Segmentação e caracterização de eventos em sinais acústicos

Paulo do Canto Hubert Junior

Tese apresentada
ao
Instituto de Matemática e Estatística
da
Universidade de São Paulo
para
obtenção do título
de
Doutor em Ciências

Programa: Matemática Aplicada
Orientador: Prof. Dr. Júlio Michael Stern

Durante o desenvolvimento deste trabalho o autor recebeu auxílio financeiro da CAPES

São Paulo, junho de 2018

Segmentação e caracterização de eventos em sinais acústicos

Esta versão da tese contém as correções e alterações sugeridas pela Comissão Julgadora durante a defesa da versão original do trabalho, realizada em 23/05/2018. Uma cópia da versão original está disponível no Instituto de Matemática e Estatística da Universidade de São Paulo.

Comissão Julgadora:

- Prof. Dr. Julio Michael Stern (orientador) - IME-USP
- Prof. Dr. Linilson Padovese - EP-USP
- Profª. Drª. Laura Letícia Ramos Rifo - IMECC-UNICAMP
- Prof. Dr. Victor Fossaluza - IME-USP
- Prof. Dr. Renato Vicente - IME-USP

Agradecimentos

À Juliana, pela paciência amorosa e apoio inestimável.

Ao professor Julio Stern, pela generosidade e confiança.

Ao professor Linilson Padovese, pela oportunidade de trabalhar com os dados do OceanPod, mas acima de tudo pelas portas científicas que abriu, e pelo que foi uma autêntica co-orientação, informal porém importantíssima.

Aos professores Victor Fossaluza e Gabriel Haeser, pela atenção no processo de qualificação e pelo apoio em diversos momentos ao longo desta trajetória.

Ao professor Carlinhos, pela inspiração; ao professor Márcio Diniz, pelo aconselhamento.

Dedicatória

Para a Bela, que será; para a Juliana, que já é.

Que os méritos de nossa prática
se estendam a todos os seres
e que possamos,todos e todas, nos tornar
o caminho iluminado.

Resumo

Hubert, P. **Um sistema para segmentação e categorização não-supervisionada de eventos em sinais acústicos.** 2018. 120 f. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2018.

A análise de sinais é uma área de intensa pesquisa e inúmeras aplicações, em contextos tão diversos como a transcrição musical automática, a cardiologia e a prospecção de óleo e gás. Um sinal, nestes contextos, é definido como uma quantidade qualquer que varia ao longo do tempo em resposta a um estímulo causado, voluntária ou involuntariamente, por um agente emissor. O sinal é transmitido através de um meio imperfeito (i.e., é contaminado por ruído) e é posteriormente captado por um receptor que deseja extrair dele informações sobre o agente causador. No presente trabalho, lidamos com sinais acústicos de longa duração, obtidos em ambients subaquáticos, e em que há pouca informação prévia sobre esses potenciais estímulos; em particular, não se sabe em que momentos do tempo os estímulos estão presentes ou ausentes, e, quando presentes, não se sabe qual sua causa específica (logo não se sabe qual sua forma funcional). Nosso objetivo então será extrair seções contíguas do sinal em que haja evidência da presença de algum estímulo; estas seções, ou segmentos, podem então ser agrupadas e categorizadas, de tal forma que um grupo de segmentos contenha amostras distintas de um mesmo evento (i.e., de uma mesma classe de emissores). A partir daí, os eventos podem ser inspecionados e métodos supervisionados de classificação podem ser construídos.

Palavras-chave: processamento de sinais acústicos, acústica submarina, segmentação de sinais, detecção de eventos, categorização de eventos, modelos bayesianos.

Abstract

Hubert, P. **A system for unsupervised segmentation and characterization of events in acoustic signals.**. 2018. 120 f. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2018.

Signal analysis is an area of intense research and inumerous applications, in such diverse contexts as automatic music transcription, cardiology and oil and gas prospection. A signal, in these contexts, is defined as a quantity that varies with time in response to some stimulus caused, voluntarily or otherwise, by an emitting agent. The signal is transmitted through an imperfect medium (i.e. it is contaminated by noise) and is then captured by a receptor that wants to extract from it information about the causing agent. In this work, we deal with long duration acoustic signals, obtained in a maritime environment. Very little is known about the events present in the signal; in particular, we do not know the moments of time when a causing agent is acting, and if an agent is acting, we also do not know what kind of agent it is (i.e., we do not know the functional form of the stimulus). Our goal is then to first extract contiguous sections of the signal where there is strong evidence of the presence of a stimulus; these sections, or segments, can then be clustered and categorized, in order to form homogeneous groups of segments that can be thought of as samples of a given event (i.e., a class of emitting agents). Given this categorization, the events can be inspected and named, and supervised methods of classification can be build.

Keywords: acoustic signal processing, subaquatic acoustics, signal segmentation, event detection, event categorization, bayesian models.

Sumário

Lista de Abreviaturas xiii

Lista de Figuras xv

Lista de Tabelas xix

1 Introdução 1

1.1 Motivação: análise de sinais acústicos submarinos . . . 1
1.2 Análise Bayesiana de sinais . . . 2
1.2.1 Modelo . . . 2
1.2.2 Estimação de sinais . . . 3
1.2.3 Seleção de modelos . . . 9
1.3 Conclusão . . . 11

2 FBST para detecção binária de sinais 13

2.1 FBST . . . 14
2.2 Sinal completamente conhecido . . . 15
2.3 Sinal parcialmente conhecido . . . 16
2.4 FBST para detecção de embarcações . . . 18
2.5 Conclusão . . . 22

3 Segmentação não-supervisionada de sinais 25

3.1 Modelo para a segmentação do sinal . . . 26
3.2 Critérios de parada do algoritmo de segmentação . . . 31
3.2.1 Critério baseado na estatística F . . . 33
3.2.2 Critério da razão de verossimilhanças com significância adaptativa . . . 34
3.2.3 Critério baseado no FBST . . . 35
3.3 Algoritmo sequencial para segmentação do sinal . . . 37
3.3.1 Critério F . . . 39
3.3.2 Critério da Razão de Verossimilhanças . . . 44
3.3.3 Critério baseado no FBST . . . 48
3.4 Segmentação de sinais e *peak detection* . . . 53
3.5 Implementação otimizada com Python + Cython . . . 61
3.5.1 Estimação do ponto de corte . . . 61
3.5.2 Cálculo da evidência . . . 62
3.5.3 Módulo *bayeseg* . . . 64

3.6 Sensibilidade aos parâmetros 66
3.7 Calibração em amostras reais 71
3.8 Conclusão 73

4 Categorização de segmentos **75**
4.1 Caracterização dos segmentos 75
4.1.1 Espectro médio 76
4.1.2 Chirpograma 78
4.2 Medidas de similaridade 80
4.3 Categorização de sinais simulados 83
4.4 Categorização de segmentos do OceanPod 90
4.4.1 Agrupamento por k-médias e k-medóides 92
4.5 Conclusão 97

5 Aplicação: análise do sinal do Parque Estadual da Laje de Santos **99**
5.1 Resultados da segmentação 99
5.2 Resultados da categorização 102
5.3 Conclusão 108

Referências Bibliográficas **109**

Lista de Abreviaturas

LACMAM	*Laboratório de Acústica e Meio Ambiente*
FBST	*Full Bayesian Significance Test*
MAXENT	*Maximum Entropy*
SNR	*Signal-to-noise ratio*
SeqSeg	*Sequential Segmentation Algorithm*
MCMC	*Markov Chain Monte Carlo*
MH	*Metropolis-Hastings*
GSL	*GNU Scientific Library*
FFT	*Fast Fourier Trasnform*
DFT	*Discrete Fourier Trasnform*
PD	*Peak Detection*
GSL	*GNU Scientific Library*
MAP	*Máximo a Posteriori*
MV	*Máxima verossimilhança*
GBR	*Estatística de Gelman-Brook-Rubin*
DTW	*Dynamic Time Warping*
DE	*Distância Euclidiana*
FRO	*Norma de Frobenius*

Lista de Figuras

1.1 Sinal simulado, com e sem adição do ruído Gaussiano 7
1.2 Espectro estimado via DFT 8
1.3 Log-posteriori para $\phi = 0$ 8
1.4 Log-posteriori num reticulado de valores para ϕ e ω 8

2.1 Evidência para a presença do sinal em dados simulados 16
2.2 Evidência para a presença do sinal em dados simulados: parâmetros desconhecidos . 18
2.3 Espectrograma da passagem de uma embarcação de grande porte 20
2.4 Espectrograma da passagem de uma embarcação de pequeno porte 20
2.5 Evidência contra H_0 - $m = 7$ (lado esquerdo) and $m = 10$ (lado direito) 22

3.1 *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 1,1$ 27
3.2 *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 1,5$ 28
3.3 *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 2$ 28
3.4 *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 1,1$ 28
3.5 *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 1,5$ 29
3.6 *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 2$ 29
3.7 *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 1,1$ 29
3.8 *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 1,5$ 30
3.9 *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 2$ 30
3.10 Segmentação estimada para um sinal com dois pontos de corte 32
3.11 Segmentação estimada para um sinal com dois pontos de corte, caso 2 32
3.12 Densidade para a distribuição de Laplace 36
3.13 Espectrograma para o sinal do dia 30/01/2015 39
3.14 Espectrograma para o sinal do dia 02/02/2015 39
3.15 Espectrograma para o sinal do dia 08/02/2015 39
3.16 Critério F com $\alpha = 1e^{-20}$ em 30/01/2015 40
3.17 Critério F com $\alpha = 1e^{-15}$ em 30/01/2015 40
3.18 Critério F com $\alpha = 1e^{-10}$ em 30/01/2015 41
3.19 Critério F com $\alpha = 1e^{-20}$ em 02/02/2015 41
3.20 Critério F com $\alpha = 1e^{-15}$ em 02/02/2015 42
3.21 Critério F com $\alpha = 1e^{-10}$ em 02/02/2015 42
3.22 Critério F com $\alpha = 1e^{-20}$ em 08/02/2015 43
3.23 Critério F com $\alpha = 1e^{-15}$ em 08/02/2015 43
3.24 Critério F com $\alpha = 1e^{-10}$ em 08/02/2015 44

3.25 Critério RV com $\alpha = 1e^{-20}$ em 30/01/2015 44
3.26 Critério RV com $\alpha = 1e^{-15}$ em 30/01/2015 45
3.27 Critério RV com $\alpha = 1e^{-10}$ em 30/01/2015 45
3.28 Critério RV com $\alpha = 1e^{-20}$ em 02/02/2015 46
3.29 Critério RV com $\alpha = 1e^{-15}$ em 02/02/2015 46
3.30 Critério RV com $\alpha = 1e^{-10}$ em 02/02/2015 47
3.31 Critério RV com $\alpha = 1e^{-20}$ em 08/02/2015 47
3.32 Critério RV com $\alpha = 1e^{-15}$ em 08/02/2015 48
3.33 Critério RV com $\alpha = 1e^{-10}$ em 08/02/2015 48
3.34 Critério FBST com $\beta = 1e^{-1}$ em 30/01/2015 49
3.35 Critério FBST com $\beta = 3e^{-5}$ em 30/01/2015 49
3.36 Critério FBST com $\beta = 1e^{-5}$ em 30/01/2015 50
3.37 Critério FBST com $\beta = 1e^{-1}$ em 02/02/2015 50
3.38 Critério FBST com $\beta = 3e^{-5}$ em 02/02/2015 51
3.39 Critério FBST com $\beta = 1e^{-5}$ em 02/02/2015 51
3.40 Critério FBST com $\beta = 1e^{-1}$ em 08/02/2015 52
3.41 Critério FBST com $\beta = 3e^{-5}$ em 08/02/2015 52
3.42 Critério FBST com $\beta = 1e^{-5}$ em 08/02/2015 53
3.43 Resultado do algoritmo SeqSeg, $\beta = 3e^{-5}$ 59
3.44 Resultado do algoritmo SeqSeg, $\beta = 1e^{-5}$ 59
3.45 Resultado do algoritmo de Palshikar, $h = 3$ 60
3.46 Resultado do algoritmo de Palshikar, $h = 5$ 60
3.47 Número de segmentos para diferentes valores de β ($\alpha = 0,1$) 69
3.48 Número de segmentos para diferentes valores de β ($\alpha = 0,1$) 70
3.49 Número de segmentos para diferentes valores de α ($\beta = 0,1$) 70
3.50 Número de segmentos em função de β 72

4.1 DFTs médias de 1 segundo, segmentos da amostra do dia 08/02/2015 77
4.2 DFTs médias de 1 segundo, segmentos da amostra do dia 22/02/2015 78
4.3 Chirpograma, segmentos da amostra do dia 08/02/2015 79
4.4 Chirpograma, segmentos da amostra do dia 22/02/2015 80
4.5 Dynamic Time Warping 81
4.6 Sinais simulados (sinais da mesma categoria na mesma linha) 84
4.7 DFT média de 1 segundo dos sinais simulados (sinais da mesma categoria na mesma linha) 85
4.8 Chirpogramas dos sinais simulados (sinais da mesma categoria na mesma linha) . . . 86
4.9 Dendrogramas: sinais simulados 88
4.10 Silhuetas médias para os sinais simulados 89
4.11 Dendrogramas: sinais do OceanPod 90
4.12 Número de categorias como função do ponto de corte 91
4.13 Silhuetas médias para os segmentos do OceanPod 92
4.14 Silhuetas médias para os segmentos do OceanPod: DTWChirp 92
4.15 Silhuetas médias para os segmentos do OceanPod 93
4.16 Medóides para $k = 2$ 94

4.17 Medóides para $k = 3$ 94
4.18 Silhuetas médias para os segmentos do OceanPod, DTWChirp 95
4.19 Comparação DTWDFT e DTWChirp 96

5.2 Número de segmentos por hora de início 100
5.3 Histograma - duração dos segmentos 101
5.4 Duração dos segmentos por hora de início 101
5.6 Silhuetas médias para os segmentos do Parque da Laje 103
5.7 Índice de Rand para DTWDFT e DTWChirp 103
5.8 DFTs dos medóides, $k = 8$, método DTWDFT 104
5.9 DFTs dos medóides, $k = 8$, método DTWChirp 104
5.10 Chirpogramas dos medóides, $k = 8$, método DTWDFT 105
5.11 Chirpogramas dos medóides, $k = 8$, método DTWChirp 105
5.12 Duração dos segmentos por categoria, DTWDFT 106
5.13 Duração dos segmentos por categoria, DTWChirp 106
5.14 Potência dos segmentos por categoria, DTWDFT 107
5.15 Potência dos segmentos por categoria, DTWChirp 107

Lista de Tabelas

3.1 Combinações para tamanho do evento, SNR mínimo e taxa de falsos positivos (significância) 35
3.2 Comparação dos algoritmos em um sinal simulado 56
3.3 Comparação dos algoritmos em sinais reais; ver o texto para detalhes 58
3.4 Efeito da resolução temporal na avaliação do ponto de corte 62
3.5 Análise dos valores de evidência por MCMC 65
3.6 Resultados para $\alpha = 0.1$ 67
3.7 Resultados para $\alpha = 0.5$ 67
3.8 Resultados para $\alpha = 0.9$ 68
3.9 Resultados para $\alpha = 0.99$ 68
3.10 Segmentação das amostras reais 71

4.1 Parâmetros de cada categoria para os segmentos simulados 83
4.2 Categorização por audição 95

5.1 Parâmetros utilizados na análise 99
5.2 Número de segmentos por categoria 104

Capítulo 1

Introdução

O problema de processamento de sinais é um problema antigo e muito estudado. De modo geral, um sinal é uma quantidade qualquer que varia no espaço ou no tempo (ou ambos), de tal forma que esta variação pode carregar informação sobre algum fenômeno, sistema ou comportamento.

Tipicamente, o sinal é gerado por um **emissor** (que codifica o sinal conforme a mensagem que pretende transmitir), e transmitido (voluntária ou involuntariamente) via algum canal de transmissão. O sinal é então capturado por um ou mais **receptores** ou **detectores**. O objetivo da análise no receptor é extrair a informação contida no sinal. O canal de transmissão é usualmente imperfeito, de tal forma que o receptor não terá acesso ao sinal tal e qual emitido. Este problema, portanto, exige a introdução de alguma noção de incerteza (para mais detalhes sobre o problema da comunicação em meios ruidosos, ver Shannon (2001)).

No contexto mais tradicional, o sinal é uma função real ou complexa com domínio no tempo ($y : \Re_+ \to \Re$, ou $y : \Re_+ \to \mathbb{C}$). Este é o caso, por exemplo, em sistemas de comunicação como o telégrafo, ou em sistemas de reprodução de áudio. Se, por outro lado, definimos o sinal como uma função não-negativa no plano cartesiano, $x : \Re^2 \to \Re_+$, o problema pode ser relacionado ao processamento de imagens.

O problema principal da tese é o de análise de sinais unidimensionais do ponto de vista do receptor: utilizar o conhecimento prévio que ele tenha sobre o sinal emitido, e sobre as características do meio de transmissão, para extrair a informação presente no sinal recebido.

Neste caso, o modelo mais usual é o modelo aditivo, escrito como $y(t) = \sum_{i=1}^{k} x_i(t) + r(t)$, onde $y(t)$ é o **sinal recebido**, $x_i : \Re_+ \to \Re$ é o **sinal emitido**, e $r(t)$ é o **ruído**. As k componentes representam a situação hipotética em que diversos sinais emitidos estão misturados no sinal recebido.

Os sinais emitidos x_i podem depender de parâmetros $\theta_x \in \Theta_X$. O ruído, ou resíduo, aparece no modelo para explicitar a imperfeição do canal de transmissão.

1.1 Motivação: análise de sinais acústicos submarinos

Um caso particular de sinal unidimensional é o sinal acústico; um sinal acústico é o resultado da propagação de ondas mecânicas (o som), no ar ou na água, e é capturado tipicamente por meio de um microfone (para sinais acústicos que se propagam no ar) ou hidrofone (sinais que se propagam na água).

O Laboratório de Meio Ambiente e Acústica (LACMAM) da Escola Politécnica da Universidade de São Paulo (EP-USP) desenvolveu no ínicio da década de 2010 o *OceanPod*: um hidrofone capaz de gravar sinais acústicos em ambientes aquáticos, a profunidades de até 350 metros. O OceanPod possui autonomia de 5 meses para gravação contínua com baterias alcalinas, e 12 meses com pilhas de lítio, e frequência de digitalização entre 4 e 48 kHz. Maiores informações sobre o OceanPod podem ser encontradas na página http://www.lacmam.poli.usp.br/Submarina.html, e igualmente em Sanchez-Gendriz e Padovese (2016).

Em 2015, um OceanPod foi instalado no *Parque Estadual Marinho da Laje de Santos*, em Santos, SP (http://www.lajeviva.org.br/parque/). O Parque da Laje é uma unidade de conservação

marinha, e foi o primeiro parque marinho dentre as unidades de conservação do Estado de São Paulo.

Os objetivos principais do projeto eram utilizar o sinal acústico obtido pelo OceanPod para auxiliar na detecção de embarcações (a pesca na região do Parque é proibida; entretanto, muitos barcos desobedecem a proibição, especialmente à noite), e também para estudar o comportamento da fauna marinha da região. O OceanPod foi instalado a uma profundidade de 20 m; a cada três meses aproximadamente ele era retirado para extração dos dados, e depois reinstalado na mesma localização.

Na nomenclatura da acústica submarina, o sinal obtido pelo OceanPod no Parque da Laje é considerado um sinal de *águas rasas*. Do ponto de vista da modelagem matemática, é um dado desafiador: inúmeras fontes de ruído concorrem simultaneamente no sinal (ruído da água, chuva, sons de pequenos animais como camarões e cracas); essas fontes apresentam caracerísticas distintas, conforme a temperatura da água, condições climáticas e de maré, etc.

Por fim, trata-se de um volume muito grande de dados; a uma taxa de amostragem de $11.025Hz$ (esta é a taxa dos dados obtidos no Parque da Laje), e considerando um tempo total de gravação de aproximadamente 3 meses (isto é, considerando apenas o sinal obtido entre duas extrações), temos um sinal da ordem de 85×10^9 pontos.

Para lidar com esses dados de forma eficiente, e ainda considerando que os problemas de interesse tem forma bastante geral (detecção de embarcações, extração de eventos de fauna marinha), adotamos como referencial teórico os métodos da *análise Bayesiana de sinais*. Este arcabouço teórico permite que toda informação prévia disponível sobre o sinal possa ser incorporada na análise; além disso, muitos métodos computacionalmente eficientes estão disponíveis para análise.

Na próxima seção, apresentamos uma breve revisão da literatura da área.

1.2 Análise Bayesiana de sinais

A teoria da análise Bayesiana de sinais, conforme a aplicaremos neste trabalho, foi desenvolvida principalmente por Edwyn Jaynes e seu aluno Larry Bretthorst, nas décadas de 80 e 90 do último século (Gregory (2001); Ruanaidh e Fitzgerald (1996)).

Jaynes, cuja formação original era em física, se interessou pelos métodos Bayesianos e pela teoria de probabilidades a partir de uma análise dos métodos da mecânica estatística Jaynes (1982). Seu livro-texto em teoria da probabilidade Jaynes (2003) é uma referência importante na área.

No contexto do processamento de sinais, os primeiros trabalhos de Jaynes dizem respeito à estimação do espectro na presença de ruído Jaynes (1987). Após esse trabalho seminal, foi seu aluno Larry Bretthorst quem se dedicou a expandir os métodos probabilísticos de Jaynes na análise de diversos outros problemas da área Bretthorst (1990a,b,c, 1991, 1992). Os métodos de Jaynes e Bretthorst, que serão aplicados neste trabalho, estão compilados em Bretthorst (1988).

1.2.1 Modelo

O modelo fundamental da análise de sinais unidimensionais é conforme segue: seja o sinal recebido representado pela função $y : \Re_+ \to \Re$. O sinal emitido é definido por $x : \Re_+ \to \Re$, e adotamos o modelo aditivo $y(t) = \xi x(t) + r(t)$. A constante $\xi \geq 0$ pode ser vista como um fator de ganho, que representa a intensidade (ou amplificação) do sinal emitido x no sinal recebido y.

A função x depende de parâmetros $\theta_x \in \Theta_x$. A componente residual $r(t)$ será chamada *ruído*, e pode ser interpretada como a soma dos demais componentes do sinal que não podem ser adequadamente modelados pela função x. A presença da componente residual significa que em nenhum caso temos esperança de encontrar uma forma funcional x tal que $x = y$.

Suponha então que observamos um conjunto finito de valores tomados de y; ou seja, obtemos o *sinal discretamente amostrado* $y(t_i)$, $i = 1, ..., N$, $t_i \in [0, T]$. Neste contexto, os principais problemas que podem surgir são:

1. **Estimação de parâmetros**: conhecendo-se a forma do sinal emitido, e estando certos de sua presença no sinal recebido, usar este último para estimar os parâmetros θ_x que definem

completamente o sinal emitido; este problema corresponde ao caso em que sabemos quando o sinal foi emitido, sabemos qual a sua forma geral, mas não conhecemos exatamente suas particularidades;

2. **Seleção de modelos**: supondo duas ou mais formas alternativas para o sinal emitido, decidir qual destas formas está presente no sinal recebido; neste caso, sabemos que um sinal, de vários possíveis, foi emitido, mas não sabemos exatamente qual deles;

3. **Detecção binária**: conhecendo-se a forma do sinal emitido, detectar a presença deste no sinal recebido; neste caso, sabemos qual sinal foi emitido, mas não sabemos quando;

O problema da estimação de parâmetros é analisado na subseção 1.2.2 deste capítulo. Este será o problema escolhido para apresentarmos uma revisão introdutória da teoria Bayesiana de análise de sinais, que será depois utilizada nos demais capítulos da tese.

O problema de seleção de modelos em processamento de sinais será analisado na seção 1.2.3. Ele diz respeito ao caso em que a forma funcional do sinal emitido não está completamente determinada, podendo ser qualquer uma dentre um conjunto discreto de possibilidades. Este problema tem interesse, por exemplo, em situações de decodificação, onde diferentes sinais emitidos correspondem a diferentes símbolos que pode ser transmitidos, e o objetivo do receptor é determinar a cada instante qual deles foi de fato transmitido.

Finalmente, o terceiro problema diz respeito ao caso de *detecção de sinais*: temos um sinal emitido determinístico e conhecido (parcial ou totalmente), que pode ou não estar embutido no sinal recebido conforme o valor de t. Queremos determinar apenas o padrão temporal do sinal emitido no sinal recebido (por exemplo, no caso mais simples, queremos determinar em um dado intervalo de tempo se o sinal foi ou não emitido). Tratamos deste problema no capítulo 2 da tese, onde propomos uma solução original aplicada à detecção de embarcações em sinais acústicos.

1.2.2 Estimação de sinais

Neste caso, admitimos uma forma conhecida para o sinal emitido, da forma

$$x(t) = \sum_{h=1}^{m} A_h G_h(t_i, \theta)$$

Ou seja, o sinal é tomado como uma combinação linear de funções G_h, $h = 1, ..., m$. Os coeficientes A_h representam as amplitudes de cada componente do sinal; as funções G_h podem depender de um conjunto θ de parâmetros, e esta dependência pode ser não-linear.

Em muitas aplicações, é comum adotarmos modelos sinusoidais para o sinal emitido. Uma vez que as funções trigonométricas descrevem quantidades periódicas, e dado que muitos dos problemas em processamento de sinais estão relacionados à análise de ondas (mecânicas, no caso do sinal acústico), estes modelos são escolhas naturais.

O modelo sinusoidal geral é dado por

$$G_h(t_i, \omega, \phi_h) = cos(2\pi h\omega t + \phi_h)$$

Neste caso, o sinal é uma função tonal, com frequência fundamental ω e m harmônicos; os parâmetros ϕ_h modelam a fase de cada harmônico.

Obtendo uma amostra $\{y_i\}_i$ com $y_i = y(t_i)$, com $t_i \in [0, T]$, o objetivo é estimar os parâmetros da função $x(t)$, ou seja: encontrar m e $\theta = (\omega, \phi_1, \phi_2, ...\phi_m)$ que permitem o melhor ajuste aos dados[1], assumindo-se que a forma $x(t)$ é correta.

Trata-se de uma situação de cálculo na presença de incerteza; vamos, portanto, abordar o problema a partir da teoria da probabilidade, da seguinte maneira: tomamos o modelo usual $y_i = x(t_i) + r_i$, onde o componente r_i é incluído para explicitar o fato de que não há solução

[1] O melhor ajuste aos dados é tomado no sentido de norma mínima para a componente residual $r(t)$.

exata para o problema. Na prática, esta componente vai abrigar toda a variação contida nos dados que não pode ser atribuída ao sinal emitido x.

Assumiremos daqui em diante apenas que $\langle r_i \rangle = E(r_i) = 0$, e que a potência média[2] desta componente de ruído é um valor $\sigma_r^2 < \infty$. Por considerações de máxima entropia Jaynes (1982, 1987), o modelo indicado para este caso é o Gaussiano

$$P(r|I) = (2\pi\sigma_r^2)^{-N/2} \times exp\left[-\frac{\sum_{i=1}^{N} r_i^2}{2\sigma_r^2}\right]$$

Aqui, seguindo a notação de Jaynes (2003), incluímos o termo I no condicionante da expressão $P(r|I)$ para deixar explícito que esse modelo probabilístico é baseado em informações a priori que possuímos sobre o problema em questão.

Indo adiante, a equação $y_i = x(t_i) + r_i$ nos permite escrever o modelo acima como

$$P(y|I, \sigma_r^2, \mathbf{A}, \theta) = (2\pi\sigma_r^2)^{-N/2} \times exp\left[-\frac{\sum_{i=1}^{N}(y_i - x(t_i))^2}{2\sigma_r^2}\right] \tag{1.1}$$

$$= (2\pi\sigma_r^2)^{-N/2} \times exp\left[-\frac{\sum_{i=1}^{N}\left(y_i - \sum_{h=1}^{m} A_h G_h(t_i, \theta)\right)^2}{2\sigma_r^2}\right] \tag{1.2}$$

As equações acima definem a verossimilhança do modelo, representada pela distribuição condicional do vetor de dados, dados os parâmetros lineares e não-lineares do modelo.

Para obtermos estimativas dos parâmetros, segundo o método Bayesiano, é necessário postularmos distribuições *a priori* para os parâmetros. Antes disso, porém, podemos fazer uma distinção entre os parâmetros lineares e não-lineares, conforme o problema de interesse.

Estimando os parâmetros não-lineares

Suponha que, num determinado problema, estamos principalmente interessados nos parâmetros não-lineares θ do sinal. Isto pode acontecer, por exemplo, num modelo sinusoidal onde o interesse principal é pela frequência fundamental do sinal.

Nestes casos, os parâmetros lineares A_h podem ser considerados *parâmetros incômodos* (*nuisance parameters*), no sentido de que sua presença é apenas um fator complicador do problema, sem que seus valores guardem nenhum interesse especial.

Em situações como essa, os métodos Bayesianos tradicionais sugerem que tais parâmetros sejam *marginalizados*; isto é feito postulando-se uma distribuição *a priori* para esses parâmetros, e integrando-se o produto desta pela verossimilhança, no domínio dos parâmetros incômodos.

Considerando então os parâmetros lineares A_h como incômodos, queremos propor uma distribuição *a priori* $\pi^A(\mathbf{A})$ e em seguida calcular a integral

$$P(y|I, \sigma_r^2, \Theta) = \int_{\Theta_A} \pi^A(\mathbf{A}) P(y|I, \sigma_r^2, \mathbf{A}, \theta_x) d\mathbf{A} \tag{1.3}$$

Novamente, cabe notar que neste momento os parâmetros de amplitude A_h são assumidos constantes em $[0, T]$. Se, além desse fato, não possuímos nenhuma outra informação relevante sobre esses valores, adotamos uma distribuição *a priori* uniforme (e portanto imprópria[3]).

Neste caso, a integral acima pode ser calculada analiticamente para valores fixos de θ_x, da seguinte forma: desenvolvemos o fator quadrático na equação 1.1 como

[2] A potência é a integral de $|r(t)|^2$ no intervalo analisado; em estatística, chamaríamos esta quantidade a *variância* do ruído.

[3] Uma distribuição imprópria é uma medida que não pode ser normalizada no espaço todo; neste caso específico, estamos admitindo qualquer valor na reta dos reais como razoáveis para A_h.

$$\sum_{i=1}^{N}\left(y_i-\sum_{h=1}^{m}A_hG_h\right)^2=\sum_{i=1}^{N}\left(y_i^2-2\sum_{h=1}^{m}y_iA_hG_h+\left(\sum_{h=1}^{m}A_hG_h\right)^2\right)$$
$$=\sum_{i=1}^{N}y_i^2-2\sum_{i=1}^{N}\sum_{h=1}^{m}y_iA_hG_h(t_i,\Theta)+\sum_{i=1}^{N}\left(\sum_{h=1}^{m}A_hG_h(t_i,\Theta)\right)^2$$
$$=\sum_{i=1}^{N}y_i^2-2\sum_{i=1}^{N}\sum_{h=1}^{m}y_iA_hG_h(t_i,\Theta)+\sum_{i=1}^{N}\sum_{h=1}^{m}\sum_{l=1}^{m}A_hA_lG_h(t_i,\Theta)G_l(t_i,\Theta)$$

O objetivo é escrever o expoente da verossimilhança como o quadrado de alguma função linear de $\mathbf{A}$, de modo que a integral na equação 1.3 se torne uma integral Gaussiana. Isto acontece se as funções G_h são tais que $\sum_{i=1}^{N}G_h(t_i,\Theta)G_l(t_i,\Theta)=\delta_{hl}$. Em outras palavras, o que queremos é diagonalizar a matriz $\{g_{jk}\}$ dada por $g_{jk}=\sum_{i=1}^{N}G_jG_k$, onde a dependência em Θ e t_i foi omitida para desonerar a notação.

Este procedimento é descrito em detalhes em Bretthorst (1990a, 1988); em termos simples, se obtemos os autovetores e_i e autovalores λ_i da matriz g_{jk}[4], podemos definir novas funções H_j e parâmetros A_j como

$$H_j(t)=\frac{1}{\sqrt{\lambda_j}}\sum_{k=1}^{m}e_{jk}G_k(t)$$
$$B_j=\sqrt{\lambda_j}\sum_{k=1}^{m}A_ke_{jk}$$

de tal forma que $\sum_{i=1}^{N}H_j(t_i)H_k(t_i)=\delta_{jk}$. Agora, se escrevemos o novo modelo como $x(t)=\sum_{j=1}^{m}B_jH_j(t_i,\Theta)$, temos

$$=\sum_{i=1}^{N}y_i^2-2\sum_{i=1}^{N}\sum_{j=1}^{m}y_iB_jH_j(t_i,\Theta)+\sum_{i=1}^{N}\sum_{j=1}^{m}\sum_{k=1}^{m}B_jB_kH_j(t_i,\Theta)H_k(t_i,\Theta)$$
$$=\sum_{i=1}^{N}y_i^2-2\sum_{i=1}^{N}\sum_{j=1}^{m}y_iB_jH_j(t_i,\Theta)+\sum_{j=1}^{m}B_j^2$$
$$=\sum_{j=1}^{m}\left(B_j^2-\sum_{i=1}^{N}y_iH_j(t_i,\Theta)\right)^2+\sum_{i=1}^{N}y_i^2-\sum_{j=1}^{m}\left(\sum_{i=1}^{N}y_iH_j(t_i,\Theta)\right)^2$$

Desta forma, a nova posteriori em termos de H_j e B_j se torna

$$P(y|I,\sigma_r^2,\mathbf{B},\Theta)=(2\pi\sigma_r^2)^{-N/2}\times exp\left[-\frac{\sum_{j=1}^{m}\left(B_j^2-\sum_{i=1}^{N}y_iH_j(t_i,\Theta)\right)^2}{2\sigma_r^2}\right]$$
$$\times exp\left[-\frac{N\overline{y^2}-m\overline{h^2}}{2\sigma_r^2}\right]$$

onde $\overline{y^2}=\frac{1}{N}\sum_{i=1}^{N}y_i^2$ e $\overline{h^2}=\frac{1}{m}\sum_{j=1}^{m}\left(\sum_{i=1}^{N}y_iH_j(t_i,\Theta)\right)^2$.

