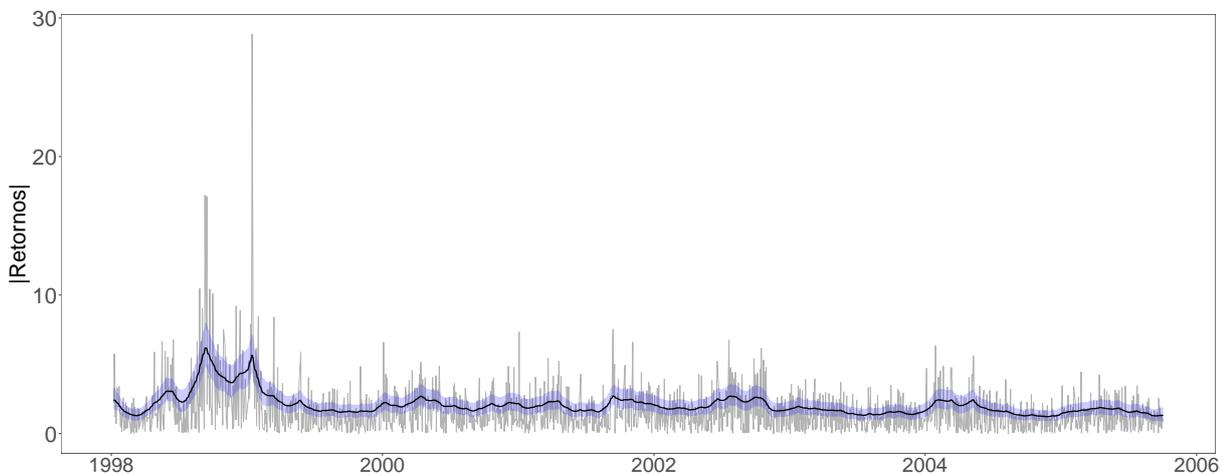


Figura 14 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-VG dos retornos IBOVESPA.

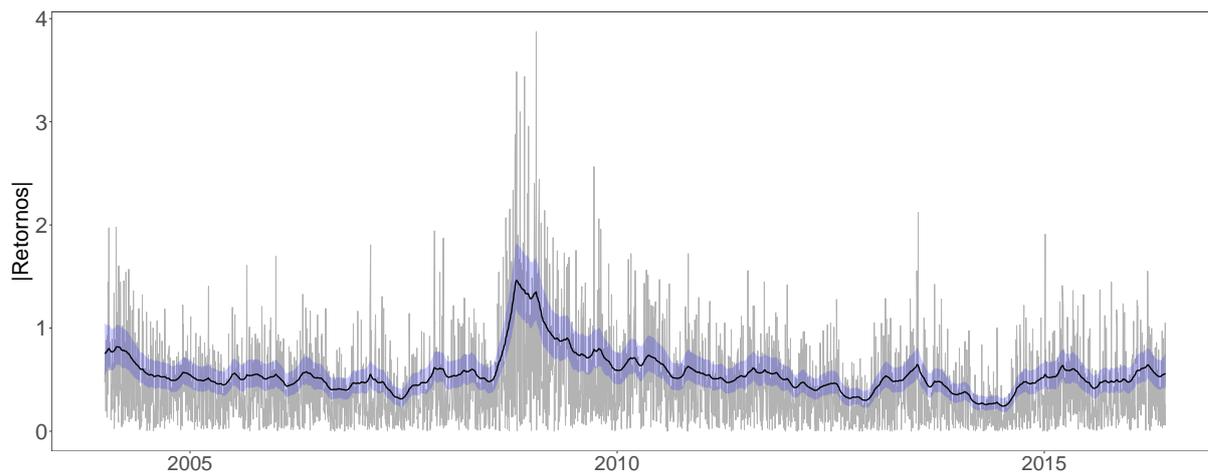


Figura 15 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (área sombreada representa a região com 95% de credibilidade) para SVM-N dos retornos £/USD.

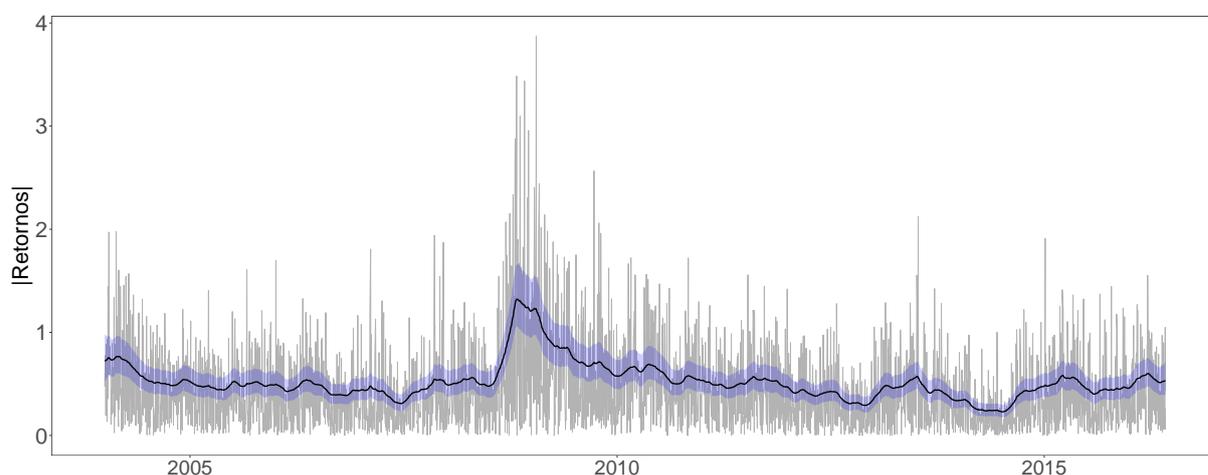


Figura 16 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-t dos retornos £/USD.

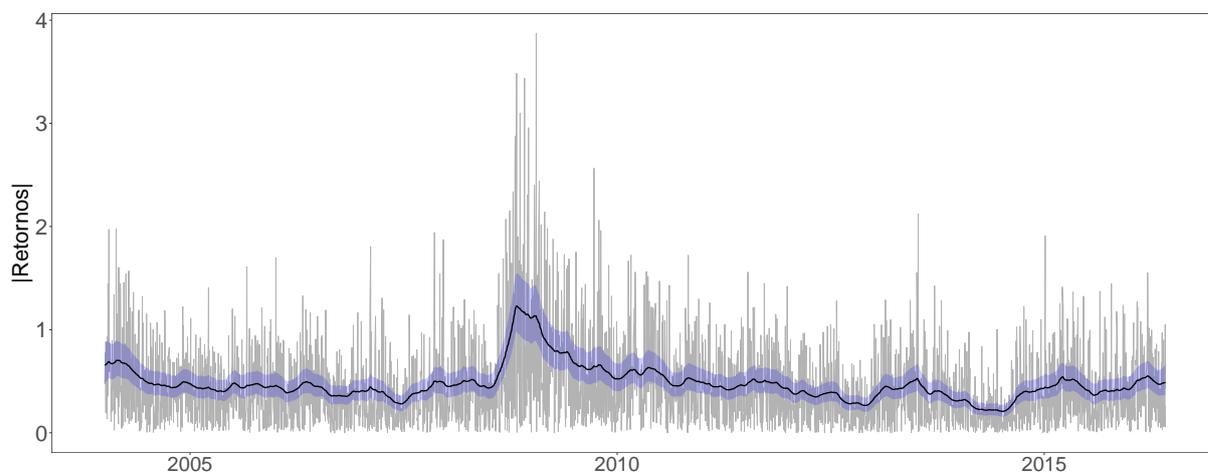


Figura 17 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-S dos retornos £/USD.

