

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Estatística

Dissertação de Mestrado

**Modelos dinâmicos com pontos de mudança
para dados de contagem**

por

Paulo Henrique Dourado da Silva

Orientador: Prof.^a Dr.^a Cibele Queiroz da Silva

Maio de 2015

Paulo Henrique Dourado da Silva

Modelos dinâmicos com pontos de mudança para dados de contagem

Dissertação apresentada ao Departamento de Estatística do Instituto de Ciências Exatas da Universidade de Brasília como requisito parcial à obtenção do título de Mestre em Estatística.

Universidade de Brasília

Brasília, maio de 2015

Agradecimentos

- Primeiramente gostaria de agradecer a Deus pelas bençãos e oportunidades que tive ao longo da minha vida acadêmica;
- A minha família, pais e irmã, pelo apoio incondicional em continuar a carreira acadêmica entre outras escolhas que tive que fazer durante minha vida;
- Aos meus primos, Mary e Victor, e meus amigos, Filipe, Arthur, Artur, Thiago, Bruno e Thiago Carvalho, pelo apoio e paciência durante as vezes que tive que me ausentar de suas vidas por conta das reuniões e as vezes que tive que estudar;
- A minha orientadora, Cibele Queiroz da Silva, por todo o conhecimento compartilhado, aprendizagem e pela dedicação em querer me ajudar em muitas dificuldades encontradas ao longo do trabalho. Além da paciência em sugerir melhorias consideráveis para esta dissertação;
- Por fim, gostaria de agradecer aos membros da Banca, Hedibert Freita Lopes e André Luiz Fernandes Cançado, pela disposição em ler e propor melhorias consideráveis para a dissertação, além de sugestão de análises para os artigos que virão como produto desta dissertação.

*'O enfoque Bayesiano e o ortodoxo (frequentista clássico)
se complementam e juntamente fornecem um entendimento
substancialmente melhor da estatística do que isoladamente'.*

Dennis Victor Lindley.

Sumário

Lista de Figuras	8
Lista de Tabelas	9
Resumo	10
Abstract	11
Introdução	12
1 Modelos Dinâmicos Lineares	18
1.1 Modelo Dinâmico Linear Normal Univariado - MDLNU	19
1.1.1 Processo de estimação do sistema e previsão	20
1.1.1.1 Processo de Filtragem	20
1.1.1.2 Suavização	22
1.1.1.3 Previsão	23
1.2 Modelo Dinâmico Linear Normal Multivariado-MDLNM	23
1.2.1 Processo de estimação	24
1.3 Especificação do modelo	25
1.3.1 Modelos Dinâmicos lineares polinomiais	25
1.3.2 Modelos Dinâmicos lineares sazonais via Série de Fourier	27
1.3.3 Modelos Dinâmicos lineares de regressão	29
1.4 Modelos dinâmicos com parâmetros desconhecidos	31
1.4.1 Abordagem clássica - Estimação por Máxima Verossimilhança	32
1.4.2 Abordagem Bayesiana - Procedimentos <i>online</i> e <i>offline</i> de estimação	32

1.4.2.1	Especificação de W_t por fatores de desconto	34
1.4.2.2	Modelos Dinâmicos Lineares com V_t desconhecido	35
1.4.2.3	Um modelo com fator de desconto para V_t variável no tempo	37
2	O pacote <i>dlm</i>	39
2.1	Exemplo 1 - MDL Polinomial	40
2.2	Exemplo 2 - MDL Sazonal	43
2.3	Exemplo 3 - MDL de Regressão	46
3	Modelos Dinâmicos Lineares Generalizados	48
3.1	Processo de estimação	50
3.2	Exemplo: Modelo Dinâmico Binário	52
3.3	Modelo Dinâmico Poisson	55
3.3.1	Processo de estimação	58
4	Filtros de Partículas Básicos	64
4.1	<i>Filtro Bootstrap</i>	69
4.1.1	Limitações dos filtros de partículas baseados no SIR	71
4.2	<i>Filtro de Partículas Auxiliar</i>	72
4.3	Exemplos	74
4.3.1	Modelo Dinâmico Linear Gaussiano	74
4.3.2	Modelo Dinâmico Poisson	77
5	Filtros de Partículas com parâmetros estáticos - <i>Parameter Learning</i>	83
5.1	Filtro de Liu e West	84
5.2	Filtro de Storvik	87
5.3	<i>Particle Learning</i>	88
5.4	Suavização	89
5.5	Exemplo	90
6	Modelos de Regressão Dinâmica com Pontos de Mudança	94
6.1	Modelo de regressão dinâmica para problemas de classificação	95

6.1.1	<i>Dynamic Model Averaging</i>	99
6.2	Regressão dinâmica Poisson Bayesiana	101
6.2.1	Análise de dados simulados	103
7	Modelos Dinâmicos para a Detecção de Pontos de Mudança via Filtro de Partículas	113
7.1	Algoritmo de Chopin (2007)	114
7.1.1	Algoritmo 2 - Rao-Blackwellização do componente discreto . .	116
7.1.2	Nova proposta - Movimento fracional	117
7.2	Algoritmo de Caron et al. (2012)	120
7.2.1	Modelo Estatístico	120
7.2.2	Inferência Exata	123
7.2.3	Estimativa recursiva dos parâmetros	126
7.2.4	Inferência Aproximada	127
7.3	Modelo Dinâmico Poisson com Ponto de Mudança	130
7.3.1	Modelo Dinâmico Poisson com Ponto de Mudança segundo o filtro de Chopin (2007)	130
7.3.1.1	Análise de dados simulados	131
7.3.2	Modelo Dinâmico Poisson com Ponto de Mudança segundo o filtro de Caron et al. (2012)	136
7.3.2.1	Análise de dados simulados	137
8	Novos desenvolvimentos	143
8.1	Algoritmo McPE (McCormick com parâmetros estáticos)	144
8.2	Algoritmo FChAP (Filtro de Chopin com aprendizado de partículas)	147
9	Análise de dados simulados usando os novos desenvolvimentos	149
9.1	Modelos para dados de contagem com superdispersão e pontos de mudança: Modelo Binomial Negativo	149
9.1.1	Análise dos dados simulados via algoritmo McPE	150
9.1.2	Análise de dados simulados via algoritmo FChAP	153
9.2	Modelos para dados inflacionados de zeros e pontos de mudança: Modelos ZIP e ZINB	158

9.2.1	Modelo Poisson Inflacionado de Zeros - ZIP	159
9.2.1.1	Análise dos dados simulados via algoritmo McPE . . .	159
9.2.1.2	Análise de dados simulados via algoritmo FChAP . . .	163
9.2.2	Modelo Binomial Negativo inflacionado de Zeros - ZINB . . .	168
9.2.2.1	Análise de dados simulados via algoritmo McPE . . .	168
9.2.2.2	Análise de dados simulados via algoritmo FChAP . . .	172
10	Aplicações a dados reais	178
10.1	Aplicações utilizando o Algoritmo de McCormick et al. (2012)	179
10.1.1	Número de Casos de Sífilis no Estado de Maryland - US	179
10.2	Aplicações utilizando os filtros de Chopin (2007), Caron et al. (2012) e FChAP	185
10.2.1	Desastres em minérios de carvão na Inglaterra	185
10.2.2	Concentração de ozônio na cidade do México	190
10.2.3	Número de Lesões na Saúde Ocupacional nos Estados Unidos	195
10.2.4	Número de Casos de Poliometite nos Estados Unidos	201
11	Conclusão e trabalhos futuros	207

Lista de Figuras

1	Série temporal sem mudança estrutural.	13
2	Série temporal com um ponto de mudança estrutural.	13
1.1	Análise sequencial do processo de estimação do modelo dinâmico linear	21
2.1	Exemplo 1 - Modelo Dinâmico Linear Polinomial	42
2.2	Exemplo 2 - Modelo Dinâmico Linear Sazonal	45
3.1	Análise sequencial do processo de estimação do modelo dinâmico linear generalizado	52
3.2	Modelo Dinâmico Poisson	63
4.1	Representação esquemática do <i>filtro bootstrap</i>	71
4.2	Séries simuladas para o MDL Gaussiano	74
4.3	Resultados do algoritmo FB	75
4.4	Resultados do algoritmo APF	76
4.5	Distribuição do erro quadrático médio para algoritmo FB	77
4.6	Distribuição do erro quadrático médio para algoritmo FPA	78
4.7	Séries simuladas segundo um modelo dinâmico Poisson.	79
4.8	Resultados do algoritmo FB	80
4.9	Resultados do algoritmo FPA	81
4.10	Distribuição do erro quadrático médio entre os valores reais e os valores estimados pelos filtros FB e FPA	82
5.1	Comparação entre os filtros LW, STORVIK e PL	92
5.2	Comparação entre os filtros LW, STORVIK e PL - EQM	93
6.1	Dados simulados - Base de dados 1	108

6.2	Resultados - Base de dados 1	109
6.3	Dados simulados - Base de dados 2	110
6.4	Resultados - Base de dados 2	111
6.5	Resultados Regressão Estática - Base de dados 3	112
7.1	Representação esquemática do processo descrito por Chopin (2007).	115
7.2	Hierarquia de modelagem do segmento segundo Caron et al. (2012).	122
7.3	Filtro Chopin para a distribuição Poisson	134
7.4	Filtro Chopin para a distribuição Poisson replicado	135
7.5	Filtro Caron para a distribuição Poisson - Base de dados 1	141
7.6	Filtro Caron para a distribuição Poisson - Base de dados 2	142
9.1	Resultado do algoritmo McPE para a distribuição BN	152
9.2	Análise da série Binomial Negativa com quebra estrutural no tempo $t = 500$ via algoritmo de McCormick com parâmetros estáticos (McPE)	153
9.3	Resultado do algoritmo FChAP para a distribuição BN	156
9.4	Filtro FChAP para a distribuição Binomial Negativa replicado	157
9.5	Série simulada ZIP	161
9.6	Resultado do algoritmo McPE para a distribuição ZIP	162
9.7	Resultado do algoritmo FChAP para a distribuição ZIP	166
9.8	Filtro FChAP para a distribuição ZIP replicado	167
9.9	Série simulada ZINB	170
9.10	Resultado do algoritmo McPE para a distribuição ZINB	171
9.11	Resultado do algoritmo FChAP para a distribuição ZINB	175
9.12	Probabilidade a <i>posteriori</i> de ocorrer mudança.	176
9.13	Filtro FChAP para a distribuição ZINB replicado	177
10.1	Série temporal observada e distribuição de frequência para os dados de ocorrência de sífilis.	181
10.2	Média filtrada para a distribuição Poisson	181
10.3	Estimativas para os parâmetros do modelo Binomial Negativo.	182
10.4	Estimativas para os parâmetros do modelo ZIP.	183
10.5	Estimativas para os parâmetros do modelo ZINB.	184

10.6	Número de desastres em minérios de carvão na Inglaterra entre 1950 e 1962	186
10.7	Estimativas para o filtro de Caron et al. (2012) utilizando a distribuição Poisson.	187
10.8	Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.	188
10.9	Estimativas para o filtro de Chopin (2007) utilizando a distribuição Binomial Negativa.	189
10.10	Número de vezes em que a concentração de ozônio ultrapassou o limite de 0.17 ppm (partes por milhão) na cidade do México os anos de 1998 e 2004.	191
10.11	Estimativas para o filtro de Caron et al. (2012) utilizando a distribuição Poisson.	192
10.12	Estimativas para o filtro de Chopin (2007) utilizando a distribuição de Poisson.	193
10.13	Estimativas para o filtro de Chopin (2007) utilizando a distribuição Binomial Negativa.	194
10.14	Número de casos mensais de lesões na Saúde Ocupacional nos Estados Unidos no período de Julho de 1988 a Outubro de 1995.	196
10.15	Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.	197
10.16	Estimativas para o filtro FChAP utilizando a distribuição Binomial Negativa.	198
10.17	Estimativas para o filtro FChAP utilizando a distribuição ZIP.	199
10.18	Estimativas para o filtro FChAP utilizando a distribuição ZINB.	200
10.19	202
10.20	Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.	203
10.21	Estimativas para o filtro FChAP utilizando a distribuição Binomial Negativa.	204
10.22	Estimativas para o filtro FChAP utilizando a distribuição ZIP.	205
10.23	Estimativas para o filtro FChAP utilizando a distribuição ZINB.	206

Lista de Tabelas

1	Guia de estudo.	17
6.1	Estimativa dos coeficientes do modelo Poisson estático	107
10.1	Critérios de informação para os modelos estimados para o número de casos de Sífilis em Maryland.	180
10.2	Comparação entre os tempos, aproximados, de processamentos dos algoritmos.	186
10.3	Comparação entre os tempos, aproximados, de processamentos dos algoritmos.	191
10.4	Critérios de informação para os modelos estimados para os dados de lesões na saúde ocupacional dos Estados Unidos.	196
10.5	Critérios de informação para os modelos estimados para os dados de Poliomielite.	202
11.1	Estimativa dos coeficientes do modelo BN estático.	221

Resumo

Nesta dissertação foram desenvolvidos modelos dinâmicos para dados de contagem quando estes apresentam quebras estruturais. Os métodos aqui desenvolvidos são baseados nos modelos de regressão dinâmica proposto por McCormick et al. (2012), e nos filtros de partículas propostos por Chopin (2007), Fearnhead e Liu (2007) e Caron et al.(2012). Inicialmente apresentam-se os principais aspectos metodológicos a respeito dos modelos dinâmicos, tais como os modelos dinâmicos lineares e os modelos dinâmicos lineares generalizados. Posteriormente, são abordados os principais métodos existentes na literatura sobre filtros de partículas.

A partir desses estudos, propomos, também, extensões inéditas para o modelo de regressão dinâmica e para o filtro de partículas de Chopin (2007) para a estimação de parâmetros estáticos, além dos estados do sistema. Tais algoritmos são denominados como *Algoritmo de McCormick com parâmetros estáticos* (McPE) e *Filtro de Chopin com Aprendizado de Partículas* (FChAP). Nesta dissertação desenvolvemos extensões para dados que apresentam superdispersão e/ou inflação de zeros por meio das distribuições Binomial Negativa, Poisson Inflacionada de zeros e Binomial Negativa Inflacionada de zeros. Tais modelos foram ilustrados por meio de dados simulados e, posteriormente, foram feitas aplicações a cinco séries temporais reais de dados de contagem.

Palavras Chave: *modelos dinâmicos, filtro de partículas, distribuição Poisson, distribuição Binomial Negativa, distribuição ZIP, distribuição ZINB, detecção dinâmica de pontos de mudança, abordagem Bayesiana, séries temporais.*

Abstract

In this thesis were developed dynamic models for count data when they have structural breaks, based on dynamic regression models proposed by McCormick et al. (2012), and particle filters proposed by Chopin (2007), Fearnhead and Liu (2007) and Caron et al. (2012). Initially we present the main methodological aspects about the dynamic models such as Dynamic Linear Models and Generalized Linear Dynamic Models. Subsequently, we present the main existing methods in the literature on particle filters.

From these studies, we propose new extensions for dynamic regression model and the particle filter methods proposed by Chopin (2007) for the estimation of static parameters, in addition to the states. We named these new algorithms by *McCormick algorithm with static parameters* (McPE) and *Chopin Filter with Particles Learning* (FChAP). Based on McPE and FChAP, it was possible to develop extensions for data that show overdispersion and / or zero inflation through the Negative Binomial, Zero Inflated Poisson (ZIP) and Zero Inflated Negative Binomial (ZINB) distributions.

Those models were illustrated using both simulated data and five real counting time series data.

Keywords: *dynamic models, particle filter, Poisson distribution, Negative Binomial distribution, ZIP distribution, ZINB distribution, dynamic changepoint detection, Bayesian approach, time series.*

Introdução

Podemos observar que o uso dos modelos dinâmicos em séries de tempo cresceu muito ao longo dos anos. Tal crescimento se deve tanto a sua versatilidade quanto a sua elegância e generalidade em descrever a correlação temporal entre as observações, além de permitir a inserção de covariáveis ao processo de modelagem. Conforme afirmam West e Harrison (1997), os modelos dinâmicos constituem uma classe mais geral de modelagem, os quais têm como casos especiais os modelos de regressão estática e alguns dos modelos mais difundidos em séries de tempo, como os modelos ARIMA e GARCH. Os modelos dinâmicos permitem que os parâmetros evoluam de maneira estocástica e tal evolução é descrita por uma estrutura Markoviana.

Muitas séries, tais como sequências de DNA, preço de estoques, poluição do ar (Achcar et. al, 2008), séries de retornos de longa duração (Chopin, 2007), crescimento bacteriológico (Whittaker e Fruhwirth-Schatter, 1994) apresentam heterogeneidade temporal. Nesse contexto, uma abordagem usual consiste em segmentar uma sequência de observações y_1, y_2, \dots, y_T em $m+1$ segmentos, escolhendo uma sequência de quantidades que indicam em que posição no tempo ocorreram mudanças estruturais na série $0 < \tau_1 < \tau_2 < \dots < \tau_m < T$, tal que as observações sejam homogêneas dentro dos segmentos, no sentido de apresentarem a mesma estrutura de modelo, e heterogêneos entre os segmentos.

A Figura 1 ilustra um exemplo de uma série temporal sem mudança estrutural. Já a Figura 2 ilustra um exemplo de uma série temporal com um ponto de mudança estrutural.

O número de publicações relacionadas aos métodos de ponto de mudanças, desde o trabalho seminal de Hinkley (1970), é extremamente grande. Achcar et al. (2008) usaram uma abordagem via processo de Poisson não homogêneo para modelar pon-

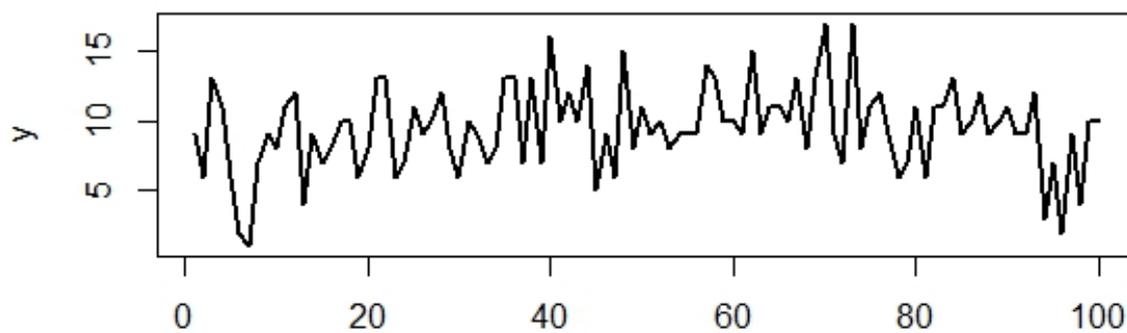


Figura 1: Série temporal sem mudança estrutural.

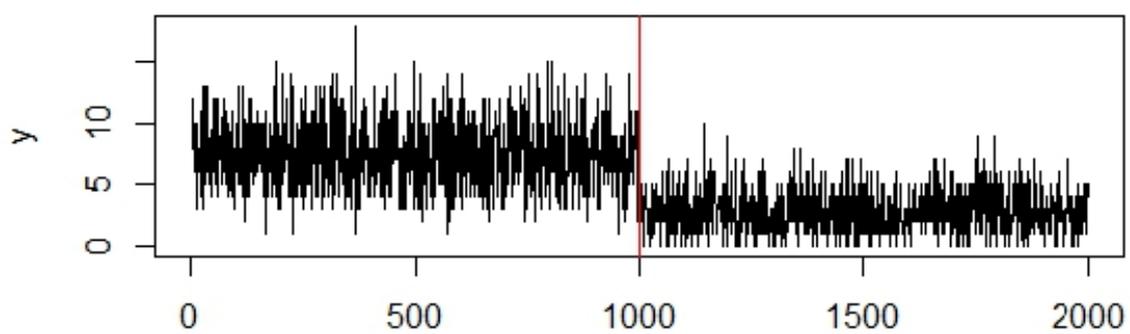


Figura 2: Série temporal com um ponto de mudança estrutural.

tos de mudanças em dados sobre poluição de ar na cidade do México. Whittaker e Fruhwirth-Schatter (1994) utilizaram modelos de multiprocessamento (West e Harrison 1997) para construir um modelo dinâmico de pontos de mudanças para detectar o início do crescimento de infecções bacteriológicas. Recentemente, foram propostos modelos nos quais é utilizado filtro de partículas para detectar a quantidade de pontos de mudanças e onde eles estão localizados baseados no modelo de partição produto proposto por Barry e Hartigan (1992). Chopin (2007), desenvolveu um filtro de partículas para detectar mudanças em séries de longa duração, Fearnhead e Liu (2007) utilizaram filtro de partículas para modelar dados de DNA e Caron et al. (2012) generalizaram o filtro proposto por Fearnhead e Liu (2007) para estimar parâmetros estáticos, além dos estados.

Existem outros métodos que apenas se adaptam a mudanças estruturais na série, mas não permitem a detecção, ou seja, contam quantas são e localizam onde tais mudanças ocorreram. Esse é o caso dos modelos de regressão dinâmica desenvolvidos por Raftery et al. (2010) e McCormick et al. (2012). Nesses modelos, incorpora-se um fator de desconto na variância da distribuição dos estados, que aumenta a incerteza em períodos de grande volatilidade. Tal parâmetro flexibiliza o modelo de uma tal maneira que mudanças bruscas na estrutura da série são corretamente captadas pelo modelo.

Nesta dissertação foram desenvolvidos modelos dinâmicos para dados de contagem quando estes apresentam quebras estruturais, baseados nos modelos de regressão dinâmica proposto por McCormick et al. (2012), e nos filtros de partículas propostos por Chopin (2007), Fearnhead e Liu (2007) e Caron et al.(2012).

A partir desses estudos propomos novos algoritmos representando extensões inéditas para o modelo de regressão dinâmica e para o filtro de partículas de Chopin (2007), para a estimação de parâmetros estáticos, além dos estados. Denominamos tais algoritmos por como *Algoritmo de McCormick com parâmetros estáticos* (McPE) e *Filtro de Chopin com Aprendizado de Partículas* (FChAP). Sendo assim, foi possível desenvolver extensões para dados que apresentam superdispersão e/ou inflação de zeros por meio das distribuições Binomial negativa, Poisson inflacionada de zeros e Binomial negativa inflacionada de zeros.

Tais modelos foram ilustrados por meio de dados simulados e, posteriormente,

foram feitas aplicações a cinco séries temporais reais de dados de contagem. A dissertação está organizada da seguinte forma:

No Capítulo 1 descreveu-se os principais aspectos teóricos relacionados aos modelos dinâmicos lineares normais univariado e multivariado. Tal capítulo é de suma importância para o entendimento das principais características dos modelos dinâmicos, como o processo de estimação sequencial. Nesse capítulo apresentam-se, também, as diferentes especificações dos modelos dinâmicos, como os modelos dinâmicos lineares polinomiais, sazonais e de regressão.

No Capítulo 2, apresenta-se o pacote *dln* associado ao software *R*. Nesse capítulo mostra-se como utilizar as funções implementadas no pacote *dln* para a simulação e estimação das diferentes especificações dos modelos dinâmicos, além de ilustrar trechos de programas para fins didáticos.

No Capítulo 3 discutiu-se o modelo dinâmico linear generalizado (West et al., 1985), para distribuições na família exponencial de Nelder e Wedderburn (1972). Nesse capítulo descreve-se como se dá o processo de estimação, exemplificando-o pelo modelo dinâmico binário. Descreve-se, também, o modelo dinâmico Poisson, ilustrando o processo de estimação. Tal modelo constitui-se em um pré-requisito essencial para o entendimento dos modelos que serão desenvolvidos nesta dissertação.

No Capítulo 4 descrevem-se as principais ideias relacionadas aos filtros de partículas. Primeiramente descreve-se o *Filtro Bootstrap*, FB, desenvolvido por Gordon et al. (1993). Posteriormente, descreve-se o Filtro de Partículas Auxiliar, FPA, desenvolvido por Pitt e Shephard (1999). Tais filtros constituem os filtros de partículas básicos (Lopes e Tsay, 2011).

Em casos em que existem parâmetros estáticos a serem estimados pelos modelos dinâmicos lineares, faz-se necessário o uso de um algoritmo que estime, além dos estados, os parâmetros estáticos. Existem três filtros de partículas amplamente divulgados na literatura para a estimação dos estados e dos parâmetros estáticos. O primeiro deles é o filtro de Liu e West (Liu e West, 2001), o segundo é o filtro de Storvik (Storvik, 2002) e o terceiro filtro é o *Particle learning* (Carvalho et al., 2010). Tais filtros são descritos no Capítulo 5.

No capítulo 6, descreve-se, de uma forma detalhada, o modelo de regressão proposto por McCormick et al. (2012), propomos o modelo de regressão dinâmica Pois-

son baseado no algoritmo de McCormick et al. (2012). Posteriormente, ilustramos o modelo de regressão dinâmica Poisson por meio de dados simulados.

No Capítulo 7 são descritos os filtros de Chopin (2007) e Caron et al. (2012). Aplicamos os filtros para a distribuição Poisson, e posteriormente os ilustramos por meio de dados simulados.

No Capítulo 8 descrevem-se duas extensões inéditas que desenvolvemos durante os trabalhos desta dissertação, para o modelo de regressão dinâmica proposto por McCormick et al. (2012), e para o filtro de partículas de Chopin (2007) para a estimação conjunta dos estados e dos parâmetros estáticos. Tais extensões foram denominadas como *Algoritmo de McCormick com parâmetros estáticos* (McPE) e *Filtro de Chopin com Aprendizado de Partículas* (FChAP).

No Capítulo 9 aplicou-se os novos algoritmos McPE e FChAP para o caso em que os dados apresentam superdispersão, onde utilizou-se a distribuição Binomial Negativa. Posteriormente, aplicaram-se os novos algoritmos para o caso em que os dados apresentam superdispersão e/ou inflação de zeros, para as distribuições Binomial Negativa Inflacionada de Zeros e Poisson Inflacionada de Zeros. Os algoritmos foram ilustrados por meio de dados simulados para cada modelo.

No Capítulo 10 os modelos propostos nos Capítulos 6, 7, 8 e 9 foram aplicados a cinco séries temporais reais de dados de contagem. Por fim, no Capítulo 11, fazemos as considerações finais da dissertação e apontamos possíveis trabalhos futuros.

A Tabela 1 traz um guia ao leitor sobre quais algoritmos foram criados, descrição e simulações e aplicações aos dados para cada modelo estudado nesta dissertação.

Tabela 1: Guia de estudo.

Modelo de Série temporal estudado	Algoritmos utilizados na estimação	Descrição do modelo, simulações e códigos	Aplicação a dados reais
Poisson	McCormick et al. (2012)	Seção 6.2	Seção 10.1.1
Poisson com ponto de mudança	Chopin (2007) e Caron et al. (2012)	Seções 7.1.3 e 7.2.5	Seções 10.2.1, 10.2.2, 10.2.3 e 10.2.4
BN	McPE	Seção 9.1 e apêndice F.1	Seção 10.1.1
BN com ponto de mudança	FChAP e McPE	Seções 9.1.1 e 9.1.2	Seções 10.2.1, 10.2.2, 10.2.3 e 10.2.4
ZIP	McPE	Seção 9.2.1.1 e apêndice F.2	Seção 10.1.1
ZIP com ponto de mudança	McPE e FChAP	Seção 9.2.1.2	Seções 10.2.3 e 10.2.4
ZINB	McPE	Seção 9.2.2.1 e apêndice F.3	Seção 10.1.1
ZINB com ponto de mudança	McPE e FChAP	Seção 9.2.2.2	Seções 10.2.3 e 10.2.4

Capítulo 1

Modelos Dinâmicos Lineares

Os modelos dinâmicos lineares representam um caso particular do que foi batizado na literatura estatística como modelos dinâmicos. Ainda que o tema central desta dissertação não envolva a suposição de normalidade ou linearidade, entendemos que diversas características de interesse podem ser mais bem visualizadas a partir do estudo do caso normal, para o qual os resultados são sempre mais simples. Nesse sentido, optou-se por explorar, neste capítulo, os principais aspectos dos modelos dinâmicos lineares a fim de descrever os fundamentos básicos dessa classe de modelos.

Este capítulo se divide da seguinte forma: na Seção 1.1 são descritos os aspectos teóricos a respeito do modelo dinâmico linear normal univariado, como estimação e previsões. Na Seção 1.2 o modelo univariado é estendido para o caso multivariado. A Seção 1.3 é responsável por mostrar as diferentes especificações do modelo dinâmico linear normal. Através dessas especificações pode-se modelar séries temporais com tendências polinomiais, séries que apresentam sazonalidade ou também ajustar um modelo de regressão dinâmica ou estática.

Para as Seções de 1.1 a 1.3 os aspectos teóricos supõem que as variâncias são conhecidas. Entretanto, na prática essa suposição é irreal e tais parâmetros devem ser estimados. Sendo assim, na Seção 1.4 o modelo dinâmico linear normal é estendido no sentido de permitir a estimação de parâmetros desconhecidos.

1.1 Modelo Dinâmico Linear Normal Univariado - MDLNU

Segundo West e Harrison (1997), o modelo dinâmico linear (MDL) normal univariado¹ é a forma mais simples de um MDL, mas, apesar de sua simplicidade, traz os principais conceitos sobre modelagem dinâmica. As idéias tratadas aqui são usadas em praticamente todos os modelos dinâmicos propostos na literatura. O Modelo Dinâmico Linear Normal Univariado (MDLNU) é descrito pelas seguintes componentes:

- **Equação das observações:**

$$y_t = \theta_t + v_t, \quad v_t \sim N(0, V_t); \quad (1.1)$$

- **Equação do sistema (ou evolução):**

$$\theta_t = \theta_{t-1} + w_t, \quad w_t \sim N(0, W_t); \quad (1.2)$$

- **Informação inicial:**

$$(\theta_0 | D_0) \sim N(m_0, C_0),$$

A série temporal observada, y_t , modelada em função do processo latente ou não observado, θ_t , é tal que os componentes aleatórios v_t e w_t são assumidos serem independentes entre si e de $(\theta_0 | D_0)$. Os componentes w_t controlam a evolução da série através da variância W_t , de forma que valores pequenos dessa variância implicam em uma evolução mais suave da série. Nos modelos dinâmicos, a razão W_t/V_t possui uma interpretação interessante acerca da trajetória da série. Quando esta razão é relativamente pequena ($V_t > W_t$), isso implica que a maior parte do movimento da série se dá pelas observações. Em contrapartida, para valores relativamente grandes ($W_t > V_t$), o movimento da série se dá, em parte pelas observações, mas também devido às variações dos parâmetros de evolução θ_t .

¹Muitos autores tratam o modelo linear dinâmico normal univariado por modelo de primeira ordem

A equação (1.1) descreve um *modelo linear polinomial de primeira ordem* em que as observações y_t variam em torno de um nível localmente constante θ_t .

1.1.1 Processo de estimação do sistema e previsão

Para a descrição das principais idéias envolvidas no processo de estimação de um MDL univariado, considere as variâncias V_t e W_t conhecidas. O processo de estimação é composto por três componentes: (1) evolução da série, (2) previsão de uma nova observação e (3) atualização dos parâmetros do sistema.

A evolução do sistema se dá pela equação do sistema (1.2). Nessa fase é utilizada a distribuição preditiva de θ_{t-1} para determinar a *priori* no tempo t . A previsão a um passo a frente segue da distribuição marginal de $(y_t|D_{t-1})$ dada por

$$p(y_t|D_{t-1}) \propto \int p(y_t|\theta_t, D_{t-1})p(\theta_t|D_{t-1})d\theta_t \propto \int p(y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t.$$

Combinando a distribuição a *priori* no tempo t com a equação de verossimilhança no tempo t , via teorema de Bayes, é feita a atualização dos parâmetros²

Mais precisamente, busca-se obter as distribuições condicionais do sistema $(\theta_s|D_t)$ para $s < t$ (*suavização*), $s = t$ (*filtragem*) e $s > t$ (*previsão do sistema*), respectivamente. Tais processos serão detalhados nas próximas seções.

1.1.1.1 Processo de Filtragem

O processo de filtragem se dá da seguinte maneira:

- **Distribuição a *posteriori* no tempo $t - 1$:**

Para uma média m_{t-1} e uma variância C_{t-1} , ambos obtidos recursivamente,

$$(\theta_{t-1}|D_{t-1}) \sim N(m_{t-1}, C_{t-1}). \quad (1.3)$$

- **Distribuição a *priori* no tempo t :**

²Distribuição a *posteriori* de θ_t .

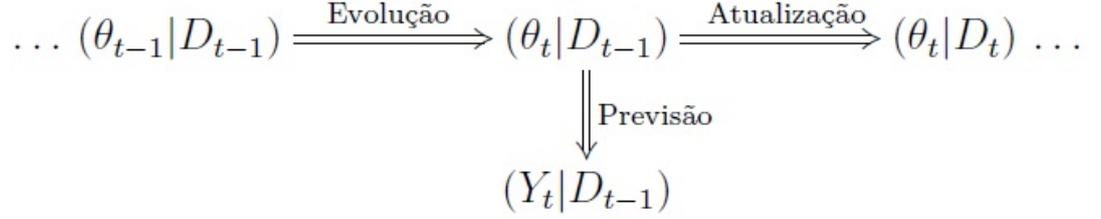


Figura 1.1: Análise sequencial do processo de estimação do modelo dinâmico linear

$$(\theta_t|D_{t-1}) \sim N(a_t, R_t), \quad (1.4)$$

onde $a_t = m_{t-1}$ e $R_t = C_{t-1} + W_t$.

- **Distribuição preditiva um passo à frente:**

$$(Y_t|D_{t-1}) \sim N(f_t, Q_t), \quad (1.5)$$

onde $f_t = a_t$ e $Q_t = R_t + V_t$.

- **Distribuição a *posteriori* no tempo t :**

$$(\theta_t|D_t) \sim N(m_t, C_t), \quad (1.6)$$

onde $m_t = a_t + A_t e_t$, $C_t = R_t - A_t^2 Q_t$, $A_t = R_t/Q_t$ e $e_t = Y_t - f_t$.

Para demonstrar os resultados acima, pode-se recorrer ao teorema de Bayes ou às propriedades da distribuição normal. Ambos os casos são demonstrados por West e Harrison (1997).

A Figura 1.1 ilustra a análise sequencial do processo de estimação associado ao modelo dinâmico linear normal univariado.

É interessante notar que e_t , definido na equação (1.6) do processo de filtragem, pode ser interpretado como o erro de previsão, e A_t como um peso adaptativo dada à observação mais recente y_t . Além disso, m_t pode ser reescrito como:

$$m_t = m_{t-1} + A_t e_t = \frac{R_t V_t}{Q_t} \left(\frac{m_{t-1}}{R_t} + \frac{y_t}{V_t} \right) = A_t y_t + (1 - A_t) m_{t-1},$$

A_t também pode ser interpretado como o quadrado da correlação entre y_t e θ_t , ou como o quadrado do coeficiente de regressão linear entre θ_t e y_t . De fato, temos que a covariância de y_t e θ_t é dada por

$$\begin{aligned}\text{Cov}(y_t, \theta_t | D_{t-1}) &= E(y_t \theta_t | D_{t-1}) - E(y_t | D_{t-1})E(\theta_t | D_{t-1}) = E[(\theta_{t-1} + w_t + v_t)(\theta_{t-1} + w_t) | D_{t-1}] \\ &= E[\theta_{t-1} + w_t + v_t | D_{t-1}]E[\theta_{t-1} + w_t | D_{t-1}] \\ &= C_{t-1} + W_t = R_t.\end{aligned}$$

As variâncias de y_t e θ_t são dadas por, respectivamente,

$$V(y_t | D_{t-1}) = V(E(y_t | D_{t-1}, \theta_t)) + E(V(y_t | D_{t-1}, \theta_t)) = C_{t-1} + W_t + V_t = Q_t,$$

e

$$V(\theta_t | D_{t-1}) = V(\theta_{t-1} + w_t | D_{t-1}) = C_{t-1} + W_t = R_t.$$

Com isso, o coeficiente de correlação de Pearson para y_t e θ_t é dado por

$$\rho_{y_t, \theta_t} = \frac{\text{Cov}(y_t, \theta_t | D_{t-1})}{\sqrt{V(y_t | D_{t-1})}\sqrt{V(\theta_t | D_{t-1})}} = \frac{R_t}{\sqrt{R_t}\sqrt{Q_t}},$$

isso implica que $\rho_{y_t, \theta_t}^2 = R_t/Q_t = A_t$.

1.1.1.2 Suavização

Em análises de séries temporais, existe o interesse em reconstruir retrospectivamente o comportamento de um sistema revisando inferências relativas a tempos passados valendo-se de toda a informação disponível D_T . Tal interesse pode ser útil para elucidar o entendimento sobre o que ocorreu de fato na série.

Para um MDL definido no início da Seção 1.1, as *distribuições retrospectivas* (*distribuições de suavização*) são todas gaussianas e, conseqüentemente, as médias e variâncias podem ser facilmente calculadas recursivamente. Ou seja, se $(\theta_{t+1} | D_T) \sim N(s_{t+1}, S_{t+1})$, então $(\theta_t | D_T) \sim N(s_t, S_t)$, onde

$$s_t = m_t + C_t G'_{t+1} R_{t+1}^{-1} (s_{t+1} - a_{t+1}) \quad (1.7)$$

$$S_t = C_t - C_t G'_{t+1} R_{t+1}^{-1} (R_{t+1} - S_{t+1}) R_{t+1}^{-1} G_{t+1} C_t, \quad (1.8)$$

com valores iniciais $s_T = m_T$ e $S_T = C_T$. A demonstração pode ser encontrada em Petris et al. (2012) página 61.

1.1.1.3 Previsão

As previsões considerando h passos à frente são baseadas em

$$\begin{aligned}
 E(y_{t+h}|D_t) &= E(\theta_{t+h} + v_{t+h}|D_t) \\
 &= E(\theta_{t+h} + w_{t+h}|D_t) = \dots = \\
 &= E\left(\theta_t + \sum_{j=1}^h w_{t+j}\right). \tag{1.9}
 \end{aligned}$$

A incerteza relativa à previsão h passos à frente, é dada por

$$\begin{aligned}
 V(y_{t+h}|D_t) &= V(\theta_{t+h} + v_{t+h}|D_t) \\
 &= V(\theta_{t+h} + w_{t+h}|D_t) = \dots = \\
 &= V\left(\theta_t + \sum_{j=1}^h w_{t+j}\right) = C_t + \sum_{j=1}^h (W_{t+j} + V_{t+j}). \tag{1.10}
 \end{aligned}$$

Fazendo $f_t(h) = E(y_{t+h}|D_t)$ e $Q_t(h) = V(y_{t+h}|D_t)$, a distribuição preditiva h passos a frente é dada por

$$(y_{t+h}|D_t) \sim N(f_t(h), Q_t(h)).$$

1.2 Modelo Dinâmico Linear Normal Multivariado-MDLNM

No MDLNM, o parâmetro θ_t é substituído por um vetor de parâmetros desconhecidos, θ_t . O MDLNM é caracterizado pelas seguintes componentes:

- **Equação das observações:**

$$y_t = F_t' \theta_t + v_t, \quad v_t \sim N(0, V_t); \tag{1.11}$$

- **Equação do Sistema:**

$$\theta_t = \mathbf{G}'_t \theta_{t-1} + w_t; \quad w_t \sim N(0, \mathbf{W}_t); \quad (1.12)$$

- **Informação inicial:**

$$(\theta_0 | D_0) \sim (m_0, C_0). \quad (1.13)$$

Em que, F_t e θ_t são vetores de dimensão p , com F_t conhecido e G_t é uma matriz $p \times p$ que define a evolução de θ_t . Novamente, assume-se que v_t e w_t são independentes entre si e de $(\theta_0 | D_0)$.

1.2.1 Processo de estimação

Assume-se que as matrizes \mathbf{F}_t , \mathbf{G}_t , \mathbf{W}_t e V_t são conhecidas. O processo de estimação se dá de maneira análoga ao caso univariado:

- **Distribuição a *posteriori* no tempo $t - 1$:**

Para um vetor de médias m_{t-1} e uma matriz de covariâncias C_{t-1} , ambos conhecidos,

$$(\theta_t | D_{t-1}) \sim N_n(m_{t-1}, C_{t-1}). \quad (1.14)$$

- **Distribuição a *priori* no tempo t :**

$$(\theta_t | D_{t-1}) \sim N_n(a_t, R_t), \quad (1.15)$$

onde $a_t = G_t m_{t-1}$ e $R_t = G_t C_{t-1} G'_t + W_t$.

- **Distribuição preditiva um passo à frente:**

$$(Y_t | D_{t-1}) \sim N(f_t, Q_t), \quad (1.16)$$

onde $f_t = F'_t a_t$ e $Q_t = F'_t R_t F_t + V_t$.

- **Distribuição a *posteriori* no tempo t :**

$$(\theta_t | D_t) \sim N_n(m_t, C_t). \quad (1.17)$$

onde $m_t = a_t + A_t e_t$, $C_t = R_t - A_t Q_t A_t'$, $A_t = R_t F_t Q_t^{-1}$ e $e_t = Y_t - f_t$.

1.3 Especificação do modelo

1.3.1 Modelos Dinâmicos lineares polinomiais

Os modelos dinâmicos lineares polinomiais são um caso particular dos MDL, e são amplamente utilizados para descrever o *nível* e a *tendência* de uma série temporal, sendo que ambas as componentes (nível e tendência) captam variações locais da série temporal (Petris et al. 2010).

Um modelo polinomial de ordem n é um MDL com matrizes F_t e G_t constantes, ou seja, $F_t = F$ e $G_t = G$, e função de previsão da forma

$$f_t(m) = E(Y_{t+m}|D_t) = a_{t,0} + a_{t,1}m + \dots + a_{t,n-1}m^{n-1}, \quad m \geq 0, \quad (1.18)$$

em que $a_{t,0}, a_{t,1}, \dots, a_{t,n-1}$ são funções lineares de $m_t = E(\theta_t|D_t)$ e são independentes de m . Na prática, as matrizes F e G não são unicamente especificadas, sendo a *forma canônica* a mais utilizada.

O MDL canônico é descrito em termos da matriz *Bloco de Jordan*, definida por

$$J_n(\lambda) = \begin{pmatrix} \lambda & 1 & 0 & 0 & \dots & 0 \\ 0 & \lambda & 1 & 0 & \dots & 0 \\ 0 & 0 & \lambda & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & \lambda \end{pmatrix}.$$

Dentre os MDL polinomiais, três submodelos se destacam:

- **Modelo de nível local - Modelo de ordem 1:** O modelo de nível local nada mais é do que o MDL univariado definido pelas equações (1.1) e (1.2). Para

esse caso, a distribuição preditiva m passos a frente é dada por

$$Y_{t+m}|D_t \sim N(m_t, Q_t(m)), \quad m \geq 1,$$

onde $Q_t(m) = C_t + \sum_{j=1}^m W_{t+j} + V_{t+m} = C_t + mW + V$ é a incerteza sobre as observações futuras. Pode-se observar que à medida que m cresce (previsões mais longíquas), a incerteza sobre tais valores também cresce. Sendo assim, os intervalos de credibilidade tendem a ficar muito largos. Para este modelo, a função de previsão é dada por $f_t(m) = m_t$, que é localmente constante.

- **Modelo de crescimento linear - Modelo de ordem 2:** No modelo de crescimento linear o vetor de estados é dado por $\theta_t = (\mu_t, \beta_t)'$, onde μ_t é usualmente interpretado como o nível local (intercepto) e β_t como a taxa de crescimento local (coeficiente angular). O sinal de β_t indica a direção do crescimento da série. Para esse caso, a função de previsão é dada por

$$f_t(m) = \hat{\mu}_t + m\hat{\beta}_t. \quad (1.19)$$

Para maiores detalhes sobre o modelo de segunda ordem, o leitor interessado deve consultar West e Harrison (1997), Capítulo 7.

- **Modelo polinomial de ordem n :** Os modelos de nível local e de crescimento linear são casos especiais do modelo polinomial de ordem n . Tal modelo possui um espaço de estados n -dimensional e é descrito pelas matrizes

$$\begin{aligned} F &= (1, 0, \dots, 0); \\ G &= J_n(1); \\ W &= \text{diag}(W_1, W_2, \dots, W_n). \end{aligned} \quad (1.20)$$

Matematicamente, o modelo pode ser escrito como

$$\begin{aligned} Y_t &= \theta_{t,1} + v_t \\ \theta_{t,j} &= \theta_{n-1,j} + \theta_{t-1,j+1} + w_{t,j} \quad j = 1, \dots, n-1 \\ \theta_{t,n} &= \theta_{t-1,n} + w_{t,n}. \end{aligned}$$

Portanto, para $j = 2, \dots, n$, o j -ésimo componente do vetor de estados, em qualquer tempo t , representa o incremento da $(j-1)$ -ésima componente durante o próximo intervalo de tempo, enquanto que a primeira componente representa a resposta média, ou o nível da série (Petris et. al. 2010). A função de previsão $f_t(m)$ é dada pela equação (1.18).

1.3.2 Modelos Dinâmicos lineares sazonais via Série de Fourier

Na prática, nos deparamos com diversos casos em que os fenômenos observados possuem comportamentos cíclicos e periódicos bem definidos. Sendo assim, se faz necessário construir um modelo matemático que possa reproduzir e prever esse comportamento cíclico. No contexto dos MDL, existem duas maneiras típicas de descrever sazonalidade, através de *efeitos sazonais de forma livre* ou através de representações por *série de Fourier* (West e Harrison 1997). Neste trabalho, no entanto, iremos focar na abordagem por série de Fourier.

Seja $g(t)$ uma função real qualquer definida nos inteiros não-negativos, $t = 0, 1, \dots$. Dizemos que $g(t)$ é cíclica se, para algum $p > 1$, e para todo $t, n \geq 0$ implica em $g(t + np) = g(t)$. Assim, qualquer função periódica discreta assume valores somente no conjunto $\psi = (\psi_1, \dots, \psi_p)$, onde $\psi_t = g(i)$.

A idéia utilizada para a construção desses modelos é escrever o vetor ψ como combinação linear de funções trigonométricas necessariamente periódicas. Sendo assim, usando identidades trigonométricas, pode-se provar que quaisquer p números reais ψ_1, \dots, ψ_p podem ser representados por

$$\psi_j = a_0 + \sum_{r=1}^h [a_r \cos(\alpha r j) + b_r \text{sen}(\alpha r j)], \quad (1.21)$$

onde $\alpha = \frac{2\pi}{p}$ e h é o maior inteiro não superior a $\frac{p}{2}$. As quantidades a_r e b_r são conhecidas como coeficientes de Fourier.

Usualmente, a média da série é modelada separadamente, resultando em $a_0 = 0$. Neste caso, pode-se escrever a equação (1.21) como

$$\psi_j = \sum_{r=1}^h S_r(j), \quad (1.22)$$

em que

$$S_r(\cdot) = a_r \cos(\alpha r \cdot) + b_r \sin(\alpha r \cdot) = A_r \cos(\alpha r \cdot + \gamma_r),$$

$$A_r = (a_r^2 + b_r^2)^{1/2} \text{ e } \gamma_r = \arctan(-b_r/a_r).$$

O termo $S_r(\cdot)$ é chamado de *r-ésimo harmônico*. As quantidades A_r , αr e γ_r representam a *amplitude*, a *frequência* e a *fase* de $S_r(\cdot)$ respectivamente.

Na visão dos MDL, faz-se necessário descrever cada $S_r(t+1)$ em função de $S_r(t)$. Petris et al. (2009) mostram que os valores de *r-ésimo harmônico* $S_r(t)$ e seu respectivo conjugado $S_r^*(t)$, determinam, conjuntamente, os valores de $S_r(t+1)$ e $S_r^*(t+1)$, ou seja, cria-se um parâmetro artificial adicional, $S_r^*(t)$, ao sistema, com a finalidade de viabilizar a descrição da evolução de cada harmônico na estrutura dos modelos dinâmicos.

Quando p é par, $S_{p/2}(t+1) = -S_{p/2}(t)$, de maneira que o último harmônico simplesmente muda de sinal a cada passagem do tempo (Rodrigues, 2011). Sendo assim, a inclusão de seu conjugado no vetor de parâmetros do sistema traz redundância ao modelo. Dessa forma, a paridade de p é importante na especificação das matrizes F e G .

Sejam

$$J_2(1, \omega) = \lambda \begin{pmatrix} \cos(\omega) & \sin(\omega) \\ -\sin(\omega) & \cos(\omega) \end{pmatrix},$$

$$\mathfrak{F}_{\text{impar}} = \begin{pmatrix} E_2 \\ E_2 \\ \vdots \\ E_2 \end{pmatrix}, \mathfrak{G}_{\text{impar}} = \begin{pmatrix} J_2(1, \omega) & 0 & \dots & 0 \\ 0 & J_2(1, 2\omega) & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & J_2(1, h\omega) \end{pmatrix},$$

$$\mathfrak{F}_{\text{par}} = \begin{pmatrix} E_2 \\ E_2 \\ \vdots \\ E_2 \\ 1 \end{pmatrix}, \mathfrak{G}_{\text{par}} = \begin{pmatrix} J_2(1, \omega) & 0 & \dots & 0 & 0 \\ 0 & J_2(1, 2\omega) & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & J_2(1, (p/2 - 1)\omega) & 0 \\ 0 & 0 & \dots & 0 & -1 \end{pmatrix}.$$

Um MDL na forma de Fourier é, então, definido como sendo qualquer MDL na forma $\{\mathfrak{F}_{\text{impar}}, \mathfrak{G}_{\text{impar}}, \dots\}$ se p for ímpar, e $\{\mathfrak{F}_{\text{par}}, \mathfrak{G}_{\text{par}}, \dots\}$ se p for par.

Posteriormente, no capítulo dedicado ao pacote estatístico *dlm* do software *R*, iremos ilustrar bem como simular e ajustar o modelo dinâmico linear sazonal.

Para facilitar o entendimento dos modelos dinâmicos lineares sazonais tratou-se o caso em que os harmônicos evoluem deterministicamente no tempo. Em contrapartida, o analista pode especificar uma matriz W diferente de zero para incorporar a natureza dinâmica da sazonalidade. Neste caso, é natural pensar em W como uma matriz bloco diagonal. Vale ressaltar que a adição de erros não nulos w_t faz com que os fatores sazonais não sejam mais, de fato, periódicos. De qualquer maneira, a função de previsão m -passos a frente, independentemente da formulação estática ou dinâmica, é periódica, a depender, unicamente, de m e de $E(\theta_t|D_t) = m_t$.

1.3.3 Modelos Dinâmicos lineares de regressão

Os modelos apresentados até o momento utilizam o vetor de parâmetros θ_t para descrever a evolução temporal de uma variável de interesse Y_t . No entanto, em certos casos, podem existir covariáveis, digamos X_t , que ajudem a descrever essa evolução. Tais covariáveis são incorporadas nos MDLs através da matriz F_t . Sendo assim, o

MDL também pode ser visto como uma extensão dos modelos de regressão estáticos, uma vez que, os parâmetros são dinâmicos.

Com base nisso, considere modelar a série Y_t utilizando uma coleção de covariáveis independentes X_1, \dots, X_n . O i -ésimo valor da covariável X_t no tempo t é assumido conhecido, e é denotado por X_{it} ($i = 1, \dots, n$ e $t = 1, \dots$). Usualmente um termo constante é incluído no modelo, fazendo a primeira coluna da matriz de covariáveis igual a 1.

Para $t = 1, \dots$, seja F_t o vetor de regressão dado por $F_t' = (X_{1t}, \dots, X_{nt})$. O modelo linear dinâmico de regressão é definido pela quádrupla $\{F_t, I, V_t, W_t\}$, para alguma variância observacional V_t e matrizes de variâncias da equação de evolução W_t . Considere a estrutura do modelo dinâmico linear normal multivariado com $G_t = I$. Dessa forma, tem-se:

- **Equação das observações:**

$$Y_t = F_t' \theta_t + v_t, \quad v_t \sim N(0, V_t);$$

- **Equação do sistema:**

$$\theta_t = \theta_{t-1} + w_t, \quad w_t \sim N(0, W_t),$$

onde $\theta_t = (\theta_{1t}, \dots, \theta_{nt})$. Logo, a equação das observações pode ser escrita como

$$Y_t = \mu_t + v_t,$$

onde a média da resposta é dada por

$$\mu_t = F_t' \theta_t = \sum_{i=1}^n x_{it} \theta_{it}.$$

Para ver que essa especificação é uma generalização do modelo regressão estático, se $W_t = 0$ para todo t , então não existe um erro estocástico nos estados. Sendo assim,

$\theta_t = \theta$, ou seja, os estados são constantes no tempo. Com isso, o modelo se reduz a

$$Y_t = F_t' \theta + v_t, \quad v_t \sim N(0, V_t),$$

que nada mais é do que o modelo de regressão estático. Logo, como observado, o MDL permite que seja adicionada uma dinâmica evolutiva no tempo para o vetor dos parâmetros. Tal propriedade pode ser vista como uma grande vantagem, pois com essa estrutura pode-se modelar dados estatísticos via amostragem aleatória, ou realizar regressão no tempo uma vez que o modelo capta essa dependência temporal.

Muitos modelos autoregressivos de séries temporais como os modelos ARMA e ARIMA podem ser especificados como um MDL ao definir-se, de maneira adequada, as matrizes F_t e G_t . Maiores informações a respeito dos modelos autoregressivos, bem como outros tipos de modelos de regressão, podem ser encontrados em West e Harrison (1997), Capítulo 9.

1.4 Modelos dinâmicos com parâmetros desconhecidos

Nas seções anteriores desenvolveu-se a teoria dos MDL's supondo-se que a quádrupla $\{F_t, G_t, V_t, W_t\}$ é completamente conhecida. Isso foi feito com o objetivo de simplificar o estudo das principais propriedades dos modelos dinâmicos lineares. Na prática, é raro o conhecimento completo dessa quádrupla. Muitas vezes, as matrizes F_t e G_t , que estão relacionadas com a especificação do modelo, são totalmente conhecidas, entretanto, as matrizes de covariâncias V_t e W_t , não o são.

Nesta seção assume-se que as matrizes do modelo dependem de um vetor de parâmetros desconhecidos ψ . Usualmente, esses parâmetros são estáticos no tempo, mas existem casos em que essa matriz pode ser dinâmica ψ_t , e assim, métodos recursivos de estimação devem ser usados para sua estimação.

Nessa seção serão ilustrados a abordagem clássica e bayesiana para a estimação do vetor de parâmetros estáticos ψ .

1.4.1 Abordagem clássica - Estimação por Máxima Verossimilhança

Suponha que temos n vetores aleatórios, Y_1, \dots, Y_n , os quais suas respectivas distribuições dependem do parâmetro estático ψ . A densidade conjunta das observações, dado um valor particular de ψ , é denotada por $p(y_1, \dots, y_n|\psi)$. Considerando que, dado ψ , as observações são independentes, a função de verossimilhança é dada por

$$L(\psi) = p(y_1, \dots, y_n|\psi) = \prod_{t=1}^n p(y_t|D_{t-1}, \psi). \quad (1.23)$$

Seja $(Y_t|\psi) \sim N(f_t, Q_t)$, então a função de log-verossimilhança é dada por

$$l(\psi) = -\frac{1}{2} \sum_{t=1}^n \log|Q_t| - \frac{1}{2} \sum_{t=1}^n (y_t - f_t)' Q_t^{-1} (y_t - f_t), \quad (1.24)$$

onde f_t e Q_t dependem, implicitamente, de ψ . O estimador de máxima verossimilhança de ψ , é o valor do parâmetro que maximiza (1.23) ou (1.24), ou seja,

$$\hat{\psi} = \operatorname{argmax}_{\psi} l(\psi).$$

Sob certas condições de regularidade (Cox e Hinkley, 1974), a matriz de informação de Fisher observada, ou a inversa da matriz Hessiana de (1.24), calculada no ponto $\hat{\psi}$, fornece a matriz de covariâncias para o estimador de máxima verossimilhança de ψ .

1.4.2 Abordagem Bayesiana - Procedimentos *online* e *offline* de estimação

Formalmente, o modelo de espaço de estados consiste de uma série temporal $\{\theta_t\}_{t=1}^{\infty}$ e uma outra série temporal $\{Y_t\}_{t=1}^{\infty}$, satisfazendo as seguintes suposições (Petris et. al 2010):

(A.1) $\{\theta_t\}$ é uma cadeia de Markov.

(A.2) Condicionalmente a θ_t , os Y_t 's são independentes e dependem de θ_t , e também,

para um caso mais geral, de um vetor de parâmetros estáticos ψ .

Na abordagem Bayesiana o vetor de parâmetro ψ é tratado como um vetor aleatório, o conhecimento *a priori* sobre ψ é expresso por $p(\psi)$. Portanto, para todo $n \geq 1$, e satisfazendo as condições (A.1)-(A.2), assume-se que

$$(\theta_0, \theta_1, \dots, \theta_n, Y_1, \dots, Y_n, \psi) \sim p(\theta_0|\psi)p(\psi) \prod_{t=1}^n p(y_t|\theta_t, \psi)p(\theta_t|\theta_{t-1}, \psi). \quad (1.25)$$

Denotando-se os dados por $D_t = (y_1, \dots, y_t)$, as inferências sobre os estados θ_s no tempo s e sobre os parâmetros ψ são feitas tomando como base a distribuição *a posteriori* conjunta, isto é,

$$p(\theta_s, \psi|D_t) = p(\theta_s|D_{t-1}, \psi)p(\psi|D_{t-1}). \quad (1.26)$$

Pode-se estar interessado em descrever $P(\theta_s, \psi|D_t)$ quando, $s = t$, para problemas de *filtragem*, quando $s > t$, para problemas de *previsão*, ou quando $s < t$ para *suavização*, conforme descrito na Seção 1.1. A distribuição marginal de θ_s , dado D_t , pode ser obtida de (1.26), integrando $P(\theta_s, \psi|D_t)$ em relação a ψ . Portanto, a densidade filtrada é dada por

$$p(\theta_s|D_{t-1}) = \int p(\theta_s|D_{t-1}, \psi)p(\psi|D_{t-1})d\psi. \quad (1.27)$$

Em alguns modelos simples e usando *prioris* conjugadas, a distribuição *a posteriori* dos estados é obtida de forma fechada, aplicando-se recursivamente o teorema de Bayes. Em geral, cálculos analíticos são intratáveis, sendo assim, os métodos MCMC (estimação *offline*) e os métodos Monte Carlo Sequenciais (estimação *online*)³ aproximam, de forma eficiente, as distribuições *a posteriori* de interesse.

Nos próximos capítulos serão abordados ambos os métodos de estimação. Sendo que, para o caso *online*, nos restringiremos às especificações das matrizes W_t e V_t . O método geral de estimação recursiva, baseada em filtros de partículas será abordado posteriormente com maiores detalhes.

³Por exemplo, métodos baseados em filtros de partículas.

1.4.2.1 Especificação de W_t por fatores de desconto

A estrutura e a magnitude das matrizes de covariância W_t possuem um papel crucial na determinação de como as observações passadas influenciarão a estimação dos estados e previsão. Por simplicidade, considere W_t uma matriz diagonal. Valores grandes dos elementos da diagonal de W_t implicam em uma grande incerteza acerca da evolução dos estados, de modo que uma grande quantidade de informação da amostra é perdida na evolução de θ_{t-1} para θ_t . As últimas observações de D_{t-1} dão informações sobre θ_t , que, no entanto, tornam-se de pouca relevância na previsão θ_t .

Nas recursões do filtro de Kalman, a incerteza sobre θ_{t-1} dado D_{t-1} é resumida pela matriz de covariância condicional $V(\theta_{t-1}|D_{t-1}) = C_{t-1}$. A evolução de θ_{t-1} para θ_t através da equação de estado $\theta_t = G_t\theta_{t-1} + w_t$, implica em um aumento da incerteza, uma vez que $V(\theta_t|D_{t-1}) = R_t = G_t' C_{t-1} G_t + W_t$. Assim, se $W_t = 0$, tem-se que $R_t = \text{Var}(G_t\theta_{t-1}|D_{t-1}) = P_t$, ou seja, P_t representa a incerteza associada à projeção do sistema, livre de variações estocásticas, no tempo t . Neste sentido, W_t expressa a perda de informação na evolução de θ_{t-1} para θ_t . Devido ao componente de erro estocástico do estado na equação de evolução, a perda depende da magnitude de W_t em relação a P_t . Portanto, pode-se pensar em expressar W_t como proporção de P_t :

$$W_t = \frac{1 - \delta}{\delta} P_t, \quad (1.28)$$

onde $\delta \in (0, 1]$. Segue que $R_t = (1/\delta)P_t$. O parâmetro δ é chamado de *fator de desconto* e pode ser interpretado como a proporção da informação que se mantém entre os períodos $t - 1$ e t . Esse método foi proposto por Harrison e Scott (1965) com o objetivo de definir uma estrutura, para cada W_t , que depende unicamente de quantidades conhecidas e do fator de desconto δ , uma vez que, na abordagem *online*, a estimação de W_t é analiticamente intratável.

Se $\delta = 1$, então $W_t = 0$, e assim, não há perda de informação na evolução de θ_{t-1} para θ_t . Na prática, o valor do fator de desconto é usualmente fixo entre 0.9 e 0.99, ou é escolhido pelo diagnóstico para a seleção de modelos, por exemplo, olhando para o desempenho do modelo preditivo para diferentes valores de δ (Petris 2010). Valores abaixo de 0.8 tendem a introduzir muita incerteza, produzindo intervalos de predição muito largos. Em contrapartida, valores muito elevados tornam o sistema

com mudanças muito suaves (da-Silva et al., 2011).

1.4.2.2 Modelos Dinâmicos Lineares com V_t desconhecido

Tipicamente, os modelos dinâmicos lineares com variância observacional desconhecida, V_t , são aplicáveis quando as matrizes de covariâncias V_t são consideradas estáticas, ou seja, $V_t = V$. Tal suposição traz maior facilidade analítica e computacional para o estudo. Seja $\phi = V^{-1}$ o parâmetro de precisão das observações. Como de costume na prática Bayesiana, atribui-se *priori* Gama, ou Normal Inversa para ϕ , no caso univariado. Para o caso em que deseja-se modelar V diretamente, usualmente é atribuído a distribuição gamma inversa como *priori*. Para o caso univariado, essa distribuição é descrita por $p(\phi|D_0)$. Para o caso multivariado, a distribuição empregada como *priori* para ϕ é a correspondente da distribuição gama no caso multivariado, denominada distribuição de Wishart.

O modelo dinâmico linear, para o caso univariado, com V constante e desconhecido, pode ser representado pelas seguintes componentes (West e Harrison 1997, pág. 53):

- *Equação das observações:*

$$Y_t = \mu_t + v_t \quad v_t \sim N(0, V),$$

- *Equação do sistema:*

$$\mu_t = \mu_{t-1} + w_t \quad w_t \sim N(0, VW_t^*),$$

- *Informação inicial:*

$$\begin{aligned} \mu_0|D_0, V &\sim N(m_0, VC_0^*) \\ \phi|D_0 &\sim \text{Gamma}\left(\frac{n_0}{2}, \frac{d_0}{2}\right), \end{aligned}$$

para algum m_0, C_0^*, W_t^*, n_0 e d_0 conhecido.

De posse da estrutura do modelo apresentada acima, temos os seguintes resultados distribucionais para $t \geq 1$ (West e Harrison 1997, pág. 54):

(a) *Condicional em V:*

Defina $R_t^* = C_{t-1}^* + W_t^*$, $f_t = m_{t-1}$, $Q_t^* = R_t^* + 1$, $e_t = Y_t - f_t$ e $A_t = R_t^*/Q_t^*$.

Então

$$(\mu_{t-1}|D_{t-1}, V) \sim N(m_{t-1}, VC_{t-1}^*),$$

$$(\mu_t|D_{t-1}, V) \sim N(m_{t-1}, VR_t^*),$$

$$(Y_t|D_{t-1}, V) \sim N(f_t, VQ_t^*),$$

$$(\mu_t|D_t, V) \sim N(m_t, VC_t^*),$$

com $m_t = m_{t-1} + A_t e_t$ e $C_t^* = R_t^* - A_t^2 Q_t^* = A_t$.

(b) *Para a precisão $\phi = V^{-1}$:*

$$(\phi|D_{t-1}) \sim \text{Gamma}\left(\frac{n_{t-1}}{2}, \frac{d_{t-1}}{2}\right),$$

$$(\phi|D_t) \sim \text{Gamma}\left(\frac{n_t}{2}, \frac{d_t}{2}\right),$$

onde $n_t = n_{t-1} + 1$ e $d_t = d_{t-1} + e_t^2/Q_t^*$.

(c) *Incondicional a V:*

Defina $S_{t-1} = d_{t-1}/n_{t-1}$, $C_{t-1} = S_{t-1}C_{t-1}^*$, $R_t = S_{t-1}R_{t-1}^*$, $Q_t = S_{t-1}Q_{t-1}^*$,

$C_t = S_t C_t^*$ e $S_t = d_t/n_t$. Então

$$(\mu_{t-1}|D_{t-1}) \sim T_{n_{t-1}}(m_{t-1}, C_{t-1}),$$

$$(\mu_t|D_{t-1}) \sim T_{n_{t-1}}(m_{t-1}, R_t),$$

$$(Y_t|D_{t-1}) \sim T_{n_{t-1}}(f_t, Q_t),$$

$$(\mu_t|D_t) \sim T_{n_t}(m_t, C_t),$$

(d) *Definição operacional das eqs. de evolução:* Defina $Q_t = R_t + S_{t-1}$ e $A_t = R_t/Q_t$.

Então

$$\begin{aligned}
m_t &= m_{t-1} + A_t e_t, \\
C_t &= (S_t/S_{t-1})[R_t - A_t^2 Q_t] = A_t S_t, \\
n_t &= n_{t-1} + 1, \\
d_t &= d_{t-1} + S_{t-1} e_t^2 / Q_t, \\
S_t &= d_t / n_t.
\end{aligned}$$

As demonstrações são encontradas em West e Harrison (1997), págs. 54 e 55. Todos os passos da derivação Bayesiana do modelo Normal-Gamma são encontrados em West e Harrison (1997), Seção 17.3.

1.4.2.3 Um modelo com fator de desconto para V_t variável no tempo

A condição de variância constante $V_t = V$ pode não ser realista. Sendo assim, a aplicação de fatores de desconto, descrita para especificação da matriz W_t , também pode ser aplicada ao parâmetro de precisão ϕ ou para a matriz V_t . Nessa seção, essa abordagem será dada resumidamente.

Considere o MDL desenvolvido na seção anterior. Suponha que, no tempo $t - 1$, a distribuição do parâmetro de precisão seja dada por

$$(\phi_{t-1} | D_{t-1}) \sim \text{Gamma}(n_{t-1}/2, d_{t-1}/2). \quad (1.29)$$

Atualizando para o tempo t , é desejável reter a forma da distribuição gamma para $p(\phi_t | D_{t-1})$, uma vez que é conjugada com a verossimilhança para a evolução do sistema baseado na próxima observação Y_t . Esta restrição levou ao desenvolvimento de um método denominado *variance discounting* para modelar um decaimento da informação sobre a precisão, e, portanto a variância entre pontos no tempo, mantendo a forma gamma para as distribuições *a priori* e *posteriori*. Com base na *posteriori* no tempo $t - 1$, equação (1.29), suponha que ϕ_t é obtido a partir de ϕ_{t-1} por um modelo

de passeio aleatório, resultando na distribuição *a priori* do tempo t

$$(\phi_t|D_{t-1}) \sim \text{Gamma} \left(\frac{\delta n_{t-1}}{2}, \frac{\delta d_{t-1}}{2} \right),$$

onde $0 < \delta < 1$. Note que o valor esperado não muda $E(\phi_t|D_{t-1}) = E(\phi_{t-1}|D_{t-1}) = n_{t-1}/d_{t-1}$, enquanto que a variância é maior $V(\phi_t|D_{t-1}) = (1/\delta)V(\phi_{t-1}|D_{t-1})$. Maiores detalhes podem ser encontrados em West e Harrison (1997) Seção 10.8. As fórmulas de recursão do filtro de Kalman com fatores de desconto na variância são encontrados em West e Harrison (1997), pág. 362.

No próximo capítulo será descrito o pacote `dlm` do software R. Tal pacote contém funções que permitem a simulação, o ajuste e previsões utilizando os modelos dinâmicos lineares normais, conforme descrito em Petris et al. (2010).