[4]Se esta matriz tiver posto m, podemos provar que ela será positiva-definida.

Esta expressão pode ser integrada com respeito a cada B_j sucessivamente, resultando na seguinte expressão:

$$P(y|I,\sigma_r^2,\Theta) \propto \sigma_r^{-N+m} \times exp\left[-\frac{N\overline{y^2} - m\overline{h^2}}{2\sigma_r^2}\right] \tag{1.4}$$

Algumas considerações devem ser feitas neste ponto. Em primeiro lugar, a expressão acima foi obtida pela diagonalização da matriz de forma do modelo; esta diagonalização, porém, depende dos valores particulares de θ_x; logo, em situações de cálculo numérico, a obtenção dos autovetores e autovalores da matriz $\{g_{jk}\}$ deve ser feita para cada novo valor de θ_x. Em especial, se algum método numérico de otimização ou integração for construído tendo a função 1.4 como função-objetivo, haverá um custo adicional importante a ser levado em consideração.

Esta situação pode ser evitada de várias formas diferentes: a primeira, se escolhermos funções base G_j tal que $\forall\theta \in \Theta \sum_{i=1}^{N} G_j(t_i,\theta)G_k(t_i,\theta) = \delta_{jk}$. Esta restrição, porém, limita sobremaneira a escolha de formas funcionais para o modelo.

Outra maneira de contornar o problema é utilizar aproximações diagonais para a matriz G, ou utilizar funções G_j que permitam expressões analíticas para os autovetores e autovalores de G; exemplos para dados de quadratura (sinais obtidos numa máquina de ressonância magnética) podem ser encontrados em Bretthorst (1990b).

Por fim, podemos abandonar a diagonalização completamente, e trabalhar no espaço paramétrico completo (ou seja, trabalhar diretamente com a função 1.1). Neste caso, nos livramos da necessidade de diagonalizar a matriz, mas em contrapartida teremos um espaço paramétrico com dimensão maior.

Como primeira ilustração, vamos aplicar a metodologia delineada acima na estimação da frequência de uma única senóide estacionária (i.e., um sinal da forma $x(t) = cos(\alpha t)$ definido para todo $t > 0$).

Temos então o modelo

$$y(t) = A_0 cos(2\pi\omega t + \phi) + r(t)$$

Neste caso, a marginalização de A_0 pode ser feita se $\sum_{i=1}^{N} cos^2(2\pi\omega t_i + \phi) = 1$. Para isto basta multiplicar o modelo de $y(t)$ por uma função $S(\omega,\phi)$ de tal forma que a igualdade esteja sempre satisfeita, isto é

$$S(\omega,\phi) = \left(\sum_{i=1}^{N} cos^2(2\pi\omega t_i + \phi)\right)^{-1}$$

Obtemos então

$$P(y|I,\omega,\phi,\sigma_r^2) \propto \sigma_r^{-N+1} exp\left[-\frac{N\overline{y^2} - h^2}{2\sigma_r^2}\right] \tag{1.5}$$

com $h^2 = \left(\sum_{i=1}^{N} S(\omega,\phi) y_i cos(2\pi\omega t_i + \phi)\right)^2$ e $\overline{y^2}$ conforme definição acima.

Aqui, se a variância do ruído é conhecida, podemos usar a equação acima para estimar os parâmetros ω e ϕ. Caso contrário, propomos uma distribuição a priori para σ_r e integramos novamente para marginalizar esse parâmetro.

Usando a priori de Jeffreys (Jeffreys (1939)), $\pi_\sigma(s) = 1/s$, chegamos à seguinte integral:

$$\int_0^\infty \sigma_r^{-N} exp\left[-\frac{N\overline{y^2} - h^2}{2\sigma_r^2}\right] d\sigma$$

Usando a regra da integração por partes iteradas vezes, chegamos a

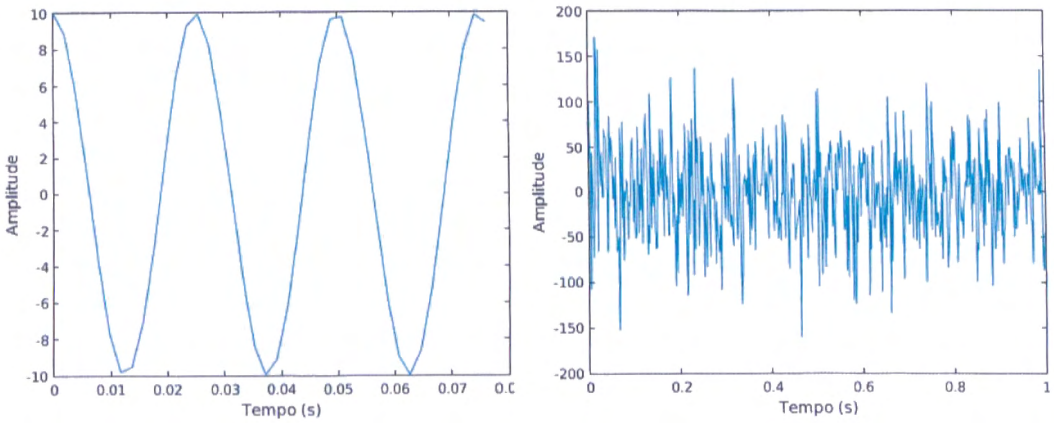


Figura 1.1: Sinal simulado, com e sem adição do ruído Gaussiano

$$P(y|I,\omega,\phi) \propto \left(N\overline{y^2} - h^2\right)^{\frac{1-N}{2}} \propto \left(1 - \frac{h^2}{N\overline{y^2}}\right)^{\frac{1-N}{2}} \tag{1.6}$$

A desigualdade de Bessel garante que $1 - h^2/N\overline{y^2} > 0$ (Bretthorst (1988)).

Se o interesse é apenas na frequência do sinal, ainda restaria a fase ϕ a ser marginalizada. A integral em relação a ϕ, porém, não pode ser realizada analiticamente, de modo que seguiremos a análise com as equações 1.5 e 1.6.

O próximo passo é obter a distribuição a posteriori do modelo. Para isso, definimos prioris π_ω e π_ϕ, e multiplicamos essas prioris por 1.5 ou 1.6 (se conhecemos ou não a variância, respectivamente), para obter a posteriori

$$P(\omega,\phi|I,y) \propto P(y|I,\omega,\phi) \times \pi_\omega \times \pi_\phi \tag{1.7}$$

Para ilustrar o método, vamos aplicá-lo à estimação da frequência de um sinal simulado. Simulamos um sinal continuo $x(t) = 10cos(2\pi \cdot 40t)$, amostrado durante $1s$ a $512\ Hz$. Adicionamos ao sinal um ruído Gaussiano com $\sigma_r^2 = 2500$ (para obter um SNR de aproximadamente 1). Nas figuras 1.1 vemos o gráfico do sinal puro, e do sinal após adição do ruído.

Na figura 1.2 temos o espectro do sinal, estimado via transformação de Fourier discreta. Podemos notar que a presença do ruído não permite uma conclusão precisa sobre a frequência do sinal: muito embora o máximo do espectro esteja na frequência correta, há outros máximos locais e um em particular com valor bastante próximo, em torno da frequência $85\ Hz$. Nas figuras 1.3 e 1.4 temos o valor da log-posteriori (utilizando a verossimilhança da equação 1.5, e assumindo prioris não-informativas) assumindo $\phi = 0$ conhecido, e para diferentes valores de ϕ.

O método Bayesiano mostra picos pronunciados no valor verdadeiro da frequência; na posteriori em si (ou seja, após exponenciação) os picos seriam ainda mais pronunciados, mostrando que o método Bayesiano permite a estimação precisa do valor verdadeiro da frequência do sinal.

Estimando os parâmetros lineares

Suponha agora que o interesse principal é a estimação do parâmetro linear do modelo (ou seja, a estimação da amplitude A_0). O processo, nesse caso, é similar ao da seção anterior: escrevemos o modelo probabilístico para a posteriori de A_0, dados os demais parâmetros e condicional à informação disponível *a priori*; procedemos à marginalização dos parâmetros incômodos (no caso do modelo sinusoidal, a frequência, a fase e a variância do ruído), e trabalhamos com a posteriori marginalizada.

Suponha por exemplo o modelo sinusoidal da forma

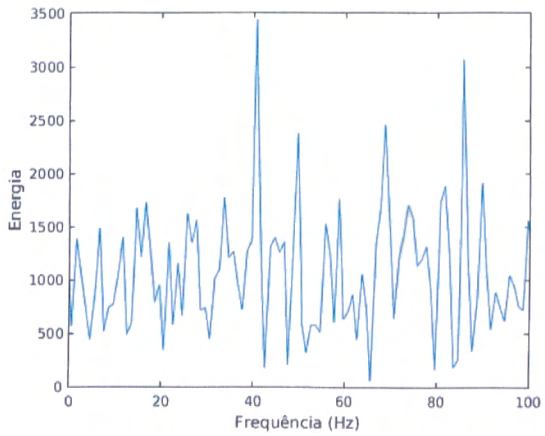


Figura 1.2: Espectro estimado via DFT

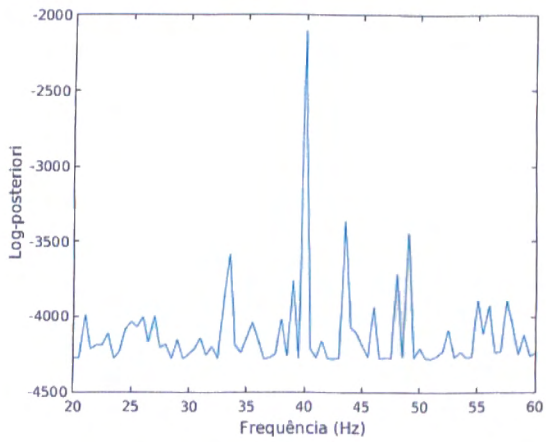


Figura 1.3: Log-posteriori para $\phi = 0$

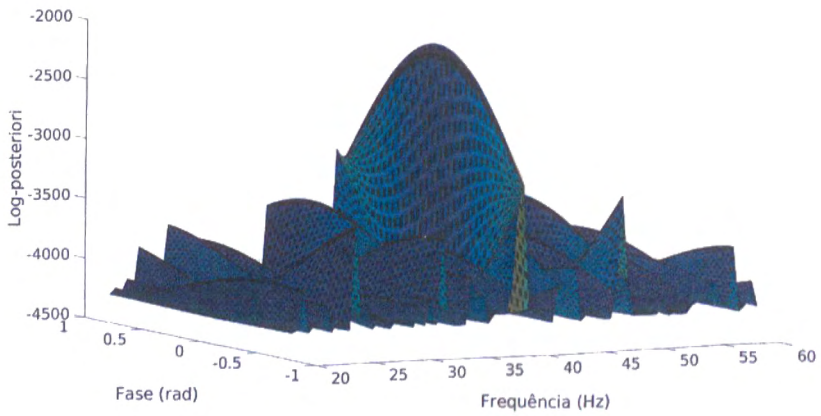


Figura 1.4: Log-posteriori num reticulado de valores para ϕ e ω

$$y(t) = Acos\,(\omega t + \phi) + r(t) \tag{1.8}$$

onde admitimos que a potência média do ruído é desconhecida mas finita, $\sigma_r^2 < \infty$.

A amplitude, por sua vez, também deve estar limitada (caso contrário, o sinal teria potência infinita); ou seja, temos $0 < A < A_{max}\infty$, pois a amplitude, representando o ganho do sinal, deve ser um parâmetro não-negativo.

Novamente por considerações de máxima entropia, adotamos o modelo Gaussiano para o ruído, o que nos leva ao seguinte modelo probabilístico para o sinal recebido (já considerado como discretamente amostrado no intervalo $[0, T]$):

$$P(y|A,\omega,\phi,\sigma_r^2,I) = \left(2\pi\sigma_r^2\right)^{-N/2} exp\left[-\frac{\sum_{i=1}^{N}\left(y_i - Acos(\omega t_i + \phi)\right)^2}{2\sigma_r^2}\right] \tag{1.9}$$

Como sabemos, o objetivo é obter $P(A|y,\omega,\phi,\sigma_r^2,I)$; para isto, aplicamos o teorema de Bayes

$$P(A|y,\omega,\phi,\sigma_r^2,I) \propto P(y|A,\omega,\phi,\sigma_r^2,I)P(A|\omega,\phi,\sigma_r^2,I) \tag{1.10}$$

$$= P(y|A,\omega,\phi,\sigma_r^2,I)P(A|I) \tag{1.11}$$

onde incluímos a hipótese de que o valor da amplitude não depende dos demais parâmetros do modelo.

A amplitude, como vimos, está limitada superiormente por A_{max}; além disso, não supomos nenhuma outra informação disponível sobre seu valor. Nestas condições, a distribuição *a priori* mais adequada é a uniforme,

$$P(A|I) = \frac{1}{A_{max}}\mathbf{1}_{(0<A<A_{max})} \tag{1.12}$$

onde $\mathbf{1}_X$ é a função característica do conjunto X.

Portanto, a posteriori será proporcional à verossimilhança do modelo, dada por

$$P(A|y,\omega,\phi,\sigma_r^2,I) \propto (2\pi\sigma_r^2)^{-N/2} exp\left[-\frac{\sum_{i=1}^{N}\left(y_i - Acos(\omega t_i + \phi)\right)^2}{2\sigma_r^2}\right] \tag{1.13}$$

Para marginalizar os parâmetros incômodos ω, ϕ e σ_r^2, calculamos a integral

$$P(A|y,I) \propto \int_\Omega \int_\Phi \int_\Sigma P(A|y,\omega,\phi,\sigma_r^2,I)d\omega d\phi d\sigma \tag{1.14}$$

Começamos por marginalizar a fase ϕ. Para isto, substituímos em (1.13) o termo $cos(\omega t_i + \phi)$ pela expressão para o cosseno de uma soma, e expandimos o quadrado. Desta forma, é possível obter uma expressão fechada para a integral utilizando funções de Bessel do primeiro tipo. Detalhes podem ser encontrados em Jaynes (1987). Neste mesmo trabalho, o parâmetro σ_r^2 também é marginalizado analiticamente. A integral com respeito à frequência ω, contudo, não pode ser obtida analiticamente. Sendo assim, é necessário adotar alguma aproximação, ou utilizar algum método de quadratura numérica. Uma análise detalhada deste problema pode ser encontrada em Bretthorst (1992, 1988); Jaynes (1987).

1.2.3 Seleção de modelos

Outro problema de interesse no processamento de sinais é a seleção de modelos. A motivação para este problema surge quando há incerteza sobre a forma mais adequada para o sinal emitido, seja por falta de informação suficiente sobre o processo causador do sinal, seja por que sabemos que diversos tipos de sinais podem ter sido emitidos (por exemplo no problema da telecomunicação, onde diferentes símbolos podem ser codificados com formas funcionais distintas). Uma vez que a mensagem é obtida pelo receptor, seu objetivo passa então a ser determinar qual, dentre um conjunto finito de formas possíveis, é a mais adequada para descrever o sinal presente nos dados.

Este problema tem relação com o problema da detecção de sinais, que analisaremos no próximo capítulo. No problema de detecção, temos dois modelos alternativos: o primeiro representado pela função constante $x_0(t) = K_0$ (ausência de sinal emitido), e o segundo o modelo com sinal emitido $y_i = K_0 + \xi x(t) + r(t)$.

No caso mais geral, se possuímos m modelos alternativos $x_i, i = 1, ..., m$ para o sinal emitido, podemos escrever

$$y(t) = \xi_1 x_1(t) + \xi_2 x_2(t) + ... + \xi_m x_m(t) + r(t) \tag{1.15}$$

Se impomos as condições de complementaridade $\xi_i \geq 0$, $\sum_i \xi_i > 0$ e $\xi_i \xi_j = 0 \quad \forall i \neq j$, estamos na prática afirmando que apenas um dos m sinais emitidos está de fato presente no sinal recebido, e o problema é de fato o de seleção de modelos.

Se, no entanto, abandonamos essas condições e impomos apenas que $\xi_i \geq 0$, estamos aceitando a presença simultânea de mais de um sinal emitido. Este tipo de problema está relacionado à análise de **modelos de mistura** em estatística, e ao problema de contagem e separação de fontes.

Para a seleção de modelos (ou seja, assumindo as condições de complementaridade), a teoria de análise de sinais baseada na inferência Bayesiana procede da seguinte maneira: sabendo que um e apenas um dos m modelos de sinal está presente no sinal recebido, calculamos a probabilidade *a posteriori* associada a cada modelo x_j, $P(x_j|y, I)$. Pelo teorema de Bayes, essas probabilidades são dadas por

$$P(x_j|y, I) = \frac{P(y|x_j, I)P(x_j|I)}{P(y|I)}, \qquad j = 1, ..., m \tag{1.16}$$

Na equação acima, $P(x_j|I)$ modela a informação que temos sobre os diversos modelos, **antes** de obtermos o sinal; $P(y|x_j, I)$ é a verossimilhança, que mede a qualidade do ajuste do modelo aos dados, e $P(y|I)$ representa a informação *a priori* que temos sobre o sinal recebido, antes de obtê-lo de fato; esta componente é a constante de normalização sobre todos os possíveis modelos, para garantir que $\sum_{j=1}^{m} P(x_j|y, I) = 1$.

Observe que desta forma o conjunto de modelos possíveis é exaustivo: ou seja, assume-se que no máximo um, e pelo menos um, dos modelos x_j está presente no sinal recebido. Para incluir a possibilidade de que um modelo não-especificado esteja presente, bastaria tomar as probabilidades *a priori* $P(x_j|I)$ com soma estritamente menor do que 1. Isso equivaleria a adotarmos uma última possibilidade não-especificada para o sinal, e impediria a aplicação dos métodos Bayesianos aqui propostos, pois o valor de $P(y|x_?, I)$ não está definido se $x_?$ é um modelo não-especificado (Bretthorst (1990b)).

No caso em que assumimos $\sum_j P(x_j|I) = 1$, a análise começa pela definição dessa partição da unidade (as probabilidades associadas *a priori* para cada possível modelo). Se não há razões para considerar qualquer um dos modelos como mais provável, tomamos $P(x_j|I) = 1/m$; ou seja, adotamos uma *priori* uniforme nos modelos.

Neste caso, a expressão da *posteriori* se torna

$$P(x_j|y, I) = \frac{P(y|x_j, I)}{\sum_{j=1}^{m} P(y|x_j, I)} \tag{1.17}$$

e o problema de seleção de modelos resume-se à obtenção dos valores $P(y|x_j, I)$.

Lembrando, porém, que cada modelo x_j depende de parâmetros θ_j, e que além disso as verossimilhanças dependem da variância σ_r^2 do resíduo (onde novamente assumimos um modelo Gaussiano para este resíduo), temos

$$P(y|x_j, I) = \int_{\Theta_j} \int_0^{\infty} P(y, \Theta_j, \sigma_r^2|x_j, I) d\sigma_r d\theta_j \tag{1.18}$$

onde os parâmetros θ_j incluem tanto os parâmetros não-lineares quanto os parâmetros lineares do modelo.

Na análise que fizemos nas seções anteriores para o modelo tonal (superposição de senóides com frequências dadas por múltiplos inteiros da frequência fundamental), os parâmetros lineares podem ser marginalizados analiticamente, à custa de uma ortogonalização do modelo. Isto foi feito a partir da adoção de *prioris* impróprias (isto é, não normalizadas) sobre os parâmetros lineares.

No contexto da seleção de modelos, porém, isto não é mais possível (Bretthorst (1990b)), pois neste caso, a menos que todos os modelos x_j possuam o mesmo número de parâmetros lineares, o modelo com mais parâmetros seria automaticamente excluído (para ver isto, suponha *prioris* uniformes em $[-a, a]$ para todos os parâmetros lineares, e tome o limite $a \to \infty$ na equação 1.17; o modelo com mais parâmetros lineares terá uma constante tendendo a infinito no denominador, e portanto terá *posteriori* nula).

Uma solução seria impor *prioris* Gaussianas para os parâmetros lineares; uma análise detalhada deste procedimento na seleção de modelos pode ser encontrada em Bretthorst (1990b).

1.3 Conclusão

Neste capítulo, introduzimos brevemente os métodos da análise Bayesiana de sinais. O principal objetivo era ilustrar a teoria e introduzir seus principais conceitos, que serão utilizados ao longo da tese.

No próximo capítulo estudamos o problema da detecção binária (presença *versus* ausência) de um sinal de forma conhecida. A análise deste problema vai revelar algumas dificuldades relacionadas à análise do sinal acústico obtido do OceanPod; estas dificuldades por sua vez motivam a análise dos capítulos 3 e 4.

Capítulo 2

FBST para detecção binária de sinais

Outro problema de interesse na análise de sinais acústicos em particular, e na área de processamento de sinais em geral, é o problema de **detecção de sinais**.

Definimos este problema da seguinte maneira: o receptor captura uma certa mensagem (os dados obtidos do(s) sensor(es)), e está interessado em descobrir se aquela mensagem contém um sinal de forma bem determinada. Este tipo de problema pode aparecer na área de telecomunicações, por exemplo, mas é também relevante em diversas outras áreas.

O modelo mais natural para essa situação é o modelo baseado no parâmetro de ganho ξ:

$$y(t) = \xi x(t) + r(t) \tag{2.1}$$

Supondo que ξ é uma variável binária, este modelo descreve uma situação em que a mensagem detectada pode carregar o sinal de interesse ($\xi = 1$) contaminado pelo ruído r, ou então pode ser composta apenas pelo ruído ($\xi = 0$). Se permitimos que ξ tome outros valores positivos, estamos admitindo que o sinal pode chegar no repector com diferentes níveis de ganho, i.e., com potências diferentes.

De qualquer modo, uma vez tendo obtido a mensagem y, o receptor está interessado em determinar se $\xi > 0$. Em outras palavras, ele deseja testar a hipótese $H_0 : \xi = 0$, contra a hipótese alternativa $H_1 : \xi > 0$ (Daly e Rushforth (1965)).

Note que no modelo 2.1, não especificamos exatamente o modelo para $x(t)$ (isto é, a forma funcional do sinal que foi possivelmente emitido). Posto desta forma, portanto, o modelo é geral e descreve situações em que: *a)* a forma funcional do sinal é conhecida exatamente; *b)* a forma funcional é conhecida a menos do valor exato de alguns parâmetros; *c)* a forma funcional não é conhecida. Cada uma destas possibilidades descreve uma situação específica, e o tratamento dado ao problema será diferente conforme o caso.

Todas elas, porém, possuem em comum a formulação da hipótese de interesse, $H_0 : \xi = 0$. Esta hipótese, se verdadeira, implica numa redução da dimensionalidade do espaço paramétrico do modelo, pois afirmar que $\xi = 0$ equivale a dizer que o modelo do ruído é suficiente para explicar os dados da mensagem recebida.

Hipóteses desse tipo são chamadas *hipóteses precisas (sharp)*, e representam um problema para os métodos tradicionais de testes de hipóteses, tanto no caso Bayesiano quanto no caso clássico.

No caso Bayesiano, o teste de hipóteses é usualmente realizado pelo cálculo da medida a posteriori associada ao conjunto definido por H_0. No caso das hipóteses precisas, porém, temos um conjunto de medida nula (H_0 define um hiperplano no espaço paramétrico completo). Logo, a abordagem Bayesiana padrão não pode ser aplicada diretamente.

Algumas correções *ad hoc* podem ser utilizadas, a mais comum sendo a atribuição artificial de uma medida positiva para o hiperplano definido por H_0. Isto é tipicamente realizado através de uma medida a priori pontual, cujo valor no conjunto de medida formal nula é interpretado como representando a confiança que o pesquisador possui sobre a veracidade da hipótese.

No caso clássico, as hipóteses precisas são menos problemáticas, pois é em geral possível encontrar uma estatística cuja distribuição sob H_0 é conhecida (pelo menos assintoticamente); a partir

dessa distribuição, um *p-valor* pode ser calculado e adotado como medida de evidência.

Por outro lado, a adoção do *p-valor* pode ser problemática por si só, por diversos motivos. Os principais e mais discutidos na literatura (Pereira e Stern (1999); Pereira e Wechsler (1993), entre outros) são a necessidade de recorrer a resultados assintóticos (por exemplo o Teorema Central do Limite) e a sensibilidade a diferentes formulações da hipótese alternativa.

Devido a todas essas considerações, neste trabalho optamos por utilizar o *teste de significância genuinamente Bayesiano (Full Bayesian Significance Test, FBST)* de Pereira e Stern (1999). Este teste, que descrevemos brevemente na próxima seção, foi construído para permitir o cálculo de evidência sobre hipóteses precisas utilizando a teoria Bayesiana, mas sem a necessidade de recorrer a procedimentos *ad hoc* quaisquer.

2.1 FBST

Seja um modelo probabilístico paramétrico qualquer, representado pela posteriori $P(\theta|y)$, e seja H_0 uma hipótese precisa qualquer. Definimos os espaços paramétricos irrestrito Θ e sob H_0, Θ_{H_0}, e $P_{H_0}(\theta|y)$ a restrição da posteriori ao conjunto Θ_{H_0}:

$$P_{H_0}(\theta|y) = P(\theta|y) \cdot \mathbf{1}_{\{\theta \in \Theta_0\}}$$

O primeiro passo para o cálculo do valor de evidência do FBST é obter $\theta_0 = argmax P_{H_0}(\theta|y)$, o ponto de máximo da posteriori restrita.

Uma vez encontrando o ponto de máximo, definimos $p_0 = P_{H_0}(\theta|y)$ o valor máximo da posteriori restrita a H_0. Este valor quantifica, num certo sentido, a força da hipótese nula considerada relativamente ao espaço paramétrico inteiro.

Definimos em seguida o *conjunto surpresa*

$$T_0 = \{\theta \in \Theta : P(\theta|y) > p_0\}$$

Este conjunto contém os elementos do espaço paramétrico completo (i.e., sem a restrição dada por H_0) que possuem densidade posteriori maior do que o máximo da posteriori sob H_0. Ou seja, este conjunto agrupa os pontos do espaço paramétrico que, considerando os dados à mão, possuem valor da densidade maior do que o máximo que a hipótese nula pode atingir.

A medida de evidência contra a hipótese nula, $ev(H_0)$, é por fim a medida a posteriori do conjunto surpresa, isto é

$$ev(H_0) = \int_{T_0} P(\theta|y) d\theta$$

Se este valor for igual a 0, a conclusão é a de que não há (quase certamente) pontos no espaço paramétrico irrestrito que possuam densidade a posteriori mais alta que o máximo sob H_0; ou seja, H_0 tem "força" empírica suficiente para nos convencer de sua validade. Se, no outro extremo, este valor for igual a 1, a interpretação é a simétrica: o conjunto de pontos no espaço paramétrico com menos "força" empírica do que H_0 tem medida nula; logo, H_0 não possui suporte empírico nenhum para o conjunto de dados em questão.

O cálculo do FBST, portanto, envolve dois passos: o primeiro, um passo de otimização (encontrar o máximo da posteriori sob H_0); o segundo, um passo de integração (obter a integral da posteriori sobre o conjunto surpresa). A depender da aplicação, um ou ambos esses passos devem ser realizados numericamente.

O FBST vem sendo utilizado no teste de hipóteses precisas em diversos contextos; para algumas aplicações, ver por exemplo Hubert *et al.* (2009) e referências aí contidas; ou para aplicações mais recentes, ver por exemplo Chakrabarty (2017); Diniz *et al.* (2012). Nesta tese, ele será utilizado tanto neste capítulo, na construção do método de detecção binária, quanto no próximo capítulo, onde será utilizado no critério de parada do algoritmo de segmentação de sinais.

2.2 Sinal completamente conhecido

A análise do problema de detecção de sinais, conforme observamos acima, deve incorporar a informação disponível sobre o sinal que se deseja detectar. Esta informação pode dizer respeito à forma do sinal, ao valor dos parâmetros, ou a ambos.

Neste capítulo escolhemos o modelo tonal para o sinal emitido:

$$x(t) = \sum_{h=1}^{m} A_h cos\left(2\pi h\omega t + \phi_h\right) \tag{2.2}$$

Este é o modelo constituído por uma frequência fundamental ω e m harmônicos, e é frequentemente utilizado para descrever fenômenos oscilatórios, como a propagação de ondas mecânicas.

Para modelar a imperfeição do meio de transmissão do sinal, incorporamos um termo de ruído aditivo ao modelo e, assim como no capítulo anterior, adotamos por máxima entropia o modelo Gaussiano de média 0 e variância constante, finita e desconhecida σ_r^2.

Nesta primeira análise, vamos supor que os parâmetros do modelo para o sinal emitido são completamente conhecidos. Ou seja, conhecemos exatamente o número de harmônicos, a frequência fundamental, e todas as amplitudes e fases. Toda a incerteza, portanto, reside apenas sobre os valores de ξ e σ_r^2.

Esta versão do problema de detecção binária é portanto bastante simplificada, e descreve situações que aparecem raramente na prática. Ainda assim, o problema será útil para introduzirmos o método de solução que será posteriormente aplicado a casos mais realistas.

Definimos a hipótese de detecção como $H_0 : \xi = 0$. Condicional a H_0, a posteriori é proporcional à verossimilhança de um modelo Gaussiano multiplicada pela priori π_σ

$$P_{H_0}(\xi|y,\Theta) \propto \pi_\sigma \times (2\pi\sigma_r^2)^{-N/2} exp\left[-\sum_{i=1}^{N} \frac{y_i^2}{2\sigma_r^2}\right] \tag{2.3}$$

Esta é a função que precisamos otimizar no primeiro passo para o cálculo da medida de evidência do FBST. Neste caso, e supondo uma priori imprópria para σ_r, o problema de otimização tem solução analítica dada por

$$\hat{\sigma}_r^2 = \frac{1}{N}\sum_{i=1}^{N} y_i^2$$

Determinado o máximo da posteriori restrita, voltamos a atenção para o espaço paramétrico irrestrito.

Caso seguíssemos à risca o método do capítulo anterior, o próximo passo seria marginalizar os parâmetros incômodos (a rigor apenas σ_r, já que os demais parâmetros são supostamente conhecidos) até deixar restar apenas o parâmetro de interesse ξ.

Contudo, a metodologia que embasa o uso do FBST propõe evitar a marginalização de parâmetros incômodos(Pereira e Stern (1999)), por entender que, a rigor, nenhum parâmetro do modelo pode ser enquadrado nessa categoria. Se o objetivo é calcular a evidência empírica que suporta uma determinada hipótese, todos os parâmetros envolvidos no modelo probabilístico que sustenta esse cálculo são relevantes e devem ser mantidos. Além disso, a marginalização de parâmetros extras pode levar a conclusões equivocadas com respeito a validade da hipótese. No cálculo do FBST, portanto, vamos trabalhar com o espaço paramétrico completo, neste capítulo e nos posteriores.

Sendo assim, o segundo passo para o cálculo do FBST é a integração da posteriori sobre o conjunto tangente, parametrizado no espaço paramétrico irrestrito. Esta integração frequentemente não tem solução analítica, e esta é a situação no modelo de detecção binária, mesmo no caso simples em que todos os parâmetros do sinal emitido são conhecidos. Será preciso portanto lançar mão de alguma estratégia de integração numérica.

Os métodos mais utilizados para a integração numérica no contexto da inferência Bayesiana são os de *Markov Chain Monte Carlo* (MCMC) (para um texto abrangente no assunto, ver

Gamerman e Lopes (2006)). Estes métodos se baseiam na geração de valores aleatórios cuja distribuição seja a distribuição a posteriori que se pretende integrar. Isto é feito a partir da construção de uma cadeia de Markov ergódica cuja distribuição estacionária coincida com a distribuição a posteriori; a partir daí, são obtidos valores simulados desta cadeia, formando uma amostra que seja suficientemente longa para permitir que se assuma que o estado estacionário tenha sido atingido. A principal vantagem deste método para a inferência Bayesiana é permitir que sejam simulados valores de uma distribuição da qual é conhecida apenas a forma, e não a constante de normalização. Como o teorema de Bayes tem no numerador um termo de simples obtenção (o produto da priori pela verossimilhança), e no denominador uma constante de normalização que raras vezes pode ser obtida analiticamente, os métodos de MCMC são uma ferramenta poderosa para cálculos Bayesianos.