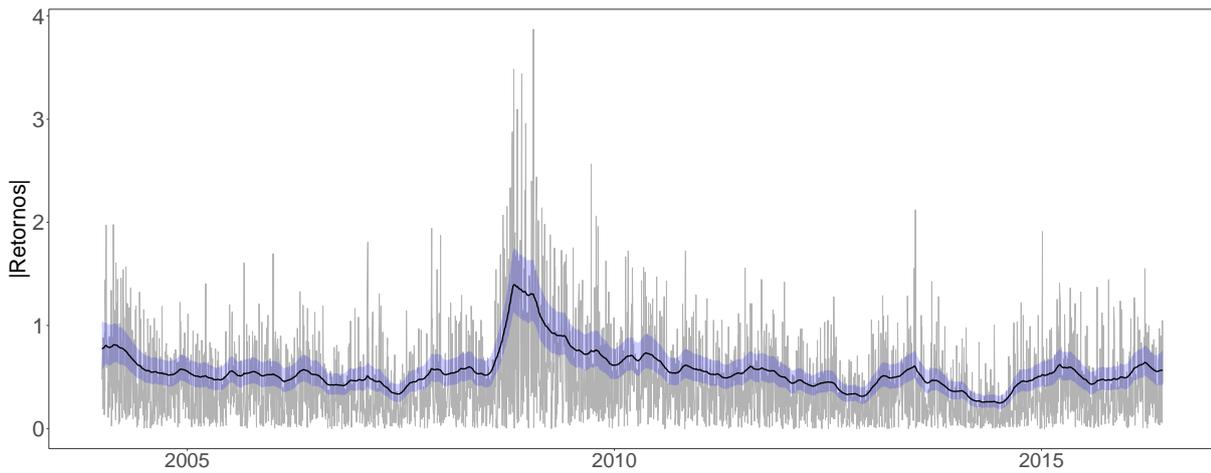


Figura 18 – Médias a posteriori suavizadas das volatilidades $e^{h_t/2}$ estimadas (a área sombreada representa a região com 95% de credibilidade) para SVM-VG dos retornos £/USD.

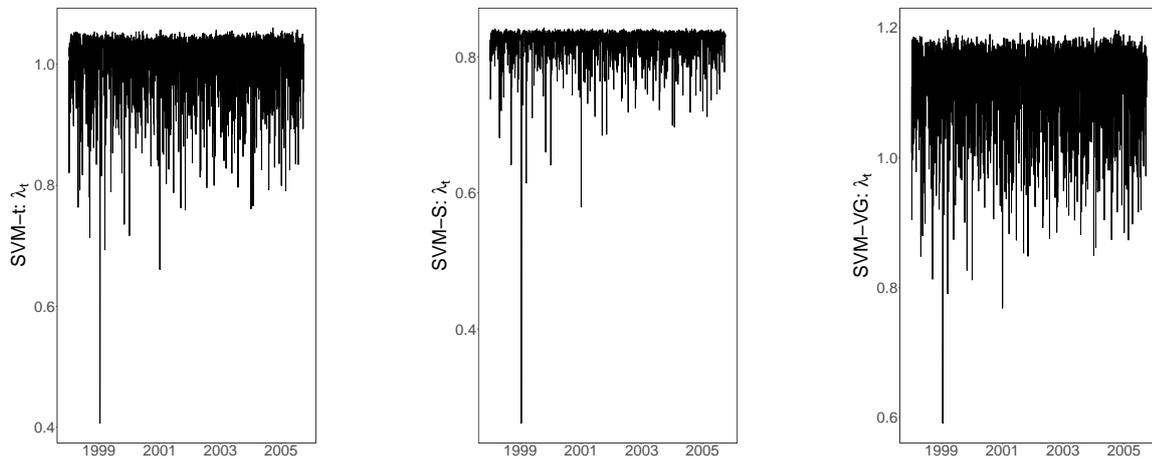


Figura 19 – Comparação das estimativas da variável de mistura λ_t para os modelos SVM-t, SVM-S e SVM-VG dos retornos IBOVESPA.

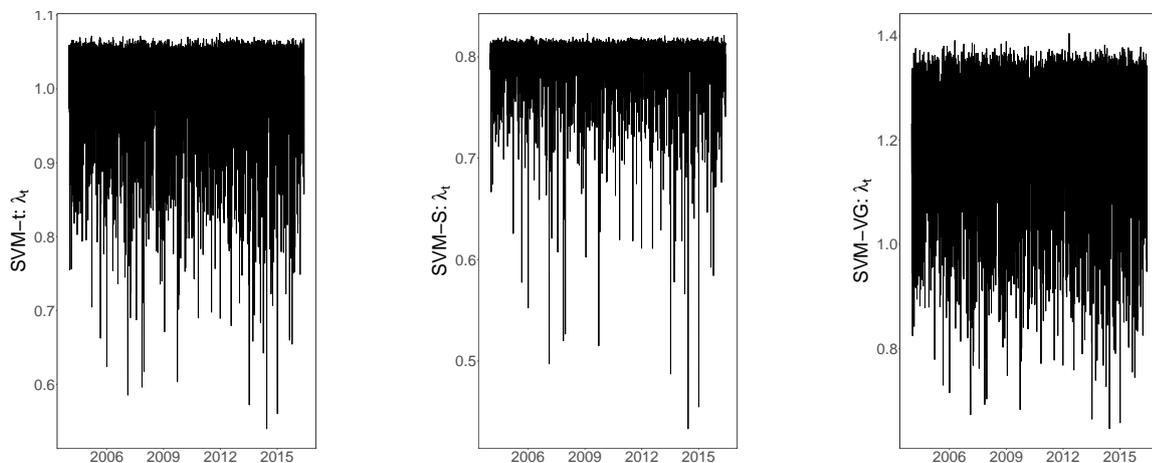


Figura 20 – Comparação das estimativas da variável de mistura λ_t para os modelos SVM-t, SVM-S e SVM-VG dos retornos £/USD.

Para a avaliação da qualidade dos modelos ajustados, utilizamos os critérios de informação DIC, WAIC e LOO-CV. Conforme demonstrado na Tabela 7, constatamos que o modelo SVM-S apresenta um ajuste superior aos dados IBOVESPA, enquanto o modelo SVM-VG se destaca com os dados £/USD. Esses resultados sugerem, em particular, que ambas as séries analisadas se afastam da suposição de normalidade.

Tabela 7 – Critérios de informação calculados.