Capítulo 2

O pacote *dlm*

Os modelos dinâmicos lineares são representados no pacote *dlm* como listas de nomes como uma classe de atributos, o que torna-os objetos da classe '*dlm*' (R development, 2011). Objetos de classe *dlm* podem representar MDL's constantes ou variantes no tempo. Um MDL constante é completamente especificado, uma vez que as matrizes de F, V, L, W, C_0 , e o vetor m_0 são dados. No *R*, estes componentes são armazenados em um objeto *dlm* como elementos de FF, V, GG, W, C_0 , e m_0 , respectivamente. O pacote também oferece várias funções que criam determinadas classes de MDL's, exigindo a especificação de poucas características (*inputs*) pelo usuário.

O MDL geral univariado ou multivariado pode ser especificado usando a função *dlm*. Esta função cria um objeto *dlm* a partir de seus componentes, realizando algumas verificações nos *inputs*, tais como testar as dimensões das matrizes para a consistência. O *input* pode ser dado como uma lista de argumentos ou apenas com um argumento. O pacote *dlm* permite fazer simulações, filtragens e suavizações.

Para ilustrar o uso do pacote, apresentam-se 3 exemplos com os códigos respectivos descritos no texto. No primeiro exemplo, simula-se um modelo dinâmico linear polinomial, no segundo simula-se um modelo dinâmico linear sazonal e por último, simula-se um modelo dinâmico linear de regressão. Maiores informações sobre modelos dinâmicos utilizando o software *R*, além de outras funções implementadas no pacote *dlm*, podem ser consultadas em Petris et. al. (2010).

2.1 Exemplo 1 - MDL Polinomial

Nesta seção apresenta-se a simulação de um modelo dinâmico linear polinomial de segunda ordem. Aqui fixamos os parâmetros do modelo como:

- $m_0 = (300, 0)$
- $C_0 = \text{diag}(1, 2)$
- $V = 1000$
- $W = (10, 1)$

Para definir o modelo utilizou-se a função *dlmModPoly*. Para simular desse modelo utilizou-se a função *dlmForecast* (Petris et. al. 2010). De posse dos dados simulados, utilizou-se a função *dlmFilter* para estimar o nível μ_t , para $t = 1, \dots, 100$. Através da função *dlmFilter*, é possível fazer previsões um passo a frente para a série simulada Y_t . Observando os valores fixados para os parâmetros, vale ressaltar que escolheu-se uma *priori* vaga $(\theta_0|D_0) \sim N_2(m_0, C_0)$. Por fim, utilizamos a função *dlmSmooth* para suavizar os valores do nível.

A programação utilizada é apresentada abaixo. A Figura 2.1 mostra a série histórica y_t em cinza, a média μ_t em vermelho, as estimativas em linhas pretas e os intervalos de credibilidade de 95% em linhas tracejadas para os modelos simulados.

MDL Polinomial

```
1 #Modelo dinâmico polinomial
3 #Bloco 1: Geração dos dados
  require(dlm)
5
  set.seed=123456
7 N <- 100
  m0=rnorm(2,c(300,0))
9 mod <- dlmModPoly(order=2, dW=c(10,1), dV=1000, m0=m0, C0=diag(1,2))
11 aux <- dlmForecast(mod, nAhead=N, sampleNew=1)
  y <- as.ts(aux$newObs[[1]])
13 mu.real <- as.ts(aux$newStates[[1]])
15 #Bloco 2: Filtragem
```

```

17 mod$m0 <- rep(0,2)
   mod$C0 <- 10000
19 filtro <- dlmFilter(y,mod)
   C <- dlmSvd2var(filtro$U.C, filtro$D.C)
21 c1 <- sapply(2:(N+1), function(x) sqrt(C[[x]][1,1]))
   z <- qnorm(0.975)
23 mu.filtro <- dropFirst(filtro$m[,1])
   lim.inf.C <- mu.filtro-z*c1
25 lim.sup.C <- mu.filtro+z*c1

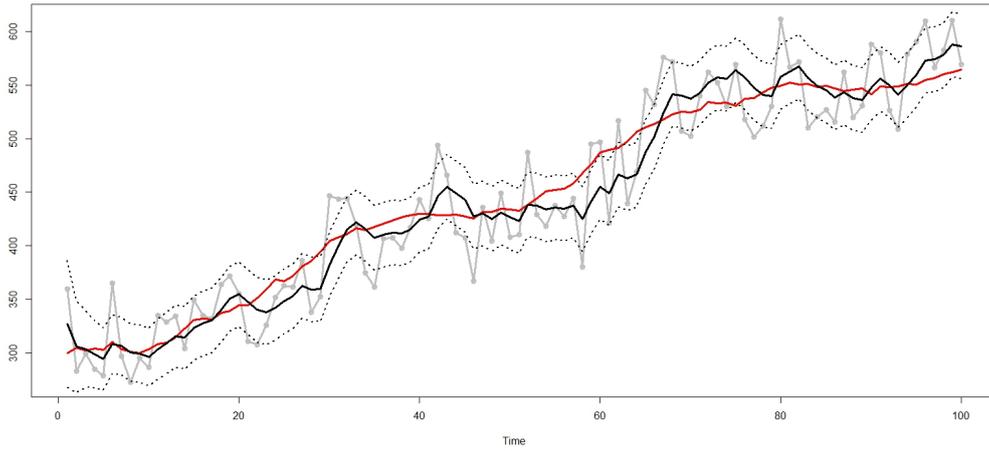
27 plot(y,type="o", lwd=3, col="gray", main="", ylab="")
   lines(mu.real[,1], col="red", lwd=3)
29 lines(mu.filtro, col="black", lwd=3)
   lines(lim.inf.C, col="black", lwd=2, lty=3)
31 lines(lim.sup.C, col="black", lwd=2, lty=3)

33 #Bloco 3: Previsão um passo a frente

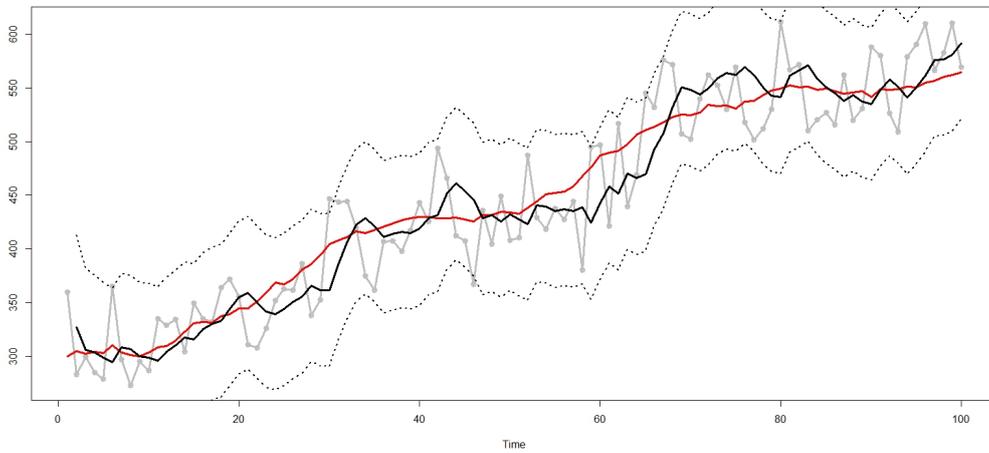
35 raiz.Q <- dropFirst(residuals(filtro)$sd)
   y.previsto <- dropFirst(filtro$f[,1])
37 lim.inf.Q <- y.previsto-z*raiz.Q
   lim.sup.Q <- y.previsto+z*raiz.Q
39
   plot(y,type="o", lwd=3, col="gray", main="", ylab="")
41 lines(mu.real[,1], col="red", lwd=3)
   lines(y.previsto, col="black", lwd=3)
43 lines(lim.inf.Q, col="black", lwd=2, lty=3)
   lines(lim.sup.Q, col="black", lwd=2, lty=3)
45
   #bloco 4: Suavização
47
   suave <- dlmSmooth(filtro)
49 S <- dlmSvd2var(suave$U.S, suave$D.S)
   s1 <- sapply(2:(N+1), function(x) sqrt(S[[x]][1,1]))
51 mu.suave <- dropFirst(suave$s[,1])
   lim.inf.S <- mu.suave-z*s1
53 lim.sup.S <- mu.suave+z*s1

55 plot(y,type="o", lwd=3, col="gray", main="", ylab="")
   lines(mu.real[,1], col="red", lwd=3)
57 lines(mu.suave, col="black", lwd=3)
   lines(lim.inf.S, col="black", lwd=2, lty=3)
59 lines(lim.sup.S, col="black", lwd=2, lty=3)

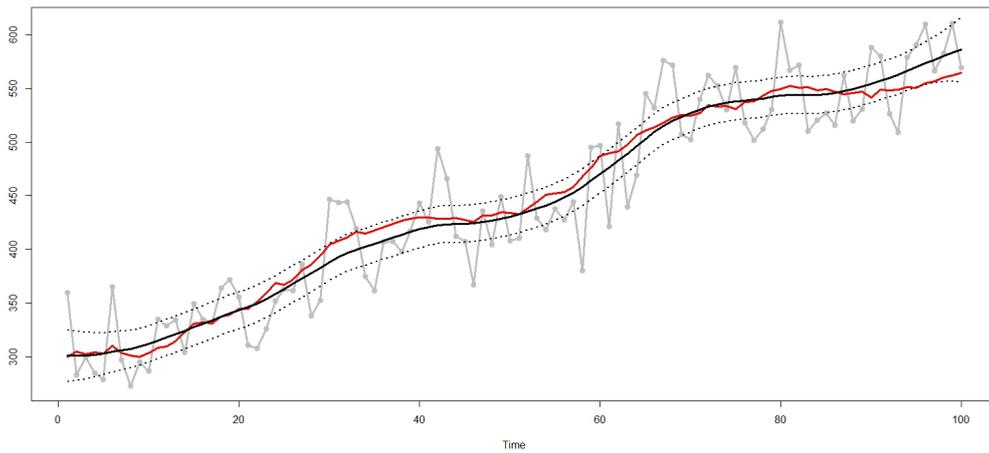
```



(a) Filtragem



(b) Previsão um passo a frente



(c) Suavização

Figura 2.1: Série histórica y_t em cinza, média real μ_t em vermelho, estimativas em linhas pretas e intervalo de credibilidade de 95% em linhas tracejadas. De cima para baixo temos: Valores filtrados, valores previstos um passo a frente e valores suavizados para μ_t respectivamente.

2.2 Exemplo 2 - MDL Sazonal

Para exemplificar os principais aspectos metodológicos discutidos na Seção 1.3.2 do Capítulo 1 sobre os modelos dinâmicos lineares sazonais na forma de Fourier, ilustraremos como utilizar o pacote *dlm* para simular e ajustar dados sazonais. Para tanto, considere o MDL definido pelas seguintes matrizes

$$\left\{ F = (1, 0, 1, 0)', G = \begin{pmatrix} J_2(1, 2\pi/5) & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & J_2(1, 4\pi/5) \end{pmatrix}, W = 10^{-4} \times I_4, V = 20 \right\}.$$

Esta especificação retrata um modelo com 5 períodos sazonais. Geramos uma série de tamanho 50, conforme descrito no Bloco 1 da programação abaixo. Utilizando uma priori vaga para o sistema, estimamos as médias μ_1, \dots, μ_{50} utilizando os dados simulados. A programação completa é apresentada abaixo. A Figura 2.2 mostra a série histórica y_t em cinza, a média μ_t em vermelho, as estimativas em linhas pretas e os intervalos de credibilidade de 95% em linhas tracejadas para os modelos simulados. Note que os valores estimados pelo procedimento inferencial foram bem próximos dos valores reais.

MDL Sazonal

```
1 #Modelos dinâmicos lineares sazonais

3 par(mfrow=c(3,1))
  set.seed=123456

5
  #Bloco 1: Geração dos dados

7
  N <- 50

9 p <- 5
  mod <- dlmModTrig(s=p, dW=rep(1e-4, p-1), dV=20, m0=rnorm(p-1), CO=diag(1,p-1))

11 aux <- dlmForecast(mod, nAhead=N, sampleNew=1)
  y <- aux$newObs[[1]]

13 mu.real <- aux$newStates[[1]]

15 #Bloco 2: Filtragem

17 mod$m0 <- rep(0,p-1)
  mod$CO <- diag(1e+7,p-1)

19 filtro <- dlmFilter(y,mod)
  C <- dlmSvd2var(filtro$U.C, filtro$D.C)

21 c1 <- sapply(2:(N+1), function(x) sqrt(mod$FF% *% C[[x]]% *% t(mod$FF)))
```

```

z <- qnorm(0.975)
23 mu.filtro <- dropFirst(filtro$m[,1])
lim.inf.C <- mu.filtro-z*c1
25 lim.sup.C <- mu.filtro+z*c1

27 plot(y,type="b", lwd=3, col="gray", main="", ylab="")
lines(mu.real[,1], col="red", lwd=3)
29 lines(mu.filtro, col="black", lwd=3)
lines(lim.inf.C, col="black", lwd=2, lty=3)
31 lines(lim.sup.C, col="black", lwd=2, lty=3)

33 #Bloco 3: Previsão um passo a frente

35 raiz.Q <- dropFirst(residuals(filtro)$sd)
y.previsto <- dropFirst(filtro$f)[,1]
37 lim.inf.Q <- y.previsto-z*raiz.Q
lim.sup.Q <- y.previsto+z*raiz.Q
39

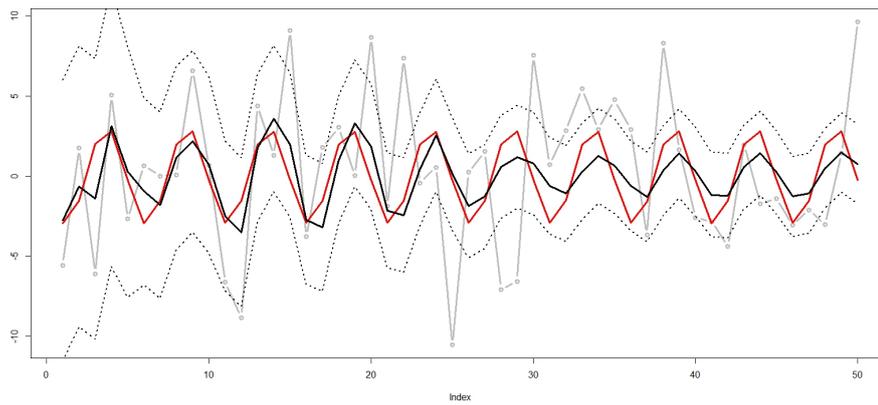
plot(y,type="b", lwd=3, col="gray", main="", ylab="", ylim=c(-20,20))
41 lines(mu.real[,1], col="red", lwd=3)
lines(y.previsto, col="black", lwd=3)
43 lines(lim.inf.Q, col="black", lwd=2, lty=3)
lines(lim.sup.Q, col="black", lwd=2, lty=3)
45

#Bloco 4: Suavização
47

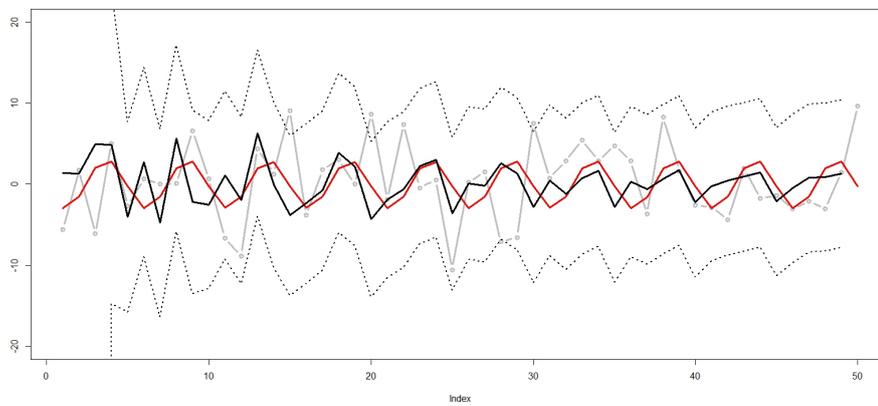
mod$m0 <- rep(0,p-1)
49 mod$C0 <- diag(1e+7,p-1)
aux <- dlmSmooth(y,mod)
51 mu.suave <- dropFirst(aux$s)
S <- dlmSvd2var(aux$U.S, aux$D.S)
53 z <- qnorm(0.975)
tmp <- sapply(2:(N+1), function(x) sqrt(mod$FF% *% S[[x]]% *% t(mod$FF)))
55 mu.suave <- mu.suave\% *%\% t(mod$FF)
lim.inf.S <- mu.suave-z*tmp
57 lim.sup.S <- mu.suave+z*tmp

59 plot(y,type="b", lwd=3, col="gray", main="", ylab="")
lines(mu.real[,1], col="red", lwd=3)
61 lines(mu.suave, col="black", lwd=3)
lines(lim.inf.S, col="black", lwd=2, lty=3)
63 lines(lim.sup.S, col="black", lwd=2, lty=3)

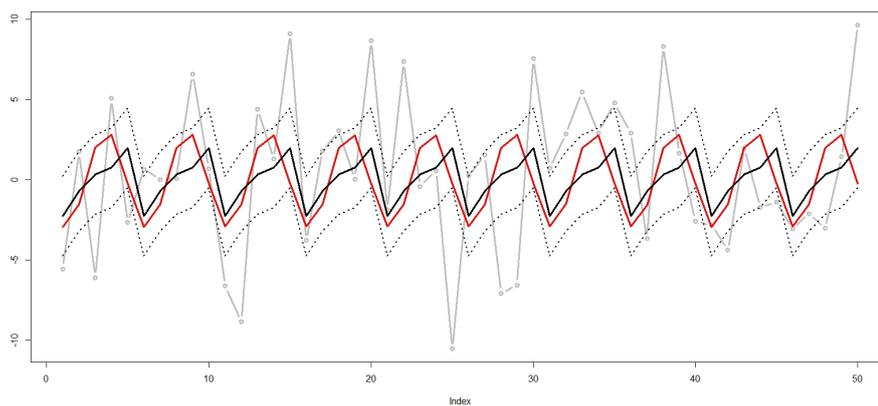
```



(a) Filtragem



(b) Previsão um passo a frente



(c) Suavização

Figura 2.2: Série histórica y_t em cinza, média real μ_t em vermelho, estimativas em linhas pretas e intervalo de credibilidade de 95% em linhas tracejadas. De cima para baixo temos: Valores filtrados, valores previstos um passo a frente e valores suavizados para μ_t respectivamente.

2.3 Exemplo 3 - MDL de Regressão

Nesta seção simula-se um modelo dinâmico linear de regressão. Para tanto, simula-se uma regressão linear estática simples com β_0 e β_1 iguais à 1 e -2 respectivamente. Os valores da covariável são gerados segundo uma distribuição normal padrão, ou seja $x_t \sim N(0, 1)$. Logo, a matriz de planejamento do modelo é dada por

$$\mathbf{F} = [\mathbf{1} \quad \mathbf{x}].$$

Fixando $\sigma^2 = 4$, o erro foi gerado como $\epsilon_t \sim N(0, \sigma^2)$. Com isso, y_t foi gerado utilizando-se

$$y_t = \beta_0 + \beta_1 x_t + \epsilon_t,$$

ou ainda,

$$\mathbf{y} = \mathbf{F}'\beta + \epsilon.$$

Feito isso, com o intuito de verificar se o MDL pode ser usado para ajustar modelos de regressão estáticos, os parâmetros foram estimados de maneira convencional, utilizando o método de mínimos quadrados ordinários (função *lm*). Após a estimação, esses valores foram usados para comparação com os valores estimados pelo MDL.

A função *dmlFilter* foi utilizada para estimar o mesmo modelo com base no filtro de Kalman. O código utilizado para elaborar esse exemplo é apresentado abaixo. Note que os valores estimados para os parâmetros são exatamente os mesmos para as diferentes funções, com diferença entre as matrizes de variâncias estimadas. Assim, fica claro que o MDL pode ser usado tanto para estimação de modelos de regressão estáticos quanto dinâmicos.

MDL de Regressão

```
1  #Modelo dinâmico de Regressão

3  #Bloco 1: Geração dos dados

5  N <- 100
   X <- rnorm(N)
7  F <- cbind(rep(1,N),X)
   beta.real <- c(1,-2)
9  epsilon <- rnorm(N,0,2)

11 y <- F% *% beta.real+epsilon

13 #Bloco 2: Estimação via MQO

15 reg.lm <- lm(y~X)
   coef(reg.lm)
17 vcov(reg.lm)

19 #Bloco 3: Estimação via MDL

21 mod <- dlmModReg(X=X)
   filtro <- dlmFilter(y,mod)
23 filtro$m[N+1,]
   dlmSvd2var(filtro$U.C, filtro$D.C)[[N+1]]
25

   #Bloco 4: Resultados
27 > reg.lm <- lm(y~X)
   > coef(reg.lm)
29 (Intercept)          X
   1.140798    -2.036467
31 > vcov(reg.lm)
               (Intercept)          X
33 (Intercept) 0.056649826 0.005989684
   X          0.005989684 0.047692976
35 > mod <- dlmModReg(X=X)
   > filtro <- dlmFilter(y,mod)
37 > filtro$m[N+1,]
   [1] 1.140798 -2.036467
39 > dlmSvd2var(filtro$U.C, filtro$D.C)[[N+1]]
               [,1]          [,2]
41 [1,] 0.010134574 0.001071546
   [2,] 0.001071546 0.008532206
```

Capítulo 3

Modelos Dinâmicos Lineares Generalizados

A classe dos modelos dinâmicos lineares generalizados (MDLG) foi introduzido por West et. al. (1985). Nessa classe a suposição de normalidade das observações é relaxada, assumindo-se que a distribuição das observações pertence à família exponencial de Nelder e Weddeburn (1972), ou seja, sua função de probabilidade pode ser escrita da seguinte forma

$$p(Y_t|\eta_t, \phi_t) = \exp \{ \phi_t [Y_t \eta_t - b(\eta_t)] \} c(Y_t, \phi_t),$$

em que as funções $b(\cdot)$ e $c(\cdot, \cdot)$ são funções conhecidas, η_t é o parâmetro natural ou canônico e $\phi_t = V_t^{-1}$ é o parâmetro de precisão da distribuição. Usando propriedades da *função escore*¹ é possível mostrar que

$$E(Y_t|\eta_t, \phi_t) = b'(\eta_t) = \mu_t,$$

e

$$V(Y_t|\eta_t, \phi_t) = \frac{b''(\eta_t)}{\phi_t}.$$

Para descrever um MDLG, considere uma série temporal Y_t univariada com $\phi_t = \phi$ conhecido. O modelo dinâmico linear generalizado é caracterizado pelos seguintes componentes:

¹Primeira derivada da função de log-verossimilhança.

- **Equação das observações:**

$$Y_t | \eta_t = \exp(\phi_t [t(y_t)\eta_t - b(\eta_t) + c(y_t, \phi_t)]); \quad t = 1, 2, \dots \quad (3.1)$$

- **Função de Ligação:**

$$g(\eta_t) = g(E(Y_t | \eta_t)) = \mathbf{F}'_t \theta_t = \lambda_t; \quad (3.2)$$

- **Equação do Sistema:**

$$\theta_t = \mathbf{G}'_t \theta_{t-1} + w_t; \quad w_t \sim (0, \mathbf{W}_t); \quad (3.3)$$

- **Informação inicial:**

$$\theta_0 | D_0 \sim (m_0, C_0). \quad (3.4)$$

Note que, diferentemente do caso normal, não há especificação funcional da distribuição de probabilidade do componente de erro da equação do sistema, w_t , nem da informação inicial θ_0 . Para essa modelagem faz-se apenas, a especificação de seus momentos. Aqui é assumido que, dado η_t , as observações Y_t são independentes entre si e dos erros de evolução w_t .

Note, também, que da equação (3.2), existe uma relação determinística entre μ_t , η_t e λ_t , ou seja, conhecido o valor de uma dessas componentes pode-se determinar os valores das outras.

Este capítulo se divide da seguinte forma: A Seção 3.1 ilustra o passo a passo do processo de estimação para os modelo lineares generalizados. Na Seção 3.2 tal processo de estimação é exemplificado para o modelo dinâmico binário, o qual assume que a variável resposta segue uma distribuição Binomial. Por fim, na seção 3.3, descreve-se o modelo dinâmico Poisson e seu processo de estimação, o qual foi objeto de estudo nessa dissertação.

3.1 Processo de estimação

Assume-se que as matrizes \mathbf{F}_t , \mathbf{G}_t , \mathbf{W}_t e o parâmetro de precisão ϕ_t são conhecidos. O processo inferencial tem o mesmo carácter sequencial dos DLM. Porém, alguns passos adicionais são necessários na estimação dos parâmetros do modelo. O procedimento inferencial pode ser descrito da seguinte forma:

- **Reconhecimento dos momentos a *priori*:**

A distribuição a posteriori de θ_{t-1} tem vetor de médias \mathbf{m}_{t-1} e matriz de covariâncias \mathbf{C}_{t-1} . Portanto, pela equação de evolução, tem-se $(\theta_{t-1}|D_{t-1}) \sim (a_t, R_t)$, onde $a_t = \mathbf{G}_{t-1}\mathbf{m}_{t-1}$ e $R_t = G_{t-1}C_{t-1}G'_{t-1} + W_t$. Como $\lambda_t = F'_t\theta_t$, a distribuição do preditor linear é parcialmente especificada por

$$(\lambda_t|D_{t-1}) \sim (f_t, q_t), \quad (3.5)$$

com $f_t = F'_t a_t$ e $q_t = F'_t R_t F_t$.

- **Especificação da *priori*:**

Segundo West e Harrison (1997), a distribuição conjugada na escala de η_t possui forma fechada e é normalmente a melhor opção. Na prática porém, a especificação da distribuição a *priori* tem que ser analisada caso a caso. Por exemplo, da-Silva et. al. (2011), ao desenvolverem o modelo dinâmico Beta, mostram que há vantagens em se definir uma *priori* não conjugada.

Suponha, sem perda de generalidade, que o analista optou por estipular a distribuição a *priori*, definida pelos hiperparâmetros r_t e s_t , diretamente para a média μ_t . O próximo passo do ciclo inferencial é, então, elicitar os valores de r_t e s_t em conformidade com a relação $g(\mu_t) = \lambda_t$ e com a equação (3.5). Isso é, resolve-se o seguinte sistema de equações

$$\begin{aligned} f_t &= E(g(\eta_t|D_{t-1})) = h_1(r_t, s_t) \\ q_t &= V(g(\eta_t|D_{t-1})) = h_2(r_t, s_t). \end{aligned}$$

- **Atualização da distribuição de η_t :**

Uma vez obtida a observação Y_t , a distribuição a *posteriori* de μ_t é obtida via Teorema de Bayes

$$p(\mu_t|D_t, \phi_t) \propto p(y_t|\mu_t, D_{t-1}, \phi_t)p(\mu_t|D_{t-1}) \quad (3.6)$$

- **Atualização dos momentos de λ_t :**

Uma vez atualizado os dois primeiros momentos de $(\eta_t|D_t)$, faz-se o caminho inverso para obter os momentos correspondentes, f_t^* e q_t^* , da *posteriori* de λ_t . Ou seja, resolve-se o sistema de equações

$$\begin{aligned} E(\lambda_t|D_t) &= f_t^* = E(g(\eta_t|D_t)) \\ V(\lambda_t|D_t) &= q_t^* = V(g(\eta_t|D_t)). \end{aligned}$$

- **Atualização dos momentos de θ_t :**

Para calcular os momentos (m_t, C_t) usa-se o estimador conhecido como *linear Bayes*. Sendo a distribuição a *priori* conjunta de $\Psi_t = (\theta_t, \lambda_t)$ parcialmente especificada por

$$(\Psi_t|D_{t-1}) \sim \left[\begin{pmatrix} f_t \\ a_t \end{pmatrix}, \begin{pmatrix} q_t & F_t' R_t \\ R_t F_t & R_t \end{pmatrix} \right], \quad (3.7)$$

a estimação dos momentos de $(\theta_t|\lambda_t, D_{t-1})$ via linear Bayes é dada por

$$\hat{E}(\theta_t|\lambda_t, D_{t-1}) = a_t + R_t F_t (\lambda_t - f_t) / q_t; \quad (3.8)$$

$$\hat{V}(\theta_t|\lambda_t, D_{t-1}) = R_t - R_t F_t F_t' R_t / q_t. \quad (3.9)$$

Pelo teorema de Bayes, a distribuição a *posteriori* de θ_t é obtida pela relação

$$P(\theta_t|D_t) = \int P(\theta_t|\lambda_t, D_{t-1})P(\lambda_t|D_t)d\lambda_t \quad (3.10)$$

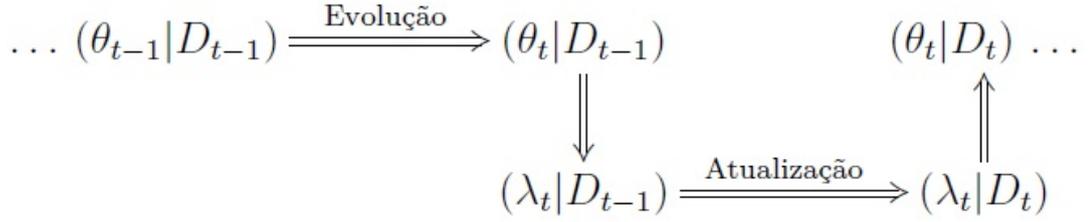


Figura 3.1: Análise sequencial do processo de estimação do modelo dinâmico linear generalizado

Via regra de Bayes, $P(\theta_t|D_t)$ não tem forma fechada conhecida, mas seus momentos são estimados por

$$\begin{aligned}
m_t &= E(\theta_t|D_t) \\
&= E[\hat{E}(\theta_t|\lambda_t, D_{t-1})|D_t] \\
&= a_t + R_t F_t (f_t^* - f_t)/q_t.
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
C_t &= V(\theta_t|D_t) \\
&= V[\hat{E}(\theta_t|\lambda_t, D_{t-1})] + E[\hat{V}(\theta_t|\lambda_t, D_{t-1})|D_t] \\
&= R_t - R_t F_t F_t' R_t (1 - q_t^*/q_t)/q_t.
\end{aligned} \tag{3.12}$$

A Figura 3.1 mostra, de maneira ilustrativa, a análise sequencial do processo inferencial associado ao modelo dinâmico linear generalizado.

3.2 Exemplo: Modelo Dinâmico Binário

Nessa subseção ilustramos um exemplo para o caso Binomial com a finalidade de tornar claro o exposto acima. Seja $Y_t \sim Bin(n_t, \pi_t)$

$$P(Y_t|\pi_t) = \exp\left(y_t \log\left(\frac{\pi_t}{1 - \pi_t}\right) + n_t \log(1 - \pi_t)\right) \binom{n_t}{y_t} \tag{3.13}$$

Pode-se reescrever $P(Y_t|\pi_t)$ na forma da família exponencial de Nelder e Wedderburn

$$\begin{aligned} P(Y_t|\eta_t) &= \exp(y_t\eta_t - n_t \log(1 + e^{\eta_t})) \binom{n_t}{y_t} \\ &= h(y_t) \exp(\phi_t^{-1}(y_t\eta_t - b(\eta_t)) + c(y_t, \phi_t)), \end{aligned} \quad (3.14)$$

com $b(\eta_t) = n_t \log(1 + e^{\eta_t})$, $\mu_t = n_t \left(\frac{e^{\eta_t}}{1 + e^{\eta_t}}\right)$ e $\eta_t = \log\left(\frac{\mu_t}{n_t - \mu_t}\right)$.

Com isso, a função de ligação é dada por

$$g(\eta_t) = \eta_t = \log\left(\frac{\pi_t}{1 - \pi_t}\right) = F'_t \theta_t \quad (3.15)$$

A única incerteza em relação à distribuição de Y_t dado a informação passada D_{t-1} , é devido à incerteza com respeito a η_t . A *priori* conjugada para η_t também é da família exponencial. Portanto, possui densidade da forma

$$P(\eta_t|D_{t-1}) = c(r_t, s_t) \exp(r_t \eta_t - s_t b(\eta_t)). \quad (3.16)$$

Os parâmetros r_t e s_t da *priori* conjugada devem satisfazer às seguintes condições:

$$\begin{aligned} f_t &= E(\eta_t|D_{t-1}) = E(\log(\mu_t(1 - \mu_t)^{-1}|D_{t-1})) \\ &= E(\log(\mu_t)|D_{t-1}) - E(\log(1 - \mu_t)|D_{t-1}) \\ &= \psi(s_t) - \psi(r_t), \end{aligned} \quad (3.17)$$

$$\begin{aligned} e \quad q_t &= V(\eta_t|D_{t-1}) = V(\log(\mu_t)|D_{t-1}) + V(\log(1 - \mu_t)|D_{t-1}) \\ &= \psi'(s_t) + \psi'(r_t), \end{aligned} \quad (3.18)$$

onde $\psi(z) = \frac{\Gamma'(z)}{\Gamma(z)}$ é a função digama.

Para valores grandes de r_t e s_t , $\psi \approx \log(z)$, enquanto que $\psi'(z) \approx z^{-1}$ (vide West e Harrison 1997). Desta forma, tem-se

$$f_t \approx \log\left(\frac{s_t}{r_t}\right) \text{ e } q_t \approx \frac{1}{s_t} + \frac{1}{r_t}.$$

Desta forma, os parâmetros da *priori* conjugada $Beta(r_t, s_t)$ têm a seguinte forma:

$$s_t \approx [1 + \exp(f_t)]/q_t \text{ e } r_t \approx [1 + \exp(-f_t)]/q_t.$$

Reparametrizando o modelo $P(Y_t|\pi_t)$ em função da média μ_t :

$$\begin{aligned} P(Y_t|\mu_t) &\propto \mu_t^{y_t}(1 - \mu_t)^{n_t - y_t} \\ &\propto \exp(y_t \log \mu_t + (n_t - y_t) \log(1 - \mu_t)), \end{aligned} \quad (3.19)$$

a *priori* conjugada pode ser reescrita, também, em função de μ_t

$$P(\mu_t|D_{t-1}) \propto \mu_t^{s_t-1}(1 - \mu_t)^{r_t-1}. \quad (3.20)$$

Logo $(\mu_t|D_{t-1}) \sim \text{Beta}(s_t, r_t)$. Portanto, a *posteriori* $P(\mu_t|D_t)$ é dada por

$$P(\mu_t|D_t) \propto \mu_t^{y_t+s_t-1}(1 - \mu_t)^{n_t+r_t-y_t-1}. \quad (3.21)$$

Logo, $(\mu_t|D_t) \sim \text{Beta}(y_t + s_t, n_t + r_t - y_t)$. A previsão 1 passo a frente é dada por

$$\begin{aligned} P(Y_t|D_{t-1}) &= \int P(Y_t|\mu_t, D_{t-1})P(\mu_t|D_{t-1})d\mu_t \\ &= \binom{n_t}{y_t} \frac{\Gamma(s_t + r_t)}{\Gamma(s_t)\Gamma(r_t)} \frac{\Gamma(y_t + s_t)\Gamma(n_t + r_t - y_t)}{\Gamma(n_t + s_t + r_t)}. \end{aligned} \quad (3.22)$$

Desta forma, $(Y_t|D_{t-1})$ segue a distribuição Beta-Binomial. Podemos resumir o exemplo acima da seguinte maneira:

- Priori para μ_t : $\mu_t|D_{t-1} \sim \text{Beta}(s_t, r_t)$
- previsão a 1 passo a frente:

$$P(Y_t|D_{t-1}) = \binom{n_t}{y_t} \frac{\Gamma(s_t + r_t)}{\Gamma(s_t)\Gamma(r_t)} \frac{\Gamma(y_t + s_t)\Gamma(n_t + r_t - y_t)}{\Gamma(n_t + s_t + r_t)}; \quad (3.23)$$

- Posteriori para μ_t : $(\mu_t|D_{t-1}) \sim \text{Beta}(y_t + s_t, n_t + r_t - y_t)$.

3.3 Modelo Dinâmico Poisson

Nesta seção será detalhado o modelo dinâmico Poisson, uma vez que tal modelo é objeto de estudo neste trabalho de dissertação. Considere Y_1, \dots, Y_t contagens de um determinado evento, com $Y_t \sim Poisson(\lambda_t)$. Vale ressaltar que, dado λ_t , as contagens Y_t são independentes, ou seja, as contagens são condicionalmente independentes. O modelo dinâmico Poisson é caracterizado pelas seguintes componentes

- **Equação das observações:**

$$P(y_t|\lambda_t) = \frac{\lambda_t^{y_t} e^{-\lambda_t}}{y_t!};$$

- **Distribuição a *priori*:**

$$(\lambda_t|D_{t-1}) \sim Gamma(\alpha_t, \beta_t);$$

- **Função de ligação: Logarítmica**

$$\eta_t = \log(\lambda_t) = F_t' \theta_t;$$

- **Equação do sistema:**

$$\theta_t = G_t' \theta_{t-1} + w_t; \quad w_t \sim (0, W_t);$$

- **Informação inicial:**

$$\theta_0|D_0 \sim (m_0, C_0).$$

Note que distribuição a *priori* para λ_t é conjugada com a distribuição a *posteriori*, ou seja, a distribuição a *posteriori* de λ_t é da família Gamma, conforme será visto na próxima seção. A escolha da *priori* conjugada implica em uma grande facilidade computacional e, usualmente, é a melhor escolha (West e Harrison 1997). A função de ligação *log* é uma função de ligação canônica que na teoria dos MLG também traz vantagens computacionais, principalmente no processo de estimação (Nelder e McCullagh 1989).

Uma característica do modelo de Poisson é que $E(Y_t|\lambda_t) = Var(Y_t|\lambda_t) = \lambda_t$, o que pode ser visto como uma limitação, pois dados de contagem podem apresentar variabilidade maior do que a média. Sendo assim, o modelo Poisson não é capaz de capturar esse excesso de variabilidade também conhecida como *superdispersão* ou *sobredispersão*. Portanto toda variação observada que excede aquela predita pelo modelo é denominada *superdispersão* ou *sobredispersão*. Segundo Hinde e Demétrio (1998) a superdispersão pode ser causada de várias maneiras, tais como: variabilidade do experimento, correlação entre respostas individuais, amostragem por *cluster*, agregação em dados de nível ou omissão de variáveis não observadas.

Para contornar o problema da superdispersão, outros modelos foram introduzidos na literatura, como o modelo dinâmico binomial negativo e o modelo dinâmico Poisson-Lognormal. No primeiro destes, é introduzido um termo aleatório δ_t multiplicando o parâmetro λ_t , ou seja, $Y_t \sim Poisson(\lambda_t \delta_t)$. Sendo assim, o modelo dinâmico binomial negativo é caracterizado pelas seguintes componentes:

- **Equação das observações:**

$$(Y_t|\lambda_t, \delta_t) \sim Poisson(\lambda_t \delta_t)$$

- **Distribuição a priori:**

$$\begin{aligned} (\lambda_t|D_{t-1}) &\sim Gamma(\alpha_t, \beta_t); \\ \delta_t &\sim Gamma(\epsilon, \epsilon); \end{aligned} \tag{3.24}$$

- **Função de ligação: Logarítmica**

$$\eta_t = \log(\lambda_t) = F_t' \theta_t;$$

- **Equação do sistema:**

$$\theta_t = G_t' \theta_{t-1} + w_t; \quad w_t \sim (0, W_t);$$

- **Informação inicial:**

$$\theta_0|D_0 \sim (m_0, C_0).$$

Utilizando estas expressões, pode-se mostrar que $(Y_t|\delta_t, \epsilon) \sim \text{BN}(\epsilon, \frac{\epsilon}{\lambda_t + \epsilon})$. Portanto tem-se que

$$\begin{aligned} E(Y_t|\lambda_t, \epsilon) &= \lambda_t \\ \text{Var}(Y_t|\lambda_t, \epsilon) &= \lambda_t + \frac{\lambda_t}{\epsilon}. \end{aligned} \quad (3.25)$$

Note que $\text{Var}(Y_t|\delta_t, \epsilon) > E(Y_t|\delta_t, \epsilon)$, portanto o modelo dinâmico binomial negativo é capaz de capturar a superdispersão através de um termo aditivo positivo na média.

No segundo modelo, acrescenta-se um termo aleatório na função de ligação *log*, ou seja, $\log(\lambda_t) = F'_t \theta_t + v_t$, com $v_t \sim N(\xi, V)$, mantendo-se a equação do sistema (Schmidt e Pereira, 2011). Sendo assim, o modelo dinâmico Poisson-Lognormal é caracterizado pelas seguintes componentes:

- **Equação das observações:**

$$(Y_t|\lambda_t^*) \sim \text{Poisson}(\lambda_t^*)$$

- **Distribuição a priori:**

$$(\lambda_t^*|D_{t-1}) \sim \text{Gamma}(\alpha_t, \beta_t);$$

- **Função de ligação: Logarítmica**

$$\eta_t = \log(\lambda_t^*) = F'_t \theta_t + v_t; \quad v_t \sim (\xi, V_t);$$

- **Equação do sistema:**

$$\theta_t = G'_t \theta_{t-1} + w_t; \quad w_t \sim (0, W_t);$$

- **Informação inicial:**

$$\theta_0|D_0 \sim (m_0, C_0).$$

Note que

$$\begin{aligned}\lambda_t^* &= \exp(\theta_t + v_t) = \exp(\theta_t)\exp(v_t) = \lambda_t\delta_t \\ (\delta_t|V) &\sim \text{LN}(\xi, V),\end{aligned}$$

onde LN representa a distribuição Lognormal. Utilizando as expressões acima, pode-se mostrar que

$$\begin{aligned}E(Y_t|\lambda_t, \xi, V_t) &= \lambda_t \exp\left(\xi + \frac{V}{2}\right) \\ \text{Var}(Y_t|\lambda_t, \xi, V_t) &= \lambda_t \exp\left(\xi + \frac{V}{2}\right) + \lambda_t \exp(2\xi + V)(\exp(V) - 1).\end{aligned}$$

Note que $\text{Var}(Y_t|\lambda_t, \xi, V_t) > E(Y_t|\lambda_t, \xi, V_t)$, portanto o modelo dinâmico Poisson-Lognormal, assim como o modelo dinâmico binomial negativo, é capaz de capturar a superdispersão através de um termo aditivo positivo na média.

Por fim, observe-se que não foi assumida uma forma funcional para a distribuição de w_t e θ_0 , e somente os momentos foram especificados. Usualmente assume-se uma distribuição normal para os termos aleatórios do modelo, ou seja, $w_t \sim N(0, W_t)$ e $(\theta_0|D_0) \sim N(m_0, C_0)$. Maiores informações a respeito do modelo dinâmico Poisson, assim como suas extensões, podem ser consultadas em Schimdt e Pereira (2011).

3.3.1 Processo de estimação

Baseado na descrição do modelo feita na seção anterior, nas ideias descritas na Seção 3.1, e, novamente, assumindo conhecidas as matrizes F_t , G_t , W_t , o procedimento inferencial do modelo de Poisson pode ser descrito através dos seguintes passos:

1. **Reconhecimento dos parâmetros a priori de θ_t e η_t :**

Como visto anteriormente, a distribuição a *posteriori* de θ_t possui vetor de médias m_{t-1} e matriz de covariâncias C_{t-1} , isso implica que $(\theta_t|D_{t-1}) \sim (a_t, R_t)$,

onde $a_t = G_t m_{t-1}$ e $R_t = G_t C_{t-1} G_t' + W_t$. Como $\eta_t = F_t' \theta$, então sua distribuição a priori é especificada pelos seguintes momentos

$$(\eta_t | D_{t-1}) \sim (f_t, q_t),$$

com $f_t = F_t' a_t$ e $q_t = F_t' R_t F_t$.

2. Obtenção dos parâmetros da priori de λ_t :

Utilizando os resultados da família exponencial em sua forma canônica, pode-se demonstrar que

$$\begin{aligned} E(\eta_t | D_{t-1}) &= E(\log(\lambda_t) | D_{t-1}) = \psi(\alpha_t) + \log(\beta_t), \\ V(\eta_t | D_{t-1}) &= V(\log(\lambda_t) | D_{t-1}) = \psi'(\alpha_t), \end{aligned}$$

onde $\psi(z) = \log(\Gamma(z))$ e $\psi'(z)$ são as funções *digamma* e *trigamma*, respectivamente. Utilizando-se uma expansão de Taylor de primeira ordem pode-se mostrar que $\psi \approx \log(z)$ e $\psi'(z) \approx 1/z$ respectivamente. Utilizando tais aproximações têm-se que

$$\begin{aligned} f_t &\approx \log(\alpha_t) - \log(\beta_t) = \log\left(\frac{\alpha_t}{\beta_t}\right), \\ q_t &\approx \frac{1}{\alpha_t}. \end{aligned}$$

Resolvendo para α_t e β_t , têm-se que

$$\begin{aligned} \alpha_t &\approx 1/q_t, \\ \beta_t &\approx \exp[-(f_t + \log(q_t))]. \end{aligned}$$

Se outra função de ligação for escolhida, é possível encontrar α_t e β_t via expansão de Taylor para a função $f(\lambda_t) = g^{-1}(\lambda_t)$.

3. Atualização dos parâmetros de λ_t :

Através do *Teorema de Bayes* pode-se derivar a distribuição a posteriori de λ_t ,

sua distribuição é dada por

$$\begin{aligned}
P(\lambda_t|D_t) &\propto P(Y_t|\lambda_t)P(\lambda_t|D_{t-1}) \\
&\propto \lambda_t^{y_t} e^{-\lambda_t} \lambda_t^{\alpha_t-1} e^{-\beta_t \lambda_t} \\
&\propto \lambda_t^{\alpha_t+y_t-1} e^{-(\beta_t+1)\lambda_t}.
\end{aligned} \tag{3.26}$$

Portanto $(\lambda_t|D_t) \sim \text{Gamma}(\alpha_t + y_t, \beta_t + 1)$, onde $D_t = \{D_{t-1}, Y_t\}$.

4. Atualização dos parâmetros de η_t :

Nessa etapa, obtem-se os parâmetros f_t^* e q_t^* em função da média e variância da distribuição a *posteriori* de λ_t , satisfazendo as seguintes igualdades:

$$\begin{aligned}
E(\eta_t|D_t) &= E(g(\lambda_t)|D_t) = f_t^*, \\
V(\eta_t|D_t) &= V(g(\lambda_t)|D_t) = q_t^*.
\end{aligned}$$

Através da função de ligação, tem-se

$$\lambda_t = g^{-1}(\eta_t) = \exp(\eta_t),$$

usando a expansão de Taylor de primeira ordem ao redor de f_t^* , temos

$$\lambda_t \approx \exp(f_t^*) + (\eta_t - f_t^*)\exp(f_t^*).$$

Daí segue que

$$E(\lambda_t|D_t) = E(\exp(\eta_t)|D_t) \approx \exp(f_t^*) = \tilde{\lambda}_t, \tag{3.27}$$

e

$$V(\lambda_t|D_t) = V(\exp(\eta_t)|D_t) \approx (\exp(f_t^*))^2 q_t^* = \tilde{V}_t. \tag{3.28}$$

Resolvendo as equações (3.27) e (3.28) em f_t^* e q_t^* , obtém-se

$$f_t^* = \log(\tilde{\lambda}_t) \text{ e } q_t^* = \frac{\tilde{V}_t}{(\tilde{\lambda}_t)^2}.$$

5. Atualização dos parâmetros de θ_t :

Para completar o ciclo de atualização, basta calcular os momentos m_t e C_t utilizando as equações (3.11) e (3.12), sendo assim, os momentos são dados por

$$\begin{aligned} m_t &= a_t + R_t F_t (f_t^* - f_t) / q_t, \\ C_t &= R_t - R_t F_t F_t' R_t (1 - q_t^* / q_t) / q_t. \end{aligned}$$

A atualização dos parâmetros do sistema, descritos acima, torna viável o processo de previsão via distribuição preditiva. A distribuição preditiva um passo a frente é dada por

$$\begin{aligned} P(y_t | D_{t-1}) &= \int_0^\infty P(y_t | \lambda_t, D_{t-1}) P(\lambda_t | D_{t-1}) d\lambda_t \\ &= \int_0^\infty \frac{\lambda_t^{y_t} e^{-\lambda_t}}{y_t!} \frac{\beta_t^{\alpha_t}}{\Gamma(\alpha_t)} \lambda_t^{\alpha_t} e^{-\beta_t \lambda_t} d\lambda_t \\ &= \frac{\beta_t^{\alpha_t}}{\Gamma(\alpha_t) y_t!} \int_0^\infty \lambda_t^{y_t + \alpha_t - 1} e^{-(\beta_t + 1) \lambda_t} d\lambda_t \\ &= \frac{\beta_t^{\alpha_t}}{\Gamma(\alpha_t) y_t!} \frac{\Gamma(y_t + \alpha_t)}{(\beta_t + 1)^{y_t + \alpha_t}}. \end{aligned} \tag{3.29}$$

Pode-se reescrever a distribuição (3.29) da seguinte forma

$$P(y_t | D_{t-1}) = \binom{\alpha_t + y_t - 1}{y_t} \left(\frac{\beta_t}{1 + \beta_t} \right)^{\alpha_t} \left(1 - \frac{1}{1 + \beta_t} \right)^{y_t}, \tag{3.30}$$

ou seja, a distribuição preditiva um passo a frente é uma Binomial negativa com $r = \alpha_t$ e $p = \beta_t / (1 + \beta_t)$.

Note que tal integral pôde ser calculada de forma analítica, mas isso nem sempre é possível. Nesses casos, deve-se usar métodos de integração numérica, aproximações ou usar funções especiais para representar essas integrais. Utilizando as propriedades de esperança e variância condicional, pode-se calcular, de maneira direta, a média e variância da distribuição preditiva um passo a frente:

$$E(Y_t|D_{t-1}) = E(E(Y_t|\lambda_t, D_{t-1})) = \frac{\alpha_t}{\beta_t},$$

$$V(Y_t|D_{t-1}) = V(E(Y_t|\lambda_t, D_{t-1})) + E(V(Y_t|\lambda_t, D_{t-1})) = \frac{\alpha_t(1 + \beta_t)}{\beta_t^2}.$$

Tais momentos são úteis para comparações do desempenho preditivo de modelos candidatos. Além disso, eles permitem, também, o monitoramento da adequabilidade do modelo ao longo do tempo (da-Silva et al., 2014).

Para ilustrar o que foi exposto acima, simulamos 100 observações a partir do modelo dinâmico Poisson utilizando $m_0 = 2$, $C_0 = 2$ e $W = 0.001$. Para a etapa de filtragem foi escolhido uma priori vaga. A série simulada e a filtragem estão ilustrados na Figura 3.2.

No próximo capítulo, descreve-se com detalhes a teoria dos filtros de partículas bem como os filtros *Bootstrap* proposto por Gordon et al. (1993), e o Filtro de partículas auxiliar, proposto por Pitt e Shephard (1999).

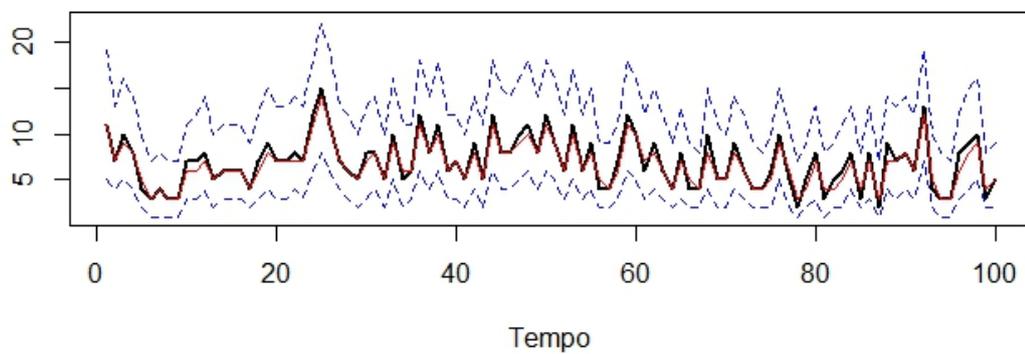
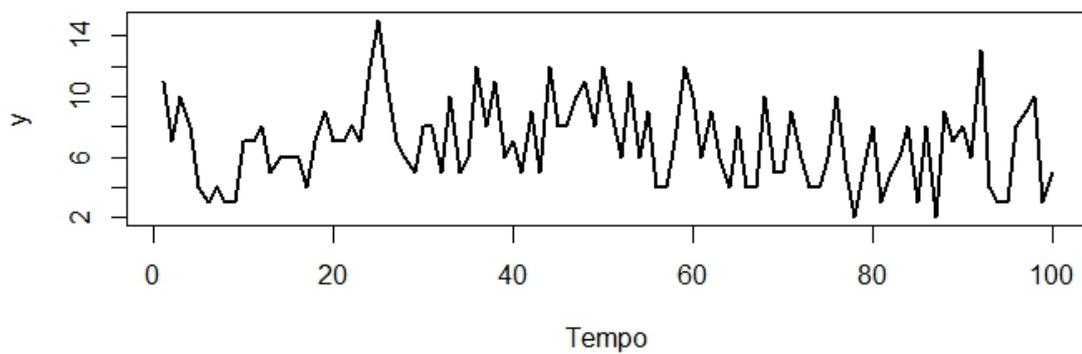


Figura 3.2: De cima para baixo temos: Série histórica y_t . Série histórica y_t em preto, valor predito em vermelho, intervalo de credibilidade em linhas azuis.

Capítulo 4

Filtros de Partículas Básicos

Neste capítulo descreveremos alguns métodos alternativos de estimação *online* para a estimação do vetor de estados e alguns parâmetros estáticos nos modelos dinâmicos. É importante notar que, a suposição de normalidade, e/ou linearidade para a equação das observações e para a equação do sistema é um tanto restritiva, e muitas vezes inadequada.

Para modelos mais complexos, em que a equação das observações (e sistema) não é normal, ou a suposição de linearidade não é válida, estimacões ótimas para o modelo de espaço de estados não admitem soluções analíticas, sendo necessário o uso de métodos numéricos e aproximações para a resolução de integrais. Nesse caso, métodos sofisticados e eficientes para a reamostragem de distribuições de probabilidades complexas precisam ser utilizados.

Desde o trabalho seminal de Gordon et. al. (1993), os métodos de estimação via *filtros de partículas* tornaram-se uma classe popular de algoritmos para resolver, numericamente, os problemas de estimação apresentados pelos modelos de espaço de estados, de uma forma geral, e em especial, dos modelos dinâmicos. Os filtros de partículas têm a vantagem de permitir a estimação *online*, isto é, recursivamente, quando uma nova observação se torna disponível. Os filtros de partículas são utilizados com muita frequência nas áreas de econometria, navegação, robótica, dentre outras.

Segundo Lopes e Tsay (2011), existem duas vertentes distintas na literatura de filtros de partícula. A primeira é definida pelo chamado *Filtro Bootstrap* (FB), intro-

duzido por Gordon et. al. (1993), o qual é baseado na amostragem por importância com reamostragem (Amostragem/Reamostragem)¹ introduzida por Smith and Gelfand (1992). A segunda vertente é definida pelo chamado *Filtro de Partícula Auxiliar* (FPA), introduzido por Pitt e Shephard (1999), o qual é baseado na reamostragem com amostragem por importância (Reamostragem/Amostragem). Tais métodos serão detalhados nas próximas seções.

Para familiarizar o leitor com as notações utilizadas ao longo desse capítulo, considere o modelo dinâmico geral, no qual as suposições de normalidade e/ou linearidade são relaxadas. Nesse cenário, as equações das observações e do sistema são escritas da seguinte forma

- **Equação das Observações:**

$$(y_t|\theta_t) \sim p(y_t|\theta_t).$$

- **Equação do Sistema**

$$(\theta_t|\theta_{t-1}) \sim p(\theta_t|\theta_{t-1}), \quad t = 1, 2, \dots$$

Denote a densidade de probabilidade inicial para os estados por $p(\theta_0)$. Neste capítulo, os parâmetros estáticos, como as variâncias/covariâncias observacionais e do sistema, V e W , no modelo normal, são assumidos conhecidos. O caso em que os parâmetros estáticos são assumidos desconhecidos será tratado no próximo capítulo.

As recursões de Kalman são substituídas, respectivamente, por

$$p(\theta_t|D_{t-1}) = \int p(\theta_t|\theta_{t-1})p(\theta_{t-1}|D_{t-1})d\theta_{t-1}, \quad (4.1)$$

$$p(\theta_t|D_t) = \frac{p(y_t|\theta_t)p(\theta_t|D_{t-1})}{p(y_t|D_{t-1})}, \quad (4.2)$$

$$p(y_t|D_{t-1}) = \int p(y_t|\theta_t)p(\theta_t|D_{t-1})d\theta_t. \quad (4.3)$$

Na prática, a integração com respeito a θ_t em (4.1) e a implementação do teorema de Bayes em (4.2) são ambas intratáveis e/ou computacionalmente custosas.

¹Em inglês SIR - *Sampling Importance Resampling*

De acordo com Lopes e Tsay (2011), os filtros de partícula combinam a natureza da estimação sequencial dos filtros de Kalman com a flexibilidade de modelagem dos métodos MCMC para aproximar e amostrar as distribuições (4.1) a (4.3).

De fato, o processo de filtragem pode ser empreendido por repetidas aplicações de um procedimento em dois estágios (Pitt e Shephard, 1999). No primeiro estágio, a densidade *a posteriori* atual deve ser *propagada* no futuro via densidade de transição, definida pela densidade (4.1). No segundo estágio, o sistema deve ser atualizado, via teorema de Bayes, conforme (4.3). Tal densidade é denominada *densidade filtrada*. Em resumo, o processo de filtragem é obtido de acordo com o esquema:

Processo de Filtragem:

O processo de filtragem é feito em dois estágios:

Estágio I: A densidade atual $p(\theta_t|D_t)$ precisa ser *propagada* no futuro (predição via densidade de transição $p(\theta_{t+1}|\theta_t)$) a fim de produzir-se a *densidade preditiva* (*priori* no tempo $t + 1$):

$$\begin{aligned} p(\theta_{t+1}|D_t) &= \int p(\theta_{t+1}, \theta_t|D_t)d\theta_t \\ &= \int p(\theta_{t+1}|\theta_t)p(\theta_t|D_t)d\theta_t. \end{aligned} \quad (4.4)$$

Estágio II: Obter a *densidade filtrada*, $p(\theta_{t+1}|D_{t+1})$, via teorema de Bayes, isto é,

$$p(\theta_{t+1}|D_{t+1}) = \frac{p(y_{t+1}|\theta_{t+1})p(\theta_{t+1}|D_t)}{p(y_{t+1}|D_t)}, \quad (4.5)$$

em que

$$p(y_{t+1}|D_t) = \int p(y_{t+1}|\theta_{t+1})p(\theta_{t+1}|D_t)d\theta_{t+1}, \quad (4.6)$$

é a *densidade preditiva a um passo a frente*. As equações (4.5) e (4.6) indicam que os dados podem ser processados sequencialmente, à medida que novas observações são disponibilizadas, de modo a atualizar o conhecimento acerca dos estados latentes.

O grande entrave na implementação dos estágios I e II é lidar com o cálculo

das integrais envolvidas, uma vez que o suporte (espaço paramétrico dos estados) é contínuo. Caso o suporte fosse de um conjunto finito de pontos conhecidos, tal problema seria trivial. Inúmeras tentativas têm sido propostas na literatura, de modo a aproximar as densidades filtradas, entre estes: Gerlach, Carter e Kohn (1996), West (1992), West e Harrison (1997, caps. 13 e 15), Gordon et al. (1993), Pitt e Shephard (1999), entre outras.

De maneira formal, os filtros de partículas constituem a classe de filtros, via simulação, que aproximam recursivamente a variável aleatória filtrada $(\theta_t|D_t)$, através de partículas $\theta_t^1, \dots, \theta_t^M$, às quais estão associadas uma distribuição de probabilidade discreta com probabilidades respectivas π_t^1, \dots, π_t^M .

Dessa forma, uma variável aleatória contínua pode ser aproximada por uma variável aleatória discreta com suporte aleatório. Estes pontos discretos (partículas) são entendidas como sendo amostras selecionadas a partir de $p(\theta_t|D_t)$, isto é, $\{\theta_t^i, i = 1, \dots, M\} \approx p(\theta_t|D_t)$.

Na literatura (até 1999) as probabilidades π_t^j eram assumidas serem todas iguais a $1/M$, com M muito grande, de tal forma que as partículas tivessem densidade $p(\theta_t|D_t)$ quando $M \rightarrow \infty$.

Nos filtros de partículas trata-se o suporte discreto, gerado pelas partículas, como a *verdadeira densidade filtrada* $p(\theta_t|D_t)$. Isto permite produzir uma aproximação para a densidade preditiva, $p(\theta_{t+1}|D_t)$, simplesmente ao usar um suporte discreto para as partículas. Denote por $\pi_t^j = p(\theta_t = \theta_t^j|D_t)$, e, de acordo com (4.5), tem-se

$$\hat{p}(\theta_{t+1}|D_t) = \sum_{j=1}^M p(\theta_{t+1}|\theta_t^j)\pi_t^j, \quad (4.7)$$

que representa a *densidade preditiva empírica* (uma mistura de distribuições), que é uma aproximação de (4.5).

Posteriormente, a densidade preditiva empírica é combinada com a densidade associada à equação das observações, via teorema de Bayes, para produzir, a menos de uma constante de proporcionalidade, a *densidade filtrada empírica*,

$$\hat{p}(\theta_{t+1}|D_{t+1}) \propto p(y_{t+1}|\theta_{t+1}) \sum_{j=1}^M p(\theta_{t+1}|\theta_t^j)\pi_t^j, \quad (4.8)$$

que é uma aproximação à verdadeira densidade filtrada dada em (4.5).

Genericamente, utilizando-se os filtros amostra-se a partir da densidade filtrada atualizada em (4.8), para produzir novas partículas $\theta_{t+1}^1, \dots, \theta_{t+1}^M$ com pesos $\pi_{t+1}^1, \dots, \pi_{t+1}^M$, e, então, uma aproximação para $\hat{p}(\theta_{t+2}|D_{t+2})$. Tal procedimento é, então, iterado através do tempo.

No caso do filtro de partículas funcionar adequadamente, então é possível estimar, entre outras distribuições, a *densidade preditiva a um passo a frente*, $p(y_{t+1}|D_t)$, de modo a possibilitar o cálculo da verossimilhança conjunta

$$L = \prod_{t=0}^{T-1} p(y_{t+1}|D_t),$$

que é útil na elaboração de medidas de diagnóstico.

Uma forma de amostrar a partir da densidade filtrada empírica, (4.8), é via teorema de Bayes, em que a *priori* (densidade preditiva) $\hat{p}(\theta_{t+1}|D_t) = \sum_{j=0}^M p(\theta_t|\theta_t^j)\pi_t^j$ é combinada com a verossimilhança, $\hat{p}(y_{t+1}|\theta_{t+1})$, para produzir a *posteriori*. Dessa forma, amostra-se a partir de $\hat{p}(\theta_{t+1}|D_t)$ ao estabelecer uma distribuição discreta para θ_t , tal que $p(\theta_t^j) = \pi_t^j$, $j = 1, \dots, M$ e, então, seleciona-se θ_{t+1}^j a partir de $p(\theta_{t+1}|\theta_t^j)$, que é reponderado com pesos $p(y_{t+1}|\theta_{t+1})$, de modo a produzir partículas com densidade $\hat{p}(\theta_{t+1}|D_{t+1})$.

Um método útil para amostrar a partir de $p(\theta_{t+1}|D_{t+1})$ é o método SIR (Rubin, 1988), que é detalhado nos apêndices.

Em resumo, o objetivo dos filtros de partículas é selecionar um conjunto de M partículas independentes e identicamente distribuídas (*i.i.d*) $\{\theta_t^i\}_{i=1}^M$ que aproximam $p(\theta_t|D_t)$, começando com a geração de um conjunto de M partículas *i.i.d* $\{\theta_{t-1}^i\}_{i=1}^M$, que aproximam $p(\theta_{t-1}|D_{t-1})$ (Lopes e Tsay, 2011).

Os filtros de partículas *Bootstrap* (FB) e de partículas auxiliar (FPA) serão detalhados nas próximas seções. Serão ilustrados, também, alguns exemplos utilizando o *software R*.

4.1 *Filtro Bootstrap*

O *Filtro Bootstrap* (FB) foi proposto por Gordon et. al. (1993), e é baseado na aplicação sequencial do método SIR² de Rubin (1988).

Suponha que tem-se um conjunto de amostras aleatórias $\{\theta_{t-1}(i) : i = 1, \dots, N\}$ retiradas de uma população com função densidade de probabilidade $p(\theta_{t-1}|D_{t-1})$. O FB é um algoritmo que *propaga* e depois *atualiza* essas partículas para obter um conjunto de valores $\{\theta_t(i) : i = 1, \dots, N\}$, os quais são aproximadamente distribuídos de acordo com $p(\theta_t|D_t)$.

O FB ocorre nos dois estágios a seguir, que são decorrentes da expressão

$$p(\theta_t, \theta_{t-1}|y_t, D_{t-1}) \propto \underbrace{p(y_t|\theta_t)}_{2. \text{ Reamostragem}} \underbrace{p(\theta_t|\theta_{t-1})p(\theta_{t-1}|D_{t-1})}_{1. \text{ Propagação}},$$

referente à distribuição a *posteriori* conjunta de θ_t e θ_{t-1} , escrita em termos proporcionais.

1. **Propagação:** Cada partícula $\theta_{t-1}^{(i)}$ é propagada para o futuro, através da equação do sistema, para obter-se partículas da *priori* no tempo t (densidade preditiva). Ao amostrar $\theta_t^*(i)$ a partir de $p(\theta_t|\theta_{t-1})$. Dessa forma, com $\theta_t^*(i) \sim p(\theta_t|\theta_{t-1} = \theta_{t-1}(i))$, obtém-se, na verdade, uma observação gerada a partir de $p(\theta_t|D_{t-1})$.
2. **Atualização:** Dada a observação obtida no tempo t , y_t , calcula-se a verossimilhança de cada partícula, avaliada em $\theta_t^*(i)$, que foi gerada a partir da *priori* $p(\theta_t|D_{t-1})$, e obtém-se o peso normalizado, w_i , para cada partícula, isto é,

$$w_i = \frac{p(y_t|\theta_t^*(i))}{\sum_{j=1}^N p(y_t|\theta_t^*(j))}.$$

Então defina uma distribuição discreta $\{\theta_t^*(i) : i = 1, \dots, N\}$, com massa de probabilidade w_i associada ao elemento i . Agora, reamostre cada partícula com probabilidade igual a w_i , para gerar amostras $\{\theta_t(i) : i = 1, \dots, N\}$, tal que para qualquer j , $p(\theta_t(j) = \theta_t^*(i)) = w_i$. Estas novas amostras seguem, aproximadamente, a distribuição $p(\theta_t|D_t)$.

²Sampling Importance Resampling

As etapas de propagação e atualização, descritas acima, formam uma simples iteração do algoritmo recursivo. Para iniciar o algoritmo, N amostras $\theta_t^*(i)$ são retiradas de uma *priori* conhecida $p(\theta_0|D_0)$. Essas partículas vão diretamente para o estágio de atualização do filtro. Espera-se que a amostra atualizada seja aproximadamente distribuída de acordo com $p(\theta_t|D_t)$ (Gordon et al. 1993).

Em suma, com o algoritmo FB primeiro propaga-se as partículas obtidas da *posteriori* no tempo $t-1$, afim de gerar-se partículas da *priori* no tempo t . Posteriormente, essas mesmas partículas propagadas são reamostradas com peso proporcional à suas verossimilhanças.

As justificativas de cada etapa do filtro foram dadas por Gordon et. al. (1993):

- **Propagação:** A fase de propagação (ou realocação) do algoritmo é intuitiva. Se $\theta_{t-1}(i)$ é uma amostra obtida de $p(\theta_{t-1}|D_{t-1})$ então, pela equação do sistema, $\theta_t^*(i) \sim p(\theta_t(i)|\theta_{t-1}(i))$. Logo as partículas $\theta_t^*(i)$ são distribuídas independentemente como $p(\theta_t|D_{t-1})$.
- **Atualização:** A justificativa para a etapa de atualização se baseia no resultado dado por Smith e Gelfand (1992). Eles mostraram que o teorema de Bayes pode ser implementado como um *bootstrap* ponderado. Suponha que as amostras $\{\theta_t^*(i) : i = 1, \dots, N\}$ são calculadas a partir de uma função densidade contínua $G(x)$, e que essas amostras são obtidas de uma f.d.p proporcional a $L(x)G(x)$, onde $L(x)$ é uma função conhecida. O teorema de Bayes diz que uma amostra retirada de uma distribuição discreta sob $\{\theta_t^*(i) : i = 1, \dots, N\}$, com massa de probabilidade $L(\theta_t^*(i)) / \sum L(\theta_t^*(j))$ em $\theta_t^*(i)$, converge, em distribuição, para a densidade desejada, quando N tende para o infinito. Se fazemos $G(x)$ como $p(\theta_t|D_{t-1})$ (*priori*) e $L(x)$ como $p(y_t|\theta_t)$ (a verossimilhança), então o teorema de Bayes justifica o procedimento de atualização das partículas.