A técnica que utilizamos para o modelo simples desta seção é um algoritmo Metropolis-Hastings tradicional, com uma distribuição candidata uniforme em $[0, 1]$ para ξ, e uma gama inversa com parâmetros 2.0001 e 5×10^4 para o parâmetro de dispersão σ_r. Este algoritmo produz uma amostra simulada cuja distribuição é (aproximadamente) aquela da posteriori no espaço paramétrico irrestrito; tendo obtido esta amostra, a estimação da integral sobre o conjunto surpresa é imediata.

Uma vez definidos todos os passos necessários para o cálculo do FBST, a metodologia de detecção binária está completa. Para testá-la, utilizaremos dados simulados: simulamos um sinal de 7 segundos de duração, composto por ruído Gaussiano branco nos primeiros e nos últimos 2 segundos (i.e., $\xi = 0$ nesses intervalos), e composto por ruído mais o sinal dado pela equação 2.2, com $m = 5, \omega = 60Hz, A_h = \{0.005, 0.004, 0.003, 0.002, 0.001\}$ e $\phi_h = \{-\pi, -\pi/2, 0, \pi/2, \pi\}$. A potência do ruído foi escolhida para resultar em um SNR (razão sinal-ruído, *signal-to-noise ratio*) igual a 2.

Nesta mensagem simulada, portanto, o sinal está presente apenas nos 3 segundos centrais; para testar o detector, dividimos o sinal simulado em 7 janelas disjuntas de 1 segundo de duração. Os resultados (valores de evidência **contra** a hipótese $H_0 : \xi = 0$) aparecem na figura 2.1.

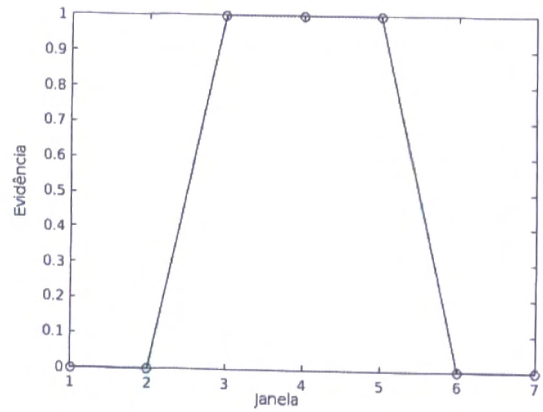


Figura 2.1: Evidência para a presença do sinal em dados simulados

Vemos portanto que nessa situação simplificada o FBST tem desempenho ótimo, atribuindo evidência 1 para a presença do sinal tonal entre $t = 3$ e $t = 5$, e 0 no restante do intervalo.

Tendo formulado o problema de detecção de sinais como um problema de teste de hipóteses precisas, era de se esperar que o método baseado no FBST apresentasse bons resultados, uma vez que o FBST é uma metodologia construída especificamente para o teste de hipóteses dessa natureza. Como vimos acima, porém, este primeiro problema é uma simplificação extrema dos casos mais realistas; na próxima seção, relaxamos um pouco as hipóteses simplificadoras e observamos novamente o desempenho do método.

2.3 Sinal parcialmente conhecido

Uma situação mais realista surge quando a forma do sinal ainda é conhecida, mas a informação sobre os parâmetros que a definem é incompleta. Em particular, uma situação mais comum é aquela

em que se conhece bem a forma do sinal emitido, mas não os seus parâmetros (muito embora possa estar disponível alguma informação a priori sobre o valor desses parâmetros).

A posteriori do modelo sob H_0 nesta situação ainda é a mesma da equação 2.3; isto porque, quando $\xi = 0$, todos os parâmetros que fazem parte do modelo tonal desaparecem simultaneamente. A hipótese H_0, portanto, induz neste caso a uma redução ainda mais drástica da dimensionalidade do espaço paramétrico. O FBST ainda pode ser aplicado neste caso, mas podem surgir problemas do fato de compararmos um modelo com apenas um parâmetro contra um modelo de muitos parâmetros, como veremos adiante.

Nesta nova versão do problema, porém, a integração da posteriori completa é mais delicada. Uma das fontes de complicação é a dimensionalidade do espaço paramétrico irrestrito: além da frequência fundamental e da variância do ruído, entram no modelo mais dois parâmetros (fase e amplitude) para cada harmônico, resultando num espaço de integração de dimensão pelo menos igual a 4.

Além disso, a integração numérica da posteriori baseada no modelo tonal sofre de uma dificuldade adicional: por definição, essa posteriori possui múltiplos máximos locais em diferentes valores de ω. A forma do modelo, baseada em múltiplos inteiros da frequência fundamental, induz a presença de máximos locais em todos os valores de ω que forem múltiplos inteiros do valor verdadeiro, ou que sejam divisores inteiros deste valor ou de algum de seus múltiplos. Uma vez que os métodos usuais do tipo MCMC são reconhecidamente inferiores na obtenção de amostras de distribuições multimodais, a integração numérica neste caso deve ser feita com cuidado.

Começamos então pela construção da posteriori. Admitimos que o número m de harmônicos é completamente conhecido, e que há alguma informação parcial disponível sobre a frequência fundamental ω do sinal emitido. Esta informação será modelada por uma priori Gaussiana de média $\mu_\omega = 50$ e desvio-padrão $\sigma_\omega = 10$. Com isto, a posteriori para o modelo completo será dada por

$$P(\xi, \Theta|y) \propto (200\pi)^{-1} exp\left[-\frac{(\omega - 50)^2}{200}\right] \times \tag{2.4}$$

$$(2\pi\sigma_r^2)^{-N/2} exp\left[-\frac{1}{2\sigma_r^2}\sum_{i=1}^{N}(y_i - \xi x(t_i))^2\right] \tag{2.5}$$

Neste modelo o espaço paramétrico completo terá dimensão $2m + 3$, com m o número de harmônicos.

Para gerar as amostras deste modelo, e levando em consideração a alta dimensionalidade do espaço e a presença de múltiplos máximos locais, utilizamos um algoritmo de *evolução diferencial* conhecido como *DREAM: DiffeRential Evolution Adaptive Markov Chain*. Este método, cuja ideia principal é utilizar uma distribuição candidata adaptativa, tem bom desempenho na simulação de distribuições multimodais sobre espaços multidimensionais e complicados (Vrugt (2016); Vrugt *et al.* (2009)).

O algoritmo em linhas gerais funciona da seguinte maneira: suponha que iniciemos diversas cadeias de Markov paralelas, começando em pontos distantes do espaço paramétrico. Supondo que a posteriori é multimodal, e se a inicialização do algoritmo for feita de maneira adequada, as diferentes cadeias serão atraídas para pontos de máximo distintos.

Num método MCMC tradicional, uma vez tendo sido atraída para a vizinhança de uma determinada moda, a probabilidade de que a cadeia se distancie desta moda é pequena, e inversamente proporcional à densidade máxima no ponto modal. Para permitir que os pontos explorem mais amplamente o espaço, o algoritmo DREAM propõe então uma evolução diferencial, no seguinte sentido: a cada passo, ao atualizar o estado de uma certa cadeia, digamos a cadeia i, começa-se sorteando aleatoriamente um (ou mais) par de cadeias $(j, k), j \neq i, k \neq i$. Obtém-se então a direção entre o estado atual das cadeias j e k; o ponto candidato para a cadeia i é então obtido dando-se um passo nesta direção, e adicionando-se um ruído adicional com baixa variância. A ergodicidade

é preservada pelo uso do quociente usual de aceitação de Metropolis-Hastings. Para detalhes do método, ver os supracitados Vrugt (2016); Vrugt *et al.* (2009).

O teste do método nesta versão mais complicada do problema foi feito utilizando os mesmos dados simulados na seção anterior. A única diferença entre os resultados anteriores e os resultados (utilizando para a estimação da integral sobre o conjunto surpresa uma amostra de 15.000 pontos após queima de outros 15.000 pontos) da figura 2.2 é a quantidade de informação disponível para o receptor do sinal em cada caso. Na figura, cada curva representa um valor diferente para o SNR da mensagem simulada.

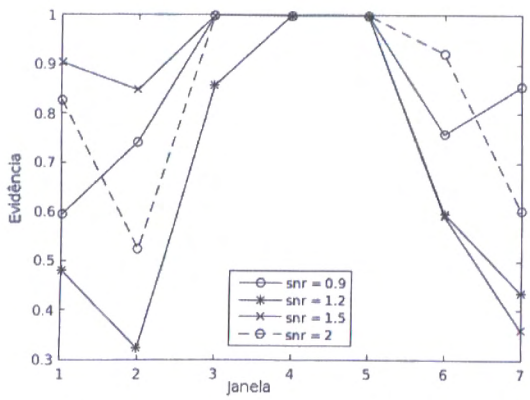


Figura 2.2: Evidência para a presença do sinal em dados simulados: parâmetros desconhecidos

O método de detecção baseado no FBST apresentou bom desempenho, mesmo nesta versão mais complicada do problema: os valores máximos de evidência em cada caso são atribuídos às janelas centrais onde há de fato a presença do sinal. Por outro lado, um valor bastante alto de evidência foi encontrado mesmo para as janelas em que havia apenas a presença do ruído branco.

O principal motivo para os altos valores da evidência contra H_0 quando esta é verdadeira é a especificação do modelo sob H_0. Como notamos acima, a hipótese H_0 impõe uma drástica redução na dimensionalidade do espaço paramétrico; sendo assim, supor H_0 verdadeira parece pouco provável em comparação à situação oposta, mesmo quando o sinal não está de fato presente. Isto se torna ainda mais crítico quando o modelo tonal depende de um grande número de parâmetros; neste caso, estamos comparando um modelo excessivamente parcimonioso (apenas um parâmetro, a variância do ruído) com um modelo altamente flexível, e portanto sujeito ao sobreajuste (*overfitting*). Explica-se assim a alta evidência contra H_0 mesmo nas janelas em que ela é verdadeira.

Para mitigar este problema, precisamos equilibrar a situação, alterando o modelo sob H_0, o modelo irrestrito, ou ambos. Isto deve ser feito conforme o problema a ser analisado, e é o que fazemos na próxima seção.

2.4 FBST para detecção de embarcações

Um dos principais objetivos na análise do sinal acústico obtido do OceanPod é a detecção de embarcações. Num primeiro momento, dada uma amostra longa do sinal (com duração de alguns meses), queremos detectar as embarcações retroativamente, com o intuito de entender melhor os padrões de comportamento dessas embarcações. Esses padrões podem ser utilizados na construção de políticas de fiscalização eficiente pela administração do parque (o parque dista 40 km da costa, o que encarece bastante as viagens de fiscalização). Uma vez tendo construído um detector eficiente, o objetivo é propor, num segundo momento, uma tecnologia de detecção *online* que permita que a administração do Parque responda imediatamente à presença de embarcações.

O sinal emitido que estamos interessados em detectar, portanto, é o ruído irradiado do motor a hélice das embarcações marinhas. Para determinar o melhor modelo para este tipo de sinal, avaliamos a literatura da área de acústica submarina (Arveson e Vendittis (2000); Kozaczka e Grelowska

(2011); McKenna *et al.* (2012); Ogden *et al.* (2011), entre outros). A literatura nesta área é bastante rica, pois o problema de detecção de embarcações tem interesse evidente (inclusive militar) e é alvo de frequentes investigações.

A literatura da área de detecção de embarcações se divide basicamente em duas vertentes: uma, a análise supervisionada de assinaturas acústicas do ruído irradiado de motores. Neste caso, são organizados experimentos em que o pesquisador conhece de antemão o momento em que as embarcações vão passar. Conhece, também, as condições atmosféricas e da água, e conhece todos os detalhes sobre o modelo e o tipo de motor da embarcação. O experimento permite então que seja obtida uma amostra anotada de sinais acústicos, que podem ser analisada diretamente ou utilizada como amostra de treinamento num algoritmo de classificação supervisionada.

Na outra vertente, a literatura trata de situações em que há múltplos detectores; o interesse neste caso é utilizar estratégias de triangulação para obter informações detalhadas sobre o trajeto da embarcação, bem como de sua velocidade e aceleração.

No caso desta tese, porém, nenhuma das duas situações se adequa ao problema: não temos dados antoados de nenhuma ordem, não foram organizados experimentos específicos para guiar o desenvolvimento da estratégia de detecção, e tampouco temos os dados de mais de um detector, já que apenas um hidrofone foi instalado no Parque.

Sendo assim, nossa abordagem deve ser do tipo *não-supervisionada*, e deve utilizar dados de um detector único. Até onde sabemos, não há literatura sobre o problema de detecção binária em situações restritas como essa.

Para obter uma solução nestas condições, uma estratégia possível é utilizar modelos paramétricos construídos com base no conhecimento substantivo dos especialistas na acústica submarina e nas características do ruído irradiado das embarcações. Esta é a abordagem que seguimos nesta tese.

Com a colaboração da equipe do LACMAM (em especial o prof. Linilson Padovese do departamento de Engenharia Mecânica da Poli-USP), analisamos a literatura da acústica submarina, e decidimos incialmente trabalhar também neste caso com o modelo tonal para o sinal emitido. Este é o modelo mais frequente utilizado na literatura da área para descrever o ruído de embarcações. Os parâmetros do modelo, em particular a frequência fundamental e as amplitudes dos harmônicos, são usados na identificação do tipo de motor, da configuração das hélices, e outras características da embarcação.

Porém, ao aplicar o modelo tonal à detecção de embarcações no sinal real obtido do OceanPod, alguns problemas surgiram.

Para entender que tipo de problema pode aparecer neste caso, suponha que tomamos um certo sinal discretizado, digamos de tamanho N, e escolhemos a frequência fundamental $1/N$ para o modelo tonal. Se tomamos um número suficiente de harmônicos, este modelo (as funções cosseno com frequência fundamental e harmônicos, calculada sobre um vetor fixo de instantes de tempo $\{t_i\}$) fornece um conjunto gerador para o espaço dos sinais discretos de dimensão N (em outras palavras, temos neste caso uma base de Fourier para $\Re^N$). Isto significa que um modelo tonal com muitos harmônicos vai permitir um ajuste satisfatório de qualquer amostra (*overfitting*), mesmo que ela contenha apenas ruído. Desta forma, se utilizamos a metodologia da seção anterior, e comparamos o modelo tonal com o modelo Gaussiano de média 0 e variância constante, observamos que o modelo tonal quase sempre fornece um melhor ajuste (principalmente se não conhecemos o valor da frequência fundamental e a deixamos variar livremente). Esta é a principal causa de observarmos valores altos para a evidência de presença do sinal, mesmo na sua ausência (falsos positivos), como vimos na seção anterior.

Além disso, há um segundo problema na utilização do modelo tonal comparado ao modelo de ruído branco na detecção de embarcações. O problema é que, dada a longa duração do sinal obtido do OceanPod, diversos tipos de embarcação podem estar presentes no sinal obtido. Isto inclui barcos grandes (transatlânticos ou petroleiros), viajando a velocidades baixas e a longa distância do hidrofone. Estas embarcações não interessam ao detector; por outro lado, o modelo tonal descreve o ruído irradiado dos barcos grandes tão bem quanto o ruído dos barcos de pesca menores.

Esses dois pontos nos levam a propor um outro tipo de modelo para a detecção de barcos de

pesca no sinal do OceanPod. Por motivos que descrevemos abaixo, um modelo adequado para este caso é o modelo de frequência fundamental variável, da forma:

$$x[t_i] = \sum_{h=1}^{m} A_i cos\left(2\pi h\omega(t_i)t_i + \phi_i\right) \tag{2.6}$$

com

$$\omega(t_i) = \omega_0 + \alpha t_i \tag{2.7}$$

Este modelo é conhecido como modelo de *chirp* na literatura Jaynes (1987). Neste modelo, a frequência é uma função quadrática do tempo, $\omega(t_i) \propto \omega_0 t_i + \alpha t_i^2$.

A principal motivação para adotarmos esse modelo vem da análise do espectrograma de sinais acústicos em duas situações: quando há passagem de barcos grandes (transatlânticos ou petroleiros), e quando há presença de embarcações pequenas. Nas figuras 2.3 e 2.4 vemos um exemplo de cada situação, obtidos do sinal real do OceanPod no Parque Estadual da Laje de Santos. Nesses espectrogramas, regiões de cor mais avermelhada representam regiões de maior energia do sinal.

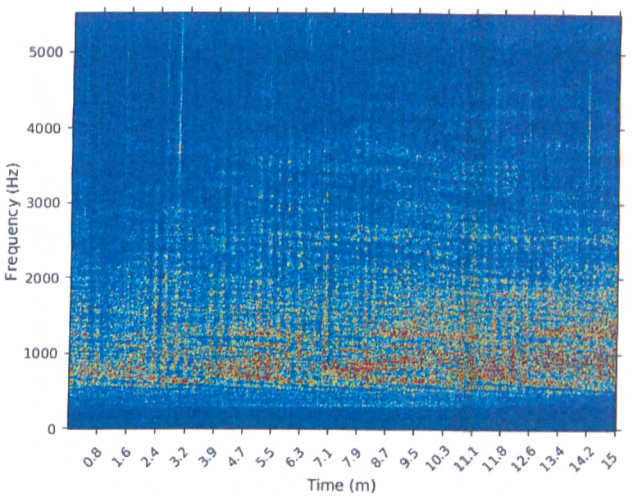


Figura 2.3: Espectrograma da passagem de uma embarcação de grande porte

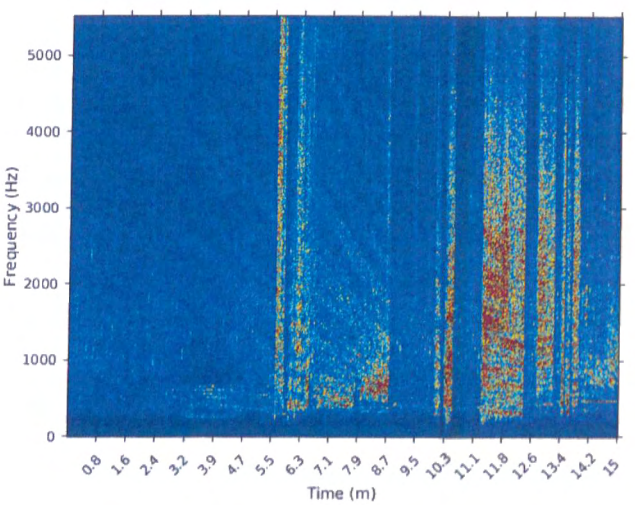


Figura 2.4: Espectrograma da passagem de uma embarcação de pequeno porte

No caso do espectrograma da embarcação de grande porte, vemos linhas com pouca inclinação,

concentradas principalmente entre 500 e $1500Hz$. Além disso, essas linhas se prolongam por diversos minutos (o espectrograma foi obtido numa janela com duração total de 15 minutos).

No caso da embarcação de pequeno porte, por outro lado, temos vários eventos de curta duração e alta energia; além disso, as linhas do espectrograma apresentam inclinação maior. Esta inclinação é relacionada principalmente à aceleração do barco (efeito Doppler).

Uma vez que o interesse do modelo é detectar embarcações de pequeno porte, é natural utilizarmos o modelo com frequência fundamental não-constante para descrever o sinal causado pela passagem de embarcações de pequeno porte. Se alguma outra embarcação está presente, ou se o sinal é composto apenas pelo ruído de fundo, o modelo tonal com frequência constante deve ser suficiente para descrever os dados, e o parâmetro de chirp α torna-se redundante.

É com base nesta observação que formulamos a nova hipótese de detecção como $H_0 : \alpha = 0$, em detrimento da hipótese anterior $H_0 : \xi = 0$. Estamos, desta forma, comparando o modelo *chirp* com um modelo tonal de frequência fundamental fixa. Adotando o modelo tonal como hipótese alternativa, minimizamos a taxa de falsos positivos (pois agora há graus de liberdade suficientes mesmo no modelo simplificado sob H_0), e também diferenciamos os dois tipos de embarcação. Note que, neste novo modelo, o parâmetro de ganho pode ser abandonado, a menos que queiramos utilizá-lo como um grau de liberdade extra para o ajuste.

Mais uma vez, modelamos a informação a priori sobre a frequência fundamental com uma distribuição Gaussiana, com média $\mu_\omega = 40$, e variância $\sigma^2_\omega = 25$. Escolhemos uma priori informativa sobre a frequência fundamental baseados na literatura da área, que aponta para frequências fundamentais no intervalo entre 20 e 40 Hz para embarcações de pequeno porte.

Desta forma, a distribuição a posteriori terá a mesma forma de 2.4, exceto que o modelo para o sinal é agora o modelo de *chirp* da equação 2.6.

Para calcular o valor de evidência contra $H_0 : \alpha = 0$, começamos por obter o máximo a posteriori sob H_0. Este problema já não tem solução analítica neste caso, e deve ser resolvido numericamente.

Os algoritmos de otimização baseados no gradiente não funcionam bem neste caso; a principal razão para isto é o fato de que, no modelo probabilístico tonal, a posteriori se encontra bastante concentrada em torno do máximo (ver discussão em Bretthorst (1992); neste trabalho, o autor propõe utilizar a metodologia de *pattern search* de Hooke e Jeeves (1961), um método de otimização que não utiliza nenhuma informação de primeira ou segunda ordem). Além disso, como já indicamos, a posteriori para o modelo tonal apresenta necessariamente diversos ótimos locais.

Optamos assim por estimar o máximo da posteriori restrita aplicando uma combinação do método DREAM com um algoritmo de pontos interiores: primeiro, usando o método DREAM, rodamos 1.000 iterações de uma cadeia de Markov cuja medida estacionária é a posteriori sob H_0; em seguida, rodamos o algoritmo de pontos interiores (implementado pela função *fmincon* do MATLAB®tendo como chute inicial o máximo da posteriori obtido dessas 1.000 iterações. Em seguida, tornamos a rodar as cadeias, usando como ponto inicial o máximo obtido do algoritmo de pontos interiores, e repetimos esse procedimento 10 vezes, ou até que o máximo assim obtido seja o mesmo por 3 iterações seguidas.

Após a obtenção do máximo da posteriori sob H_0, aplicamos novamente o algortimo DREAM no espaço paramétrico completo para estimar o valor da evidência. Rodamos 5 cadeias em paralelo, por 30.000 pontos após queima de 15.000 pontos.

Note que o número m de harmônicos do modelo tonal não é incluído como parte dos parâmetros simulados via MCMC. Este parâmetro é de natureza especial, pois está diretamente vinculado à dimensionalidade do modelo. Optamos por deixar este valor livre, para que possa funcionar como parâmetro de calibração do método: nos testes abaixo, rodamos o algoritmo com $m = 7$ e $m = 10$ para verificar o impacto da escolha desse parâmetro nos resultados do detector.

Para avaliar o método de detecção baseado no teste de chirp, escolhemos desta vez uma amostra de 10 segundos do sinal real obtido do OceanPod na região do Parque da Laje. Nesta amostra, podemos identificar pela audição do sinal acústico a passagem de uma embarcação de pequeno porte, começando pouco depois de 2 segundos, diminuindo a intensidade até aproximadamente 8 segundos, e cessando completamente nos dois segundos finais.

Para avaliar o comportamento do método na presença de embarcações de grande porte, utilizamos uma segunda amostra de 10 segundos obtida igualmente no Parque da Laje, na qual é possível identificar pela audição do sinal a passagem de uma embarcação desse tipo.

A figura 2.5 apresenta os resultados; nos gráficos, cada ponto representa o valor de evidência calculado para uma janela contígua de $0,5$ segundos.

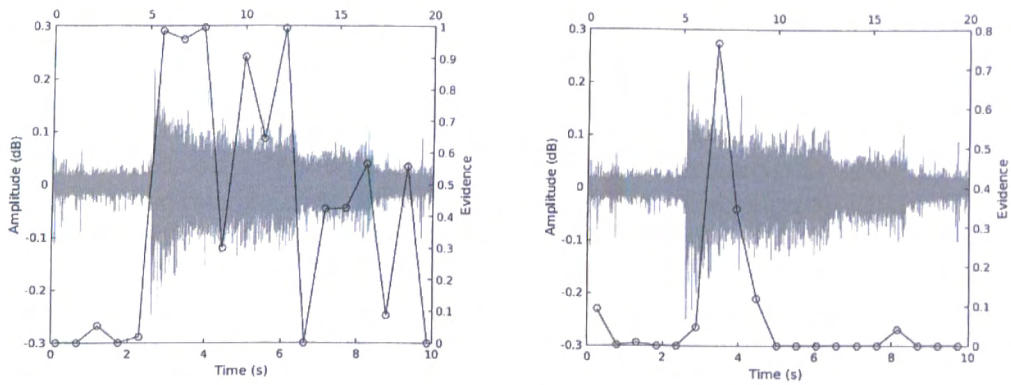


Figura 2.5: Evidência contra H_0 - $m = 7$ (lado esquerdo) and $m = 10$ (lado direito)

O método de teste da taxa de *chirp* foi eficiente para distinguir a presença da embarcação de pequeno porte; o valor da evidência *a favor da presença da embarcação* foi igual a 0, nas primeiras 5 janelas de $0,5$ segundos, e igual a 1 na sexta janela. Há indícios de um único falso positivo importante, na penúltima janela e quando $m = 7$.

O valor de m, o número de harmônicos, teve um efeito sobre a sensibilidade do detector: aumentar o número de harmônicos torna o método menos sensível, sendo necessária a presença de um sinal com chirp e alta potência para que seja acusada a passagem da embarcação. Quando $m = 10$, a evidência pela presença do *chirp* atinge valor máximo em $t = 3,5\ s$, e depois decai rapidamente para 0.

Quando aplicamos o método de teste da taxa de chirp ao sinal correspondente à passagem de uma embarcação de grande porte, o resultado foi de evidência nula para todas as janelas, o que mostra que a inclusão do parâmetro de chirp é suficente para a distinção entre os dois tipos de embarcação.

2.5 Conclusão

A utilização do FBST como método de obtenção de evidência para hipóteses precisas, aliado à utilização do modelo tonal com taxa de chirp e com o modelo probabilístico obtido via máxima entropia mostrou-se uma combinação eficiente e promissora na construção de métodos de detecção binária de sinais. A resposta do método foi satisfatória, tanto na presença do sinal, quanto na sua ausência ou na presença de sinais similares mas com características diversas.

Por outro lado, a necessidade de simularmos valores de uma posteriori complicada e multidimensional dificulta a utilização do método. O problema principal diz respeito à eficiência computacional. Num Pentium Quadricore 1.6 GHz, com 8 Mb RAM, rodando Ubuntu 16.04, o cálculo da evidência para uma janela de $0,5$ segundos de duração (a 11.025 Hz, o que resulta um vetor de tamanho 5.512), levou aproximadamente 15 minutos, um tempo proibitivo.

É possível melhorar esse desempenho; há, porém, outra dificuldade envolvida no algoritmo que desenvolvemos neste capítulo, e que deve ser abordada antes de qualquer esforço para melhorar o desempenho do método. Trata-se da necessidade de se definir, de maneira a princípio *ad hoc*, as janelas sobre as quais se vai aplicar o teste. No exemplo da seção anterior, escolhemos janelas contíguas de $0,5$ segundos, mas esta escolha foi arbitrária.

A dificuldade em se determinar estas janelas se deve ao seguinte fato: na análise dos dados do OceanPod, tempos pouca ou nenhuma informação prévia sobre o instante de início dos eventos

(i.e., em que dias e horários estão passando as embarcações; esta, afinal, é justamente a pergunta que queremos responder), e tampouco sobre sua duração (mesmo as embarcações de pequeno porte podem ter seu motor acionado por diferentes intervalos de tempo, quando estão se aproximando ou afastando, ou quando estão simplesmente executando manobras curtas).

Sendo este o caso, uma abordagem para o problema seria, antes de tudo, determinar seções do sinal em que haja evidência de que *algum evento esteja ocorrendo*, independente de sua natureza específica. Podendo separar tais seções, torna-se mais fácil analisá-las em busca de algum evento em particular.

No próximo capítulo propomos uma abordagem probabilística para resolver este novo problema.

Os resultados descritos neste capítulo foram apresentados no *37th Maximum Entropy Methods in Science and Engineering*, em 2017; o artigo correspondente foi aceito para publicação nos anais do congresso (Hubert *et al.* (2017)).

Capítulo 3

Segmentação não-supervisionada de sinais

Na análise do sinal acústico obtido do OceanPod, o objetivo particular da detecção de embarcações é um caso especial do objetivo mais geral de explorar o sinal em busca de quaisquer eventos significativos.

Tanto num caso quanto no outro, a principal dificuldade é o fato de que não conhecemos os instantes do tempo em que algum evento possa estar ocorrendo; isto se torna ainda mais crítico quando observamos que o sinal é de longa duração e esparso, i.e., existem poucos eventos de curta duração e longos períodos sem nenhum evento.

Por isso, uma estratégia potencialmente útil consiste em buscar instantes de tempo (que podem ser traduzidos, no sinal discreto, por índices de um vetor) em que haja alguma mudança importante nas características do sinal, mudança esta que possa indicar o início de um evento.

Propor uma solução para este problema é o objetivo deste capítulo.

Sendo assim, começamos por supor novamente que temos um sinal recebido discretizado, representado pelo vetor $y \in \Re^N$. Além disso, sabemos que ao longo do tempo $T = N/f_s$ (onde f_s é a taxa de amostragem do sinal contínuo) podem ter ocorrido diversos eventos, definidos como segmentos do sinal que incorporam um ou mais sinais emitidos, além do ruído.

Para manter nossa abordagem tão geral quanto possível, queremos reduzir a um mínimo as suposições que fazemos sobre as características do evento. O que desejamos, portanto, é utilizar uma descrição a mais geral possível daquilo que define um evento no sinal acústico.

Se permanecemos com o modelo aditivo para a composição de ruído mais sinal emitido, uma suposição bastante geral é a de que o início de um evento altera a potência média do sinal (a variância da variável aleatória y_i). No caso do sinal acústico, os eventos serão fenômenos ondulatórios, assim como o ruído. A composição de duas ondas, como se sabe, pode em tese ocorrer de tal maneira que a potência da combinação seja exatamente igual à potência de uma das componentes. Desde um ponto de vista probabilístico, porém, esta combinação exata ocorre apenas num conjunto de medida nula; ou seja, é razoável supor que, no sinal real obtido do sensor, o início de um evento causa uma alteração na potência média, e o final deste mesmo evento causa outra alteração nesta mesma potência.

O modelo introduzido neste capítulo, portanto, terá como objetivo estimar os instantes de início e fim de cada evento, tomando como única suposição a alteração da potência média do sinal na presença de um evento.

Este problema, em uma formulação geral (que não necessariamente define um evento pela alteração na potência), aparece em diferentes contextos (Kuntamalla e Reddy (2014); Makowsky e Hossa (2014); Schwartzman *et al.* (2011); Thedorou *et al.* (2014); Ukil e Zivanovic (2006)) e é conhecido como o problema de segmentação de sinais.

3.1 Modelo para a segmentação do sinal

Assumindo-se a hipótese acima, é natural adotar a seguinte parametrização para o modelo: se no ruído a potência média do sinal é dada por σ_0^2, a partir do momento em que um sinal emitido é incorporado ao ruído, a nova potência se torna $\delta\sigma_0^2$, com $\delta > 0$.

Sendo $y \in \Re^N$ o sinal de duração T discretizado por uma taxa de amostragem igual a f_s, começamos por supor que ao longo desse tempo houve apenas uma alteração na potência média do sinal, e esta alteração ocorreu a partir do índice $1 < \bar{t} < N - 1$. Ou seja, temos que

$$Var(y_i) = \begin{cases} \sigma_0^2 & \text{se } i \leq \bar{t} \\ \delta\sigma_0^2 & \text{se } i > \bar{t} \end{cases} \tag{3.1}$$

Sem nenhuma premissa adicional para o sinal exceto que $E(y_i) = 0$, adotamos por máxima entropia o modelo Gaussiano para y

$$y_i \sim \begin{cases} \mathcal{N}(0, \sigma_0^2) & \text{se } i \leq \bar{t} \\ \mathcal{N}(0, \delta\sigma_0^2) & \text{se } i > \bar{t} \end{cases} \tag{3.2}$$

O modelo acima nos leva à seguinte forma para a verossimilhança:

$$\mathcal{L}(\bar{t}, \sigma_0^2, \delta | y) = (2\pi\sigma_0^2)^{-N/2}\delta^{-(N-\bar{t})/2} exp\left[-\frac{\sum_{i=1}^{\bar{t}} y_i^2}{2\sigma_0^2} - \frac{\sum_{i=\bar{t}+1}^{N} y_i^2}{2\delta\sigma_0^2} \right] \tag{3.3}$$

Nosso interesse é na estimação de $\bar{t}$; por isso, em primeiro lugar, adotamos uma *priori* uniforme e imprópria para δ e procedemos à marginalização analítica de 3.3 obtendo

$$\mathcal{L}(\bar{t}, \sigma_0^2 | y) = \int_0^\infty \mathcal{L}(\bar{t}, \sigma_0^2, \delta | y) d\delta \tag{3.4}$$

$$\propto \left[\frac{\sum_{i=\bar{t}+1}^{N} y_i^2}{2\sigma_0^2} \right]^{-\frac{(N-\bar{t}-6)}{2}} \Gamma\left(\frac{N - \bar{t} - 2}{2} \right) exp\left[-\frac{\sum_{i=1}^{\bar{t}} y_i^2}{2\sigma_0^2} \right] \tag{3.5}$$

A adoção de uma priori uniforme ainda obedece ao objetivo de manter máxima generalidade no modelo; não supomos portanto conhecimento algum acerca da intensidade do sinal emitido, em comparação ao ruído de fundo.