Aplicação	Critério	SVM-N	SVM-t	SVM-S	SVM-VG
IBOVESPA	DIC	8057.41	8059.73	8039.80	8084.64
	WAIC	8067.77	8049.19	8052.08	8051.35
	LOO-CV	8074.67	8070.86	8069.71	8078.33
£/USD	DIC	5073.837	5053.829	5031.766	5116.144
	WAIC	5082.683	5041.932	5068.982	5012.633
	LOO-CV	5088.413	5073.225	5079.261	5064.002

CONCLUSÕES

Nesta dissertação, apresentamos uma implementação para estimação Bayesiana dos modelos de volatilidade estocástica na média com erros observacionais pertencentes a família mistura de escala normal (SVM-SMN) utilizando o método de Monte Carlo Hamiltoniano e uma de suas modificações, denominada método de Monte Carlo em Variedade Riemanniana. Estes modelos nos permite estudar a relação entre retornos financeiros e suas volatilidades na presença de observações extremas (*outliers*).

O método de estimação proposto, foi ilustrado por meio de dados simulados e algumas propriedades estatísticas relevantes foram avaliadas. Realizamos também uma aplicação a dados reais considerando os índices IBOVESPA. O primeiro estudo de simulação mostrou que o método proposto é capaz de encontrar os valores de interesse de modo satisfatório. O segundo estudo mostrou que o viés e o raiz do erro quadrático médio relativos tendem a zero ao passo que mais informação se torna disponível, indicando um desempenho robusto dos estimadores.

Na aplicação empírica, foram encontrados altos valores de ϕ , indicando alta persistência das volatilidades. As estimativas para b_2 , que mede a relação entre retornos e volatilidade, bem como o efeito *feedback*, foram encontradas negativas. Estes resultados estão de acordo com a hipótese de que investidores que esperam altos níveis de persistência das volatilidades futuras, exigem uma compensação na forma de maiores retornos. Outro fato estilizado que pôde ser encontrado foi o de caudas pesadas, uma vez que os critério de informação utilizados selecionaram o modelo SVM-S, mostrando que a suposição de normalidade não é adequada.

Em futuras pesquisas, pretendemos estender a classe SVM-SMN em várias direções. Isso incluirá a incorporação de erros correlacionados para medir o efeito de alavancagem, a exploração de distribuições assimétricas por meio da família de mistura de escala normal assimétrica (SMSN) e a possibilidade de permitir variações temporais nos parâmetros da média. Além disso, para a seleção de modelos planejamos utilizar o critério de informação recentemente desenvolvido *leave-future-out cross-validation* (LFO-CV), visto que ele se mostra mais apropriado no contexto

de séries temporais.

REFERÊNCIAS

- ABANTO-VALLE, C. A.; BANDYOPADHYAY, D.; LACHOS, V. H.; ENRIQUEZ, I. Robust bayesian analysis of heavy-tailed stochastic volatility models using scale mixtures of normal distributions. **Computational Statistics & Data Analysis**, Elsevier, v. 54, n. 12, p. 2883–2898, 2010. Citado na página [54](#).
- ABANTO-VALLE, C. A.; GARRAFA-ARAGÓN, H. B. Threshold stochastic volatility models with heavy tails: A bayesian approach. **Economía**, v. 42, n. 83, p. 32–53, 2019. Citado nas páginas [54](#) e [55](#).
- ABANTO-VALLE, C. A.; LANGROCK, R.; CHEN, M.-H.; CARDOSO, M. V. Maximum likelihood estimation for stochastic volatility in mean models with heavy-tailed distributions. **Applied Stochastic Models in Business and Industry**, Wiley Online Library, v. 33, n. 4, p. 394–408, 2017. Citado na página [62](#).
- ABANTO-VALLE, C. A.; MIGON, H. S.; LACHOS, V. H. Stochastic volatility in mean models with heavy-tailed distributions. 2012. Citado nas páginas [23](#) e [65](#).
- ABANTO-VALLE, C. A.; RODRÍGUEZ, G.; GARRAFA-ARAGÓN, H. B. Stochastic volatility in mean: Empirical evidence from latin-american stock markets using hamiltonian monte carlo and riemann manifold hmc methods. **The Quarterly Review of Economics and Finance**, Elsevier, v. 80, p. 272–286, 2021. Citado nas páginas [23](#) e [47](#).
- ALDER, B. J.; WAINWRIGHT, T. E. Studies in molecular dynamics. i. general method. **The Journal of Chemical Physics**, American Institute of Physics, v. 31, n. 2, p. 459–466, 1959. Citado na página [37](#).
- AMARI, S.-i.; NAGAOKA, H. **Methods of information geometry**. [S.l.]: American Mathematical Soc., 2000. v. 191. Citado na página [43](#).
- ANDREWS, D. F.; MALLOWS, C. L. Scale mixtures of normal distributions. **Journal of the Royal Statistical Society: Series B (Methodological)**, Wiley Online Library, v. 36, n. 1, p. 99–102, 1974. Citado na página [26](#).
- BETANCOURT, M. A conceptual introduction to hamiltonian monte carlo. **arXiv preprint arXiv:1701.02434**, 2017. Citado nas páginas [39](#) e [41](#).
- BOLLERSLEV, T. Generalized autoregressive conditional heteroskedasticity. **Journal of econometrics**, Elsevier, v. 31, n. 3, p. 307–327, 1986. Citado na página [21](#).
- BROTO, C.; RUIZ, E. Estimation methods for stochastic volatility models: a survey. **Journal of Economic surveys**, Wiley Online Library, v. 18, n. 5, p. 613–649, 2004. Citado na página [22](#).
- CARNERO, A.; PENA, D.; RUIZ, E. Persistence and curtosis in autoregressive stochastic volatility models (manuscript, universidad carlos iii de madrid). 2003. Citado na página [45](#).
- CHIB, S.; GREENBERG, E. Understanding the metropolis-hastings algorithm. **The american statistician**, Taylor & Francis, v. 49, n. 4, p. 327–335, 1995. Citado nas páginas [30](#) e [31](#).