O algoritmo FB, pode ser descrito através das seguintes etapas (Lopes e Tsay, 2011), e é ilustrado na Figura 4.1:

Filtro *Bootstrap*

1. Propague $\{\theta_{t-1}^i\}_{i=1}^N$ para $\{\tilde{\theta}_t^i\}_{i=1}^N$ via $p(\theta_t|\theta_{t-1})$.
2. Reamostragem $\{\theta_t^i\}_{i=1}^N$ a partir de $\{\tilde{\theta}_t^i\}_{i=1}^N$ com pesos proporcionais à verossimilhança, isto é,

$$w_t^i \propto p(y_t|\tilde{\theta}_t^i).$$

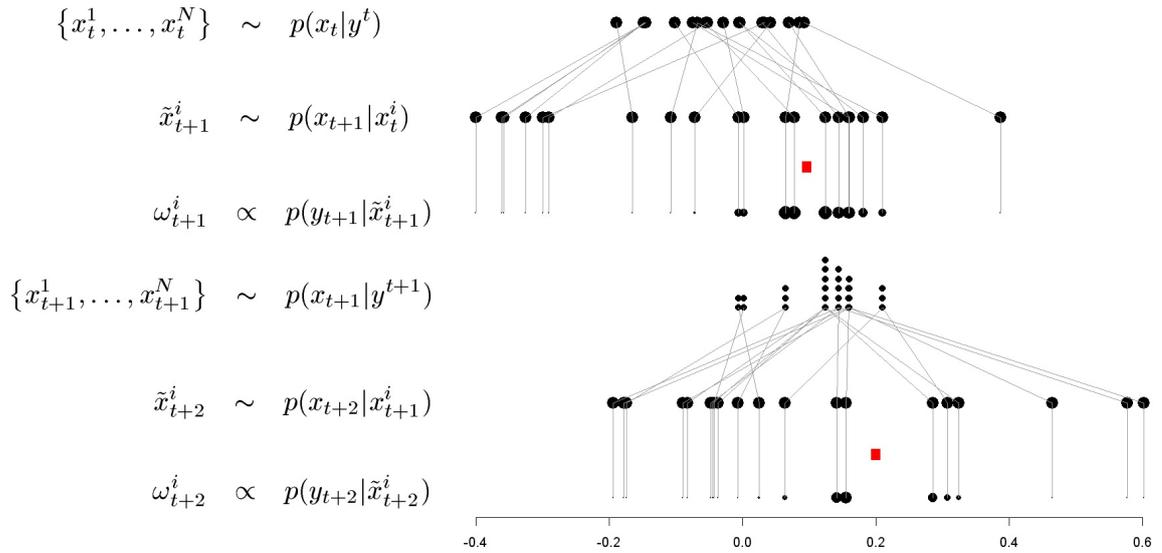


Figura 4.1: Representação esquemática do *filtro bootstrap* sobre dois períodos de tempo. Os quadrados são y_{t+1} e y_{t+2} . De cima para baixo, primeiro, segundo, quarto e quinto conjunto de pontos representando as partículas, enquanto o terceiro e sexto conjunto de pontos representa os pesos das partículas (Lopes e Tsay, 2011).

4.1.1 Limitações dos filtros de partículas baseados no SIR

Pitt e Shephard (1999) citaram duas limitações básicas relacionadas aos filtros de partículas baseados em amostragem por importância (SIR). A primeira limitação ocorre quando existe um *outlier*. Na presença de valores extremos, a distribuição dos pesos utilizada no processo de amostragem de importância contém componentes com pesos muito desiguais, sendo necessário o uso de um número, N , extremamente grande de partículas para que estas estejam próximas das amostras geradas a partir da densidade de filtragem empírica.

Logo, a primeira questão que surge é como amostrar eficientemente a partir de (4.2). Esse problema foi tratado no artigo de Pitt e Shephard (1999) por meio de variáveis auxiliares, que serão detalhadas na seção 4.2.

A segunda limitação ocorre em filtros de partículas para os quais os pesos π^j são iguais. Quando $N \rightarrow \infty$, então os pesos amostrais podem ser usados para aproximar, de maneira satisfatória, o centro da densidade (4.1). Entretanto, as caudas de (4.3) não são aproximadas de maneira satisfatória. Portanto a segunda questão é como melhorar o comportamento da densidade preditiva empírica nas caudas. Tal problema foi analisado por Pitt e Shephard (1998).

4.2 *Filtro de Partículas Auxiliar*

O *Filtro de Partículas Auxiliar* (FPA) foi proposto por Pitt e Shephard (1999) e, diferentemente do FB, este filtro reamostra partículas, a partir da distribuição a *posteriori* no tempo $t - 1$, com peso incorporando o valor da observação no tempo t , y_t . Posteriormente, propaga-se a partícula reamostrada.

Sendo assim, os filtros de partículas em geral amostram a partir da densidade (4.2) para produzir novas partículas $\{\theta_t^j\}_{j=1}^N$ com pesos $\{\pi_t^j\}_{j=1}^N$. Tal procedimento é iterado um número grande de vezes através do tempo.

Entretanto, os autores argumentam que essa estrutura de mistura da equação (4.1) traz uma dificuldade computacional em se implementar o SIR ou MCMC. Pitt e Shephard (1999) advogam que muito desses problemas são reduzidos quando realiza-se a filtragem em uma dimensão superior, isto é, levando em conta a distribuição conjunta de variáveis que influenciam o processo.

Os autores propõem amostrar a partir da densidade conjunta $p(\theta_t, k|D_t)$, onde k é um índice da mistura em (4.1). Pitt e Shephard (1999), definiram a seguinte densidade,

$$p(\theta_t, k|D_t) \propto p(y_t|\theta_t)p(\theta_t|\theta_{t-1}^k)\pi^k, \quad k = 1, \dots, N. \quad (4.9)$$

Ao retirar-se amostras da densidade conjunta (4.9) e descartar-se o índice, então produz-se uma amostra a partir da densidade (4.2).

No método SIR baseado no APF, a densidade (4.9) é aproximada por

$$g(\theta_t, k|D_t) \propto p(y_t|\mu_t^k)p(\theta_t|\theta_{t-1}^k)\pi^k, \quad k = 1, \dots, N,$$

onde μ_t^k é a média, mediana, moda ou algum outro valor associado com a densidade de $(\theta_t|\theta_{t-1}^k)$. A densidade a *posteriori* do índice k é dada por

$$g(k|D_t) \propto \pi^k \int p(y_t|\mu_t^k)p(\theta_t|\theta_{t-1}^k)d\theta_t = \pi^k p(y_t|\mu_t^k). \quad (4.10)$$

Portanto, pode-se amostrar pares $(\theta_t, k|D_t)$, a partir de $g(\theta_t, k|D_t)$, ao simular o índice k com probabilidade $\lambda_k \propto g(k|D_t)$, e, então, amostrar θ_t a partir da densidade do sistema dada pela mistura $p(\theta_t|\theta_{t-1}^k)$.

Pitt e Shephard (1999) denominam λ_k por *peso do primeiro estágio*. A vantagem do procedimento exposto, segundo os autores, é que simulamos a partir de partículas associadas com verossimilhanças preditivas altas.

Após amostrar da densidade conjunta $g(\theta_t, k|D_t)$ N vezes, é realizado uma reponderação, atribuindo à (θ_t^j, k^j) , retirada no primeiro estágio, os respectivos pesos

$$w_j = \frac{p(y_t|\theta_t^j)}{p(y_t|\mu_t^{k^j})} \text{ e } \pi_j = \frac{w_j}{\sum_{i=1}^N w_i}, \quad j = 1, \dots, R.$$

Espera-se que esses pesos sejam menos variáveis que os pesos do método SIR original. O algoritmo pode ser descrito da seguinte forma (Lopes e Tsay, 2011):

Filtro de Partículas Auxiliar:

1. Reamostre $\{\tilde{\theta}_{t-1}^i\}_{i=1}^N$ a partir de $\{\theta_{t-1}^i\}_{i=1}^N$ com pesos proporcionais a

$$w_t^i \propto p(y_t|g(\theta_{t-1}^i)).$$

2. Propague $\{\tilde{\theta}_{t-1}^i\}_{i=1}^N$ para $\{\tilde{\theta}_t^i\}_{i=1}^N$ via $p(\theta_t|\tilde{\theta}_{t-1})$.

3. Reamostre $\{\theta_t^i\}_{i=1}^N$ a partir de $\{\tilde{\theta}_t^i\}_{i=1}^N$ com pesos proporcionais a

$$w_t^i \propto \frac{p(y_t|\tilde{\theta}_t^i)}{p(y_t|g(\tilde{\theta}_t^i))}.$$

4.3 Exemplos

Afim de ilustrar uma aplicação dos filtros acima, foram considerados dois exemplos.

4.3.1 Modelo Dinâmico Linear Gaussiano

O primeiro exemplo está relacionado ao modelo dinâmico linear gaussiano (Vide Capítulo 2). Foram simulados 3 modelos de tamanho 50 com $m_0 = 0$ e $C_0 = 100$, $\tau^2 = 0.5$ variando-se σ^2 para os valores 0.25, 0.5, 1. As séries geradas estão ilustradas na Figura 4.2.

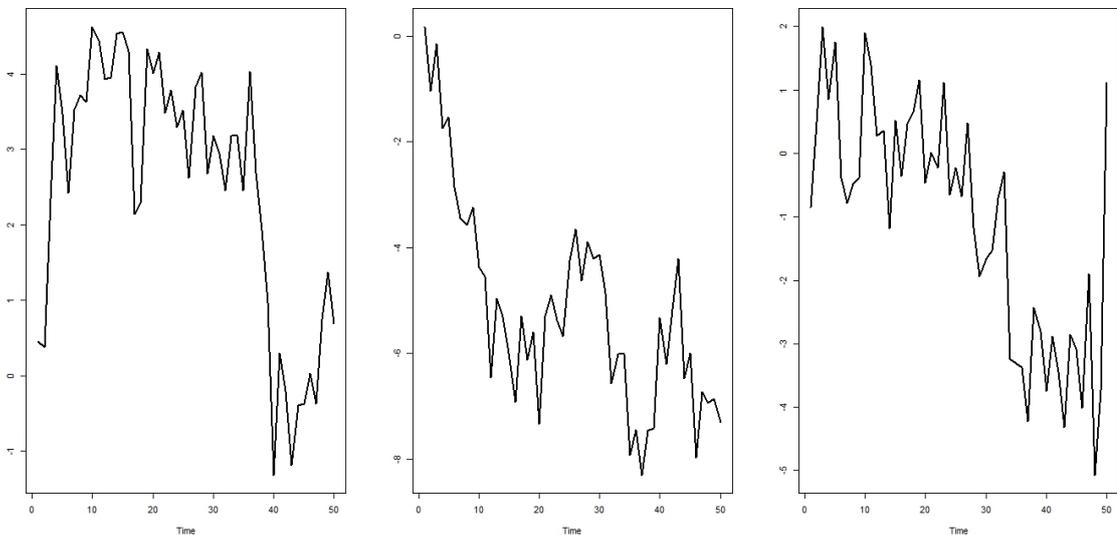


Figura 4.2: Da esquerda para a direita. A primeira série foi gerada usando $W = 0.001$. A segunda série foi gerada usando $W = 0.01$. E, por fim, a terceira série foi gerada usando $W = 0.1$.

Os resultados do algoritmo FB estão ilustrados na Figura 4.3, já os resultados do filtro APF estão ilustrados na Figura 4.4.

A Figura 4.5 mostra a distribuição do erro quadrático médio entre os valores simulados e os valores estimados pelo algoritmo FB.

A Figura 4.6 mostra a distribuição do erro quadrático médio entre os valores simulados e os valores estimados pelo algoritmo FPA.

Para essas simulações nota-se que a distribuição do erro quadrático médio para ambos os filtros são parecidos. Entretanto o filtro FB apresenta menos variabilidade se comparado com o filtro FPA.

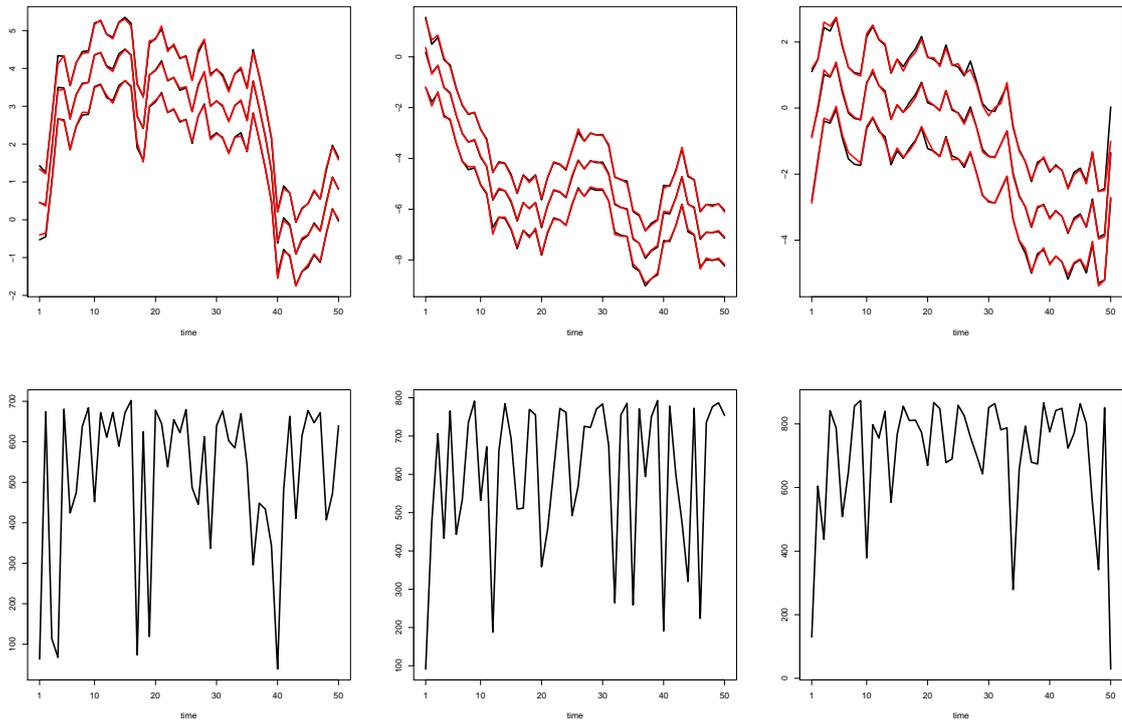


Figura 4.3: Resultados do algoritmo FB. A linha preta representa os dados reais, a linha contínua vermelha representa o valor estimado (Mediana das partículas) e as linhas tracejadas representam o intervalo de credibilidade de 95% (quantis 2,5% e 97,5%). Da esquerda para a direita temos $W = 0.001$, $W = 0.01$ e $W = 0.1$ respectivamente.

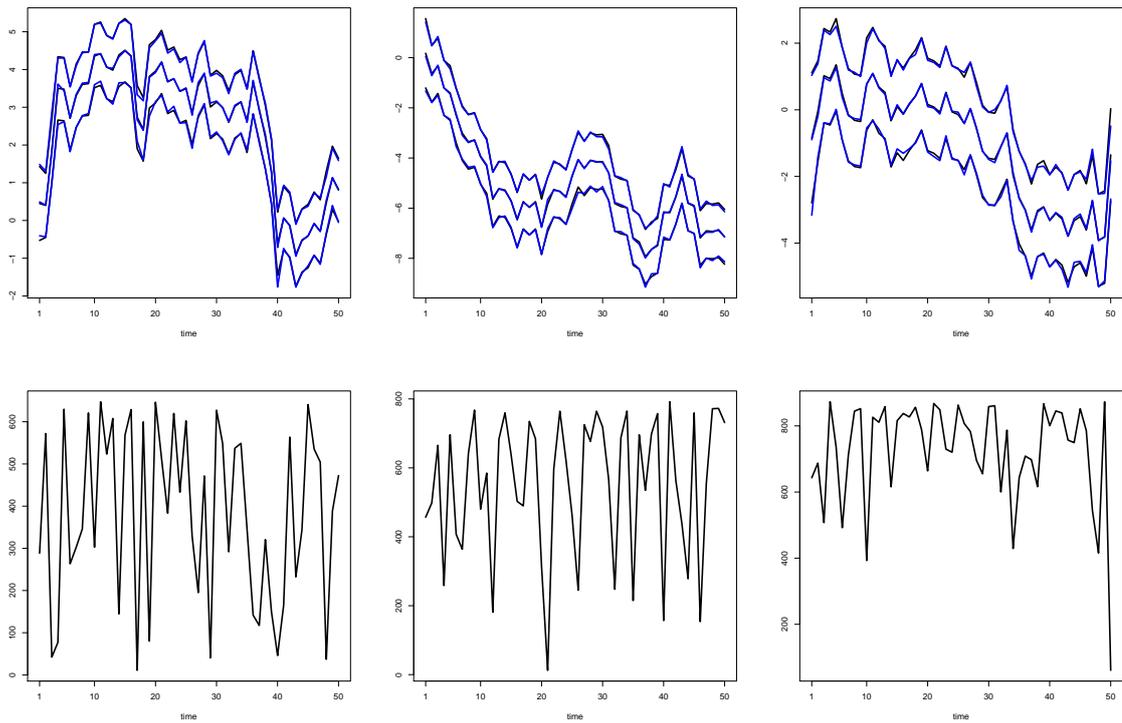


Figura 4.4: Resultados do algoritmo APF. A linha preta representa os dados reais, a linha contínua azul representa o valor estimado (Mediana das partículas) e as linhas tracejadas representam o intervalo de credibilidade de 95% (quantis 2,5% e 97,5%). Da esquerda para a direita temos $W = 0.001$, $W = 0.01$ e $W = 0.1$ respectivamente.

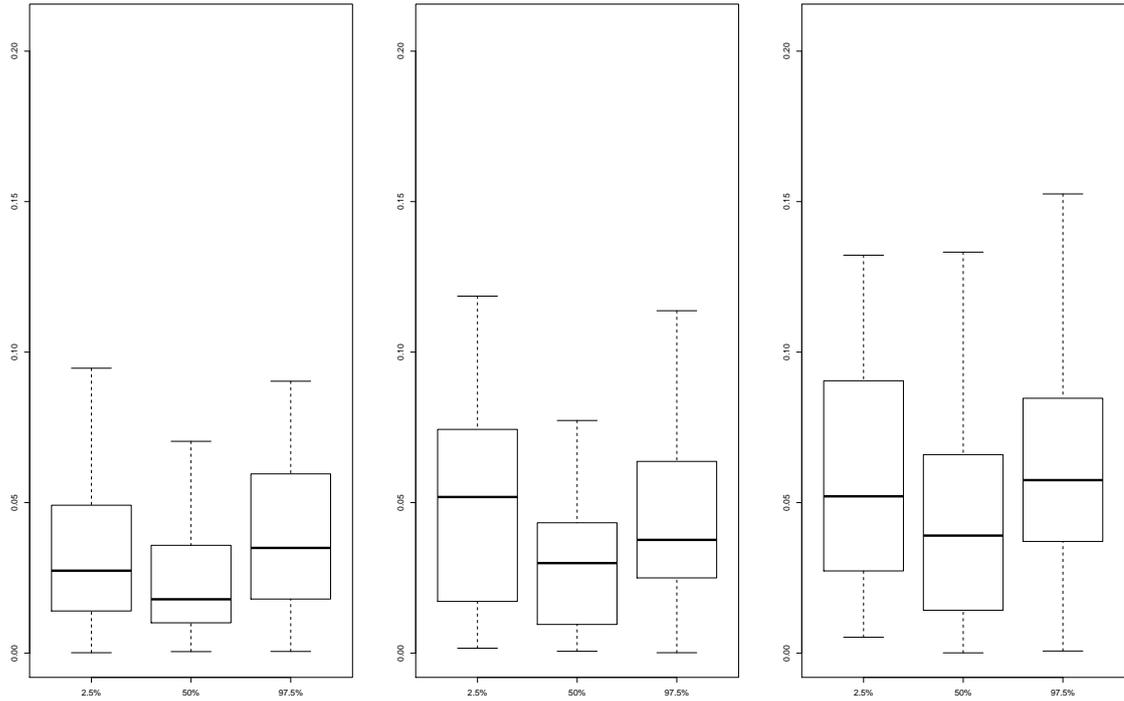


Figura 4.5: Distribuição do erro quadrático médio para o valor da mediana, percentil 2.5% e percentil 97.5% para o algoritmo FB. Da esquerda para a direita temos $W = 0.001$, $W = 0.01$ e $W = 0.1$ respectivamente.

4.3.2 Modelo Dinâmico Poisson

O segundo modelo simulado foi um modelo dinâmico Poisson de primeira ordem, dado por:

- Equação das observações:

$$P(y_t | \lambda_t) = \frac{\lambda_t^{y_t} e^{-\lambda_t}}{y_t!};$$

- Função de ligação: Logarítmica

$$\eta_t = \log(\lambda_t) = \theta_t;$$

- Equação do sistema:

$$\theta_t = \theta_{t-1} + w_t; \quad w_t \sim N(0, W);$$

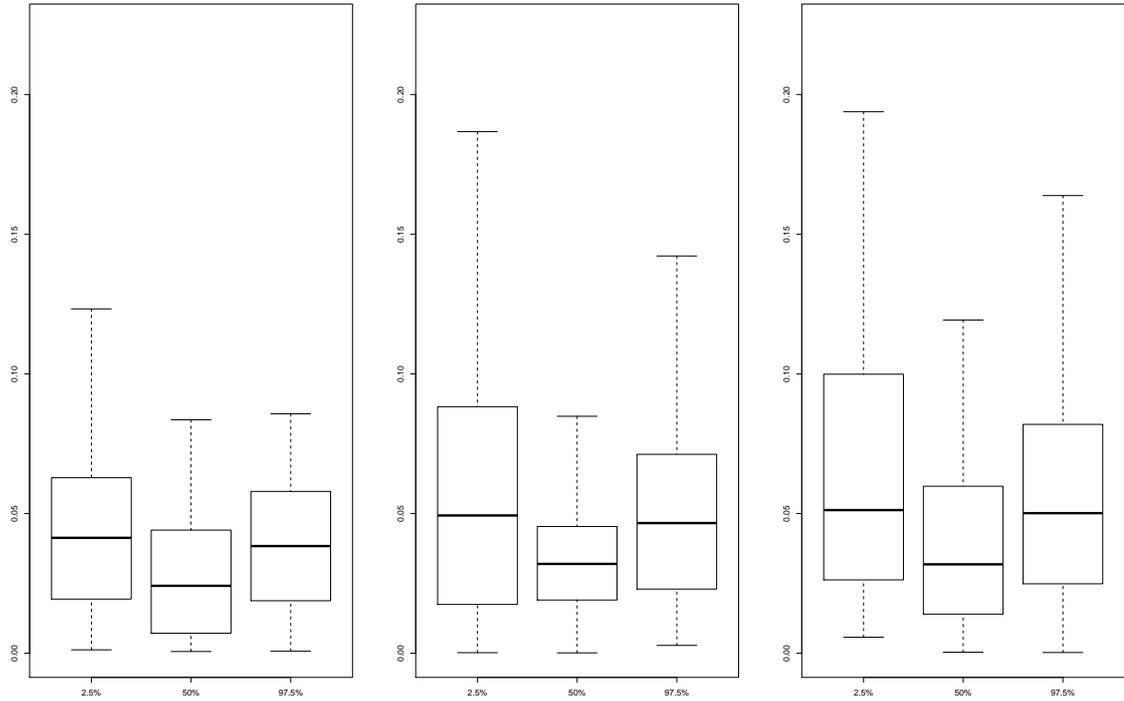


Figura 4.6: Distribuição do erro quadrático médio para o valor da mediana, percentil 2.5% e percentil 97.5% para o algoritmo FPA. Da esquerda para a direita temos $W = 0.001$, $W = 0.01$ e $W = 0.1$ respectivamente.

- **Informação inicial:**

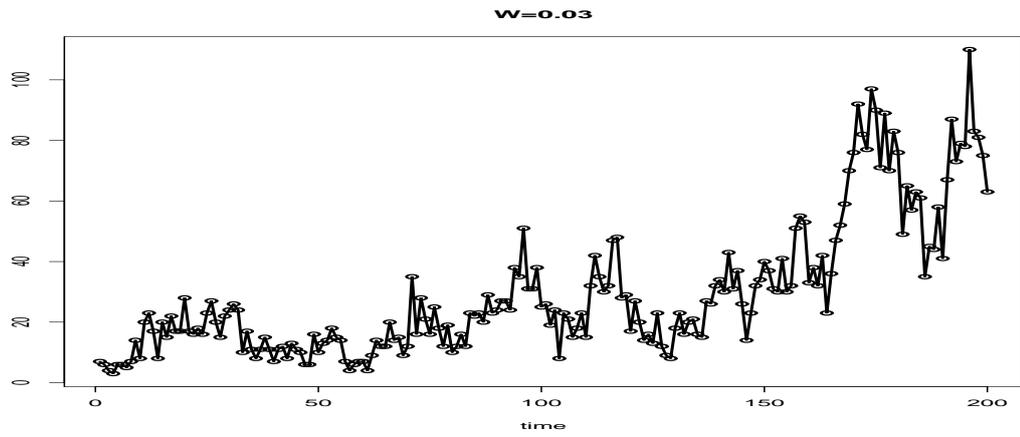
$$\theta_0 | D_0 \sim N(m_0, C_0).$$

Foram simuladas 3 amostras de tamanho 200 com $m_0 = 0$, $C_0 = 100$, e W variando tal que $W = 0.03, 0.05, 0.1$. As séries geradas estão ilustradas na Figura 4.7.

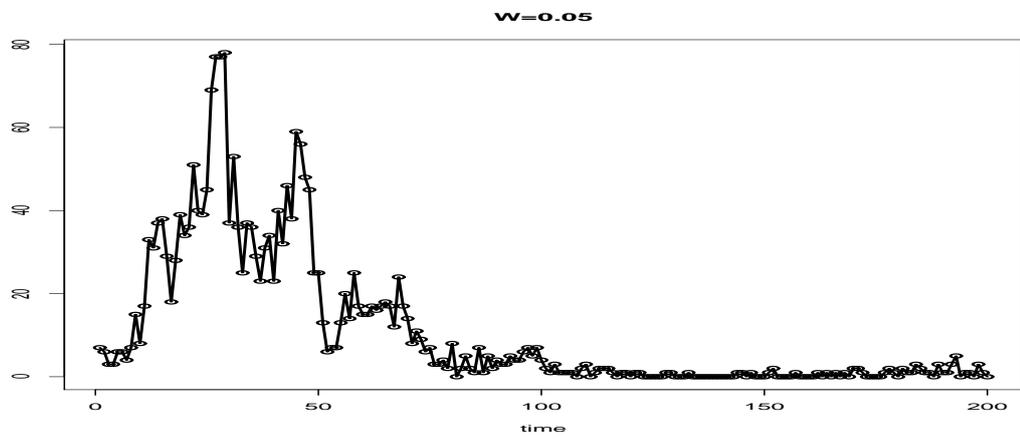
Os resultados do algoritmo FB e FPA estão ilustrados na Figura 4.8. Já os resultados do filtro FPA estão ilustrados na Figura 4.9.

A Figura 4.10 mostra a distribuição do erro quadrático médio entre os valores simulados e os valores estimados pelos algoritmos FB e FPA respectivamente.

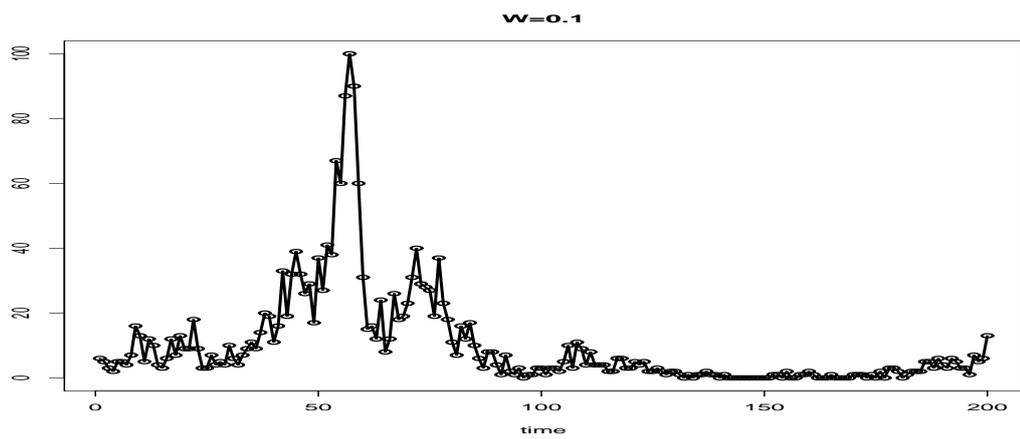
Observando a Figura 4.10, nota-se que, para esse exemplo, o filtro FPA teve uma melhor desempenho quando comparado com o filtro FB, pois possui menores erros médios quadráticos.



(a)

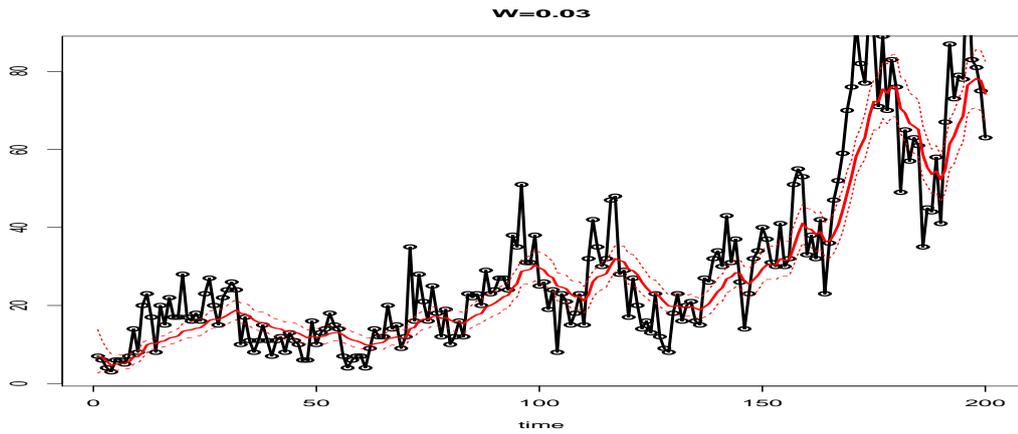


(b)

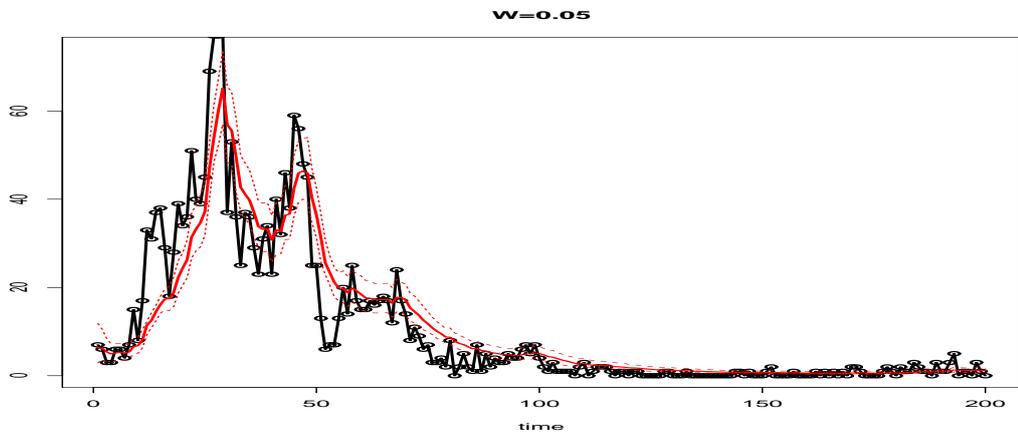


(c)

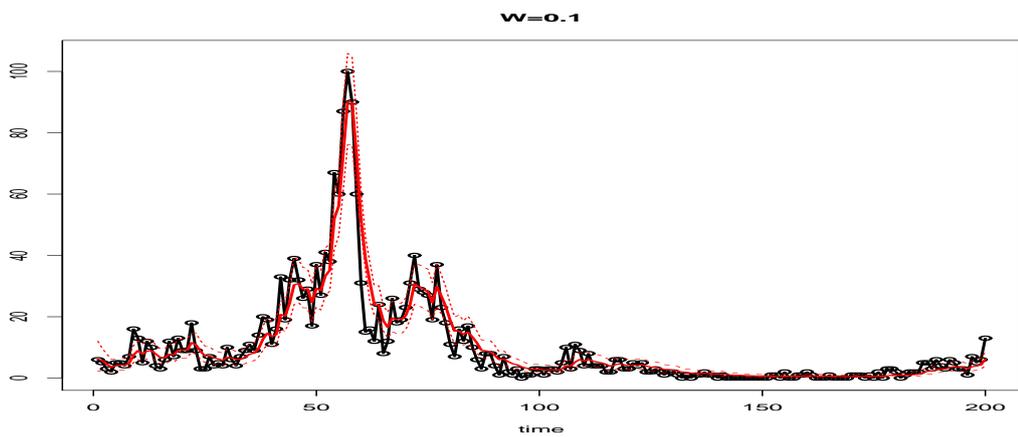
Figura 4.7: Séries simuladas segundo um modelo dinâmico Poisson.



(a)

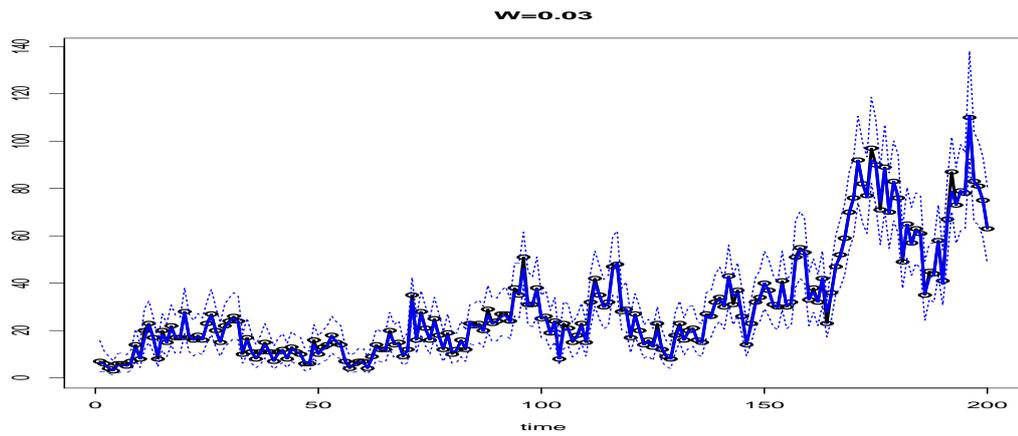


(b)

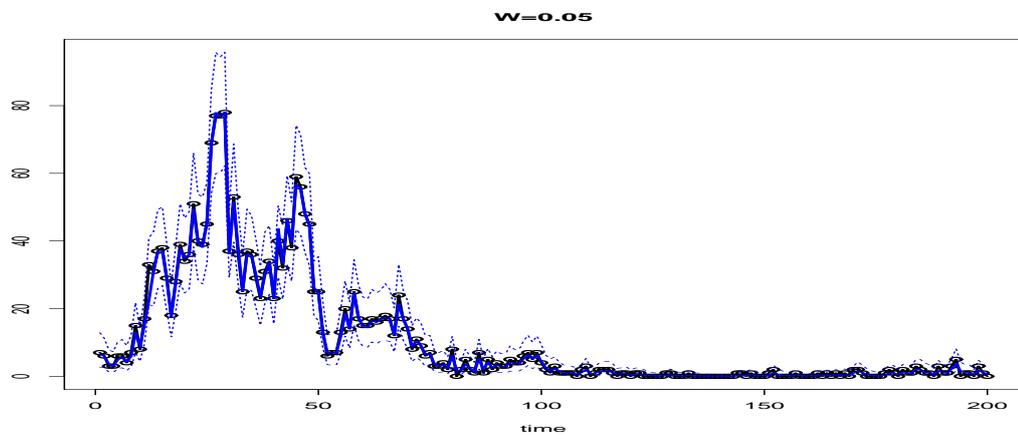


(c)

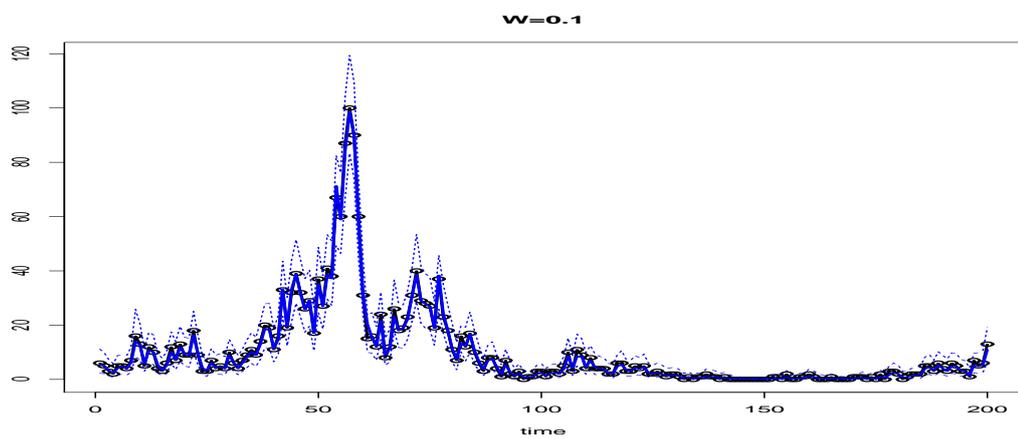
Figura 4.8: Resultados do algoritmo FB. A linha preta representa os dados reais, a linha contínua vermelha representa o valor estimado (Mediana das partículas) e as linhas tracejadas representam o intervalo de credibilidade de 95%.



(a)

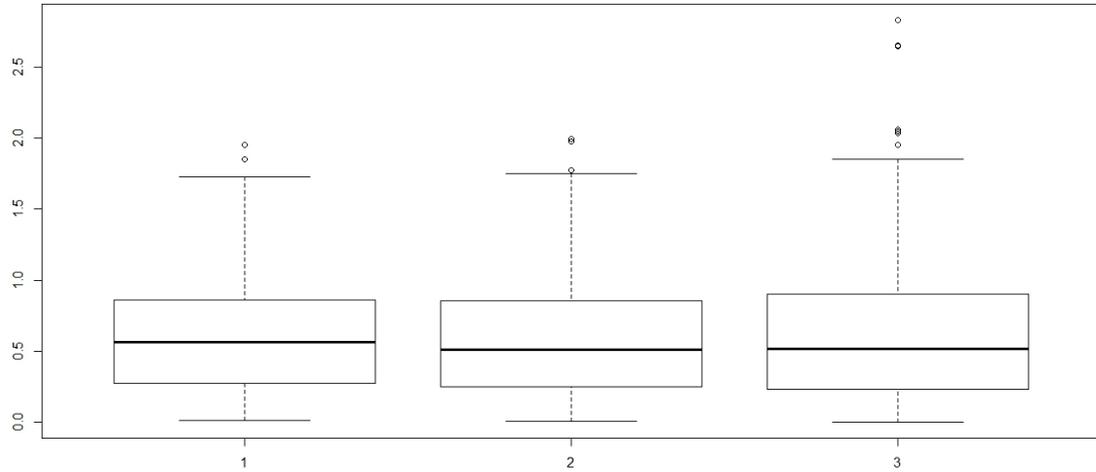


(b)

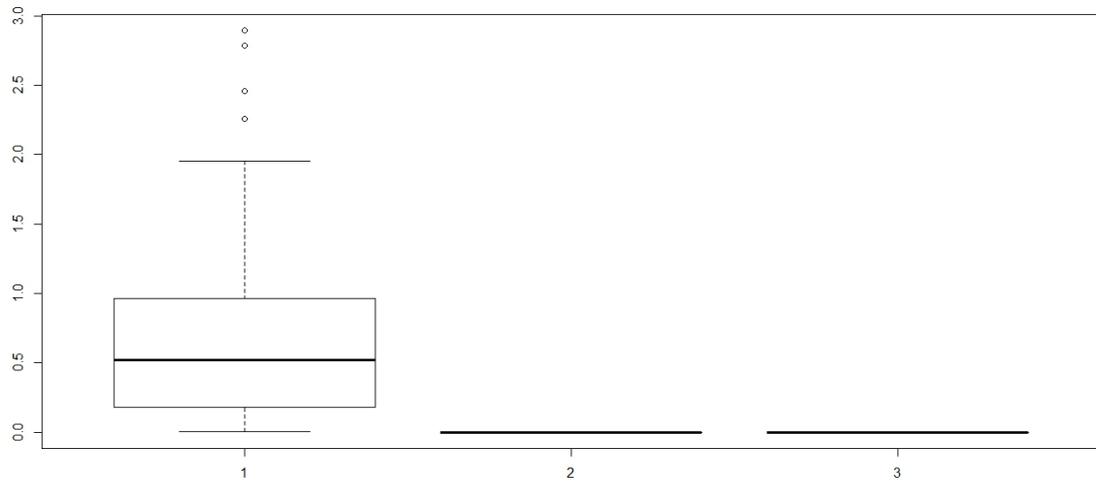


(c)

Figura 4.9: Resultados do algoritmo FPA. A linha preta representa os dados reais, a linha contínua azul representa o valor estimado (Mediana das partículas) e as linhas tracejadas representam o intervalo de credibilidade de 95%.



(a) Filtro *Bootstrap*



(b) Filtro de Partículas Auxiliar

Figura 4.10: Distribuição do erro quadrático médio entre os valores reais e os valores estimados pelos filtros FB e FPA respectivamente. Da esquerda para a direita temos $W = 0.03$, $W = 0.05$ e $W = 0.1$ respectivamente.

Capítulo 5

Filtros de Partículas com parâmetros estáticos - *Parameter Learning*

No Capítulo 4 introduziu-se o conceito de filtro de partículas, onde assumiu-se que os parâmetros estáticos são conhecidos. Porém, na prática, essa suposição é muito restritiva e irreal. Portanto, necessitamos estender a teoria desenvolvida no Capítulo 4 a fim de incorporar aos filtros de partículas a estimação *online* dos parâmetros estáticos do modelo.

Considere o modelo dinâmico geral. Trataremos explicitamente da estimação do vetor de parâmetros estáticos desconhecidos Ψ do modelo de espaço de estados

- **Equação das Observações:**

$$(y_t|\theta_t, \Psi) \sim p(y_t|\theta_t, \Psi);$$

- **Equação do Sistema:**

$$(\theta_t|\theta_{t-1}, \Psi) \sim p(\theta_t|\theta_{t-1}, \Psi),$$

para $t = 1, \dots, T$, com densidade inicial $p(\theta_0|D_0, \Psi)$ e *priori* $p(\Psi)$.

Tipicamente, existem três maneiras de tratar o problema de aprendizado de Ψ : *batch sampling*¹, amostragens *online* e um híbrido entre as duas técnicas.

Aqui trataremos apenas dos métodos *online* para a estimação do vetor Ψ . O leitor interessado pode consultar, por exemplo, Chopin (2002) e Del Moral et. al. (2006), para maiores informações e detalhes teóricos a respeito dos métodos *offline*, do tipo MCMC, para a estimação do vetor Ψ . Outras referências importantes podem ser encontradas no trabalho de Lopes e Tsay (2011).

Neste capítulo serão introduzidos, brevemente, três filtros amplamente difundidos na literatura para amostrar, sequencialmente, θ_t e Ψ conjuntamente. O primeiro método é o filtro proposto por Liu e West (2001), o segundo método é o filtro proposto por Storvik (2002), e, por último, o filtro proposto por Carvalho et al. (2010) e Lopes et al. (2010), denominado de *Parameter Learning filter*.

5.1 Filtro de Liu e West

Liu e West (2001), combinaram duas ideias para desenvolver um filtro que permite estimar os estados e o vetor de parâmetros desconhecidos e estáticos, Ψ , sequencialmente. Nessa abordagem, os autores combinaram métodos que utilizam mistura de normais multivariadas para aproximar a distribuição *a posteriori* de Ψ , $p(\Psi|D_{t-1})$, com o até então, inovador, *Filtro de Partículas Auxiliar* proposto por Pitt e Shephard (1999). Eles incorporaram, também, uma evolução artificial para Ψ , que é o vetor de parâmetros estáticos, sem a perda da informação associada.

Considere um conjunto de partículas $\{\theta_{t-1}^i, \Psi_{t-1}^i\}_{i=1}^N$ que aproximam a distribuição *a posteriori* conjunta $p(\theta_{t-1}, \Psi|D_{t-1})$ tal que $p(\Psi|D_{t-1})$ pode ser aproximado, via método de suavização por kernel, por

$$p(\Psi|D_{t-1}) \approx \sum_{i=1}^N w_{t-1}^i N(\theta|m_{t-1}^i, h^2V_{t-1}), \quad (5.1)$$

¹amostragem *offline* - Métodos MCMC

onde a média m_{t-1}^i é dada por

$$\begin{aligned} m_{t-1}^i &= a\Psi_{t-1}^i + (1-a)\bar{\Psi}_{t-1}, \\ \bar{\Psi}_{t-1} &= \frac{\sum_{i=1}^N \Psi_{t-1}^i}{N}, \\ a &= \sqrt{1-h^2}, \\ h^2 &= 1 - \left(\frac{3\delta}{2\delta}\right)^2, \end{aligned}$$

onde δ é um *fator de desconto*, definido no intervalo $(0, 1]$, que controla o nível de suavização do estimador do tipo kernel. O parâmetro a é conhecido como parâmetro de afinação, ou de ajuste, ou ainda de sintonia (*tunning parameter*). A variância V_{t-1} é dada por

$$V_{t-1} = \frac{\sum_{i=1}^N (\Psi_{t-1}^i - \bar{\Psi})(\Psi_{t-1}^i - \bar{\Psi})'}{N}. \quad (5.2)$$

O subscrito t em Ψ_t serve, apenas, para indicar que as amostras vieram de $p(\Psi|D_t)$. Uma vez que, em geral, como no APF, $p(y_t|\theta_{t-1}, \Psi)$ é complicada e/ou $p(\theta_t|\theta_{t-1}, \Psi_t, D_t)$ não é de fácil amostragem, então Liu e West (2001) reamostram as partículas antigas com pesos proporcionais a $p(y_t|\mu_{t-1}, m_t)$, onde $\mu_t = E(\theta_t|\theta_{t-1}, \Psi)$, e m_t descrito acima. Dessa forma, Ψ_t é propagado a partir da densidade de propagação $p(\Psi_t|\Psi_{t-1})$, enquanto θ_t é propagado, condicionalmente a Ψ_t , a partir da densidade de evolução $p(\theta_t|\theta_{t-1}, \Psi_t)$.

As partículas propagadas (θ_t, Ψ_t) possuem pesos associados proporcionais a

$$w_t \propto \frac{p(y_t|\theta_t, \Psi_t)}{p(y_t|\mu_t, m_t)}.$$

O algoritmo de Liu e West pode ser resumido da seguinte forma (Liu e West 2001):

Algoritmo de Liu e West (2001)

1. Para cada $i = 1, \dots, N$, identifique a estimativa pontual a *priori* dada por (μ_{t+1}^i, m_t^i) , onde

$$\begin{aligned}\mu_{t+1}^i &= E(\theta_{t+1} | \theta_t^i, \Psi_t^i), \\ m_t^i &= a\Psi_t^i + (1-a)\bar{\Psi}_t.\end{aligned}$$

2. Amostre uma variável inteira auxiliar, a partir do conjunto $\{1, \dots, N\}$, com probabilidade proporcional a

$$g_{t+1}^i = w_t^i p(y_{t+1} | \mu_{t+1}^i, m_t^i),$$

e denote por k o índice amostrado.

3. Amostre um novo vetor de parâmetros Ψ_{t+1}^k , a partir do k -ésimo componente da mistura de normais, ou seja

$$\Psi_{t+1}^k \sim N(\cdot | m_{t+1}^k, \Psi_{t+1}^k).$$

4. Calcule o valor do vetor de estados θ_{t+1}^k , a partir da equação do sistema

$$p(\cdot | x_t^k, \Psi_{t+1}^k).$$

5. Calcule o valor do peso correspondente:

$$w_{t+1}^k \propto \frac{p(y_{t+1} | \theta_{t+1}^k, \Psi_{t+1}^k)}{p(y_{t+1} | \mu_{t+1}^k, \Psi_{t+1}^k)}.$$

6. Repita os passos (2)-(5), várias vezes, para produzir a aproximação final da distribuição a *posteriori* $(\theta_{t+1}^k, \Psi_{t+1}^k)$, com pesos w_{t+1}^k .

A escolha adequada do parâmetro a^2 é de fundamental importância para uma melhor aproximação da mistura de normais à distribuição $p(\Psi | D_t)$. Na prática, uma escolha comum para tal parâmetro é algo em torno de 0.98 ou mais (Lopes e Tsay, 2011).

²Na literatura esse parâmetro é chamado de *tuning parameter*.

5.2 Filtro de Storvik

No algoritmo proposto por Storvik (2002) para a estimação da distribuição a *posteriori* do vetor de parâmetro estáticos Ψ , considera-se a distribuição condicional de Ψ dado as informações e o vetor de estados, i.e, $p(\Psi|\theta_t, D_t)$, escrita em função de um conjunto de baixa dimensionalidade, s_t , de estatísticas condicionalmente suficientes. Dessa forma, escreve-se $P(\Psi|\theta_t, D_t)$ em termos de $P(\Psi|s_t)$, em que as estatísticas s_t podem ser calculadas recursivamente em função da tripla (s_t, θ_t, y_t) . Sendo assim, o algoritmo de Storvik é visto como uma extensão do *filtro bootstrap* com alguns passos adicionais de atualização sequencial das estatísticas suficientes e amostragens de Ψ (Lopes e Tsay 2010). O algoritmo é resumido da seguinte forma:

Algoritmo de Storvik (2002):

1. Propagar $\{\theta_{t-1}^i\}_{i=1}^N$ para $\{\tilde{\theta}_{t-1}^i\}_{i=1}^N$ via $q(\theta_t|\theta_{t-1}, \Psi, D_t)$;
2. Reamostrar $\{(\theta_t, s_{t-1})^i\}_{i=1}^N$ a partir de $\{(\tilde{\theta}_t, s_{t-1})^i\}_{i=1}^N$ com pesos

$$w_t^i \propto \frac{p(y_t|\tilde{\theta}_t^i, \Psi)p(\tilde{\theta}_t^i|\theta_{t-1}^i, \Psi)}{q(\tilde{\theta}_t^i|\theta_{t-1}^i, \Psi, D_t)}; \quad (5.3)$$

3. Calcular as estatísticas suficientes $s_t = S(s_{t-1}, \theta_t, y_t)$;
4. Amostrar Ψ a partir de $p(\theta|s_t)$.

Segundo Lopes e Tsay (2011), os exercícios de simulações feitos em Storvik (2002) são baseados na regra de propagação às cegas (*blind*), i.e, sem contar com a informação proveniente dos dados, ou seja, $q(\theta_t|\theta_{t-1}, \Psi, D_t) = p(\theta_t|\theta_{t-1}, \Psi)$. No entanto, na etapa da reamostragem, a propagação é feita com pesos $w_t \propto p(y_t|\theta_t, \Psi)$, incluindo, assim, informação sobre os dados. Como todo filtro de partícula com propagação às cegas, tal como o *filtro bootstrap*, este filtro sofre degeneração das partículas que, por sua vez, compromete diretamente a estimativa sequencial dos parâmetros (Lopes e Tsay, 2011).

5.3 Particle Learning

Carvalho et al. (2010) descreveram métodos para filtragem sequencial, *particle learning* (PL)³ e suavização para o modelo de espaço de estados geral. Eles estenderam a mistura de filtro de Kalman proposto por Chen e Liu (2000) permitindo a estimação de parâmetros estáticos. Foram realizadas várias simulações afim de verificar se o algoritmo PL supera o algoritmo de Liu e West (2001) e o filtro de Storvik (2002). Segundo os autores, a vantagem do algoritmo PL frente aos concorrentes é mais evidente para séries longas de tempo.

O método PL possui duas características principais: primeiro, as estatísticas suficientes s_t são usadas para representar a distribuição a *posteriori* de Ψ , e as estatísticas suficientes para os estados latentes, s_t^θ , são utilizadas sempre que a estrutura do modelo permite. Isso implica na redução da variância dos pesos amostrais, aumentando a eficiência do algoritmo.

Em segundo lugar, ao contrário de outras abordagens que primeiro propagam e em seguida reamostram as partículas, o algoritmo PL reamostra e depois propaga. Isso evita a deterioração da partícula associada aos métodos do tipo SIR.

O algoritmo PL pode ser resumido nos seguintes passos (Carvalho et al., 2010; Lopes e Tsay, 2011):

Algoritmo de Carvalho et al. (2010)

1. Reamostragem ($\tilde{\Psi}, \tilde{s}_{t-1}^\theta, \tilde{s}_{t-1}$) a partir de $(\Psi, s_{t-1}^\theta, s_{t-1})$ com pesos $w_t \propto p(y_t | s_{t-1}^\theta, \Psi)$.
2. Amostre θ_t a partir de $p(\theta_t | \tilde{s}_{t-1}^\theta, \tilde{\Psi}, D_t)$.
3. Atualize as estatísticas suficientes do parâmetro, s_t , da seguinte maneira: $s_t = S(\tilde{s}_{t-1}, \theta_t, y_t)$.
4. Amostre Ψ a partir de $p(\Psi | s_t)$.
5. Atualize as estatísticas suficientes dos estados latentes: $s_t^\theta = K(\tilde{s}_{t-1}^\theta, \Psi, y_t)$.

³Uma tradução razoável para *Particle Learning* seria "Aprendizado de Partícula". Aqui será utilizado o nome em inglês.

5.4 Suavização

Godsill, Doucet e West (2004) propuseram uma abordagem para realizar suavização em modelos de espaço de estados gerais para os quais os parâmetros estáticos Ψ são conhecidos. Afim de obter representações amostrais de $p(\theta_{0:T}|D_T)$, é usada a seguinte fatoração

$$p(\theta_{0:T}|D_T) = p(\theta_T|D_T) \prod_{t=0}^{T-1} p(\theta_t|\theta_{(t+1):T}, D_T) \quad (5.4)$$

onde

$$\begin{aligned} p(\theta_t|\theta_{(t+1):T}, D_T) &= p(\theta_t|\theta_{t+1}, D_t) = \frac{p(\theta_t|D_t)p(\theta_{t+1}|\theta_t)}{p(\theta_{t+1}|D_t)} \\ &\propto p(\theta_t|D_t)p(\theta_{t+1}|\theta_t). \end{aligned} \quad (5.5)$$

Então, é possível obter uma aproximação da partícula modificada

$$p(\theta_t|\theta_{t+1}, D_T) \approx \sum_{m=1}^N w_{t|t+1}^{(m)} \delta_{\theta_t^{(m)}}(\theta_t),$$

com

$$w_{t|t+1}^m = \frac{w_t^{(m)} p(\theta_{t+1}|\theta_t^{(m)})}{\sum_{i=1}^N w_t^{(i)} p(\theta_{t+1}|\theta_t^{(i)})}. \quad (5.6)$$

Sendo assim, tem-se o seguinte algoritmo para suavização

Algoritmo de Godsill, Doucet e West (2004):

1. No tempo T escolha $\tilde{\theta}_T = \theta_T^{(m)}$ com probabilidade $w_T^{(m)}$.
2. Para $t = (T - 1), (T - 2), \dots, 0$.
 - (a) Calcule $w_{t|t+1}^{(m)} \propto w_t^{(m)} p(\tilde{\theta}_{t+1}|\theta_t^{(m)})$ para $m = 1, \dots, M$;
 - (b) Escolha $\tilde{\theta}_t = \theta_t^{(m)}$ com probabilidade $w_{t|t+1}^{(m)}$.
3. Tome $\tilde{\theta}_{0:T} = (\tilde{\theta}_0, \dots, \tilde{\theta}_T)$ como uma realização aproximada da distribuição $p(\theta_{0:T}|D_T)$.

Os passos 1 e 2 podem ser repetidos várias vezes para obter realizações de $p(\theta_{0:T}|D_T)$ aproximadamente independentes entre si. Godsill, Doucet e West (2004) mostraram a convergência, em erro quadrático médio, das trajetórias suavizadas, testaram esse

método em uma aplicação relacionada a processamento de sinais e fala, que foi representado por modelos autoregressivos parametrizados em termos de coeficientes de correlação parciais variáveis no tempo.

Carvalho et al. (2010) estenderam o algoritmo acima considerando os parâmetros Ψ desconhecidos. O algoritmo é descrito a seguir:

Algoritmo de Carvalho et al. (2010)

1. No tempo T , escolha aleatoriamente $(\tilde{\theta}_T, \tilde{s}_T)$ a partir da densidade $p(\theta_T, s_T | D_T)$, tais como as obtidas pelo algoritmo PL descrito na seção anterior. Então, amostramos $\tilde{\Psi} \sim p(\Psi | \tilde{s}_T)$.
2. Para $t = (T - 1) : 0$, escolha $\tilde{\theta}_t = \theta_t^{(m)}$ a partir das partículas ponderadas filtradas $\{(\theta_t, w_{t|t+1})^{(m)}; m = 1 : M\}$ com pesos, $w_{t|t+1}^{(m)} \propto w_t^{(m)} p(\tilde{\theta}_{t+1} | \theta_t^{(m)}, \tilde{\Psi})$.
3. Tome $\tilde{\theta}_{0:T} = (\tilde{\theta}_0, \dots, \tilde{\theta}_T)$ como uma realização aproximada da distribuição $p(\theta_{0:T} | D_t)$.

Note que os cálculos realizados nos passos 2 e 3 acima também podem ser empreendidos se o processo de filtragem não for feito por meio de algoritmos que utilizam estatísticas suficientes, que é o caso do algoritmo de Liu e West (2001).

5.5 Exemplo

Aqui reproduziremos um exemplo apresentado por Lopes e Tsay (2011), no qual foi comparado o desempenho dos 3 filtros de partículas apresentados neste capítulo⁴. Os dados utilizados para a construção desse exemplo foram simulados de um modelo de nível local descrito pelas equações abaixo, para $t = 1, \dots, T$.

- **Equação da observação:**

$$y_t = \theta_t + v_t, \quad v_t \sim N(0, \sigma^2);$$

- **Equação do sistema (ou evolução):**

$$\theta_t = \alpha + \beta\theta_{t-1} + w_t, \quad w_t \sim N(0, \tau^2);$$

⁴A programação utilizada foi disponibilizada pelo professor Hedibert Freitas Lopes enquanto professor da Universidade de Chicago.

onde

$$\begin{aligned} \theta_0|D_0 &\sim N(m_0, C_0), \quad \sigma^2 \sim IG\left(\frac{n_0}{2}, \frac{n_0\sigma_0^2}{2}\right) \\ (\alpha, \beta)|\tau^2 &\sim N_2(b_0, \tau^2 B_0), \quad \text{e } \tau^2 \sim IG\left(\frac{v_0}{2}, \frac{v_0\tau_0^2}{2}\right). \end{aligned} \quad (5.7)$$

O vetor de parâmetros estáticos a ser estimado pelos filtros é dado por $\Psi = (\alpha, \beta, \tau^2, \sigma^2)$. A distribuição a *priori* de Ψ é $p(\Psi) = p(\sigma^2)p(\tau^2)p(\alpha, \beta|\tau^2)$.

A série simulada possui comprimento $T = 200$ e foi simulada usando $\Psi = (0, 0.9, 0.5, 1)$ e $\theta_0 = 0$. Os hiperparâmetros da distribuição a *priori* são $m_0 = 0$, $C_0 = 10$, $b_0 = (0, 0.9)'$, $B_0 = I_2$, $n_0 = v_0 = 10$, $\tau_0^2 = 0.5$ e $\sigma_0^2 = 1$. A escolha desses valores para os hiperparâmetros reflete uma certa *priori* não informativa. O desempenho dos filtros é avaliado através da execução de cada algoritmo $R = 100$ vezes, baseado em $N = 1000$ partículas. Um PL baseado em $N = 100.000$ partículas foi executado e usado como *Benchmark* para a comparação.

Seja $q(\gamma, \alpha, t)$ o percentil 100α de $p(\gamma|D_t)$, onde γ é um elemento de Ψ . Como medida de qualidade do ajuste dos filtros, Lopes e Tsay (2011) usaram a seguinte versão do erro quadrático médio

$$EQM(\gamma, \alpha, f, t) = \sum_{t,r} \frac{[q(\gamma, \alpha, t) - q_{fr}(\gamma, \alpha, t)]^2}{R},$$

para o filtro f (LW, STORVIK ou PL) e replicação $r = 1, \dots, R$. Por fim, uma adaptação completa é implementada para os três filtros. Em outras palavras, o filtro LW difere do PL apenas por meio da estimação sequencial de Ψ , o filtro de Storvik difere do PL apenas na medida em que o filtro Storvik primeiro propaga e depois reamostra as partículas, enquanto o filtro PL primeiro reamostra e depois propaga a partícula.

Os resultados estão resumidos nas Figuras 5.1 e 5.2. Podemos observar que os filtros Storvik e PL são significativamente melhores que o filtro LW, enquanto que o filtro PL é moderadamente melhor que o filtro Storvik, particularmente na estimação do par (σ^2, τ^2) .

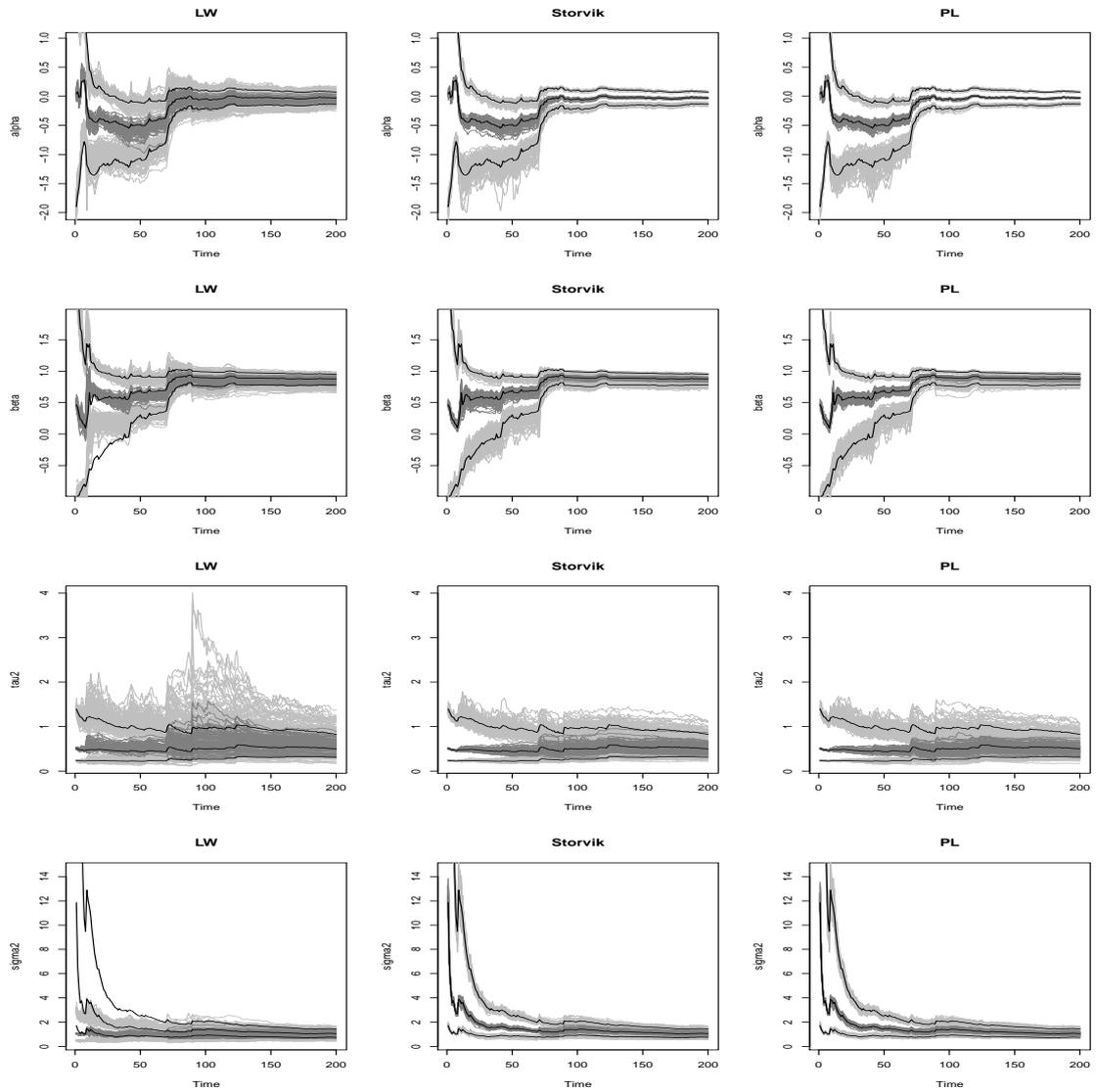


Figura 5.1: Comparação entre os filtros LW, STORVIK e PL. Percentis de $p(\Psi|D_t)$ (2,5%, 50% e 97,5%) baseados em 100 replicações de cada filtro com 1000 partículas (Linhas cinzas). As linhas pretas são baseados em um filtro PL com 100.000 partículas. As estimaticas do filtro LW está ilustrada na coluna a esquerda, o filtro do Storvik na coluna central e o filtro PL na coluna a direita De cima para baixo temos os componentes $\Psi = (\alpha, \beta, \tau_2, \sigma^2)$ (Lopes e Tsay, 2011).

Nos próximos capítulos serão descritos detalhadamente os métodos que foram utilizados para a criação dos algoritmos utilizados nesta dissertação, os quais incorporam quebras estruturais nas séries temporais. Primeiramente, no Capítulo 6, será descrito o modelo de regressão dinâmica proposto por McCormick et al. (2012). Em seguida, no Capítulo 7, serão descritos os filtros de partículas proposto por Chopin (2007) e por Caron et al. (2012).

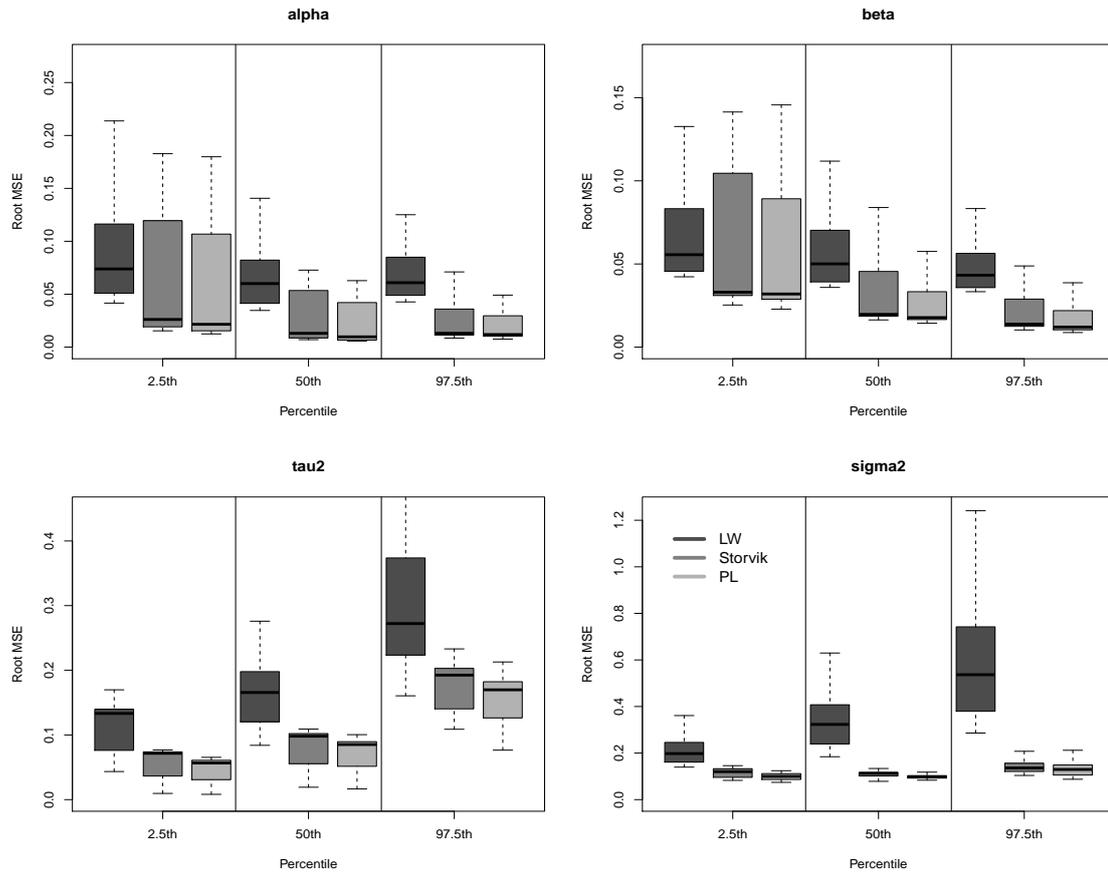


Figura 5.2: Comparação entre os filtros LW, STORVIK e PL. Raiz do erro médio quadrático de 100 replicações para cada filtro. Todos os filtros são baseados em 1000 partículas, os EMQs são calculados sobre o filtro PL com 100.000 partículas, (Lopes e Tsay, 2011).