Caso a variância σ_0^2 do sinal entre $i = 1$ e $i = \bar{t}$ seja conhecida, a verossimilhança marginal acima já pode ser utilizada diretamente na obtenção da *posteriori* para $\bar{t}$; caso contrário, é possível ir um passo adiante, adotando a *priori* de Jeffreys para σ_0^2, dada por $\pi(\sigma_0) = 1/\sigma_0$, e marginalizando analiticamente o parâmetro σ_0 para obter

$$\mathcal{L}(\bar{t} | y) = \int_0^\infty \mathcal{L}(\bar{t}, \sigma_0^2 | y) \pi(\sigma_0) d\sigma_0 \tag{3.6}$$

$$\propto \left(\sum_{i=\bar{t}+1}^{N} y_i^2 \right)^{-\frac{(N-\bar{t}-6)}{2}} \Gamma\left(\frac{N - \bar{t} - 2}{2} \right) \left(\sum_{i=1}^{\bar{t}} y_i^2 \right)^{-\frac{(\bar{t}+6)}{2}} \Gamma\left(\frac{\bar{t} + 6}{2} \right) \tag{3.7}$$

A partir desse ponto, para calcularmos o numerador da *posteriori* para $\bar{t}$, basta multiplicar uma das expressões acima (conforme supomos σ_0^2 conhecido ou não) pela *priori* $\pi_t(t)$.

$$P(\bar{t}|y) \propto \pi(t) \cdot \left(\sum_{i=1}^{\bar{t}} y_i^2\right)^{-\frac{(\bar{t}+6)}{2}} \left(\sum_{i=\bar{t}+1}^{N} y_i^2\right)^{-\frac{(N-\bar{t}-6)}{2}} \times \tag{3.8}$$

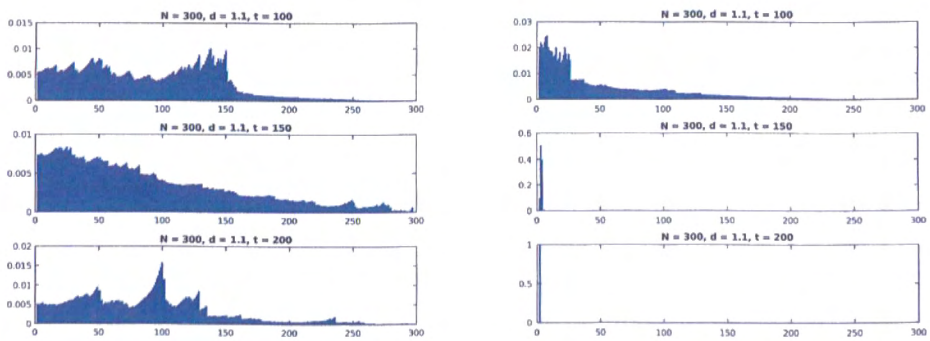
$$\Gamma\left(\frac{\bar{t}+6}{2}\right)\Gamma\left(\frac{N-\bar{t}-2}{2}\right) \tag{3.9}$$

Essa é uma distribuição discreta com suporte em $2 < \bar{t} < N-2$; é possível, portanto, obtê-la para todos os pontos de seu suporte e assim calcular também a constante de normalização. A partir daí, o valor de $\bar{t}$ pode ser estimado pelo máximo *a posteriori*, ou pela média.

A relação entre δ e N é crucial para que esse modelo capture corretamente a mudança no regime da potência; com N pequeno, a variância da estimativa da potência média terá erro padrão bastante grande, suficiente para encobrir completamente os efeitos de um δ que seja igualmente pequeno. Sendo assim, essa abordagem deve ter sucesso em situações onde o sinal é de longa duração, e / ou quando os eventos tiverem valores altos de SNR. Formalmente, a qualidade do estimador calculado a partir do modelo acima será tanto melhor quanto menor for a intersecção entre os intervalos de credibilidade para as estimativas de σ_0^2 e $\delta\sigma_0^2$; a amplitude desses intervalos depende inversamente de N, e a distância entre os centros do intervalo depende diretamente de δ.

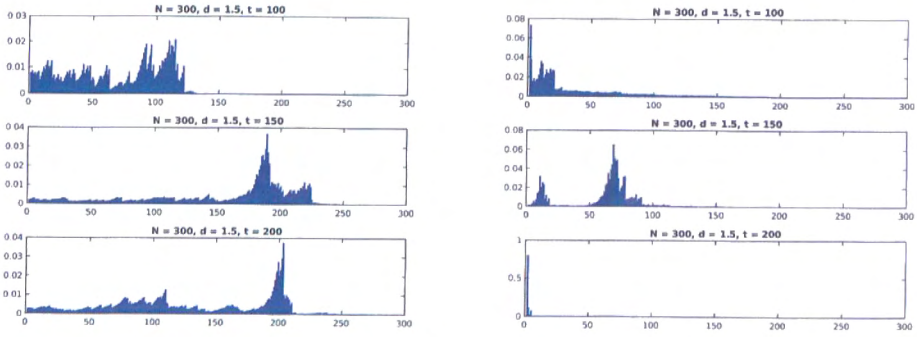
Para verificar essas observações, simulamos um sinal $y \in \Re^N$, com $y_i \sim \mathcal{N}(0, \sigma^2(i))$, onde $\sigma^2(i) = \sigma_0^2$ se $i \leq \bar{t}$ e $\sigma^2(i) = \delta\sigma_0^2$ caso contrário. Adotamos $N \in \{300, 3000, 30000\}$, $\delta \in \{1, 1, 1, 5, 2\}$ e $\sigma_0^2 = 1$. Para o parâmetro de interesse $\bar{t}$, adotaremos os valores $N/3, N/2, 2N/3$. Adotamos uma *priori* uniforme para $\bar{t}$.

Os resultados aparecem nas figuras 3.1 a 3.9.



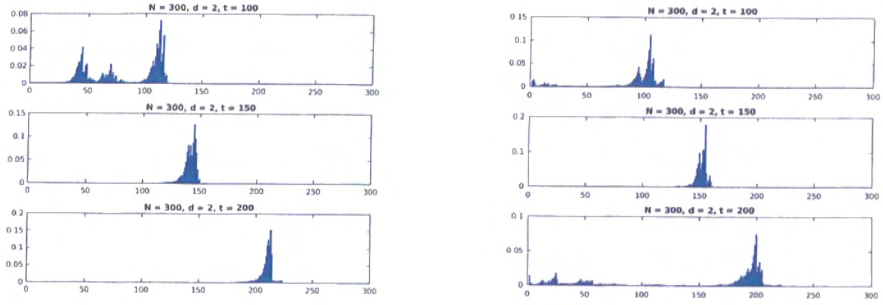
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.1: *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 1, 1$



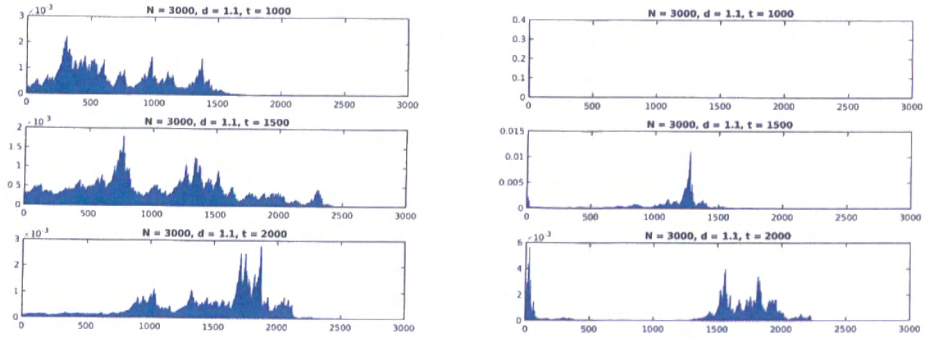
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.2: *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 1,5$



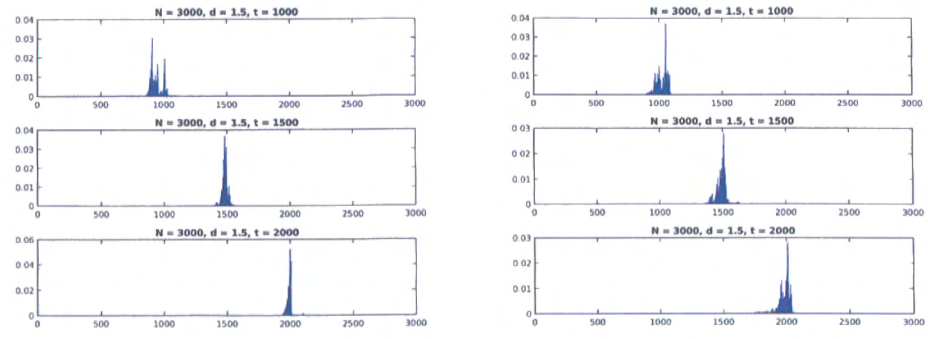
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.3: *Posteriori* para o ponto de corte em dados simulados: $N = 300$, $\delta = 2$



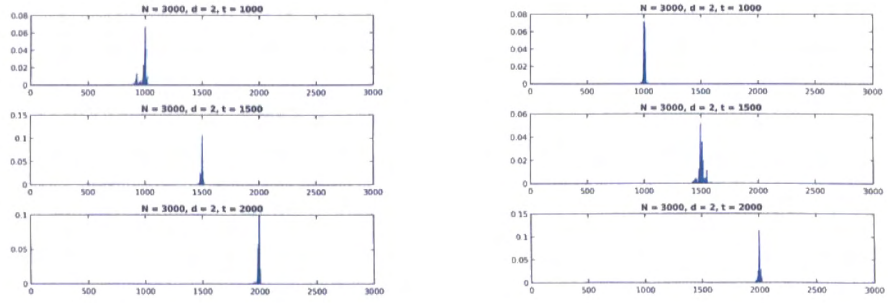
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.4: *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 1,1$



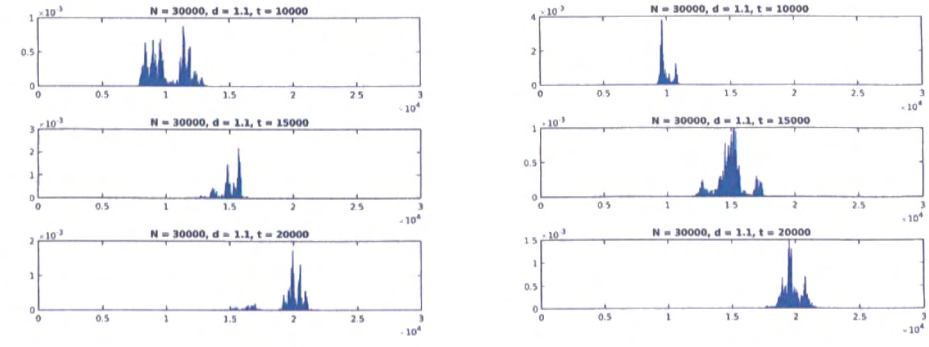
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.5: *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 1,5$



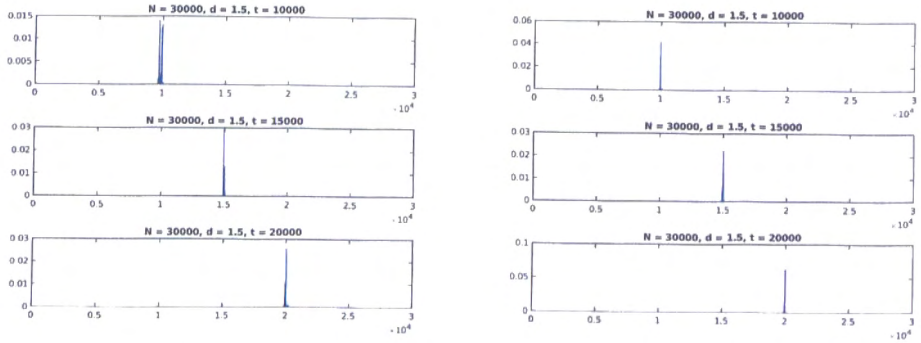
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.6: *Posteriori* para o ponto de corte em dados simulados: $N = 3000$, $\delta = 2$



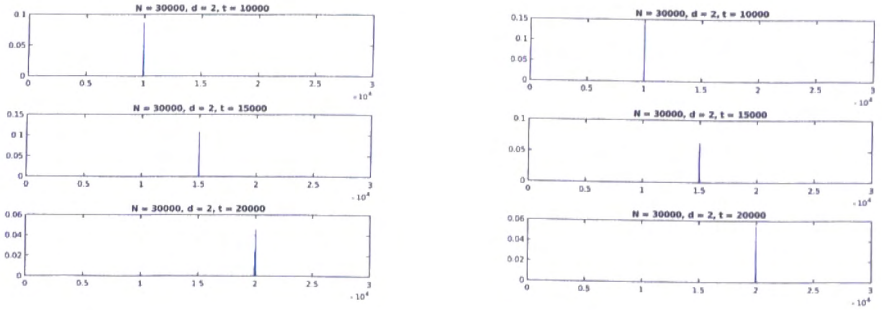
(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.7: *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 1,1$



(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.8: *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 1,5$



(a) σ_0^2 conhecido (b) σ_0^2 desconhecido

Figura 3.9: *Posteriori* para o ponto de corte em dados simulados: $N = 30000$, $\delta = 2$

Os resultados da simulação confirmam as observações anteriores: para um sinal curto, $N = 300$, apenas o sinal com $\delta = 2$ foi segmentado corretamente; para sinais mais longos, porém, o método apresenta bons resultados mesmo quando a intensidade da alteração na potência média não é tão dramática. Sendo assim, há boas perspectivas para sua aplicação à análise de sinais acústicos, onde temos sinais sempre longos, e buscamos tipicamente eventos de alta potência (no caso da detecção de embarcações).

Nas situações em que a *posteriori* tem extremo distante do ponto de mudança de regime, é frequente encontrarmos esses extremos nos pontos próximos aos limites do suporte da posteriori. Este fenômeno é um artefato da diferença entre os tamanhos das janelas quando $\bar{t}$ está proximo a 1 ou $N - 1$; uma estratégia possível para tratar essa situação pode ser incorporar informação *a priori* sobre o ponto de corte, se for possível assumir que ele se encontra mais próximo ao centro do intervalo do que aos seus extremos. No caso limite, por exemplo, podemos impor uma *priori* que é 0 se $|\bar{t} - N/2| > S$, e 1 caso contrário; na prática, isto implica em calcular a *posteriori* acima apenas nos pontos de probabilidade *a priori* positiva.

O modelo que construímos até aqui partiu da hipótese de que haveria apenas uma mudança na potência do sinal. Esta suposição não é realista, especialmente em sinais mais longos. Sendo assim, consideramos em seguida o caso em que há mais de uma mudança de potência no sinal recebido. Isto ocorre, por exemplo, na situação em que o sinal contém início e fim de um certo evento específico.

Uma possibilidade para tratar este caso seria estender o modelo anterior, escrevendo

$$Var(y_i) = \begin{cases} \sigma_0^2 & \text{se } i \leq \bar{t}_1 \\ \delta_1 \sigma_0^2 & \text{se } \bar{t}_1 < i \leq \bar{t}_2 \\ \delta_2 \sigma_0^2 & \text{se } i > \bar{t}_2 \end{cases} \tag{3.10}$$

com $\bar{t}_1 < \bar{t}_2$ e assumindo $\delta_1 \neq \delta_2$. Este modelo pode ser estendido para um número qualquer k de pontos de corte; teríamos, então, uma função $v[i]$, em $i \in [1, \ldots, N]$, constante por pedaços e com k descontinuidades. O modelo probabilístico em essência não se altera. A dimensão do conjunto suporte da *posteriori*, porém, aumenta exponencialmente com k: no caso $k = 1$, temos $N - 2$ pontos nesse conjunto; no caso $k = 2$, teríamos $o(N^2)$ pontos, e assim sucessivamente.

Além disso, na análise do problema geral, surge o problema em que k é desconhecido; este problema, embora apresente um interesse intrínseco, torna a análise bastante complexa e os métodos numéricos computacionalmente intensivos (pois será necessário percorrer todos os possíveis valores de k, obtendo a *posteriori* em cada caso).

Para evitar essas dificuldades, outra possibilidade é a aplicação sequencial do modelo para mudança única de regime, da seguinte forma: dado um sinal $y \in \Re^N$, estimamos $\bar{t}$ conforme o modelo inicial; disto resultam dois novos sinais, $y_1 \in \Re^{\bar{t}}$ e $y_2 \in \Re^{N-\bar{t}}$. Aplicamos em seguida a estimação do ponto de corte novamente para y_1 e y_2, e assim sucessivamente até que um certo critério de parada seja atingido.

Esta abordagem funciona pois o método de segmentação com ponto de corte único sempre procura formar os segmentos de forma a maximizar a diferença de potência entre eles; veja, por exemplo, as figuras 3.10 e 3.11. Nestas figuras, simulamos um sinal com dois pontos de mudança da potência total, e aplicamos a estimativa de segmentação assumindo uma única alteração na potência. Como vemos, a estimativa do ponto de corte coincide com um dos pontos corretos.

Desta maneira, é natural proceder sequencialmente e aplicar a segmentação única sucessivas vezes, obtendo assim uma sequência de pontos de corte do sinal, onde cada ponto sucessivo divide um segmento em subsegmentos.

3.2 Critérios de parada do algoritmo de segmentação

Aplicando o método de segmentação de forma sequencial, a questão que surge é a de definir um critério de parada. Sem um tal critério, o algoritmo prosseguiria indefinidamente até segmentar o sinal na máxima granularidade possível.

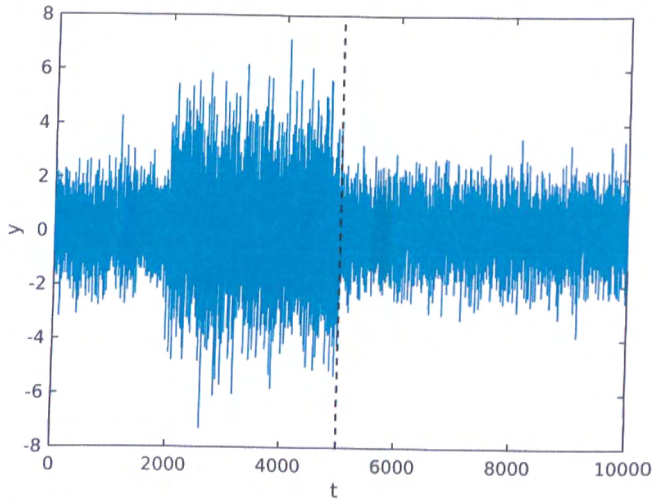


Figura 3.10: Segmentação estimada para um sinal com dois pontos de corte

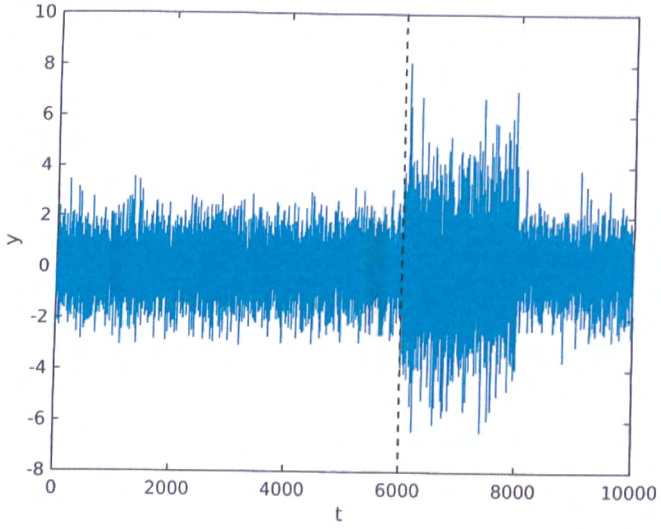


Figura 3.11: Segmentação estimada para um sinal com dois pontos de corte, caso 2

Para definir este critério, lembramos que o modelo de segmentação definido na seção anterior está construído para detectar alterações na potência média do sinal. Sendo assim, um critério natural de parada para a segmentação sequencial será parar a segmentação quando não houver mudança de potência dentro de um segmento.

Na parametrização da seção anterior, isto equivale a parar a segmentação quando $\delta = 1$. Não observamos, porém, o valor exato de δ; somos capazes apenas de obter uma estimativa construída com base no sinal amostrado. Por isto, não é possível verificar exatamente se $\delta = 1$; em lugar disso, observaremos o valor estimado para δ, e adotaremos como critério de parada o teste estatístico da hipótese $H_0 : \delta = 1$.

No modelo de máxima entropia que estamos adotando, se os segmentos (obtidos pela separação de um sinal contíguo original) são dados por $y_1 \in \Re^{n_1}$ e $y_2 \in \Re^{n_2}$, com $y_{1,i} \sim \mathcal{N}(0, \sigma_0^2)$, $y_{2,i} \sim \mathcal{N}(0, \delta\sigma_0^2)$, a hipótese $H_0 : \delta = 1$ corresponde a uma hipótese de igualdade de variâncias entre duas amostras de populações normais.

Para calcular a evidência contra essa hipótese, a literatura estatística clássica propõe alguns testes, sendo os mais conhecidos o teste baseado na estatística F, e o teste de Levene Levene (1960). Esses testes estão baseados no conceito de *p-valor*; de maneira geral, esse conceito representa uma

medida de evidência contra a hipótese, mas que é calculado sobre o espaço amostral. Define-se uma estatística de teste, $s(y)$, e obtém-se sua distribuição, condicional aos valores dos parâmetros. Calcula-se por fim a medida, sob H_0, da(s) cauda(s) dessa distribuição, onde define-se cauda pelas regiões do espaço amostral com valores da estatística s mais extremos do que o valor na amostra efetivamente observada.

O conceito de *p-valor*, muito embora constitua o padrão em algumas áreas de pesquisa que aplicam a estatística, tem diversas limitações (para discussões interessantes e instrutivas a esse respeito, ver por exemplo Cox (1977); Good (1992); Pereira e Wechsler (1993); Stern (2008)). Em particular, p-valores calculados sem especificação da hipótese alternativa podem apresentar resultados enganadores, dando origem à situações do tipo *aumente o tamanho da amostra para rejeitar*. Com amostras grandes, a distribuição da estatística de teste tipicamente terá dispersão pequena, que levará à rejeição da hipótese nula mesmo para pequenos desvios e em situações em que alguma hipótese alternativa teria poder explicativo superior.

Sabe-se que esta deficiência pode ser compensada por uma estratégia de escolha do *nível de significância* do teste dependente do tamanho da amostra L. Pericchi (2016). Neste contexto, quanto maior o tamanho da amostra, mais abrangente será a definição de evento extremo (i.e., do que de fato define a cauda de uma distribuição).

Essa discussão tem relevância no presente contexto, pois na análise de sinais acústicos temos amostras de tamanho tipicamente grande, que tendem a gerar p-valores bastante baixos mesmo para situações em que a hipótese nula é verdadeira (ou seja, tenderão a continuar segmentando o sinal mesmo quando a diferença de potência entre segmentos for pequena). Portanto, para construirmos um critério de parada no algoritmo de segmentação, teremos que levar essa questão em consideração. Em termos práticos, isto significa que temos de considerar a sensibilidade do critério de parada, impedindo que a hipótese $\delta = 1$ seja rejeitada mesmo para desvios mínimos, e ao mesmo tempo impedindo que o algoritmo de segmentação "pare antes da hora", não indicando a segmentação de trechos com características distintas.

Na linguagem da inferência estatística, precisamos atingir um equilíbrio entre erro do tipo I (rejeição de hipótese nula verdadeira, evento falso positivo) e erro do tipo II (não-rejeição de hipótese falsa, evento falso negativo).

Na teoria estatística, este equilíbrio é atingido tipicamente através da construção de uma **região de rejeição**, ou região crítica, adequada para um teste e para as características da amostra.

Estudamos em seguida três alternativas para a construção da região crítica: a primeira é a aplicação do teste F tradicional com nível de significância fixo. A segunda será baseada no teste da razão de verossimilhanças, mas adotando um nível de significância adaptativo conforme os trabalhos de Perez e Pericchi (2014) e L. Pericchi (2016).

A terceira alternativa é baseada na medida de evidência do FBST (Pereira e Stern (1999)). Lembrando que o FBST é um teste especialmente desenhado para o teste de hipóteses precisas (hipóteses de igualdade, como $H_0 : \delta = 1$), e cujo desempenho é frequentemente superior aos procedimentos estatísticos padrão nestes casos, a aplicação do FBST se justifica neste contexto. Neste caso, a sensibilidade do teste será controlada via a introdução de uma *priori* subdispersa sobre δ.

No restante dessa seção, trabalhamos com dois sinais $y_1 \in \Re^{n_1}$ e $y_2 \in \Re^{n_2}$, sob o modelo $y_{1,i} \sim \mathcal{N}(0, \sigma_0^2)$ e $y_{2,i} \sim \mathcal{N}(0, \delta\sigma_0^2)$.

3.2.1 Critério baseado na estatística F

Definimos a estatística

$$f(y_1, y_2) = \frac{(n_2 - 1)\sum_{i=1}^{n_1} y_{1,i}^2}{(n_1 - 1)\sum_{i=1}^{n_2} y_{2,i}^2} \tag{3.11}$$

A estatística f nada mais é do que a razão entre os estimadores não-viciados para a potência média de cada janela. Ou seja, na nossa parametrização, f equivale a um estimador para δ.

Sob $H_0 : \delta = 1$, e assumindo normalidade para $y_{1,i}$ e $y_{2,i}$, temos que $f(y_1, y_2) \sim F(n_1 - 1, n_2 - 1)$. É portanto imediato calcular o p-valor $P(F > f(y_1, y_2)|H_0)$ (ou $P(F < f(y_1, y_2)|H_0)$, conforme $f > 1$ ou $f < 1$ respectivamente).

O p-valor é a medida de evidência estatística mais tradicional e amplamente utilizada. Para construir um critério de decisão baseado nele, frequentemente adota-se um valor crítico de referência, por exemplo 0.05; então, p-valores menores do que esta referência são considerados suficientemente pequenos para levar à rejeição da hipótese nula (ou seja, o algoritmo não para).

Por outro lado, se conhecemos N, o tamanho do sinal (ou mais precisamente se conhecemos n_1 e n_2), a escolha pelo nível crítico (chamado nível de significância) determina univocamente um certo valor limite para a estatística f de teste. Poderíamos, então, igualmente escolher primeiro esse limite, f_c, e depois calcular o nível de significância correspondente.

Uma vez que a estatística f é um estimador para δ, a escolha de f_c tem uma interpretação direta no contexto do algoritmo de segmentação: estamos estabelecendo um valor de corte para o efeito do evento na potência total do sinal, e decidimos parar o algoritmo se a estiamtiva desse efeito estiver suficientemente próxima de 1, por exemplo entre 0.9 e 1.1.

Este critério seria o mais simples possível, pois exigiria apenas a avaliação da estatística de teste. Por outro lado, a cada iteração do algoritmo de segmentação o tamanho dos sinais se altera. A distribuição de probabilidade da estatística de teste então também se altera, ou seja, a escala em que a estatística é medida muda. Logo, os valores críticos definidos em um caso não tem a mesma interpretação em outro.

Por esta razão, o mais comum nas aplicações da teoria de testes de hipóteses é a escolha por um nível de significância, um valor limite em $]0, 1[$, a ser comparado com o p-valor obtido da amostra. A escala de probabilidades é, pelo menos intuitivamente, a mesma nos dois casos; assim, fixar o nível de significância é mais prático, e garante que a cada experimento o valor crítico da estatística de teste vai se alterar de acordo com a variação no tamanho da amostra.

3.2.2 Critério da razão de verossimilhanças com significância adaptativa

O procedimento de fixar o nível de significância tem alguns custos, porém. Good (1992) discute esses custos já em 1992; mais recentemente, Perez e Pericchi (2014) e L. Pericchi (2016) mostram que esse procedimento tem o efeito colateral de gerar testes com erro do tipo II (falsos negativos) diminutos, e decrescentes com o tamanho da amostra (de forma que o quociente entre os dois tipos de erro, α/β, cresce sem limite).

Como alternativa para o procedimento padrão (encontrar o teste de máximo poder com nível de significância fixo), Pereira *et al.* (2017) propõem obter um teste que minimize uma combinação linear entre os erros do tipo I e II, $A \cdot \alpha + B \cdot \beta$. Este teste é obtido utilizando-se como medida de evidência uma **razão de verossimilhanças médias**, ou **fatores de Bayes**. Ou seja, o teste ótimo, segundo este critério da combinação dos erros, é um teste Bayesiano (se aceitamos tomar médias ponderadas das verossimilhanças, e igualmente aceitamos interpretar os pesos como densidades de probabilidade *a priori* no espaço paramétrico).

O método geral de construção desse teste, porém, exige a avaliação de algumas integrais complicadas Pereira *et al.* (2017), que em muitos casos terão de ser obtidas via métodos MCMC. Pensando nisso (e também numa possível rejeição *a priori* de métodos Bayesianos, que postulam probabilidades *a priori* sobre os parâmetros do modelo), Perez e Pericchi (2014) propõem um método para aproximar a função de decisão do teste ótimo usando a função de decisão calculada com base no p-valor tradicional; esta aproximação é baseada na adoção de *níveis de significância adaptativos*, que variam conforme o tamanho da amostra.

A ideia é definir um nível de significância adequado para um certo experimento (isto é, para um certo tamanho de amostra N_0), e calcular o novo nível de significância para um novo experimento (com amostra de tamanho N). A fórmula que permite obter o novo nível de significância é a *regra da raiz de n log(n)* de Perez e Pericchi (2014):

$$\alpha(n) = \frac{\alpha\sqrt{n_0 \cdot (log(n_0) + \chi^2_\alpha(1))}}{\sqrt{n \cdot (log(n) + \chi^2_\alpha(1))}}$$

onde n_0 é o tamanho da amostra para o primeiro teste (o *experimento de referência*), α_0 é a significância obtida para este mesmo teste, e $\chi^2_\alpha(1)$ é o quantil α da qui-quadrado com 1 grau de liberdade.

Com base nesta regra, precisamos fixar um tamanho de amostra n_0 para o qual obteremos o primeiro nível de significância α_0. Em seguida, diante de um novo experimento com amostra de tamanho n, calculamos o p-valor segundo o método tradicional (o teste da razão de verossimilhanças de Neyman-Pearson-Wilks), e rejeitamos a hipótese nula se esse p-valor estiver abaixo do nível de significância $\alpha(n)$.

No contexto da segmentação de sinais, cada segmento representará um evento, que em seguida desejaremos classificar. Faz sentido, portanto, fixar o tamanho n_0 a partir da mínima duração que aceitamos para um evento; o produto do valor em segundos desta duração pela taxa de amostragem do áudio fornecerá o tamanho das janelas n_0. Escolhemos em seguida o mínimo SNR aceitável, δ_c, e obtemos o nível de significância. Na tabela 3.1 abaixo, exibimos algumas combinações possíveis para estes valores.

Duração mínima	SNR mínimo	Taxa de falsos positivos
1s	1%	0,2989
1s	5%	0,0035
1s	10%	1,6e-08
1s	20%	6,2e-32
2s	1%	0,2278
2s	5%	7e-05
2s	10%	2,6e-15
2s	20%	8,0e-62

Tabela 3.1: Combinações para tamanho do evento, SNR mínimo e taxa de falsos positivos (significância)

3.2.3 Critério baseado no FBST

O método Bayesiano ortodoxo para testes de hipóteses, que utiliza a razão de verossimilhanças médias (ou fatores de Bayes) sob H_0 e sob H_1, encontra um dilema quando as hipóteses H_0 e H_1 definem espaços paramétricos de dimensões distintas. Isto ocorre porque a distribuição *a priori* (ou os pesos a serem usados na média da verossimilhança) devem ser definidos sobre o espaço paramétrico completo, de dimensão d. Sendo assim, se $dim(\Theta_H) < d$, a medida do conjunto $\Theta_H \subset \Theta$ será necessariamente nula.

Para tratar essa situação sem recorrer à atribuição artificial de medida positiva a conjuntos de medida nula, vamos utilizar novamente o FBST de Pereira e Stern (1999), que descrevemos na seção 2.1.

No caso do modelo de segmentação, para construir o modelo probabilístico definimos a distribuição *a priori* dos parâmetros da seguinte forma: para σ_0 propomos uma *priori* invariante de Jeffreys, $\pi_\sigma(s) = 1/s$; para δ, por outro lado, queremos uma *priori* informativa, para representar o conhecimento de que o sinal real do OceanPod é esparso.

Uma distribuição que tem sido usada frequentemente em situações de esparsidade é a distribuição de Laplace. A densidade dessa distribuição, centrada em $d = 1$, é dada por

$$\pi_\delta(d) = \frac{1}{\beta} e^{-\frac{|d-1|}{\beta}} \tag{3.12}$$

Esta distribuição é simétrica em torno de $d = 1$, e não-diferenciável precisamente nesse ponto. Ela atribui densidade alta para $\delta = 1$, tanto mais alta conforme o valor do hiperparâmetro β: se β é pequeno, haverá um pico pronunciado no ponto $d = 1$; conforme β cresce, a distribuição se "achata"; a densidade da Laplace para diferentes valores de β aparece na figura 3.12.