- DUANE, S.; KENNEDY, A. D.; PENDLETON, B. J.; ROWETH, D. Hybrid monte carlo. **Physics letters B**, Elsevier, v. 195, n. 2, p. 216–222, 1987. Citado na página 37.
- EDDELBUETTEL, D.; FRANÇOIS, R. Rcpp: Seamless r and c++ integration. **Journal of statistical software**, v. 40, p. 1–18, 2011. Citado na página 57.
- EHLERS, R. S. Análise de séries temporais. **Laboratório de Estatística e Geoinformação. Universidade Federal do Paraná**, v. 1, p. 1–118, 2007. Citado na página 21.
- ENGLE, R. F. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. **Econometrica: Journal of the econometric society**, JSTOR, p. 987–1007, 1982. Citado na página 21.
- FISCHER, A.; GAUNT, R. E.; SARANTSEV, A. The variance-gamma distribution: A review. **arXiv preprint arXiv:2303.05615**, 2023. Citado na página 27.
- FRENCH, K. R.; SCHWERT, G. W.; STAMBAUGH, R. F. Expected stock returns and volatility. **Journal of financial Economics**, Elsevier, v. 19, n. 1, p. 3–29, 1987. Citado na página 22.
- GAMERMAN, D.; LOPES, H. F. **Markov chain Monte Carlo: stochastic simulation for Bayesian inference**. [S.l.]: CRC press, 2006. Citado na página 28.
- GELFAND, A. E.; SMITH, A. F. Sampling-based approaches to calculating marginal densities. **Journal of the American statistical association**, Taylor & Francis, v. 85, n. 410, p. 398–409, 1990. Citado nas páginas 31 e 35.
- GELMAN, A.; CARLIN, J. B.; STERN, H. S.; RUBIN, D. B. **Bayesian data analysis**. [S.l.]: Chapman and Hall/CRC, 1995. Citado na página 34.
- GEMAN, S.; GEMAN, D. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. **IEEE Transactions on pattern analysis and machine intelligence**, IEEE, n. 6, p. 721–741, 1984. Citado nas páginas 31 e 32.
- GEWEKE, J. Evaluating the accuracy of sampling-based approaches to the calculations of posterior moments. **Bayesian statistics**, Clarendon Press, v. 4, p. 641–649, 1992. Citado nas páginas 33 e 58.
- GIROLAMI, M.; CALDERHEAD, B. Riemann manifold langevin and hamiltonian monte carlo methods. **Journal of the Royal Statistical Society Series B: Statistical Methodology**, Oxford University Press, v. 73, n. 2, p. 123–214, 2011. Citado nas páginas 40, 41, 42 e 62.
- HASTINGS, W. K. Monte carlo sampling methods using markov chains and their applications. Oxford University Press, 1970. Citado na página 30.
- JACQUIER, E.; POLSON, N. G.; ROSSI, P. E. Bayesian analysis of stochastic volatility models. **Journal of Business & Economic Statistics**, Taylor & Francis, v. 20, n. 1, p. 69–87, 1994. Citado na página 22.
- KOOPMAN, S. J.; USPENSKY, E. H. The stochastic volatility in mean model: empirical evidence from international stock markets. **Journal of applied Econometrics**, Wiley Online Library, v. 17, n. 6, p. 667–689, 2002. Citado nas páginas 22 e 23.

- LEÃO, W. L.; ABANTO-VALLE, C. A.; CHEN, M.-H. Bayesian analysis of stochastic volatility-in-mean model with leverage and asymmetrically heavy-tailed error using generalized hyperbolic skew student's t-distribution. **Statistics and its Interface**, NIH Public Access, v. 10, p. 529, 2017. Citado na página 62.
- LEIMKUHLER, B.; REICH, S. **Simulating hamiltonian dynamics**. [S.l.]: Cambridge university press, 2004. Citado nas páginas 38, 39 e 42.
- LEYDOLD, J.; HORMANN, W.; LEYDOLD, M. J. Package 'gigrvg'. **R Package Documentation**, 2017. Citado na página 53.
- MELINO, A.; TURNBULL, S. M. Pricing foreign currency options with stochastic volatility. **Journal of econometrics**, Elsevier, v. 45, n. 1-2, p. 239–265, 1990. Citado na página 22.
- METROPOLIS, N.; ROSENBLUTH, A. W.; ROSENBLUTH, M. N.; TELLER, A. H.; TELLER, E. Equation of state calculations by fast computing machines. **The journal of chemical physics**, American Institute of Physics, v. 21, n. 6, p. 1087–1092, 1953. Citado nas páginas 30 e 37.
- MORETTIN, P. A. **Econometria financeira: um curso em séries temporais financeiras**. 2nd. ed. [S.l.]: Editora Blucher, 2011. Citado nas páginas 22, 25 e 26.
- MORETTIN, P. A.; TOLOI, C. M. C. **Análise de séries temporais**. 2nd. ed. [S.l.]: Editora Blucher, 2006. Citado nas páginas 21 e 45.
- NADARAJAH, S.; KOTZ, S. R programs for truncated distributions. **Journal of Statistical Software**, v. 16, p. 1–8, 2006. Citado na página 53.
- NEAL, R. M. **Bayesian Learning for Neural Networks**. [S.l.]: Springer-Verlag., 1996. Citado na página 37.
- NEAL, R. M. *et al.* Mcmc using hamiltonian dynamics. **Handbook of markov chain monte carlo**, Chapman and Hall/CRC, v. 2, n. 11, p. 2, 2011. Citado nas páginas 33, 38, 39 e 41.
- NUGROHO, D. B.; MORIMOTO, T. Estimation of realized stochastic volatility models using hamiltonian monte carlo-based methods. **Computational Statistics**, Springer, v. 30, p. 491–516, 2015. Citado na página 41.
- PITT, M. K.; MALIK, S.; DOUCET, A. Simulated likelihood inference for stochastic volatility models using continuous particle filtering. **Annals of the Institute of Statistical Mathematics**, Springer, v. 66, p. 527–552, 2014. Citado na página 22.
- PLUMMER, M.; BEST, N.; COWLES, K.; VINES, K. Package 'coda'. URL <http://cran.r-project.org/web/packages/coda/coda.pdf>, accessed January, v. 25, p. 2015, 2015. Citado na página 58.
- R Core Team. **R: A Language and Environment for Statistical Computing**. Vienna, Austria, 2022. Disponível em: [<https://www.R-project.org/>](https://www.R-project.org/). Citado na página 53.
- ROBERT, C.; CASELLA, G. **Monte Carlo Statistical Methods**. 2nd. ed. [S.l.]: Springer, 2004. (Springer Texts in Statistics). ISBN 0387212396; 9780387212395. Citado na página 28.
- RUIZ, E. Quasi-maximum likelihood estimation of stochastic volatility models. **Journal of econometrics**, Elsevier, v. 63, n. 1, p. 289–306, 1994. Citado na página 22.

SEBER, G. A.; LEE, A. J. **Linear regression analysis**. [S.l.]: John Wiley & Sons, 2012. Citado na página 43.

SHEPHARD, N. Fitting nonlinear time-series models with applications to stochastic variance models. **Journal of Applied Econometrics**, Wiley Online Library, v. 8, n. S1, p. S135–S152, 1993. Citado na página 22.

SPIEGELHALTER, D. J.; BEST, N. G.; CARLIN, B. P.; LINDE, A. V. D. Bayesian measures of model complexity and fit. **Journal of the royal statistical society: Series b (statistical methodology)**, Wiley Online Library, v. 64, n. 4, p. 583–639, 2002. Citado nas páginas 33 e 34.

STAN DEVELOPMENT TEAM. **RStan: the R interface to Stan**. 2022. Versão 2.30. Disponível em: <<https://mc-stan.org>>. Citado na página 35.

TAYLOR, S. J. **Modelling financial time series**. 1nd. ed. [S.l.]: world scientific, 1986. Citado nas páginas 21 e 45.

THOMAS, S.; TU, W. Learning hamiltonian monte carlo in r. **The American Statistician**, Taylor & Francis, v. 75, n. 4, p. 403–413, 2021. Citado na página 41.

TSAY, R. S. **Analysis of financial time series**. 3nd. ed. [S.l.]: John wiley & sons, 2010. Citado na página 26.

VEHTARI, A.; GELMAN, A.; GABRY, J. Efficient implementation of leave-one-out cross-validation and waic for evaluating fitted bayesian models. **arXiv preprint arXiv:1507.04544**, Technical report, 2015. Citado na página 35.

VEHTARI, A.; GELMAN, A.; GABRY, J.; YAO, Y.; BÜRKNER, P.-C.; GOODRICH, B.; PIIRONEN, J.; MAGNUSSON, M. Loo: Efficient leave-one-out cross-validation and waic for bayesian models (2019). **R package version**, v. 2, n. 0, 2022. Citado na página 35.