Capítulo 6

Modelos de Regressão Dinâmica com Pontos de Mudança

Muitas séries, tais como sequências de DNA, preços de estoques, poluição do ar (Achcar et al., 2008), séries de longa duração (Chopin 2007), crescimento bacteriológico (Whittaker e Fruhwirth-Schatter 1994), entre outros, apresentam heterogeneidade temporal. Nesse contexto, uma abordagem usual consiste em segmentar uma sequência de observações y_1, y_2, \dots, y_T escolhendo uma sequência de quantidades que indicam em que posição, no tempo, ocorreram mudanças estruturais na série $0 < \tau_1 < \tau_2 < \dots < \tau_m < T$, tal que as observações sejam homogêneas dentro dos segmentos, no sentido de apresentarem a mesma estrutura de modelo, e heterogêneas entre os segmentos.

O número de publicações relacionadas com métodos de ponto de mudanças, desde o trabalho seminal de Hinkley (1970), é extremamente elevado. Achcar et al. (2008) utilizaram um processo de Poisson não homogêneo para modelar pontos de mudanças em dados sobre poluição de ar na cidade do México. Whittaker e Fruhwirth-Schatter (1994) utilizaram modelos de multiprocessamento (West e Harrison 1997) para construir um modelo dinâmico de pontos de mudanças para detectar o início do crescimento de infecções bacteriológicas.

Existem métodos que apenas captam mudanças estruturais na série e não contam *quantas* ou localizam *onde* tais mudanças ocorreram. Esse é o caso dos modelos de regressão dinâmico desenvolvidos por Raftery et al. (2010) e McCormick et al. (2012).

Nesses modelos incorpora-se um fator de desconto na variância da distribuição dos estados, que aumenta a incerteza em períodos de grande volatilidade. Tal parâmetro flexibiliza o modelo de uma tal maneira que mudanças bruscas na estrutura da série são corretamente captadas pelo modelo.

Neste capítulo abordaremos os modelos de regressão dinâmica propostos por Raftery et al. (2010) e McCormick et al. (2012). Posteriormente aplicaremos a metodologia descrita ao caso em que os dados seguem a distribuição de Poisson.

6.1 Modelo de regressão dinâmica para problemas de classificação

Nesta seção pretende-se descrever, de maneira generalizada e unificada, os modelos de regressão dinâmica propostos por Raftery et al. (2010) e McCormick et al. (2012). Utilizando um modelo de espaço de estados, Raftery et al. (2010) e McCormick et al. (2012) propuseram um procedimento útil para ajustar séries temporais em que o processo gerador dos dados sofre mudanças ao longo do tempo. Trabalhando com séries temporais para dados binários, os autores desenvolveram um processo de estimação on-line que permite incorporar a incerteza com respeito ao modelo, considerando-se um conjunto de K modelos, e as mudanças dos parâmetros, ao longo do tempo, de cada um dos modelos.

Apesar de tal metodologia não tratar especificamente dos pontos de mudança, ela é muito útil como uma primeira abordagem ao problema de detecção de pontos de mudança. Além disso, as estimativas obtidas com tal metodologia podem servir como valores iniciais para a estimação dos pontos de mudança de uma série temporal via filtro de partículas.

O modelo proposto por McCormick et al. (2012) tem a grande vantagem de ser muito simples, sem ser simplista, de permitir estimativas on-line para a imediata atualização dos parâmetros com a chegada de uma nova observação e de permitir a acomodação de mudanças estruturais na série temporal ao considerar um mecanismo auto-ajustável através de um fator de desconto dinâmico.

McCormick et al. (2012) trabalharam com o caso específico de séries temporais

binárias. No entanto, as idéias do artigo podem ser extendidas para uma série temporal seguindo uma distribuição genérica.

O método de estimação sequencial proposto por McCormick et al. (2012) é compreendido em dois passos: *atualização* e *predição*. Para tanto, considere uma variável resposta, y_t , e um conjunto de preditores $\mathbf{x}_t = (x_{1,t}, x_{2,t}, \dots, x_{d,t})$, tais que, no tempo t ,

$$y_t \sim p(y_t|\mu_t) \text{ com } \mu_t = g(\theta) = \mathbf{x}_t^T \theta_t,$$

sendo θ_t um vetor d -dimensional de coeficientes da regressão e $g(\theta_t)$ é uma função desses coeficientes¹. Em um dado tempo t , o procedimento proposto por McCormick et al. (2012) usa a moda a *posteriori* de θ a partir do tempo $t - 1$, utilizando-a para construir a *priori* no tempo t , conforme visto ao longo de toda a dissertação. Isto é feito, primeiramente, usando-se a informação obtida no tempo $t - 1$, para construir a estimativa dos parâmetros do tempo t , através da distribuição preditiva de θ . Essa etapa é denominada de *predição*. A equação de predição é, então, combinada com o dado observado no tempo t , y_t , na atualização dos parâmetros estimados.

Etapa da Predição

Seja a equação do sistema descrita por $\theta_t = \theta_{t-1} + w_t$, onde os w_t 's são vetores aleatórios independentes $N(0, W_t)$ (Raftery et al., 2010). Para toda a característica observada no passado, D_{t-1} , e utilizando-se valores iniciais razoáveis, a estimação recursiva se inicia supondo que (McCormick et al. 2012)

$$(\theta_{t-1}|D_{t-1}) \approx N(\hat{\theta}_{t-1}, \hat{\Sigma}_t).$$

Então, a predição é aproximada por

$$(\theta_t|D_{t-1}) \approx N(\hat{\theta}_{t-1}, R_t), \tag{6.1}$$

onde

$$R_t = \frac{\hat{\Sigma}_{t-1}}{\lambda_t}. \tag{6.2}$$

¹Podemos enxergar essa função como uma função de ligação, em analogia aos modelos lineares generalizados.

O fator de desconto, λ_t , é especificado através equação (6.2), com $0 < \lambda_t < 1$. Conforme visto no Capítulo 2, sobre modelos dinâmicos lineares, o modelo pode ser especificado, sem o desconto, utilizando-se a matriz de covariâncias W_t . Sendo assim, tem-se $R_t = \hat{\Sigma}_{t-1} + W_t$. Essa abordagem, entretanto, requer a especificação de toda a matriz de covariâncias W_t , e isso pode implicar em um grande esforço computacional, além de aumentar a complexidade do modelo. Tais implicações tornam a metodologia proposta por McCormick et al. (2012) muito atraente.

Atualização

Uma vez realizada a etapa de predição, combina-se a equação de predição (6.1) com a observação no tempo t , para atualizar as estimativas dos parâmetros. A distribuição *a posteriori* das estimativas atualizadas, θ_t , pode ser escrita da seguinte forma

$$p(\theta_t|D_t) \propto p(y_t|\theta_t)p(\theta_t|D_{t-1}). \quad (6.3)$$

Portanto, a equação (6.3) é o produto da equação de predição e da verossimilhança no tempo t . A expressão do lado direito de (6.3) tipicamente não possui forma fechada, podendo ser aproximada utilizando-se métodos MCMC. Entretanto, para viabilizar os cálculos com baixo esforço computacional, McCormick et al. (2012) aproximam o lado direito de (6.3) por uma distribuição Normal, tendo, antes, tomado uma aproximação Normal para a distribuição $(\theta_t|D_{t-1})$. Dessa forma, toma-se

$$p(\theta_t|D_t) \propto p(y_t|\theta_t)N(\hat{\theta}_{t-1}, R_t). \quad (6.4)$$

Considere

$$\begin{aligned} l(\theta_t) &= \log [p(y_t|\theta_t)p(\theta_t|D_{t-1})] \\ &\approx \log p(y_t|\theta_t) - \frac{1}{2} \left[\theta_t^T R_t^{-1} - 2\hat{\theta}_{t-1}^T R_t^{-1} \theta_t \right], \end{aligned} \quad (6.5)$$

e defina, via Newton-Raphson

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \left[D^2 l(\hat{\theta}_{t-1}) \right]^{-1} D l(\hat{\theta}_{t-1}), \quad (6.6)$$

em que $Dl(\theta)$ é a primeira derivada de $l(\theta)$ e $D^2 l(\theta)$ é a segunda derivada de $l(\theta)$. Para atualizar a variância dos estados, utilizamos $\hat{\Sigma}_t = \left[-D^2 l(\hat{\theta}_{t-1}) \right]$.

Para a estimação do parâmetro de desconto λ_t , McCormick et al. (2012) propuseram um procedimento utilizando a verossimilhança preditiva,

$$f(y_t|D_{t-1}) = \int_{\theta_t} p(y_t|\theta_t, D_{t-1})p(\theta_t|D_{t-1})d\theta_t. \quad (6.7)$$

Tal integral tipicamente não possui forma fechada. Entretanto, ela pode ser aproximada, de maneira satisfatória, utilizando-se a aproximação de Laplace (Tierney e Kadane, 1986), apresentada nos apêndices. Utilizando essa aproximação tem-se

$$f(y_t|D_{t-1}) \approx (2\pi)^{d/2} \left[D^2l(\hat{\theta}_t) \right]^{-1} p(y_t|D_{t-1}, \hat{\theta}_t)p(\hat{\theta}_t|D_{t-1}). \quad (6.8)$$

Para atualizar o parâmetro λ_t , para cada tempo t , escolhe-se o valor de λ_t que maximiza a equação (6.8), ou seja

$$\lambda_t = \arg \max_{\lambda_t} f(y_t|D_{t-1}). \quad (6.9)$$

Uma vez que $\lambda_t \in (0, 1]$, necessita-se de uma maximização restrita para λ_t , segundo McCormick (2012), uma alternativa a essa abordagem seria estimar λ_t de uma maneira inteiramente Bayesiana (Seção 12.3 de West e Harrison, 1997) e maximizar a *posteriori* completa. Entretanto, essa abordagem requer um grande esforço computacional.

O procedimento descrito acima pode ser resumido no seguinte algoritmo:

Algoritmo de McCormick et al. (2012):

1. Inicie o algoritmo. Por exemplo, pode-se iniciar o algoritmo fixando-se: $\hat{\theta}_0 = g(y_1 + 0.00001)$, $\hat{y}_1 = g^{-1}(x_1\hat{\theta}_0)$, $\lambda_1 = 0.99$, $\Sigma_1 = 100$, $R_1 = \Sigma_1/\lambda_1$, $Dl(\hat{\theta}_0)$ e $D^2l(\hat{\theta}_0)$;
2. Calcule $\hat{y}_t = g^{-1}(x_t\hat{\theta}_{t-1})$;
3. Calcule $R_t = \hat{\Sigma}_{t-1}/\lambda_{t-1}$;
4. Atualize $Dl(\hat{\theta}_{t-1})$ e $D^2l(\hat{\theta}_{t-1})$;
5. Atualize o vetor de parâmetros $\hat{\theta}_t$ de acordo com as equações (6.5) e (6.6);
6. Atualize a matriz de covariâncias, $\hat{\Sigma}_t = \left[-D^2l(\hat{\theta}_{t-1}) \right]^{-1}$;
7. De posse dos valores obtidos nos passos anteriores, maximize a equação (6.8), utilizando métodos de maximização restrita, para obter o valor de λ_t ;
8. Atualize, novamente, $R_t = \hat{\Sigma}_{t-1}/\lambda_t$.

6.1.1 *Dynamic Model Averaging*

Para o caso de multimodelos (*Model Averaging*) que permitem a incorporação de mecanismos úteis para que sejam levados em conta a incerteza a respeito do modelo adotado, considere K modelos candidatos (M_1, \dots, M_K) . A principal característica desse método é que as probabilidades de cada modelo também são dinâmicas, permitindo, assim, maior flexibilidade ao longo do tempo e também que eventuais sobreajustamentos de cada observação sejam evitados (McCormick et. al., 2012).

Seja L_t o indicador do modelo, tal que se $L_t = k$, o processo é governado pelo modelo M_k no tempo t . Para o caso de multimodelos tem-se

$$(y_t | L_t = k) \sim p(y_t | g(\theta_t^k)),$$

onde

$$g^{-1}(\lambda_t^k) = x_t^{(k)T} \theta_t^k.$$

Segundo Raftery et. al. (2010), atualiza-se θ_t^k condicionalmente a $L_t = k$. Como no caso de um único modelo, o processo de estimação ocorre em dois passos: predição e atualização. Para o caso de multimodelos, entretanto, o espaço de estados em cada tempo consiste agora do par (L_t, θ_t) , onde $\theta_t = (\theta_t^1, \dots, \theta_t^K)$. Agora a estimação recursiva ocorre no par (L_t, θ_t)

$$\sum_{l=1}^K p(\theta_t^l | L_t = l, D_{t-1}) p(L_t = l | D_{t-1}). \quad (6.10)$$

Note que (6.10) é uma mistura de distribuições, sendo assim, os passos da predição e da atualização ocorrem, separadamente, para o modelo, L_t , e para os parâmetros dentro do dado modelo. De acordo com McCormick et. al. (2012), o processo de estimação é dado por

Processo de estimação:

- **Passo 1:** Predição para o modelo L_t .

A equação de predição do modelo é dada por

$$p(L_t = k | D_{t-1}) = \sum_{l=1}^K p(L_{t-1} = l | D_{t-1}) p(L_t = k | L_t = l).$$

Para evitar especificar uma matriz de transição $K \times K$ para os termos $p(L_t = k|L_t = l)$, McCormick et. al. (2012) atualiza a matriz utilizando um fator de desconto α_t , em que $0 < \alpha_t \leq 1$,

$$p(L_t = k|D_{t-1}) = \frac{p(L_{t-1} = k|D_{t-1})^{\alpha_t}}{\sum_{l=1}^K p(L_{t-1} = l|D_{t-1})^{\alpha_t}}, \quad (6.11)$$

O fator aumenta a incerteza, tornando a distribuição de L_t mais platicúrtica. Com essa proposta, apenas um parâmetro precisa ser especificado, ao invés da matriz $K \times K$, simplificando, consideravelmente, a especificação do modelo.

- **Passo 2:** Atualização do modelo. A equação de atualização do modelo é dada por

$$p(L_t = k|D_{t-1}) = \frac{w_t^k}{\sum_{l=1}^K w_t^l}, \quad (6.12)$$

onde

$$w_t^l = p(L_{t-1} = l|D_{t-1})f^l(y_t|D_{t-1}). \quad (6.13)$$

O ajuste de α_t é feito usando a verossimilhança preditiva entre todos os modelos candidatos, $f(y_t|D_{t-1})$, tal que

$$f_{y_t|D_{t-1}} = \sum_{l=1}^K f^l(y_t|D_{t-1})p(L_t = l|D_{t-1}),$$

onde $p(L_t = l|D_{t-1})$ é dado por (6.11). Seleciona-se α_t tal que

$$\operatorname{argmax}_{\alpha_t} \sum_{l=1}^K f^l(y_t|D_{t-1})p(L_t = l|D_{t-1}). \quad (6.14)$$

Para prever y_t utiliza-se

$$\hat{y}_t^{DMA} = \sum_{l=1}^K p(L_t = l|D_t)\hat{y}_t^l,$$

onde \hat{y}_t^l é a resposta predita para o modelo l no tempo t .

6.2 Regressão dinâmica Poisson Bayesiana

Nesta seção descreveremos o modelo de regressão dinâmica Poisson bayesiana como um caso particular do modelo de regressão dinâmica desenvolvido na Seção 6.1 do Capítulo 6. Para tanto, considere uma contagem como variável resposta, y_t , e um conjunto de preditores $\mathbf{x}_t = (x_{1,t}, x_{2,t}, \dots, x_{d,t})$, tais que, no tempo t ,

$$y_t \sim \text{Poisson}(\xi_t) \text{ onde } \log(\xi_t) = \mathbf{x}_t^T \theta_t,$$

onde θ_t é um vetor d -dimensional de coeficientes da regressão.

Conforme a Seção 6.1 do Capítulo 6, primeiramente desenvolvemos a equação de predição e, como feito em Raftery et al. (2010), assumimos que a equação de evolução é dada por $\theta_t = \theta_{t-1} + w_t$, onde os w_t 's são vetores aleatórios independentes $N(0, W_t)$. Para toda a característica observada no passado, D_{t-1} , e valores iniciais, a estimação recursiva se inicia supondo

$$(\theta_{t-1} | D_{t-1}) \approx N(\hat{\theta}_{t-1}, \hat{\Sigma}_t).$$

Então a equação de predição é dada por

$$(\theta_t | D_{t-1}) \approx N(\hat{\theta}_{t-1}, R_t), \quad (6.15)$$

onde

$$R_t = \frac{\hat{\Sigma}_{t-1}}{\lambda_t}. \quad (6.16)$$

De acordo com a Seção 6.1 do Capítulo 6, a estimação de θ_t é feita utilizando o método de Newton-Raphson. Sendo assim, a atualização de θ_t é dada por

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \left[D^2 l(\hat{\theta}_{t-1}) \right]^{-1} D l(\hat{\theta}_{t-1}), \quad (6.17)$$

onde $l(\theta) = \log(p(y_t | \theta)p(\theta | D_{t-1}))$, $Dl(\theta)$ é a primeira derivada de $l(\theta)$ e $D^2l(\theta)$ é a segunda derivada de $l(\theta)$.

Para o modelo Poisson, e considerando-se a aproximação Gaussiana de $(\theta_t|D_{t-1})$, tem-se

$$l(\theta_t) \approx -\exp(x_t^T \theta_t) + y_t x_t^T \theta_t - \frac{1}{2} \left(\theta_t^T R_t^{-1} - 2\hat{\theta}_t^T R_t^{-1} \theta_t \right). \quad (6.18)$$

De (6.18) tem-se que a primeira derivada de $l(\theta)$, avaliada em $\hat{\theta}_{t-1}$, é dada por

$$Dl(\hat{\theta}_{t-1}) = (y_t - \hat{y}_t) \mathbf{x}_t.$$

em que $\hat{y}_t = \exp(\mathbf{x}_t^T \hat{\theta}_{t-1})$. A segunda derivada de $l(\theta_t)$, avaliada em $\hat{\theta}_{t-1}$, é dada por

$$D^2l(\hat{\theta}_{t-1}) = -\mathbf{x}_t \mathbf{x}_t^T \hat{y}_t - R_t^{-1}.$$

As demonstrações das identidades acima são apresentadas nos apêndices. Para atualizar a variância dos estados, utilizamos $\hat{\Sigma}_t = -\left[D^2l(\hat{\theta}_{t-1})\right]^{-1}$.

Para a estimação do parâmetro de desconto λ_t , McCormick et al. (2012) propuseram um procedimento usando a verossimilhança preditiva

$$f(y_t|D_{t-1}) = \int_{\theta_t} p(y_t|\theta_t, D_{t-1})p(\theta_t|D_{t-1})d\theta_t. \quad (6.19)$$

Através da aproximação de Laplace, a integral (6.19) pode ser aproximada por

$$f(y_t|D_{t-1}) \approx (2\pi)^{d/2} \left| \left[D^2l(\hat{\theta}_t) \right]^{-1} \right| p(y_t|D_{t-1}, \hat{\theta}_t) p(\hat{\theta}_t|D_{t-1}). \quad (6.20)$$

Para atualizar o parâmetro λ_t , para cada tempo t , escolhe-se o valor de λ_t que maximiza a equação (6.20), ou seja

$$\lambda_t = \arg \max_{\lambda_t} f(y_t|D_{t-1}). \quad (6.21)$$

Uma vez que $\lambda_t \in (0, 1]$, é necessário uma maximização restrita, no intervalo $(0, 1]$, da verossimilhança preditiva.

Para a modelo Poisson, o algoritmo, descrito na seção anterior, pode ser resumido da seguinte forma:

Regressão dinâmica Poisson:

1. Inicie o algoritmo. Como exemplo, pode-se iniciá-lo como: $\hat{\theta}_0 = \log(y_1 + 0.00001)$, $\hat{y}_1 = \exp(x_1 \hat{\theta}_0)$, $\lambda_1 = 0.99$, $\Sigma_1 = 100$, $R_1 = \Sigma_1 / \lambda_1$, $Dl(\hat{\theta}_0)$ e $D^2l(\hat{\theta}_0)$;
2. Calcule $\hat{y}_t = \exp(x_t \hat{\theta}_{t-1})$;
3. Calcule $R_t = \hat{\Sigma}_{t-1} / \lambda_{t-1}$;
4. Atualize $Dl(\hat{\theta}_{t-1})$ e $D^2l(\hat{\theta}_{t-1})$;
5. Atualize o vetor de parâmetros $\hat{\theta}_t$ de acordo com a equação (6.17);
6. Atualize a matriz de covariâncias, $\hat{\Sigma}_t = - [D^2l(\hat{\theta}_{t-1})]^{-1}$;
7. De posse dos valores obtidos nos passos anteriores, maximize a função (6.20), utilizando métodos de maximização restrita, para obter o valor de λ_t ;
8. Atualize, novamente, $R_t = \hat{\Sigma}_{t-1} / \lambda_t$.

6.2.1 Análise de dados simulados

Para ilustrar a flexibilidade em acomodar mudanças estruturais de uma série temporal pela abordagem do método proposto por McCormick et al. (2012), simulamos 3 bases de dados, duas delas com diferentes mudanças estruturais. Também simulamos uma base de dados segundo um modelo Poisson estático.

Para as duas primeiras simulações foram geradas bases de tamanho $T = 2000$, para a terceira base fixou-se $T = 1000$. As simulações podem ser resumidas como a seguir

- **Base de dados 1:** Mudança na inclinação de θ_t :

$$\theta_t = \begin{cases} -1 + 2t/1000 & : t < 1000 \\ 1 - 2t/1000 & : t \geq 1000 \end{cases}$$

Os dados simulados estão ilustrados na Figura 6.1.

A programação utilizada para gerar a base acima é dada por

Simulação - Base 1

```
n=1000
2 beta <- rep(0,n)
  beta2 <- rep(0,n)
4 for (t in 1:n){
    beta[t] = -1+2*t/n
6    beta2[t] = 1-2*t/n
  }
8 beta1 <- beta
  par(mfrow=c(2,1))
10 plot(beta1, type="l")
  plot(beta2, type="l")
12
  theta1=beta1
14 theta2=beta2

16 par <- rep(0,2*n)
  lambda <- par
18 y=par
  for (t in 1:length(par)){
20    if (t<=length(par)/2){
        par[t] = theta1[t]
22    } else if (t>=length(par)/2){
        par[t] = theta2[t-1000]
24    }
    lambda[t] <- exp(par[t])
26    y[t] <- rpois(1,lambda[t])
  }
28 par(mfrow=c(2,1))
  plot(par, type="l", ylim=c(-1.5,1.5))
30 abline(v=1000, col="red")
  plot(y, type="l")
32 abline(v=1000, col="red")
```

A programação utilizada para ajustar o modelo aos dados acima é dada por

Ajuste - Base 1

```
exemplo1 <- poisson.uni(Y=y,X=NULL, lambda0=0.95, c=0.95)
2 names(exemplo1)

4 par(mfrow=c(2,1))
  plot(par, type="l", lty=2, lwd=2, ylab="Theta", xlab="Tempo")
6 lines(exemplo1$estimates[-c(1,2),1], col="red")
  lines(exemplo1$estimates[-c(1,2),1]-2*exemplo1$derror[-c(1,2)],
8    col="blue")
```

```

lines(exemplo1$estimates[-c(1,2),1]+2*exemplo1$sderror[-c(1,2)],
10 col="blue")
abline(v=1000, lty=2, lwd=2)

```

A Figura 6.2 mostra as estimativas geradas pelo algoritmo detalhado há pouco. Pode-se observar que o algoritmo foi capaz de captar a mudança estrutural ocorrida na série.

- **Base de dados 2:** Mudança abrupta em θ_t , no tempo $t = 1000$. Os dados simulados estão ilustrados na Figura 6.3.

A programação utilizada para gerar a base acima é dada por

Simulação - Base 2

```

1  n=1000
   beta <- rep(0,n)
3  beta2 <- rep(0,n)
   for (t in 1:n){
5     beta[t] = 2
     beta2[t] = 1
7  }
   beta1 <- beta
9  par(mfrow=c(2,1))
   plot(beta1, type="l")
11 plot(beta2, type="l")

13 theta1=beta1
   theta2=beta2
15 par <- rep(0,2*n)
   lambda <- par
17 y=par
   for (t in 1:length(par)){
19     if (t<=length(par)/2){
       par[t] = theta1[t]
21     } else if (t>=length(par)/2){
       par[t] = theta2[t-1000]
23     }
     lambda[t] <- exp(par[t])
25     y[t] <- rpois(1,lambda[t])
   }
27 par(mfrow=c(2,1))
   plot(par, type="l", ylim=c(0,2.5))
29 abline(v=1000, col="red")
   plot(y, type="l")
31 abline(v=1000, col="red")

```

A programação utilizada para ajustar o modelo aos dados acima é dada por

Ajuste - Base 2

```
1 exemplo2 <- poisson.uni(Y=y,X=NULL, lambda0=0.95, c=0.95)
  names(exemplo2)
3
  par(mfrow=c(2,1))
5 plot(par, type="l", lty=2, lwd=2, ylab="Theta", xlab="Tempo")
  lines(exemplo2$estimates[-c(1,2),1], col="red")
7 lines(exemplo2$estimates[-c(1,2),1]-2*exemplo2$sderror[-c(1,2)],
  col="blue")
9 lines(exemplo2$estimates[-c(1,2),1]+2*exemplo2$sderror[-c(1,2)],
  col="blue")
11 abline(v=1000, lty=2, lwd=2)
```

A Figura 6.4 mostra as estimativas geradas pelo algoritmo detalhado há pouco. Pode-se ver que o algoritmo também foi capaz de captar a mudança estrutural ocorrida na série.

- **Base de dados 3:** Por fim, afim de mostrar que o modelo dinâmico Poisson pode ser visto como uma generalização do modelo de regressão estática Poisson, simulou-se um modelo regressão estática Poisson, o qual, para $t = 1, 2, \dots, 1000$, o preditor linear é dado por

$$\log(\lambda_t) = 1 + 0.2x_{t,1} + 0.5x_{t,2},$$

em que $X_1, X_2 \sim U(0, 1)$ e $y_t \sim \text{Poisson}(\exp(1 + 0.2x_{t,1} + 0.5x_{t,2}))$.

A programação utilizada para gerar dados utilizando a formulação acima é dada por

Simulação - Base 3

```
1 set.seed(456)
  x1 <- runif(1000)
3 x2 <- runif(1000)
  beta0 <- 1
5 beta1 <- -0.2
  beta2 <- 0.5
7 y <- rep(0,1000)
  eta <- exp(beta0+beta1*x1+beta2*x2)
9 for (i in 1:length(x1)){
  y[i] <- rpois(1,lambda=eta[i])
11 }
```

A programação utilizada para ajustar o modelo aos dados acima é dada por

Ajuste - Base 3

```

1 exemplo <- poisson.sm(Y=y, X=cbind(x1,x2), intercept=TRUE, lambda0=1, c=1)
  names(exemplo)
3
  par(mfrow=c(3,1))
5 plot(exemplo$estimates[,1],type="l")
  lines(exemplo$estimates[,1]-2*exemplo$sderror[,1], col="blue")
7 lines(exemplo$estimates[,1]+2*exemplo$sderror[,1], col="blue")

9 plot(exemplo$estimates[,2],type="l")
  lines(exemplo$estimates[,2]-2*exemplo$sderror[,2], col="blue")
11 lines(exemplo$estimates[,2]+2*exemplo$sderror[,2], col="blue")

13 plot(exemplo$estimates[,3],type="l")
  lines(exemplo$estimates[,3]-2*exemplo$sderror[,3], col="blue")
15 lines(exemplo$estimates[,3]+2*exemplo$sderror[,3], col="blue")

17 summary(glm(y~cbind(x1,x2), family=poisson))
  exemplo$estimates[length(y),]
19 exemplo$sderror[length(y),]

```

A estimativa dos parâmetros no tempo $T = 1000$ é dada na Tabela 6.1, onde encontram-se, também, as estimativas dos parâmetros obtido utilizando-se a função *glm*, a qual ajusta um modelo de Poisson estático. Pode-se observar que as estimativas pontuais (e seus respectivos erros padrão) estão muito próximos entre os modelos e também próximos dos valores reais, indicando que o modelo de regressão dinâmica de Poisson também pode ser utilizado para analisar dados estáticos. Sendo assim, esse modelo é uma generalização do Modelo Linear Generalizado Poisson, que ajusta dados estáticos e dinâmicos. A evolução temporal das estimativas dos parâmetros pode ser vista na Figura 6.5.

Tabela 6.1: Estimativa dos coeficientes do modelo Poisson estático

Modelo	COEFICIENTE ESTIMADO	ESTIMATIVA PONTUAL	ERRO PADRÃO
MLG Poisson dinâmico	INTERCEPTO	1.018137	0.04895716
	β_1	-0.1897312	0.06073841
	β_2	0.4672894	0.06186289
MLG Poisson estático	INTERCEPTO	1.01003	0.04920
	β_1	-0.19204	0.06087
	β_2	0.47616	0.06219

No próximo capítulo abordaremos os Filtros de Chopin (2007) e Caron et. al (2012) e os aplicaremos para a distribuição de Poisson.

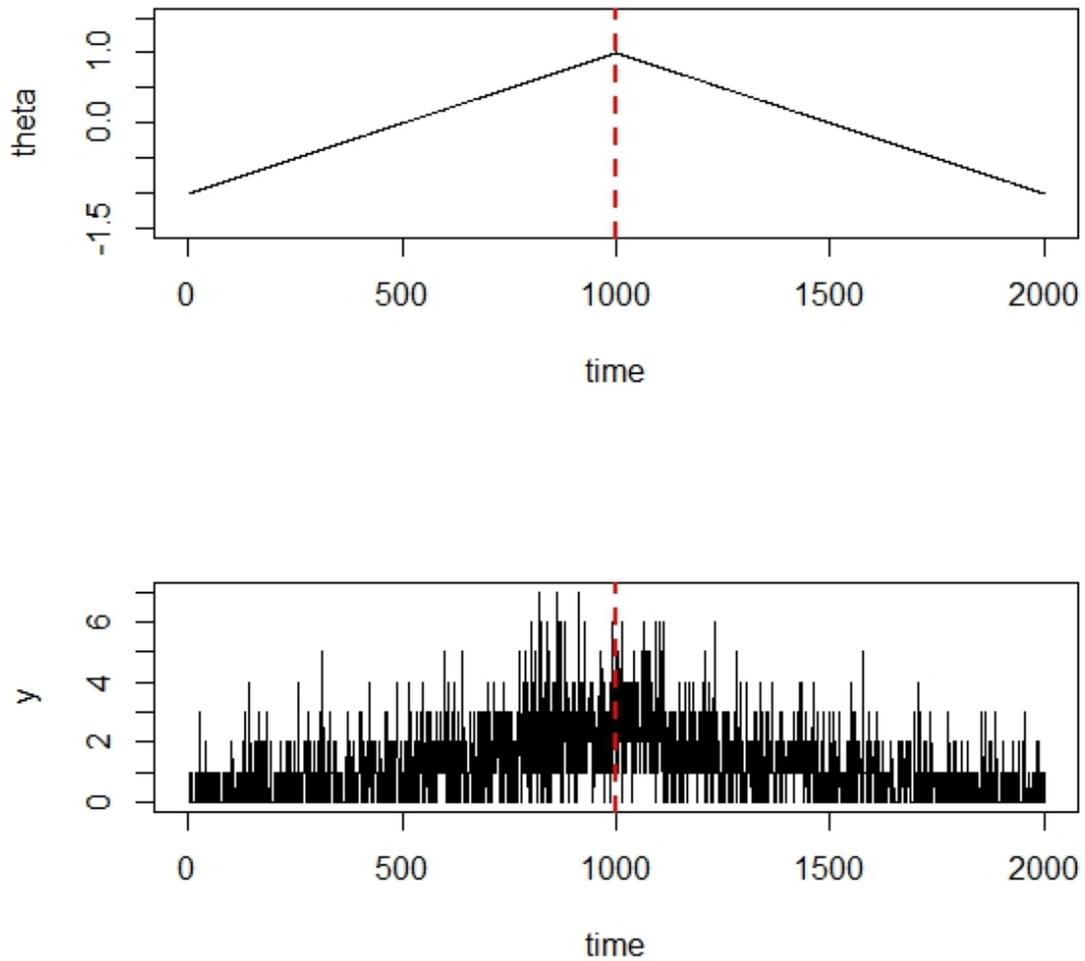


Figura 6.1: Dados simulados com mudança na inclinação de θ_t . A linha pontilhada indica onde houve a mudança estrutural da série y_t .

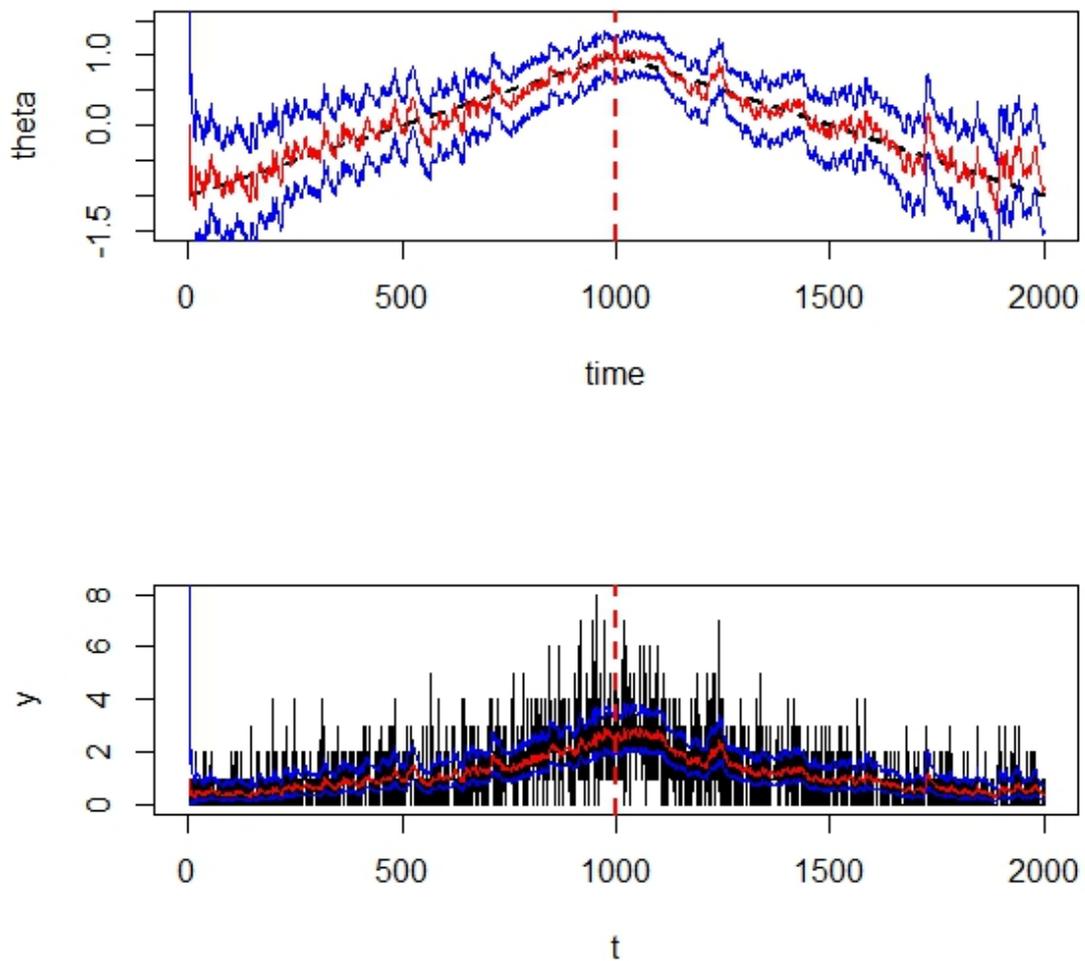


Figura 6.2: De cima para baixo tem-se: Série histórica y_t em preto, valor predito μ_t em vermelho, intervalo de credibilidade em linhas azuis. Valor real de θ_t em preto, estimativa $\hat{\theta}_t$ em vermelho e intervalo de credibilidade em linhas azuis.

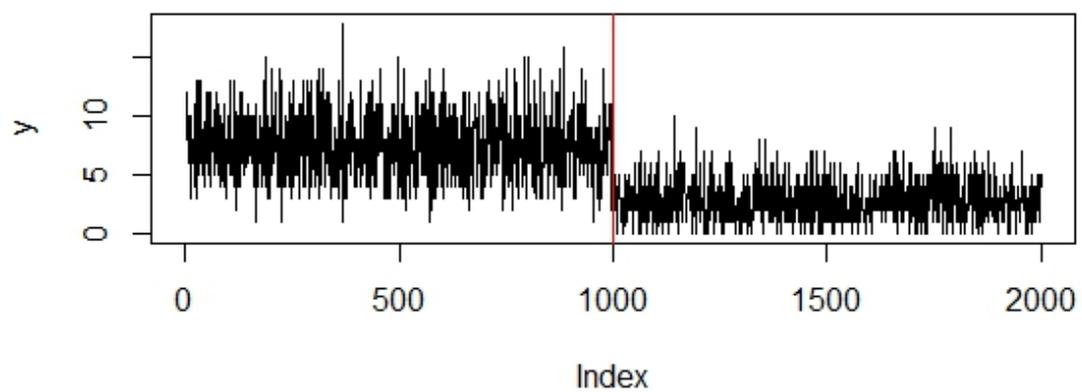
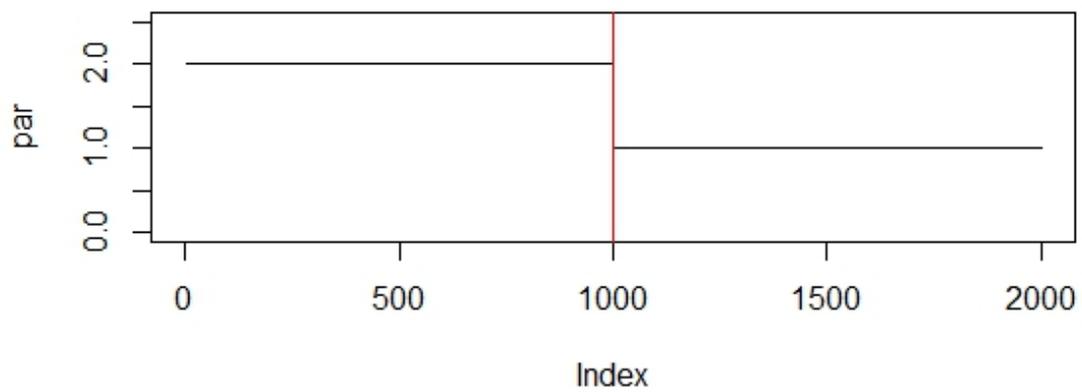


Figura 6.3: Dados simulados com um ponto abrupto em θ_t . A linha pontilhada indica onde houve a mudança estrutural da série y_t .

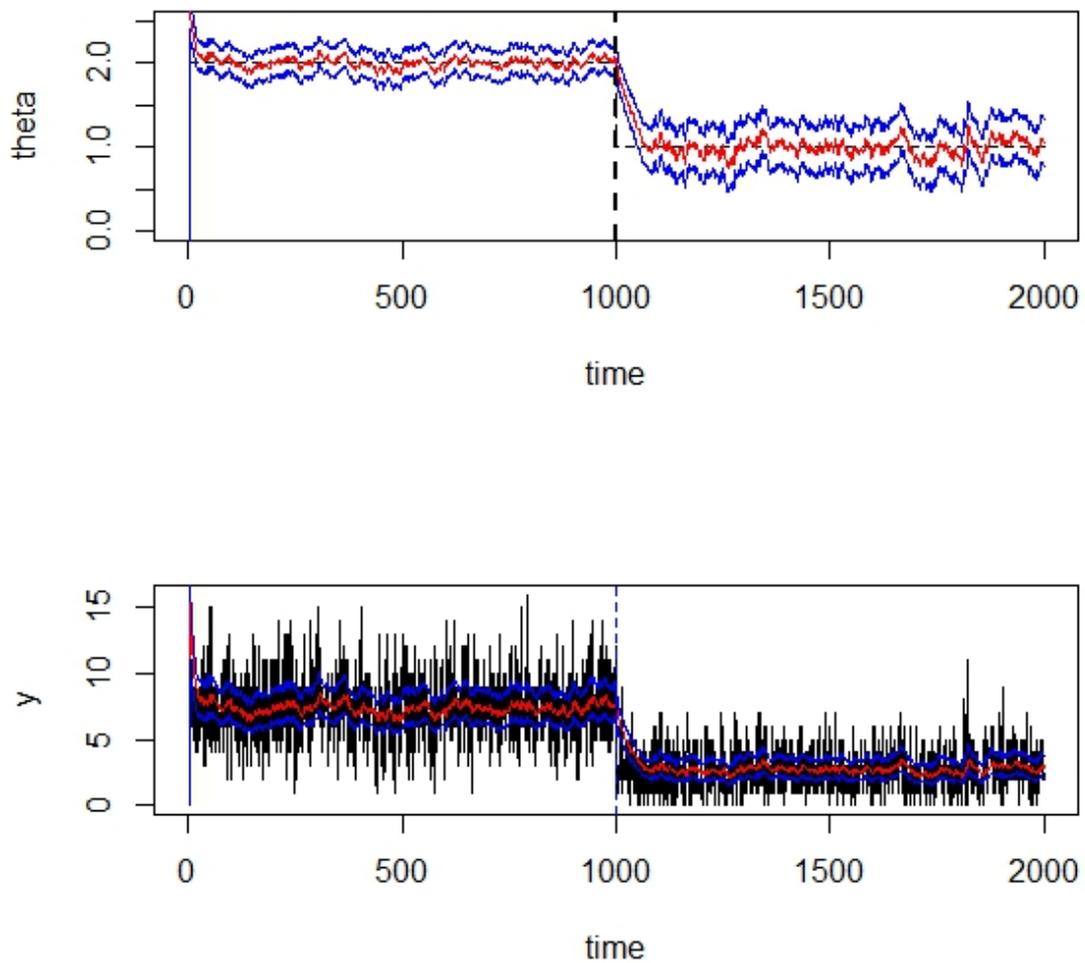
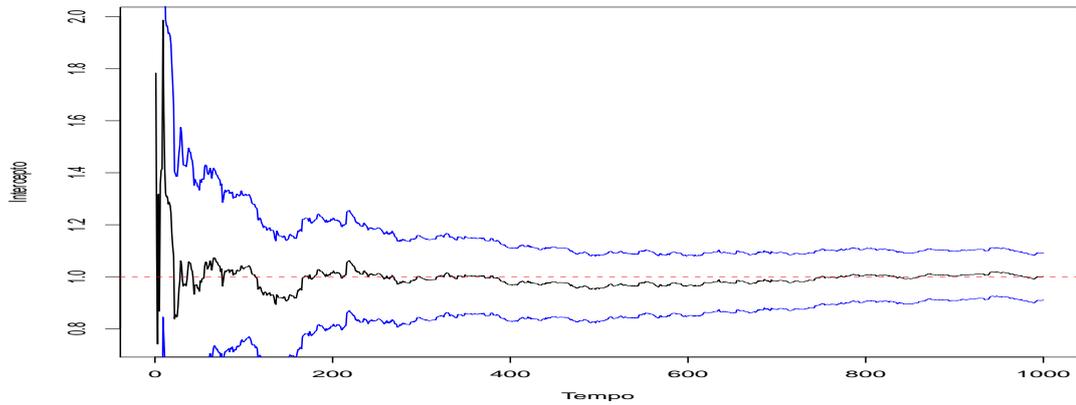
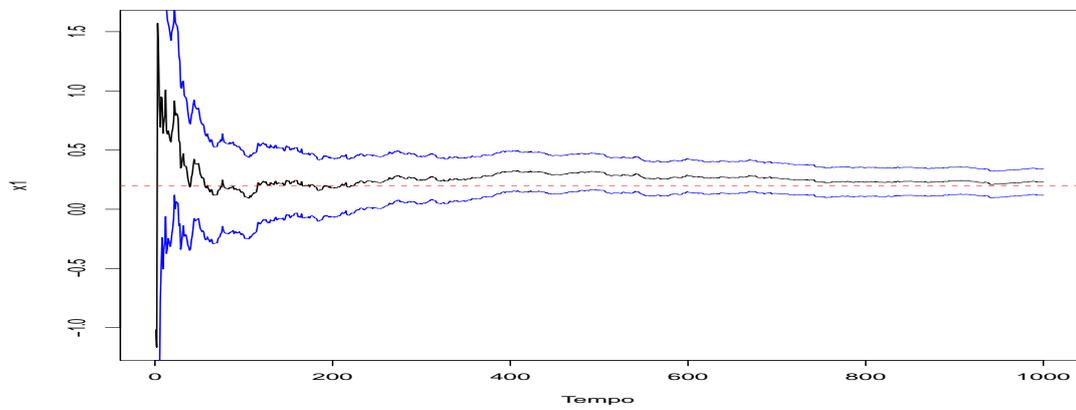


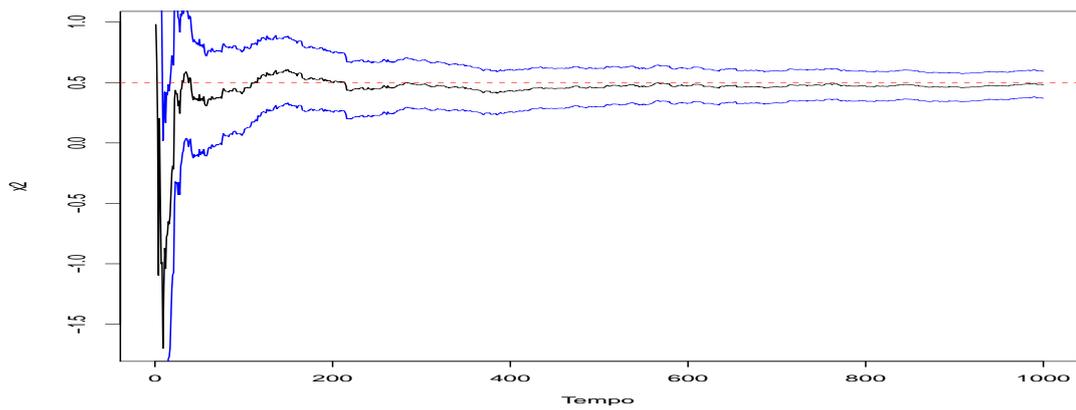
Figura 6.4: De cima para baixo tem-se: Série histórica y_t em preto, valor predito μ_t em vermelho, intervalo de credibilidade em linhas azuis. Valor real de θ_t em preto, estimativa $\hat{\theta}_t$ em vermelho e intervalo de credibilidade em linhas azuis.



(a) Intercepto



(b) β_1



(c) β_2

Figura 6.5: De cima para baixo tem-se: Evolução temporal das estimativas do intercepto, x_1 e x_2 . Valor real de θ_t em linhas pontilhadas vermelhas, estimativa $\hat{\theta}_t$ linhas pretas e intervalo de credibilidade em linhas azuis.

Capítulo 7

Modelos Dinâmicos para a Detecção de Pontos de Mudança via Filtro de Partículas

Recentemente, foram propostos modelos nos quais são utilizados filtro de partículas para detectar a quantidade de pontos de mudanças, e onde eles estão localizados, baseados no método de Barry e Hartigan (1992). Chopin (2007) desenvolveu um algoritmo de filtro de partículas para detectar mudanças em séries de longa duração, Fearnhead e Liu (2007) desenvolveram um filtro de partículas para a detecção de pontos de mudança baseados nas ideias de Chopin (2007) e o utilizaram para modelar dados de DNA. Caron et al. (2012) generalizaram o filtro proposto por Fearnhead e Liu (2007) para estimar parâmetros estáticos do modelo além de detectar os pontos de mudança.

Diferentemente da abordagem dada no capítulo anterior, aqui considera-se o problema de detectar pontos de mudanças (mudanças estruturais) sem o conhecimento *a priori* da quantidade, e os locais onde ocorreram as mudanças. Os algoritmos descritos neste capítulo são baseados em filtros de partículas e suavização de um modelo de espaço de estados não gaussianos.

Primeiramente será descrito o algoritmo proposto por Chopin (2007). Este algoritmo combina as principais ideias dos filtros de partículas e dos métodos MCMC¹.

¹Amostragem por rejeição e algoritmo de Metropolis-Hastings.

Em segundo lugar será descrito o algoritmo proposto por Caron et al. (2012). Tal algoritmo é uma extensão do filtro de partículas proposto por Fearnhead e Liu (2007), por permitir a estimação de parâmetros estáticos dos modelos.

Na terceira seção os algoritmos de Chopin (2007) e Caron et. al (2012) serão aplicados ao modelo Poisson por meio de dados simulados.

7.1 Algoritmo de Chopin (2007)

Chopin (2007), propôs um algoritmo híbrido que combina as principais ideias dos filtros de partículas e dos métodos MCMC para a detecção dinâmica dos pontos de mudanças, sem o conhecimento *a priori* dessas mudanças.

Considere um modelo de série temporal discreta indexada por um parâmetro de mudança θ_t , $t \geq 1$,

$$y_t \sim p(y_t | D_{t-1}, \theta_t), \quad (7.1)$$

onde D_{t-1} denota a subsequência y_1, \dots, y_{t-1} . Assume-se que o parâmetro de mudança, θ_t , segue um processo constante definido por partes, tal que

$$\theta_t = \epsilon_k, \text{ sob a condição de } \sum_{i=1}^{k-1} \delta_i \leq t \leq \sum_{i=1}^k \delta_i,$$

ou seja, para as δ_1 primeiras observações, o valor de parâmetro é ϵ_1 , para as δ_2 observações seguintes o valor é ϵ_2 e assim por diante. A descrição acima define a equação das observações do modelo dinâmico.

Os δ_i 's e os ϵ_k 's são desconhecidos, com densidades *a priori* $\pi_\delta(\cdot)$ e $\pi_\epsilon(\cdot)$ respectivamente. A densidade $\pi_\delta(\cdot)$ possui suporte nos inteiros positivos. Por simplicidade, as densidade são assumidas, *a priori*, independentes e identicamente distribuídas, mas essa suposição pode ser relaxada (Chopin, 2007).

Chopin (2007), propôs uma reformulação desse modelo genérico para um modelo de espaço de estados geral, ou seja um modelo de um processo observado (y_t), para o qual o comportamento é expresso condicionalmente a um processo de Markov oculto (x_t). Seja d_t o tempo de duração, no tempo t , desde a última mudança, ou seja $d_t = t - \delta_1 - \dots - \delta_{k-1}$, se o processo está no regime k .

Esquemáticamente, o processo descrito por Chopin (2007) pode ser ilustrado de acordo com a Figura 7.1

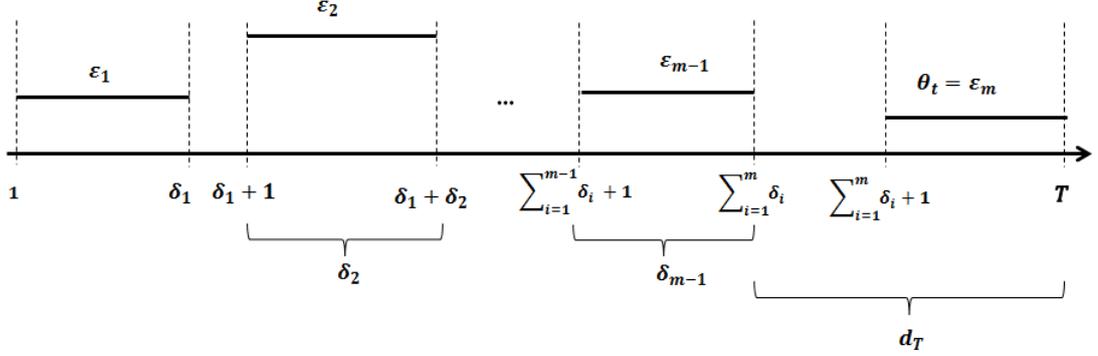


Figura 7.1: Representação esquemática do processo descrito por Chopin (2007).

Seja $x_t = (\theta_t, d_t)$, uma variável latente estendida tal que, a equação do sistema é condicional a $x_{t-1} = (\theta_{t-1}, d_{t-1})$, descrita por

$$x_t = (\theta_t, d_t) = \begin{cases} (\theta_{t-1}, d_{t-1} + 1) & \text{com probabilidade } \pi_\delta(\delta \geq d_{t-1} + 1 | \delta \geq d_{t-1}) \\ (\epsilon^*, 1) & \text{com probabilidade } \pi_\delta(\delta = d_{t-1} + 1 | \delta \geq d_{t-1}) \end{cases} \quad (7.2)$$

onde ϵ^* é amostrado, independentemente, da priori $\pi_\epsilon(\cdot)$.

Em linhas gerais, na etapa de filtragem, que é uma seqüência de derivação da densidade $p(x_t | D_{t-1})$, é feita uma estimação dinâmica do instante da última mudança e o valor dos parâmetros. Em contrapartida, a etapa de suavização permite a estimação conjunta de todos os pontos de mudança e dos parâmetros de cada regime para todo o conjunto de dado D_t .

Chopin (2007), descreveu várias propostas de algoritmos para o problema em questão. O primeiro filtro de partículas consiste na geração e atualização de partículas $x_t^{(j)}$, $j = 1, \dots, H$, através dos passos iterativos descritos a seguir:

Algoritmo 1

- **Passo 1:** Simule independentemente para $j = 1, \dots, H$

$$x_t^{(j)} \sim p(x_t | x_{t-1}^{(j)}),$$

onde $p(x_t|x_{t-1})$ descreve a densidade condicional da cadeia de Markov oculta $\{x_t\}$.

- **Passo 2:** Pondere as partículas, para $j = 1, \dots, H$, onde os pesos são dados por

$$w_t^{(j)} = p(y_t|D_{t-1}, x_t^{(j)}),$$

em que $p(y_t|D_{t-1}, x_t^{(j)})$ define a verossimilhança condicional do processo observado y_t .

- **Passo 3:** Reamostre as partículas, isto é, substitua o conjunto atual de partículas por um conjunto contendo $n_t^{(j)}$ réplicas de $x_t^{(j)}$, $j = 1, \dots, H$, onde $n_t^{(j)}$ é uma amostra que satisfaz

$$E(n_t^{(j)}) = \frac{Hw_t^{(j)}}{\sum_{j=1}^H w_t^{(j)}} \text{ e } \sum_{j=1}^H n_t^{(j)} = H$$

- **Passo 4:** Faça $t = t + 1$ e vá para o passo 1.

Segundo Chopin (2007), as transições Markovianas do modelo em questão não possuem boa mistura, de modo que outras estratégias foram desenvolvidas para melhorar o algoritmo inicial.

7.1.1 Algoritmo 2 - Rao-Blackwellização do componente discreto

Considere a simulação de $x_t^{(j)}$ condicional a $x_{t-1}^{(j)}$ no passo 1 do algoritmo 1. Dada a estrutura particular de $p(x_t|x_{t-1})$, isto envolve a simulação de um componente binário que designa se houve ou não ponto de mudança no tempo t .

Como a probabilidade desse evento pode ser calculada exatamente, esse componente binário pode ser "Rao-Blackwellizado", isto é, as contas podem ser feitas integrando com respeito a este (Marginalizando). Segundo Chopin (2007), a Rao-Blackwellização causa diminuição da variância assintótica das estimativas dos parâmetros do modelo.

Assuma que $x_{t-1}^{(j)} = (\epsilon, d)$ e crie duas partículas, cada uma correspondendo a uma das duas possibilidades, mesmo regime ou regime diferente, com pesos respectivos,

- $x_t^{(j,1)} = (\epsilon, d + 1)$, $w_t^{(j,1)} = \pi_\delta(\delta \geq d + 1 | \delta \geq d)p(y_t|D_{t-1}, \theta_t = \epsilon)$ - Mesmo regime.

- $x_t^{(j,2)} = (\epsilon^*, 1)$, $w_t^{(j,2)} = p(y_t | D_{t-1}, \theta_t = \epsilon^*)$ - Regime distinto.
- ϵ^* é gerado, independentemente, a partir de $\pi_\epsilon(\cdot)$.

Desse modo, obtém-se um conjunto com $2H$ partículas, que podem ser amostradas com respeito aos pesos $w_t^{(j,1)}$ e $w_t^{(j,2)}$, de modo a obter H partículas.

As probabilidades, $\pi_\delta(\delta \geq d+1 | \delta \geq d)$ e $\pi_\delta(\delta = d | \delta \geq d)$, não necessitam estar disponíveis em forma fechada. No entanto, note que

$$\begin{aligned}
\pi_\delta(\delta = d | \delta \geq d) &= 1 - \pi_\delta(\delta \geq d+1 | \delta \geq d) = \frac{P([\delta = d] \cap [\delta \geq d])}{P(\delta \geq d)} \\
&= \frac{P(\delta = d)}{P(\delta \geq d)} = \frac{P(\delta = d)}{1 - P(\delta \leq d)} \\
&= \frac{\pi_\delta(d)}{1 - \sum_{k=1}^{d-1} \pi_\delta(k)}. \tag{7.3}
\end{aligned}$$

Pode-se salvar as somas parciais $\sum_{k=1}^{d-1} \pi_\delta(k)$ e reutilizá-las quando necessário.

Problema: O filtro Rao-Blackwellizado continua a ser altamente ineficiente, devido à falta de boas condições de *mixing*² do processo latente x_t .

Considerando o problema em questão, ocasionado pelos parâmetros constantes incluídos no processo Markoviano oculto, Gilks e Berzuini (2001) propuseram a criação de um "efeito de rejuvenescimento" artificial com o movimento das partículas feito através de um processo de MCMC. Tal processo de MCMC precisa ser invariante à densidade alvo.

O leitor interessado no método de Rao-Blackwellização pode consultar Casella e Robert (1996).

7.1.2 Nova proposta - Movimento fracional

Segundo Chopin (2007), o método de Rao-Blackwellização continua altamente ineficiente, com degeneração devida, principalmente, à presença de parâmetros constantes nos estados do modelo dinâmico, isto é, nos θ_t 's, pois

$$\theta_t = \epsilon_k, \quad t \in \left(\sum_{i=1}^{k-1} \delta_i, \sum_{i=1}^k \delta_i \right].$$

²Devido a baixa variabilidade a integração não consegue varrer todo o espaço paramétrico.

Chopin (2007), propõe que sejam feitos movimentos (passos de rejuvenescimento, para aumentar a variância) somente nos componentes θ_t de cada partícula $x_t^{(j)} = (\theta_t^{(j)}, d_t^{(j)})$, através de um kernel MCMC com distribuição invariante dada por

$$P(\theta_t = \epsilon | d_t = d_t^{(j)}, D_t) \propto \pi_\epsilon(\epsilon) \prod_{b=(t-d_t^{(j)})+1}^t P(y_b | D_{b-1}, \theta_b = \epsilon). \quad (7.4)$$

Isso implica em implementar um passo MCMC com respeito ao modelo correspondente ao período atual, desde a última mudança. Isto é conceitualmente mais simples e computacionalmente mais barato do que considerar o modelo completo correspondente às observações até o tempo t .

Na expressão (7.4) considerou-se, apenas, as observações correspondentes aos tempos $b = (t - d_t^{(j)}) + 1$ até t , isto é, correspondente ao tempo atual, desde a última mudança.

Sumarizando:

- A técnica de movimento fracional destina-se a 'oxigenar' o filtro de partículas de modo a atenuar o problema da degeneração do processo. Dessa forma, introduz-se variabilidade extra através de movimentos (perturbações) introduzidas via MCMC, na parte estática do processo. O processo degenerativo ocorre, principalmente, devido à parte estática dos parâmetros desconhecidos.
- O kernel do MCMC é dado por

$$\eta_t^{(j)} = P(\theta_t = \epsilon | d_t = d_t^{(j)}, D_t) \propto \pi_\epsilon(\epsilon) \prod_{b=(t-d_t^{(j)})+1}^t P(y_b | D_{b-1}, \theta_b = \epsilon). \quad (7.5)$$

- Na expressão acima considera-se no kernel proposto apenas a influência das observações correspondentes ao tempo t atual, desde a última mudança.

Sendo assim, o filtro de partículas Rao-Blackwellizado com movimento fracionado pode ser resumido através dos seguintes passos (Chopin, 2007):

Algoritmo 3:

1. **Passo 1:** Simule, independentemente, para $j = 1, \dots, H$, $\epsilon^{(j)} \sim \pi_\epsilon(\cdot)$, e condicional a $x_{t-1}^{(j)} = (\theta_{t-1}^{(j)}, d_{t-1}^{(j)})$, crie duas partículas, cada uma correspondendo as duas possibilidades, de ter havido ou não mudança no tempo t :

$$\begin{aligned} x_t^{(j,1)} &= (\theta_{t-1}^{(j)}, d_{t-1}^{(j)} + 1), \text{ quando não houve mudança;} \\ x_t^{(j,2)} &= (\epsilon^{(j)}, 1), \text{ quando houve mudança.} \end{aligned}$$

2. **Passo 2:** Repondere as partículas com pesos respectivos

$$\begin{aligned} w_t^{(j,1)} &= \pi_\delta(\delta \geq d_{t-1}^{(j)} + 1 | \delta \geq d_{t-1}^{(j)}) p(y_t | \theta_t = \theta_{t-1}^{(j)}, D_{t-1}), \\ w_t^{(j,2)} &= \pi_\delta(\delta = d_{t-1}^{(j)} | \delta \geq d_{t-1}^{(j)}) p(y_t | \theta_t = \epsilon^{(j)}, D_{t-1}). \end{aligned}$$

3. **Passo 3:** Reamostrare as $2H$ partículas com pesos respectivos $w_t^{(j,1)}$, $w_t^{(j,2)}$, de forma a obter H partículas reamostradas.
4. **Passo 4:** Para cada partícula selecionada $x_t^{(j)} = (\theta_t^{(j)}, d_t^{(j)})$ substitua $\theta_t^{(j)}$ por

$$\tilde{\theta}_t^{(j)} \sim k_t^{(j)}(\theta_t^{(j)}, \cdot),$$

onde $k_t^{(j)}$ é um kernel MCMC com distribuição invariante $\eta_t^{(j)}$ como em (7.5).

5. **Passo 5:** Faça $t = t + 1$ e retorne ao passo 1.

A atualização dos parâmetros no passo 4 do algoritmo dado acima se dá por meio de um passo utilizado o algoritmo Metropolis-Hastings Gaussiano, ou seja considere-se a densidade candidata $\epsilon | \theta_t^{(j)} \sim q(\epsilon | \theta_t^{(j)}) = N(\theta_t^{(j)}, \hat{\Sigma}_t)$ e simula-se um passo MCMC da seguinte forma

$$\tilde{\theta}_t^{(j)} = \begin{cases} \epsilon & \text{com probabilidade } \min\left(1, \frac{q(\theta_t^{(j)} | \epsilon) \eta_t^{(j)}(\epsilon)}{q(\epsilon | \theta_t^{(j)}) \eta_t^{(j)}(\theta_t^{(j)})}\right), \\ 0 & \text{caso contrário.} \end{cases} \quad (7.6)$$

A matriz de covariâncias é dada por

$$\hat{\Sigma}_t = \gamma^2 \left[\frac{1}{H} \sum_{j=1}^H \theta_t^{(j)} (\theta_t^{(j)})^T - \left(\frac{1}{H} \sum_{j=1}^H \theta_t^{(j)} \right) \left(\frac{1}{H} \sum_{j=1}^H \theta_t^{(j)} \right)^T \right],$$

onde γ é um parâmetro de ajuste. Segundo Chopin (2007), valores de γ entre 0.5 e 1 garantem uma melhor performance para o filtro. Para maiores detalhes do filtro, bem como outras informações, o leitor interessado pode consultar Chopin (2007).

O procedimento via filtro de partículas para a detecção de pontos de mudança proposto por Chopin (2007), não incorpora a possibilidade de haver parâmetros estáticos. Tal como discutido no Capítulo 6, os modelos de espaço de estados que incorporam parâmetros estáticos demandam sofisticadas metodologias para a estimação eficiente.

Nesta dissertação, propomos uma extensão ao filtro de Chopin (2007) para incorporar a estimação de parâmetros estáticos. Tal modelo será visto em detalhes na Seção 8.2 do Capítulo 8.

Na próxima seção descreveremos, em detalhes, a técnica proposta por Caron et al. (2012).

7.2 Algoritmo de Caron et al. (2012)

7.2.1 Modelo Estatístico

De acordo com Fearnhead e Liu (2007), considere uma série temporal observada z_t , $t = 1, \dots, T$, em que, para cada período de tempo de i a j , para $i < j$, tem-se $z_{i:j} = (z_i, \dots, z_j)$.

Suponha que a série $\{z_t\}$ seja sujeita a heterogeneidade temporal de modo que mudanças abruptas ocorram em determinados pontos no tempo. Uma abordagem bastante utilizada em tais contextos, consiste em segmentar a sequência de observações z_1, \dots, z_T de acordo com uma sequência de localizações dos *pontos de mudança* $0 < \tau_1 < \tau_2 < \dots < \tau_m < T$, tais que as observações sejam homogêneas dentro de cada segmento e heterogênea entre os segmentos.

Para os m pontos de mudança (PM) há $m + 1$ segmentos contíguos:

$$z_{1:\tau_1}, z_{\tau_1+1:\tau_2}, z_{\tau_2+1:\tau_3}, \dots, z_{\tau_{m+1}:T}.$$

Modelo Específico dos Pontos de Mudança

De acordo com Fearnhead e Liu (2007), considere um modelo de ponto de mudança

em que as posições no tempo em que as mudanças ocorrem são modeladas de acordo com um processo Markoviano de modo que

$$\begin{aligned} Pr(\text{O próximo ponto de mudança ocorre em } t_2 | \text{ A última mudança ocorreu em } t_1) = \\ = Pr(\text{PM em } t_2 | \text{ PM em } t_1) = h(t_2 - t_1). \end{aligned} \quad (7.7)$$

Dessa forma, a probabilidade da ocorrência de um ponto de mudança depende, apenas, do "índice de distância" do ponto de mudança anterior. Esse modelo é um caso especial do modelo de partição produto para pontos de mudança (Barry e Hartigan, 1992).

Em (7.7), considere que $h(\cdot)$ descreve alguma distribuição de probabilidade com suporte nos inteiros positivos. Adicionalmente, denote por $H(\cdot)$ a função de distribuição acumulada (f.d.a) associada a $h(\cdot)$, i.e,

$$H(l) = \sum_{i=1}^l h(i). \quad (7.8)$$

A f.d.a. $H(\cdot)$ será utilizada na descrição do processo de estimação.

Pressupostos do Modelo

Um pressuposto básico na formulação do modelo é o da independência condicional entre os pontos de mudança (Barry e Hartigan, 1992; Fearnhead e Liu, 2007): *dada a posição de um ponto de mudança, a data anterior àquele ponto de mudança é independente da data posterior ao ponto de mudança.*

Multimodelos

Para cada segmento de observações $z_{\tau_i+1:\tau_{i+1}}$, $i = 0, 1, \dots, m$, assume-se que existem M possíveis modelos.