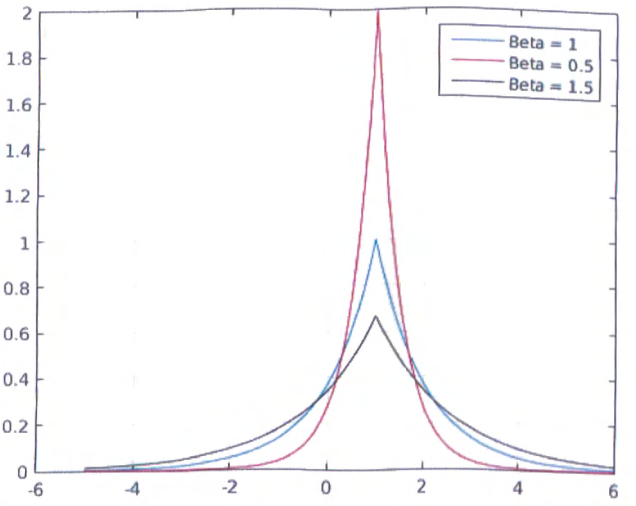


Figura 3.12: Densidade para a distribuição de Laplace

A escolha por essa distribuição é interessante pois permite que nossa expectativa acerca da potência do sinal emitido (i.e. a potência do evento) seja ajustada com um único parâmetro, β. Diminuir o valor de β tem o efeito de induzir um pico na *posteriori* ao longo de $\delta = 1$. Tende, portanto, a aumentar a medida em torno de H_0, favorecendo-a.

Uma vez obtida a medida de evidência a favor da hipótese, ainda resta o problema de construir uma regra de decisão. Na teoria dos testes Bayesianos, isso tradicionalmente é feito via teoria das decisões, onde uma certa função de perda é definida para controlar simultaneamente os dois tipos de erro.

Em Madruga e L. G. Esteves (2001), demonstra-se que o FBST é a estatística ótima de teste para uma certa função de perda, dada por

$$L(R, \theta) = a\,[1 - \mathbf{I}(\theta \in T(y))] \tag{3.13}$$

$$L(A, \theta) = b + c \cdot \mathbf{I}(\theta \in T(x)) \tag{3.14}$$

onde $L(R, \theta)$ $(L(A, \theta))$ é a perda incorrida pela rejeição (aceitação) da hipótese nula, quando o valor do(s) parâmetro(s) é θ, $\mathbf{I}(\Xi)$ é a função característica do conjunto Ξ, e $T(y)$ é o **conjunto surpresa** definido por

$$T(y) = \{\theta : P(\theta|X) > sup_{\Theta_0} P(\theta|X)\}$$

O teste de mínima perda esperada para as funções L é dado finalmente pelo seguinte critério: aceitar H_0 sempre que

$$ev(H_0, x) = 1 - \bar{ev}(H_0, x) > \frac{b + c}{a + c}$$

Analisando o critério de decisão acima, vemos que a hipótese será sempre rejeitada caso $a < b$. Além disso, se c cresce enquanto a e b permanecem fixos, a aceitação de H_0 se tornará cada vez mais difícil (o quociente converge para 1). Por fim, no caso em que c é mantido fixo, o critério de decisão se aproxima de 0 conforme b/a se aproxima de 0.

A função de perda na equação 3.13 tem uma característica interessante: ela não depende do espaço paramétrico restrito sob H_0, exceto por intermédio do conjunto surpresa. A perda, portanto, depende não de uma possível igualdade entre θ e θ_0, mas sim da presença ou ausência de θ (o valor "verdadeiro" do parâmetro) no conjunto surpresa. A constante a, por exemplo, é a perda incorrida por uma rejeição da hipótese caso o verdadeiro valor do parâmetro não se encontre no conjunto surpresa; pode ser interpretada, portanto, como o custo percebido pelo pesquisador em afirmar que $\theta \notin \Theta_0$, quando os dados de fato atribuem maior evidência em θ_0 do que em θ.

A constante b, por outro lado, é a perda incorrida por uma aceitação de H_0 quando $\theta \notin T(y)$. Podemos então interpretar a/b como uma medida do quão mais confiantes estamos na aceitação de H_0, relativamente à sua rejeição, quando a densidade da *posteriori* é maior em θ_0 do que em θ.

Já o valor de c pode ser interpretado da seguinte forma: c é o acréscimo na perda por aceitação, quando $\theta \in T(y)$; ou seja, representa o quanto aumenta nossa confiança na rejeição caso os dados passem a fornecer alta evidência para θ, em comparação a θ_0.

3.3 Algoritmo sequencial para segmentação do sinal

Uma vez fixado um critério de parada CP, o algoritmo sequencial de segmentação pode ser formulado de maneira recursiva, ver algoritmo 1.

Algorithm 1 Segmentação sequencial de sinais (SeqSeg)

procedure SEQSEG($(y$, *minlength*$)$)
  $N = dim(y)$
  $\bar{t} = argmax_t P(t|y)$ ▷ Otimiza a posteriori para t
  if $\bar{t} <$ *minlength* **ou** $N - \bar{t} <$ *minlength* **then**
    retorna o vetor vazio
  else
    $y^1 = \{y_i : i = 1, ..., \bar{t}\}$
    $y^2 = \{y_i : i = \bar{t} + 1, ..., N\}$
    if $CP(y^1, y^2) = 1$ **then** ▷ Critério de parada
      $t_1 =$ **SeqSeg**(y^1, *minlength*)
      $t_2 =$ **SeqSeg**(y^2, *minlength*)
      retorna $[t_1, \bar{t}, \bar{t} + t_2]$
    else
      retorna o vetor vazio
    end if
  end if
end procedure

Nesta definição, incluímos o parâmetro *minlength*, o mínimo tamanho de sinal que aceitamos tentar segmentar, para garantir que o algoritmo estará bem definido para qualquer critério de parada (que agora poderiam ser melhor denominados critérios de parada *precoce*). Ao final da execução deste algoritmo, o resultado será um vetor ordenado $\tau \in \mathbb{N}^m$, onde m é o número de mudanças de regime de potência ao longo do sinal, e os componentes do vetor indicam as posições de cada mudança de regime.

O critério de parada CP retorna 1 para aceitação de H_0, e 0 para rejeição, e é definido conforme o caso:

Algorithm 2 Critério de parada F

procedure $CP_F((y^1, y^2, \alpha))$
  Obtenha a estatística f para o teste de igualdade de variâncias
  Obtenha da distribuição F o p-valor p associado à estatística f
  if $p < \alpha$ **then**
    Retorna 1
  else
    Retorna 0
  end if
end procedure

Algorithm 3 Critério de parada LR

procedure $CP_{LR}((y^1, y^2, \alpha))$
  $LR \leftarrow -2log\left(\frac{sup_{\theta \in \Theta_0} \mathcal{L}(y_1, y_2, \theta)}{sup_{\theta \notin \Theta_0} \mathcal{L}(y_1, y_2, \theta)}\right)$
  Obtenha da distribuição $\chi^2(1)$ o p-valor p associado a LR
  $\alpha_{ad} = f(n, n_0, \alpha_0)$ ▷ Obtém o nível de significância adaptativo
  if $p < \alpha_{ad}$ **then**
    Retorna 1
  else
    Retorna 0
  end if
end procedure

Algorithm 4 Critério de parada FBST

procedure $CP_{FBST}((y^1, y^2, \alpha))$
  Obtenha $ev(H_0)$ o valor da evidência a favor de $H_0 : \delta = 1$
  if $p < \alpha$ **then**
    Retorna 1
  else
    Retorna 0
  end if
end procedure

Para testar os diferentes critérios de parada em dados reais, escolhemos algumas janelas de 15 minutos obtidos do OceanPod no Parque Estadual da Laje de Santos. As janelas possuem características diversas com respeito aos padrões de ocorrência de eventos.

A primeira é o registro do dia 30/01/2015, um sábado; o trecho foi obtido entre 02h02m56s e 02h17m56s, e captura um momento em que não se verifica nenhuma atividade além do ruído. O espectrograma deste sinal aparece na figura 3.13. Percebe-se a energia mais alta nas regiões de alta frequência, sem que se possa visualmente anotar nenhuma mudança importante no comportamento do sinal ao longo do período.

A segunda janela é o registro do dia 02/02/2015, segunda-feira, obtido a partir das 07h50m49s. Neste trecho o espectrograma exibe um evento longo, cujo início é anterior ao início do trecho, e que dura aproximadamente 10,5 minutos. A audição do trecho mostra que se trata provavelmente de uma embarcação grande (petroleiro, transatlântico, etc) passando a uma distância considerável do hidrofone.

A terceira janela foi obtida no dia 08/02/2015, domingo, a partir das 11:26m39s. Este trecho exibe intensa atividade de eventos; a audição do trecho mostra que são motores de embarcações pequenas, registrados a curta distância.

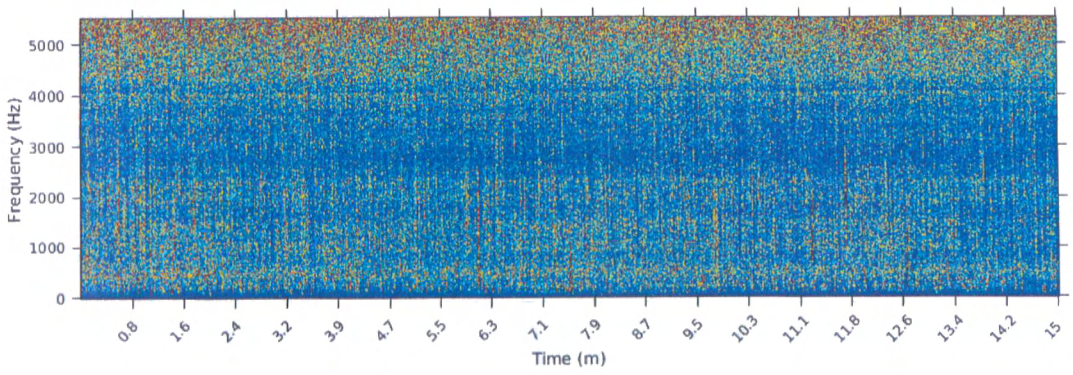


Figura 3.13: Espectrograma para o sinal do dia 30/01/2015

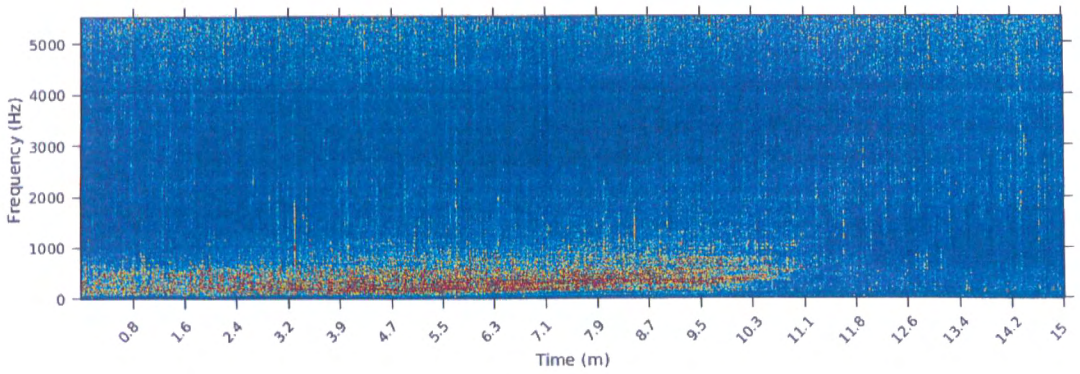


Figura 3.14: Espectrograma para o sinal do dia 02/02/2015

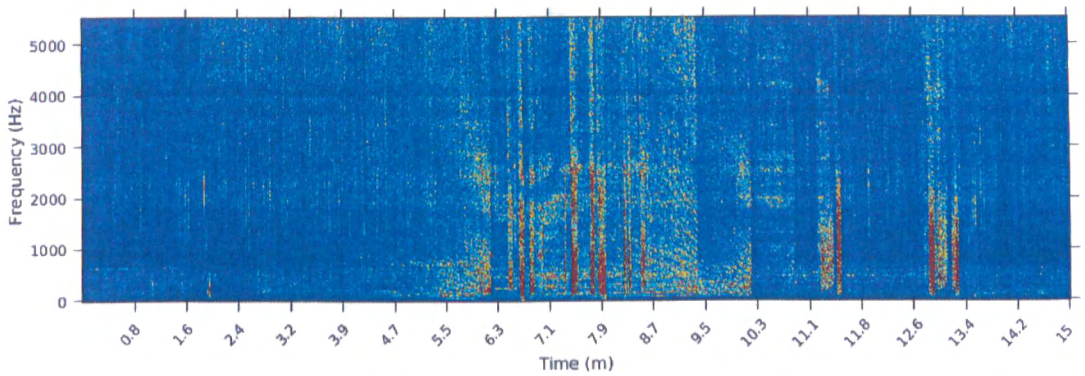


Figura 3.15: Espectrograma para o sinal do dia 08/02/2015

Testaremos os critérios de parada com diferentes parâmetros; para o critério F, utilizaremos os níveis de significância fixos $\{1e^{-20}, 1e^{-15}, 1e^{-10}\}$; para o critério da razão de verossimilhanças, adotaremos estas mesmas significâncias para o sinal de tamanho $2s$ (22.050 pontos) e calcularemos a cada teste o valor de significância adaptativo. Para o FBST, utilizaremos $\beta \in \{1, 1e^{-1}, 3e^{-5}\}$, e o valor mínimo de evidência igual a $0,01$ (equivalente à situação em que $a/99 + 100b/99 = c$).

3.3.1 Critério F

O algoritmo de segmentação com o critério de parada baseado na estatística F foi capaz de identificar trechos do sinal com diferenças de potência média. Foi necessário adotar níveis de significância extremamente baixos (da ordem de $1e^{-20}$) para evitar que mesmo os trechos de aparente silêncio fossem completamente segmentados.

Nos trechos mais curtos, porém, o algoritmo não conseguiu separar adequadamente os eventos, conforme fica claro nos espectrogramas referentes ao dia 08/02/2015.

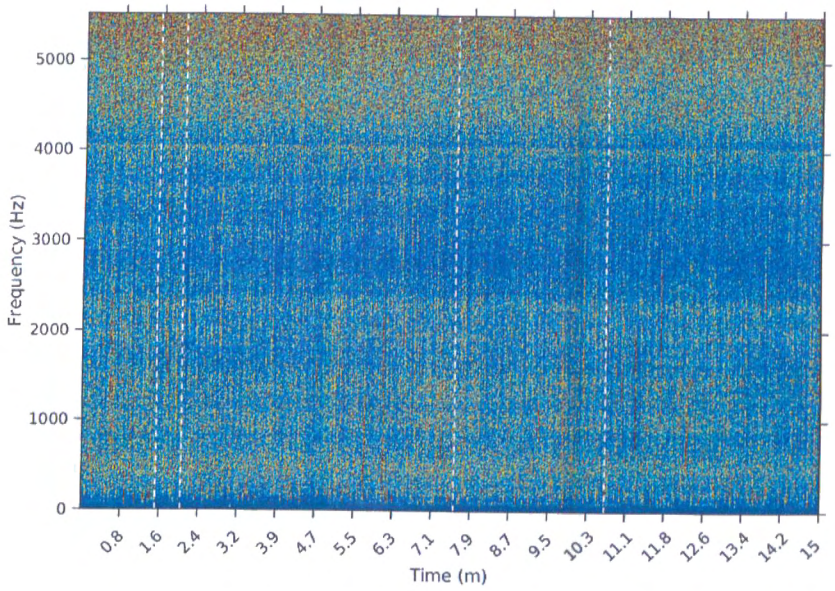


Figura 3.16: Critério F com $\alpha = 1e^{-20}$ em 30/01/2015

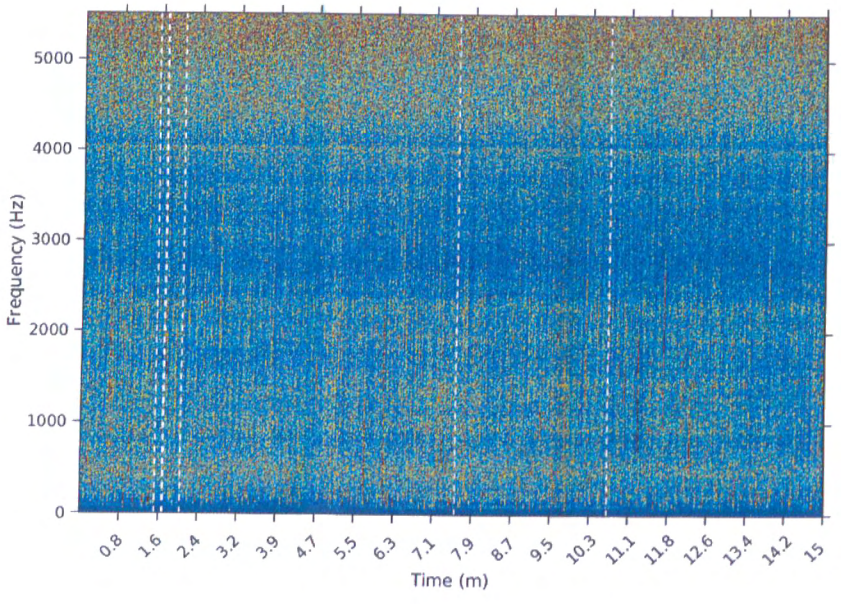


Figura 3.17: Critério F com $\alpha = 1e^{-15}$ em 30/01/2015

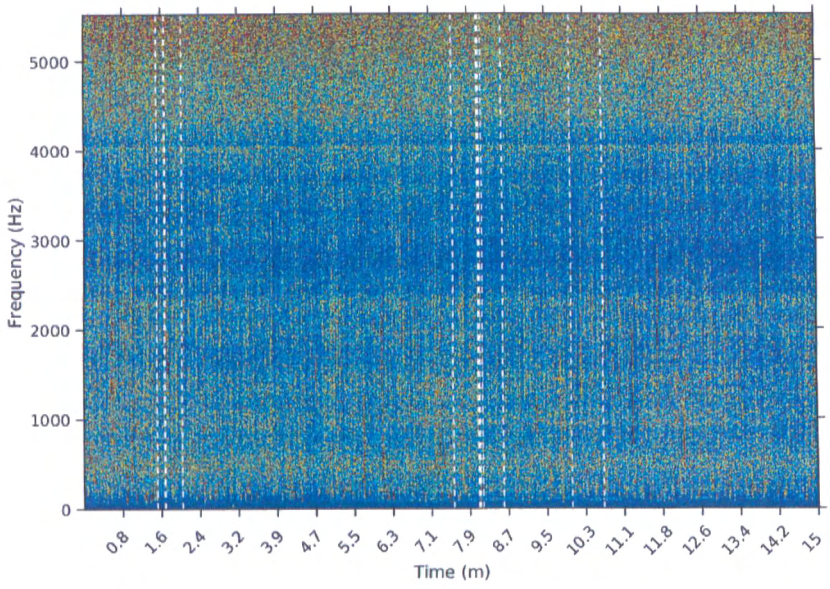


Figura 3.18: Critério F com $\alpha = 1e^{-10}$ em 30/01/2015

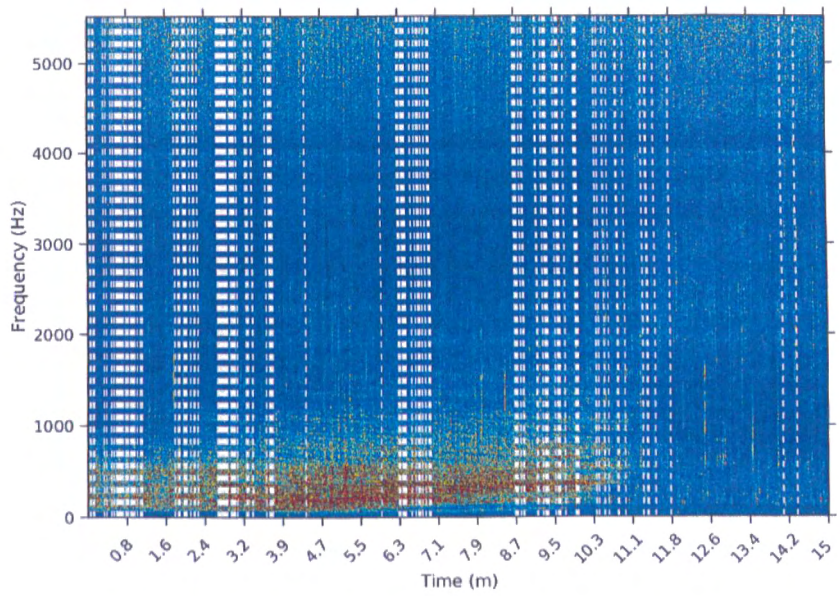


Figura 3.19: Critério F com $\alpha = 1e^{-20}$ em 02/02/2015

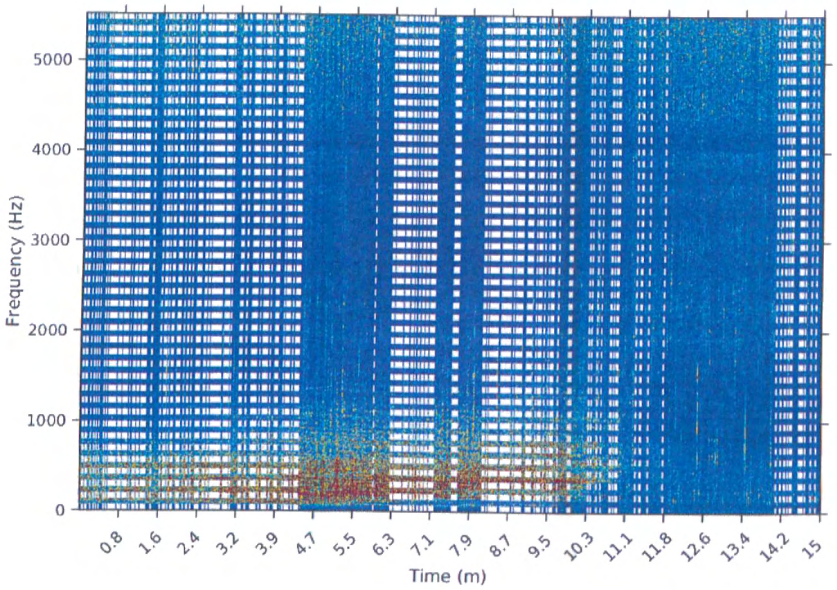


Figura 3.20: Critério F com $\alpha = 1e^{-15}$ em 02/02/2015

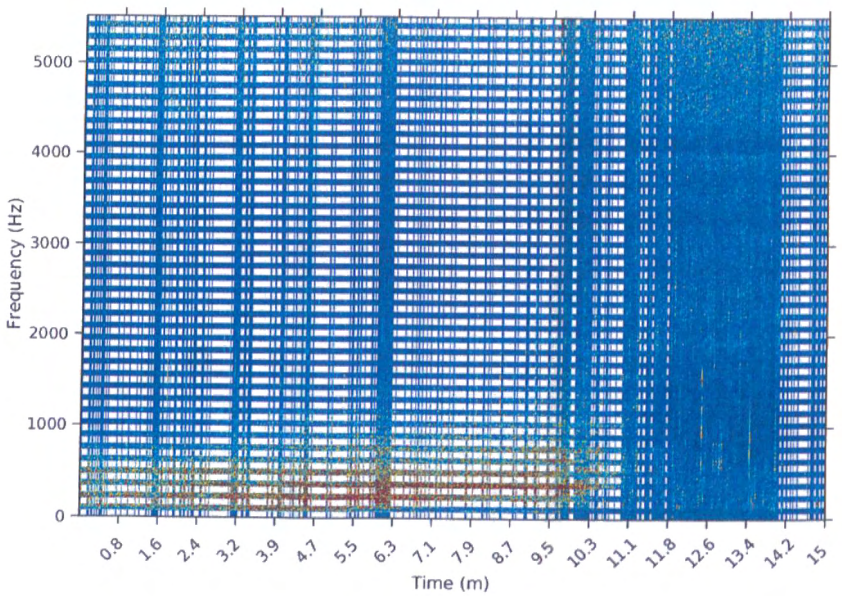


Figura 3.21: Critério F com $\alpha = 1e^{-10}$ em 02/02/2015

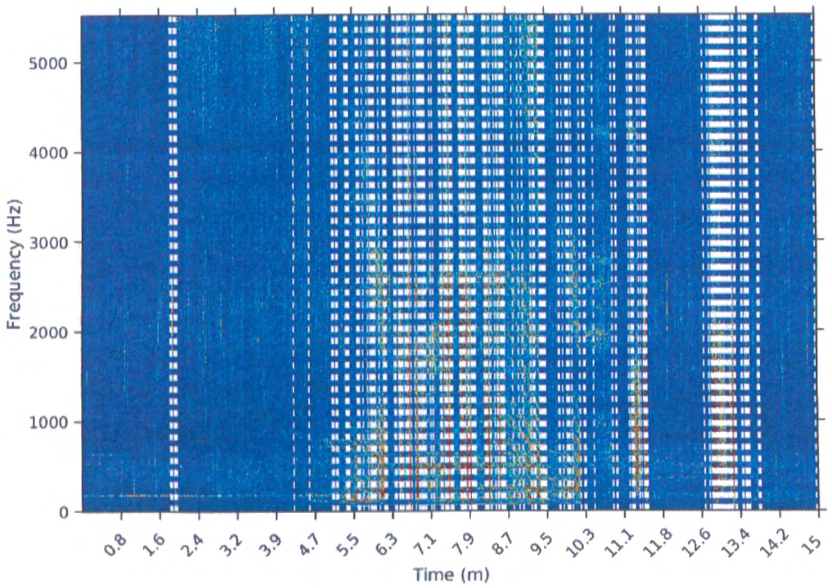


Figura 3.22: Critério F com $\alpha = 1e^{-20}$ em 08/02/2015

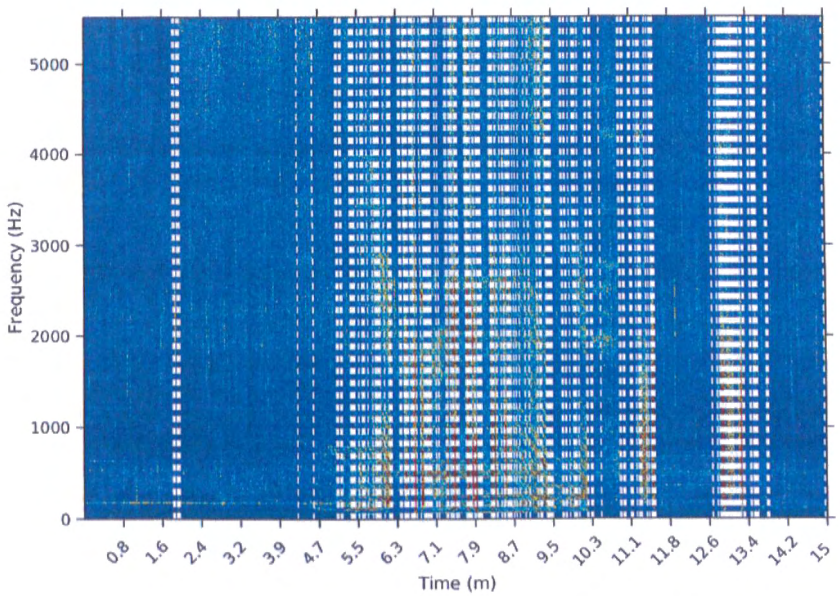


Figura 3.23: Critério F com $\alpha = 1e^{-15}$ em 08/02/2015

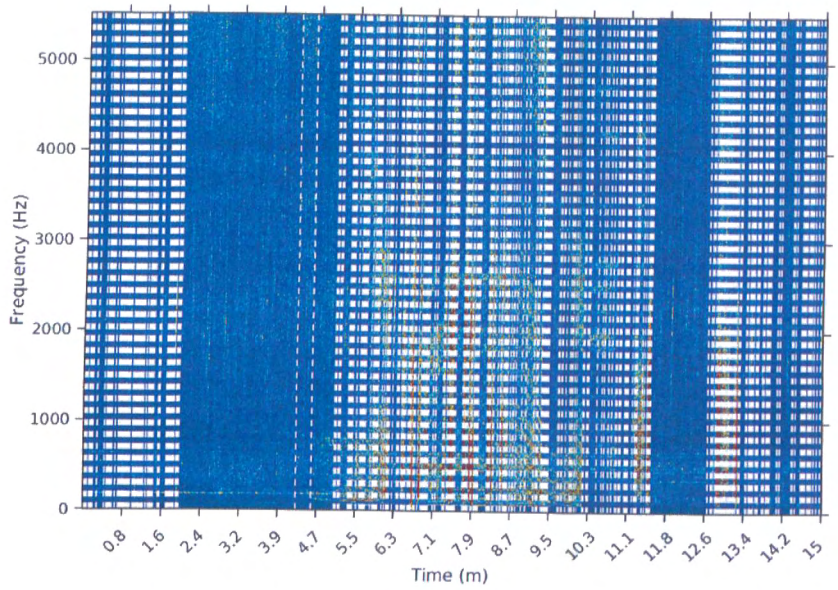


Figura 3.24: Critério F com $\alpha = 1e^{-10}$ em 08/02/2015

3.3.2 Critério da Razão de Verossimilhanças

O critério da razão de verossimilhanças com nível de significância adaptativo foi o que apresentou os piores resultados nos testes que conduzimos. A hipótese nula é rejeitada quase sempre, de modo que a segmentação falha em capturar os padrões existentes nos trechos de sinal.

Esta é uma possível indicação de que neste caso a regra do $n\ log(n)$ não decresce com velocidade suficiente.