WATANABE, S.; OPPER, M. Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. **Journal of machine learning research**, v. 11, n. 12, 2010. Citado na página 34.

ZEVALLOS, M.; GASCO, L.; EHLERS, R. Riemann manifold langevin methods on stochastic volatility estimation. **Communications in Statistics-Simulation and Computation**, Taylor & Francis, v. 46, n. 10, p. 7942–7956, 2017. Citado na página 62.

DEMONSTRAÇÕES

Definição 6. A matriz simétrica A $n \times n$ é dita ser positiva definida se $v'Av > 0$, $\forall v \neq 0 \in \mathbb{R}^n$.

Proposição 2. Sejam $v, u, z \in \mathbb{R}^n$. Então $v'(u + z) = v'u + v'z$.

Demonstração: Basta notar que

$$\begin{aligned} v'(u + z) &= (v_1, \dots, v_n) \begin{pmatrix} u_1 + z_1 \\ \dots \\ u_n + z_n \end{pmatrix} \\ &= \sum_{i=1}^n v_i(u_i + z_i) \\ &= \left(\sum_{i=1}^n v_i u_i \right) + \left(\sum_{i=1}^n v_i z_i \right) \\ &= v'u + v'z. \end{aligned}$$

Proposição 3. Sejam A e B matrizes $n \times n$ positivas definidas. Então $A + B$ também é uma matriz $n \times n$ positiva definida.

Demonstração: Seja $v \neq 0$ um vetor em \mathbb{R}^n . Logo,

$$\begin{aligned} v'(A + B)v &= v'(Av + Bv) \\ &= v'Av + v'Bv, \text{ pela Proposição 2} \\ &> 0. \end{aligned}$$

Portanto $A + B$ é definida positiva, pela Definição 6.

Proposição 4. O erro quadrático médio do estimador $\hat{\theta}$ pode ser escrito em termos de sua variância e seu viés, já que

$$\begin{aligned}
 EQM(\theta) &= E\{(\hat{\theta} - \theta)^2\} \\
 &= E\{(\hat{\theta} - E(\hat{\theta}) + E(\hat{\theta}) - \theta)^2\} \\
 &= E\{(\hat{\theta} - E(\hat{\theta}))^2\} + 2E\{\hat{\theta} - E(\hat{\theta})\}E\{E(\hat{\theta}) - \theta\} + E\{(E(\hat{\theta}) - \theta)^2\} \\
 &= E\{(\hat{\theta} - E(\hat{\theta}))^2\} + E\{(E(\hat{\theta}) - \theta)^2\} \\
 &= \text{Var}(\hat{\theta}) + \text{viés}(\hat{\theta})^2,
 \end{aligned}$$

sendo $E(\hat{\theta} - \theta)$ constante e $E[\theta - E(\theta)] = 0$.

Proposição 5. O viés relativo (viés_{rel}) e a raiz do erro quadrático médio relativo (REQM_{rel}) podem ser expressos em termos do viés e raiz do erro quadrático médio, respectivamente, pois

$$\begin{aligned}
 \text{viés}_{rel} &= \frac{1}{m} \sum_{i=1}^m \left(\frac{\hat{\theta}^{(i)} - \theta}{\theta} \right) \\
 &= \frac{1}{\theta} \left\{ \frac{1}{m} \sum_{i=1}^m (\hat{\theta}^{(i)} - \theta) \right\} \\
 &= \frac{1}{\theta} \text{viés},
 \end{aligned}$$

e

$$\begin{aligned}
 \text{REQM}_{rel} &= \sqrt{\frac{1}{m} \sum_{i=1}^m \left(\frac{\hat{\theta}^{(i)} - \theta}{\theta} \right)^2} \\
 &= \sqrt{\frac{1}{\theta^2} \frac{1}{m} \sum_{i=1}^m (\hat{\theta}^{(i)} - \theta)^2} \\
 &= \frac{1}{\sqrt{\theta^2}} \sqrt{\frac{1}{m} \sum_{i=1}^m (\hat{\theta}^{(i)} - \theta)^2} \\
 &= \frac{1}{|\theta|} \text{REQM}.
 \end{aligned}$$

Proposição 6. (Método da Transformação Inversa) Seja U uma variável aleatória uniformemente distribuída no intervalo $(0, 1)$. Para qualquer função distribuição acumulada contínua G , a variável aleatória X , definida por $X = G^{-1}(U)$, tem distribuição G .

Demonstração: Seja G_X a distribuição acumulada de $X = G^{-1}(U)$. Desta forma, $\forall x \sim X$, temos

$$\begin{aligned}
G_X(x) &= P\{X \leq x\} \\
&= P\{G^{-1}(U) \leq x\} \\
&= P\{G(G^{-1}(U)) \leq G(x)\}, \text{ pela monotonicidade da função } G, \\
&= P\{U \leq G(x)\} \\
&= G(x), \text{ uma vez que } U \text{ é uniforme em } (0,1),
\end{aligned}$$

como queríamos mostrar.

Proposição 7. Suponha que X seja uma variável aleatória contínua com função densidade de probabilidade $f(\cdot)$ e função densidade acumulada $F(\cdot)$. Seja Y uma variável aleatória representando a versão truncada de X no intervalo (a, b) , em que $-\infty < a < b < \infty$. Se pudermos calcular $F^{-1}(u), \forall u \in (0, 1)$, então simulações de Y são facilmente obtidas pelo método da Transformação Inversa.

Demonstração: Primeiro note que para $u \in (0, 1)$, temos

$$F_Y^{-1}(u) = F^{-1}(F(a) + u(F(b) - F(a))),$$

em que

$$F_Y(y) = \frac{F(y) - F(a)}{F(b) - F(a)}, \quad y \in (a, b).$$

Portanto, pela Proposição 6 podemos obter uma realização de Y tomando $y = F_Y^{-1}(u)$, em que $u \sim U(0, 1)$.

CÓDIGOS COMPUTACIONAIS

Os códigos computacionais das bibliotecas *svm_svm_n* e *svm_smn*, utilizadas neste trabalho, estão disponíveis no GitHub por meio do link: <https://github.com/holtz27/svmsmn>.