Distribuição a *priori* do modelo

Para cada modelo

$$r \in \{1, 2, \dots, M\},$$

que ocorre com probabilidade a *priori* $p(r)$, está associado um conjunto Ψ_r de parâmetros desconhecidos, com distribuição a *priori* π_r .

A Figura 7.2 sumariza a hierarquia elaborada para modelar o segmento $z_{\tau_i+1:\tau_{i+1}}$.

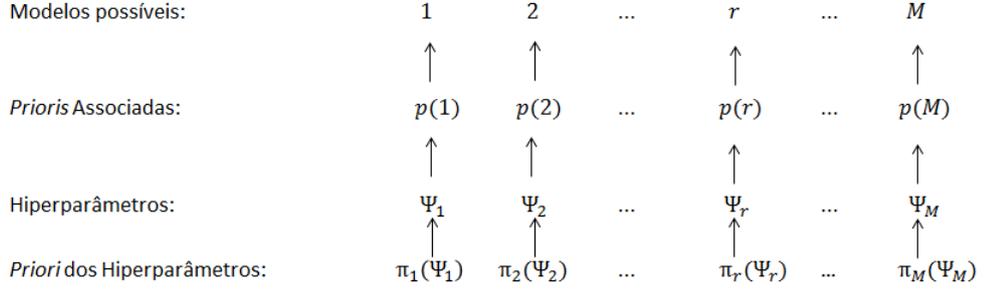


Figura 7.2: Hierarquia de modelagem do segmento segundo Caron et al. (2012).

Distribuição de probabilidade de um segmento genérico $z_{i:j}$

Tome $i \leq j$. Condicionalmente a um dado modelo $r \in \{1, 2, \dots, M\}$, a distribuição de probabilidade associada ao segmento $z_{i:j}$, é dada por

$$P(Z_{i:j}|r) := P(i, j|r) = \int p(z_{i:j}, \Psi_r|r) d\Psi_r = \int p(z_{i:j}|r, \Psi_r) \pi_r(\Psi_r) d\Psi_r. \quad (7.9)$$

A distribuição marginal associada ao segmento $z_{i:j}$ é dada por

$$P(i, j) = \sum_{r=1}^M P(i, j|r) p(r). \quad (7.10)$$

Tal probabilidade pode ser calculada analiticamente para casos bem específicos como, por exemplo, para *prioris* conjugadas. No entanto, tipicamente, tal integral é analiticamente intratável e métodos numéricos, como quadratura gaussiana, ou aproximações, como a de Laplace, devem ser utilizadas. Um exemplo em que essa integral é analiticamente tratável é quando considera-se o modelo em que $y_t \sim \text{Poisson}(\lambda_t)$ e $\lambda_t \sim \text{Gamma}(a, b)$. Sendo assim, tem-se que, para $r = 1$ e $\Psi_1 = (a, b)$, obtem-se

$$P(i, j|r) = \frac{b^a}{\Gamma(a) \left[\prod_{k=i}^j y_k! \right]} \frac{\Gamma(s^* + a - 1)}{(n^* + b)^{s^* + a - 1}}, \quad (7.11)$$

onde $s^* = \sum_{k=i}^j y_k$ e $n^* = j - i + 1$.

Detecção dos pontos de mudança

Seja o evento,

$$C_t = \text{Tempo do ponto de mudança mais recente anterior no tempo } t, \quad (7.12)$$

que descreve o estado latente no tempo t .

Condicionalmente a $C_{t-1} = i$, tem-se que as probabilidades de transição entre os estados são descritas por

$$f(C_t = j | C_{t-1} = i) = \begin{cases} p_1 & \text{se } j = i \text{ (Não houve ponto de mudança no tempo } t) \\ p_2 & \text{se } j = t - 1 \text{ (A mudança mais recente ocorreu em } t - 1) \\ 0 & \text{Não houve PM algum anterior a } t. \end{cases} \quad (7.13)$$

Fearnhead e Liu (2007) descrevem (7.13) como a seguir

$$f(C_t = j | C_{t-1} = i) = \begin{cases} \frac{1-H(t-i-1)}{1-H(t-i-2)} & \text{se } j = i \\ \frac{H(t-i-1)-H(t-i-2)}{1-H(t-i-2)} & \text{se } j = t - 1 \\ 0 & \text{caso contrário} \end{cases} \quad (7.14)$$

A prova da expressão (7.14) é apresentada nos apêndices. Tal expressão descreve, de fato, a equação do sistema de um modelo dinâmico.

7.2.2 Inferência Exata

Podemos reescrever esse modelo de ponto de mudança como um modelo de espaço de estados seguindo as idéias de Chopin (2007) e Fearnhead e Liu (2007).

Formulação do problema via Modelos de Espaço de Estados

Como já mencionado, C_t representa o estado latente no tempo t . A equação das observações associada ao modelo dinâmico para pontos de mudança é descrita por

$$g(z_t | C_t = j, D_{t-1}) = \frac{P(z_{1:t} | C_t = j)}{P(z_{1:t-1} | C_t = j)},$$

mas, utilizando-se a propriedade da independência condicional, mencionada nos pressupostos do modelo,

$$g(z_t|C_t = j, D_{t-1}) = \begin{cases} \frac{P(z_{j:t}|C_t=j)}{P(z_{j:t-1}|C_t=j)} = \frac{P(j,t)}{P(j,t-1)} & \text{se } j < t - 1 \\ \frac{P(z_{t-1:t}|C_t=j)}{P(z_{t-1:t-1}|C_t=j)} = \frac{P(t-1,t)}{P(t-1,t-1)} & \text{se } j = t - 1 \end{cases}, \quad (7.15)$$

em que $P(i, j)$ é dada pela equação (7.10).

Em resumo, o modelo de espaço de estados em questão é dado por

- **Equação das observações:**

$$g(z_t|C_t = j, D_{t-1}) = \begin{cases} \frac{P(j,t)}{P(j,t-1)} & \text{se } j < t - 1 \\ P(t - 1, t) & \text{se } j = t - 1 \end{cases}, \quad (7.16)$$

- **Equação do Sistema:**

$$f(C_t = j|C_{t-1} = i) = \begin{cases} \frac{1-H(t-i-1)}{1-H(t-i-2)} & \text{se } j = i \\ \frac{H(t-i-1)-H(t-i-2)}{1-H(t-i-2)} & \text{se } j = t - 1 \\ 0 & \text{caso contrário.} \end{cases} \quad (7.17)$$

A expressão $g(z_t|C_t = j, D_{t-1})$ também descreve o que Caron et al. (2012) designam como *densidade preditiva condicional*.

Filtragem - obtenção da distribuição $P(C_t|D_t)$

A distribuição a *posteriori* (Filtrada) de $(C_t|D_t)$, i.e, pode ser obtida recursivamente. Seja

$$P(C_t|D_t) = \frac{P(C_t, D_t)}{\sum_{i=0}^{t-1} P(C_t = i, D_t)} = \frac{\xi(C_t, D_t)}{\sum_{i=0}^{t-1} \xi(C_t = i, D_t)},$$

em que

$$\xi(C_t, D_t) = g(z_t|C_t, D_{t-1})P(C_t|D_{t-1}), \quad (7.18)$$

$$\begin{aligned} P(C_t = i|D_{t-1}) &= \sum_{j=0}^{t-2} P(C_t = i, C_{t-1} = j|D_{t-1}) \\ &= \sum_{j=0}^{t-2} P(C_t = i|C_{t-1} = j)P(C_{t-1} = j|D_{t-1}) \end{aligned} \quad (7.19)$$

- No caso de $i < t - 1$, i.e, $i \leq t - 2$, então o tempo da última mudança, anterior ao tempo $t - 1$, não pode ter ocorrido para tempos inferiores a i e, muito menos, para tempos superiores a i . Logo,

$$P(C_t = j, C_{t-1} = i|D_t) = \begin{cases} 0, & \text{se } j < i \\ P(C_t = j, C_{t-1} = j|D_{t-1}), & \text{se } j = i \end{cases}$$

Dessa forma, tem-se que (7.18) é descrita por

$$\xi(C_t = j, D_t) = g(z_t|C_t = j, D_{t-1})P(C_t = j|C_{t-1} = j)P(C_{t-1}=j|D_{t-1}). \quad (7.20)$$

- No caso de $i = t - 1$, então o tempo da última mudança, anterior ao tempo $t - 1$, pode ter ocorrido em qualquer tempo $j \leq t - 2$. Logo, a expressão (7.19) é descrita por

$$P(C_t = j|D_t) = \begin{cases} \sum_{i=1}^{t-2} P(C_t = j|C_{t-1} = i)P(C_{t-1} = i|D_{t-1}), & \text{se } j < t - 1 \\ P(C_t = j|C_{t-1} = i)P(C_{t-1} = i|D_{t-1}), & \text{se } j = t - 1 \end{cases}$$

Dessa forma, a expressão (7.18) é dada por

$$\xi(C_t = j, D_t) = \begin{cases} g(z_t|C_t = j, D_{t-1})f(C_t = j|C_{t-1} = j)p(C_t = j|D_{t-1}) & \text{se } j < t - 1 \\ g(z_t|C_t = j, D_{t-1}) \sum_{i=0}^{t-2} f(C_t = j|C_{t-1} = i)P(C_t = i|D_{t-1}) & \text{se } j = t - 1 \end{cases} \quad (7.21)$$

Como as distribuições filtradas $P(C_t|D_t)$ são armazenadas para todo $t = 1, \dots, T$, então pode-se simular a partir da distribuição conjunta a *posteriori* dos parâmetros no tempo T usando a ideia de Chopin (2007)

Simulação dos pontos de mudança a partir da distribuição conjunta a posteriori:

1. Simule τ_1 de $p(C_T|D_T)$. Faça $k = 1$.
2. Enquanto $\tau_k > 0$, amostre τ_{k+1} e faça $k + 1$.

7.2.3 Estimativa recursiva dos parâmetros

Usualmente as probabilidades de transição $f(C_t|C_{t-1})$ e a densidade preditiva condicional $g(z_t|C_t, D_t)$ dependem de alguns parâmetros $\theta \in \mathfrak{R}^{n_\theta}$ que necessitam ser estimados a partir dos dados. Caron et al. (2012) propuseram a estimação por máxima verossimilhança recursiva de θ . O subscrito θ será utilizado para enfatizar a dependência sob os parâmetros da densidade de filtragem $p_\theta(C_t|D_t)$, da probabilidade de transição $f_\theta(C_t|C_{t-1})$ e da densidade preditiva condicional $g_\theta(z_t|C_t = j, D_t)$. A log verossimilhança dos dados $Z_{1:t}$ é descrita por

$$l_t(\theta) = \log(p_\theta(z_1)) + \sum_{k=2}^t \log [p_\theta(z_k|D_{k-1})],$$

onde, de acordo com a expressão (7.18),

$$p_\theta(z_t|D_{t-1}) = \sum_{j=0}^{t-1} \xi_\theta(C_t = j, D_t).$$

Quando $t \rightarrow \infty$ então $\lim_{t \rightarrow \infty} \frac{l_t(\theta)}{t} = l(\theta)$. A estratégia consiste em encontrar o máximo local de $l(\theta)$. Sendo assim, utiliza-se um algoritmo de aproximação estocástica (Benveniste et al. 1990)

$$\theta_t = \theta_{t-1} + \gamma_t \nabla \log p_{\theta_{1:t-1}}(z_t|D_{t-1}), \quad (7.22)$$

onde a sequência $\{\gamma_t\}$ é não crescente positiva tal que $\sum_{t=1}^{\infty} \gamma_t = \infty$, $\sum_{t=1}^{\infty} \gamma_t^2 < \infty$ e $\nabla \log p_{\theta_{1:t-1}}(z_t|D_{t-1})$ é o gradiente da log verossimilhança preditiva calculada em θ_t . O subscrito $\theta_{1:t-1}$ indica que esse gradiente é calculado utilizando as recursões atualizadas com $\theta = \theta_k$ no tempo $k + 1$. Sob condições de regularidades (Benveniste et al 1990) pode-se mostrar que θ_t converge para o máximo local de $l(\theta)$. Para

acelerar a taxa de convergência do algoritmo, pode-se usar o método de Newton ou quasi-Newton³. Para calcular o gradiente em (7.22), note que

$$\nabla \log p_\theta(z_t | D_{t-1}) = \frac{\nabla p_\theta(z_t | D_{t-1})}{p_\theta(z_t | D_{t-1})} = \frac{\sum_{j=0}^{t-1} \nabla \xi_\theta(C_t = j, D_t)}{\sum_{j=0}^{t-1} \xi_\theta(C_t = j, D_t)}. \quad (7.23)$$

Calculando as derivadas de $p_\theta(C_t | D_t)$ com respeito a θ , obtém-se

$$\nabla p_\theta(z_t | D_{t-1}) = \frac{\nabla \xi_\theta(C_t, D_t)}{\sum_{j=0}^{t-1} \xi_\theta(C_t, D_t)} - p_\theta(z_t | D_{t-1}) \frac{\sum_{j=0}^{t-1} \nabla \xi_\theta(C_t, D_t)}{\sum_{j=0}^{t-1} \xi_\theta(C_t, D_t)} \quad (7.24)$$

O termo $\nabla \xi_\theta(C_t, D_t)$ é obtido tomando-se as derivadas de (7.21)

$$\nabla \xi_\theta(C_t = j, D_t) = \begin{cases} g_\theta(z_t | C_{t-1} = j, D_{t-1}) f_\theta(C_t = j | C_{t-1} = j) p_\theta(C_t = j | D_{t-1}) \pi_t^{(j,j)} & \text{se } j < t-1 \\ g_\theta(z_t | C_t = j, D_{t-1}) \sum_{i=0}^{t-2} f_\theta(C_t = j | C_{t-1} = i) p_\theta(C_{t-1} = i | D_{t-1}) \pi_t^{(i,j)} & \text{se } j = t-1 \end{cases} \quad (7.25)$$

onde

$$\begin{aligned} \pi_t^{(i,j)} &:= \nabla \log g_\theta(z_t | C_t = j, D_{t-1}) + \nabla \log f_\theta(C_t = j | C_{t-1} = i) + \\ &+ \nabla \log p_\theta(C_{t-1} = i | D_{t-1}). \end{aligned} \quad (7.26)$$

7.2.4 Inferência Aproximada

O custo computacional para calcular as recursões de $p_\theta(C_t | D_t)$ e $\nabla \log p_\theta(z_t | D_{t-1})$ em cada tempo t é proporcional a t . Logo, à medida que a base se torna grande, existe um grande esforço computacional que torna o procedimento exato proibitivo. Sendo assim, Fearnhead e Liu (2007) e Caron et al. (2012) propuseram um esquema de aproximação numérica para aproximar a densidade filtrada.

Em cada tempo t o algoritmo exato armazena o conjunto de probabilidades $p_\theta(C_t = j | D_t)$ para $j = 0, 1, \dots, t-1$. Dado que muitas dessas probabilidades são negligenciáveis, é razoável aproximar a distribuição de filtragem por um conjunto pequeno N_t de pontos do suporte da distribuição $c_t^{(1)}, \dots, c_t^{(N_t)}$, chamado de partículas, com probabilidade associada $w_t^{(1)}, \dots, w_t^{(N_t)}$, chamada de pesos.

³Vide Poyiadjis et al. (2011) para uma aplicação dessa abordagem para o caso de modelos de espaço de estados gerais.

Para limitar o número de partículas N_t no tempo t , Caron et al. (2012) adotaram um esquema simples, onde todas as partículas às quais os pesos estão abaixo de um limiar ϵ são descartadas. Já Fearnhead e Liu (2007) utilizaram reamostragem estratificada ótima proposta por Carpenter et al. (1999). No tempo $t - 1$, suponha que $\xi_\theta(C_{t-1}, D_{t-1})$ e $p(C_{t-1}|D_{t-1})$ são aproximados como

$$\begin{aligned}\hat{\xi}_\theta(C_{t-1}, D_{t-1}) &= \sum_{i=1}^{N_{t-1}} \tilde{w}_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1}) \\ \hat{p}_\theta(C_{t-1}|D_{t-1}) &= \sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1}),\end{aligned}$$

onde $\delta_{c_{t-1}^{(i)}}(C_{t-1}) = 1$ se $C_{t-1} = c_{t-1}^{(i)}$ e 0 caso contrário, ou seja, $\tilde{w}_t^{(i)}$ é uma aproximação de $\xi_\theta(C_{t-1} = c_{t-1}^{(i)}, D_{t-1})$, enquanto $w_t^{(i)}$ é uma aproximação de $p_\theta(C_{t-1} = c_{t-1}^{(i)}|D_{t-1})$, com $w_t^{(i)} \propto \tilde{w}_t^{(i)}$ e $\sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} = 1$. Caron et al. (2012) propuseram a seguinte aproximação para $\nabla p_\theta(C_{t-1}|D_{t-1})$:

$$\hat{\nabla} p_\theta(C_{t-1}|D_{t-1}) = \sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \beta_{t-1}^{(i)} \delta_{c_{t-1}^{(i)}}(C_{t-1}), \quad (7.27)$$

onde $\sum_{i=1}^{N_{t-1}} w_{t-1}^{(i)} \beta_{t-1}^{(i)} = 0$. Aqui $w_{t-1}^{(i)} \beta_{t-1}^{(i)}$ é uma aproximação de $\nabla p_\theta(C_{t-1} = c_{t-1}^{(i)}|D_{t-1})$. Sendo assim, $\beta_{t-1}^{(i)}$ pode ser visto como uma aproximação de $\nabla \log p_\theta(C_{t-1} = c_{t-1}^{(i)}|D_{t-1})$.

No tempo t seja $\tilde{c}_t^{(i)} = c_{t-1}^{(i)}$ e $\tilde{c}_t^{(N_{t-1}+1)} = t - 1$ para cada partícula $i = 1, \dots, N_{t-1}$. Para calcular uma aproximação de $p_\theta(C_{t-1}|D_{t-1})$ substitui-se a aproximação (7.27) em (7.21) para se obter pesos não normalizados para $i = 1, \dots, N_{t-1}$

$$\tilde{w}_t^{(i)} = g_\theta(z_t|C_t = \tilde{c}_t^{(i)}, D_{t-1}) f_\theta(C_t = \tilde{c}_t^{(i)}|C_{t-1} = \tilde{c}_t^{(i)}) w_{t-1}^{(i)}, \quad (7.28)$$

e

$$\tilde{w}_t^{(N_{t-1}+1)} = g_\theta(z_t|C_t = \tilde{c}_t^{(i)}, D_{t-1}) \sum_{i=1}^{N_{t-1}} f_\theta(C_t = t - 1|C_{t-1} = \tilde{c}_t^{(i)}) w_{t-1}^{(i)}. \quad (7.29)$$

Similarmente, substituindo-se (7.27) em (7.25) obtem-se uma aproximação $\tilde{\alpha}_t^{(i)}$ de $\nabla \xi_\theta(C_t = \tilde{c}_t^{(i)}, D_t)$, para $i = 1, \dots, N_{t-1}$, dada por

$$\begin{aligned} \tilde{\alpha}_t^{(i)} &= g_\theta(z_t | C_t = \tilde{c}_t^{(i)}, D_{t-1}) f_\theta(C_t = \tilde{c}_t^{(i)} | C_{t-1} = \tilde{c}_t^{(i)}) w_{t-1}^{(i)} \times \\ &\times \left[\nabla \log g_\theta(z_t | C_t = \tilde{c}_t^{(i)}, D_{t-1}) + \nabla \log f_\theta(C_t = \tilde{c}_t^{(i)} | C_{t-1} = \tilde{c}_t^{(i)}) + \beta_{t-1}^{(i)} \right] \end{aligned} \quad (7.30)$$

e

$$\begin{aligned} \tilde{\alpha}_t^{(N_{t-1}+1)} &= g_\theta(z_t | C_t = \tilde{c}_t^{(i)}, D_{t-1}) \sum_{i=1}^{N_{t-1}} f_\theta(C_t = t-1 | C_{t-1} = \tilde{c}_t^{(i)}) w_{t-1}^{(i)} \times \\ &\times \left[\nabla \log g_\theta(z_t | C_t = t-1, D_{t-1}) + \nabla \log f_\theta(C_t = t-1 | C_{t-1} = \tilde{c}_t^{(i)}) + \beta_{t-1}^{(i)} \right]. \end{aligned} \quad (7.31)$$

Usando (7.23), obtem-se

$$\nabla \hat{\log} p_\theta(z_t | D_{t-1}) = \frac{\sum_{i=1}^{N_{t-1}+1} \tilde{\alpha}_t^{(i)}}{\sum_{i=1}^{N_{t-1}+1} \tilde{w}_t^{(i)}}. \quad (7.32)$$

O algoritmo de Caron et al. (2012) pode ser resumido através dos seguintes passos:

Algoritmo de Caron et al. (2012):

- No tempo $t = 1$: Escolha θ_0 , $c_1^1 = 1$, $w_1^1 = 1$, $\beta_1^1 = 0$ e $N_1 = 1$.
- No tempo $t \geq 2$: para $i = 1, \dots, N_{t-1}$ seja $\tilde{c}_t^i = c_{t-1}^i$ e faça $\tilde{c}_t^{N_{t-1}+1} = t-1$.
- Para $i = 1, \dots, N_{t-1} + 1$ calcule \tilde{w}_t^i usando (7.28)-(7.29) e θ_{t-1} .
- Para $i = 1, \dots, N_{t-1} + 1$ calcule $\tilde{\alpha}_t^i$ usando (7.30)-(7.31) e θ_{t-1} .
- Atualize o vetor de parâmetros utilizando

$$\theta_t = \theta_{t-1} + \gamma_t \frac{\sum_{i=1}^{N_{t-1}+1} \tilde{\alpha}_t^i}{\sum_{i=1}^{N_{t-1}+1} \tilde{w}_t^i}.$$

- Seleção de Partículas: Selecione as partículas tal que seu peso normalizado seja $\bar{w}_t^{(i)} > \epsilon$.
Faça N_t igual ao número de partículas selecionadas.
- Para $i = 1, \dots, N_t$, calcule $\beta_t^{(i)}$ usando

$$\beta_t^{(i)} = \frac{\tilde{\alpha}_t^{\phi(i)}}{w_t^{(i)} \sum_{j=1}^{N_t} \tilde{w}_t^{\phi(j)}} - \frac{\sum_{j=1}^{N_t} \tilde{\alpha}_t^{\phi(j)}}{\sum_{j=1}^{N_t} \tilde{w}_t^{\phi(j)}},$$

para $i = 1, \dots, N_t$, onde $\phi : \{1, \dots, N_t\} \rightarrow \{1, \dots, N_{t-1} + 1\}$ é a função injetiva, tal que $w_t^{(i)} = \bar{w}_t^{\phi(i)}$.

Tipicamente as estimativas dos parâmetros convergem antes de T para grandes valores de T . Para o caso em que há uma base de dados pequena, Caron et al. (2012) sugerem executar o filtro de partículas $K > 1$ vezes na base de dados, usando $\theta_1^{(j)} = \theta_T^{(j-1)}$ e $\gamma_1^{(j)} = \gamma_T^{(j-1)}$ para $j = 2, \dots, K$, para se obter uma convergência.

7.3 Modelo Dinâmico Poisson com Ponto de Mudança

7.3.1 Modelo Dinâmico Poisson com Ponto de Mudança seguindo o filtro de Chopin (2007)

Nesta seção será descrito o algoritmo de Chopin (2007) para a distribuição de Poisson. Mas a abordagem adotada nesta dissertação segue a linha de Fearnhead e Liu (2007), o qual ao invés de modelar o tempo de duração de cada segmento entre pontos de mudança, como apresentado na seção, ele modela os tempos em que ocorreram as mudanças. A justificativa de se usar essa abordagem é que assim unificou-se a metodologia para ambos os filtros de partículas descritos nesta dissertação, o que facilitou a implementação de ambos.

Para tanto, os pontos de mudança C_t no modelo seguem um processo de Markov com probabilidade de transição igual a equação definida em (7.14). Para esta dissertação $h(\cdot)$ foi escolhida como uma distribuição geométrica. Sendo assim, a probabilidade de transição na equação (7.14) é simplificada, e é dada por

$$P(C_t = j | C_{t-1} = i) = \begin{cases} 1 - p & \text{se } j = i, \\ p & \text{se } j = t - 1. \end{cases} \quad (7.33)$$

De posse dessas premissas e seguindo as ideias dadas em Fearnhead e Liu (2007), o algoritmo de Chopin para a distribuição de Poisson, implementado nesta dissertação, pode ser resumido nos seguintes passos:

Filtro de partículas Rao-Blackwellizado segundo Fearnhead e Liu (2007):

- **Passo 1:** Para cada partícula $x_{t-1}^{(j)} = (c_{t-1}^{(j)}, \theta_{t-1}^{(j)})$, para $j = 1, 2, \dots, H$, amostre

$$\begin{aligned}x_t^{(j,1)} &= (\theta_{t-1}^{(j)}, c_{t-1}^{(j)}) \\x_t^{(j,2)} &= (\epsilon^{(j)}, t-1).\end{aligned}$$

- **Passo 2:** Repondere as partículas com pesos

$$\begin{aligned}w_t^{(j,1)} &= p(y_t | \theta_t = \theta_{t-1}^{(j)}, D_{t-1})(1-p) \\w_t^{(j,2)} &= p(y_t | \theta_t = \epsilon^{(j)}, D_{t-1})p.\end{aligned}$$

- **Passo 3:** Reamostre as $2H$ partículas com pesos respectivos $w_t^{(j,1)}$, $w_t^{(j,2)}$, de forma a obter H partículas reamostradas.

- **Passo 4:** Para cada partícula selecionada $x_t^{(j)} = (\theta_t^{(j)}, c_t^{(j)})$, substitua $\theta_t^{(j)}$ por

$$\tilde{\theta}_t^{(j)} \sim k_t^{(j)}(\theta_t^{(j)}, \cdot),$$

onde $k_t^{(j)}$ é um kernel MCMC com distribuição invariante $\eta_t^{(j)}$ como em (7.5).

- **Passo 5:** Faça $t = t + 1$ e vá ao passo 1.

O algoritmo para suavização pode ser resumido nos seguintes passos

- Faça $k = T$;
- Enquanto $k > 0$
 1. Amostre $x_k^{(j)}$ a partir de $p(x_k | D_k)$, ou seja, reamostre, com reposição, a partícula $x_k^{(j)}$ com peso proporcional a $w_k^{(j)}$;
 2. Faça $k \leftarrow c_k^{(j)}$, para $j = 1, 2, \dots, H$.

7.3.1.1 Análise de dados simulados

Para ilustrar o filtro de partículas de Chopin (2007), aplicado à distribuição de Poisson, foi simulada a seguinte base de dados:

- Simulou-se uma série temporal de tamanho $T = 600$, tal que os segmentos são dados por
 1. **Segmento 1:** para $0 \leq t \leq 200$ amostra-se de $y_t \sim \text{Poisson}(\lambda_t = 3)$;
 2. **Segmento 2:** para $201 \leq t \leq 400$ amostra-se de $y_t \sim \text{Poisson}(\lambda_t = 1)$;
 3. **Segmento 3:** para $401 \leq t \leq 600$ amostra-se de $y_t \sim \text{Poisson}(\lambda_t = 3)$.
- Portanto, as mudanças ocorrem em $\tau_1 = 200$ e $\tau_2 = 400$.

A base de dados foi gerada no software *R*, a programação utilizada para a geração da base é dada por

Simulação

```

1  #SIMULANDO UM MODELO POISSON COM PONTOS DE MUDANÇA
   set.seed(456)
3  T = 600
   y <- rep(0,T)
5  d <- y
   #Regimes (Pontos de mudanças ocorrem em 200, 400)
7  tau = c(0, 200, 400, T)
   #Valores dos parâmetros em cada regime.
9  lambdat = c(3, 1, 3)
   #simulando a série
11 for (j in 1:(length(tau)-1)){
     for (i in (tau[j]+2):(tau[j+1])){
13       y[i] <- rpois(1, lambda=lambdat[j])
     }
15 }

17 plot(y, type="l")
   abline(v=c(200, 400), col="red", lty=2)

```

Nesta dissertação foram criadas duas funções chamadas *ChopinFilter* e *ChopinSmooth*, que foram utilizadas para ajustar o filtro de Chopin a dados de contagem e fazer a suavização, respectivamente. O número de partículas utilizado para a aplicação do filtro foi $H = 50000$. As linhas de programação criadas para analisar os dados simulados são dadas por

Ajuste

```

filt_poisson <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), NParticle=50000,
2  family=POISSON, gama=0.75)
smooth_poisson <- ChopinSmooth(y=y, particles=filt1$Particles,

```

```

4 weight=filt1$Weight)

6 par(mfrow=c(3,1))
  plot(y, type="l")
8 abline(v=c(200, 400), col="red", lty=2)

10 #ESTADOS
  plot(filt_poisson$thetaf,lwd=1, lty=1, type="l",
12 ylim=c(min(filt_poisson$lthetaf), max(filt_poisson$uthetaf)),
  main="filter")
14 lines(filt_poisson$lthetaf,lty=2)
  lines(filt_poisson$uthetaf,lty=2)
16 abline(v=c(200, 400), lty=2, col="red")

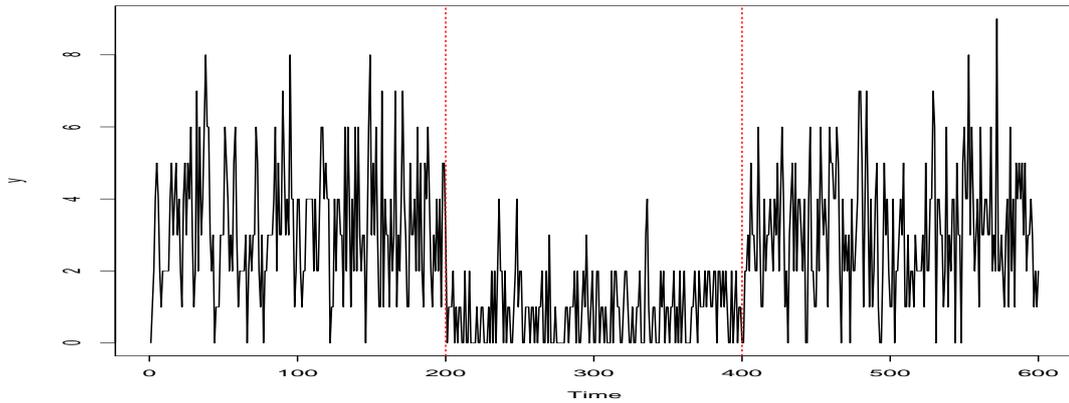
18 #PROBABILIDADE A POSTERIORI DE MUDANÇA
  plot(smooth_poisson$prob, type="h", ylab="", lwd=2,
20 col="black",
  xlab="Tempo", main="")
22 abline(v=c(200, 400), lty=2, col="red")

```

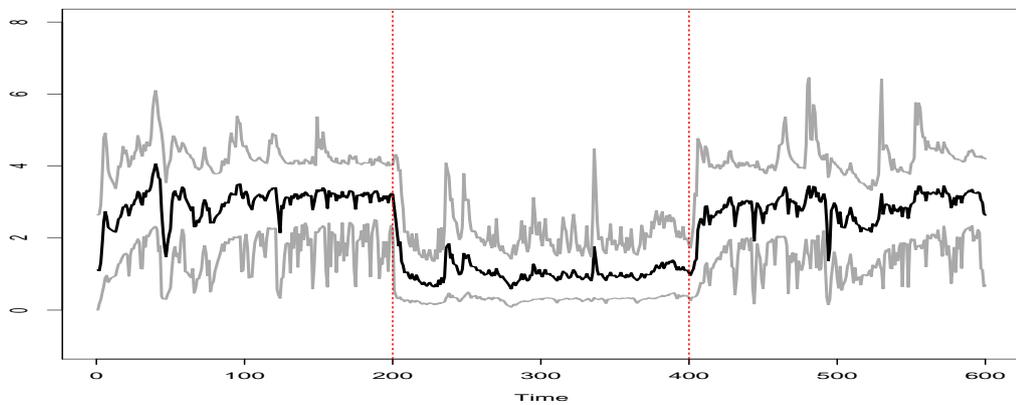
Os resultados são ilustrados graficamente na Figura 7.3.

Afim de verificar a consistência das estimativas, o filtro foi executado 40 vezes utilizando 1000 partículas, posteriormente construímos os histogramas dos estados em cada segmento. Os resultados estão ilustrados na Figura 7.4. Cada uma das 40 trajetórias foram plotadas em cinza com as trajetórias medianas plotadas em preto.

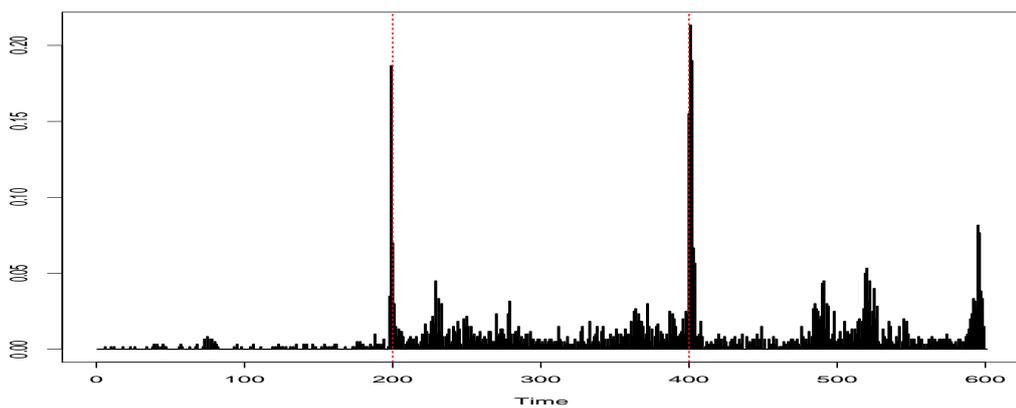
Nota-se que os histogramas estão concentrados próximos dos valores reais, marcados pelas linhas tracejadas em azul. Dessa forma, pode-se ver que uma estratégia que pode ser utilizada para se analisar dados reais via Filtro de Partículas consiste em executar o filtro mais de uma vez e utilizar a mediana das trajetórias como estimativa final para os estados, o que garante uma melhor aproximação para o estado real.



(a) Série simulada y_t

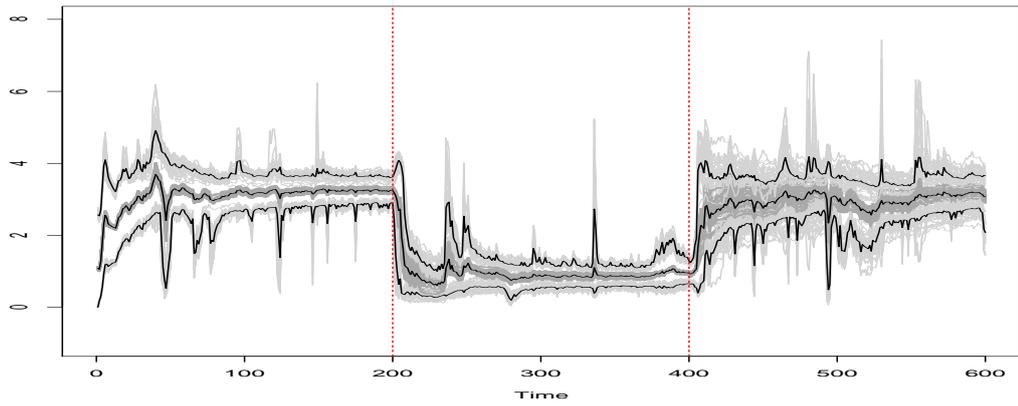


(b) Média filtrada

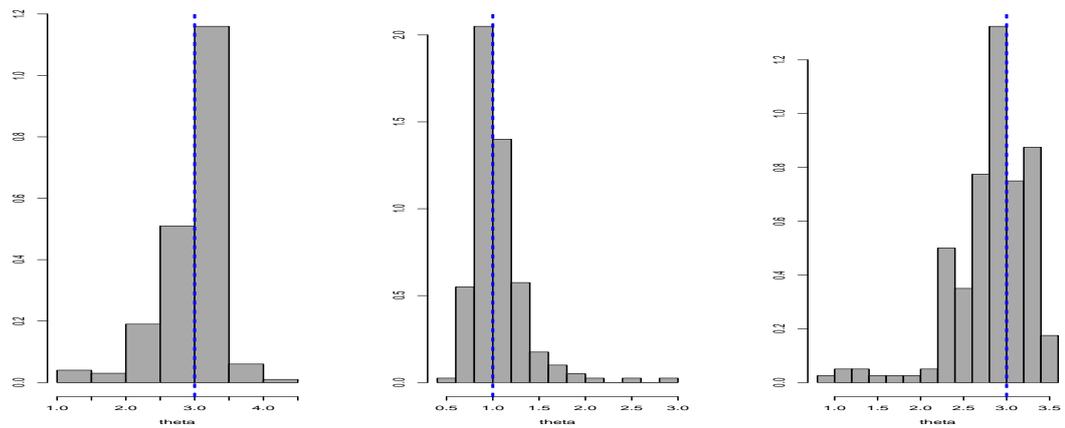


(c) Probabilidade a *posteriori* de ocorrer mudança

Figura 7.3: De cima para baixo, gráfico da série y_t , as estimativas dos estados com intervalo de confiança de 95% e probabilidade a posteriori de ocorrer um mudança no tempo t . As barras tracejadas verticais vermelhas mostram os pontos de mudança simulados.



(a) Filtragem replicada



(b) Densidade empírica dos estados em cada segmento

Figura 7.4: De cima para baixo, trajetórias replicadas para cada execução do filtro com 1000 partículas em cinza e trajetória mediana em preto. Histograma dos estados em cada segmento, os valores reais são representados pelas linhas tracejadas azul.

7.3.2 Modelo Dinâmico Poisson com Ponto de Mudança segundo o filtro de Caron et al. (2012)

Para a aplicação do filtro proposto por Caron et al. (2012) utilizou-se o modelo Poisson-gamma, ou seja

$$\begin{aligned} y_t &\sim \text{Poisson}(\lambda_t), \\ \lambda_t &\sim \text{Gamma}(a, b). \end{aligned}$$

Conforme mostrado na equação (7.11), a distribuição do segmento $z_{i:j}$, $P(i, j|r)$, é dada por

$$P(i, j|r) = \frac{b^a}{\Gamma(a) \left[\prod_{k=i}^j y_k! \right]} \frac{\Gamma(s^* + a - 1)}{(n^* + b)^{s^* + a - 1}}.$$

onde $s^* = \sum_{k=i}^j y_k$ e $n^* = j - i + 1$. Tal como no filtro de Chopin (2007), a probabilidade de transição foi escolhida tal que $h(\cdot)$ siga uma distribuição geométrica. Sendo assim, a equação (7.14) se reduz a

$$P(C_t = j | C_{t-1} = i) = \begin{cases} 1 - p & \text{se } j = i, \\ p & \text{se } j = t - 1, \end{cases} \quad (7.34)$$

Para a atualização do parâmetro θ_t , vide equação (7.22), foi utilizado o método de Newton-Raphson conforme descrito em Poyiadis et al. (2011). Sendo assim $\gamma_t = \nabla^2 \log p_{\theta_{1:t-1}}(z_t | D_{t-1})$. Afim de criar uma função mais geral, utilizou-se as derivadas calculadas numericamente, apesar do modelo Poisson-gamma admitir o cálculo das derivadas analiticamente.

De posse das premissas adotadas acima, a implementação do filtro para a distribuição de Poisson seguiu os passos do algoritmo geral resumidos na Seção 6.3.2 do Capítulo 6.

7.3.2.1 Análise de dados simulados

Para ilustrar o filtro de partículas de Caron et al. (2012), aplicado a distribuição de Poisson, foram simuladas duas bases de dados

- **Base de dados 1:** Simulou-se uma série de tamanho $T = 400$ com 3 pontos de mudança nos pontos $\tau_1 = 100$, $\tau_2 = 200$ e $\tau_3 = 300$. O valor de λ_t varia dentro de cada segmento da seguinte maneira:
 1. **Segmento 1:** para $1 \leq t \leq 100$ utilizou-se $\lambda_t = 4$;
 2. **Segmento 2:** para $101 \leq t \leq 200$ utilizou-se $\lambda_t = 1$;
 3. **Segmento 3:** para $201 \leq t \leq 300$ utilizou-se $\lambda_t = 4$;
 4. **Segmento 4:** para $301 \leq t \leq 400$ utilizou-se $\lambda_t = 1$.
- **Base de dados 2:** Simulou-se uma série de tamanho $T = 250$ com 4 pontos de mudança nos pontos $\tau_1 = 50$, $\tau_2 = 100$, $\tau_3 = 150$ e $\tau_4 = 200$. O valor de λ_t varia dentro de cada segmento da seguinte maneira:
 1. **Segmento 1:** para $1 \leq t \leq 50$ utilizou-se $\lambda_t = 4$;
 2. **Segmento 2:** para $51 \leq t \leq 100$ utilizou-se $\lambda_t = 1$;
 3. **Segmento 3:** para $101 \leq t \leq 150$ utilizou-se $\lambda_t = 4$;
 4. **Segmento 4:** para $151 \leq t \leq 200$ utilizou-se $\lambda_t = 1$;
 5. **Segmento 5:** para $201 \leq t \leq 250$ utilizou-se $\lambda_t = 4$.

As bases de dados foram geradas no software *R*, a programação utilizada para a geração de cada base é dada por

– Base de dados 1:

Simulação - Base 1

```
1 k <- 100
  y <- c(rpois(k, 4), rpois(k, 1), rpois(k, 4), rpois(k,1))
3 n <- length(y)

5 plot(y, type="l")
  abline(v=c(100, 200, 300), col="red", lty=2, xlab="Time")
```

– Base de dados 2:

Simulação - Base 2

```
  k <- 50
2 y <- c(rpois(k, 4), rpois(k, 1), rpois(k, 4), rpois(k,1), rpois(k,4))
  n <- length(y)
4
  plot(y, type="l")
6 abline(v=c(50, 100, 150, 200), col="red", lty=2, xlab="Time")
```

Nesta dissertação foram criadas duas funções chamadas *caronFilter* e *caronSmooth*, em que a primeira delas ajusta o filtro de Caron et al. (2012) a dados de contagem e a segunda faz a suavização. As linhas de programação criadas para analisar os dados simulados são dadas por

– Base de dados 1:

Ajuste - Base 1

```
#Ajustando o filtro de Caron
2 filter = caronFilter(y=y, p=0.001, theta0=c(2,1), epsilon=0.00001)
  smooth = caronSmooth(y=y, particles=filter$Particles, weight=filter$Weight
  )
4
#Ajustando o filtro de Chopin
6 filt1 <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), NParticle=1000,
  family=POISSON, gama=0.75)
8 smooth1 <- ChopinSmooth(y=y, particles=filt1$Particles,
  weight=filt1$Weight)
10
  par(mfrow=c(2,2))
12 plot(y, type="l", xlab="Tempo", main="Dados Poisson")
  abline(v=c(50, 100, 150, 200), col="red", lty=2, xlab="Time")
14
#PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL PARA O ALGORITMO
```

```

16 DE CARON
   plot(smooth$prob, type="h", lwd=2, xlab="Tempo", ylab="Posteriori da
18 mudança",
   main="Algoritmo Caron et. al. (2011)")
20 abline(v=c(50, 100, 150, 200), col="red", lty=2, xlab="Time")

22 #NÚMERO DE PARTÍCULAS UTILIZADO EM CADA ITERAÇÃO DO
   #ALGORITMO DE CARON ET AL. (2011)
24 plot(filter$NP,type="l", xlab="Tempo",
   ylab="Número de partículas",
26 main="Número de partículas")

28 #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL PARA O
   #ALGORITMO DE CHOPIN
30 plot(smooth1$prob, type="h", ylab="Posteriori da mudança", lwd=2,
   col="black",
32 xlab="Tempo", main="Algoritmo Chopin (2006)")
   abline(v=c(50, 100, 150, 200), col="red", lty=2, xlab="Time")

```

– Base de dados 2:

Ajuste - Base 2

```

#Ajustando o filtro de Caron
2 filter = caronFilter(y=y, p=0.001, theta0=c(2,1), epsilon=0.00001)
   smooth = caronSmooth(y=y, particles=filter$Particles, weight=filter$Weight
   )
4
#Ajustando o filtro de Chopin
6 filt1 <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), NParticle=1000,
   family=POISSON, gama=0.75)
8 smooth1 <- ChopinSmooth(y=y, particles=filt1$Particles,
   weight=filt1$Weight)
10
   par(mfrow=c(2,2))
12 plot(y, type="l", xlab="Tempo", main="Dados Poisson")
   abline(v=c(100, 200, 300, col="red", lty=2)
14
   #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL PARA
16 #O ALGORITMO DE CARON
   plot(smooth$prob, type="h", lwd=2, xlab="Tempo",
18 ylab="Posteriori da mudança",
   main="Algoritmo Caron et. al. (2011)")
20 abline(v=c(100, 200, 300, col="red", lty=2)

22 #NÚMERO DE PARTÍCULAS UTILIZADO EM CADA ITERAÇÃO DO ALGORITMO DE
   #CARON ET AL. (2011)
24 plot(filter$NP,type="l", xlab="Tempo", ylab="Número de partículas",

```

```

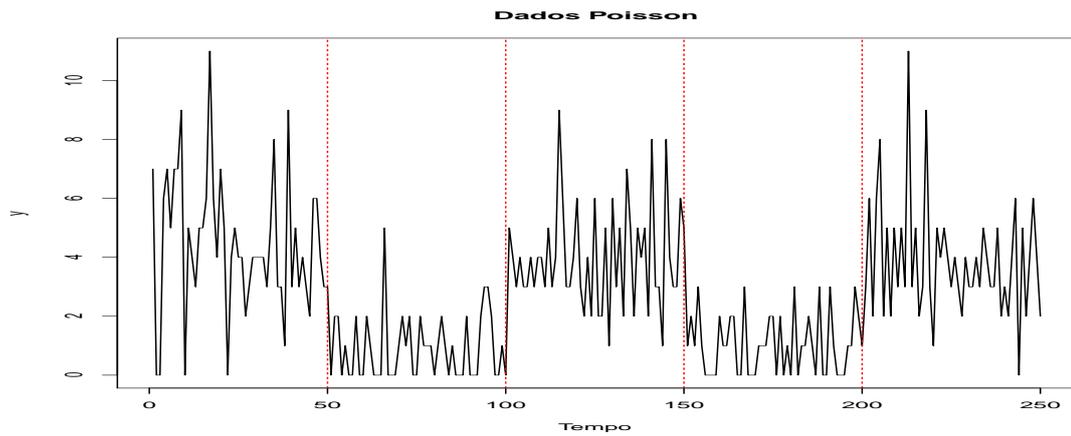
    main="Número de partículas")
26
    #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL PARA O
28 #ALGORITMO DE CHOPIN
    plot(smooth1$prob, type="h", ylab="Posteriori da mudança", lwd=2,
30 col="black",
    xlab="Tempo", main="Algoritmo Chopin (2006)")
32 abline(v=c(100, 200, 300, col="red", lty=2)

```

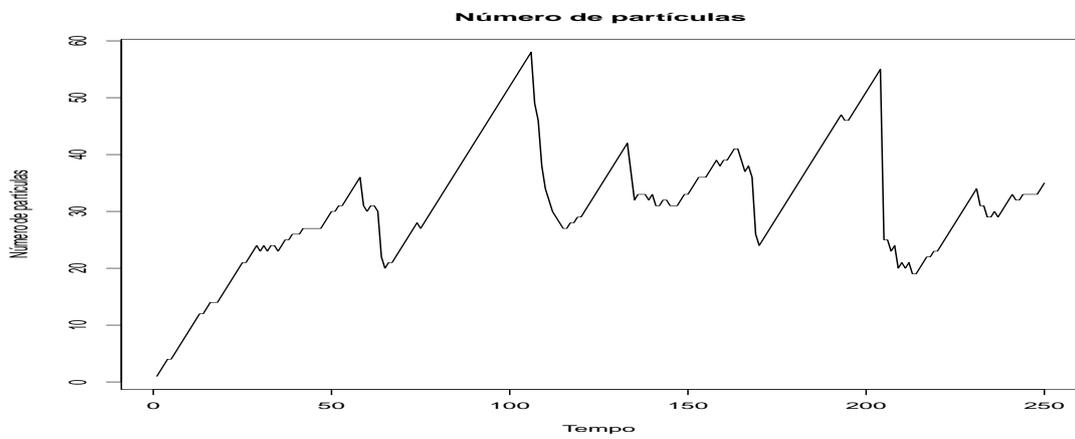
Os resultados, bem como a série simulada, estão ilustrados nas figuras 7.5 e 7.6.

O tempo de processamento do filtro de Caron para a primeira base foi de, aproximadamente, 5 segundos, enquanto que o filtro de Chopin, com $H = 1000$ partículas, levou aproximadamente 12 segundos. Para a segunda base o filtro de Caron levou aproximadamente 8 segundos contra 15 do filtro de Chopin. Dessa forma, observa-se que o filtro de Caron possui um menor tempo de processamento, por utilizar menos partículas.

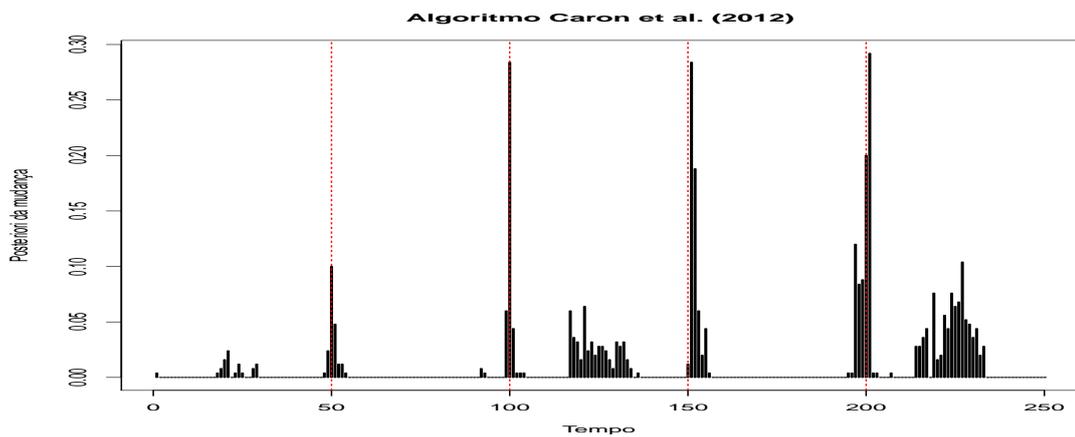
No próximo capítulo será ilustrado as novas extensões propostas para o algoritmo de McCormick et. al (2012) e para o filtro de Chopin (2007). Tais extensões permitem que esses algoritmos estimem parâmetros estáticos, além dos estados.



(a) Série simulada y_t

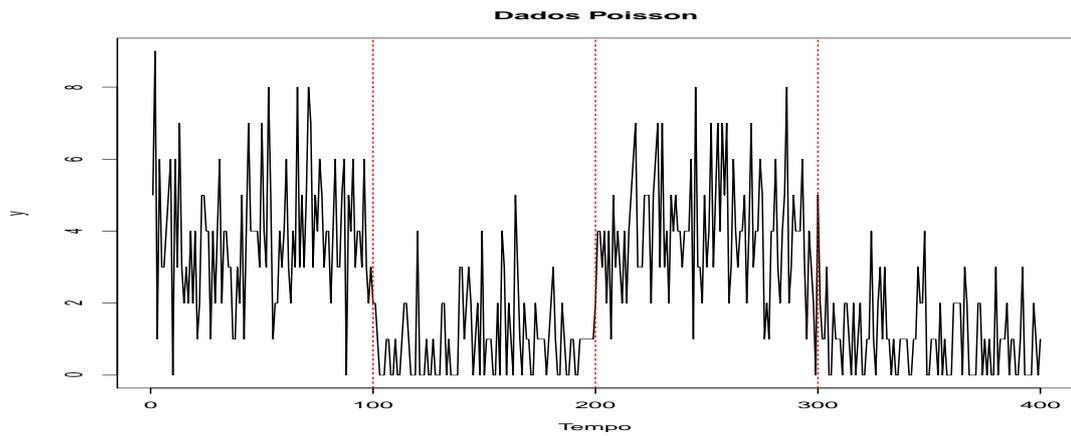


(b) Número de partículas selecionadas

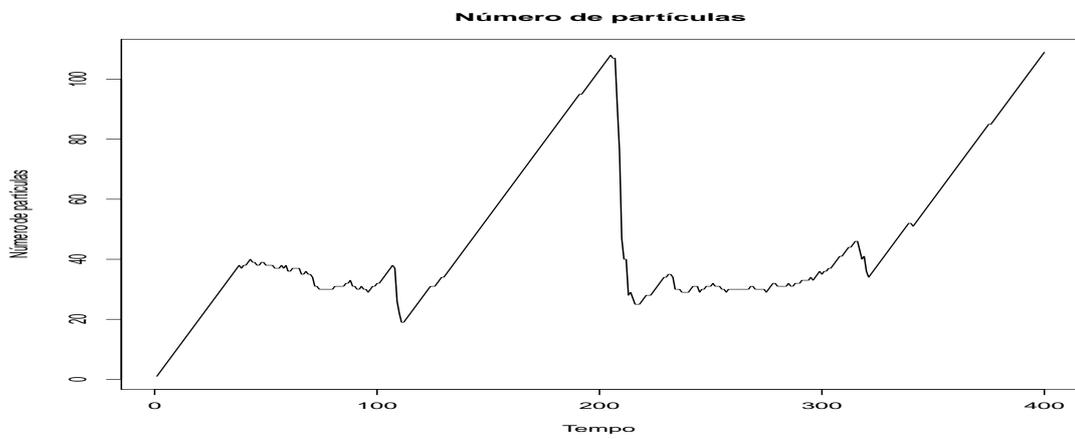


(c) Probabilidade *a posteriori* de ocorrer mudança

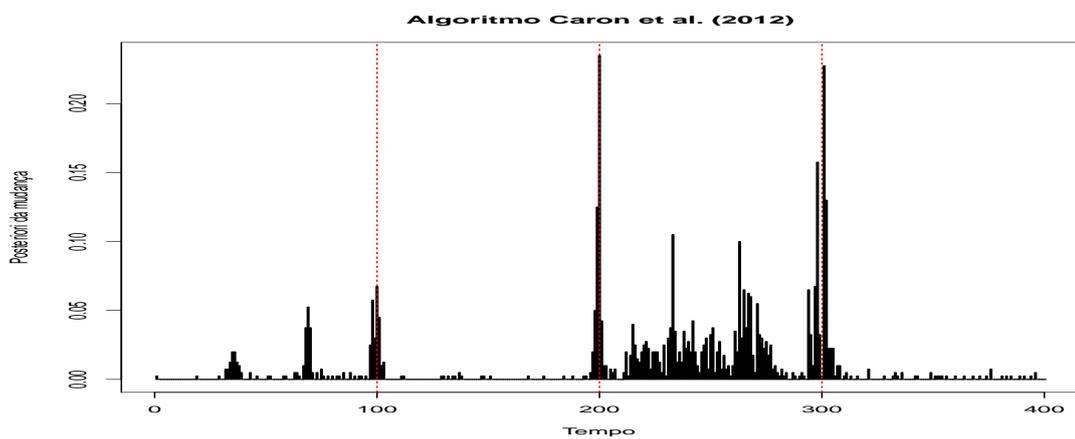
Figura 7.5: De cima para baixo, gráfico da série y_t . Número de partículas utilizadas em cada iteração do filtro de Caron et al. (2012). Probabilidade *a posteriori* de ocorrer um mudança no tempo t segundo filtro de Caron et al. (2012).



(a) Série simulada y_t



(b) Número de partículas selecionadas



(c) Probabilidade *a posteriori* de ocorrer mudança

Figura 7.6: De cima para baixo, gráfico da série y_t . Número de partículas utilizadas em cada iteração do filtro de Caron et al. (2012). Probabilidade *a posteriori* de ocorrer um mudança no tempo t segundo filtro de Caron et al. (2012).

Capítulo 8

Novos desenvolvimentos

Ao se analisar dados reais de contagem, frequentemente o analista depara-se com superdispersão e/ou excesso de zeros. Sendo assim, o modelo Poisson passa a não ser adequado e outros modelos para dados de contagem devem ser utilizados. Nesta dissertação foram desenvolvidos dois algoritmos inéditos para permitir a estimação conjunta dos estados e dos parâmetros estáticos, generalizando os algoritmos propostos por McCormick et al. (2012) e Chopin (2007). Estes algoritmos permitiram a criação dos modelos de regressão dinâmica e filtros de partículas para a detecção de pontos de mudança para os modelos Binomial Negativo, Poisson inflacionado de zeros (ZIP) e Binomial Negativo inflacionado de zeros (ZINB), que serão apresentados nos próximos capítulos e ilustrados por meio de dados simulados. Tais algoritmos são denominados como *Algoritmo de McCormick com parâmetros estáticos* (McPE) e *Filtro de Chopin com Aprendizado de Partículas* (FChAP).

Este capítulo está dividido da seguinte forma: Na Seção 8.1 apresentamos o algoritmo McPE, que representa uma extensão do algoritmo proposto por McCormick et al (2012), de modo a permitir a estimação dos parâmetros estáticos. Na Seção 8.2 apresentamos o filtro FChAP, no qual combinamos a metodologia descrita por Chopin (2007) para a estimação dos estados de uma série com pontos de mudança, com aprendizado de partículas, segundo o esquema proposto por Liu e West (2001), na estimação de parâmetros estáticos.

8.1 Algoritmo McPE (McCormick com parâmetros estáticos)

Nesta seção apresentamos uma generalização para o modelo de regressão dinâmica proposto por McCormick et al. (2012) que denominamos por McPE (Algoritmo de McCormick com parâmetros estáticos), para a estimação conjunta dos estados e dos parâmetros estáticos. Para tanto, considere para $t = 1, 2, \dots, T$.

$$\begin{aligned} y_t &\sim f(\cdot | \mu_t, \Psi), \\ \mu_t &= g(\theta_t). \end{aligned}$$

Aqui Ψ representa o vetor composto pelos parâmetros estáticos das equações de observação. Como no algoritmo descrito na Seção 6.1, o procedimento de estimação é feito em duas etapas: (1) predição e (2) atualização. O processo de estimação começa supondo

$$\theta_{t-1} | D_{t-1} \sim N(\hat{\theta}_t, \hat{\Sigma}_{t-1}).$$

Então a equação de predição é

$$\theta_t | D_{t-1} \sim N(\hat{\theta}_t, R_t), \quad (8.1)$$

onde $R_t = \frac{\hat{\Sigma}_{t-1}}{\lambda_t}$. Para a etapa de atualização, combina-se a equação de predição e a observação adicional no tempo t . A distribuição a posteriori da estimativa atualizada é dada por

$$p(\theta_t | \Psi, D_t) \propto p(y_t | \Psi, \theta_t) p(\theta_t | \Psi, D_{t-1}). \quad (8.2)$$

Novamente, o lado direito da equação (8.2) é aproximado por uma distribuição Normal. Para a estimação do estado θ_t utiliza-se o método de Newton-Raphson:

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \left[D^2 l(\hat{\theta}_{t-1}) \right]^{-1} D l(\hat{\theta}_{t-1}),$$

onde $l(\theta) = p(y_t|\theta, \Psi)p(\theta|D_{t-1})$. Para a estimação dos parâmetros estáticos Ψ e o fator de desconto λ_t utiliza-se a verossimilhança preditiva:

$$f(y_t|\Psi, D_{t-1}) = \int_{\theta_t} p(y_t|\theta_t, D_{t-1})p(\theta_t, D_{t-1})d\theta_t. \quad (8.3)$$

Em muitos casos, essa integral não pode ser calculada de forma fechada. Sendo assim, usa-se a aproximação de Laplace:

$$f(y_t|\Psi, D_{t-1}) \approx (2\pi)^{d/2}|[D^2(\hat{\theta}_t)]^{-1}|p(y_t|\Psi, \hat{\theta}_t, D_{t-1})p(\hat{\theta}_t|D_{t-1}). \quad (8.4)$$

Note que $\frac{\partial}{\partial \Psi \partial \lambda_t} f(y_t|\Psi, D_{t-1}) = \frac{\partial}{\partial \lambda_t \partial \Psi} f(y_t|\Psi, D_{t-1}) = 0$. Logo, λ_t e Ψ são ortogonais e podem ser estimados separadamente. Para tanto, utilizou-se a seguinte estratégia para a estimação de λ_t e Ψ :

- **Passo 1:** Dado $\hat{\lambda}_{t-1}$ e $\hat{\theta}_t$, estime Ψ a partir de $f(y_t|\Psi, D_{t-1})$ utilizando estimação por máxima verossimilhança;
- **Passo 2:** Dado $\hat{\Psi}$, estimado no passo 1, e $\hat{\theta}_t$, atualize λ_t como

$$\hat{\lambda}_t = \operatorname{argmax}_{\lambda_t} f(y_t|\hat{\Psi}, D_{t-1}).$$

Considerando o que foi apresentado até o momento, a extensão do algoritmo de McCormick que permite a inclusão de parâmetros estáticos pode ser resumida da seguinte forma:

Algoritmo McPE:

1. Inicie o algoritmo. Como exemplo, pode-se iniciá-lo como: $\hat{\theta}_0 = g(y_1 + 0.00001)$, $\hat{\Psi}_0 = \Psi_{EMV}$, $\hat{y}_1 = g^{-1}(x_1\hat{\theta}_0)$, $\lambda_1 = 0.99$, $\Sigma_1 = 100$, $R_1 = \Sigma_1/\lambda_1$, $Dl(\hat{\theta}_0)$ e $D^2l(\hat{\theta}_0)$;
2. Calcule $\hat{y}_t = g^{-1}(x_t\hat{\theta}_{t-1})$;
3. Calcule $R_t = \hat{\Sigma}_{t-1}/\lambda_{t-1}$;
4. Atualize $Dl(\hat{\theta}_{t-1})$ e $D^2l(\hat{\theta}_{t-1})$;
5. Atualize o vetor de parâmetros $\hat{\theta}_t$ de acordo com a equação (6.6);

6. Dado λ_{t-1} e $\hat{\theta}_t$, estime Ψ a partir de $f(y_t|\Psi, D_{t-1})$ utilizando o método de máxima verossimilhança;
7. Dado $\hat{\Psi}$, estimado no passo anterior, e $\hat{\theta}_t$ atualize λ_t como $\hat{\lambda}_t = \operatorname{argmax}_{\lambda_t} f(y_t|\hat{\Psi}, D_{t-1})$;
8. Atualiza a matriz de covariâncias, $\hat{\Sigma}_t = -\left[D^2l(\hat{\theta}_{t-1})\right]^{-1}$;
9. Atualize, novamente, $R_t = \hat{\Sigma}_{t-1}/\lambda_t$.
10. Faça $t = t + 1$ e volte ao passo 1.

Observações de cunho prático:

1. Para os modelos que apresentam muitos parâmetros estáticos, podem ocorrer frequentes problemas numéricos nas maximizações das funções. Sendo assim, pode-se adotar um procedimento *ad-hoc*: para os parâmetros estáticos relacionados com a variabilidade de y_t , pode-se atualizá-lo utilizando um fator de desconto δ (por exemplo $\delta = 0.9999$), ou seja, $\Psi_t = \Psi_{t-1}/\delta$. Tal procedimento será ilustrado para o caso em que os dados seguem uma distribuição Binomial negativa inflacionada de zeros, para a qual o parâmetro de dispersão, k , é atualizado como $k_t = k_{t-1}/\delta$. Iremos mostrar que, mesmo escolhendo valores iniciais para k distantes do real, as estimativas dos parâmetros de regressão obtidas são muito próximas dos verdadeiros.
2. Uma forma de se escolher parâmetros iniciais para os parâmetros estáticos dos modelos é ajustar um modelo de regressão estático e escolher as estimativas provenientes dessa modelagem e utilizá-las como parâmetros iniciais. A justificativa se dá pela lei forte dos grandes números, que, mesmo em dados correlacionados, garante que os estimadores de máxima verossimilhança pontuais dos parâmetros convergem quase certamente para os parâmetros reais (populacionais).
3. Para amostras pequenas, pode-se adotar o procedimento sugerido por Caron et al. (2012), ou seja, executar a regressão $K > 1$ vezes na base de dados, usando $\theta_1^{(j)} = \theta_T^{(j-1)}$ e $\Psi_1^{(j)} = \Psi_T^{(j-1)}$, para $j = 2, \dots, K$, para se obter convergência.

4. As derivadas $Dl(\hat{\theta})$ e $D^2l(\hat{\theta})$ podem ser calculadas numericamente, dependendo da complexidade do modelo.

8.2 Algoritmo FChAP (Filtro de Chopin com aprendizado de partículas)

Nesta seção apresentamos uma generalização do filtro de partículas proposto por Chopin (2007), de modo a permitir a estimação dos parâmetros estáticos Ψ da equação das observações. Denominamos o novo algoritmo por FChAP (Filtro de Chopin com aprendizado de partículas). Para tanto, combinou-se o filtro de Liu e West (2001), com o filtro de Chopin (2007), para a estimação conjunta dos estados e dos parâmetros estáticos. É interessante (e lamentável) observar que Nemeth et al. (2014) se anteciparam em relação a nós ao publicarem um artigo que traz, justamente, esse tipo de desenvolvimento. O algoritmo de Chopin com aprendizado de partículas pode ser resumido da seguinte forma

Algoritmo FChAP:

- Gere valores iniciais para $\theta_0^{(i)}$ a partir de $\pi_\epsilon(\cdot)$.
- **Passo 1:** Para $i = 1, 2, \dots, H$, amostre

$$\begin{aligned}\Psi_t^{(i)} &\sim P(\Psi_1), \\ \epsilon_t^{(i)} &\sim \pi_\epsilon(\cdot).\end{aligned}\tag{8.5}$$

- **Passo 2:** Faça

$$\begin{aligned}x_t^{(i,1)} &= (t-1, \epsilon^i), \\ x_t^{(i,2)} &= (c_{t-1}^i, \theta_{t-1}^i);\end{aligned}$$

– **Passo 3:** Calcule

$$\begin{aligned} m_t^{(i)} &= a\Psi_t^{(i)} + (1-a)\bar{\Psi}_t \\ \hat{\Sigma}_t^{(i)} &= (1-a^2) \frac{\sum_{i=1}^H (\Psi_t^{(i)} - \bar{\Psi}_t)(\Psi_t^{(i)} - \bar{\Psi}_t)^T}{H} \\ w_t^{(j,1)} &= p(y_t | \theta_t = \theta_{t-1}^{(j)}, D_{t-1})(1-p) \\ w_t^{(j,2)} &= p(y_t | \theta_t = \epsilon^{(j)}, D_{t-1})p, \end{aligned}$$

onde $\bar{\Psi}_t = \frac{\sum_{i=1}^H \Psi_t^{(i)}}{H}$;

– **Passo 4:** Reamostramos as $2H$ partículas com pesos respectivos $w_t^{(j,1)}$, $w_t^{(j,2)}$, de forma a obter H partículas reamostradas;

– **Passo 5:** Para cada partícula selecionada $x_t^{(j)} = (\theta_t^{(j)}, c_t^{(j)})$, substitua $\theta_t^{(j)}$ por

$$\tilde{\theta}_t^{(j)} \sim k_t^{(j)}(\theta_t^{(j)}, \cdot),$$

onde $k_t^{(j)}$ é um kernel MCMC com distribuição invariante $\eta_t^{(j)}$ como em (7.5);

– **Passo 6:** Se $\left[w_t^{(i,1)} / (w_t^{(i,1)} + w_t^{(i,2)}) \right] > u$, onde $U \sim U(0, 1)$, então amostramos $\hat{\Psi}_t^{(i)} \sim N(m_t^{(i)}, \hat{\Sigma}_t^{(i)})$. Caso contrário faça $\hat{\Psi}_t^{(i)} = \Psi_t^{(i)}$;

– **Passo 7:** Faça $t = t + 1$ e vá ao passo 1.

Para os modelos desenvolvidos nos próximos capítulos, a probabilidade da função de transição foi fixada em $p = 1 \times 10^{-6}$. No próximo capítulo, aplicaremos os algoritmos desenvolvidos aqui aos modelos Binomial Negativo, ZIP e ZINB. Posteriormente ilustraremos os resultados por meio de dados simulados.