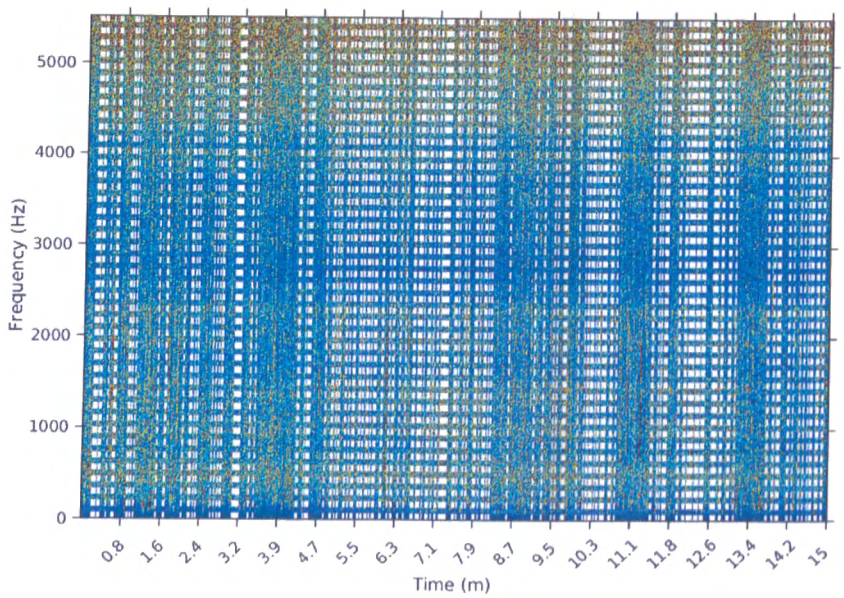


Figura 3.25: Critério RV com $\alpha = 1e^{-20}$ em 30/01/2015

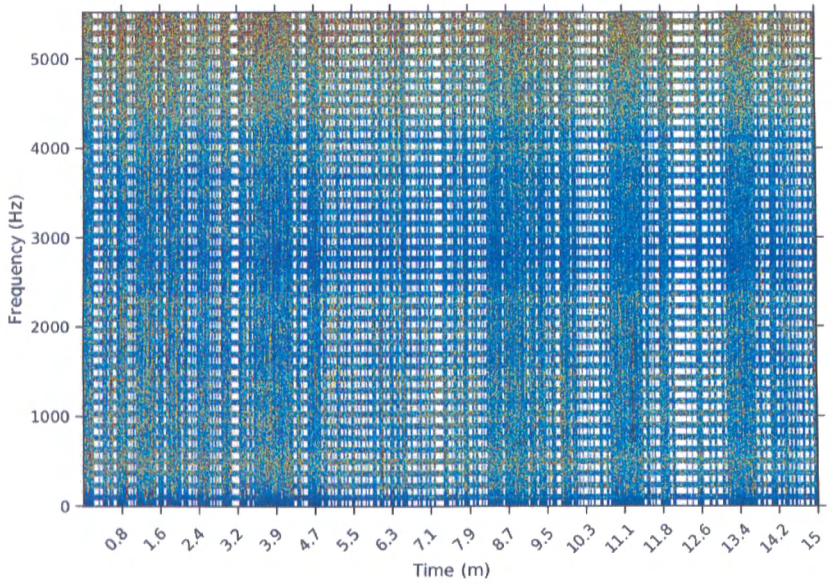


Figura 3.26: Critério RV com $\alpha = 1e^{-15}$ em 30/01/2015

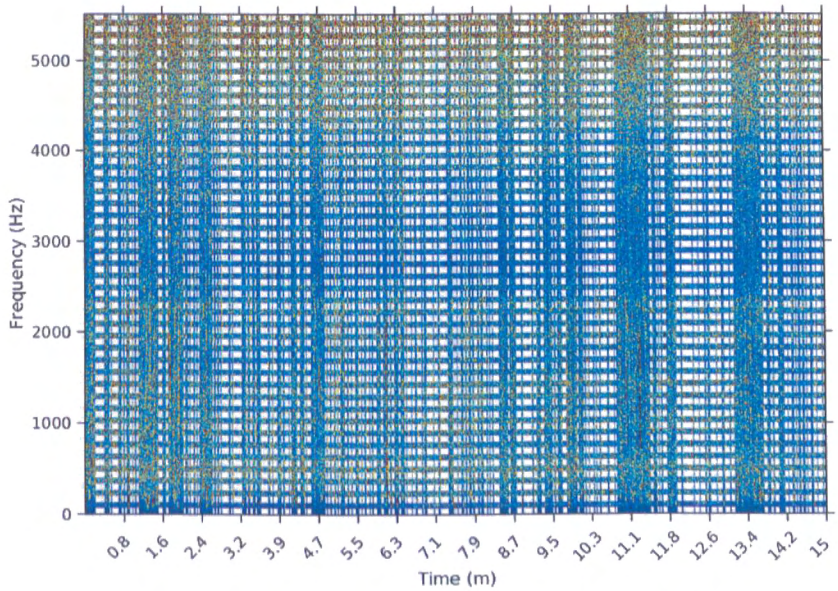


Figura 3.27: Critério RV com $\alpha = 1e^{-10}$ em 30/01/2015

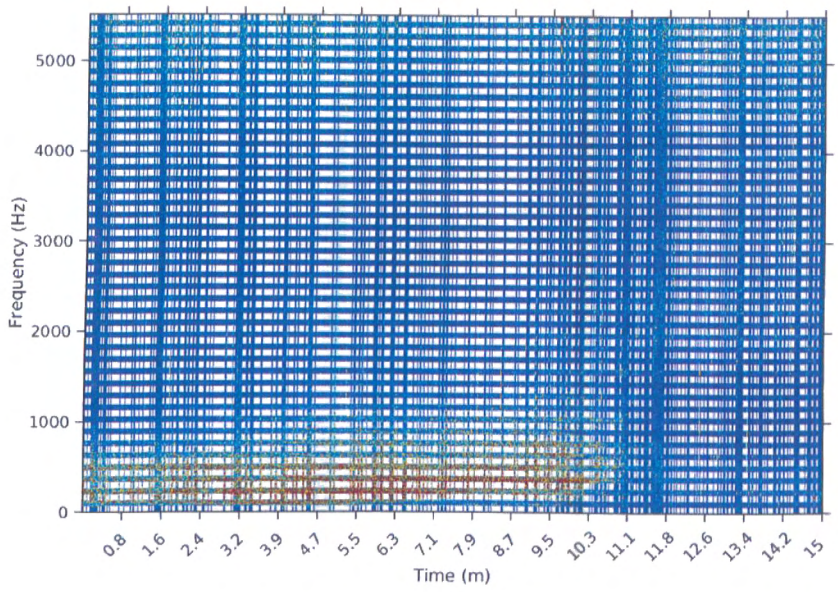


Figura 3.28: Critério RV com $\alpha = 1e^{-20}$ em 02/02/2015

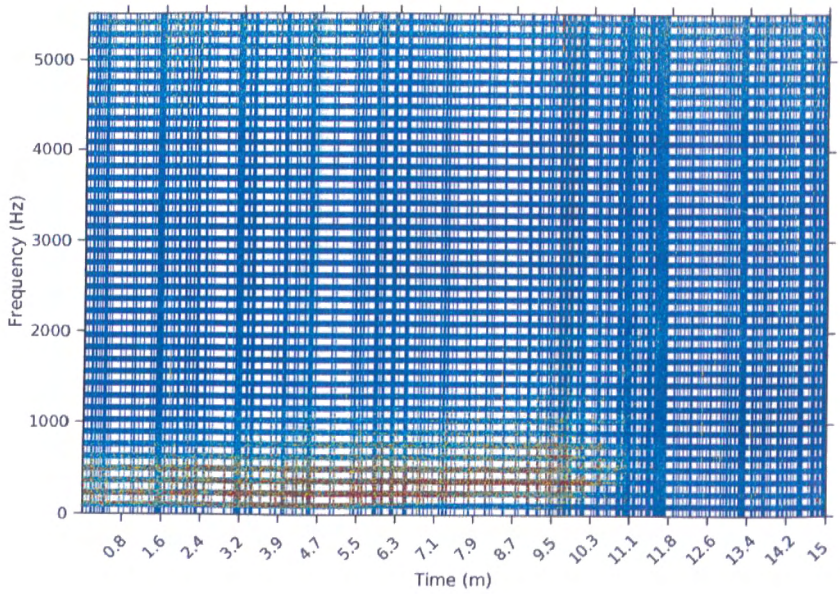


Figura 3.29: Critério RV com $\alpha = 1e^{-15}$ em 02/02/2015

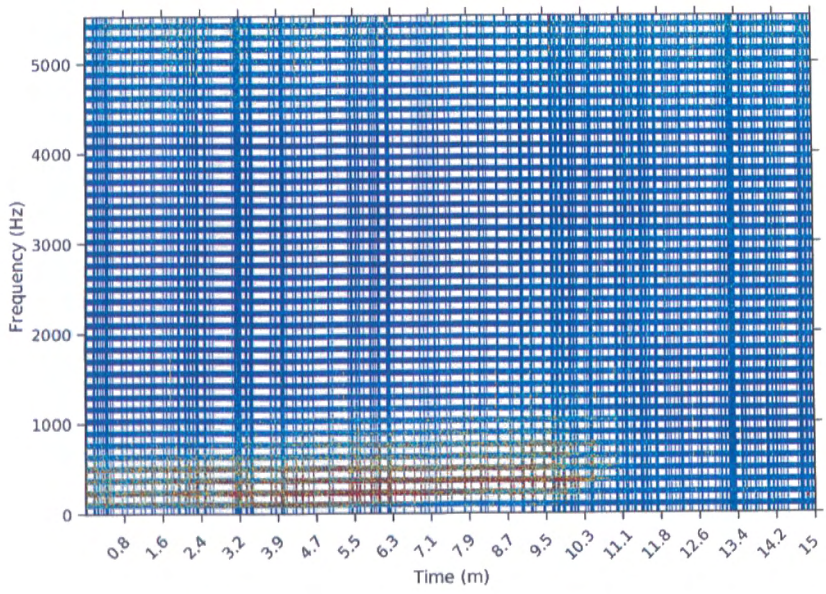


Figura 3.30: Critério RV com $\alpha = 1e^{-10}$ em 02/02/2015

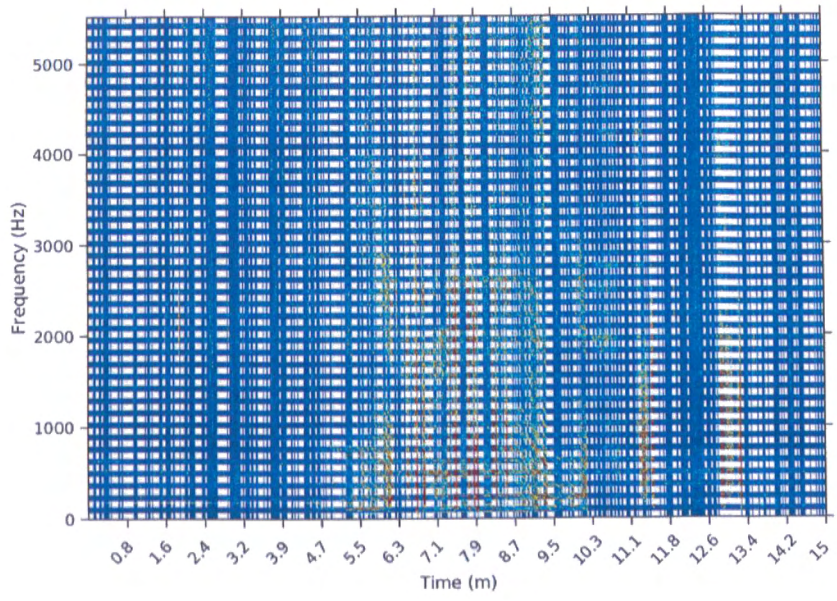


Figura 3.31: Critério RV com $\alpha = 1e^{-20}$ em 08/02/2015

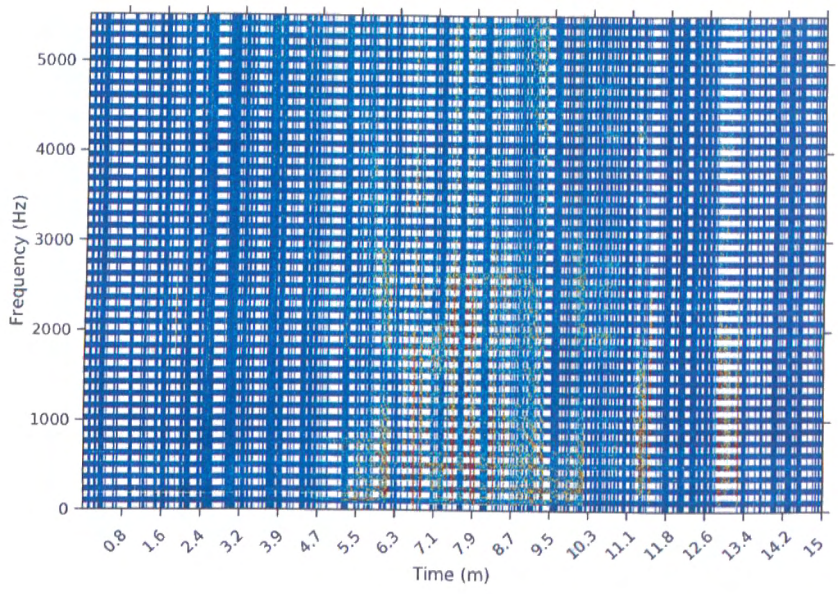


Figura 3.32: Critério RV com $\alpha = 1e^{-15}$ em 08/02/2015

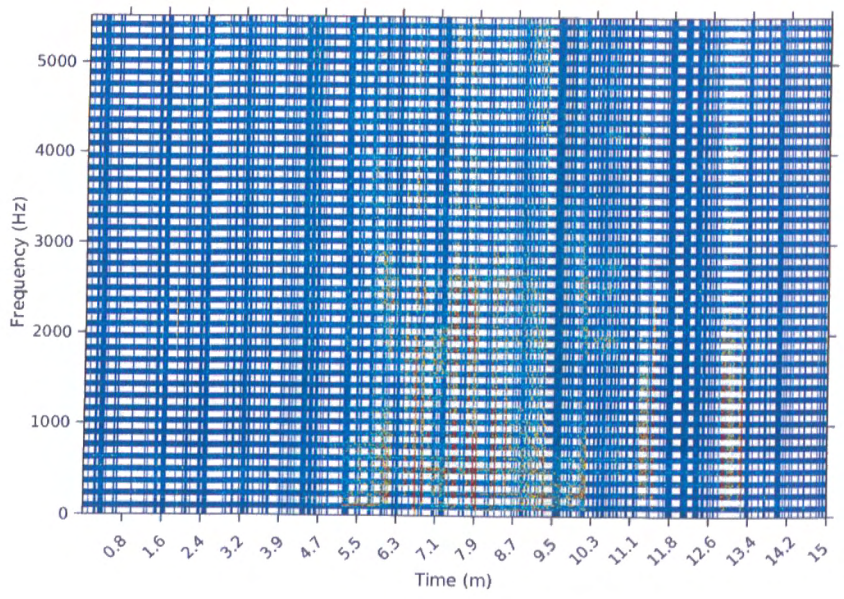


Figura 3.33: Critério RV com $\alpha = 1e^{-10}$ em 08/02/2015

3.3.3 Critério baseado no FBST

O critério baseado no FBST foi o que apresentou melhores resultados. Foi necessário utilizar um β bastante pequeno (da ordem de 10^{-5}), induzindo um pico forte em torno de $\delta = 1$ na *posteriori*; ao fazê-lo, porém, obtemos bons resultados da segmentação: o trecho do dia 30/01/2015, onde não há aparentemente a presença de nenhum evento, é mantido inteiro pelo algoritmo. No trecho do dia 08/02/2015, com diversos eventos bem separados, vemos o algoritmo capturar com precisão o início e fim da maioria dos eventos, enquanto que no trecho do dia 02/02/2015, onde há aparentemente apenas um evento de duração longa, o algoritmo captura bem o limite direito do evento, muito

embora ainda divida o evento inteiro em três trechos.

Por outro lado, por exigir uma integração numérica, o critério do FBST é o de execução mais lenta, levando em média 140 segundos para cada execução (num máquina com processador Intel i5 de 1,6 GHz, com 8 Gb de RAM), contra uma média de 20 segundos para os demais critérios.

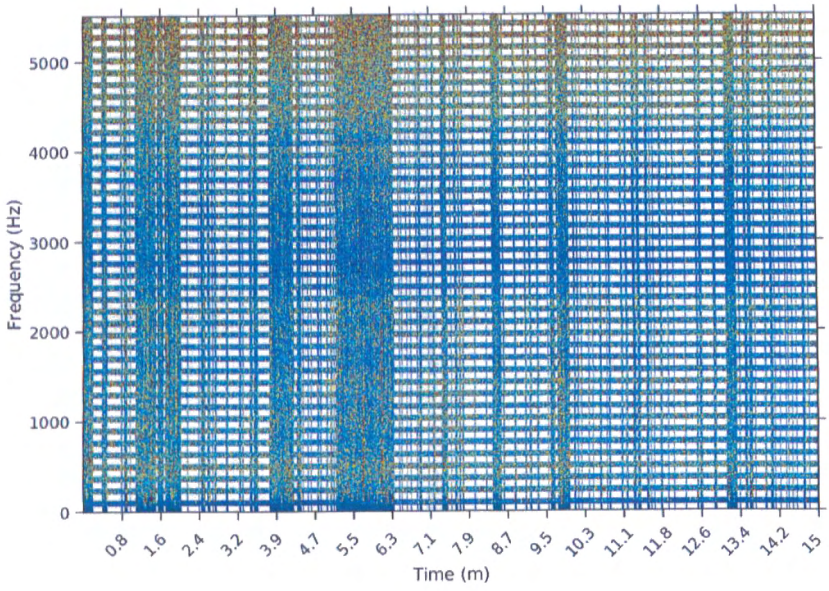


Figura 3.34: Critério FBST com $\beta = 1e^{-1}$ em 30/01/2015

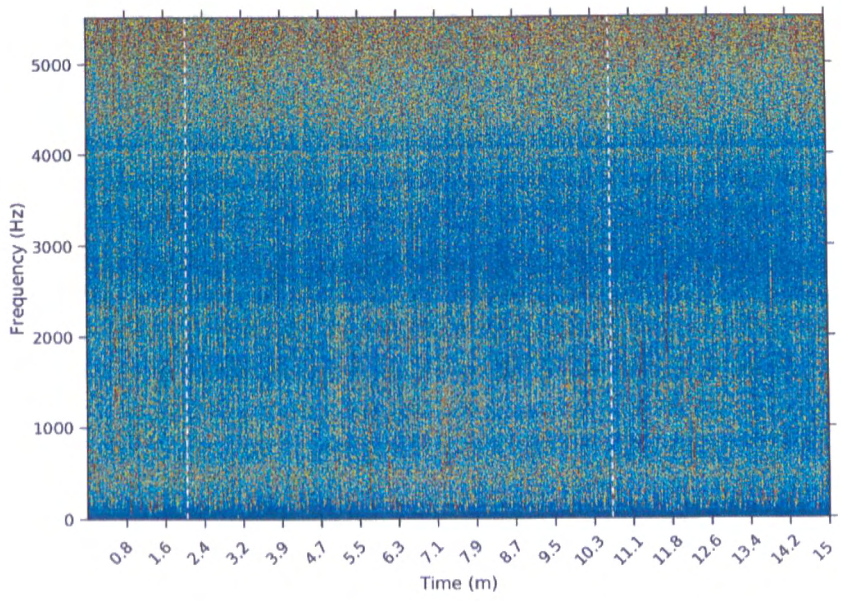


Figura 3.35: Critério FBST com $\beta = 3e^{-5}$ em 30/01/2015

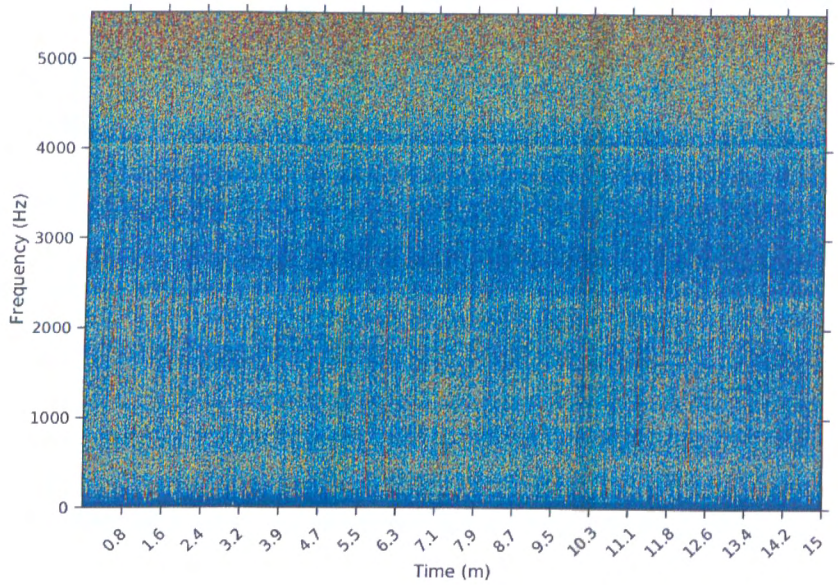


Figura 3.36: Critério FBST com $\beta = 1e^{-5}$ em 30/01/2015

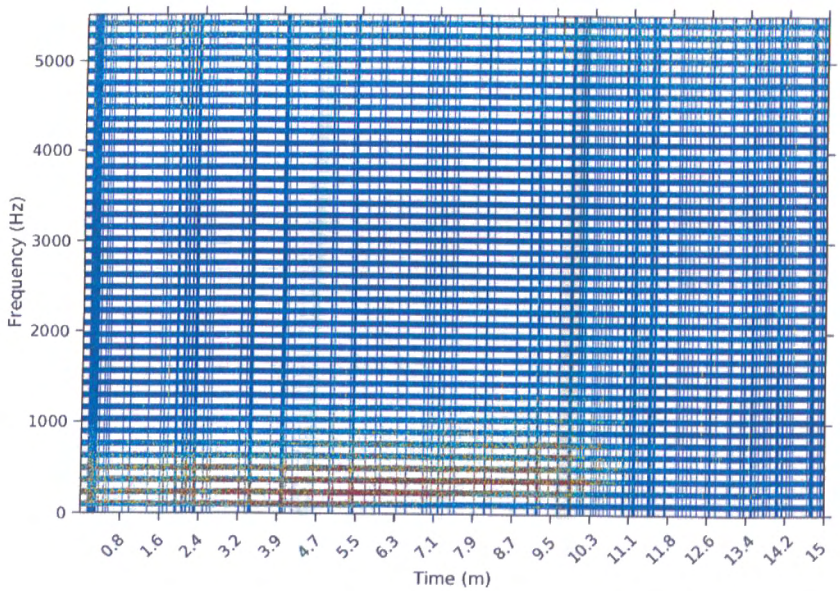


Figura 3.37: Critério FBST com $\beta = 1e^{-1}$ em 02/02/2015

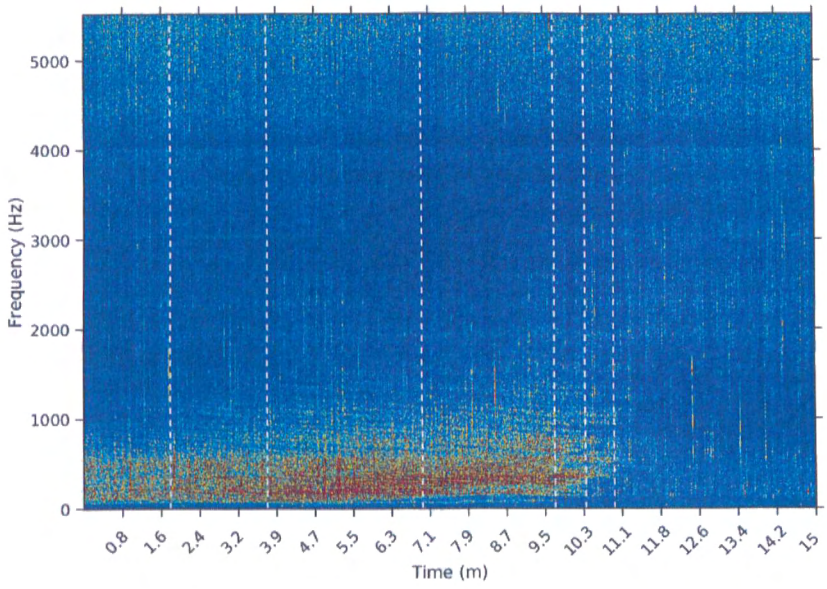


Figura 3.38: Critério FBST com $\beta = 3e^{-5}$ em 02/02/2015

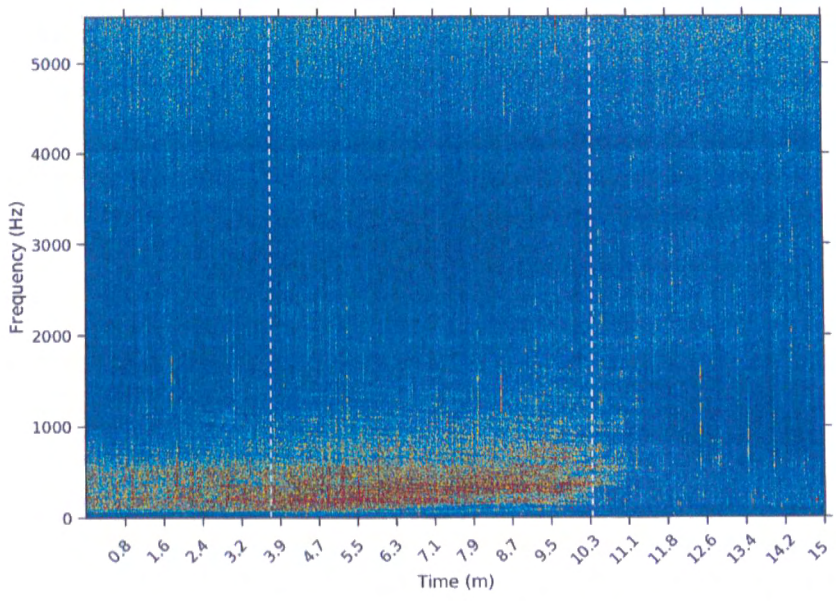


Figura 3.39: Critério FBST com $\beta = 1e^{-5}$ em 02/02/2015

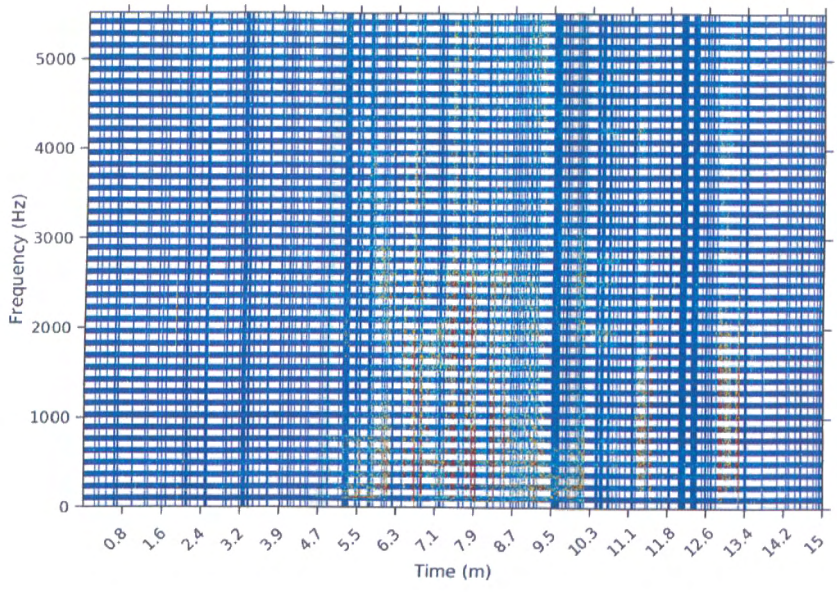


Figura 3.40: Critério FBST com $\beta = 1e^{-1}$ em 08/02/2015

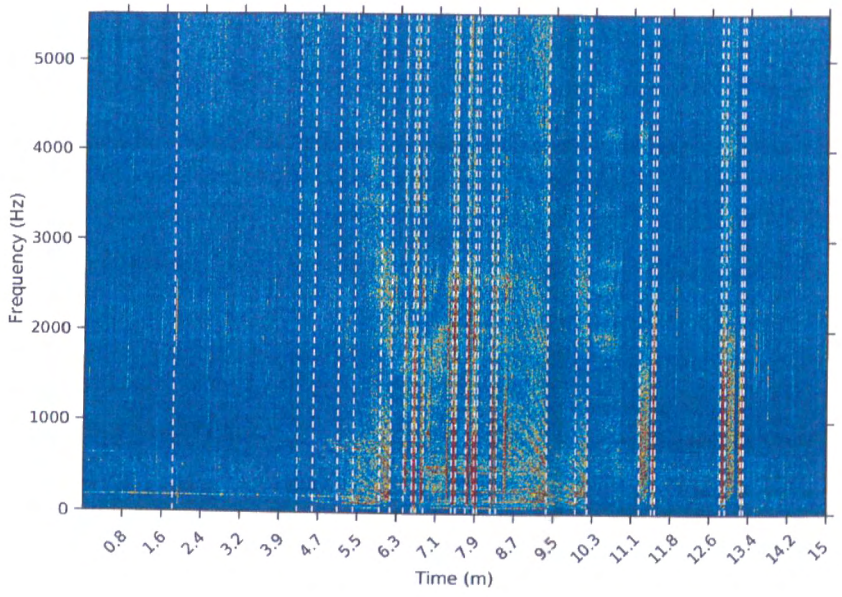


Figura 3.41: Critério FBST com $\beta = 3e^{-5}$ em 08/02/2015

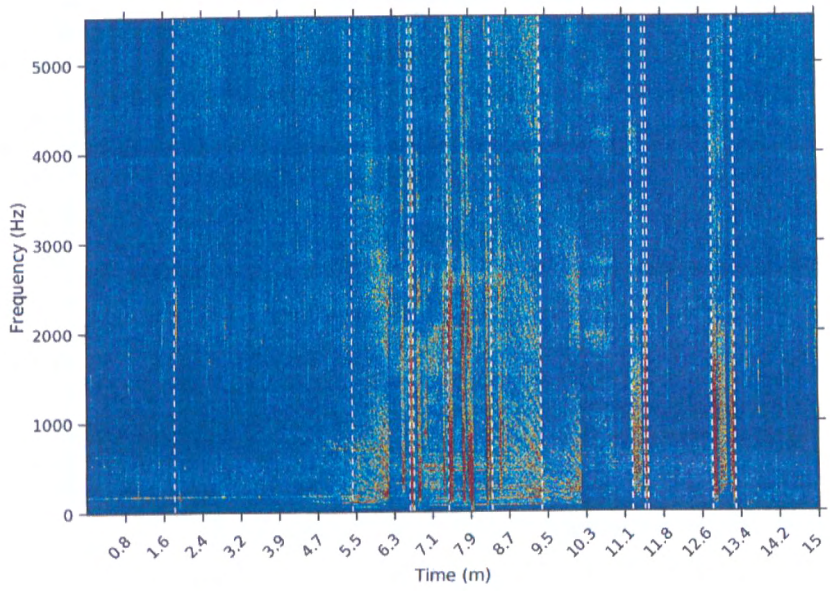


Figura 3.42: Critério FBST com $\beta = 1e^{-5}$ em 08/02/2015

3.4 Segmentação de sinais e *peak detection*

A análise da seção anterior sugere a adoção do critério do FBST para a formulação final do algoritmo de segmentação; o procedimento completo utilizando este critério está descrito no algoritmo 5.

Algorithm 5 Segmentação sequencial de sinais (SeqSeg)

procedure SEQSEG(y, *minlength*, α)
  $N = dim(y)$
  $\bar{t} = argmax_t P(t|y)$ ▷ Otimiza a posteriori para t
  if $\bar{t} <$ *minlength* **ou** $N - \bar{t} <$ *minlength* **then**
    retorna
  else
    $y^1 = \{y_i : i < \bar{t}\}$, $y^2 = \{y_i : i \geq \bar{t}\}$
    $p0 = max_{\delta=1} P(\sigma|y^1.y^2)$ ▷ Max. posteriori sob H_0
    $ev = \int_{T_{p0}} P(\sigma, \delta|y^1, y^2)$ ▷ Evidência contra H_0
    if $1 - ev < \alpha$ **then** ▷ Critério de parada
      $t_1 =$ **SeqSeg**(y^1, *minlength*, α)
      $t_2 =$ **SeqSeg**(y^2, *minlength*, α)
      retorna $[t_1, \bar{t}, \bar{t} + t_2]$
    else
      retorna
    end if
  end if
end procedure

Denominamos este algoritmo a partir de agora como *SeqSeg* (de ***sequential segmentation***).

Para avaliar a performance do algoritmo, comparamos seus resultados com um algoritmo de *peak detection* (detecção de picos, PD). Em particular, utilizamos o algoritmo de Palshikar (2009). Este algoritmo define um pico no sinal utilizando tanto propriedades locais quanto globais. As

propriedades locais são obtidas a partir de funções calculadas em janelas de tamanho k, centradas sucessivamente em cada coordenada do sinal. Os valores destas funções locais são então comparados com os mesmo valores calculados para todas as diferentes janelas; caso uma determinada janela apresente um valor distante da média por mais do que um certo número de desvios-padrão, o centro desta janela é aceito como um pico.

O algoritmo de Palshikar exige a escolha de dois parâmetros: o tamanho k das janelas, e o valor de corte de h desvios-padrão. São definidas diferentes funções para refletir as propriedades locais das coordenadas individuais. A primeira, S_1, calcula para cada ponto y_i do sinal a diferença máxima entre y_i e seus vizinhos à direita $\{y_j : j > i, |i - j| < k/2\}$ e à esquerda $\{y_j : j < i, |i - j| < k/2\}$, obtendo por fim a média desses dois máximos. A segunda, S_2, calcula a diferença média entre o centro da janela y_i e seus vizinhos à direita e à esquerda, calculando por fim a média dessas médias. A terceira, S_3, calcula a diferença entre o centro y_i e a média dos vizinhos à direita e à esquerda, calculando novamente a média entre esses dois valores. Para mais detalhes do método de Palshikar, e uma revisão de métodos de detecção de picos, referimos o leitor interessado ao artigo original Palshikar (2009).

Em primeiro lugar, comparamos os dois algoritmos num sinal simulado $y \in \Re^2 0000$ com $y_i \sim \mathcal{N}(0, \sigma_i^2)$, com a variância dada por

$$\sigma_i^2 = \begin{cases} 1 & \text{if } i \leq 5000 \\ 1.1 & \text{if } 5000 < i \leq 10000 \\ 1 & \text{if } 10000 < i \leq 12000 \\ 1.5 & \text{if } 12000 < i \leq 15000 \\ 1 & \text{if } i > 15000 \end{cases} \tag{3.15}$$

Para o algoritmo SeqSeg, adotamos $\beta \in \{0, 01, 1\}$ e $\alpha \in \{0, 01, 0, 1\}$; para o algoritmo de Palshikar, aplicamos cada uma das funções S_1. S_2 e S_3 em oito combinações diferentes para h e k. Os resultados aparecem na tabela 3.2.