Algoritmo 6 – svm_n.cpp

```

1: // [[Rcpp::depends( RcppArmadillo )]]
2: #include "svm_smn_n.h"
3: // set seed function
4: void set_seed( int seed ){
5:   Rcpp::Environment base_env("package:base");
6:   Rcpp::Function set_seed_r = base_env["set.seed"];
7:   set_seed_r( std::floor( std::fabs( seed ) ) );
8: }
9: // svm_n function
10: List svm_n(int N, int L_theta, double eps_theta, int L_b, double eps_b, int L_h, double
    eps_h, vec y_T, int seed ){
11:   wall_clock timer;
12:   timer.tic();
13:   se seed != 0 então set_seed( seed );
14:   fim se
15:   int T = y_T.n_elem - 1, a = floor( 0.1 * N );
16:   // initial theta
17:   int acc_theta = 0;
18:   vec theta_cur = zeros<vec>(3, 1);
19:   theta_cur[ 0 ] += 0.1;
20:   theta_cur[ 1 ] += 0.5 * ( log( 1 + 0.98 ) - log( 1 - 0.98 ) );
21:   theta_cur[ 2 ] += log( 0.15 );
22:   // initial h
23:   int acc_b = 0;
24:   vec h_cur = zeros<vec>(T, 1);
25:   h_cur[ 0 ] += 0.1 + 0.15 / sqrt(1 - 0.98 * 0.98 ) * randn();

```

```

26: para (int kt = 1 ; kt < T ; kt++) faça h_cur[ kt ] += 0.1 + 0.98 * ( h_cur[ kt - 1 ] - 0.1 ) +
    0.15 * randn();
27: fim para
28: // initial b
29: int acc_h = 0;
30: vec b_cur = zeros<vec>(3, 1);
31: b_cur[ 0 ] += 0.1;
32: b_cur[ 1 ] += 0.5 * ( log( 1 + 0.03 ) - log( 1 - 0.03 ) );
33: b_cur[ 2 ] += -0.10;
34: // state initial chain
35: mat chain_theta = zeros<mat>( 3, N + 1 );
36: chain_theta.col( 0 ) += theta_cur;
37: mat chain_b = zeros<mat>( 3, N + 1 );
38: chain_b.col( 0 ) += b_cur;
39: mat chain_h = zeros<mat>( T, N + 1 );
40: chain_h.col( 0 ) += h_cur;
41: // building the chains
42: para ( int it = 1 ; it < N + 1 ; it ++ ) faça
43:   theta_cur = rmhmc_theta( theta_cur, h_cur, 5, L_theta, eps_theta, T, acc_theta );
44:   b_cur = rmhmc_b( b_cur, h_cur, 5, L_b, eps_b, T, y_T , acc_b );
45:   h_cur = hmc_h( h_cur, theta_cur, b_cur, L_h, eps_h, T, y_T, acc_h );
46:   // chain update
47:   chain_theta.col( it ) += theta_cur;
48:   chain_b.col( it ) += b_cur;
49:   chain_h.col( it ) += h_cur;
50: fim para
51: // Transformations
52: chain_theta.row( 1 ) = tanh( chain_theta.row( 1 ) );
53: chain_theta.row( 2 ) = exp( chain_theta.row( 2 ) );
54: chain_b.row( 1 ) = tanh( chain_b.row( 1 ) );
55: List chain = List::create(
56:   Named("chain_theta") = chain_theta,
57:   Named("chain_b") = chain_b,
58:   Named("chain_h") = chain_h,
59: );
60: vec acc = zeros<vec>(3, 1);
61: acc[ 0 ] += acc_theta;
62: acc[ 1 ] += acc_b;
63: acc[ 2 ] += acc_h;
64: double time = timer.toc();
65: return List::create( Named("chain") = chain, Named("acc") = acc, Named("time") = time );
66: }

```

Algoritmo 7 – svm_t.cpp

```

1: // [[Rcpp::depends( RcppArmadillo )]]
2: #include "svm_smn.h"
3: // sample lambda vector
4: vec l_gibbs(double e, vec y_T, vec h, vec b, int T, double alpha, double li, double ls){
5: double v = 0.5 * ( (ls - li) * tanh( 0.5 * alpha * e ) + (ls + li) );
6: double b0 = b[0], b1 = tanh( b[1] ), b2 = b[2];
7: vec l_out = zeros<vec>(T, 1);
8: vec aux = y_T.subvec( 1, T ) - b0 - b1 * y_T.subvec( 0, T - 1 ) - b2 * exp( h );
9: vec u = exp( - h ) % aux % aux;
10: para ( int i = 0 ; i < T ; i++ ) faça l_out[ i ] = R::rgamma( 0.5 * (v + 1), 2.0 / ( u[ i ] + v ) );
11: fim para
12: return l_out;
13: }
14: // set seed function
15: void set_seed( int seed ){
16: Rcpp::Environment base_env("package:base");
17: Rcpp::Function set_seed_r = base_env["set.seed"];
18: set_seed_r( std::floor( std::fabs( seed ) ) );
19: }
20: // [[Rcpp::export]]
21: List svm_t(int N, int L_theta, double eps_theta, int L_b, double eps_b, int L_h, double eps_h,
    int L_v, double eps_v, double alpha, double li, double ls, vec y_T, int seed ){
22: wall_clock timer;
23: timer.tic();
24: se seed != 0 então set_seed( seed );
25: fim se
26: // initial theta
27: int acc_theta = 0;
28: vec theta_cur = zeros<vec>(3, 1);
29: theta_cur[ 0 ] += 0.1;
30: theta_cur[ 1 ] += 0.5 * ( log( 1 + 0.98 ) - log( 1 - 0.98 ) );
31: theta_cur[ 2 ] += log( 0.15 );
32: // initial h
33: int acc_h = 0;
34: vec h_cur = zeros<vec>(T, 1);
35: h_cur[ 0 ] += 0.1 + 0.15 / sqrt(1 - 0.98 * 0.98 ) * randn();
36: para (int kt = 1 ; kt < T ; kt++) faça h_cur[ kt ] += 0.1 + 0.98 * ( h_cur[ kt - 1 ] - 0.1 ) +
    0.15 * randn();
37: fim para
38: // initial b
39: int acc_b = 0;
40: vec b_cur = zeros<vec>(3, 1);
41: b_cur[ 0 ] += 0.1;
42: b_cur[ 1 ] += 0.5 * ( log( 1 + 0.03 ) - log( 1 - 0.03 ) );
43: b_cur[ 2 ] += -0.10;

```

```

44: // initial v
45: int acc_v = 0;
46: double v_cur = 10.0;
47: // initial lambda
48: vec l_cur = zeros<vec>(T, 1);
49: para (int i = 0; i < T; i++) faça l_cur[ i ] = R::rgamma( 0.5 * v_cur, 2.0 / v_cur );
50: fim para
51: v_cur = (2 / alpha) * atanh((2 * v_cur - ls - li) / (ls - li));
52: // initial chain
53: mat chain_theta = zeros<mat>( 3, N + 1 );
54: chain_theta.col( 0 ) += theta_cur;
55: mat chain_b = zeros<mat>( 3, N + 1 );
56: chain_b.col( 0 ) += b_cur;
57: mat chain_h = zeros<mat>( T, N + 1 );
58: chain_h.col( 0 ) += h_cur;
59: mat chain_v = zeros<mat>( 1, N + 1 );
60: chain_v.col( 0 ) += v_cur;
61: mat chain_l = zeros<mat>( T, N + 1 );
62: chain_l.col( 0 ) += l_cur;
63: // building the chains
64: para ( int it = 1 ; it < N + 1 ; it ++ ) faça
65:   theta_cur = rmhmc_theta( theta_cur, h_cur, 5, L_theta, eps_theta, T, acc_theta );
66:   b_cur = rmhmc_b( b_cur, h_cur, l_cur, 5, L_b, eps_b, T, y_T , acc_b );
67:   h_cur = hmc_h( h_cur, theta_cur, b_cur, l_cur, L_h, eps_h, T, y_T, acc_h );
68:   v_cur = rmhmc_v_vg(v_cur, l_cur, 5, L_v, eps_v, T, acc_v, alpha, li, ls );
69:   l_cur = l_gibbs(v_cur, y_T, h_cur, b_cur, T, alpha, li, ls);
70:   // chain update
71:   chain_theta.col( it ) += theta_cur;
72:   chain_b.col( it ) += b_cur;
73:   chain_h.col( it ) += h_cur;
74:   chain_v.col( it ) += v_cur;
75:   chain_l.col( it ) += l_cur;
76: }
77: fim para
78: // Transformations
79: chain_theta.row( 1 ) = tanh( chain_theta.row( 1 ) );
80: chain_theta.row( 2 ) = exp( chain_theta.row( 2 ) );
81: chain_b.row( 1 ) = tanh( chain_b.row( 1 ) );
82: // v = 0.5 * ( (ls - li) * tanh( 0.5 * alpha * e ) + (ls + li) )
83: chain_v.row( 0 ) = 0.5 * ( (ls - li) * tanh( 0.5 * alpha * chain_v.row( 0 ) )
84: List chain = List::create(
85:   Named("chain_theta") = chain_theta,
86:   Named("chain_b") = chain_b,
87:   Named("chain_h") = chain_h,
88:   Named("chain_v") = chain_v,
89:   Named("chain_l") = chain_l
90: );