Capítulo 9

Análise de dados simulados usando os novos desenvolvimentos

Neste capítulo aplicaremos os algoritmos que foram desenvolvidos no capítulo anterior para criar os modelos de regressão dinâmica com pontos de mudança e detecção de pontos de mudança para dados de contagem, com superdispersão e/ou inflação de zeros por meio das distribuições Binomial Negativa (BN), Poisson inflacionado de zeros (ZIP) e Binomial Negativa inflacionada de zeros (ZINB). Tais modelos serão exemplificados por meio de dados simulados.

9.1 Modelos para dados de contagem com superdispersão e pontos de mudança: Modelo Binomial Negativo

Considere que $Y_t \sim \text{BN}(k, \mu_t)$ para $t = 1, 2, \dots, T$. A função de probabilidade da variável aleatória Y_t é dada por

$$p(y_t|k, \mu_t) = \frac{\Gamma(y_t + 1/k)}{\Gamma(y_t + 1)\Gamma(1/k)} \frac{(k\mu_t)^{y_t}}{(1 + k\mu_t)^{y_t + 1/k}}, \quad y_t = 0, 1, 2, \dots$$

em que

$$\mu_t = \log(x_t^T \theta_t). \quad (9.1)$$

A média e variância do modelo Binomial Negativo são dados por

$$\begin{aligned} E(Y_t) &= \mu_t, \\ V(Y_t) &= \mu_t + k\mu_t^2. \end{aligned} \quad (9.2)$$

Note que k é um parâmetro estático, que está relacionado com a dispersão do modelo. Nas próximas seções serão ilustrados, através de dados simulados, nossas propostas, McPE e FChAP, de extensão dos algoritmos de McCormick et al. (2012) e Chopin (2007), que permitem a inclusão de parâmetros estáticos no modelo, aplicados ao modelo Binomial Negativo. O apêndice F apresentam uma aplicação do modelo baseado no algoritmo McPE a dados simulados de regressão estática.

9.1.1 Análise dos dados simulados via algoritmo McPE

Para ilustrar o algoritmo de McCormick com parâmetros estáticos (McPE) para o modelo Binomial Negativo, simulou-se uma série Binomial Negativa de tamanho $T = 1000$, na qual houve uma quebra estrutural no ponto $t = 500$. Essa quebra se deu da seguinte forma: para $1 \leq t \leq 500$ fixou-se $\mu_t = 2$ e para $501 \leq t \leq 1000$ então $\mu_t = 1$.

A programação utilizada para a geração da série é dada por

Simulação

```
1 set.seed(12345)
  n=500
3 beta <- rep(0,n)
  beta2 <- rep(0,n)
5 for (t in 1:n){
  beta[t] = 2
7 beta2[t] = 1
  }
9 beta1 <- beta
```

```

    par(mfrow=c(2,1))
11 plot(beta1, type="l")
    plot(beta2, type="l")
13

15 theta1=beta1
    theta2=beta2
17

    par <- rep(0,2*n)
19 lambda <- par
    y=par
21 for (t in 1:length(par)){
    if (t<=length(par)/2){
23     par[t] = theta1[t]
    } else if (t>=length(par)/2){
25     par[t] = theta2[t-n]
    }
27     lambda[t] <- exp(par[t])
    y[t] <- rnbinom(1,mu=lambda[t],size=1)
29 }
    par(mfrow=c(2,1))
31 plot(par, type="l", ylim=c(0,2.5), xlab="Tempo")
    abline(v=n, col="red", lty=2)
33 plot(y, type="l", xlab="Tempo")
    abline(v=n, lty=2, col="red")

```

O dado simulado está ilustrado na Figura 9.1.

A programação utilizada para a análise do conjunto de dados, utilizando o algoritmo McPE, é dada por

Ajuste

```

2 exemplo2 <- NB.uni(Y=y,X=NULL, lambda0=0.95, c=0.95)
    names(exemplo2)
4

    par(mfrow=c(2,1))
6 plot(par, type="l", ylim=c(-1.5,2.5), lty=2, lwd=2, ylab="Theta",
    xlab="Tempo")
8 lines(exemplo2$estimates[-c(1,2),1], col="red")
    lines(exemplo2$estimates[-c(1,2),1]-2*exemplo2$sderror[-c(1,2)], col="blue")
10 lines(exemplo2$estimates[-c(1,2),1]+2*exemplo2$sderror[-c(1,2)], col="blue")
    abline(v=500, lty=2, lwd=2)
12

    plot(exemplo2$estimates[-c(1,2),2], type="l", ylab="dispersão",
14 xlab="Tempo")
    abline(h=1, lty=2, lwd=2, col="red")

```

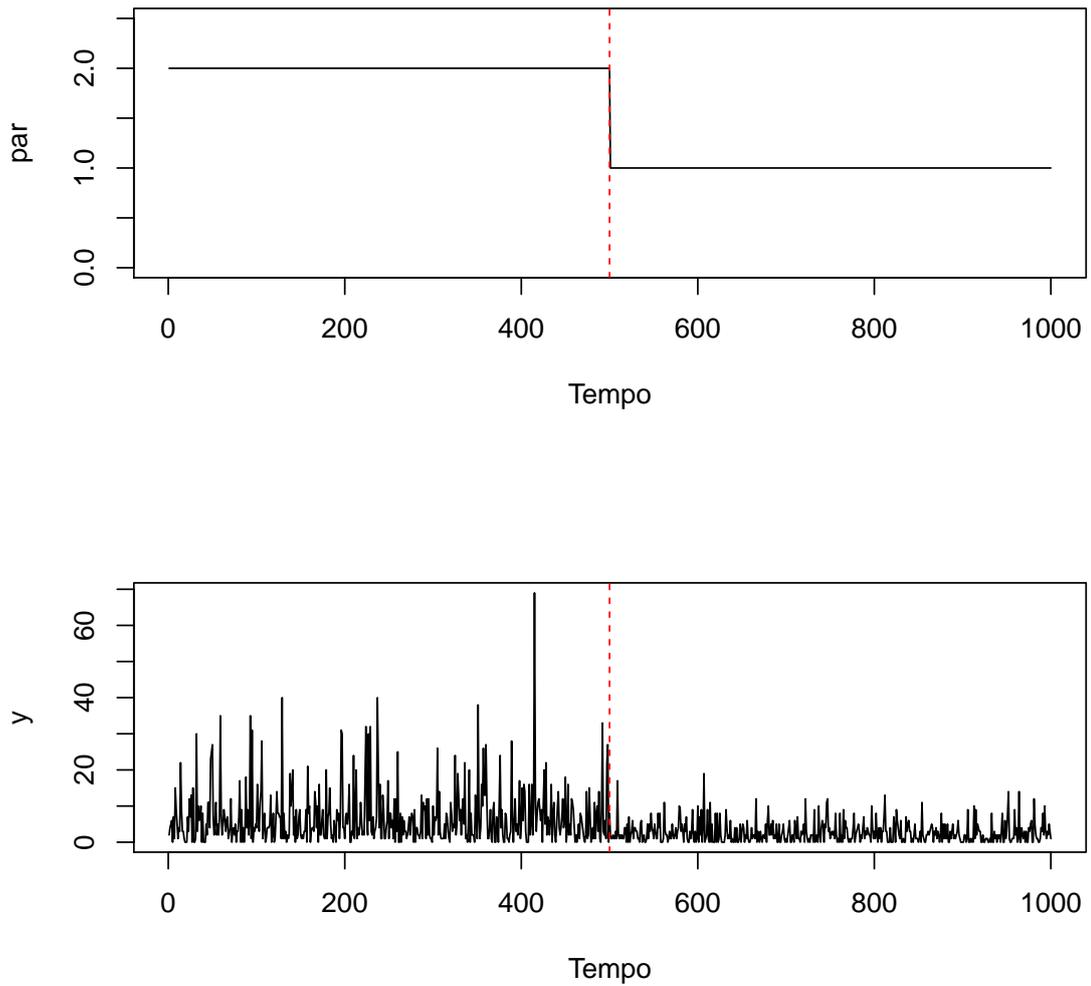


Figura 9.1: De cima para baixo, estado real. Série simulada, y_t , Binomial Negativa com quebra estrutural no ponto $t = 500$. A barra tracejada vertical vermelha indica o ponto da quebra.

Os resultados estão ilustrados na Figura 9.2, em que observa-se que a metodologia proposta é capaz de identificar o ponto de mudança.

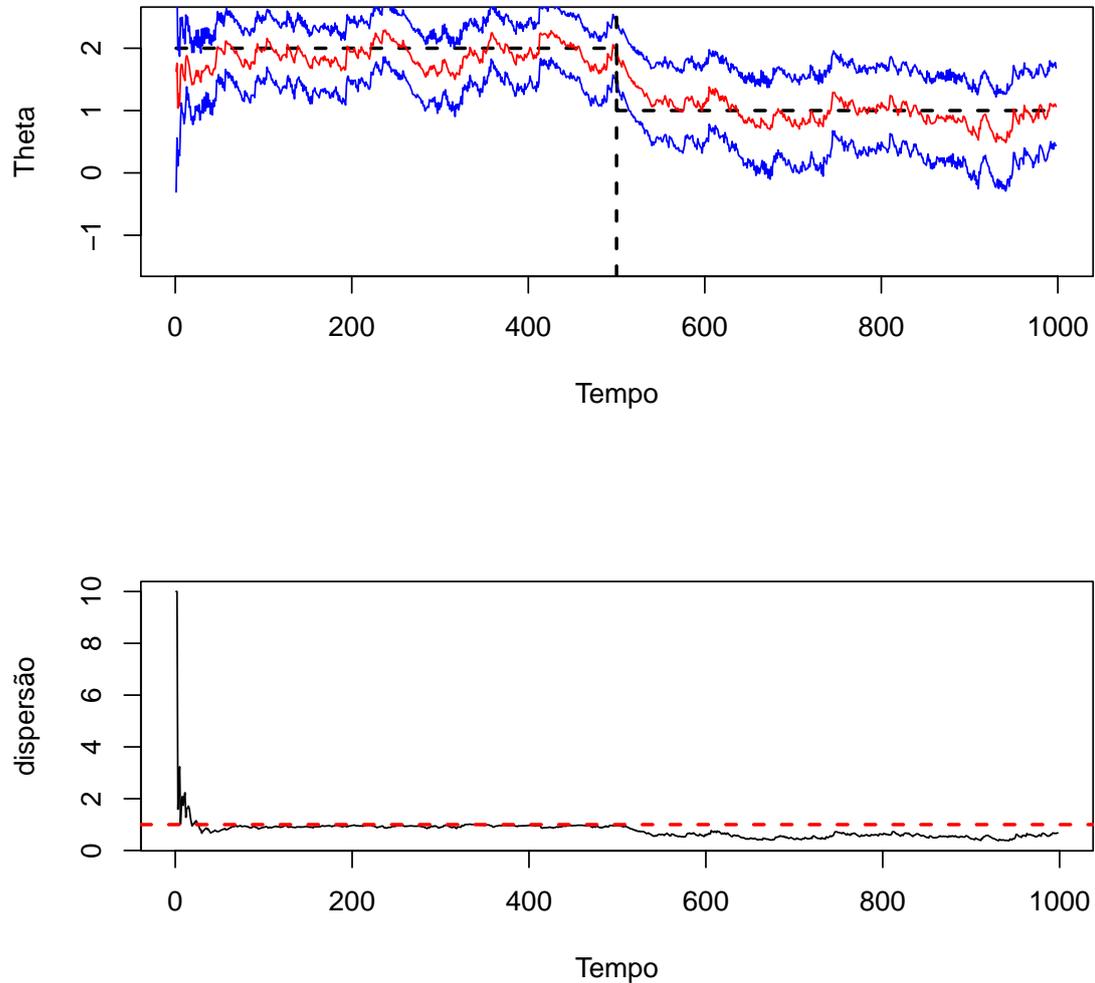


Figura 9.2: Análise da série Binomial Negativa com quebra estrutural no tempo $t = 500$ via algoritmo de McCormick com parâmetros estáticos (McPE)

9.1.2 Análise de dados simulados via algoritmo FChAP

Para ilustrar o uso do filtro de Chopin com aprendizado de partículas (FChAP) para o modelo Binomial Negativo, foi simulada a seguinte base de dados:

- Simulou-se uma série temporal de tamanho $T = 600$, tal que os segmentos são dados por

1. **Segmento 1:** para $0 \leq t \leq 200$ amostra-se de $y_t \sim \text{BN}(k = 2, \mu = 3)$;
2. **Segmento 2:** para $201 \leq t \leq 400$ amostra-se de $y_t \sim \text{BN}(k = 2, \mu = 1)$;
3. **Segmento 3:** para $401 \leq t \leq 600$ amostra-se de $y_t \sim \text{BN}(k = 2, \mu = 3)$.

– Portanto, as mudanças ocorrem em $\tau_1 = 200$ e $\tau_2 = 400$.

A programação utilizada para a geração da série é dada a seguir

Simulação

```

1 #SIMULANDO UM MODELO BINOMIAL NEGATIVO COM PONTOS DE MUDANÇA
  set.seed(456)
3 T = 600
  y <- rep(0,T)
5 d <- y

7 #Regimes (Pontos de mudanças ocorrem em 200 e 400)
  tau = c(0, 200, 400, T)
9 #Valores dos parâmetros em cada regime.
  lambdat = c(3, 1, 3)
11 #simulando a série
  for (j in 1:(length(tau)-1)){
13   for (i in (tau[j]+2):(tau[j+1])){
      y[i] <- rnbinom(1, mu=lambdat[j], size=2)
15   }
  }
17 plot(y, type="l")
  abline(v=c(200, 400), col="red", lty=2)

```

A programação para o ajuste do modelo Binomial Negativo com pontos de mudança, através do filtro FChAP, aos dados simulados, utilizando-se 50000 partículas, é dada a seguir

Ajuste

```

filt_nb <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), tuning=0.975, NParticle
  =50000,
2 family=NB, shiper=(2,1), k0=1.5, gama=0.75)
  smooth_nb <- ChopinSmooth(y=y, particles=filt1$Particles, weight=filt1$Weight)
4
  par(mfrow=c(3,1))
6 plot(y, type="l")
  abline(v=c(200, 400), col="red", lty=2)

```

```

8
#ESTADOS
10 plot(filt_nb$thetaf,lwd=1, lty=1, type="l",
      ylim=c(min(filt_nb$lthetaf), max(filt_nb$uthetaf)), main="filter")
12 lines(filt_nb$lthetaf,lty=2)
   lines(filt_nb$uthetaf,lty=2)
14 abline(v=c(200, 400), lty=2, col="red")

16 #PARÂMETRO DE DISPERSÃO
   plot(filt_nb$dispf,lwd=1, lty=1, type="l", main="filter", ylim=c(0, 4), ylab="k
      ")
18 lines(filt_nb$ldispf,lty=2)
   lines(filt_nb$udispf,lty=2)
20 abline(h=1, lty=2, col="red")

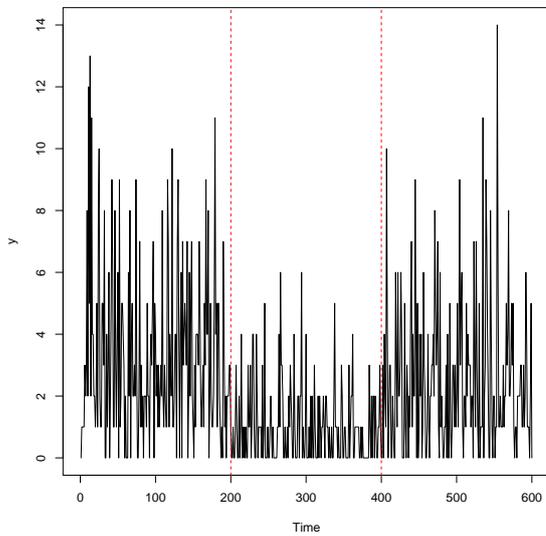
22 #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL NA SÉRIE
   plot(smooth_nb$prob, type="h", ylab="", lwd=2, col="black",
24 xlab="Tempo", main="")
   abline(v=c(200, 400), lty=2, col="red")

```

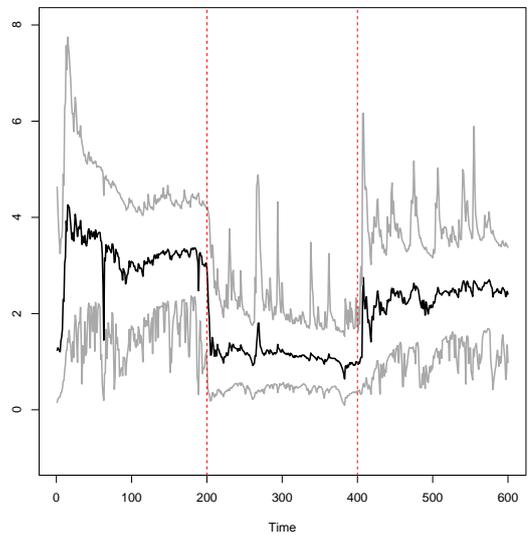
Os resultados, bem como a série simulada, estão ilustrados na Figura 9.3. Nota-se que a média filtrada muda de valor ao mudar o regime da série se aproximando dos valores verdadeiros. Nota-se, também, que a dispersão é um pouco superestimada, mas seu valor encontra-se próximo do verdadeiro. Já as probabilidades a posteriori de mudança são mais evidentes à medida que o tempo se aproxima dos pontos de mudança, logo o modelo acerta quantas e onde ocorreram as mudanças.

Afim de verificar a consistência das estimativas, o filtro foi executado 40 vezes utilizando 1000 partículas, posteriormente construímos os histogramas dos estados em cada segmento. Os resultados estão ilustrados na Figura 9.4. Cada uma das 40 trajetórias foram plotadas em cinza com as trajetórias medianas plotadas em preto.

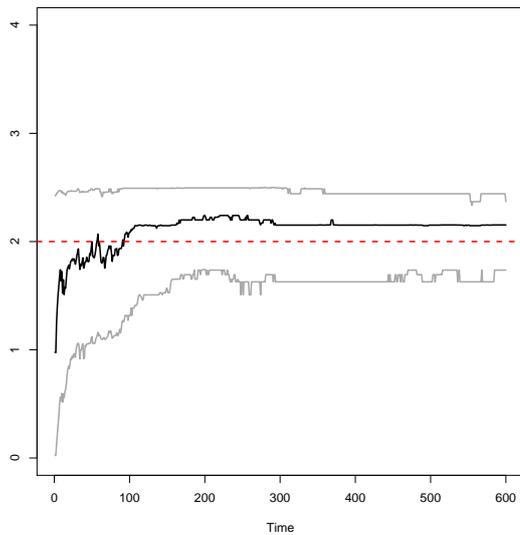
Nota-se que os histogramas estão concentrados próximos dos valores reais, marcados pelas linhas tracejadas em azul. Dessa forma, pode-se ver que uma estratégia que pode ser utilizada para se analisar dados reais via Filtro de Partículas consiste em executar o filtro mais de uma vez e utilizar a mediana das trajetórias como estimativa final para os estados, o que garante uma melhor aproximação para o estado real e para os parâmetros estáticos.



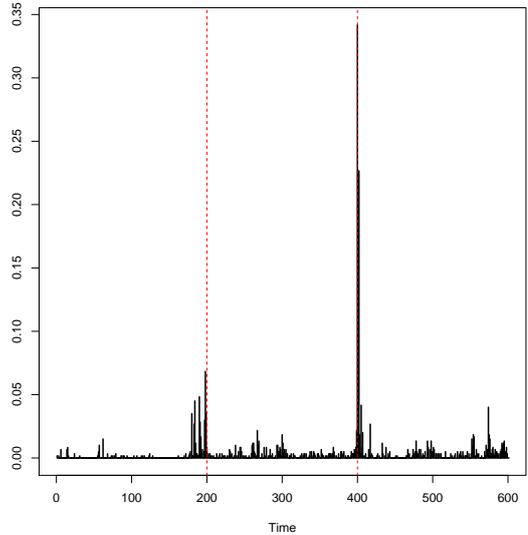
(a) Série simulada y_t



(b) Média Filtrada

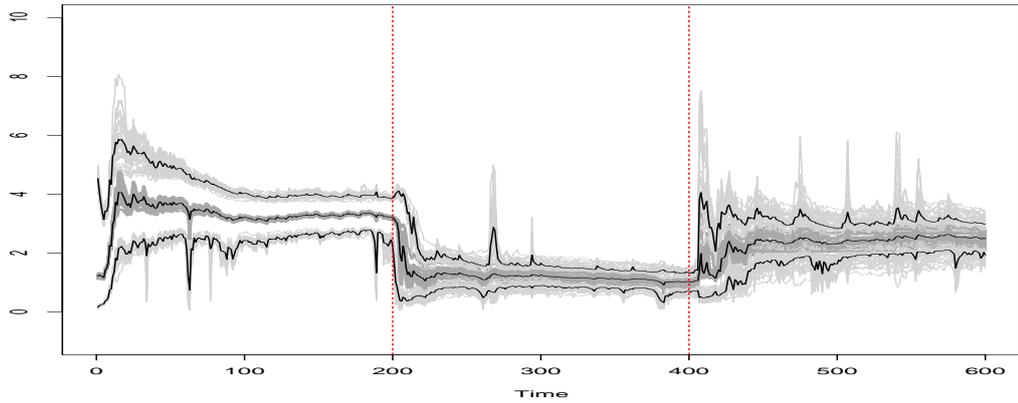


(c) Dispersão Filtrada

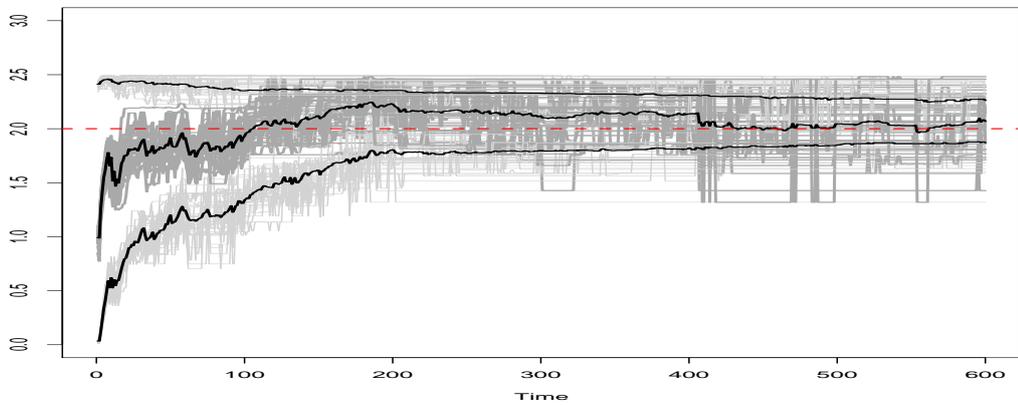


(d) Probabilidade *a posteriori* de mudança

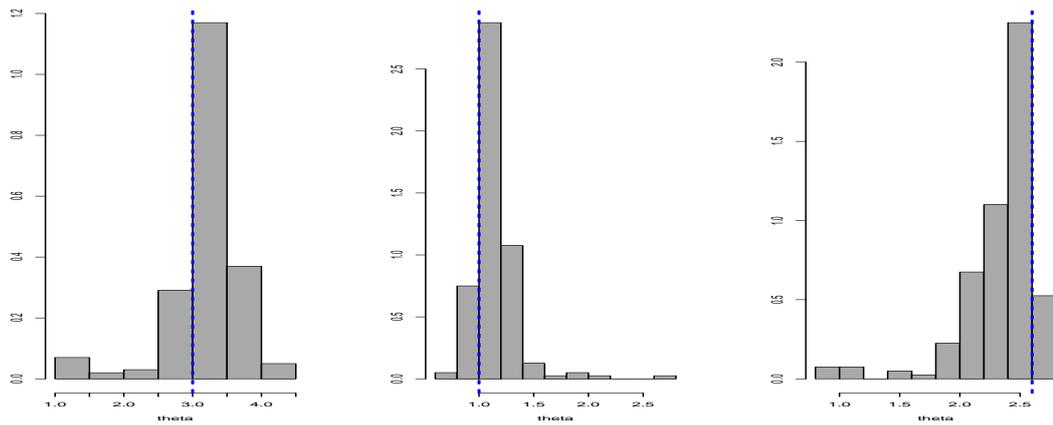
Figura 9.3: (a) Série simulada y_t . (b) Estimativas dos estados com intervalo de credibilidade de 95%. (c) Estimativa do parâmetro de dispersão com intervalo de credibilidade de 95%. (d) Probabilidades *a posteriori* de ocorrer uma mudança no tempo t via algoritmo FChAP. As barras verticais vermelhas mostram os pontos de mudança simulados.



(a) Filtragem replicada para os estados



(b) Filtragem replicada para a dispersão



(c) Densidade empírica dos estados em cada segmento

Figura 9.4: De cima para baixo, Trajetórias replicadas para cada execução do filtro com 1000 partículas em cinza e trajetória mediana dos estados em preto. Trajetórias replicadas para o parâmetro de dispersão. Histograma dos estados em cada segmento, os valores reais são representados pelas linhas tracejadas azul.

9.2 Modelos para dados inflacionados de zeros e pontos de mudança: Modelos ZIP e ZINB

Em dados de contagem é possível que ocorram alguns desvios do modelo Poisson, que são devidos ou a ocorrência excessiva de zeros, ou devidos a algum mecanismo sistemático, e/ou a ocorrência de superdispersão, em que a variância da variável observada excede o valor da média correspondente. No caso do modelo de Poisson, se $x \sim Poisson(\lambda)$, então $E(X) = V(X) = \lambda$. Na formulação de um modelo com excesso de zeros, considere a formulação descrita por Lambert (1992). Em particular, para cada observação i , existem duas possibilidades para o processo de geração dos dados y_i . O resultado de um ensaio de Bernoulli determina qual processo será utilizado. Para a observação i , o processo 1 é escolhido com probabilidade p_i , e o processo 2 com probabilidade $1 - p_i$.

O processo 1 gera contagens nulas, ou zeros estruturais, e, nesse caso, $P(Y_i = 0) = 1$. Já o processo 2 gera as contagens a partir de uma distribuição de Poisson ou Binomial Negativa.

No caso do modelo com excesso de zeros, seja x_i uma covariável associada à observação i , que é utilizada na descrição dos parâmetros do modelo, e defina

$$\begin{aligned}
 P(Y_i = y_i | p_i) &= \begin{cases} P(Y_i = 0 | \text{processo 1})P(\text{processo 1}) + \\ + P(Y_i = 0 | \text{processo 2})P(\text{processo 2}) & \text{se } y_i = 0 \\ P(Y_i = y_i | \text{processo 2})P(\text{processo 2}) & \text{se } y_i > 0. \end{cases} \\
 &= \begin{cases} p_i + (1 - p_i)P(Y_i = 0 | \text{processo 2}) & \text{se } y_i = 0 \\ (1 - p_i)P(Y_i = y_i | \text{processo 2}) & \text{se } y_i > 0. \end{cases} \quad (9.3)
 \end{aligned}$$

Na prática pode-se, também, utilizar covariáveis, Z_i , para explicar a probabilidade p_i associada ao processo gerador do excesso de zeros, utilizando uma função de ligação apropriada para p_i , $g(p_i | z_i) = \gamma' z_i$. Sendo assim, a probabili-

dade associada a $(Y_i = y_i | \gamma_i, z_i)$ é dada por

$$P(Y_i = y_i | \gamma_i, z_i) = \begin{cases} p(\gamma' z_i) + (1 - p(\gamma' z_i))p(0|x_i) & \text{se } y_i = 0 \\ (1 - p(\gamma' z_i))p(y_i|x_i) & \text{se } y_i > 0. \end{cases} \quad (9.4)$$

Nas próximas seções serão ilustrados, através de dados simulados, os algoritmos McPE e FChAP, aplicados aos modelos ZIP e ZINB. O apêndice F apresentam uma aplicação dos modelos baseados no algoritmo McPE a dados simulados de regressão estática.

9.2.1 Modelo Poisson Inflacionado de Zeros - ZIP

O modelo ZIP é um caso especial da estrutura geral apresentada pela equação (9.4) fazendo-se $p(y_i | \lambda_t) = \frac{\lambda_t^{y_i} e^{-\lambda_t}}{y_i!}$, onde $\lambda_t = \log(x_t^T \theta_t)$. Sendo assim, a distribuição de probabilidade é dada por

$$p(y_t | p, \lambda_t) = \begin{cases} p + (1 - p)e^{-\lambda_t}, & \text{se } y_t = 0; \\ (1 - p) \frac{\lambda_t^{y_t} e^{-\lambda_t}}{y_t!}, & \text{se } y_t > 0. \end{cases}$$

A média e a variância do modelo ZIP são dados por

$$\begin{aligned} E(Y_t) &= \lambda_t(1 - p), \\ V(Y_t) &= \lambda_t(1 - p)(1 + \lambda_t p). \end{aligned} \quad (9.5)$$

9.2.1.1 Análise dos dados simulados via algoritmo McPE

Para ilustrar o algoritmo McPE para o modelo Poisson inflacionado de zeros, simulou-se uma série temporal ZIP de tamanho $T = 1000$, na qual foi criada uma quebra estrutural em $t = 500$. Essa quebra se deu da seguinte forma: para $1 \leq t \leq 500$ fixou-se $\lambda_t = -0.5 + 2\frac{t}{T}$. Para $501 \leq t \leq 1000$ fixou-se $\lambda_t = 1$. Fixou-se, também, o valor $p = 0.1$.

A programação utilizada para a geração da série é dada por

Simulação

```
1 set.seed(12345)
  n=500
3 beta <- rep(0,n)
  beta2 <- rep(0,n)
5 for (t in 1:n){
  beta[t] = -0.5+2*t/n
7 beta2[t] = 1
  }
9 beta1 <- beta
  par(mfrow=c(2,1))
11 plot(beta1, type="l")
  plot(beta2, type="l")
13
  theta1=beta1
15 theta2=beta2

17 par <- rep(0,2*n)
  lambda <- par
19 y=par
  for (t in 1:length(par)){
21   if (t<=length(par)/2){
     par[t] = theta1[t]
23   } else if (t>=length(par)/2){
     par[t] = theta2[t-n]
25   }
     lambda[t] <- exp(par[t])
27   y[t] <- rZIP(1,mu=lambda[t],sigma=0.1)
  }
29 par(mfrow=c(2,1))
  plot(par, type="l", ylim=c(-1.5,2.5), ylab="Theta", xlab="tempo")
31 abline(v=n, col="red", lty=2)
  plot(y, type="l", xlab="tempo")
33 abline(v=n, col="red", lty=2)
```

A série simulada está ilustrada na Figura 9.5.

A programação utilizada para a análise da série é

Ajuste

```
1 exemplo1 <- ZIP.uni(Y=y,X=NULL, lambda0=0.95, c=0.95)
  names(exemplo1)
3
  par(mfrow=c(2,1))
5 plot(par, type="l", ylim=c(-1.5,2.5), lty=2, lwd=2, ylab="Theta",
  xlab="Tempo")
7 lines(exemplo1$estimates[-c(1,2),1], col="red")
```

```

lines(exemplo1$estimates[-c(1,2),1]-2*exemplo1$sderror[-c(1,2)],
9 col="blue")
lines(exemplo1$estimates[-c(1,2),1]+2*exemplo1$sderror[-c(1,2)],
11 col="blue")
abline(v=500, lty=2, lwd=2)
13
plot(exemplo1$estimates[-c(1,2),2], type="l", ylab="Probabilidade",
15 xlab="Tempo", ylim=c(0,1))
abline(h=0.1, lty=2, lwd=2, col="red")

```

Os resultados do ajuste estão ilustrados na Figura 9.6. Em que observa-se que o algoritmo McPE é capaz de captar a mudança de regime.

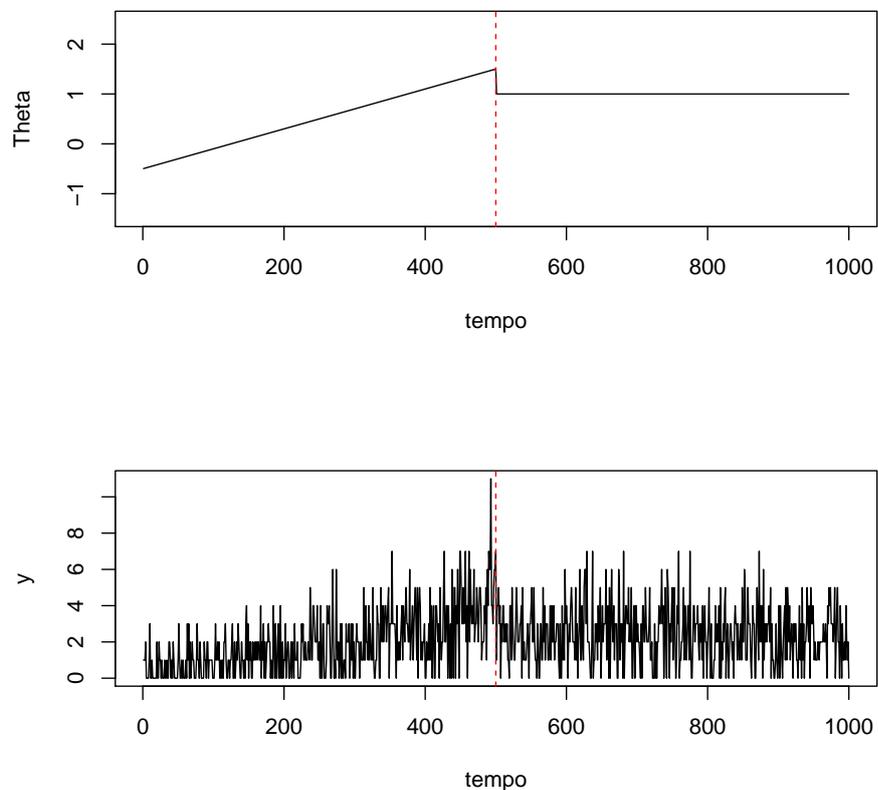


Figura 9.5: De cima para baixo, estado real, série ZIP simulada y_t . A barra tracejada vertical vermelha indica o ponto da quebra.

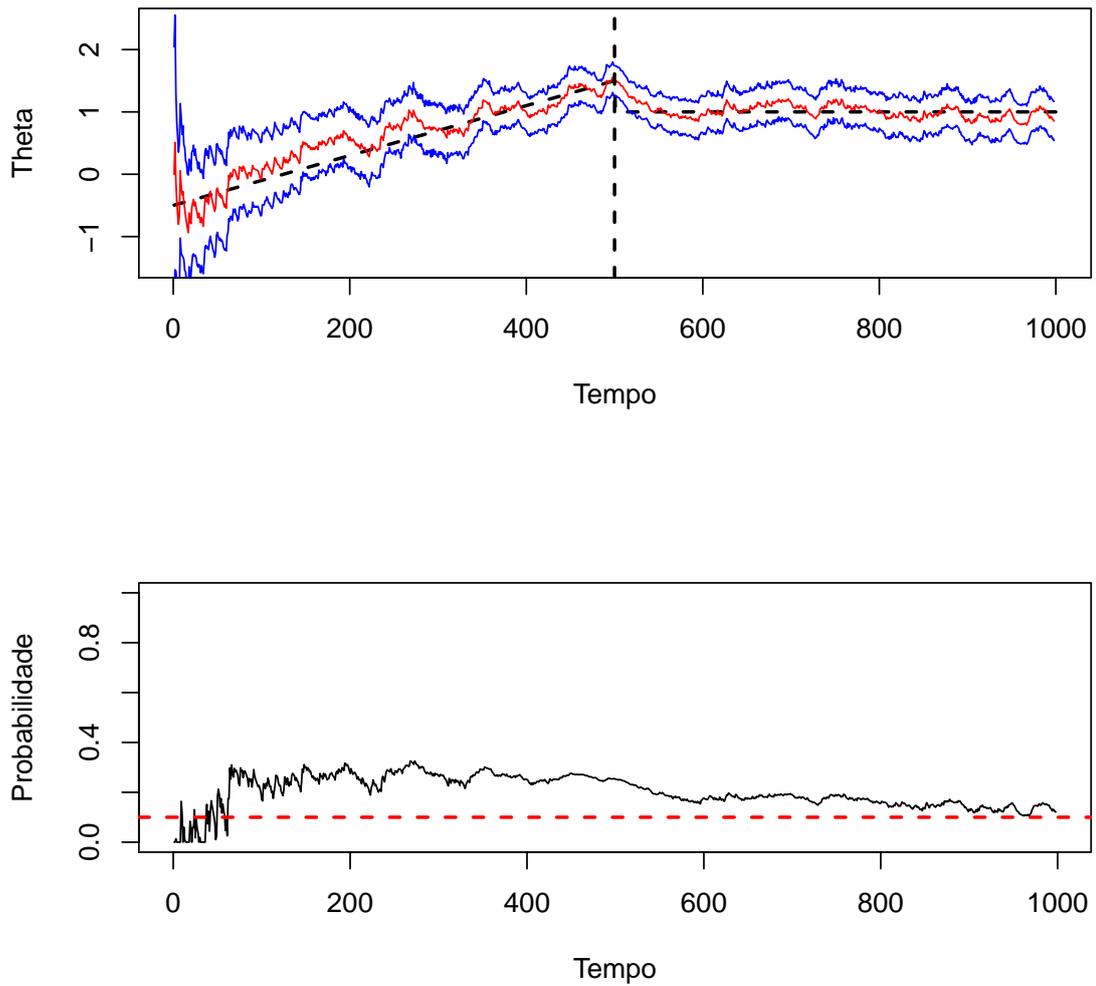


Figura 9.6: De cima para baixo, estados estimados (linha vermelha), via algoritmo McPE, com intervalo de confiança de 95% (linhas azuis), estimativa do parâmetro estático p .

9.2.1.2 Análise de dados simulados via algoritmo FChAP

Para ilustrar o filtro FChAP para o modelo ZIP, foi simulada a seguinte base de dados:

- Simulou-se uma série temporal de tamanho $T = 600$, tal que os segmentos são dados por
 1. **Segmento 1:** para $0 \leq t \leq 200$ amostra-se de $y_t \sim \text{ZIP}(p = 0.2, \mu = 3)$;
 2. **Segmento 2:** para $201 \leq t \leq 400$ amostra-se de $y_t \sim \text{ZIP}(p = 0.2, \mu = 1)$;
 3. **Segmento 3:** para $401 \leq t \leq 600$ amostra-se de $y_t \sim \text{ZIP}(p = 0.2, \mu = 3)$.
- Portanto, as mudanças ocorrem em $\tau_1 = 200$ e $\tau_2 = 400$.

A programação utilizada para a geração da série é dada por

Simulação

```
require(gamlss.dist)
2 #SIMULANDO UM MODELO POISSON INFLACIONADO DE ZEROS COM PONTOS DE MUDANÇA
  set.seed(456)
4 T = 600
  y <- rep(0,T)
6 d <- y
  #Regimes (Pontos de mudanças ocorrem em 200, 400)
8 tau = c(0, 200, 400, T)
  #Valores dos parâmetros em cada regime.
10 lambdat = c(3, 1, 3)
  #simulando a série
12 for (j in 1:(length(tau)-1)){
  for (i in (tau[j]+2):(tau[j+1])){
14   y[i] <- rZIP(1, mu=lambdat[j], sigma=0.2)
  }
16 }

18 plot(y, type="l")
  abline(v=c(200, 400), col="red", lty=2)
```

A programação para o ajuste do modelo aos dados simulados, utilizando 50000 partículas e o algoritmo FChAP é dada por:

Ajuste

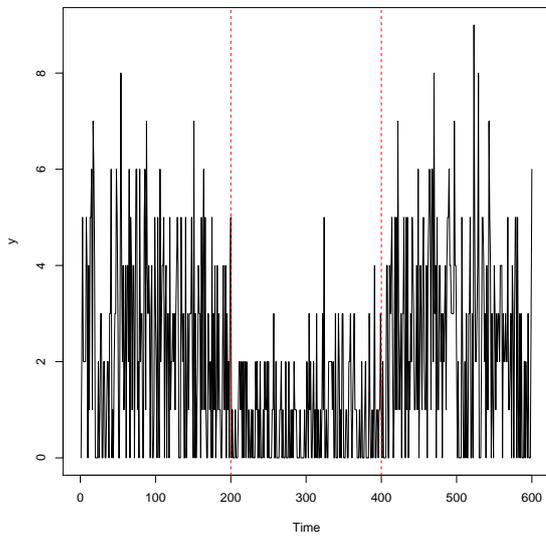
```
1 filt_ZIP <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), tuning=0.975,
  NParticle=50000,
3 family=ZIP, shiper=(3,3,2,1), p0=0.15, gama=0.75)
  smooth_ZIP <- ChopinSmooth(y=y, particles=filt1$Particles, weight=filt1$Weight)
5
  par(mfrow=c(3,1))
7 plot(y, type="l")
  abline(v=c(200, 400), col="red", lty=2)
9
  #ESTADOS
11 plot(filt_ZIP$thetaf,lwd=1, lty=1, type="l",
  ylim=c(min(filt_ZIP$lthetaf), max(filt_ZIP$uthetaf)), main="filter")
13 lines(filt_ZIP$lthetaf,lty=2)
  lines(filt_ZIP$uthetaf,lty=2)
15 abline(v=c(200, 400), lty=2, col="red")

17 #PROBABILIDADE DE OCORRER ZERO
  plot(filt_ZIP$pf,lwd=1, lty=1, type="l", main="filter", ylim=c(0, 1),
19 ylab="p")
  lines(filt_ZIP$pf,lty=2)
21 lines(filt_ZIP$pf,lty=2)
  abline(h=1, lty=2, col="red")
23
  #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL NA SÉRIE
25 plot(smooth_ZIP$prob, type="h", ylab="Posteriori da mudança", lwd=2, col="black
  ",
  xlab="Tempo", main="Algoritmo Chopin (2006)")
27 abline(v=c(200, 400), lty=2, col="red")
```

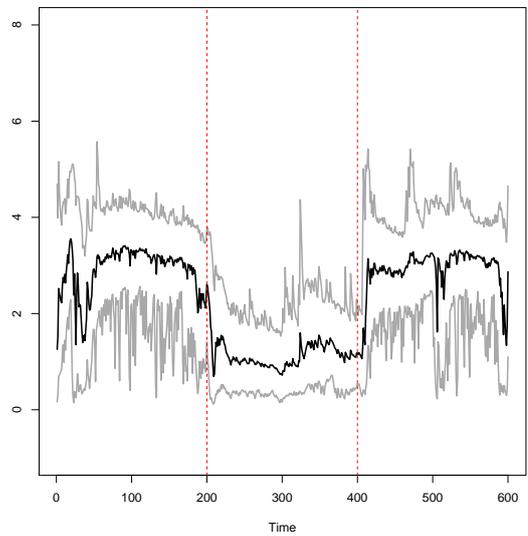
Os resultados, bem como a série simulada, estão ilustrados na Figura 9.7. Nota-se que a média filtrada muda de valor ao mudar o regime da série se aproximando dos valores verdadeiros. Nota-se, também, que a probabilidade de zero encontra-se bem próximo do verdadeiro. Já as probabilidades a posteriori de mudança são mais evidentes à medida que o tempo se aproxima dos pontos de mudança, logo o modelo acerta quantas e onde ocorreram as mudanças.

Afim de verificar a consistência das estimativas, o filtro foi executado 40 vezes utilizando 1000 partículas, posteriormente construímos os histogramas dos estados em cada segmento. Os resultados estão ilustrados na Figura 9.8. Cada uma das 40 trajetórias foram plotadas em cinza com as trajetórias medianas plotadas em preto.

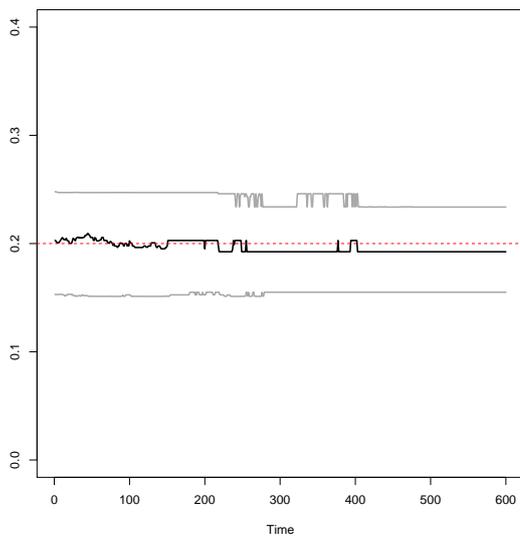
Nota-se que os histogramas estão concentrados próximos dos valores reais, marcados pelas linhas tracejadas em azul. Dessa forma, pode-se ver que uma estratégia que pode ser utilizada para se analisar dados reais via Filtro de Partículas consiste em executar o filtro mais de uma vez e utilizar a mediana das trajetórias como estimativa final para os estados, o que garante uma melhor aproximação para o estado real e para os parâmetros estáticos.



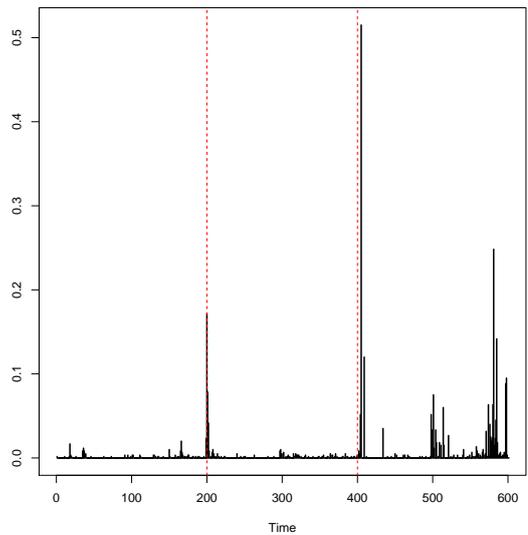
(a) Série simulada y_t



(b) Média filtrada

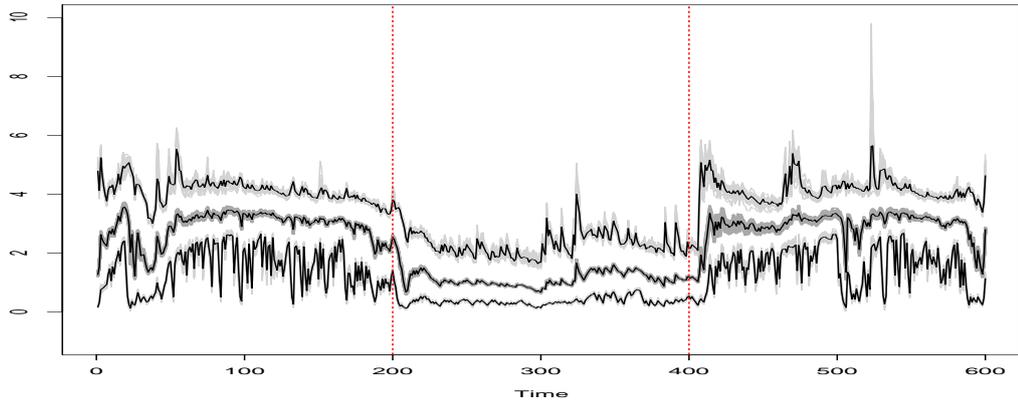


(c) Parâmetro p filtrado

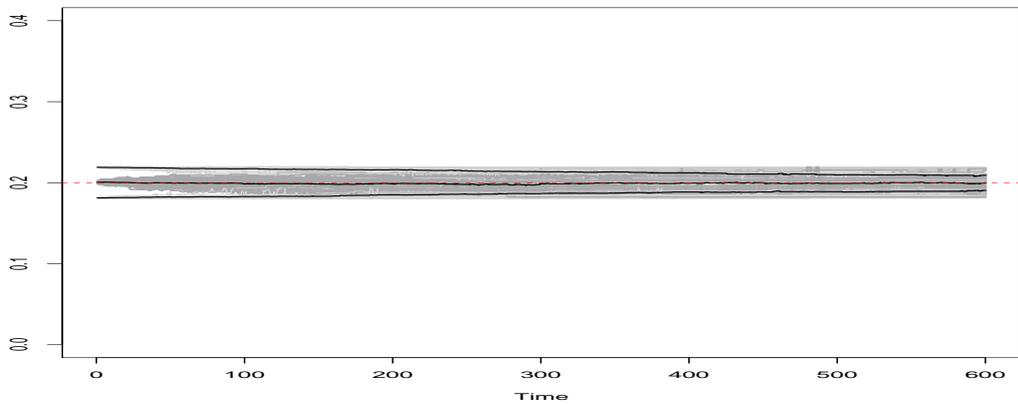


(d) Probabilidade *a posteriori* de ocorrer mudança

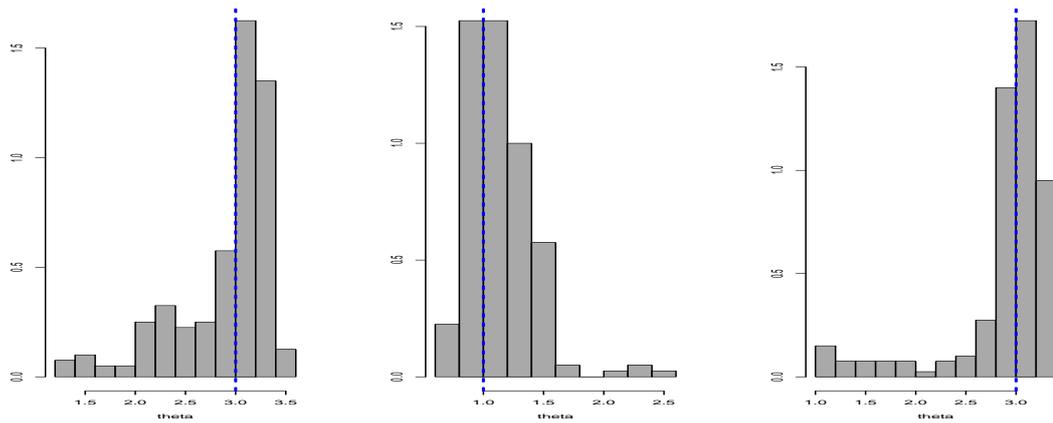
Figura 9.7: (a) Série simulada y_t . (b) Estimativas dos estados com intervalo de credibilidade de 95%. (c) Estimativa do parâmetro estático p com intervalo de credibilidade de 95%. (d) Probabilidades *a posteriori* de ocorrer uma mudança no tempo t via algoritmo FChAP. As barras verticais vermelhas mostram os pontos de mudança simulados.



(a) Filtragem replicada para os estados



(b) Filtragem replicada para o parâmetro p



(c) Densidade empírica dos estados em cada segmento

Figura 9.8: De cima para baixo, trajetórias replicadas para cada execução do filtro com 1000 partículas em cinza e trajetória mediana dos estados em preto. Trajetórias replicadas para a probabilidade de zero. Histograma dos estados em cada segmento, os valores reais são representados pelas linhas tracejadas azul.

9.2.2 Modelo Binomial Negativo inflacionado de Zeros - ZINB

O modelo ZINB é um caso especial da estrutura geral descrita pela equação (9.4) em que $p(y_i|x_i) = \frac{\Gamma(y_t+1/k)(k\mu_t)^{y_t}}{\Gamma(y_t+1)\Gamma(1/k)(1+k\mu_t)^{y_t+1/k}}$ e $\mu_t = \log(x_t^T \theta_t)$. Sendo assim, a distribuição de probabilidade para o modelo ZINB é dada por

$$p(y_t|p, k, \mu_t) = \begin{cases} p + (1-p)(1+k\mu_t)^{-1/k}, & \text{se } y_t = 0; \\ (1-p) \frac{\Gamma(y_t+1/k)(k\mu_t)^{y_t}}{\Gamma(y_t+1)\Gamma(1/k)(1+k\mu_t)^{y_t+1/k}}, & \text{se } y_t > 0. \end{cases}$$

A média e a variância do modelo ZINB são dados por

$$\begin{aligned} E(Y_t) &= \mu_t(1-p), \\ V(Y_t) &= \mu_t(1-p)(1 + \mu_t(p+k)). \end{aligned} \tag{9.6}$$

9.2.2.1 Análise de dados simulados via algoritmo McPE

Para ilustrar o uso do algoritmo McPE para o modelo Binomial Negativo inflacionado de zeros, simulou-se um base de tamanho $T = 1000$, a partir do modelo ZINB, na qual foi criada uma quebra estrutural $t = 500$. Essa quebra se deu da seguinte forma, para $1 \leq t \leq 500$ fixou-se $\lambda_t = -0.5 + 1\frac{t}{T}$. Para $501 \leq t \leq 1000$ fixou-se $\lambda_t = 1$. Fixou-se, também, o valor $p = 0.1$ e $k = 1$.

A programação utilizada para a geração da série é dada a seguir

Simulação

```
1 set.seed(123)
  n=500
3 beta <- rep(0,n)
  beta2 <- rep(0,n)
5 for (t in 1:n){
    beta[t] = -0.5+1*t/n
7   beta2[t] = 1
  }
9 beta1 <- beta
  par(mfrow=c(2,1))
11 plot(beta1, type="l")
  plot(beta2, type="l")
13
  theta1=beta1
15 theta2=beta2

17 par <- rep(0,2*n)
  lambda <- par
19 y=par
  for (t in 1:length(par)){
21   if (t<=length(par)/2){
    par[t] = theta1[t]
23   } else if (t>=length(par)/2){
    par[t] = theta2[t-n]
25   }
    lambda[t] <- exp(par[t])
27   y[t] <- rzinb(1,k=1,lambda=lambda[t],omega=0.1)
  }
29 par(mfrow=c(2,1))
  plot(par, type="l", ylim=c(-1.5,2.5), ylab="Theta", xlab="tempo")
31 abline(v=n, col="red", lty=2)
  plot(y, type="l", xlab="tempo")
33 abline(v=n, col="red", lty=2)
```

A série simulada ZINB com ponto de mudança está ilustrada na Figura 9.9.

A programação utilizada para o ajuste do modelo é dada a seguir

Ajuste

```
1 exemplo1 <- ZINB.uni(Y=y,X=NULL, lambda0=1, autotune=TRUE)
  names(exemplo1)
3
  par(mfrow=c(2,1))
5 plot(par, type="l", ylim=c(-1.5,2.5), lty=2, lwd=2, ylab="Theta",
  xlab="Tempo")
7 lines(exemplo1$estimates[-c(1,2),1], col="red")
```

```

lines(exemplo1$estimates[-c(1,2),1]-2*exemplo1$sderror[-c(1,2)],
9 col="blue")
lines(exemplo1$estimates[-c(1,2),1]+2*exemplo1$sderror[-c(1,2)],
11 col="blue")
abline(v=500, lty=2, lwd=2)
13
plot(exemplo1$estimates[,3], type="l", ylab="p", xlab="Tempo", ylim=c(0,1))
15 abline(h=0.1, lty=2, lwd=2, col="red")

```

O resultado da análise está ilustrado na Figura 9.10. Nesse exemplo, utilizando o algoritmo McPE, o parâmetro de dispersão, k , foi atualizado utilizando o parâmetro de desconto $\delta = 0.9999$, ou seja, no tempo t o parâmetro de dispersão é atualizado como $k_t = k_{t-1}/\delta$.

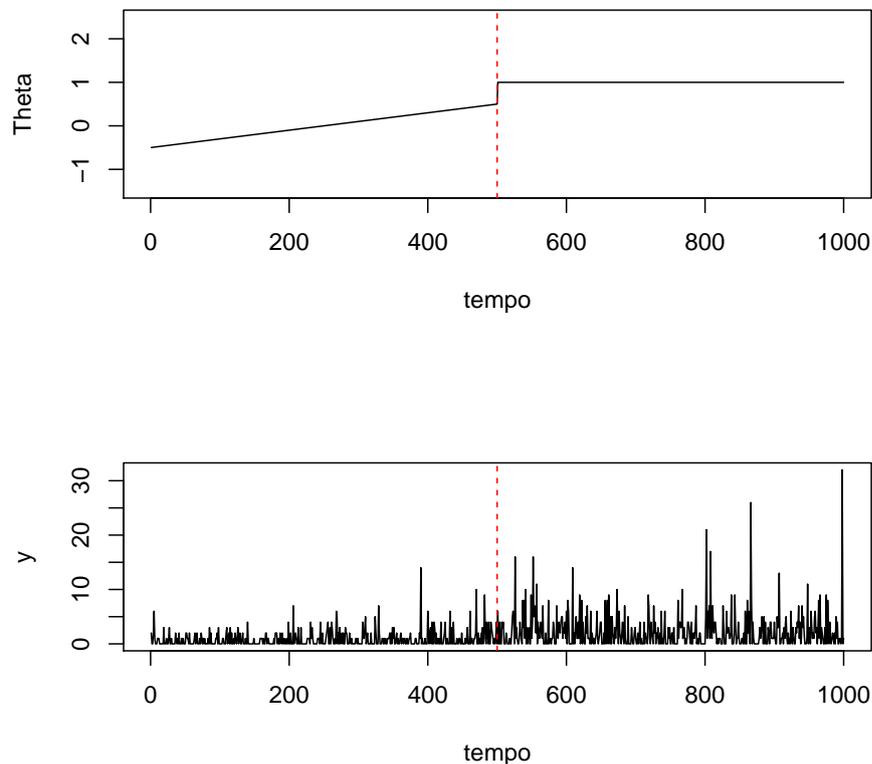


Figura 9.9: De cima para baixo, estado real, série ZINB simulada y_t . A barra tracejada vertical vermelha indica o ponto da quebra.

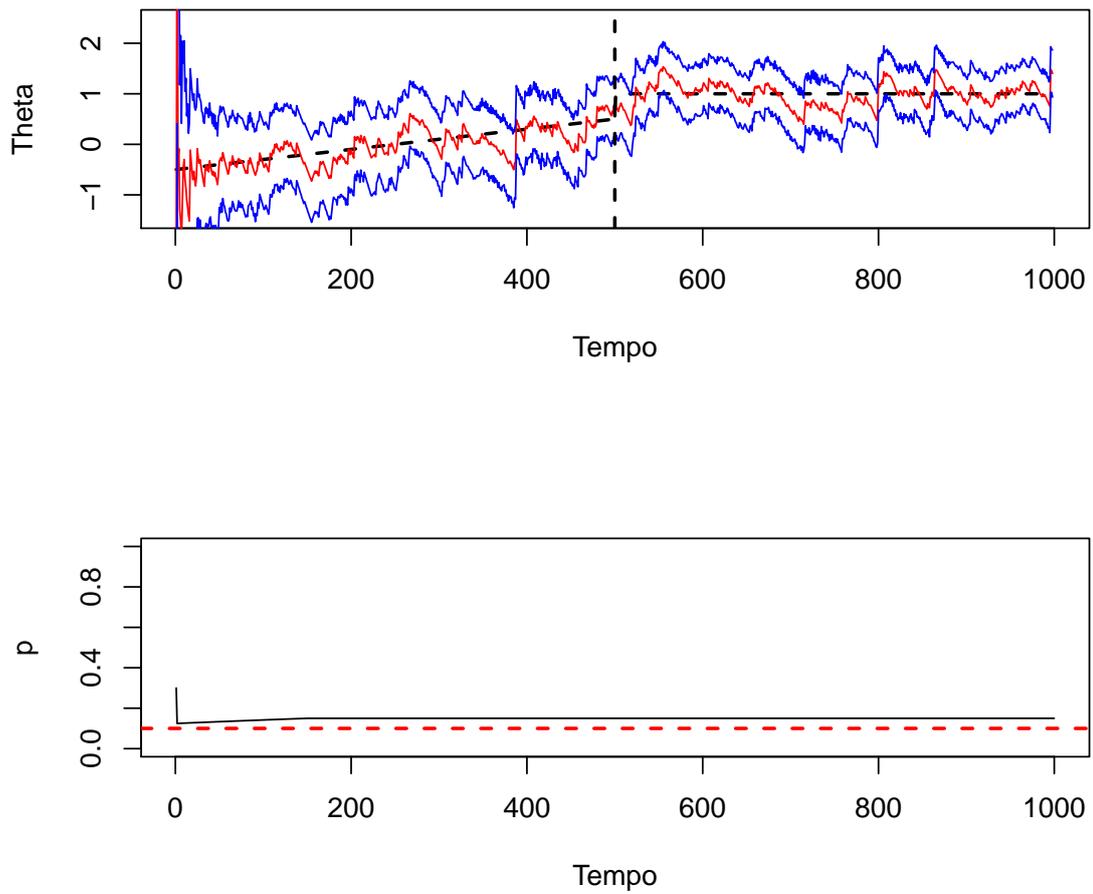


Figura 9.10: De cima para baixo, estados estimados (linha vermelha), via algoritmo McPE, com intervalo de confiança de 95% (linhas azuis), estimativa do parâmetro estático p .

9.2.2.2 Análise de dados simulados via algoritmo FChAP

Para ilustrar o modelo, novamente simulou-se a seguinte base de dados:

- Simulou-se uma série temporal de tamanho $T = 600$, tal que para os segmentos são dados por
 1. **Segmento 1:** para $0 \leq t \leq 200$ amostra-se de $y_t \sim \text{ZINB}(k = 2, p = 0.2, \mu = 3)$;
 2. **Segmento 2:** para $201 \leq t \leq 400$ amostra-se de $y_t \sim \text{ZINB}(k = 2, p = 0.2, \mu = 1)$;
 3. **Segmento 3:** para $401 \leq t \leq 600$ amostra-se de $y_t \sim \text{ZINB}(k = 2, p = 0.2, \mu = 3)$.
- Portanto, as mudanças ocorrem em $\tau_1 = 200$ e $\tau_2 = 400$.

A programação utilizada para a geração da série é dada a seguir:

Simulação

```
1 require(ZIM)
  #SIMULANDO UM MODELO ZINB COM PONTOS DE MUDANÇA
3 T = 600
  y <- rep(0,T)
5 d <- y

7 #Regimes (Pontos de mudanças ocorrem em 200, 400)
  tau = c(0, 200, 400, T)
9

11 #Valores dos parâmetros em cada regime.
  lambdat = c(3, 1, 3)
13
  #simulando
15 for (j in 1:(length(tau)-1)){
  for (i in (tau[j]+2):(tau[j+1])){
17   y[i] <- rzinb(1, k=2,lambda=lambdat[j], omega=0.2)
  }
19 }
```

A programação para o ajuste do modelo aos dados simulados, utilizando $H = 50000$ partículas, é dada a seguir:

Ajuste

```
filt_ZINB <- ChopinFilter(y=y, p=1e^-6, hiper=(2,1), tuning=0.975,
2 NParticle=50000,
  family=ZINB, shiper=(3,3,2,1), p0=0.15, gama=0.75)
4 smooth_ZINB <- ChopinSmooth(y=y, particles=filt1$Particles, weight=filt1$Weight
  )

6 par(mfrow=c(3,1))
  plot(y, type="l")
8 abline(v=c(200, 400), col="red", lty=2)

10 #ESTADOS
  plot(filt_ZINB$thetaf,lwd=1, lty=1, type="l",
12 ylim=c(min(filt_ZINB$lthetaf), max(filt_ZINB$uthetaf)), main="filter")
  lines(filt_ZINB$lthetaf,lty=2)
14 lines(filt_ZINB$uthetaf,lty=2)
  abline(v=c(200, 400), lty=2, col="red")
16

18 #PROBABILIDADE DE OCORRER ZERO
  plot(filt_ZINB$pf,lwd=1, lty=1, type="l", main="filter", ylim=c(0, 1), ylab="p"
  )
20 lines(filt_ZINB$pf,lty=2)
  lines(filt_ZINB$pf,lty=2)
22 abline(h=0.1, lty=2, col="red")

24
  #PARÂMETRO DE DISPERSÃO
26 plot(filt_ZINB$dispf,lwd=1, lty=1, type="l", main="filter", ylim=c(0, 8),
  ylab="k")
28 lines(filt_ZINB$dispf,lty=2)
  lines(filt_ZINB$dispf,lty=2)
30 abline(h=2, lty=2, col="red")

32 #PROBABILIDADE A POSTERIORI DE OCORRER MUDANÇA ESTRUTURAL NA SÉRIE
  plot(smooth_ZINB$prob, type="h", ylab=" ", lwd=2,
34 col="black",
  xlab="Tempo", main=" ")
36 abline(v=c(200, 400), lty=2, col="red")
```

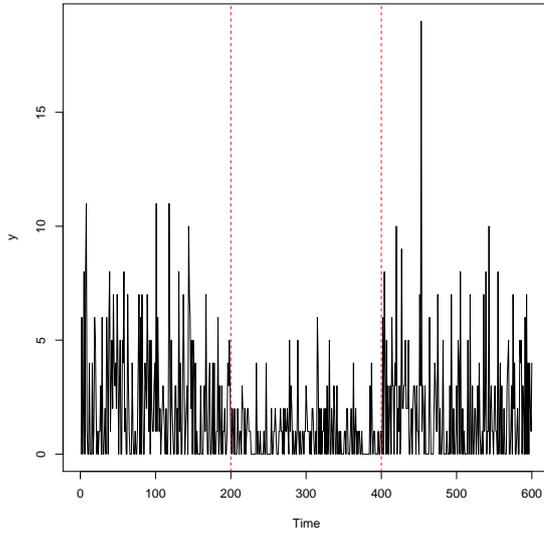
Os resultados, bem como a série simulada, estão ilustrados nas Figuras 9.11 e 9.12. Nota-se que a média filtrada muda de valor ao mudar o regime da série se aproximando dos valores verdadeiros. Nota-se, também, que os parâmetros estáticos, probabilidade de zero e dispersão, encontram-se bem próximos dos verdadeiros. Já as probabilidades a posteriori de mudança são mais evidentes à

medida que o tempo se aproxima dos pontos de mudança, logo o modelo acerta quantas e onde ocorreram as mudanças.

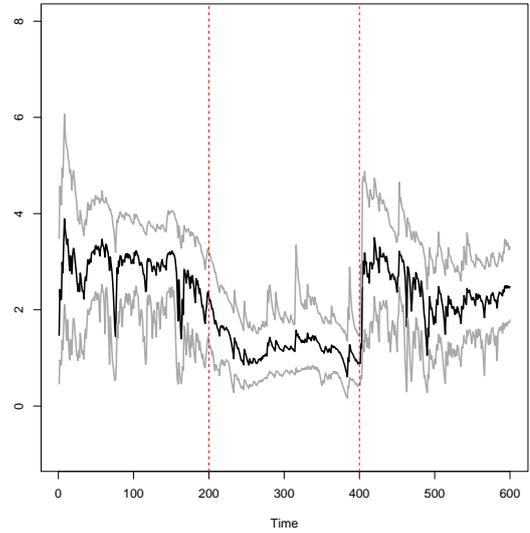
Afim de verificar a consistência das estimativas, o filtro foi executado 40 vezes utilizando 1000 partículas, posteriormente construímos os histogramas dos estados em cada segmento. Os resultados estão ilustrados na Figura 9.13. Cada uma das 40 trajetórias foram plotadas em cinza com as trajetórias medianas plotadas em preto.

Nota-se que os histogramas estão concentrados próximos dos valores reais, marcados pelas linhas tracejadas em azul. Dessa forma, pode-se ver que uma estratégia que pode ser utilizada para se analisar dados reais via Filtro de Partículas consiste em executar o filtro mais de uma vez e utilizar a mediana das trajetórias como estimativa final para os estados, o que garante uma melhor aproximação para o estado real e para os parâmetros estáticos.

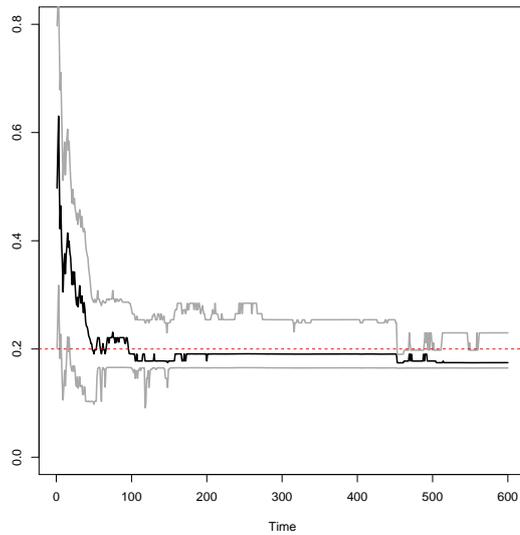
De posse do aprendizado adquirido das análises de dados simulados desse capítulo e dos Capítulos 6 e 7, os algoritmos desenvolvidos nesta dissertação foram aplicados em 5 séries temporais reais afim de validar a teoria estudada até aqui.