Algoritmo	Parâmetros	Tempo total (s)	# de segmentos	Primeiro ponto de corte	Último ponto de corte
SeqSeg	$\beta = 1, \alpha = 0,01$	9,8912	5	4.990	15.001
SeqSeg	$\beta = 0,01, \alpha = 0,01$	11,8567	5	4.990	15.001
SeqSeg	$\beta = 1, \alpha = 0.1$	12,7605	5	4.990	15.001
SeqSeg	$\beta = 0.01, \alpha = 0.1$	11,5746	5	4.990	15.001
Palshikar S_1	$h = 3, k = 100$	0,3116	31	4.778	14.852
Palshikar S_2	$h = 3, k = 100$	0,3303	13	5.039	14.470
Palshikar S_3	$h = 3, k = 100$	0,3443	9	1.608	14.249
Palshikar S_1	$h = 3, k = 500$	0,4264	5	5.039	14.179
Palshikar S_2	$h = 3, k = 500$	1,4768	55	1.608	19.892
Palshikar S_3	$h = 3, k = 500$	1,3090	25	964	19.892
Palshikar S_1	$h = 3, k = 1000$	1,0846	16	964	19.892
Palshikar S_2	$h = 3, k = 1000$	1,2406	10	964	19.892
Palshikar S_3	$h = 3, k = 1000$	1,4050	55	1.608	19.892
Palshikar S_1	$h = 3, k = 2000$	0,9451	25	964	19.892
Palshikar S_2	$h = 3, k = 2000$	0,9889	16	964	19.892
Palshikar S_3	$h = 3, k = 2000$	1,1212	10	964	19.892
Palshikar S_1	$h = 4, k = 100$	0,1718	5	13.171	14.311
Palshikar S_2	$h = 4, k = 100$	0,2303	3	13.171	14.311
Palshikar S_3	$h = 4, k = 100$	0,3023	4	12.606	14.727
Palshikar S_1	$h = 4, k = 500$	0,4439	3	12.606	14.727
Palshikar S_2	$h = 4, k = 500$	0,8924	16	1.608	14.923
Palshikar S_3	$h = 4, k = 500$	0,9922	10	1.608	14.923
Palshikar S_1	$h = 4, k = 1000$	1,0793	9	1.608	14.727
Palshikar S_2	$h = 4, k = 1000$	1,2486	6	1.608	14.311
Palshikar S_3	$h = 4, k = 1000$	0,8439	16	1.608	14.923
Palshikar S_1	$h = 4, k = 2000$	0,9167	10	1.608	14.923
Palshikar S_2	$h = 4, k = 2000$	1,0030	9	1.608	14.727
Palshikar S_3	$h = 4, k = 2000$	1,1367	6	1.608	14.311

Tabela 3.2: Comparação dos algoritmos em um sinal simulado

Os resultados no sinal simulado mostram em primeiro lugar que o algoritmo SeqSeg é mais robusto à escolha dos parâmetros do que o algoritmo de Palshikar. Todas as combinações de parâmetros do algoritmo SeqSeg levaram ao mesmo resultado, segmentando corretamente o sinal em cinco segmentos, o primeiro começando em torno de $i = 5000$ e o último em torno de $i = 15000$. O algoritmo de detecção de picos, por outro lado, é mais sensível à escolha dos parâmetros, obtendo o melhor resultado para a função S_1 com $h = 3$ e $k = 500$. Por outro lado, o algoritmo SeqSeg é muito menos eficiente do ponto de vista computacional.

Comparando agora os dois algoritmos utilizando algumas amostras do OceanPod, encontramos os resultados da tabela 3.3. As amostras utilizadas foram as mesmas da seção anterior (comparação dos critérios de parada).

Algoritmo	Amostra	Tempo total (s)	Segmentos	Primeiro corte (minutos)	Último corte (minutos)
SeqSeg, $\beta = 1e^{-5}$	8 fevereiro 2015	245,41	13	1,80	13,32
SeqSeg, $\beta = 3e^{-5}$	8 fevereiro 2015	174,13	28	1,80	13,32
Palshikar $h = 3$	8 fevereiro 2015	113,33	56	0,51	13,30
Palshikar $h = 5$	8 fevereiro 2015	112,23	29	6,65	13,28
SeqSeg, $\beta = 1e^{-5}$	30 janeiro 2015	149,61	0	2,02	10,63
SeqSeg, $\beta = 3e^{-5}$	30 janeiro 2015	44,83	3	-	-
Palshikar $h = 3$	30 janeiro 2015	88,27	130	0,00	14,91
Palshikar $h = 5$	30 janeiro 2015	87,30	27	0,30	14,21
SeqSeg, $\beta = 1e^{-5}$	2 fevereiro 2015	101,45	3	1,77	10,90
SeqSeg, $\beta = 3e^{-5}$	2 fevereiro 2015	94,24	7	3,75	10,32
Palshikar $h = 3$	2 fevereiro 2015	89,07	114	0,00	14,83
Palshikar $h = 5$	2 fevereiro 2015	91,15	22	0,69	14,20

Tabela 3.3: Comparação dos algoritmos em sinais reais; ver o texto para detalhes

Nas amostras do OceanPod, ambos algoritmos exibiram desempenho computacional impraticável, levando cerca de 10 minutos para segmentar cada sinal. Por isso, foi necessário aplicar uma limitação à *resolução* dos métodos, da seguinte maneira: para o algoritmo SeqSeg, avaliamos a posteriori do ponto de corte $\bar{t}$ apenas num conjunto da forma $\{i = k \cdot t_{res}\}$, onde t_{res} foi fixado em 11025; isto significa que os pontos de corte estimados devem pertencer a este conjunto. Para o algoritmo de Palshikar, centramos as janelas em y_i para o mesmo conjunto $\{i = k \cdot t_{res}\}$, mas com $t_{res} = 10$. Esta escolha foi suficiente para reduzir o tempo de computação do método de Palshikar, sem prejudicar demais seus resultados.

Mesmo com essas limitações na resolução, o algoritmo sequencial apresentou resultado superior ao método de *peak detection*, em particular nas amostras com poucos eventos. O algoritmo de PD tende a sobresegmentar o sinal, pois em regiões de alta potência a ocorrência de extremos (picos) é muito mais frequente. Para ilustrar melhor a segmentação obtida por cada método, reapresentamos nas figuras 3.43 a 3.46 os espectrogramas do sinal de 8 de fevereiro (diversos eventos de curta duração), acrescentando linhas verticais nos pontos de corte de cada algoritmo.

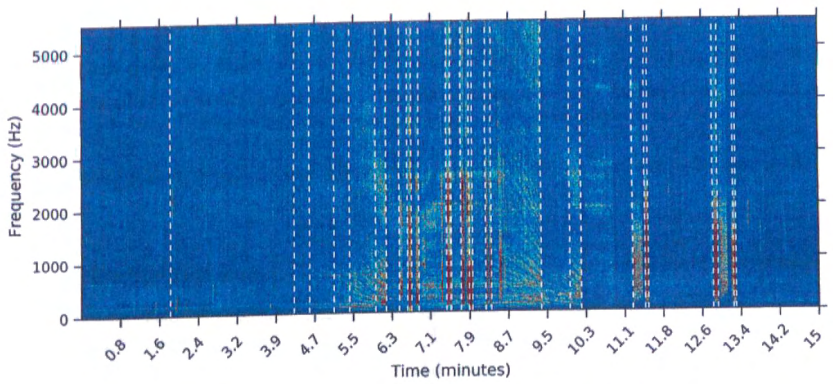


Figura 3.43: Resultado do algoritmo SeqSeg, $\beta = 3e^{-5}$

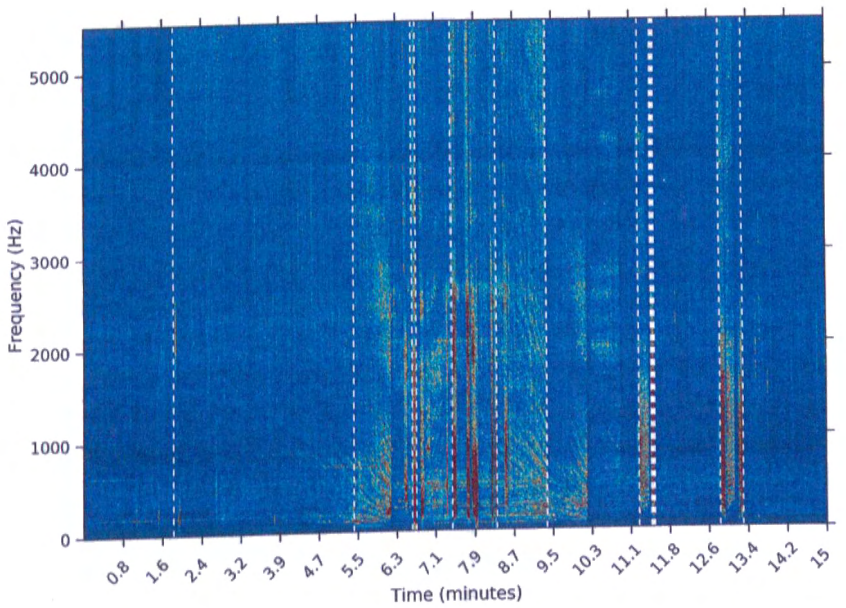


Figura 3.44: Resultado do algoritmo SeqSeg, $\beta = 1e^{-5}$

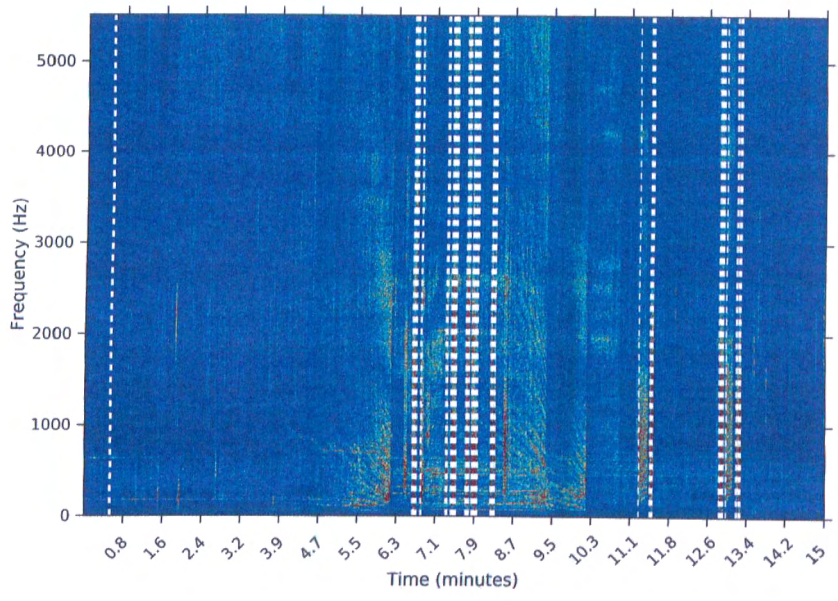


Figura 3.45: Resultado do algoritmo de Palshikar, $h = 3$

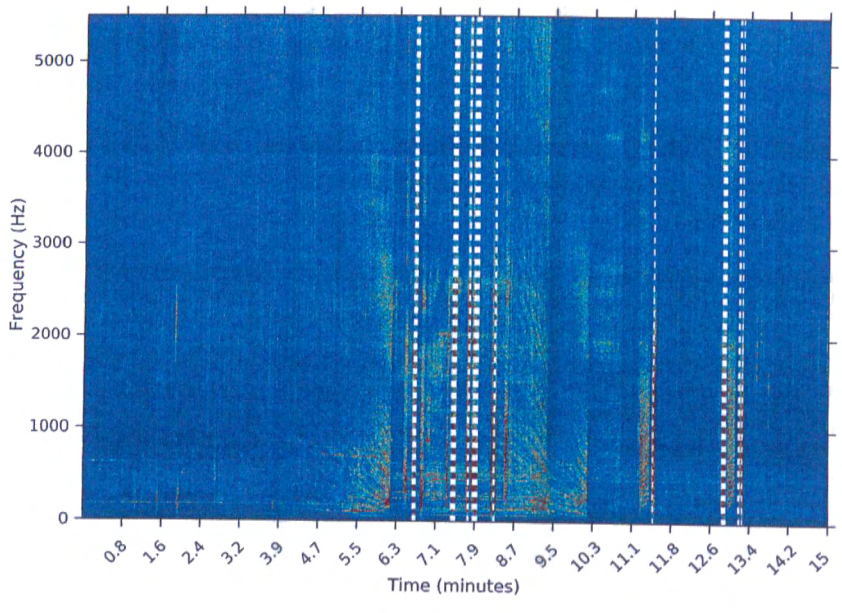


Figura 3.46: Resultado do algoritmo de Palshikar, $h = 5$

A análise dos segmentos obtidos em cada caso mostra que o algoritmo SeqSeg tem resultados mais precisos e também mais parcimoniosos do que o algoritmo PD.

Resta, contudo, o problema do desempenho numérico do algoritmo SeqSeg; mesmo com uma resolução temporal baixa na estimativa dos pontos de corte, o tempo de execução foi da ordem de 2 minutos. Isto implica numa razão de 1 : 7, aproximadamente, entre o tempo de processamento e a duração do sinal. Para a análise de sinais de longa duração, como os obtidos do OceanPod em campo, esta razão é muito baixa: a segmentação de um mês de sinal, por exemplo, levaria aproximadamente 4 dias.

Para tratar dessa questão, na próxima seção apresentamos uma implementação otimizada do algoritmo sequencial, utilizando o pacote *Cython* da linguagem *Python*.

3.5 Implementação otimizada com Python + Cython

Para permitir a aplicabilidade do algoritmo de segmentação na análise de sinais longos, implementamos uma versão otimizada do método utilizando a linguagem *python*. Os motivos para escolha dessa linguagem são, em primeiro lugar, a facilidade de implementação, principalmente das interfaces de entrada e saída de dados. Em segundo lugar, o fato de que esta linguagem tem ganhado adeptos na computação científica, principalmente a partir da disponibilização de pacotes como *numpy* e *scipy*, e da biblioteca *tensorflow* implementada pelo Google; e finalmente pela possibilidade de utilizar o pacote *cython* (http://cython.org).

Este pacote é um compilador que traduz código python (adaptado com tipagem forte das variáveis) para código C otimizado. A utilização deste pacote permite ganhos importantes de performance, além de facilitar a utilização de outras bibliotecas escritas em C (Seljebotn (2009)). Em particular, utilizamos os métodos numéricos implementados em *math.h* para as funções gama, exponencial e logaritmo, e a biblioteca *GSL* (*GNU Scientific Library*) para a geração de números aleatórios.

Além disso, o pacote Cython é compatível com a biblioteca *OpenMP* (http://www.openmp.org) para computação em paralelo; utilizando todas essas melhorias, como veremos a seguir, foi possível reduzir o tempo médio necessário para segmentação de um sinal de 15 minutos de duração para menos de 1 segundo.

Todo o código descrito nesta seção está disponível abertamente em https://github.com/paulohubert/bayeseg, bem como as três amostras do OceanPod utilizadas em diversos pontos deste trabalho.

3.5.1 Estimação do ponto de corte

Dado um sinal discretizado $y \in \Re^N$, o primeiro passo do algoritmo consiste em estimar o ponto de corte $\bar{t}$ de máxima posteriori. A posteriori de $\bar{t}$ é dada por

$$\mathcal{L}(\bar{t}|y) = \int_0^\infty \mathcal{L}(\bar{t}, \sigma_0^2|y)\pi(\sigma_0)d\sigma_0 \tag{3.16}$$

$$\propto \left(\sum_{i=\bar{t}+1}^{N} y_i^2\right)^{-\frac{(N-\bar{t}-6)}{2}} \Gamma\left(\frac{N-\bar{t}-2}{2}\right) \left(\sum_{i=1}^{\bar{t}} y_i^2\right)^{-\frac{(\bar{t}+6)}{2}} \Gamma\left(\frac{\bar{t}+6}{2}\right) \tag{3.17}$$

Esta função depende de y somente através das somas acumuladas $\sum_i y_i^2$. Uma vez que ao longo de uma rodada do algoritmo diversos pontos de corte serão estimados para o mesmo sinal, um primeiro método para acelerar o algoritmo é armazenar inicialmente as somas $\sum_{i=1}^{T} y_i^2$ para todos os valores de T, evitando que essa soma seja recalculada a cada iteração.

A abordagem tradicional na estimação por MAP (máximo a posteriori) e MV (máxima verossimilhança) consiste em otimizar o logaritmo da respectiva função (densidade posteriori ou verossimilhança). Neste caso, a avaliação de 3.16 depende da função log-gama. Utilizamos a implementação da biblioteca *math.h*, em C, para obter esta função.

Por fim, como a otimização é por inspeção (uma vez que a posteriori é discreta com suporte limitado), este passo pode ser acelerado via paralelização. Para isto, utilizamos o arcabouço do OpenMP, que é diretamente suportado pela biblioteca Cython, da seguinte maneira: o suporte de 3.16 é dividido em partes iguais (o número de partes é decidido em tempo de execução pela biblioteca), e a avaliação da posteriori nos pontos correspondentes a cada parte é distribuída pelos processadores da máquina.

Discutimos na seção anterior que é possível definir um parâmetro de resolução temporal para a estimação do ponto de corte, reduzindo o tempo de execução. Na tabela 3.4 avaliamos o impacto de

Tabela 3.4: Efeito da resolução temporal na avaliação do ponto de corte

Tamanho do sinal	Resolução	Tempo (s)
10000	1	0,01982
10000	10	0,009003
10000	100	0,008039
10000	1000	0,004694
100000	1	0,05072
100000	10	0,007656
100000	100	0,002263
100000	1000	0,001422
1000000	1	0,6214
1000000	10	0,1034
1000000	100	0,02704
1000000	1000	0,02186

diferentes valores da resolução no tempo de computação gasto para obter $\bar{t}$. Quanto menor o valor da resolução, mais cara é a obtenção de $\bar{t}$, e mais preciso se torna o algoritmo.

Em máxima resolução, mesmo para um sinal com $1MM$ de pontos, o tempo para obtenção de $\bar{t}$ é da ordem de 0.6 segundos. Sendo assim, nesta versão do código é possível utilizar a máxima resolução e ainda assim obter tempos razoáveis de execução.

3.5.2 Cálculo da evidência

Após a obtenção do ponto de corte, o próximo passo é calcular a medida de evidência do FBST para a hipótese $H_0 : \delta = 1$ de igualdade das variâncias.

Este cálculo por sua vez é dividido em dois procedimentos: primeiro, obtemos o máximo da posteriori sob H_0, ou seja, o máximo da posteriori para σ_0^2 assumindo $\delta = 1$. Este passo pode ser resolvido analiticamente, o que resulta no seguinte ponto de máximo:

$$s_0 = argmax_s P(1, s|y^1, y^2) = \frac{\sum_{i=1}^{N} y_i^2}{N+1} \tag{3.18}$$

Calculamos então $p_0 = P(1, s_0|y^1, y^2)$, e o próximo passo é avaliar a integral da posteriori irrestrita no conjunto surpresa $\{(s, \delta) \in \Theta : P(\delta, s) > p_0\}$. Esta integral não pode ser obtida analiticamente, o que nos leva a escolher um método numérico para sua avaliação.

Nesta versão do algoritmo utilizamos o método adaptativo de Haario *et al.* (2001). Este método é baseado no algoritmo de Metropolis-Hastings (MH) com uma distribuição Gaussiana para geração de pontos candidatos. A matriz de covariância desta Gaussiana é ajustada iterativamente durante o período de queima (*burn-in*) para aumentar a probabilidade de aceitação da cadeia sem prejudicar as propriedades ergódicas do método.

A simulação é feita em três fases: na primeira, utilizamos uma distribuição fixa para gerar candidatos para σ^2 e δ. Cada ponto é aceito ou rejeitado de acordo com a probabilidade usual de MH com passeios aleatórios. Nesta fase, iniciamos a cadeia usando distribuições Gaussianas com médias dadas por

$$\tilde{s} = \frac{\sum_{i=1}^{N_1} y_{1,i}^2}{N_1 - 1} \tag{3.19}$$

$$\tilde{d} = \frac{\sum_{i=1}^{N_2} y_{2,i}^2}{N_2 - 1} \cdot \frac{N_1 - 1}{\sum_{i=1}^{N_1} y_{1,i}^2} \tag{3.20}$$

onde $y_{1,\cdot}$ e N_1 ($y_{2,\cdot}$ e N_2) são os pontos e o tamanho do primeiro (segundo) segmento. A variância dessas Gaussianas iniciais é fixada em $\tilde{s}/3$ e $\tilde{d}/3$.

Esta primeira etapa tem como objetivo gerar as primeiras estimativas para a matriz de covariância da distribuição candidata. Após t_0 rodadas, estimamos σ_d^2, σ_s^2 e $cov(d,s)$ da seguinte maneira:

$$\sigma_d^2 = \frac{\sum_{i=1}^{t_0}(d_i - \bar{d})^2}{t_0 - 1} \tag{3.21}$$

$$\sigma_s^2 = \frac{\sum_{i=1}^{t_0}(s_i - \bar{s})^2}{t_0 - 1} \tag{3.22}$$

$$cov(d,s) = \frac{\sum_{i=1}^{t_0}(d_i - \bar{d})(s_i - \bar{s})}{t_0 - 1} \tag{3.23}$$

Aqui s_i e d_i são as amostras dos dois parâmetros do modelo (o desvio-padrão do sinal σ_0 e o quociente entre as variâncias δ).

Após a obtenção dessas estimativas, a fase de queima começa. Nesta fase, as amostras geradas serão descartadas; os objetivos desta etapa são ajustar a covariância da distribuição candidata, e aproximar a cadeia de sua distribuição estacionária.

São gerados novos pontos para os parâmetros, de acordo com a distribuição $(d_{cand}, s_{cand}) \sim \mathcal{N}\left((d_{cur}, s_{cur}), \Sigma_t\right)$, e a matriz de covariância Σ_t é atualizada de acordo com a fórmula recursiva (Haario *et al.* (2001))

$$\Sigma_{t+1} = \frac{t-1}{t}\Sigma_t + \frac{s_d}{t}\left(t\bar{X}_{t-1}\bar{X}_{t-1}^T - (t+1)\bar{X}_t\bar{X}_t^T + \epsilon I_2\right) \tag{3.24}$$

Aqui $\bar{X}_t$ é o vetor coluna com a média dos pontos amostrais até o ponto y, e I_2 é a matriz identidade bidimensional.

Esse método depende de dois parâmetros: s_d e ϵ; o primeiro é um parâmetro de escala que deve ser determinado de acordo com a dimensão do espaço paramétrico do qual estamos amostrando. Seguindo a recomendação de G.O. Roberts e Gelman (1996), definimos este parâmetro como $s_d = (2.24)^2/2$. O segundo parâmetro, ϵ, serve apenas para evitar que a matriz de covariância se torne singular; o valor de ϵ deve ser pequeno, porém estritamente positivo ($\epsilon > 0$ é condição necessária para a ergodicidade da cadeia); em nossa implementação do algoritmo, definimos $\epsilon = 1e^{-30}$

Enfim, após o período de queima, a amostragem começa. Nesta fase, geramos candidatos (d_{cand}, s_{cand}) de uma Gaussiana $\mathcal{N}\left((d_{cur}, s_{cur}), \Sigma\right)$, onde agora a matriz de covariância está fixada. A cada passo, obtemos (d_{cur}, s_{cur}) aplicando a probabilidade de aceitação usual de MH nos pontos candidatos. Por fim, obtemos a posteriori $P(d_{cur}, s_{cur}|y^1, y^2)$, comparamos seu valor ao máximo sob H_0, e atualizamos o número de pontos com posteriori maior do que este máximo. Este número, dividido pelo número total de pontos amostrados, é o valor de evidência $\overline{ev(H_0)}$ **contra** H_0 (ou seja, o valor de evidência **a favor** da divisão atual em segmentos).

A implementação atual do algoritmo permite a simulação de cadeias em paralelo. Muito embora não haja ganho substancial de desempenho, uma vez que a fase de queima não pode ter o tamanho reduzido / paralelizado, o uso de cadeias paralelas permite a análise da convergência da cadeia, utilizando por exemplo a estatística $\hat{R}$ de Gelman-Brooks-Rubin (Brooks e Gelman (1998)).

Essa estatística pode ser obtida da seguinte maneira: suponha que rodamos M cadeias independentes, cada um com pontos iniciais distintos obtidos idealmente de uma distribuição com sobredispersão (para garantir que as cadeias iniciam de pontos distantes). De cada cadeia obtemos n amostras dos parâmetros de interesse θ, e definimos

$$\hat{\theta}_m = \frac{1}{n}\sum_{i=1}^{n}\theta_{m,i} \tag{3.25}$$

$$\hat{\sigma}_m^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\theta_{m,i} - \hat{\theta}_m)^2 \tag{3.26}$$

$$\hat{\theta} = \frac{1}{M}\sum_{i=1}^{M}\hat{\theta}_m \tag{3.27}$$

$$B = \frac{n}{M-1}\sum_{i=1}^{M}\left(\hat{\theta}_m - \hat{\theta}\right)^2 \tag{3.28}$$

$$W = \frac{1}{M}\sum_{i=1}^{M}\hat{\sigma}_m^2 \tag{3.29}$$

Finalmente, definimos a variância combinada $\hat{V}$ como

$$\hat{V} = \frac{n-1}{M}W + \frac{M+1}{Mn}B \tag{3.30}$$

e a estatística $\hat{R}$ de Gelman-Brooks-Rubin como

$$\hat{R} = \frac{\hat{V}}{W} \tag{3.31}$$

O valor de $\hat{R}$ deve ser próximo de 1 em cadeias com convergência adequada. Se esta estatística apresentar valores maiores do que 1, o período de queima e / ou o tamanho da amostra devem ser aumentados.

Em nosso algoritmo de segmentação, as amostras geradas por MCMC serão usadas num teste de igualdade de variância entre dois segmentos. Para analisar a convergência das cadeias em situações como essa, simulamos sinais Gaussianos com $1MM$ de pontos, variância 1 e $\delta \in \{1; 1,1; 1,5; 2\}$. Quando $\delta > 1$, o ponto de corte será em $\bar{t} = 500.000$.

Obtemos amostras de tamanho $\{1.000, 10.000, 100.000\}$, a partir de $M = 4$ cadeias paralelas em cada situação. Queimamos o mesmo número de pontos em cada situação, e rodamos a fase inicial do algoritmo com $t_0 = 500$. Adotamos $\beta = 1$ em todas as simulações. Na tabela 3.5 reportamos a estatística $\hat{R}$ e algumas outras medidas resumo para as simulações.

De acordo com a estatística GBR, e com o sinal simulado conforme especificações acima, um tamanho de amostra igual a 10.000 pontos já é suficiente para atingir convergência das cadeias utilizadas no cálculo da evidência. A taxa de aceitação fica entre 6% e 8%, que podem ser consideradas razoáveis. Com amostras de tamanho 100.000 essa taxa aumenta para cerca de 13%, indicando que o método adaptativo de Haario de fato melhora a distribuição candidata.

A escolha de um valor específico para o tamanho da cadeia, porém, depende das características do sinal analisado; sendo assim, este valor será um parâmetro do algoritmo.

Finalmente, na próxima seção, descrevemos o módulo Python *bayeseg*, que implementa o algoritmo de segmentação sequencial.

3.5.3 Módulo *bayeseg*

A versão otimizada do algoritmo SeqSeg está implementada no módulo *bayeseg*, disponivel em http://github.com/paulohubert/bayeseg.

Este módulo é composto de duas classes principais: *OceanPod*, uma classe que faz a interface com os arquivos de áudio, em especial permitindo a recuperação de segmentos a partir dos *timestamp* de início e duração (apoiando-se nos padrões de data e hora no nome dos arquivos do OceanPod),

Tabela 3.5: Análise dos valores de evidência por MCMC

δ	Tamanho da cadeia	Evidência mínima	Evidência máxima	Taxa de aceitação	$\hat{\delta}$	$\hat{\sigma}_0$	$\hat{R}_\delta$	$\hat{R}_\sigma$
1,0	1.000	1,00000	1,00000	0,019250	1,000802	0,998831	1,239326	1,097055
1,0	10.000	0,99760	1,00000	0,061075	0,999650	0,998841	1,006411	1,001696
1,0	100.000	0,99562	0,99747	0,147852	0,999800	0,998811	1,000015	1,000212
1,1	1.000	0,00000	0,00000	0,014500	1,103171	0,998679	1,081629	1,053349
1,1	10.000	0,00000	0,00000	0,088725	1,102898	0,998835	1,005252	1,003423
1,1	100.000	0,00000	0,00000	0,126205	1,102852	0,998865	1,000383	1,000240
1,5	1.000	0,00000	0,00000	0,034500	1,503393	0,998099	1,179845	1,041876
1,5	10.000	0,00000	0,00000	0,089450	1,502498	0,998415	1,002443	1,002263
1,5	100.000	0,00000	0,00000	0,126858	1,502448	0,998406	1,000255	1,000358

e *SeqSeg*, a classe que implementa os métodos utilizados para a segmentação do sinal. A classe permite que o usuário carregue os dados do sinal, inicialize os valores dos parâmetros, e obtenha a segmentação.

O método *feed_data* recebe um vetor NumPy como argumento, armazena este vetor e pré-processa o sinal calculando as somas acumuladas $\sum_i y_i^2$.

O método *initialize* recebe como argumentos os seguintes valores:

- *double* $\beta > 0$: o hiperparâmetro da distribuição a priori para δ;
- *double* $0 < \alpha < 1$: o valor de corte para a função de decisão; o algoritmo vai aceitar o ponto de corte atual sempre que a evidência a favor de $H_0 : \delta = 1$ for maior do que α; ou seja, quanto maior o valor de α, maior o número de segmentos retornados;
- *int mciter*: número de amostras MCMC para o cálculo da evidência;
- *int mcburn*: número de pontos no período de queima do MCMC;
- *int nchains*: número de cadeias em paralelo.

Para definir completamente o algoritmo SeqSeg, restam dois parâmetros: o tamanho mínimo de segmento, e a resolução temporal.

O tamanho mínimo de segmento é o parâmetro que garante a parada do algoritmo. Sempre que um ponto estimado de corte resultar em algum segmento com tamanho menor do que este mínimo, o cálculo de evidência é ignorado e o algoritmo rejeita automaticamente a segmentação.

A resolução temporal, como vimos, é um parâmetro que permite acelerar o algoritmo ao custo de alguma precisão. Quando a resolução temporal é igual a 1, a estimação MAP do ponto de corte vai envolver a avaliação da posteriori em todos os pontos do seu suporte. Uma resolução maior do que 1 implica em que a posteriori será avaliada apenas num reticulado uniforme do suporte, com os pontos separados por um inteiro t_{res}.

Ambos parâmetros são passados para o método *segments* da classe. Este é o método principal, que retorna o vetor (não-ordenado) de pontos de corte obtidos pelo algoritmo.

Na próxima seção, utilizamos essa implementação rápida para avaliar o desempenho do algoritmo e a sua sensibilidade aos parâmetros α e β em dados simulados.

3.6 Sensibilidade aos parâmetros

Em toda esta seção utilizaremos sinais simulados de acordo com as seguintes especificações: os sinais serão Gaussianos, de amplitude média igual a 0 e variância igual a 1. O tamanho total do sinal será de $1MM$ pontos. Dividimos o sinal em 6 segmentos, com pontos de corte em { 0; 10.000; 110.000; 200.000; 500.000; 750.000; 1.000.000 }. O primeiro segmento será mantido com variância igual a 1; a partir daí, multiplicamos a variância dos segmentos por δ, sempre alternando entre um segmento com variância 1 e outro com variância δ.

Na primeira simulação, δ é sempre o mesmo para todos os segmentos. Rodamos testes com $\delta \in \{1,0; 1,1; 1,5\}$. Os parâmetros do algoritmo de segmentação foram fixados com resolução temporal igual a 1 e 10.000 amostras para o MCMC (com mais 10.000 pontos de queima).

Por fim, variamos $\beta \in \{ 1; 0,1; 0,01; 0,001; 0,0001; 0,00001 \}$ e $\alpha \in \{0,1; 0,5; 0,9; 0,99\}$. Para cada combinação de α e β rodamos a segmentação 30 vezes, armazenando o tempo médio de computação, e o número mínimo e máximo de segmentos obtidos. As tabelas 3.6 a 3.9 apresentam os resultados.