```

```

91: vec acc = zeros<vec>(4, 1);
92: acc[ 0 ] += acc_theta;
93: acc[ 1 ] += acc_b;
94: acc[ 2 ] += acc_h;
95: acc[ 3 ] += acc_v;
96: double time = timer.toc();
97: return List::create( Named("chain") = chain, Named("acc") = acc, Named("time") = time );
98: }

```

Algoritmo 8 – svm_s.cpp

```

1: // [[Rcpp::depends( RcppArmadillo )]]
2: #include "svm_smn.h"
3: // sample right truncated gamma
4: double right_tgamma(double max, double shape, double rate){
5: double scale = 1 / rate;
6: se max <= 0 então stop( "argumnto b é menor ou igual a 0!");
7: fim se
8: double rtg, q, log_Gb;
9: se std::isfinite( max ) então log_Gb = R::pgamma( max, shape, scale, true, true );
10: se std::isinf( log_Gb ) então stop( "Instabilidade numérica: log( Gb ) = Inf ");
11: fim se
12: fim se
13: double u = R::runif( 0, 1 );
14: q = log( u ) + log_Gb;
15: rtg = R::qgamma( q, shape, scale, true, true );
16: return rtg;
17: }
18: // sample left truncated gamma
19: double left_tgamma( double min, double shape, double rate ){
20: double scale = 1 / rate;
21: se min <= 0 então stop( "argumnto min tem que ser positivo!");
22: fim se
23: double ltg, q;
24: double u = R::runif( 0, 1 );
25: q = R::pgamma( min, shape, scale, true, false );
26: q += u * R::pgamma( min, shape, scale, false, false );
27: ltg = R::qgamma( q, shape, scale, true, false );
28: return ltg;
29: }
30: // sample lambda vector
31: vec l_gibbs( double v, vec y_T, vec h, vec b, int T ){
32: double b0 = b[0], b1 = tanh(b[1]), b2 = b[2];
33: vec l_out = zeros<vec>(T, 1);
34: vec aux = y_T.subvec( 1, T ) - b0 - b1 * y_T.subvec( 0, T - 1 ) - b2 * exp( h );
35: vec u = 0.5 * exp( - h ) % aux % aux;
36: para (int i = 0 ; i < T ; i++) faça l_out[ i ] = right_tgamma( 1.0, v + 0.5, u[ i ] );
37: fim para

```

```

38: return l_out;
39: }
40: // set seed function
41: void set_seed( int seed ){
42: Rcpp::Environment base_env("package:base");
43: Rcpp::Function set_seed_r = base_env["set.seed"];
44: set_seed_r( std::floor( std::fabs( seed ) ) );
45: }
46: // [[Rcpp::export]]
47: // svm_s function
48: List svm_s(int N, int L_theta, vec eps_theta, int L_b, vec eps_b, int L_h, double eps_h, vec
  y_T, int seed ){
49: wall_clock timer;
50: timer.tic();
51: se seed != 0 então set_seed( seed );
52: fim se
53: int T = y_T.n_elem - 1, a = floor( 0.1 * N );
54: // initial theta
55: int acc_theta = 0;
56: vec theta_cur = zeros<vec>(3, 1);
57: theta_cur[ 0 ] += 0.1;
58: theta_cur[ 1 ] += 0.5 * ( log( 1 + 0.98 ) - log( 1 - 0.98 ) );
59: theta_cur[ 2 ] += log( 0.15 );
60: // initial h
61: int acc_h = 0;
62: vec h_cur = zeros<vec>(T, 1);
63: h_cur[ 0 ] += 0.1 + 0.15 / sqrt(1 - 0.98 * 0.98 ) * randn();
64: para ( int kt = 1 ; kt < T ; kt++ ) faça h_cur[ kt ] += 0.1 + 0.98 * ( h_cur[ kt - 1 ] - 0.1 ) +
  0.15 * randn();
65: fim para
66: // iniciando b
67: int acc_b = 0;
68: vec b_cur = zeros<vec>(3, 1);
69: b_cur[ 0 ] += 0.1;
70: b_cur[ 1 ] += 0.5 * ( log( 1 + 0.03 ) - log( 1 - 0.03 ) );
71: b_cur[ 2 ] += -0.10;
72: // iniciando v
73: double v_cur = 2.0;
74: // iniciando l
75: vec l_cur = zeros<vec>(T, 1);
76: para ( int k = 0 ; k < T ; k++ ) faça l_cur[ k ] = R::rbeta( v_cur, 1.0 );
77: fim para
78: // iniciando cadeia
79: mat chain_theta = zeros<mat>( 3, N + 1 );
80: chain_theta.col( 0 ) += theta_cur;
81: mat chain_b = zeros<mat>( 3, N + 1 );
82: chain_b.col( 0 ) += b_cur;
83: mat chain_h = zeros<mat>( T, N + 1 );
84: chain_h.col( 0 ) += h_cur;

```

```

85: mat chain_v = zeros<mat>( 1, N + 1 );
86: chain_v.col( 0 ) += v_cur;
87: mat chain_l = zeros<mat>( T, N + 1 );
88: chain_l.col( 0 ) += l_cur;
89: // building the chains
90: para ( int it = 1 ; it < N + 1 ; it ++ ) faça
91:   theta_cur = rmhmc_theta( theta_cur, h_cur, 5, L_theta, eps_theta, T, acc_theta );
92:   b_cur = rmhmc_b( b_cur, h_cur, l_cur, 5, L_b, eps_b, T, y_T , acc_b );
93:   h_cur = hmc_h( h_cur, theta_cur, b_cur, l_cur, L_h, eps_h, T, y_T, acc_h );
94:   v_cur = left_tgamma( 1.0, T + 0.08, 0.04 - sum( log(l_cur) ) );
95:   l_cur = l_gibbs( v_cur, y_T, h_cur, b_cur, T );
96:   // chain update
97:   chain_theta.col( it ) += theta_cur;
98:   chain_b.col( it ) += b_cur;
99:   chain_h.col( it ) += h_cur;
100:   chain_v.col( it ) += v_cur;
101:   chain_l.col( it ) += l_cur;
102: fim para
103: // Transformations
104: chain_theta.row( 1 ) = tanh( chain_theta.row( 1 ) );
105: chain_theta.row( 2 ) = exp( chain_theta.row( 2 ) );
106: chain_b.row( 1 ) = tanh( chain_b.row( 1 ) );
107: List chain = List::create(
108:   Named("chain_theta") = chain_theta,
109:   Named("chain_b") = chain_b,
110:   Named("chain_h") = chain_h,
111:   Named("chain_v") = chain_v,
112:   Named("chain_l") = chain_l
113: );
114: vec acc = zeros<vec>(3, 1);
115: acc[ 0 ] += acc_theta;
116: acc[ 1 ] += acc_b;
117: acc[ 2 ] += acc_h;
118: double time = timer.toc();
119: return List::create( Named("chain") = chain, Named("acc") = acc, Named("time") = time );
120: }