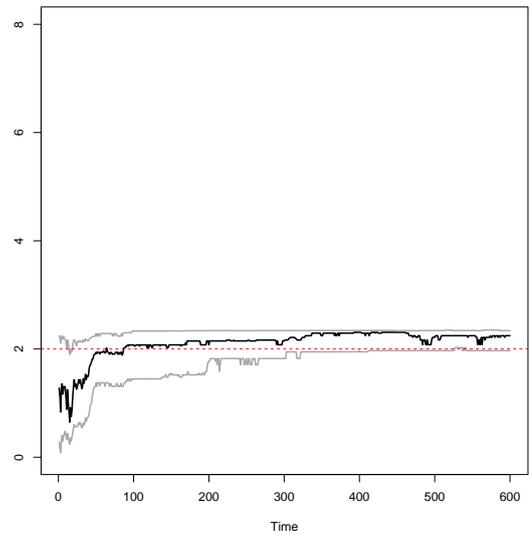
(a) Série simulada y_t



(b) Média filtrada



(c) Parâmetro p filtrado



(d) Dispersão filtrada

Figura 9.11: (a) Série simulada y_t . (b) Estimativas dos estados com intervalo de credibilidade de 95%. (c) Estimativa do parâmetro estático p com intervalo de credibilidade de 95%. (d) Estimativa do parâmetro de dispersão com intervalo de credibilidade de 95%. As barras verticais vermelhas mostram os pontos de mudança simulados.

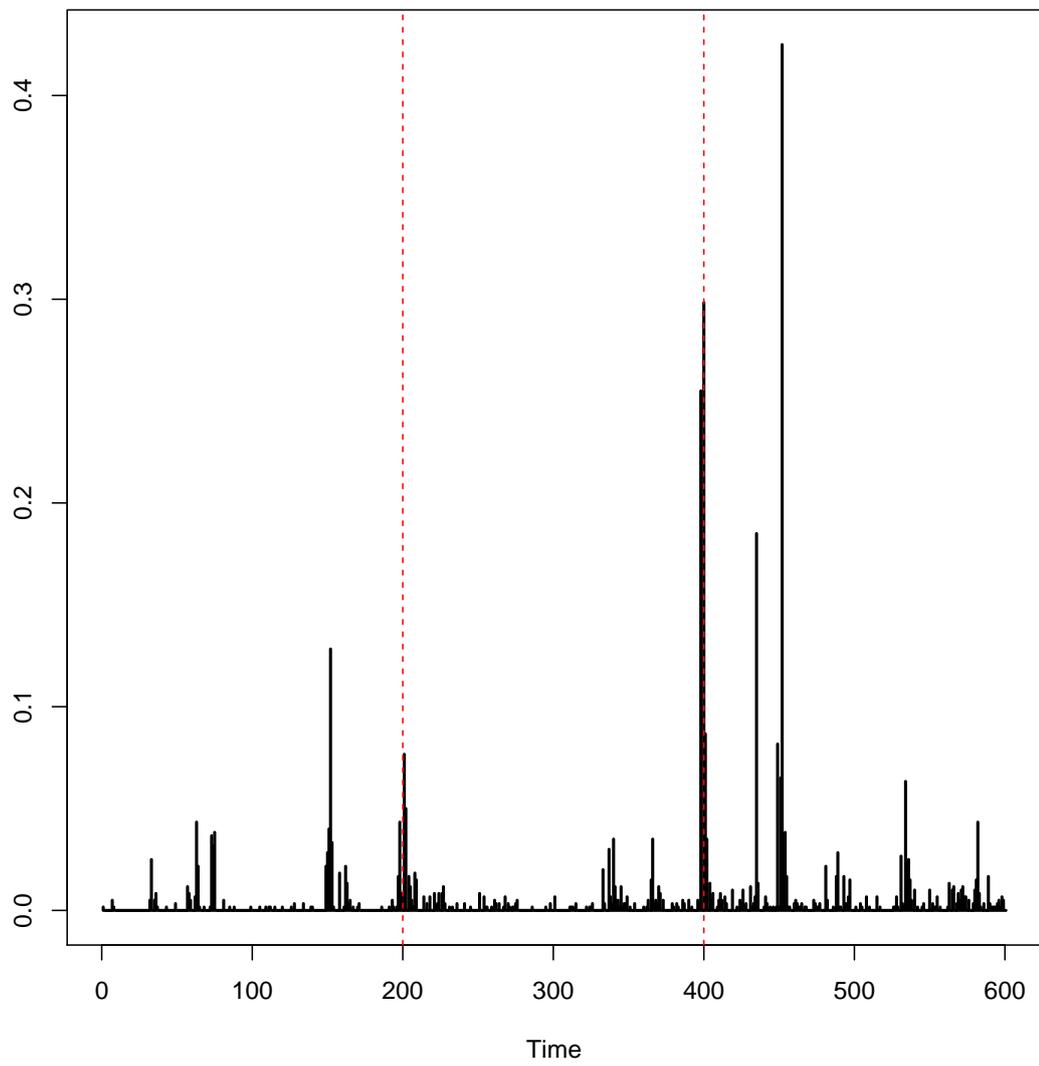
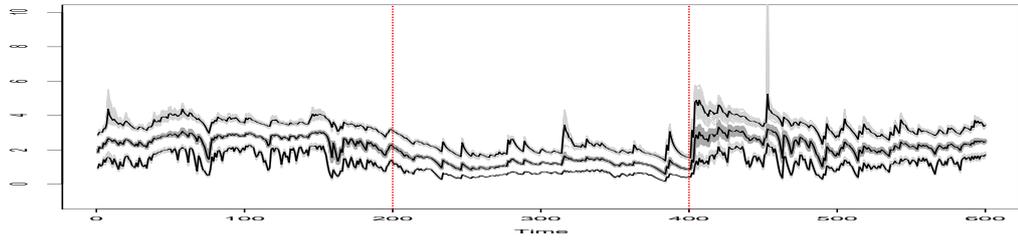
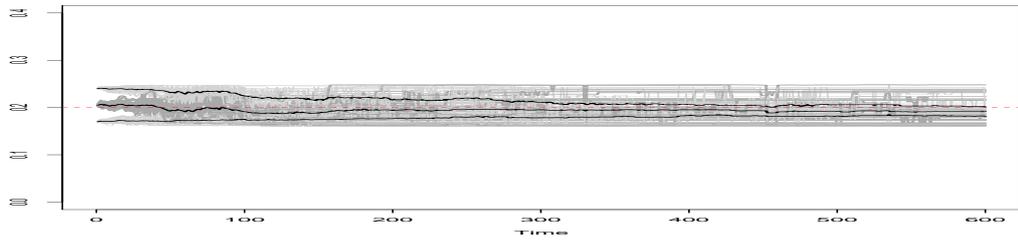


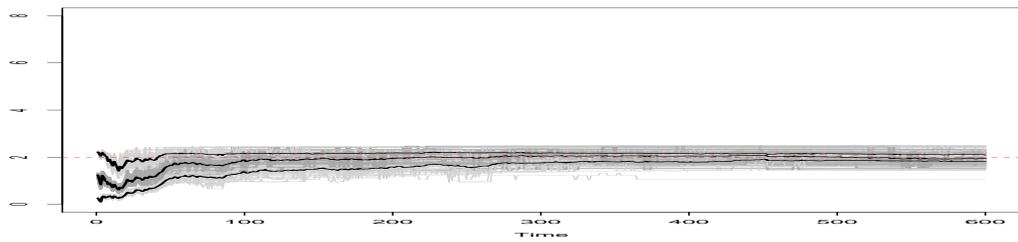
Figura 9.12: Probabilidade a *posteriori* de ocorrer mudança.



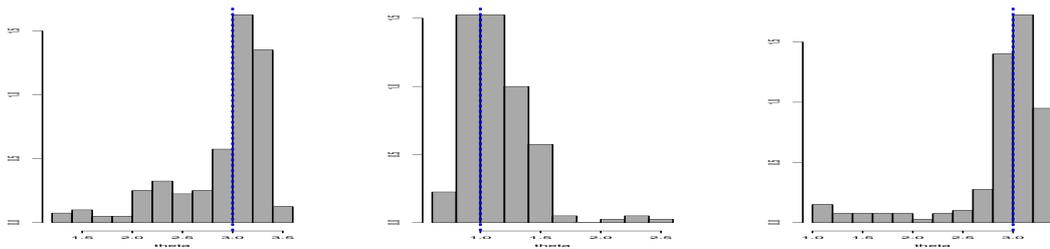
(a) Filtragem replicada para os estados



(b) Filtragem replicada para o parâmetro p



(c) Filtragem replicada para a dispersão



(d) Densidade empírica dos estados em cada segmento

Figura 9.13: De cima para baixo, trajetórias replicadas para cada execução do filtro com 1000 partículas em cinza e trajetória mediana dos estados em preto. Trajetórias replicadas para a dispersão. Trajetórias replicadas para a probabilidade de zero. Histograma dos estados em cada segmento, os valores reais são representados pelas linhas tracejadas azul.

Capítulo 10

Aplicações a dados reais

Neste capítulo ajustaram-se os modelos propostos nos Capítulos 6, 7, 8 e 9 a cinco conjuntos de dados reais. Na Seção 10.1 aplicaram-se os modelos de regressão dinâmica utilizando o algoritmo de McCormick (2012) e o Algoritmo de McCormick com parâmetros estáticos (McPE), utilizando as distribuições Poisson, Binomial Negativa, Poisson Inflacionada de Zeros e Binomial Negativa Inflacionada de Zeros, aos dados relacionados ao número de casos de Sífilis no estado de Maryland - US. Os modelos foram comparados utilizando-se os critérios de informação de Akaike (AIC), Akaike corrigido (AICC), Bayesiano (BIC), DIC e foi calculado o Erro Quadrático Médio (EQM) afim de verificar o poder de predição dos modelos.

Na Seção 10.2 aplicou-se os Filtros de Chopin (2007), Caron et al. (2012) e o FChAP, utilizando as distribuições Poisson, Binomial Negativa, Poisson Inflacionada de Zeros e Binomial Negativa Inflacionada de Zeros, aos dados sobre número de acidentes em minérios de carvão na Inglaterra, concentração de ozônio na cidade do México, número de Lesões na Saúde Ocupacional nos Estados Unidos e número de Casos de Poliomielite nos Estados Unidos, para identificar possíveis pontos de quebras estruturais nas séries. Os modelos foram comparados utilizando-se os critérios de informação de Akaike (AIC), Akaike corrigido (AICC), Bayesiano (BIC), DIC e foi calculado, também, o Erro Quadrático Médio (EQM) afim de verificar o poder de predição dos modelos.

A estratégia utilizada para aplicação dos filtros de Chopin (2007) e FChAP foi executá-los 10 vezes utilizando 50000 partículas. As estimativas foram calculadas utilizando o valor mediano das 10 trajetórias. A justificativa para se utilizar essa estratégia se encontra na análise de dados simulados, pois pode-se ver que essa abordagem tornam as estimativas próximas dos valores reais.

Conforme dito no Capítulo 7, o algoritmo de Caron et al. (2012) foi implementado apenas para o caso Poisson.

10.1 Aplicações utilizando o Algoritmo de McCormick et al. (2012)

10.1.1 Número de Casos de Sífilis no Estado de Maryland - US

Nesta seção ajustaram-se os modelos de regressão dinâmica Poisson e Binomial Negativa a dados relacionados ao número de Casos de sífilis no Estado de Maryland - US, ocorridos entre os anos de 2001 a 2011. Tais dados estão disponibilizados no pacote ZIM do software *R*. A Figura 10.1 ilustra graficamente a série de casos de sífilis.

Pela Figura 10.2, nota-se que existe uma grande frequência de zeros nos dados. Sendo assim, há indicações de que os modelos inflacionados de zeros são mais adequados para descrever a evolução da série, se comparado aos modelos Poisson e Binomial Negativo. Para tanto, aplicaram-se os algoritmos de McCormick et al. (2012) e McPE utilizando os modelos Poisson, Binomial Negativo, Poisson inflacionado de zeros e Binomial Negativo inflacionado de zeros. Os resultados dos ajustes para os modelos de regressão estão ilustrados nas Figuras de 10.2 a 10.5.

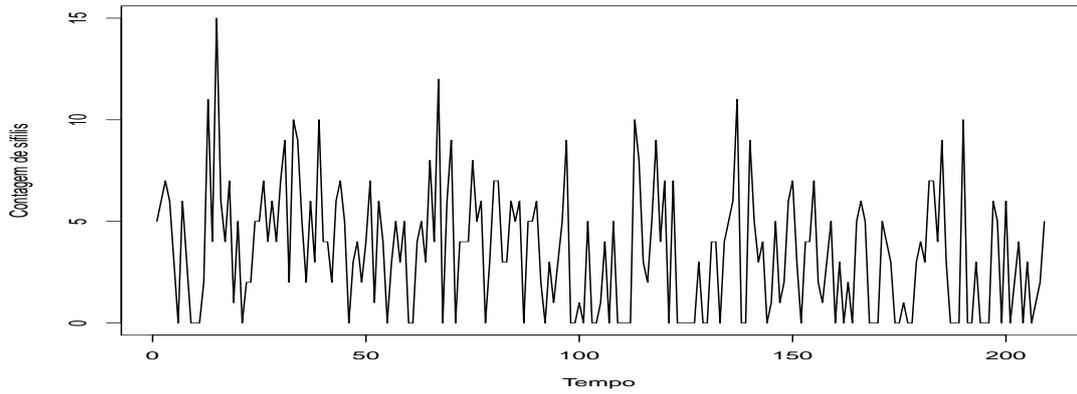
A fim de verificar quais dos modelos ajustaram-se melhor aos dados, foram calculados os critérios de informação de Akaike (AIC), Akaike corrigido (AICC), critério Bayesiano (BIC), DIC e o Erro Quadrático Médio (EQM). Os valores

calculados estão ilustrados na Tabela 10.1.

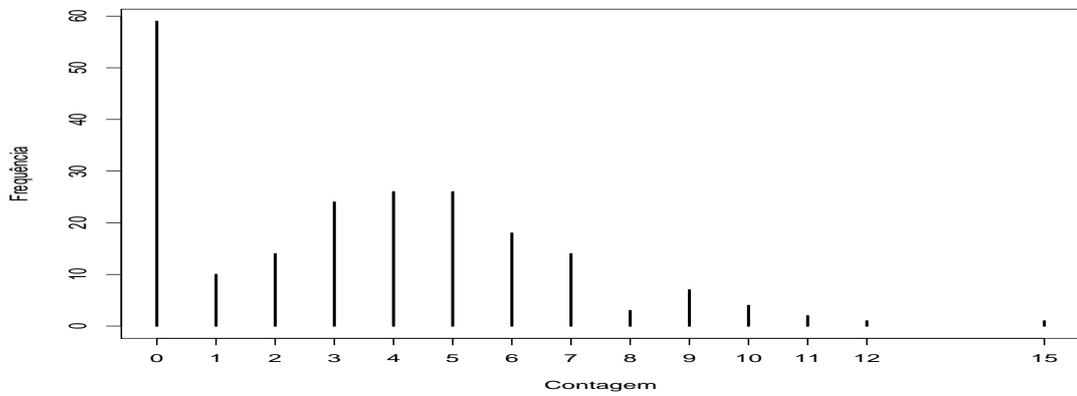
Tabela 10.1: Critérios de informação para os modelos estimados para o número de casos de Sífilis em Maryland.

Modelo	AIC	AICC	BIC	DIC	EQM
Poisson	-271,455	-271,034	-254,707	-260,991	0,06732708
Binomial Negativo	-270,165	-269,98	-258,999	-262,492	0,04788317
ZIP	-276,514	-276,329	-265,348	-268,841	0,04149637
ZINB	-278,543	-278,497	-272,96	-273,662	0,03304609

Observando a Tabela 10.1, nota-se que, de forma geral, os modelos inflacionados de zeros ajustaram-se melhor aos dados se comparado com os modelos Poisson e Binomial Negativo, por apresentarem os menores valores de AIC, DIC, BIC, DIC e EQM.



(a) Número de casos de Sífilis em Maryland - US.



(b) Frequência das contagens da série de sífilis.

Figura 10.1: Série temporal observada e distribuição de frequência para os dados de ocorrência de sífilis.

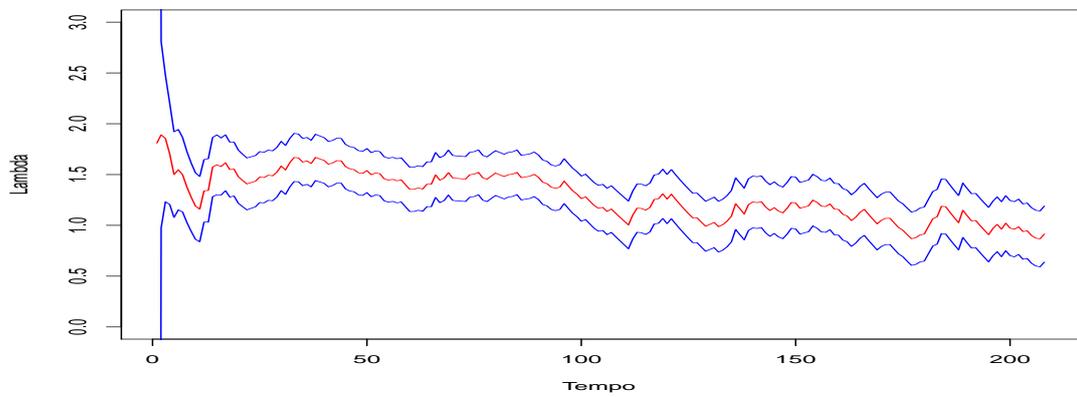
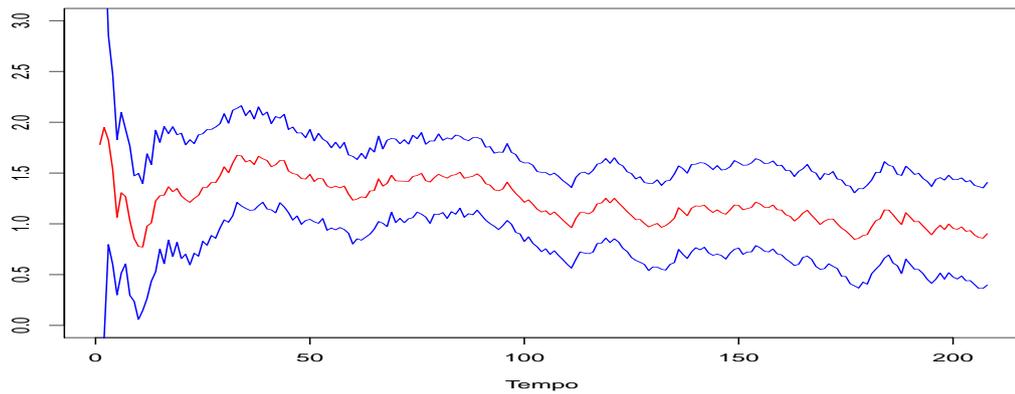
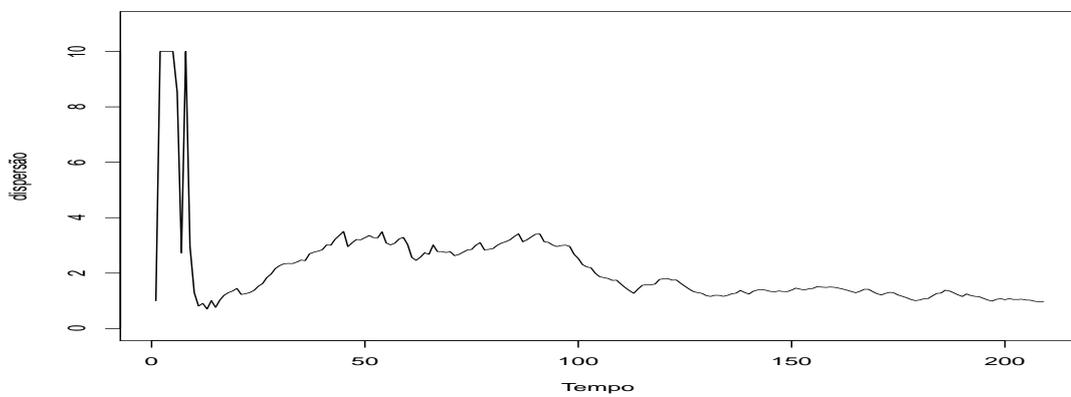


Figura 10.2: Média filtrada para a distribuição Poisson

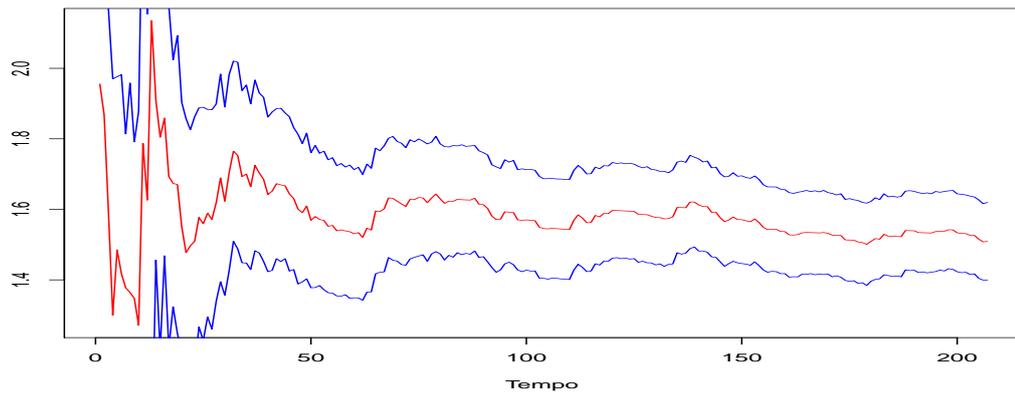


(a) Média filtrada para a distribuição Binomial Negativa.

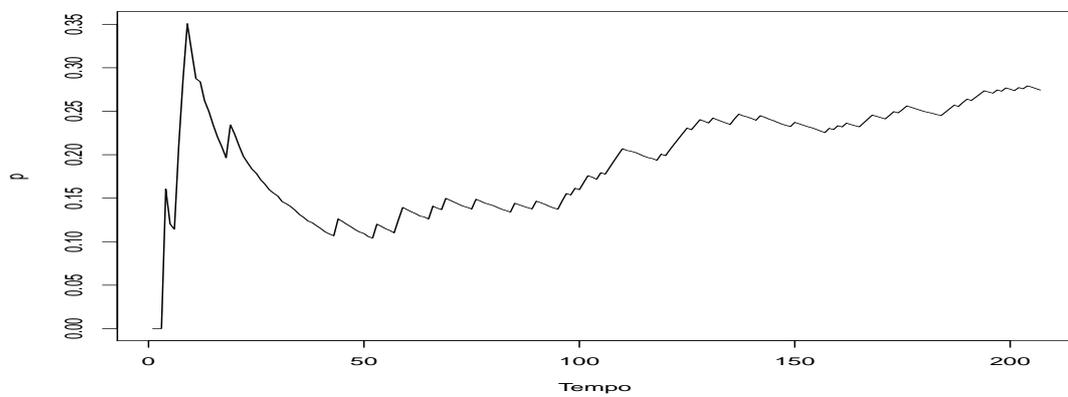


(b) Dispersão filtrada para a distribuição Binomial Negativa.

Figura 10.3: Estimativas para os parâmetros do modelo Binomial Negativo.

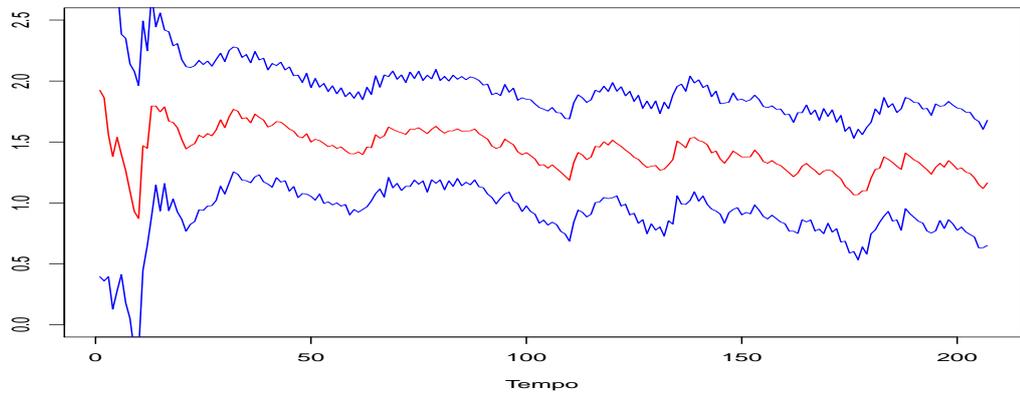


(a) Média filtrada para a distribuição ZIP.

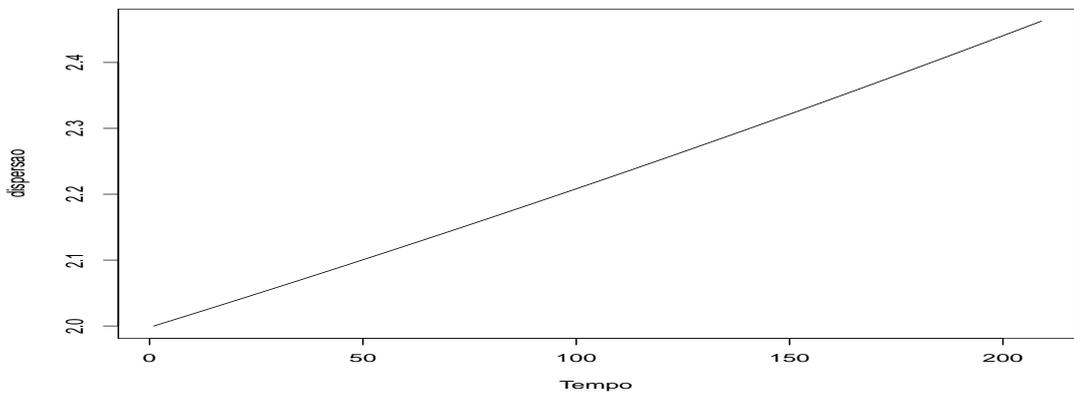


(b) Probabilidade de zero filtrada para a distribuição ZIP.

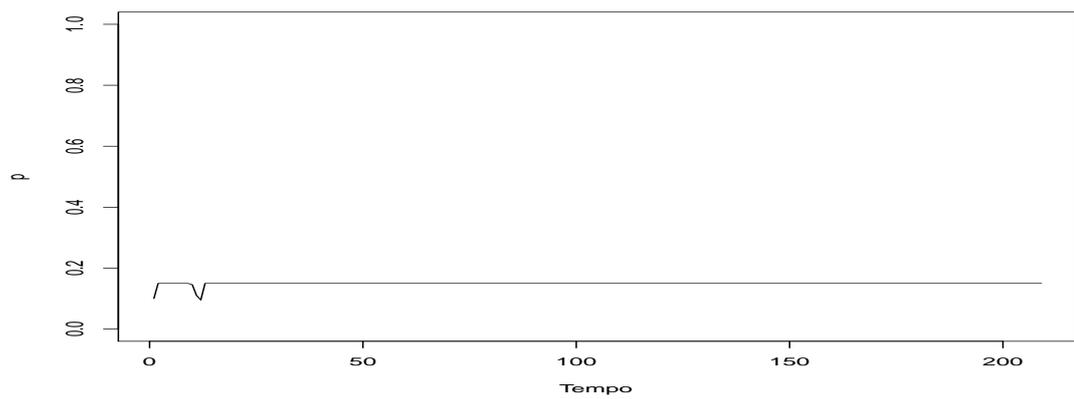
Figura 10.4: Estimativas para os parâmetros do modelo ZIP.



(a) Média filtrada para a distribuição ZINB.



(b) Dispersão filtrada para a distribuição ZINB.



(c) Probabilidade de zero filtrada para a distribuição ZINB.

Figura 10.5: Estimativas para os parâmetros do modelo ZINB.

10.2 Aplicações utilizando os filtros de Chopin (2007), Caron et al. (2012) e FChAP

10.2.1 Desastres em minérios de carvão na Inglaterra

Nesta seção aplicou-se os filtros de Caron et al. (2012), Chopin (2007) e o FChAP para analisar a série temporal anual do número de desastres em minérios de carvão na Inglaterra entre 1950 e 1962. Tais dados foram analisados por Raftery e Akman (1986) e Lai e Xing (2011), os quais propuseram métodos alternativos para detectar, dinamicamente, possíveis quebras estruturais nessa série. A Figura 10.6 ilustra graficamente a série de desastres.

Observando a Figura 10.6 nota-se que a série muda seu nível próximo do ano de 1890, anteriormente a uma quantidade maior de desastres. Em contrapartida, após o ano de 1890, ocorre uma queda no número de desastres.

Para a análise da série aplicou-se o Filtro de Caron et al. (2012) utilizando a distribuição Poisson, conforme Seção 7.2 do Capítulo 7. Aplicou-se, também, o Filtro de Chopin (2007) utilizando a distribuição de Poisson, conforme Seção 7.1 do Capítulo 7. Finalmente, aplicou-se o Filtro FChAP utilizando a distribuição Binomial Negativa, conforme Capítulos 8 e 9. Os resultados para os três filtros encontram-se ilustrados nas Figuras de 10.7 a 10.9 respectivamente.

De acordo com o resultado gerado pelo Filtro de Caron et al. (2012), Figura 10.7, nota-se que a mudança estrutural na série ocorre em torno do ano de 1890. Tal resultado foi obtido, também, por Raftery e Akman (1986).

Já os resultados dos filtros de Chopin (2007) utilizando a distribuição de Poisson, Figura 10.8, e FChAP utilizando a distribuição Binomial Negativa, Figura 10.9, indicam que não só houve mudança em torno de 1890, como também em torno do ano de 1946, aproximadamente. Tais resultados também foram observados por Lai e Xing (2011).

Observando as médias estimadas por ambos os filtros, nota-se que houve uma queda significativa de seu valor filtrado após o ano de 1890, corroborando a evidência de quebra estrutural na série após esse ano. O parâmetro de dispersão filtrado, para a distribuição Binomial Negativa aumenta gradativamente e, após o ano de 1890, estabiliza-se próximo do valor 2,4.

A Tabela 10.2 ilustra o tempo de processamento de cada algoritmo. Observa-se que o filtro de Caron et al. (2012) leva menos tempo para ser executado.

Tabela 10.2: Comparação entre os tempos, aproximados, de processamentos dos algoritmos.

Algoritmo	Tempo de processamento
Caron	5 segundos
Chopin - Poisson	20 segundos
FChAP - BN	30 segundos

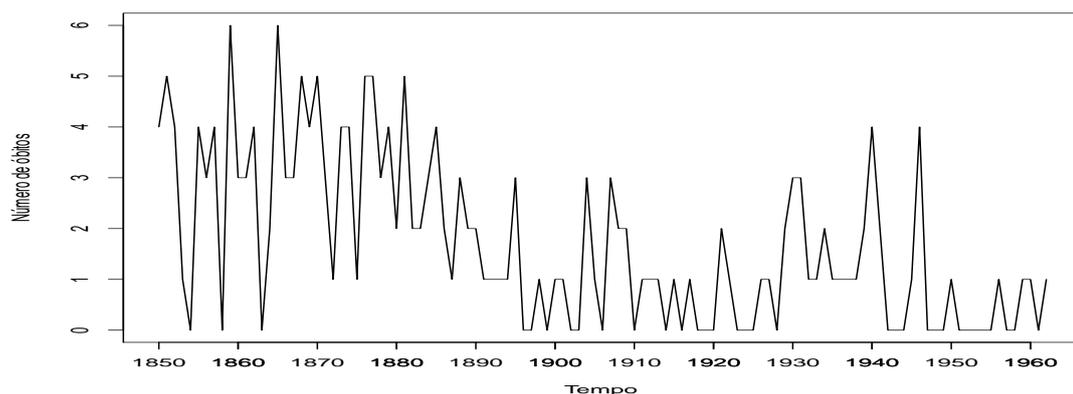
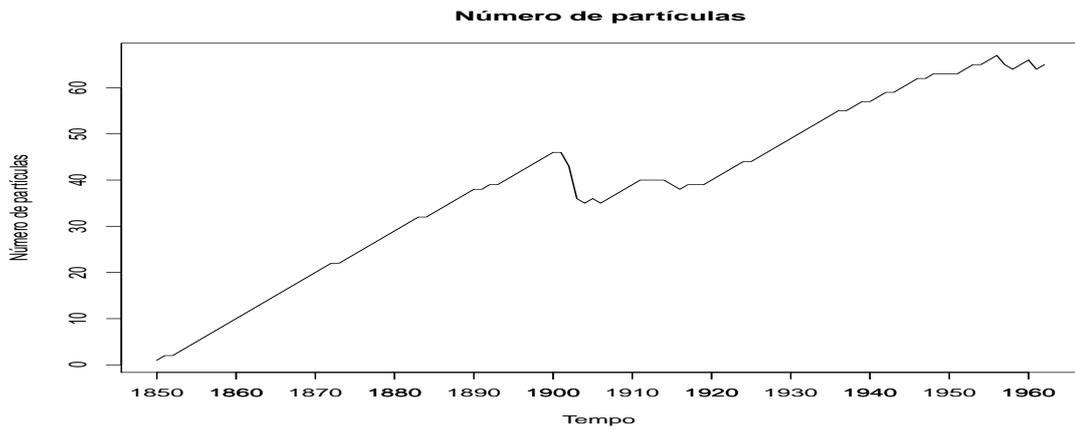
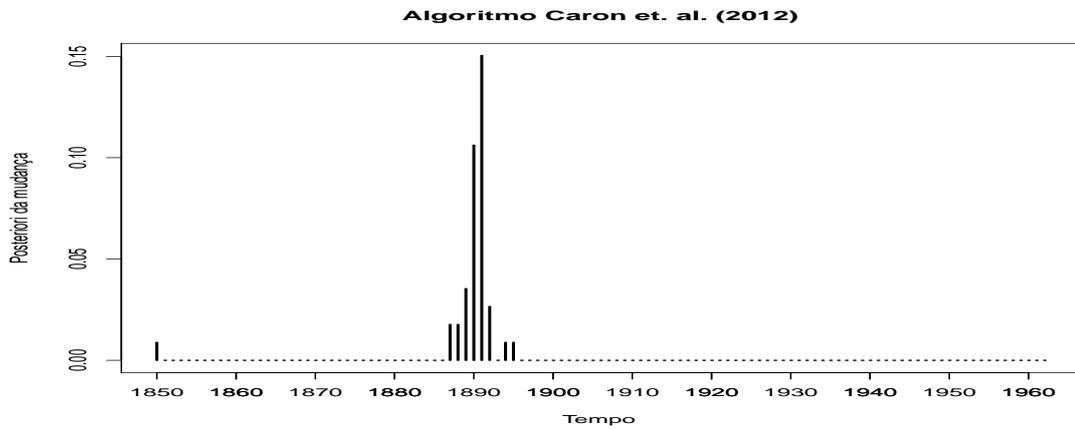


Figura 10.6: Número de desastres em minérios de carvão na Inglaterra entre 1950 e 1962

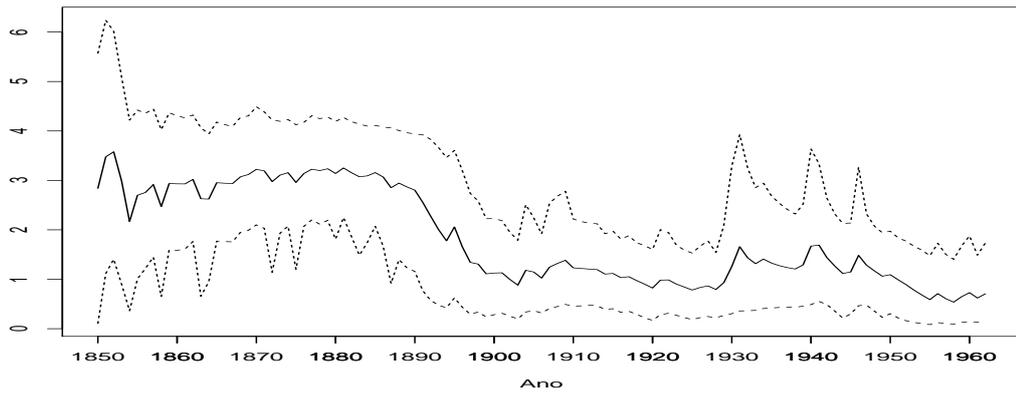


(a) Número de partículas utilizadas em cada iteração.

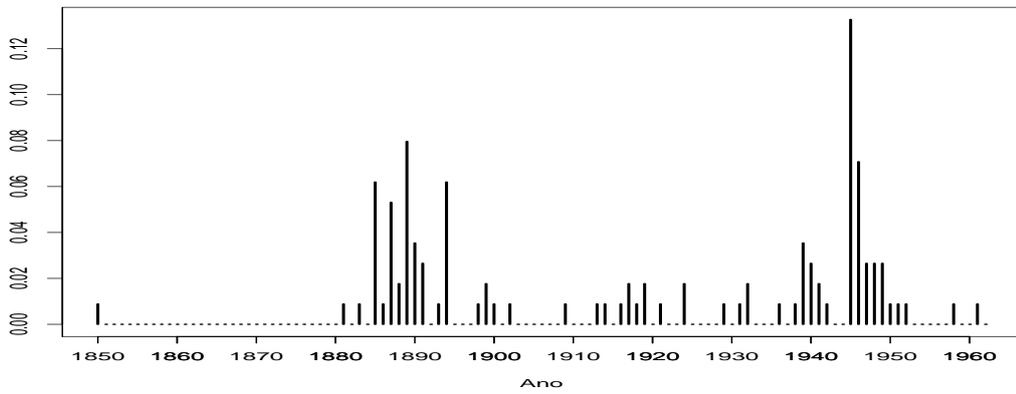


(b) Probabilidade a posteriori de ocorrer mudança.

Figura 10.7: Estimativas para o filtro de Caron et al. (2012) utilizando a distribuição Poisson.

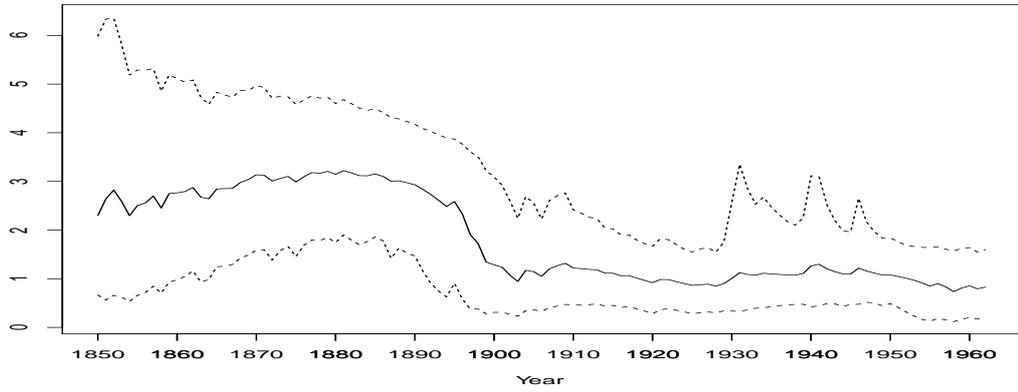


(a) Média filtrada, utilizando o filtro de Chopin (2007), para a distribuição Poisson.

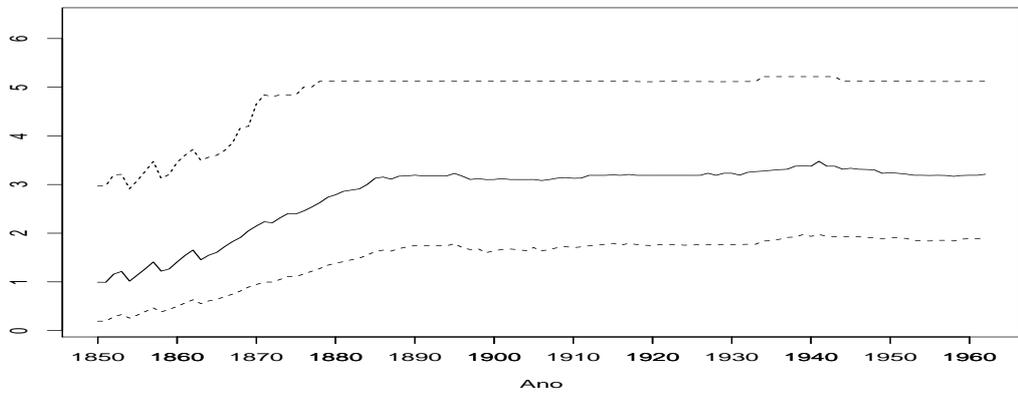


(b) Probabilidade a posteriori de ocorrer mudança segundo filtro de Chopin (2007) utilizando a distribuição Poisson.

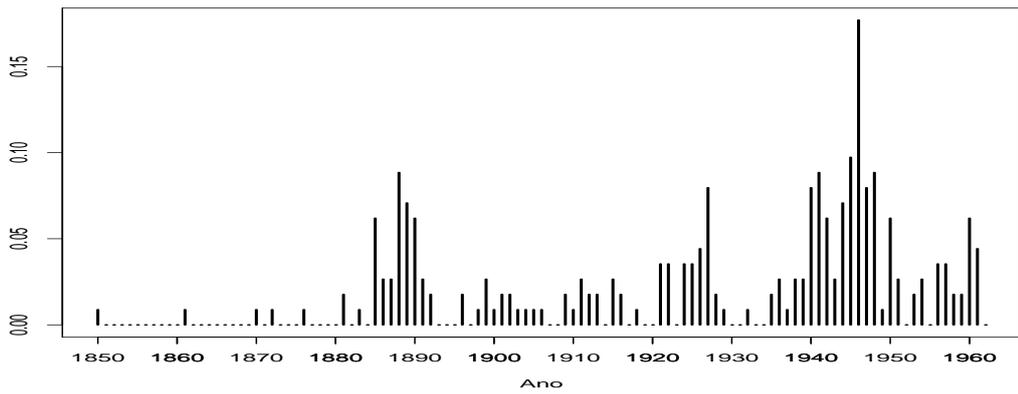
Figura 10.8: Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.



(b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.



(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição Binomial Negativa.

Figura 10.9: Estimativas para o filtro de Chopin (2007) utilizando a distribuição Binomial Negativa.

10.2.2 Concentração de ozônio na cidade do México

Nesta seção aplicou-se os filtros de Caron et al. (2012), Chopin (2007) e o FChAP para analisar a série temporal mensal do número de vezes em que a concentração de ozônio ultrapassou o limite de 0.17 ppm (partes por milhão) na cidade do México os anos de 1998 e 2004. Tais dados foram analisados por Achcar et al. (2008), os autores utilizam processo de poisson não homogêneo para modelar o número de vezes em que a concentração de ozônio superou o limiar estabelecido. A concentração de ozônio acima de 0.11 já implica em uma deterioração da saúde. Sendo assim, torna-se importante monitorar as concentrações de ozônio no ar, de modo a permitir que ações governamentais sejam tomadas de modo a minimizar danos à saúde da população. A Figura 10.10 ilustra a série de concentrações de ozônio.

Observando a Figura 10.10 nota-se que a série muda seu nível a partir de dezembro de 2012 (ponto 60 no gráfico). Antes desse período há um grande número de vezes em que a concentração do ozônio ultrapassou o valor de 0.17 ppm. Em contrapartida, após essa data, ocorre uma queda no número de vezes em que a concentração do ozônio ultrapassou o limite.

Para a análise da série aplicou-se o Filtro de Caron et al. (2012) utilizando a distribuição Poisson, conforme Seção 7.2 do Capítulo 7. Aplicou-se, também, o Filtro de Chopin (2007) utilizando a distribuição de Poisson, conforme Seção 7.1 do Capítulo 7. Finalmente, aplicou-se o Filtro FChAP, utilizando a distribuição Binomial Negativa, conforme Capítulos 8 e 9. Os resultados para os três filtros encontram-se ilustrados nas Figuras de 10.11 a 10.13 respectivamente.

De acordo com o resultado gerado pelo Filtro de Caron et al. (2012), Figura 10.11, nota-se que a mudança estrutural na série ocorre em torno de dezembro de 2012. Já os resultados dos filtros de Chopin (2007), Figura 10.12, utilizando a distribuição de Poisson, e FChAP utilizando a distribuição Binomial Negativa, Figura 10.13, indicam, também, o mesmo período em que ocorreu a quebra estrutural da série.

Observando as médias estimadas por ambos os filtros, nota-se que houve uma

queda significativa de seu valor filtrado após dezembro de 2012, corroborando a evidência de quebra estrutural na série após essa data. O parâmetro de dispersão filtrado, para a distribuição Binomial Negativa aumenta gradativamente e, após dezembro de 2012, estabiliza-se próximo do valor 2.

A Tabela 10.3 ilustra o tempo de processamento de cada algoritmo. Observa-se que o filtro de Caron et al. (2012) leva menos tempo para ser executado.

Tabela 10.3: Comparação entre os tempos, aproximados, de processamentos dos algoritmos.

Algoritmo	Tempo de processamento
Caron	10 segundos
Chopin - Poisson	30 segundos
FChAP - BN	45 segundos

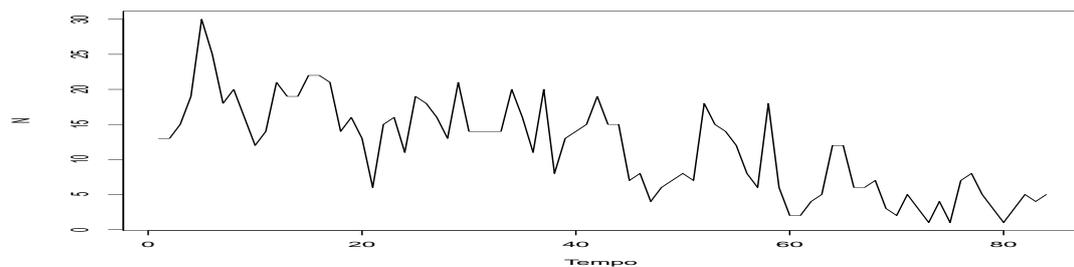
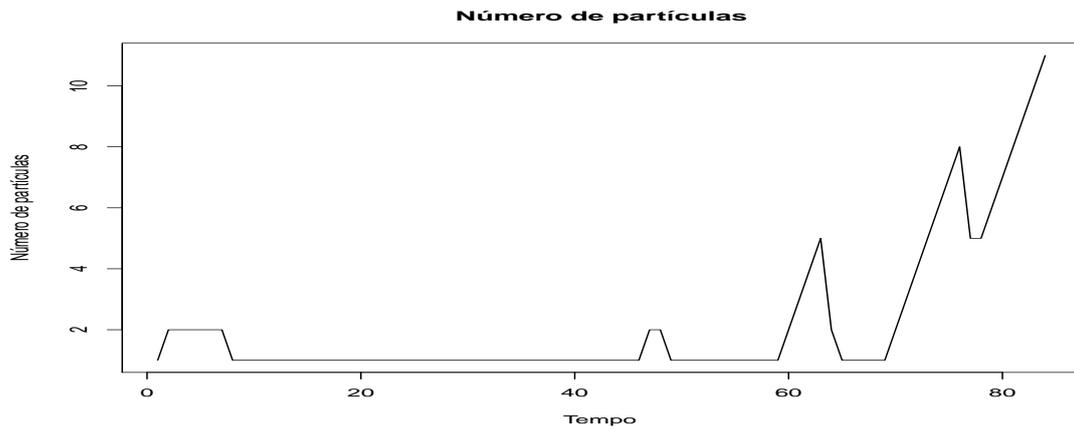
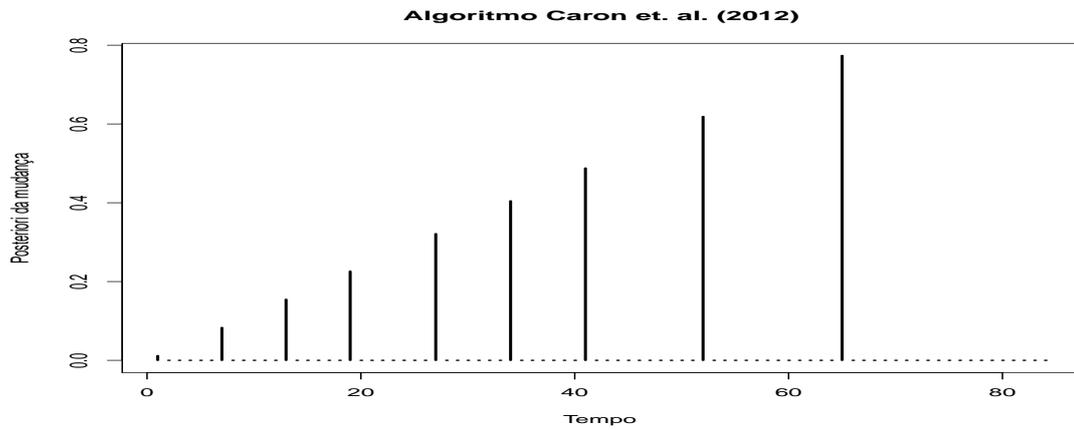


Figura 10.10: Número de vezes em que a concentração de ozônio ultrapassou o limite de 0.17 ppm (partes por milhão) na cidade do México os anos de 1998 e 2004.

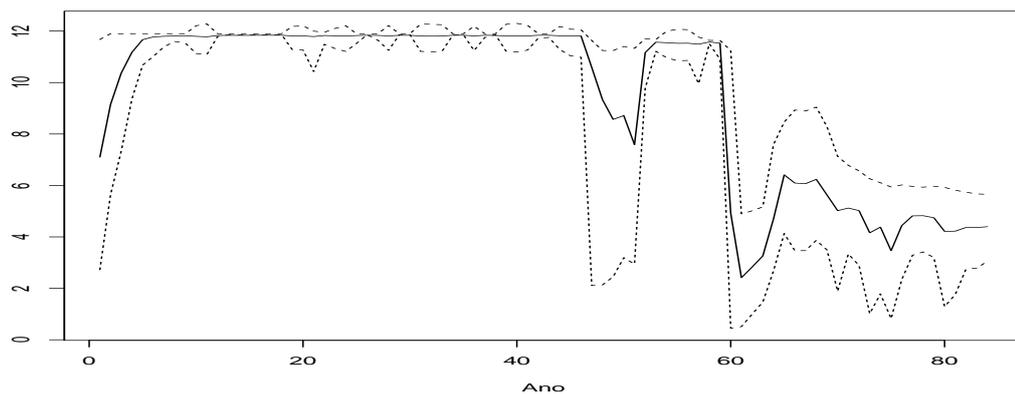


(a) Número de partículas utilizadas em cada iteração.

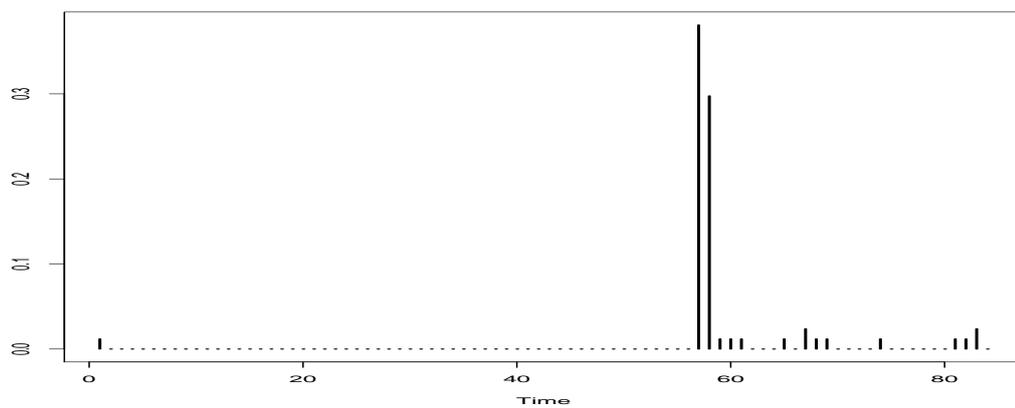


(b) Probabilidade a posteriori de ocorrer mudança.

Figura 10.11: Estimativas para o filtro de Caron et al. (2012) utilizando a distribuição Poisson.

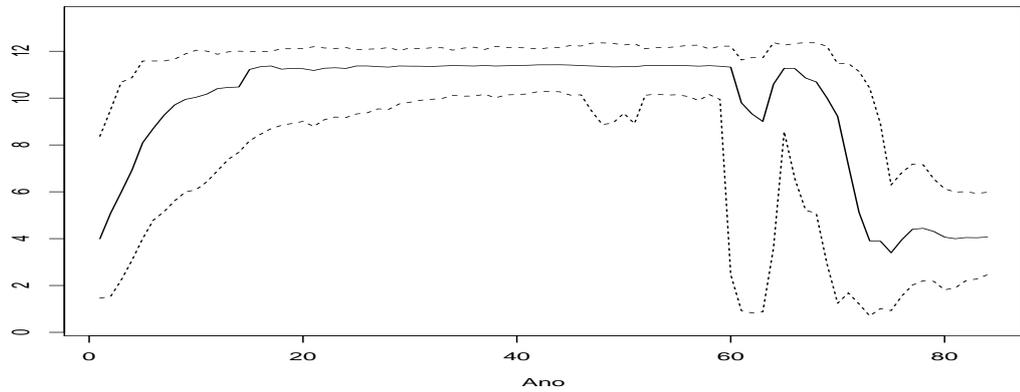


(a) Média filtrada, utilizando o filtro de Chopin (2007), para a distribuição Poisson.

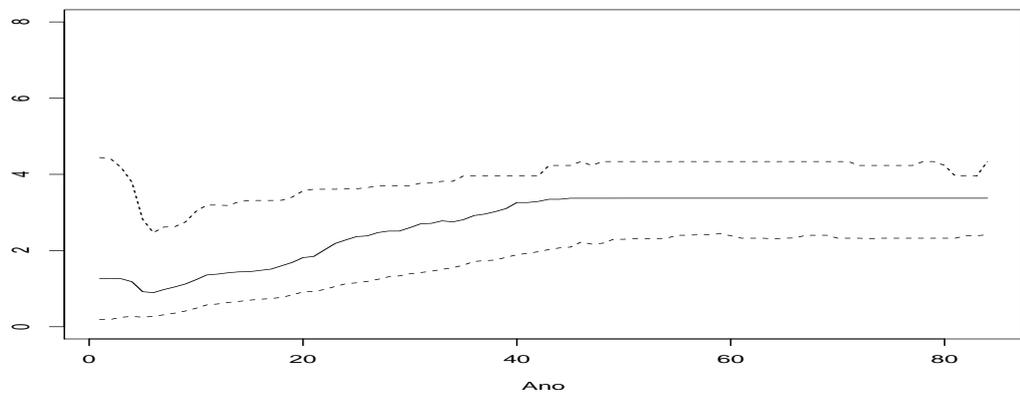


(b) Probabilidade a posteriori de ocorrer mudança segundo filtro de Chopin (2007) utilizando a distribuição Poisson.

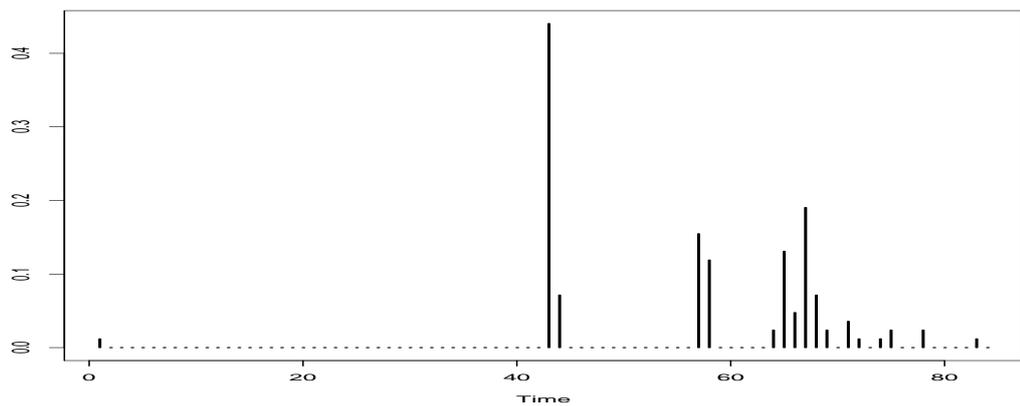
Figura 10.12: Estimativas para o filtro de Chopin (2007) utilizando a distribuição de Poisson.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.



(b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.



(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição Binomial Negativa.

Figura 10.13: Estimativas para o filtro de Chopin (2007) utilizando a distribuição Binomial Negativa.

10.2.3 Número de Lesões na Saúde Ocupacional nos Estados Unidos

Nesta seção aplicou-se os filtros de Chopin (2007) e o FChAP a dados relacionados ao número de casos mensais de lesões na Saúde Ocupacional nos Estados Unidos no período de Julho de 1988 a Outubro de 1995. Tais dados foram analisados por Yau et al. (2004) e estão disponibilizados no pacote ZIM do software *R*. A Figura 10.14 ilustra a série de casos de lesões.

Pela Figura 10.14, nota-se que existe uma grande frequência de zeros nos dados. Sendo assim, há indicações de que os modelos inflacionados de zeros são mais adequados para descrever a evolução da série, se comparado aos modelos Poisson e Binomial Negativo.

Para tanto, se ajustou os Filtros de Chopin (2007), para a distribuição de Poisson, e o FChAP para as distribuições Binomial Negativa, ZIP e ZINB, aos dados. Os resultados dos ajustes estão ilustrados nas Figuras de 10.15 a 10.18. Tais figuras ilustram a evolução dos estados para o modelo Poisson, e a evolução dos estados e dos parâmetros estáticos para os modelos Binomial Negativo e ZIP e ZINB.

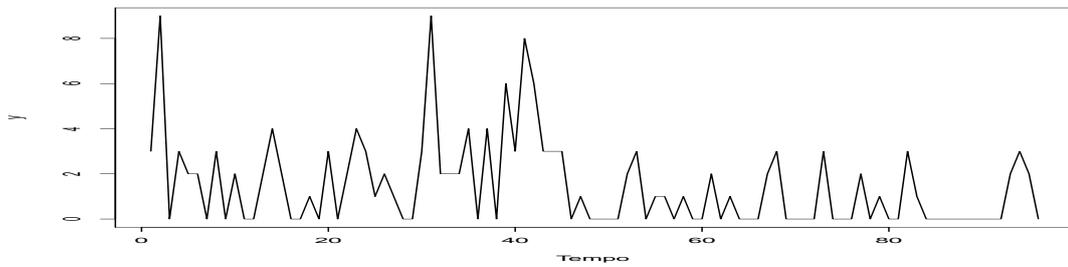
Observando-se as Figuras de 10.15 a 10.18, nota-se que existem evidências de que tenha ocorrido uma quebra estrutural em torno de setembro de 1991. As médias estimadas mostram que após esse ano houve uma queda no valor filtrado, reforçando a evidência de quebra estrutural nessa série em torno desse período.

A fim de verificar quais dos modelos ajustaram-se melhor aos dados, foram calculados os critérios de informação de Akaike (AIC), Akaike corrigido (AICC), critério Bayesiano (BIC), DIC e o Erro Quadrático Médio. Os valores calculados estão ilustrados na Tabela 10.4.

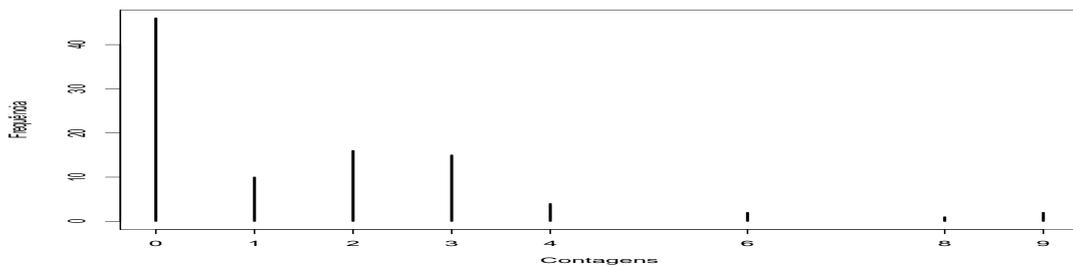
Observando a Tabela 10.4, nota-se que, de forma geral, os modelos inflacionados de zeros ajustaram-se melhor aos dados se comparado com os modelos Poisson e Binomial Negativo, por apresentarem os menores valores de AIC, DIC, BIC, DIC e EQM.

Tabela 10.4: Critérios de informação para os modelos estimados para os dados de lesões na saúde ocupacional dos Estados Unidos.

Modelo	AIC	AICC	BIC	DIC	EQM
Poisson	-130,479	-130,416	-125,123	-126,841	0,06878054
Binomial Negativo	-131,836	-128,293	-129,105	-130,4705	0,06182332
ZIP	-140,482	-140,396	-135,353	-136,957	0,03923451
ZINB	-147,699	-147,678	-145,135	-145,457	0,03073045

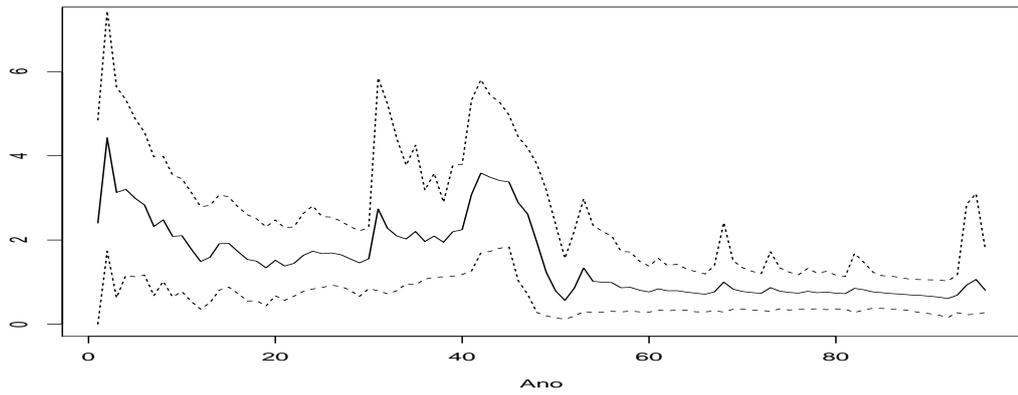


(a) Série y_t .

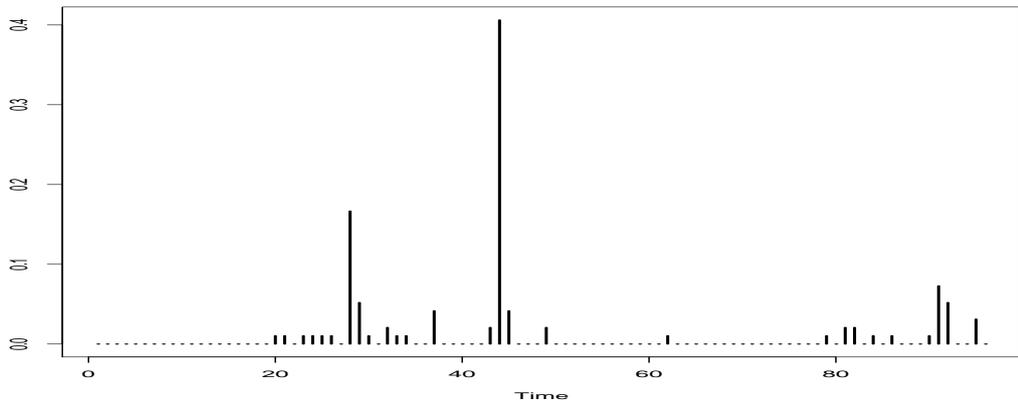


(b) Frequência das contagens para a série de lesões.

Figura 10.14: Número de casos mensais de lesões na Saúde Ocupacional nos Estados Unidos no período de Julho de 1988 a Outubro de 1995.

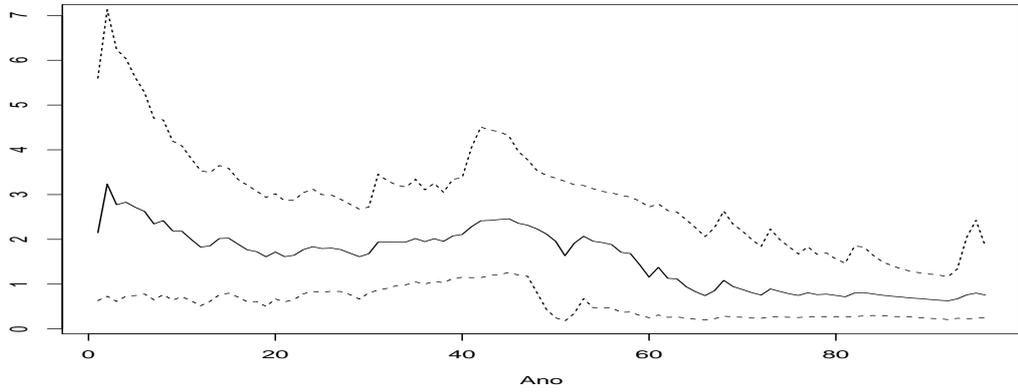


(a) Média filtrada, utilizando o filtro de Chopin (2007), para a distribuição Poisson.

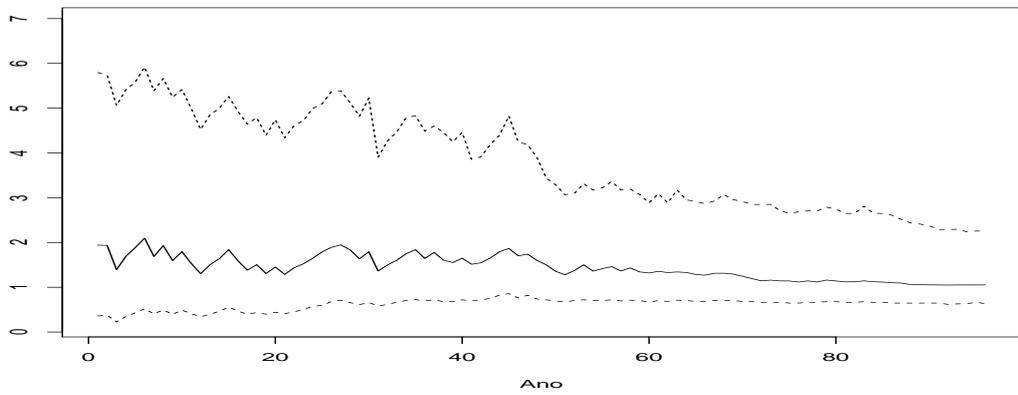


(b) Probabilidade a posteriori de ocorrer mudança segundo filtro de Chopin (2007) utilizando a distribuição Poisson.

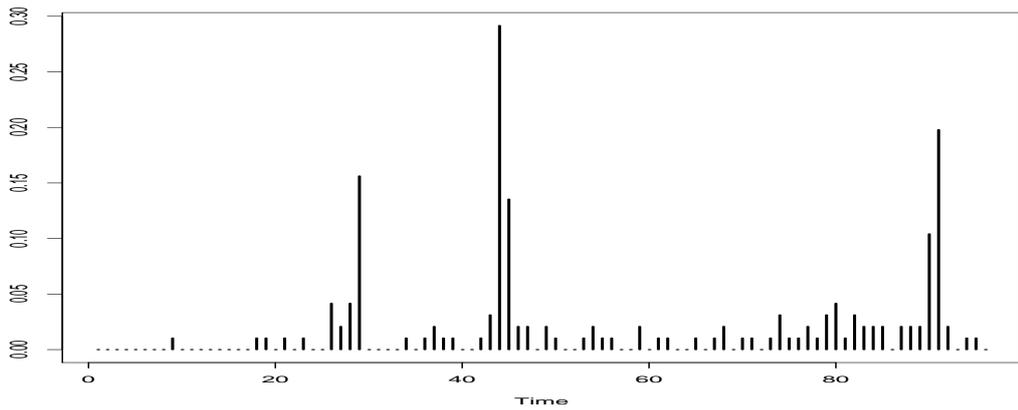
Figura 10.15: Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.

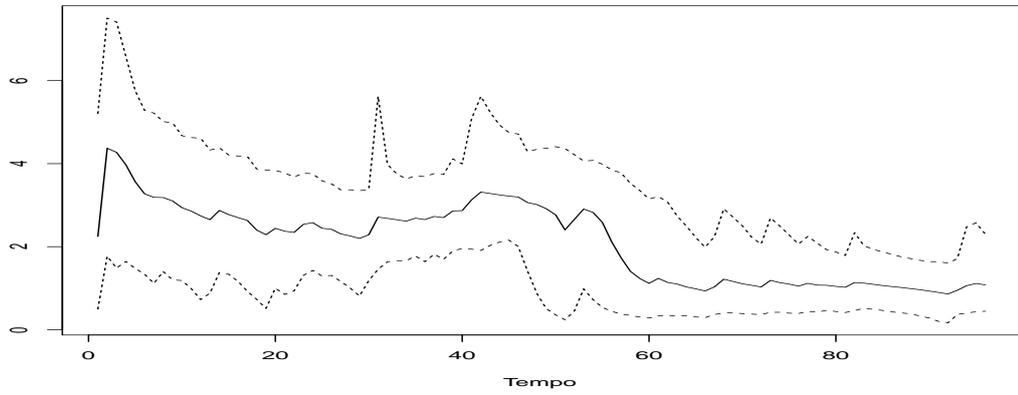


(b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.

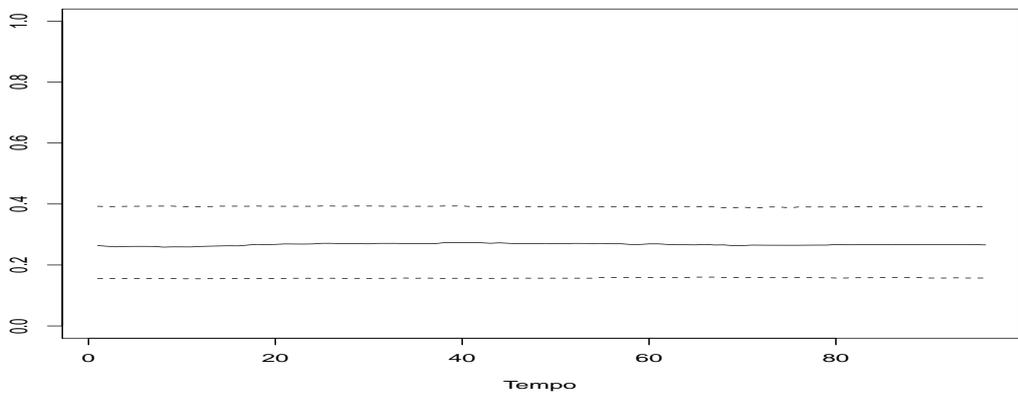


(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição Binomial Negativa.

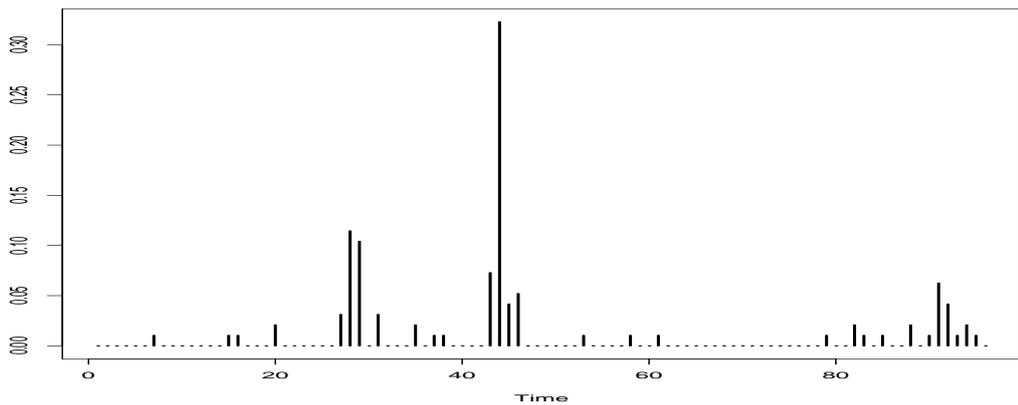
Figura 10.16: Estimativas para o filtro FChAP utilizando a distribuição Binomial Negativa.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição ZIP.

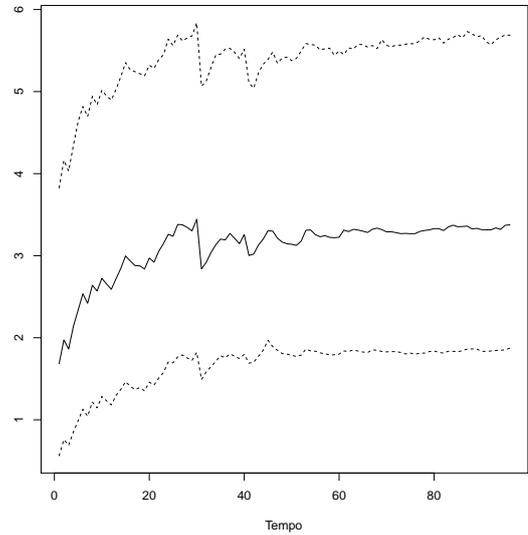
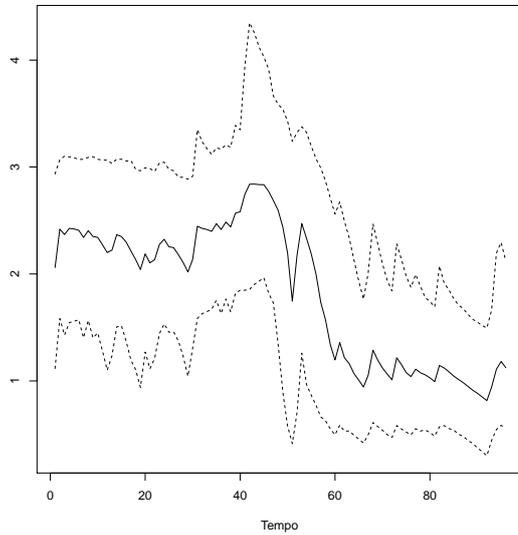


(b) Probabilidade de zero filtrada, utilizando o filtro FChAP, para a distribuição ZIP.

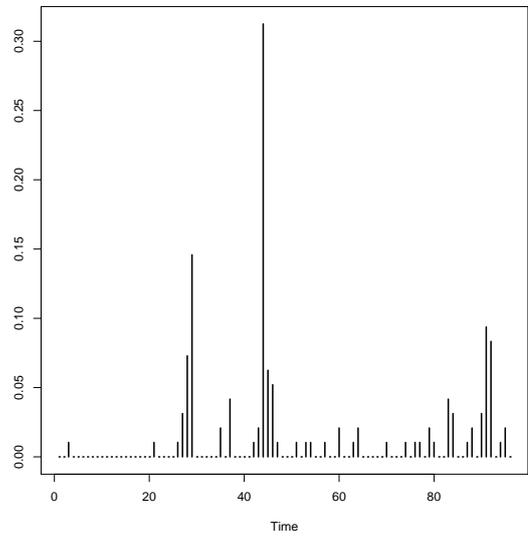
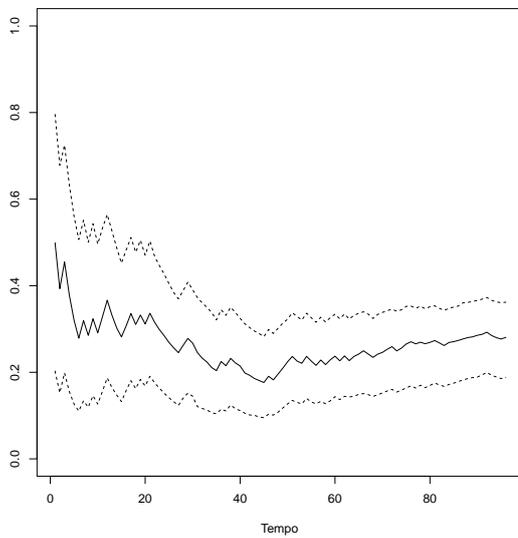


(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição ZIP.

Figura 10.17: Estimativas para o filtro FChAP utilizando a distribuição ZIP.



(a) Média filtrada, utilizando o filtro FChAP, (b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição ZINB.



(c) Probabilidade de zero filtrada, utilizando o filtro FChAP, para a distribuição ZINB. (d) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição ZINB.

Figura 10.18: Estimativas para o filtro FChAP utilizando a distribuição ZINB.

10.2.4 Número de Casos de Poliomelite nos Estados Unidos

Nesta seção aplicou-se os filtros de Chopin (2007) e o FChAP a dados relacionados ao número de casos mensais de Poliomelite nos Estados Unidos no período de 1970 a 1983. Tais dados foram analisados por Zeger (1988). A Figura 10.19 ilustra a série de casos de Poliomelite.

Pela Figura 10.19, nota-se que existem uma grande frequência de zeros nos dados. Sendo assim, há indicações de que os modelos inflacionados de zeros são mais adequados para descrever a evolução da série, se comparado aos modelos Poisson e Binomial Negativo.

Para tanto, ajustaram-se os Filtros de Chopin (2007), para a distribuição de Poisson, e o FChAP para as distribuições Binomial Negativa, ZIP e ZINB, aos dados. Os resultados dos ajustes estão ilustrados nas Figuras 10.20 a 10.23. Tais figuras ilustram a evolução dos estados para o modelo Poisson, e a evolução dos estados e dos parâmetros estáticos para os modelos Binomial Negativo, ZIP e ZINB.

Observando-se as Figuras de 10.20 a 10.23, nota-se que existe evidências de que tenha ocorrido uma quebra estrutural em torno de dezembro de 1979 (ponto 120 nos gráficos). As médias estimadas mostram que, após o ano de 1979, houve uma queda no valor filtrado, reforçando a evidência de quebra estrutural nessa série nesse ano.

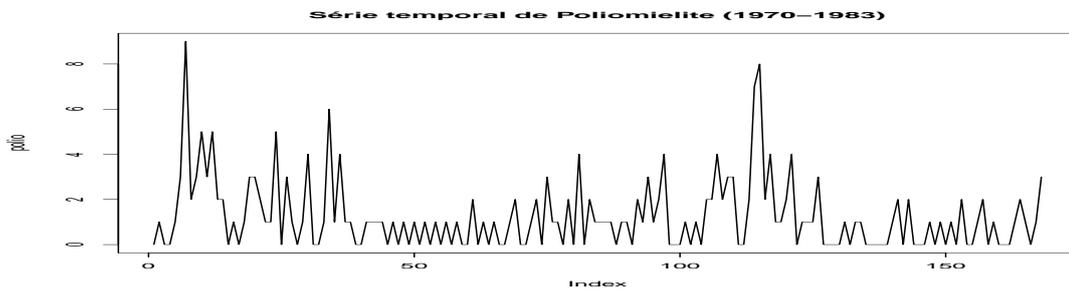
A fim de verificar quais dos modelos proporcionaram as melhores descrições dos dados, foram calculados os critérios de informação de Akaike (AIC), Akaike corrigido (AICC), critério Bayesiano (BIC), DIC e o Erro Quadrático Médio (EQM). Os valores calculados estão sumarizados na Tabela 10.5.

De acordo com a Tabela 10.5, nota-se que, de forma geral, os modelos inflacionados de zeros proporcionaram melhor ajuste aos dados se comparados com os modelos Poisson e Binomial Negativo por apresentar menores valores de AIC, AICC, BIC, DIC e EQM.

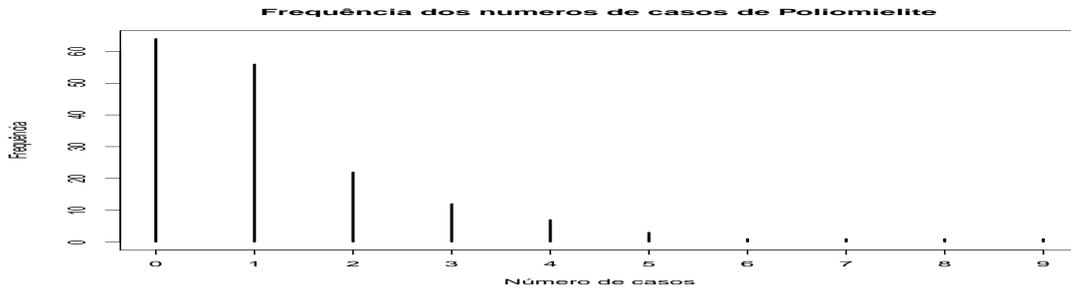
Tabela 10.5: Critérios de informação para os modelos estimados para os dados de Poliomielite.

Modelo	AIC	AICC	BIC	DIC	EQM
Poisson	-235,559	-235,22	-222,096	-227,148	0,05387581
Binomial Negativo	-241,352	-241,203	-232,377	-235,185	0,04649438
ZIP	-246,02	-245,824	-236,296	-239,478	0,01989555
ZINB	-262,862	-262,85	-259,738	-259,62	0,01614005

No próximo capítulo serão dadas as considerações finais para esta dissertação e propostas de trabalhos futuros.

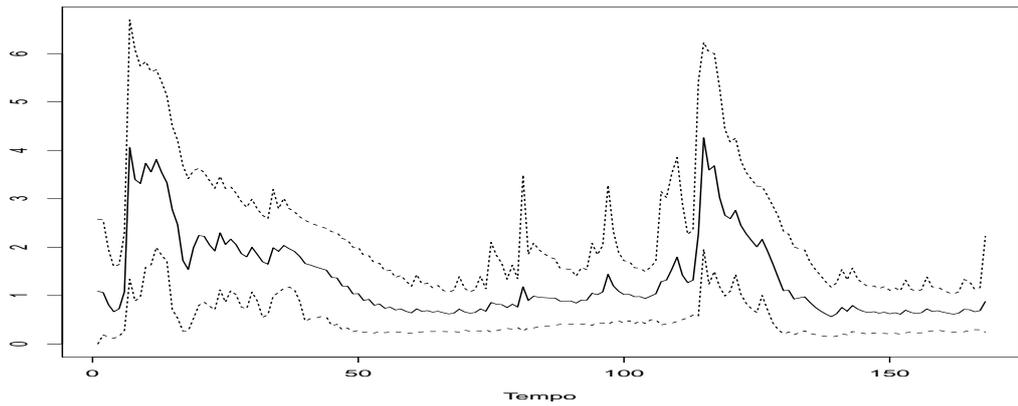


(a) Número de casos, mensais, de Poliomielite nos Estados Unidos no período de 1970 a 1983.

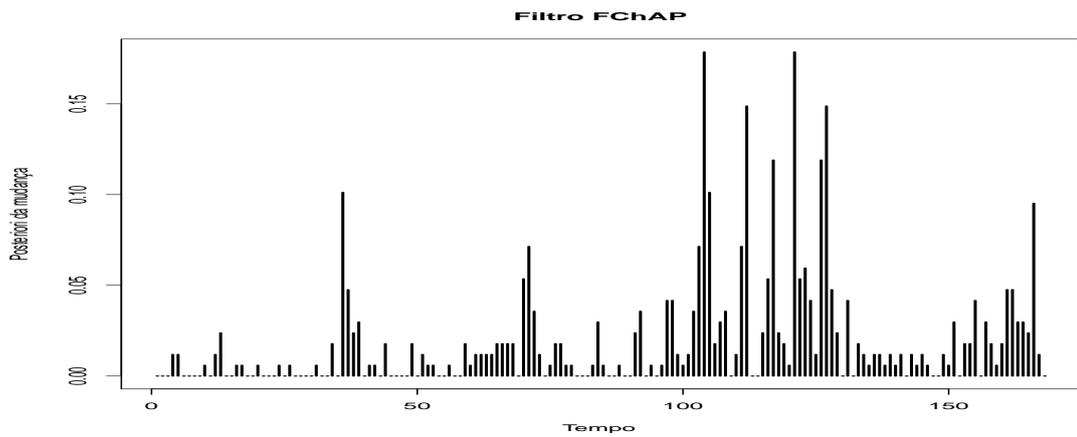


(b) Frequência das contagens para a série de Poliomielite.

Figura 10.19

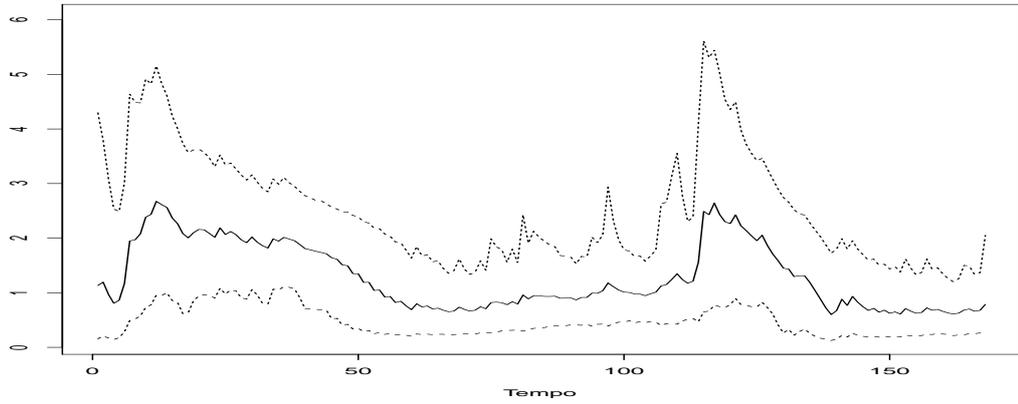


(a) Média filtrada, utilizando o filtro de Chopin (2007), para a distribuição Poisson.

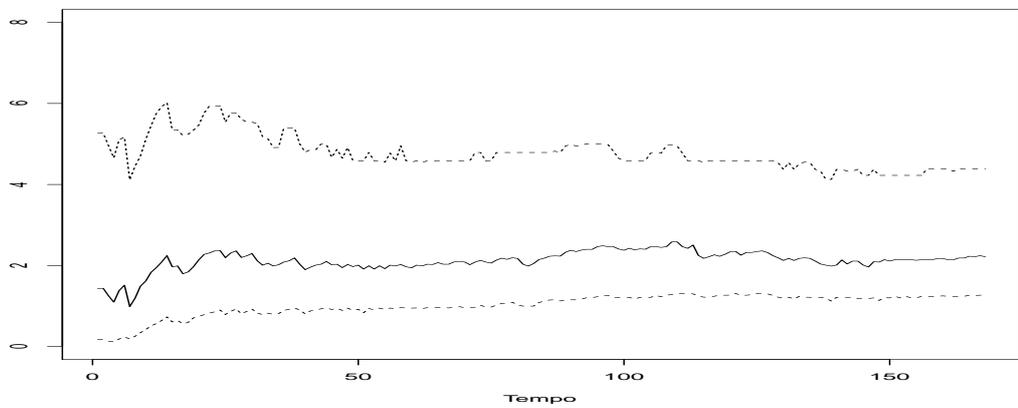


(b) Probabilidade a posteriori de ocorrer mudança segundo filtro de Chopin (2007) utilizando a distribuição Poisson.

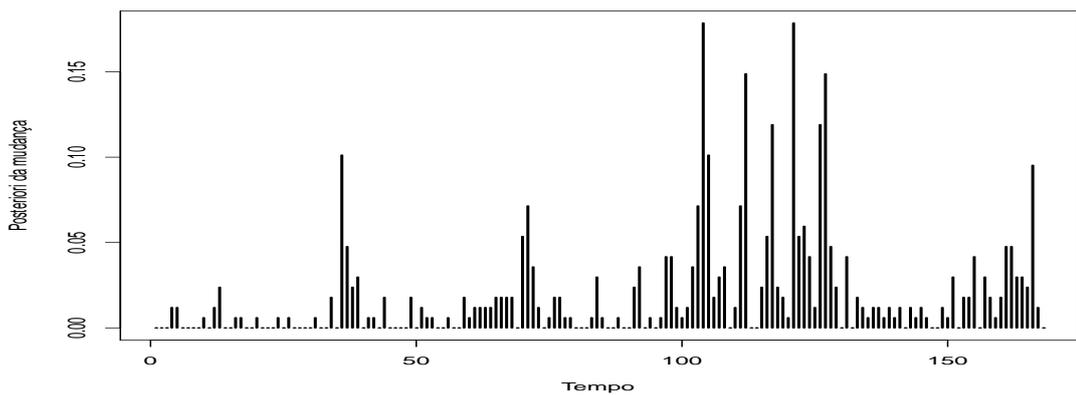
Figura 10.20: Estimativas para o filtro de Chopin (2007) utilizando a distribuição Poisson.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.

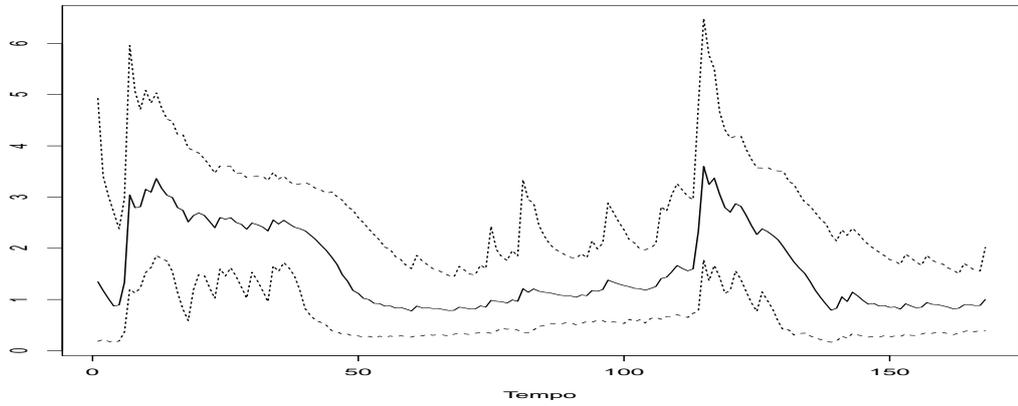


(b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição Binomial Negativa.

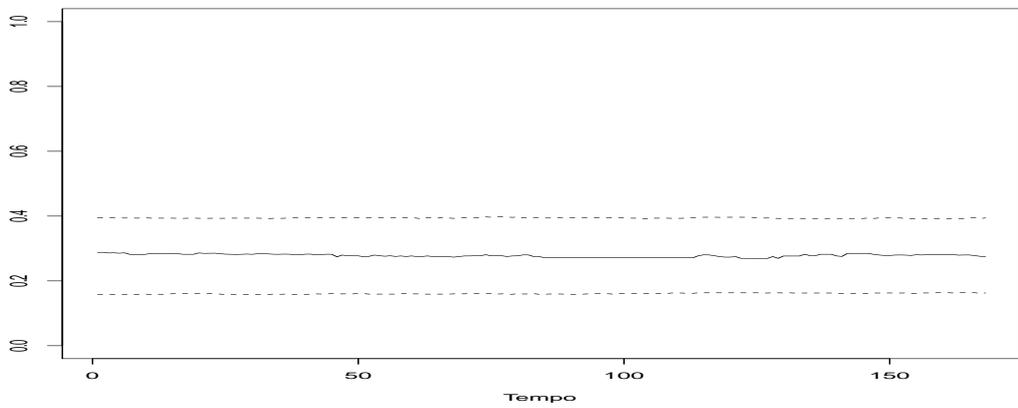


(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição Binomial Negativa.

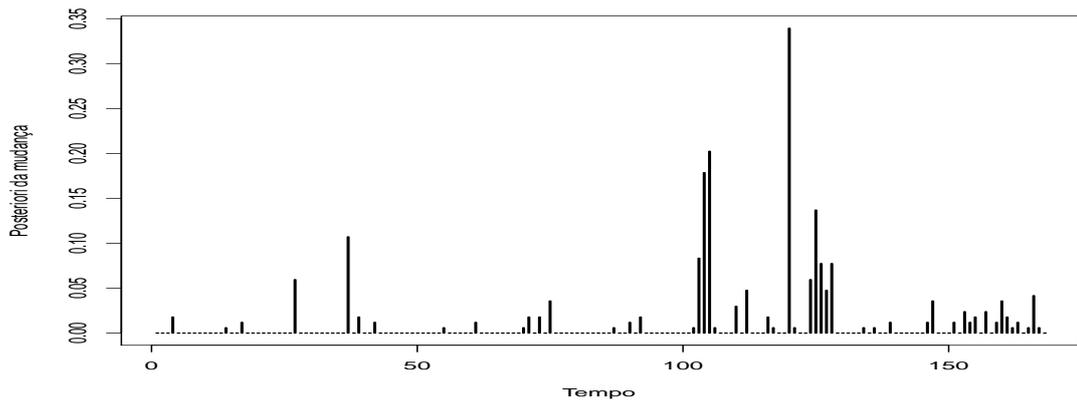
Figura 10.21: Estimativas para o filtro FChAP utilizando a distribuição Binomial Negativa.



(a) Média filtrada, utilizando o filtro FChAP, para a distribuição ZIP.

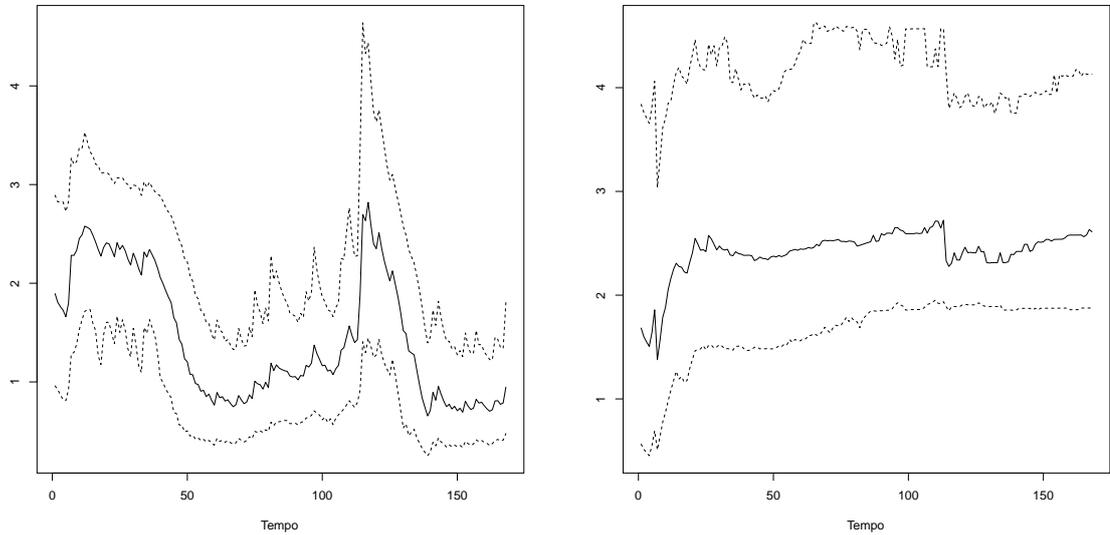


(b) Probabilidade de zero filtrada, utilizando o filtro FChAP, para a distribuição ZIP.

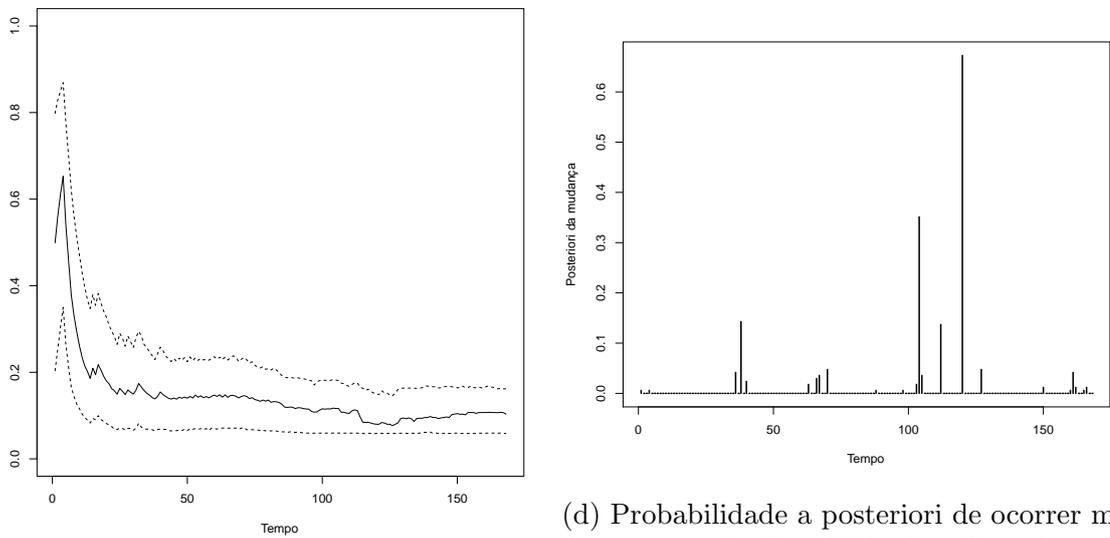


(c) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição ZIP.

Figura 10.22: Estimativas para o filtro FChAP utilizando a distribuição ZIP.



(a) Média filtrada, utilizando o filtro FChAP, (b) Dispersão filtrada, utilizando o filtro FChAP, para a distribuição ZINB.



(c) Probabilidade de zero filtrada, utilizando o filtro FChAP, para a distribuição ZINB. (d) Probabilidade a posteriori de ocorrer mudança segundo filtro FChAP utilizando a distribuição ZINB.

Figura 10.23: Estimativas para o filtro FChAP utilizando a distribuição ZINB.

Capítulo 11

Conclusão e trabalhos futuros

Nesta dissertação estudamos os principais aspectos relacionados aos modelos dinâmicos lineares, dinâmicos lineares generalizados, além dos filtros de partículas mais utilizados na prática. Para o caso dos modelos dinâmicos lineares foram estudados o processo de estimação, suas especificações, e, também, abordamos como se dá o processo de estimação na presença de parâmetros estáticos desconhecidos. Ilustramos, também, como utilizar essa metodologia no software *R* através do pacote *dlm*.

Além dos modelos dinâmicos lineares, descreveu-se, detalhadamente, os modelos dinâmicos lineares generalizados, além dos principais filtros de partículas amplamente divulgados na literatura, como os filtros *Bootstrap*, filtro de partículas auxiliar, de Liu e West, de Storvik e *Particle Learning*.

Mostrou-se, também, o modelo de regressão dinâmica proposto por McCormick et al. (2012), e os filtros de partículas propostos por Chopin (2007) e Caron et al. (2012) para o estudo de séries temporais que apresentam pontos de mudança. Tais algoritmos foram aplicados e ilustrados para a distribuição de Poisson.

Como dados de contagem podem apresentar superdispersão e/ou inflação de zeros, aplicou-se os algoritmos descritos no Capítulo 6 para as distribuições Binomial Negativa, Binomial Negativa Inflacionada de Zeros e Poisson Inflacionada de Zeros. Para tanto, como essas distribuições apresentam parâmetros estáticos, viu-se a necessidade de estender os algoritmos existentes para a estimação dos

estados e dos parâmetros estáticos. Sendo assim, nesta dissertação propusemos duas extensões inéditas, uma para o algoritmo de McCormick et al. (2012), denominada McPE (Algoritmo de McCormick com Parâmetros Estáticos), e outra, denominada FChAP, para o algoritmo de Chopin (2007).

Nos Capítulos 7, 9 e 10 foram realizados exercícios de simulação para ilustrar os algoritmos propostos nesta dissertação. Para os algoritmos baseados na metodologia proposta por McCormick et al. (2012) e McPE, viu-se que a incorporação de um fator de desconto auxilia o processo de filtragem a se adaptar a possíveis mudanças abruptas nas séries. Outro fato a ser observado é que esse algoritmo também pode ser utilizado para estimar dados estáticos (vide apêndice F). Portanto, os algoritmos de McCormick et al. (2012) e McPE podem ser vistos como uma alternativa aos MDLG e podem ser vistos, também, como generalizações dos modelos de regressão estáticos (MLG, regressão múltipla).

Através dos dados simulados e de dados reais, observou-se que o Filtro proposto por Caron et al. (2012) leva menos tempo para ser executado, se comparado ao Filtro proposto por Chopin (2007) e ao FChAP. Entretanto, os Filtros de Chopin (2007) e FChAP são mais robustos que o filtro de Caron et al. (2012), no sentido da convergência, uma vez que o filtro de Caron et al. (2012) apresenta certos problemas de convergência relacionados ao uso de funções gama. Outro fato que deve ser observado é que uma estratégia de análise para os filtros de Chopin (2007) e FChAP é executá-los mais de uma vez e utilizar os valores medianos das trajetórias como estimativas finais para os parâmetros estáticos e para os estados. De maneira geral, os filtros apresentaram resultados similares, no sentido de detectar os mesmos pontos de mudança em uma mesma região da série.

No Capítulo 10 aplicaram-se os algoritmos propostos nesta dissertação a cinco séries reais de dados de contagem. Os resultados mostram ajustes que representam boas descrições dos dados. Como era de se esperar, para os dados que apresentam superdispersão e/ou inflação de zero, segundo os critérios AIC, AICC, BIC e DIC, os modelos baseados nas distribuições Binomial Negativa, Poisson inflacionada de zeros e Binomial Negativa inflacionada de zeros, apre-

sentaram melhores ajustes, se comparados com o modelo Poisson. Para os dados reais, os filtros de Chopin (2007) e Caron et al. (2012) apresentaram resultados semelhantes. Entretanto, o filtro de Caron et al. (2012) necessita de menos tempo de processamento, por utilizar menos partículas.

Devido ao artigo de Caron et al. (2012) ser obscuro quanto a forma de implementação do filtro, e devido a problemas numéricos envolvendo a função gama, foi implementado, nesta dissertação, o filtro de Caron et al. (2012) apenas para o caso em que os dados seguem uma distribuição de Poisson. Pretende-se estende-lo para os demais modelos utilizados nesta dissertação em trabalhos futuros.

Sendo assim, as propostas de trabalhos futuros são:

- Aplicar o filtro de partículas proposto por Caron et al. (2012) para as distribuições Binomial Negativa, Binomial Negativa Inflacionada de Zeros e Poisson Inflacionada de Zeros;
- Utilizar estimação Bayesiana para estimar os parâmetros estáticos no algoritmo McPE, incorporando-se, no algoritmo, um passo MCMC, conforme utilizado no Filtro de Chopin (2007);
- Incorporar covariáveis para explicar a probabilidade de zeros estruturais para os modelos Binomial Negativo e Poisson inflacionados de zeros. Incorporar, também, covariáveis para explicar o parâmetro de dispersão no modelo Binomial Negativo, tais incorporações deverão ser feitas no algoritmo McPE;
- Estimar, dinamicamente, a probabilidade para o modelo geométrico utilizado para modelar as probabilidades de transição nos Filtros de FChAP;
- Testar diferentes modelos para modelar as probabilidades de transição para os Filtros de Chopin (2007), Caron et al. (2007) e o nosso filtro FChAP. Por exemplo, pode-se utilizar a distribuição Binomial Negativa.
- Finalização do pacote para o ajuste dos Modelos Dinâmicos Lineares Generalizados.

Referências Bibliográficas

- Achcar, J. A., Fernández-Bremauntz, A. A., Rodrigues, E. R., Tzintzum, G. (2008). Estimating the number of ozone peaks in Mexico City using a non-homogeneous Poisson Model. *Environmetrics*. 19: 469-485.
- Barry, D. & Hartigan, J. (1992). Product partition models for change point problems. *Ann. Stat.* 20: 260-279.
- Caron, F., Doucet, A., & Gottardo, R. (2012). On-line changepoint detection and parameter estimation with application to genomic data. *Stat. Comput.* 22: 579-595.
- Carvalho, C. M., Johnnes, M., Lopes, H. F. & Polson, N. (2010). Particle learning and smoothing. *Statistical Science*. 25: 88-106.
- Casella, G., Robert, C. (1996). Rao-Blackwallisation of sampling schemes. *Biometrika*, 1: 81-84.
- Chen, R. and Liu, J. S. (2000). Mixture Kalman filter. *Journal of the Royal Statistical Society Series B*. 62:493-508.
- Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*. 89: 539-552.
- Chopin, N. (2007). Dynamic detection of change points in long times series. *Ann. Inst. Math. Sci.* 59: 349-366.
- Cox, D. R. & Hinkley, D. V. (1974). *Theoretical Statistics*. Chapman and Hall. London.

- da-Silva, C. Q., Migon, H. S. & Correia, L. T. (2011). Dynamic Bayesian beta models. *Computational Statistics and Data Analysis*, 55: 2074-2089.
- da-Silva, C. Q., Rodrigues, G. S. & Migon, H. S. (2014). Dynamic Bayesian Dirichlet Models. *Communications in Statistics. Simulation and Computation*, 44: 787-818.
- Doucet, A. & Johansen, A. (2008). A Note on Auxiliary Particle Filters. *Statistics & Probability Letters*. 78: 1498-1504.
- Fearnhead, P. & Liu, Z. (2007). On-line inference for multiple change points problems. *Journal of Royal Statistical Society*, B, 69: 589-605.
- Garcia, L. P., Freitas, L. R. S. & Hofelmann, D. A. (2013). Avaliação do impacto da Lei Maria da Penha sobre a mortalidade de mulheres por agressões no Brasil, 2001-2011. *Epidemiol. Serv. Saúde [online]*. 22: 383-394.
- Gilks, W. R. & Berzuini, C. (2001). Following a moving target - Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society Series B*, 63: 127-146.
- Godsill, S. J., Doucet, A. & West, M. (2004). Monte Carlo Smoothing for Non-linear Time Series. *Journal of the American Statistical Association*. 99: 156-168.
- Gordon, N., Salmond, D., Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEEE Proceedings F. Radar Signal Process*, 140:107-113.
- Hinde, J. & Demétrio, C. G. B. (1998). Overdispersion: model and estimation. *Computational Statistics and Data Analysis*. 27, 151-170.
- Hinkley, D. V. (1970). Inference about the change-point in a sequence of random variables. *Biometrika*. 57: 1?17.
- Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5: 1-25.

- Lai, T. L. & Xing, H. (2011). A Simple Bayesian Approach to Multiple Change-Points. *Statistica Sinica*. 21: 539-569.
- Lambert, D. (1992) Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*. 34:1-14.
- Liu, J. & West, M. (2001). Combined parameters and state estimation in simulation-based filtering. In A. Doucet, N. de Freitas and N. Gordon, editors, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag.
- Lopes, H. F. & Tsay, R. S. (2011). Particle Filters and Bayesian Inference in Financial Econometrics. *Journal of Forecasting*. 30: 168-209.
- Lopes, H. F., Carvalho, C. M., Johannes, M. & Polson, N. G. (2010). Particle learning for sequential Bayesian computation. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West, editors, *Bayesian Statistics 9*. Oxford: Oxford University Press.
- McCormick, T. H., Raftery, A. E., Madigan, D. & Burd, R. S. (2012). Dynamic logistic regression and dynamic model averaging for binary classification. *Biometrics*. 68(1): 23-30.
- McCullagh, P. & Nelder, J. A. (1994). *Generalized Linear Models*, (2nd edition ed.). The University Press. Cambridge: Monographs on Statistics and Applied Probability 37.
- Nelder, J. A. & Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society Series A*. 135:370-384.
- Nemeth, C., Fearnhead, P., Mihaylova, L. (2014). Sequential Monte Carlo Methods for State and Parameter Estimation in Abruptly Changing Environments. *IEEE Transactions on Signal Processing*. 62, 1245-1255.
- Petris, G. (2010). An R package for dynamic linear models. *Journal of Statistical Software*, 36(12): 1-16.
- Petris, G., Petrone, S., & Campagnoli, P. (2009). *Dynamic Linear Models with R*. Springer.

- Pitt, M. & Sherphard, N. (1998). A Fixed Lag Auxiliary Particle Filter With Deterministic Sampling Rules. Unpublished manuscript, Nuffield College.
- Pitt, M. & Sherphard, N. (1999). Filtering via simulation: auxiliary particle filters. *Journal of the American Statistical Association*, 94:590-599.
- Poyiadjis, G., Doucet, A., Singh, S. S. (2011). Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika*, 98: 65-80.
- R development Core Team (2011). *R: A language and Environment for Statistical Computing*. R Foundation for Statistical Computing. ISBN 3-900051-07-0.
- Raftery, A. E., Akman, V.E. (1986) Bayesian Analysis of a Poisson Process with Change-Point. *Biometrika*. 73: 85-89.
- Raftery, A. E., Kárný, M. & Ettlér, P. (2010) Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*. 52: 52-66.
- Rubin, D. B. (1988). Using the SIR algorithm to simulate posterior distribution. *Bayesian Statistics*. 3: 395-402.
- Schmidt, A. M., Pereira, J. B. M. (2011). Modelling Time Series of Counts in Epidemiology. *International Statistical Review*. 79: 48-69.
- Smith, A. F. M. & Gelfand, A. E. (1992). Bayesian statistics without tears: a sampling-resampling perspective. *American Statistician*. 46: 84-88.
- Storvik, G. (2002). Particle filters for state-space models with the presence of unknown static parameters. *IEEE Transactions on Signal Processing*, 50: 281-289.
- West, M. (1992). Modelling with mixtures. In J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith, editors, *Bayesian Statistics 4*. Oxford: CLarendon.

- West, M., Harrison, P. J. & Migon, H. S. (1985). Dynamic generalized linear models and bayesian forecasting (with discussion). *Journal of the American Statistical Association*, 80: 73-97.
- West, M. & Harrison, P. J. (1997). *Bayesian Forecasting and Dynamic Models*, (2nd edition ed.). Springer.
- Whittaker, J. C. & Fruhwirth-Schnatter, S. (1994). A dynamic change-point model for detecting the onset of growth in bacteriological infections. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*. 43: 625-640.
- Yau, K. K. W., Lee, A. H. & Carrivick, P. J. W. (2004) A regression-model for time series of counts. *Biometrika*. 75: 621-629.
- Zeger, S. L. (1988) Modelling zero-inflated count series with application to occupational health. *Computer Methods and Programs in Biomedicine*. 74: 47-52.

Apêndices

A - Teoremas de Bayes

Para aqueles leitores não familiarizados com estatística bayesiana, se faz necessário a leitura desse apêndice. Aqui daremos o princípio básico para o entendimento da estatística bayesiana e conseqüentemente um melhor entendimento sobre os MDLs.

Seja x_1, \dots, x_n uma amostra aleatória de observações com função densidade de probabilidade indexada pelo parâmetro θ , $p(x_i|\theta)$. De posse das observações, a função de verossimilhança, que é uma função de θ , é dada por $l(\theta) = p(x_1, \dots, x_n) = p(\mathbf{x}|\theta)$.

Diferentemente da estatística clássica, no qual o parâmetro é considerado uma constante, na estatística bayesiana existe uma incerteza sobre o parâmetro θ , e essa incerteza é caracterizada por meio de uma função densidade de probabilidade $p(\theta)$, tal densidade é chamada de distribuição *a priori*. Tal incerteza corresponde ao conhecimento do analista à respeito do parâmetro θ .

O foco da estatística bayesiana é revisar o conhecimento a respeito de θ após observar um fenômeno, ou seja, após observar \mathbf{x} . Essa revisão é dada pela distribuição *a posteriori* de θ , esta por sua vez é obtida através do *teorema de Bayes*

$$p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})},$$

onde

$$p(\mathbf{x}) = \int p(\mathbf{x}|\theta)p(\theta)d\theta \quad (11.1)$$

Usualmente é usado a seguinte notação para representar o cálculo da distribuição a *posteriori*

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

o sinal \propto significa a menos de uma constante que não depende de θ . Ou seja, em alguns casos podemos caracterizar completamente a distribuição a *posteriori* apenas conhecendo a verossimilhança e a distribuição a *priori*, todas as quantidades que não depende de θ podem ser condensadas no sinal \propto .

B- Método SIR

Para simplificar a explicação do método SIR, denote por $p(\theta)$ a distribuição a *priori* de θ , $p(y|\theta)$ a verossimilhança, e $p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$ a distribuição a *posteriori* de θ .

O SIR (Rubin, 1988) é um método com o qual amostra-se $\theta^1, \dots, \theta^R$ a partir da *priori* $p(\theta)$ e, então, associa, a cada um dos valores selecionados θ^j , um peso π_j , em que $w_j = p(y|\alpha^j)$ e $\pi_j = \frac{w_j}{\sum_{i=1}^R w_i}$, $j = 1, \dots, R$. Observe que podemos reescrever a distribuição a *posteriori* de θ da seguinte forma

$$\begin{aligned} p(\theta|y) &= \frac{p(y|\theta)p(\theta)}{p(y)} \\ &= \left[\frac{p(y|\theta)}{p(y)} \right] p(\theta) \\ &\approx \sum_{j=1}^R \pi_j \hat{p}(\theta_j). \end{aligned}$$

Então, a amostra, ponderada pelos w'_j s, converge, quando $R \rightarrow \infty$, para uma

amostra da distribuição alvo $p(\theta|y)$, uma vez que

$$\frac{1}{R} \sum_{i=1}^R w_i \xrightarrow{P} p(y)$$

com $p(y) = \int p(y|\theta)p(\theta)d\theta = E_{\theta}(p(y|\theta)) \approx \frac{1}{R} \sum_{j=1}^R p(y|\theta^j)$.

C - Regressão dinâmica Poisson bayesiana

Nesta seção serão ilustrados os cálculos para obter-se as expressões do Capítulo

6. Para tanto, considere as seguintes expressões introduzidas na Seção 6.1

$$\begin{aligned} p(y_t|\theta_t) &= \frac{\lambda_t^{y_t} \exp(-\lambda_t)}{y_t!}; \\ p(\theta_t|D_t) &= \frac{\exp\left(-\frac{1}{2R_t}(\theta_t - \hat{\theta}_{t-1})^2\right)}{\sqrt{2\pi R_t}}, \end{aligned} \quad (11.2)$$

onde $\lambda = \exp(x_t\theta_t)$. As derivadas das densidades acima são dadas por

$$\begin{aligned} p'(y_t|\theta_t) &= p(y_t|\theta_t) [y_t x_t - x_t \exp(x_t\theta_t)]; \\ p'(\theta_t|D_t) &= -p(\theta_t|D_t) \frac{1}{R_t} (\theta_t - \hat{\theta}_{t-1}). \end{aligned} \quad (11.3)$$

Para obter-se as expressões da Seção 6.2, devemos derivar duas vezes $l(\theta_t) = \log[p(y_t|\theta_t)p(\theta_t|D_t)]$. A primeira derivada de $l(\theta_t)$ é dada por

$$\begin{aligned} Dl(\theta_t) &= \frac{1}{p(y_t|\theta_t)p(\theta_t|D_t)} [p'(y_t|\theta_t)p(\theta_t|D_t) + p(y_t|\theta_t)p'(\theta_t|D_t)] \\ &= \frac{p'(y_t|\theta_t)}{p(y_t|\theta_t)} + \frac{p'(\theta_t|D_t)}{p(\theta_t|D_t)}. \end{aligned} \quad (11.4)$$

Usando as equações (11.2) e (11.3), temos que

$$Dl(\theta_t) = x_t (y_t - \exp(x_t\theta_t)) - \frac{(\theta_t - \hat{\theta}_{t-1})}{R_t}. \quad (11.5)$$

A segunda derivada, $Dl(\theta)$, é facilmente obtida a partir da equação (11.5)

$$D^2l(\theta_t) = -x_t x_t' \exp(x_t \theta_t) - \frac{1}{R_t}. \quad (11.6)$$

Fazendo $\theta_t = \hat{\theta}_{t-1}$ temos

$$\begin{aligned} Dl(\hat{\theta}_{t-1}) &= x_t(y_t - \hat{y}_t); \\ D^2l(\hat{\theta}_{t-1}) &= -x_t x_t' \hat{y}_t - R_t^{-1}, \end{aligned}$$

que são as expressões obtidas na seção 7. Lembrando que $\hat{y}_t = \exp(x_t \hat{\theta}_{t-1})$.

D - Aproximação de Laplace

A idéia aqui é aproximar a integral que possui a seguinte forma

$$I(t) = \int \exp(-Nh(x)) dx,$$

onde N é o tamanho da amostra. Utilizando-se da expansão de Taylor de primeira ordem, temos que

$$I(N) \approx \sqrt{2\pi\sigma} N^{-1/2} \exp(-Nh(\hat{x})),$$

em que

$$\hat{x} = \operatorname{argmax}_x h(x) \text{ e } \sigma^2 = \frac{1}{h(\hat{x})}.$$

No caso multivariado temos que

$$\int \exp(-Nh(x)) dx \approx \exp(-Nh(\hat{x})) (2\pi)^{d/2} |\Sigma|^{1/2} N^{d/2},$$

com x d -dimensional, $\Sigma = (D^2h(\hat{x}))^{-1}$.

¹ $D^2h(\hat{x})$ é a inversa da matriz Hessiana de h avaliada em \hat{x}

Uma aplicação dessa aproximação está na obtenção da verossimilhança marginal

$$M = \int p(x|\theta)\pi(\theta)d\theta = \int \exp \left[-N \left(-\frac{1}{N} \log P(x|\theta) - \frac{1}{N} \log \pi(\theta) \right) \right] d\theta,$$

fazendo $h(\theta) = -\frac{1}{N} \log P(x|\theta) - \frac{1}{N} \log \pi(\theta)$ e utilizando a aproximação de Laplace, podemos aproximar M da seguinte forma

$$M \approx P(x|\hat{\theta})\pi(\hat{\theta})(2\pi)^{d/2}|\Sigma|^{1/2}N^{-d/2},$$

com $\hat{\theta} = \operatorname{argmax}_{\theta} h(\theta)$.

E - Equação de Transição

Nesta seção iremos demonstrar a equação (7.14) localizada na Seção 7.2 do Capítulo 7.

Para o caso em que $j = i$, e utilizando a expressão (7.14), tem-se

$$\begin{aligned} P(C_t = i|C_{t-1} = i) &= \frac{P(\text{Mudança em } i \text{ e nenhuma até } t-1)}{P(\text{Mudança em } i \text{ e nenhuma até } t-2)} \\ &= \frac{P(\text{Nenhuma mudança até } t-1|\text{Mudança em } i)P(\text{Mudança em } i)}{P(\text{Nenhuma mudança até } t-2|\text{Mudança em } i)P(\text{Mudança em } i)} \\ &= \frac{P(\text{Mudança em } t \text{ ou } t+1 \text{ ou } \dots|\text{Mudança em } i)}{P(\text{Mudança em } t-1 \text{ ou } t \text{ ou } \dots|\text{Mudança em } i)} \\ &= \frac{\sum_{u=t}^{\infty} P(\text{Mudança em } u|\text{Mudança em } i)}{\sum_{u=t-1}^{\infty} P(\text{Mudança em } u|\text{Mudança em } i)} \\ &= \frac{\sum_{u=t}^{\infty} h(u-i)}{\sum_{u=t-1}^{\infty} h(u-i)} = \frac{\sum_{z=t-i}^{\infty} h(z)}{\sum_{z=t-i-1}^{\infty} h(z)} = \frac{1-H(t-i-1)}{1-H(t-i-2)}. \end{aligned} \quad (11.7)$$

No caso em que $i \neq j$, tem-se que $j = t-1$, sendo assim, tem-se que

$$P(C_t = t-1|C_{t-1} = i) = 1 - P(C_t = i|C_{t-1} = i) = \frac{H(t-1-i) - H(t-2-i)}{1 - H(t-2-i)}. \quad (11.8)$$

Finalizando a demonstração.

F - Modelos de regressão estática

F.1 - Modelo Binomial Negativo

Para ilustrar o algoritmo de McCormick com parâmetros estáticos (McPE) aplicado a um modelo de regressão Binomial Negativa estática, simulou-se um modelo o qual, para $t = 1, 2, \dots, 1000$, o preditor linear é dado por

$$\log(\mu_t) = -0.3x + 0.5,$$

em que $X \sim N(0, 1)$ e $y_t \sim \text{BN}(k = 1, \exp(-0.3x + 0.5))$. Embora os dados gerados sejam independentes, utilizou-se o modelo de série temporal para analisá-los.

A programação utilizada para a geração dos dados é dada por

Simulação

```
x <- rnorm(500)
2 k=1
  mu <- exp(-0.3*x+0.5)
4 y = NULL
  ybin = NULL
6 for (i in 1:500){
  y[i]=rnbinom(n=1, mu=mu[i], size=k)
8  #y[i]=rpois(n=1, lambda=mu[i])
}
```

A programação utilizada para a análise do conjunto de dados, utilizando o algoritmo McPE, é dada por

Ajuste

```
1 exemplo <- NB.sm(Y=y, X=x, intercept=TRUE, lambda0=1, c=1)
  names(exemplo)
3
  par(mfrow=c(2,1))
5 plot(exemplo$estimates[,1],type="l", ylim=c(-2,2))
  lines(exemplo$estimates[,1]-2*exemplo$sderror[,1], col="blue")
7 lines(exemplo$estimates[,1]+2*exemplo$sderror[,1], col="blue")
  abline(h=1, lty=2, col="red")
9
  plot(exemplo$estimates[,2],type="l", ylim=c(-2,2))
```

```

11 lines(exemplo$estimates[,2]-2*exemplo$sderror[,2], col="blue")
   lines(exemplo$estimates[,2]+2*exemplo$sderror[,2], col="blue")
13 abline(h=-0.2, lty=2, col="red")

15 summary(glm(y~cbind(x1,x2), family=NB))
   exemplo$estimates[length(y),]
17 exemplo$sderror[length(y),]

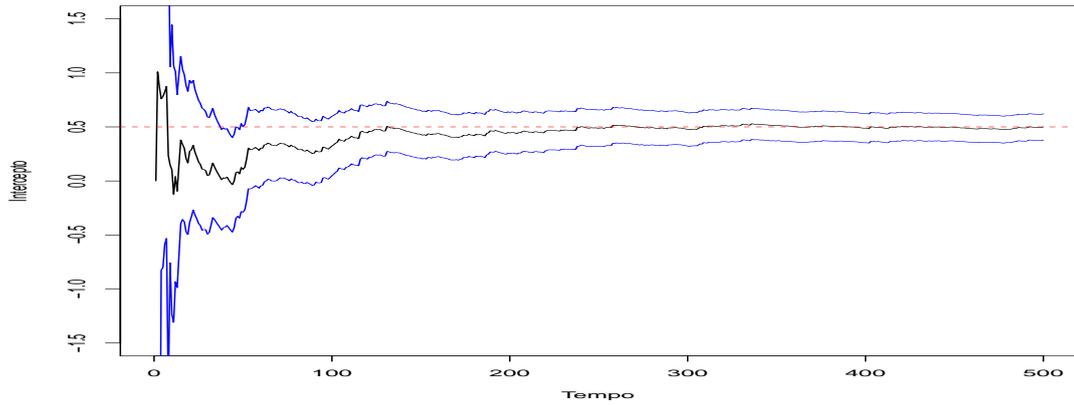
```

Os resultados da análise de regressão estática Binomial Negativa estão ilustrados na Figura 11.1. Nota-se que os parâmetros estimados convergem para os valores próximos dos verdadeiros.

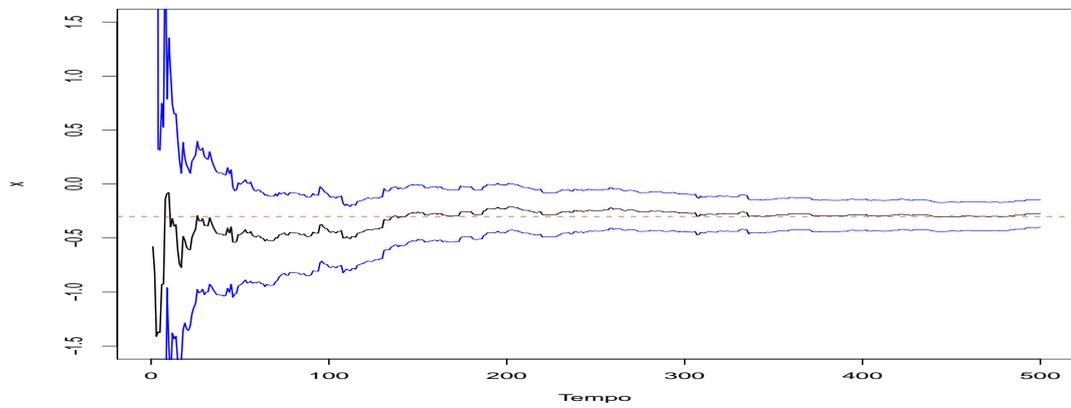
A Tabela 11.1 obtidos com o modelo Binomial Negativo estático mostra os valores dos parâmetros estimados no ponto $T = 500$.

Tabela 11.1: Estimativa dos coeficientes do modelo BN estático.

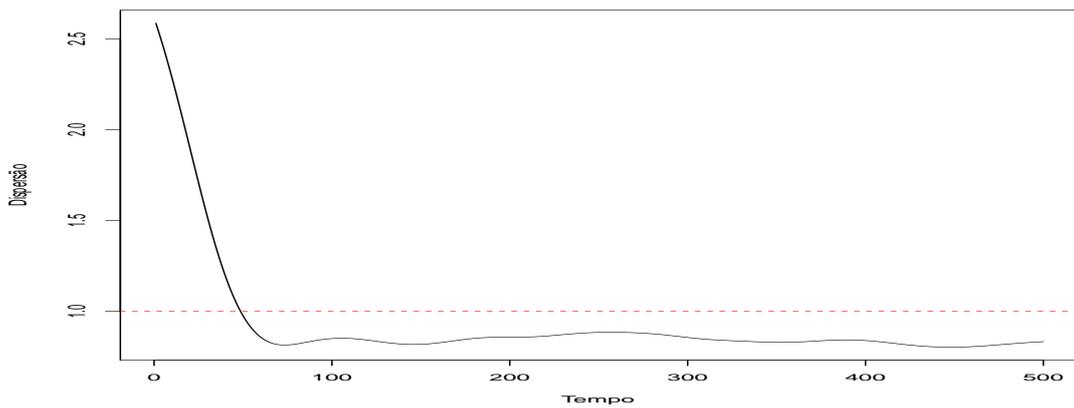
Modelo	COEFICIENTE ESTIMADO	ESTIMATIVA PONTUAL	ERRO PADRÃO
MLG BN dinâmico	INTERCEPTO	0.4303781	0.05321851
	β	-0.3372079	0.05427982
	DISPERSÃO	0.854592	-



(a) Intercepto



(b) Coeficiente angular



(c) Dispersão

Figura 11.1: Estimativas dos parâmetros de regressão Binomial Negativa estática (linhas pretas) com seus respectivos intervalos de credibilidade de 95% (linhas azuis). As linhas tracejadas vermelhas indicam o valor dos parâmetros populacionais. Estimativas obtidas com o auxílio do algoritmo McPE.

F.2 - Modelo ZIP

Para ilustrar o algoritmo McPE para o modelo de regressão Poisson inflacionado de zeros estático, simulou-se um modelo de tamanho $T = 1000$ com o seguinte preditor linear

$$\log(\lambda_t) = -0.2x - 1,$$

onde $X \sim U(0, 1)$ e $y_t \sim \text{ZIP}(p = 0.1, \lambda_t = \exp(-0.2x - 1))$.

A programação utilizada para a geração dos dados é dada por

Simulação

```
1 set.seed(456)
  x <- runif(1000)
3 p=0.1
  mu <- exp(-0.2*x-1)
5 y = NULL
  ybin = NULL
7 for (i in 1:1000){
  y[i]=rZIP(n=1, mu=mu[i], sigma=p)
9 }
```

A programação utilizada para a análise dos dados é dada por

Ajuste

```
1 exemplo <- ZIP.sm(Y=y, X=x, intercept=TRUE, lambda0=1, c=1)
  names(exemplo)
3
  par(mfrow=c(1,2))
5 plot(exemplo$estimates[,1], type="l", ylim=c(-2,2), xlab="Tempo",
  ylab="Intercepto")
7 lines(exemplo$estimates[,1]+2*exemplo$estimates[,1], col="blue")
  lines(exemplo$estimates[,1]-2*exemplo$estimates[,1], col="blue")
9 abline(h=-1, col="red", lty=2)

11 plot(exemplo$estimates[,2], type="l", ylim=c(-2,2), xlab="Tempo", ylab="x")
  lines(exemplo$estimates[,2]+2*exemplo$estimates[,2], col="blue")
13 lines(exemplo$estimates[,2]-2*exemplo$estimates[,2], col="blue")
  abline(h=-0.2, col="red", lty=2)
15
  theta[n,]
17 sdtheta[n,]
  p[n]
```

O resultado da análise da regressão estática Poisson inflacionado de zero está ilustrado na Figura 11.2. Observa-se que as estimativas dos parâmetros obtidas para o modelo estático são razoáveis e estão próximos dos valores verdadeiros.

A Tabela 11.2 traz os valores dos parâmetros estimados no ponto $T = 1000$. O parâmetro estático foi estimado por máxima verossimilhança.

Tabela 11.2: Estimativa dos coeficientes do modelo ZIP estático

Modelo	COEFICIENTE ESTIMADO	ESTIMATIVA PONTUAL	ERRO PADRÃO
MLG ZIP dinâmico	INTERCEPTO	-0.9776964	0.1123192
	β	-0.1819508	0.1892570
	p	0.1018876	-

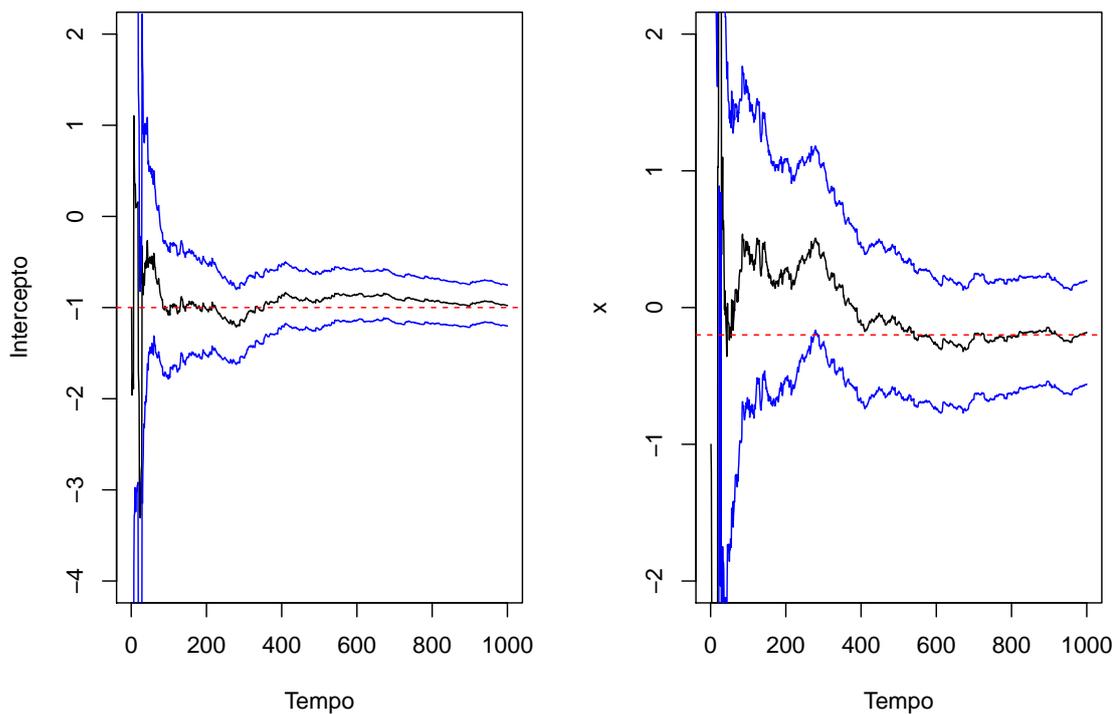


Figura 11.2: Estimativas dos parâmetros de regressão ZIP (linhas pretas), via algoritmo McPE, com seus respectivos intervalos com 95% de confiança (linhas azuis). As linhas tracejadas vermelhas indicam o valor dos parâmetros populacionais.

F.3 - Modelo ZINB

Para ilustrar o uso do algoritmo McPE para o modelo de regressão Binomial Negativo inflacionado de zeros estático, simulou-se um modelo de tamanho $T = 1000$ com o seguinte preditor linear

$$\log(\mu_t) = -0.2x - 1,$$

onde $X \sim N(0, 1)$ e $y_t \sim \text{ZINB}(p = 0.15, k = 1, \mu_t = \exp(-0.2x - 1))$.

A programação utilizada para a geração dos dados é dada a seguir

Simulação

```
set.seed(456)
2 x <- rnorm(1000)
  mu1 <- exp(-0.2*x-1)
4 y = NULL
  ybin = NULL
6 for (i in 1:length(x)){
  y[i]=rzinb(n=1, k=1, lambda=mu1[i], omega=0.15)
8 #y[i]=rpois(n=1, lambda=mu[i])
}
```

A programação utilizada para o ajuste do modelo é dada a seguir

Ajuste

```
1 exemplo <- ZINB.sm(Y=y, X=x, intercept=TRUE, lambda0=1, c=1)
  names(exemplo)
3
  par(mfrow=c(1,2))
5 plot(exemplo$estimates[,1], type="l", ylim=c(-2,2), xlab="Tempo",
  ylab="Intercepto")
7 lines(exemplo$estimates[,1]+2*exemplo$estimates[,1], col="blue")
  lines(exemplo$estimates[,1]-2*exemplo$estimates[,1], col="blue")
9 abline(h=-1, col="red", lty=2)

11 plot(exemplo$estimates[,2], type="l", ylim=c(-2,2), xlab="Tempo", ylab="x")
  lines(exemplo$estimates[,2]+2*exemplo$estimates[,2], col="blue")
13 lines(exemplo$estimates[,2]-2*exemplo$estimates[,2], col="blue")
  abline(h=-0.2, col="red", lty=2)
15
  theta[n,]
17 sdtheta[n,]
  disp[n]
19 p[n]
```

O resultado da análise de regressão estática Binomial Negativa inflacionada de zeros está ilustrado na Figura 11.3, em que se observa um ajuste muito razoável, já que os parâmetros estimados estão próximos dos valores verdadeiros.

Na Tabela 11.3 apresenta-se os valores dos parâmetros estimados no ponto $T = 1000$. Os parâmetros estáticos foram estimados por máxima verossimilhança.

Tabela 11.3: Estimativa dos coeficientes do modelo ZINB estático

Modelo	COEFICIENTE ESTIMADO	ESTIMATIVA PONTUAL	ERRO PADRÃO
MLG ZINB dinâmico	INTERCEPTO	-0.8344806	0.0704261
	β	-0.2057458	0.07243260
	k	1.105066	-
	p	0.1119776	-

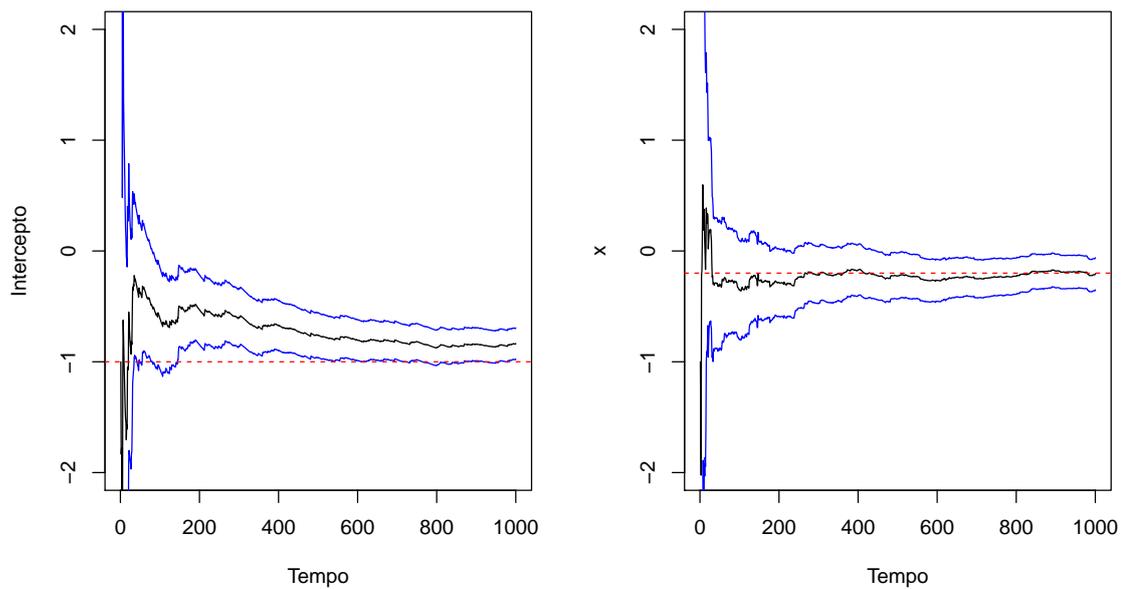


Figura 11.3: Estimativas dos parâmetros de regressão ZINB (Linhas pretas), via algoritmo McPE, com seus respectivos intervalos com 95% de confiança (Linhas azuis). As linhas tracejadas vermelhas indicam o valor dos parâmetros populacionais.