```

Algoritmo 9 – svm_vg.cpp

```

1: // [[Rcpp::depends( RcppArmadillo )]]
2: #include "svm_smn.h"
3: // sample lambda vector
4: vec l_gibbs(double e, vec y_T, vec h, vec b, int T, double alpha, double li, double ls){
5: Environment pkg = Environment::namespace_env("GIGrvg");
6: Function f = pkg["rgig"];
7: double v = 0.5 * ( (ls - li) * tanh( 0.5 * alpha * e ) + (ls + li) );
8: double b0 = b[0], b1 = tanh( b[1] ), b2 = b[2];

```

```

9: NumericVector x;
10: vec l_out( T );
11: vec mu_t = y_T.subvec(1, T) - b0 - b1 * y_T.subvec(0, T - 1) - b2 * exp( h );
12: vec psi = exp( -h ) % mu_t % mu_t;
13: para (int i = 0 ; i < T; i++) faça
14:   x = f(Named("n") = 1, Named("lambda") = 0.5 * (1 - v), _["chi"] = v, _["psi"] = psi[ i ] );
15:   l_out[ i ] = x[ 0 ];
16: fim para
17: return l_out;
18: }
19: // set seed function
20: void set_seed( int seed ){
21:   Rcpp::Environment base_env("package:base");
22:   Rcpp::Function set_seed_r = base_env["set.seed"];
23:   set_seed_r( std::floor( std::fabs( seed ) ) );
24: }
25: // [[Rcpp::export]]
26: List svm_vg(int N, int L_theta, double eps_theta, int L_b, double eps_b, int L_h, double
   eps_h, int L_v, double eps_v, double alpha, double li, double ls, vec y_T, int seed ){
27:   wall_clock timer;
28:   timer.tic();
29:   se seed != 0 então set_seed( seed );
30:   fim se
31:   // initial theta
32:   int acc_theta = 0;
33:   vec theta_cur = zeros<vec>(3, 1);
34:   theta_cur[ 0 ] += 0.1;
35:   theta_cur[ 1 ] += 0.5 * ( log( 1 + 0.98 ) - log( 1 - 0.98 ) );
36:   theta_cur[ 2 ] += log( 0.15 );
37:   // initial h
38:   int acc_h = 0;
39:   vec h_cur = zeros<vec>(T, 1);
40:   h_cur[ 0 ] += 0.1 + 0.15 / sqrt(1 - 0.98 * 0.98 ) * randn();
41:   para (int kt = 1 ; kt < T ; kt++) faça h_cur[ kt ] += 0.1 + 0.98 * ( h_cur[ kt - 1 ] - 0.1 ) +
   0.15 * randn();
42:   fim para
43:   // initial b
44:   int acc_b = 0;
45:   vec b_cur = zeros<vec>(3, 1);
46:   b_cur[ 0 ] += 0.1;
47:   b_cur[ 1 ] += 0.5 * ( log( 1 + 0.03 ) - log( 1 - 0.03 ) );
48:   b_cur[ 2 ] += -0.10;
49:   // initial v
50:   int acc_v = 0;
51:   double v_cur = 10.0;
52:   // initial lambda
53:   vec l_cur = zeros<vec>(T, 1);
54:   para (int i = 0 ; i < T; i++) faça l_cur[ i ] = 1 / R::rgamma( 0.5 * v_cur, 2.0 / v_cur );
55:   fim para

```

```

56: v_cur = (2 / alpha) * atanh((2 * v_cur - ls - li) / (ls - li));
57: // initial chain
58: mat chain_theta = zeros<mat>( 3, N + 1 );
59: chain_theta.col( 0 ) += theta_cur;
60: mat chain_b = zeros<mat>( 3, N + 1 );
61: chain_b.col( 0 ) += b_cur;
62: mat chain_h = zeros<mat>( T, N + 1 );
63: chain_h.col( 0 ) += h_cur;
64: mat chain_v = zeros<mat>( 1, N + 1 );
65: chain_v.col( 0 ) += v_cur;
66: mat chain_l = zeros<mat>( T, N + 1 );
67: chain_l.col( 0 ) += l_cur;
68: // building the chains
69: para ( int it = 1 ; it < N + 1 ; it ++ ) faça
70:   theta_cur = rmhmc_theta( theta_cur, h_cur, 5, L_theta, eps_theta, T, acc_theta );
71:   b_cur = rmhmc_b( b_cur, h_cur, l_cur, 5, L_b, eps_b, T, y_T , acc_b );
72:   h_cur = hmc_h( h_cur, theta_cur, b_cur, l_cur, L_h, eps_h, T, y_T, acc_h );
73:   v_cur = rmhmc_v_vg(v_cur, l_cur, 5, L_v, eps_v, T, acc_v, alpha, li, ls );
74:   l_cur = l_gibbs(v_cur, y_T, h_cur, b_cur, T, alpha, li, ls);
75:   // chain update
76:   chain_theta.col( it ) += theta_cur;
77:   chain_b.col( it ) += b_cur;
78:   chain_h.col( it ) += h_cur;
79:   chain_v.col( it ) += v_cur;
80:   chain_l.col( it ) += l_cur;
81: }
82: fim para
83: // Transformations
84: chain_theta.row( 1 ) = tanh( chain_theta.row( 1 ) );
85: chain_theta.row( 2 ) = exp( chain_theta.row( 2 ) );
86: chain_b.row( 1 ) = tanh( chain_b.row( 1 ) );
87: //  $v = 0.5 * ( (ls - li) * \tanh( 0.5 * \alpha * e ) + (ls + li) )$ 
88: chain_v.row( 0 ) = 0.5 * ( (ls - li) * tanh( 0.5 * alpha * chain_v.row( 0 ) )
89: List chain = List::create(
90:   Named("chain_theta") = chain_theta,
91:   Named("chain_b") = chain_b,
92:   Named("chain_h") = chain_h,
93:   Named("chain_v") = chain_v,
94:   Named("chain_l") = chain_l
95: );
96: vec acc = zeros<vec>(4, 1);
97: acc[ 0 ] += acc_theta;
98: acc[ 1 ] += acc_b;
99: acc[ 2 ] += acc_h;
100: acc[ 3 ] += acc_v;
101: double time = timer.toc();
102: return List::create( Named("chain") = chain, Named("acc") = acc, Named("time") = time );
103: }

```