Para os sinais simulados e nesta região de α e β, os resultados do algoritmo são satisfatórios: quando $\delta = 1$ (ou seja, no caso em que não há nenhum segmento), todas as combinações de parâmetros resultam em 1 segmento, correspondente ao sinal inteiro. Com $\delta = 1,1$, os seis segmentos são identificados corretamente, exceto quando β é muito pequeno. Isto é esperado, pois β é o parâmetro da priori de Laplace para δ; quanto menor o seu valor, mais pronunciado o pico da priori

Tabela 3.6: Resultados para $\alpha = 0.1$

δ	β	# mínimo de segmentos	# máximo de segmentos	Tempo médio (s)
1,0	1,00000	1	1	0,013725
1,0	0,10000	1	1	0,013490
1,0	0,01000	1	1	0,013503
1,0	0,00100	1	1	0,013505
1,0	0,00010	1	1	0,016907
1,0	0,00001	1	1	0,018005
1,1	1,00000	6	6	0,098041
1,1	0,10000	6	6	0,113072
1,1	0,01000	6	6	0,096792
1,1	0,00100	5	5	0,097185
1,1	0,00010	1	1	0,040666
1,1	0,00001	1	1	0,029612
1,5	1,00000	6	6	0,093532
1,5	0,10000	6	6	0,097517
1,5	0,01000	6	6	0,096968
1,5	0,00100	6	6	0,097118
1,5	0,00010	3	4	0,064936
1,5	0,00001	1	1	0,028431

Tabela 3.7: Resultados para $\alpha = 0.5$

δ	β	# mínimo de segmentos	# máximo de segmentos	Tempo médio (s)
1,0	1,00000	1	1	0,013530
1,0	0,10000	1	1	0,013495
1,0	0,01000	1	1	0,013553
1,0	0,00100	1	1	0,013726
1,0	0,00010	1	1	0,013643
1,0	0,00001	1	1	0,018850
1,1	1,00000	6	6	0,096758
1,1	0,10000	6	6	0,096947
1,1	0,01000	6	6	0,097188
1,1	0,00100	5	5	0,096727
1,1	0,00010	1	1	0,028671
1,1	0,00001	1	1	0,029201
1,5	1,00000	6	6	0,093676
1,5	0,10000	6	6	0,100382
1,5	0,01000	6	6	0,096994
1,5	0,00100	6	6	0,096927
1,5	0,00010	4	4	0,083853
1,5	0,00001	1	1	0,028569

Tabela 3.8: Resultados para $\alpha = 0.9$

δ	β	# mínimo de segmentos	# máximo de segmentos	Tempo médio (s)
1,0	1,00000	1	1	0,015604
1,0	0,10000	1	1	0,013489
1,0	0,01000	1	1	0,013576
1,0	0,00100	1	1	0,013651
1,0	0,00010	1	1	0,013650
1,0	0,00001	1	1	0,013503
1,1	1,00000	6	6	0,093792
1,1	0,10000	6	6	0,097041
1,1	0,01000	6	6	0,096902
1,1	0,00100	5	5	0,098112
1,1	0,00010	1	1	0,028455
1,1	0,00001	1	1	0,028557
1,5	1,00000	6	6	0,093640
1,5	0,10000	6	6	0,098186
1,5	0,01000	6	6	0,097044
1,5	0,00100	6	6	0,097160
1,5	0,00010	4	4	0,083806
1,5	0,00001	1	1	0,028375

Tabela 3.9: Resultados para $\alpha = 0.99$

δ	β	# mínimo de segmentos	# máximo de segmentos	Tempo médio (s)
1,0	1,00000	1	1	0,013522
1,0	0,10000	1	1	0,013504
1,0	0,01000	1	1	0,013584
1,0	0,00100	1	1	0,013589
1,0	0,00010	1	1	0,013940
1,0	0,00001	1	1	0,013625
1,1	1,00000	6	6	0,103654
1,1	0,10000	6	6	0,096897
1,1	0,01000	6	6	0,097490
1,1	0,00100	5	5	0,097019
1,1	0,00010	1	1	0,028837
1,1	0,00001	1	1	0,028453
1,5	1,00000	6	6	0,093719
1,5	0,10000	6	6	0,101384
1,5	0,01000	6	6	0,096989
1,5	0,00100	6	6	0,097059
1,5	0,00010	4	4	0,082309
1,5	0,00001	1	1	0,039567

em torno de $\delta = 1$. Ou seja, quanto menor o valor de β mais evidência exigimos dos dados para aceitar que de fato $\delta \neq 1$.

Nesta simulação, a escolha de α não teve influência alguma no comportamento do algoritmo. Isto significa que, para estes dados, e para um mesmo valor de β, a evidência sobre a hipótese de igualdade de variâncias convergiu para 0 ou 1, de forma que o próprio valor de evidência já se comporta como uma função binária de decisão.

Por fim, os tempos de computação deste sinal são da ordem de décimos de segundo, o que mostra o ganho de desempenho com a utilização do pacote Cython e da paralelização da estimativa MAP do ponto de corte.

Como um segundo teste, adotamos a mesma estrutura de segmentação do sinal simulado, mas variando δ conforme os segmentos. Utilizamos $\delta = 1,1$ para o segundo segmento, $\delta = 1,5$ para o quarto segmento, e $\delta = 1,2$ para o sexto e último. Fixamos $\alpha = 0,1$ e rodamos o algoritmo com β num reticulado uniforme de 100 pontos entre $1e^{-5}$ e $1e^{-3}$, e em outro reticulado uniforme de 100 pontos entre $1e^{-3}$ e $1e^{-1}$. A definição dos segmentos portanto é como segue:

1. De $i = 0$ to $i = 10,000$, com variância 1;
2. De $i = 10.001$ to $i = 110.000$, com variância $1,1$;
3. De $i = 110.001$ to $i = 200.000$, com variância 1;
4. De $i = 200.001$ to $i = 500.000$, com variância $1,5$;
5. De $i = 500.001$ to $i = 750.000$, com variância 1;
6. De $i = 750.001$ to $i = 1.000.000$, com variância $1,2$

Os resultados mostram que não há segmentação alguma quando β é muito pequeno (menor do que $4e^{-05}$). Após isso, o número de segmentos cresce rapidamente, atingindo 5 segmentos quando $\beta \approx 0,0001$. A partir desse valor, o algoritmo estabiliza, até finalmente atingir o número correto de segmentos quando $\beta \approx 0,0007$. A partir daí o algoritmo se estabiliza novamente, exceto por alguns valores de β quando aparece um segmento extra. Este comportamento está ilustrado nas figuras 3.47 e 3.48.

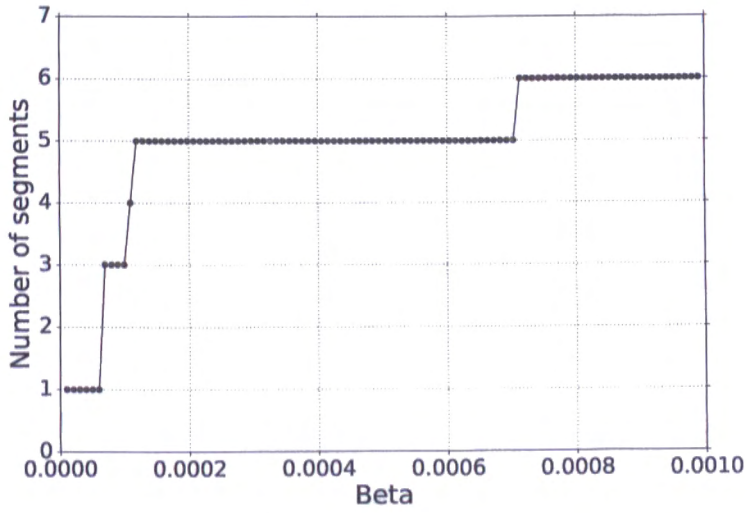


Figura 3.47: Número de segmentos para diferentes valores de β ($\alpha = 0,1$)

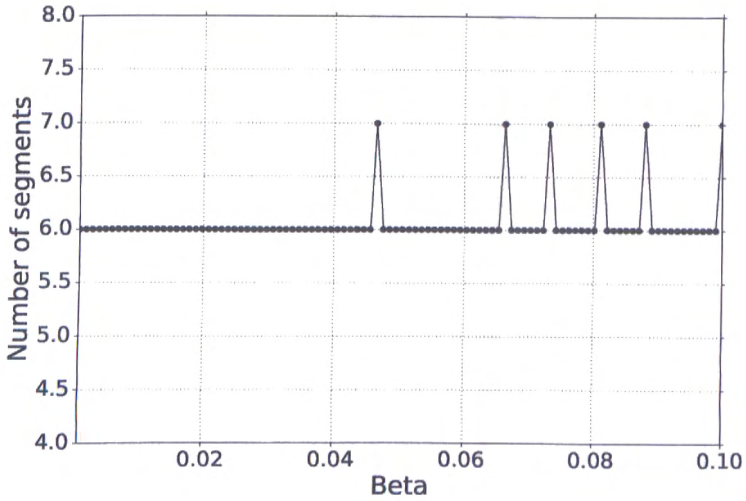


Figura 3.48: Número de segmentos para diferentes valores de β ($\alpha = 0,1$)

Se analisamos a seguência de segmentos obtidos conforme aumentamos o valor de β, vemos que os primeiros pontos de corte encontrados são $\bar{t} \approx 500.000$ e $\bar{t} \approx 750.000$, que separam a seção de maior variância, $500.000 < i < 750.000$, do restante do sinal. Em seguida, os pontos de corte são estimados em $\bar{t} = 200.000$ e $\bar{t} = 110.000$; por fim, o último segmento a ser identificado é o de tamanho menor, em $\bar{t} = 10.000$.

Quando ocorre a detecção de um falso positivo (para alguns valores altos de β), este falso positivo sempre aparece no segmento entre $i = 500.000$ e $i = 750.000$, o segmento com maior variância e portanto com maior probabilidade de ocorrência de valores extremos.

Em seguida, rodamos o algoritmo num reticulado uniforme de 100 pontos para α, entre $[0,01, 0,99]$. Neste caso, se fixamos $\beta = 0,001$, não há qualquer variação no número de segmentos conforme o valor de α, o que confirma a observação anterior de que o algoritmo é bastante robusto a valores deste parâmetro. Esta robustez porém depende de uma boa calibração de β: se fixamos$\beta = 0,1$ e variamos α no mesmo reticulado, vamos sobressegmentação ocorrendo para valores mais altos de α, conforme a figura 3.49.

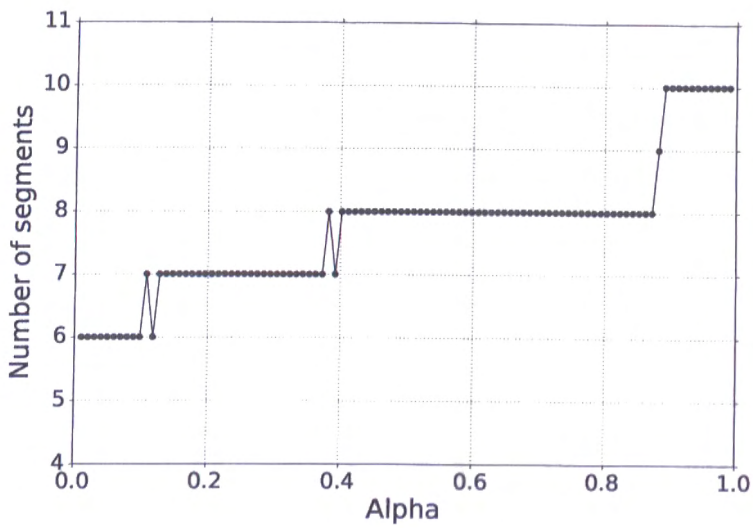


Figura 3.49: Número de segmentos para diferentes valores de α ($\beta = 0,1$)

Esses testes indicam que o parâmetro crítico a ser calibrado neste algoritmo é o valor de β; se este parâmetro for bem ajustado, os valores de α perdem importância. Se o valor de β for muito alto, a sobressegmentação vai aumentar com α.

Tabela 3.10: Segmentação das amostras reais

Amostra	β	# de segmentos	Tempo (s)
2015,01,30_02,02,56,wav	0,000005	1,0	0,195583
2015,01,30_02,02,56,wav	0,000010	1,0	0,136689
2015,01,30_02,02,56,wav	0,000014	2,0	0,194111
2015,01,30_02,02,56,wav	0,000028	3,0	0,253772
2015,01,30_02,02,56,wav	0,000041	3,0	0,238949
2015,01,30_02,02,56,wav	0,000050	3,0	0,249274
2015,02,02_07,50,49,wav	0,000005	3,0	0,207852
2015,02,02_07,50,49,wav	0,000010	3,0	0,228707
2015,02,02_07,50,49,wav	0,000014	4,0	0,270753
2015,02,02_07,50,49,wav	0,000028	7,0	0,435874
2015,02,02_07,50,49,wav	0,000041	9,0	0,528216
2015,02,02_07,50,49,wav	0,000050	11,0	0,661587
2015,02,08_11,26,39,wav	0,000005	7,0	0,463836
2015,02,08_11,26,39,wav	0,000010	13,0	0,741053
2015,02,08_11,26,39,wav	0,000014	15,0	1,319725
2015,02,08_11,26,39,wav	0,000028	27,0	1,429226
2015,02,08_11,26,39,wav	0,000041	30,0	1,581087
2015,02,08_11,26,39,wav	0,000050	32,0	1,651879

Resta observar se estes fatos se repetem na aplicação do algoritmo a sinais reais; isto é o que fazemos na próxima seção.

3.7 Calibração em amostras reais

Para avaliar a calibração dos parâmetros do algoritmo SeqSeg a sinais reais, escolhemos as mesmas amostras do OceanPod que foram utilizadas ao longo deste capítulo. Isto vai permitir que os tempos de computação e resultados sejam diretamente comparáveis entre as seções.

Estas três amostras possuem características bem diferentes: a primeira consiste em apenas ruído, sem nenhum evento significativo; a segunda contém um evento de longa duração, identificado como o som de uma embarcação de longo porte; e o terceiro contém diversos eventos curtos, identificados como o ruído de motores de embarcações de pequeno porte.

Fixamos o tamanho mínimo de segmento e a resolução temporal em 11.025 pontos; além disso, amostramos 20.000 pontos da cadeia MCMC. Estes são os mesmos parâmetros utilizados na seção anterior em que primeiro apresentamos os resultados do algoritmo SeqSeg.

Nestes testes, o valor de α foi fixado em $0,1$, e rodamos o algoritmo tomando valores de β em um reticulado uniforme de 100 pontos em $[5e^{-6}, 5e^{-5}]$. Os resultados estão resumidos na tabela 3.10, e o gráfico do número de segmentos como função de β para cada amostra estão na figura 3.50.

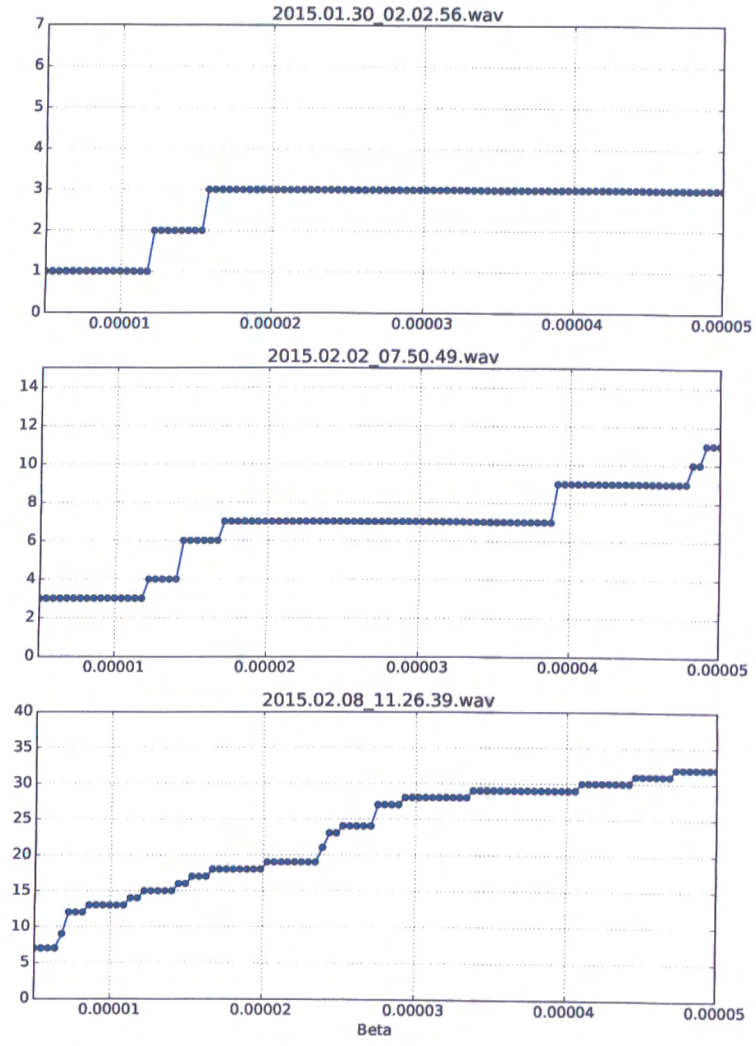


Figura 3.50: Número de segmentos em função de β

O tempo de execução total nestas amostras em comparação com a tabela 3.3 caiu de ≈ 100 a $200s$ para $\approx 0,2$ a $1,7s$, uma redução da ordem de $100\times$.

Observamos que o número de segmentos cresce com β, principalmente na amostra do dia 08 de fevereiro (a amostra contendo vários eventos de curta duração). Se tomamos o número real de segmentos nas duas primeiras amostras (30/janeiro e 02/fevereiro) como sendo 1 e 3, respectivamente, vemos que seria necessário utilizar $\beta < 0,000012$ para obter o número correto; com estes valores, porém, obteríamos na amostra do dia 08 apenas em torno de 12 segmentos, o que é muito pouco (a inspeção visual do espectrograma indica em torno de 30 segmentos nesta amostra).

Isto implica em que não há necessariamente um único valor de β ótimo para cada possível amostra. Se for necessário escolher um único valor de β para segmentar diferentes amostras, será preciso escolher entre a sobressegmentação de sinais com poucos eventos ou a subsegmentação de sinais com muitos eventos.

Por outro lado, podemos adotar uma versão do algoritmo que seleciona automaticamente o valor de β mais adequado, da seguinte maneira: observando os resultados das três segmentações acima, vemos que existem regiões para β onde o algoritmo se estabiliza. Isto é verdade nos valores menores de β nas duas primeiras amostras, e em valores um pouco mais altos (em torno de $\beta = 0,000035$) para a terceira amostra. Um método possível, portanto, seria escolher um valor inicial pequeno para β, efetuar a segmentação, e aumentar β ligeiramente, repetindo a segmentação. A partir do momento em que o número de segmentos se tornasse estável (por exemplo, quando for o mesmo ao longo de r execuções sucessivas), paramos o procedimento, adotando esta segmentação estável.

Este procedimento está descrito no algoritmo 6 a seguir.

Algorithm 6 Estabilização do algoritmo SeqSeg

procedure CALIBRAÇÃO(y, β_{min}, Δ_β, $maxiter$, N_{min})
  $n_{eq} = 0$, $t = 0$, $cont = 0$
  $\beta = \beta_{min}$
  $n_0 = \#\{SeqSeg(y, \beta_{min})\}$
  while $n_{eq} < N_{min}$ e $cont < maxiter$ **do**
    $n_t = \#\{SeqSeg(y, \beta)\}$
    if $n_t = n_{t-1}$ **then**
      $n_{eq} = n_{eq} + 1$
    else
      $n_{eq} = 0$
    end if
    $\beta = \beta + \Delta_\beta$
    $cont = cont + 1$
  end while
end procedure

No algoritmo acima, $\#\{SeqSeg(y)\}$ denota a cardinalidade do conjunto retornado pelo algoritmo de segmentação (i.e., o número de segmentos).

Em resumo, o algoritmo de estabilização começa adotando $\beta = \beta_{min}$, e vai incrementar β em Δ_β a cada passo, obtendo uma nova segmentação; se o número de segmentos permanecer o mesmo por N_{min} iterações, o algoritmo para, adotando a segmentação atual.

Utilizando este método de calibração nas amostras acima, com $N_{min} = 10$, $\beta_{min} = 0,000005$ e $\Delta_\beta = 0,000005$, obteríamos 1 segmentos para a amostra do dia 30/janeiro, 3 segmentos para o dia 02/fevereiro, e 29 segmentos para o dia 08/fevereiro.

3.8 Conclusão

Neste capítulo desenvolvemos um método para segmentação não-supervisionada de sinais, baseado inteiramente em argumentos probabilísticos. A única hipótese feita sobre os segmentos é de que eles possuem potências (i.e. variâncias) diferentes, e a partir desta hipótese os pontos de segmentação são estimados de maneira sequencial.

O FBST foi utilizado novamente, desta vez para comparar as variâncias dos sucessivos segmentos obtidos. Comparado a outros critérios de evidência para a mesma hipótese, o FBST se mostrou superior.

A única dificuldade com o critério do FBST é o tempo de computação. Apresentamos, no entanto, uma implementação do algoritmo de segmentação utilizando a biblioteca Cython da linguagem Python, que permitiu atingirmos um desempenho computacional bastante satisfatório.

Apresentamos, por fim, uma proposta de metodologia para escolha dos hiperparâmetros do algoritmo, em particular o parâmetro β da priori de Laplace. Com este algoritmo de estabilização / calibração, fica completamente definida a metodologia de segmentação de sinais.

O próximo passo após a segmentação é processar os segmentos em busca de exemplos de um mesmo tipo de evento. Este é o assunto do próximo capítulo.

Uma parte dos resultados deste capítulo está publicada em Hubert *et al.* (2018a). O *preprint* de um artigo descrição detalhada do módulo *bayeseg* está disponivel em Hubert *et al.* (2018b).

Capítulo 4

Categorização de segmentos

No capítulo anterior construímos um algoritmo de segmentação de sinais baseado na hipótese de mudança na potência média do sinal entre segmentos. O algoritmo mostrou-se eficiente na segmentação de sinais simulados e, principalmente, de sinais acústicos reais obtidos do OceanPod.

Segmentar o sinal já representa um ganho importante na análise do sinal acústico; baseado nas características de cada segmento (duração total, potência, hora de ocorrência) é possível executar um filtro inicial para separar para a inspeção manual apenas os segmentos de maior interesse (por exemplo, apenas segmentos curtos de alta potência). Apenas executando a segmentação o trabalho de anotação desta base de dados é drasticamente reduzido.

Por outro lado, ainda será necessário inspecionar muitos segmentos para obter uma amostra anotada de eventos subaquáticos, em particular se o sinal analisado for de longa duração (o que é o caso no problema em que estamos interessados nesta tese). Isto nos motiva a ir um passo além, e propor um algoritmo de **categorização** de sinais.

O objetivo é agrupar os segmentos obtidos no passo anterior em classes ou categorias, de forma que os segmentos dentro de uma mesma classe sejam homogêneos entre si, e de modo que as classes sejam heterogêneas. Desta forma, esperamos que uma determinada classe reúna segmentos representativos de um mesmo tipo de evento.

Para efetuar essa categorização, precisamos levar em conta que os segmentos terão durações distintas; portanto, os vetores a serem agrupados não estarão no mesmo espaço. Será preciso, portanto, propor uma transformação que leve todos os segmentos a um mesmo espaço, para então definirmos nesse espaço uma ou mais métricas que permitam a comparação dos segmentos. Esta estratégia corresponde à ideia de comparar indivíduos com base num conjunto de *features*, ou características. Será preciso argumentar que tais características representam suficientemente bem os segmentos (ou ao menos os aspectos dos segmentos que importam na construção da categorização).

Outra estratégia será a de utilizar alguma medida de similaridade (ou dissimilaridade) que seja capaz de trabalhar com vetores de dimensões distintas. Uma tal medida comum na literatura de análise e processamento de sinais (entre outras áreas) é a técnica conhecida como *Dynamic Time Warping* (DTW). Esta técnica permite obter a medida de similaridade diretamente sobre os segmentos originais, mesmo quando estes apresentam durações distintas. Apresentaremos esta técnica com mais detalhe abaixo; antes, porém, vamos descrever duas possibilidades de extração de características dos sinais para posterior classificação.

4.1 Caracterização dos segmentos

Denominamos caracterização dos segmentos a tarefa de encontrar uma representação do sinal de tal forma que sinais de dimensões distintas sejam representados no mesmo espaço. Esta tarefa corresponde à extração de características do sinal que carreguem informação relevante sobre sua origem, de modo que a categorização feita com base nessas características de fato capture grupos representativos de eventos da mesma natureza.

Conforme já observamos acima, os segmentos obtidos do algoritmo SeqSeg terão durações distintas. Sendo assim, nossa tarefa é encontrar alguma transformação $F : \Re^{N_i} \rightarrow \Re^D$ (ou uma família de transformações $F_i : \Re^{N_i} \rightarrow \Re^D$) que leve cada segmento de dimensão N_i a um vetor de características do sinal, de modo que esse vetor pertença sempre ao mesmo espaço de dimensão D.

Vamos propor neste trabalho duas alternativas para a caracterização do sinal: uma baseada na transformada discreta de Fourier, e outra no modelo probabilístico de chirp.

4.1.1 Espectro médio

Uma maneira de obter essa transformação é trabalhar com a DFT (transformação discreta de Fourier) de cada segmento. A DFT transforma um sinal no domínio do tempo em um sinal no domínio da frequência; sabe-se, porém, que a resolução no domínio da frequência é função da duração do sinal no domínio do tempo, enquanto a frequência máxima, obtida do teorema de Nyquist-Shannon, depende apenas da frequência de amostragem. Portanto, obter a DFT de dois sinais com durações distintas mas frequências de amostragem idênticas vai resultar em vetores não comparáveis. A DFT por si só, portanto, não resolve nosso problema de caracterização dos segmentos.

Por outro lado, uma estratégia simples, ainda baseada na DFT, seria escolher um certo período de tempo, por exemplo 1 segundo, e percorrer o sinal em janelas dessa duração, obtendo a DFT de cada uma dessas janelas. Por fim, combinamos essas janelas num único espectro, tomando por exemplo o espectro médio de todas as janelas. Implícita nesta abordagem esta a hipótese de que o espectro do sinal é constante ao longo das diferentes janelas.

Nas figuras 4.1 e 4.2 vemos as DFTs médias de 1 segundo dos segmentos obtidos da aplicação do algoritmo SeqSeg em duas amostras distintas do OceanPod: a primeira, a amostra já conhecida do dia 08/02/2015 às 11h:26m da manhã; a outra, do dia 22/02/2015 às 16h12min; esta última amostra, assim como a do dia 08/02, apresenta diversas ocorrências de sons de motores de embarcações de pequeno porte. Os espectros que aparecem nesses gráficos foram normalizados, e truncados nos primeiros 500 pontos. Os eixos verticais foram normalizados para a mesma escala.

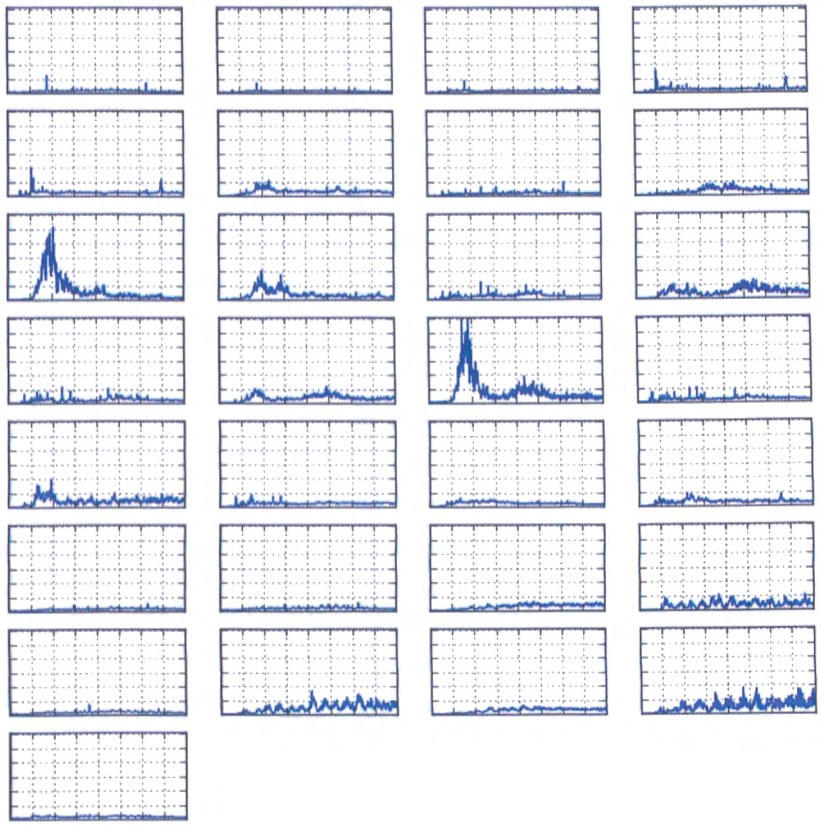


Figura 4.1: DFTs médias de 1 segundo, segmentos da amostra do dia 08/02/2015

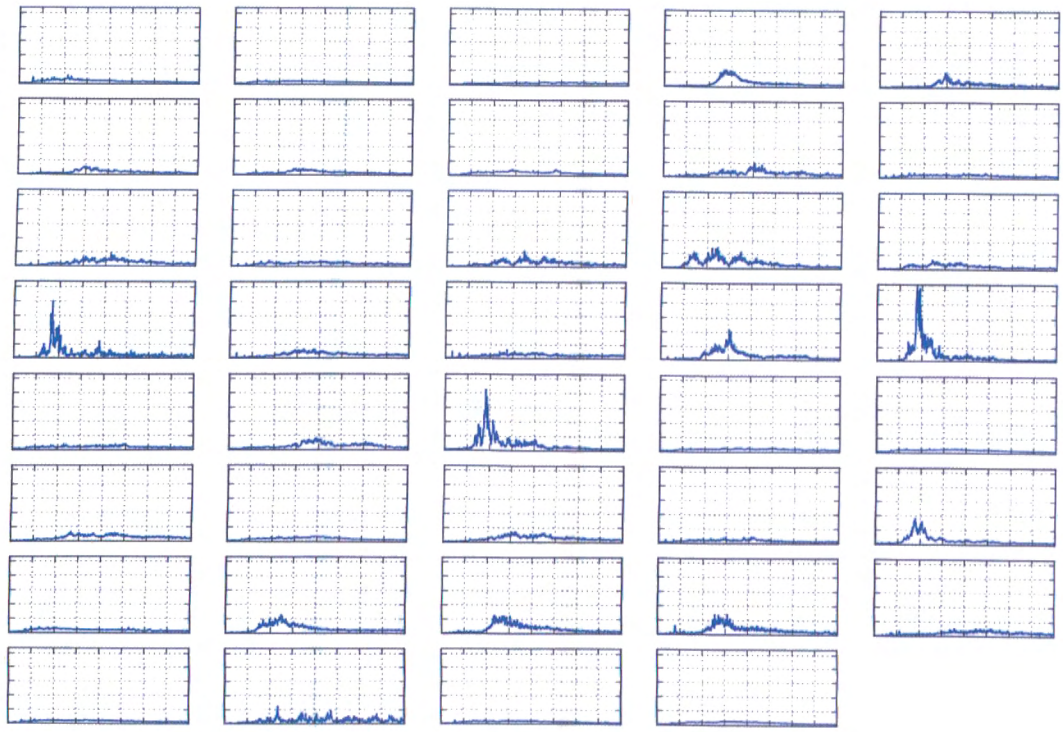


Figura 4.2: DFTs médias de 1 segundo, segmentos da amostra do dia 22/02/2015

Para caracterizar e, posteriormente, categorizar os segmentos, o método da DFT média tem a vantagem da simplicidade, e de levar em consideração todo o espectro do sinal, e não apenas uma ou algumas frequências (por exemplo a frequência de máxima energia). A principal desvantagem desse método é a hipótese implícita de que o espectro do sinal é constante (e por isso pode ser bem estimado em janelas contíguas de 1 segundo de duração).

4.1.2 Chirpograma

Conforme observamos no capítulo 2 deste trabalho, a análise da frequência fundamental não é suficiente para a caracterização de certos sinais de interesse na análise acústica submarina; ao construir a metodologia de detecção binária, foi necessário acrescentar mais um parâmetro e trabalhar com o modelo de *chirp*

$$f(t_i) = cos(\alpha t_i^2 + \omega t_i + \phi) \tag{4.1}$$

Além disso, a literatura da análise Bayesiana de sinais, que revisamos no capítulo 1, mostra que a DFT não é a metodologia ótima para estimação de frequências em sinais contaminados por ruído; quando a frequência fundamental varia no tempo, como é o caso nos modelos de *chirp*, a DFT torna-se ainda menos adequada.

Sendo assim, para caracterizar os segmentos obtidos da análise do sinal acústico, propomos utilizar o *chirpogram* de Jaynes (1987), definido por

$$C(\omega, \alpha) = \frac{1}{N} \sum_{t=1}^{N} \sum_{s=1}^{N} y_t y_s cos\left[\omega(t-s) + \alpha(t^2 - s^2)\right] \tag{4.2}$$

Esta função é obtida da análise probabilística do modelo de *chirp*, quando a fase do sinal é

marginalizada; Jaynes (1987) mostra que a função de verossimilhança marginal obtida no modelo de chirp depende dos dados apenas através da função $C(\cdot)$; na terminologia da estatística, portanto, o *chirpograma* é uma estatística conjuntamente suficiente para os parâmetros ω e α do modelo.

Como o chirpograma concentra toda a informação que os dados trazem sobre os parâmetros α e ω, é natural que o utilizemos na caracterização dos segmentos. A utilização do chirpograma com este objetivo já aparece na literatura, por exemplo em Loredo (1990), onde o autor sugere a aplicação da análise do chirpograma a problemas de astrofísica (em particular à análise de ondas gravitacionais).

O chirpograma pode ser obtido para quaisquer valores de α e ω. Isto significa que podemos definir um reticulado para esses dois parâmetros que seja independente do tamanho do segmento; obtemos então para cada segmento o valor de $C(\omega, \alpha)$ nesse reticulado, e podemos enfim utilizar a matriz resultante para caracterizar o sinal.

A dificuldade deste método está em definir os limites para o reticulado; uma definição precisa exigiria conhecimento prévio sobre o intervalo mais provável para os dois parâmetros. Ao mesmo tempo, um intervalo mais amplo exige uma menor resolução do chipograma (pontos mais espaçados no reticulado), ou um custo computacional maior, já que a função em 4.2 precisará ser avaliada em mais pontos.

Nos gráficos 4.3 e 4.4 vemos os chirpogramas dos segmentos para as amostras de 08/02/2015 e 22/02/2015. Em cada chirpograma, a coordenada horizontal representa a frequência (entre 10 e 500 Hz, em intervalos de 5 Hz) e a coordenada vertical representa o chirp (entre 0 e 2π em intervalos de $0,02\pi$).

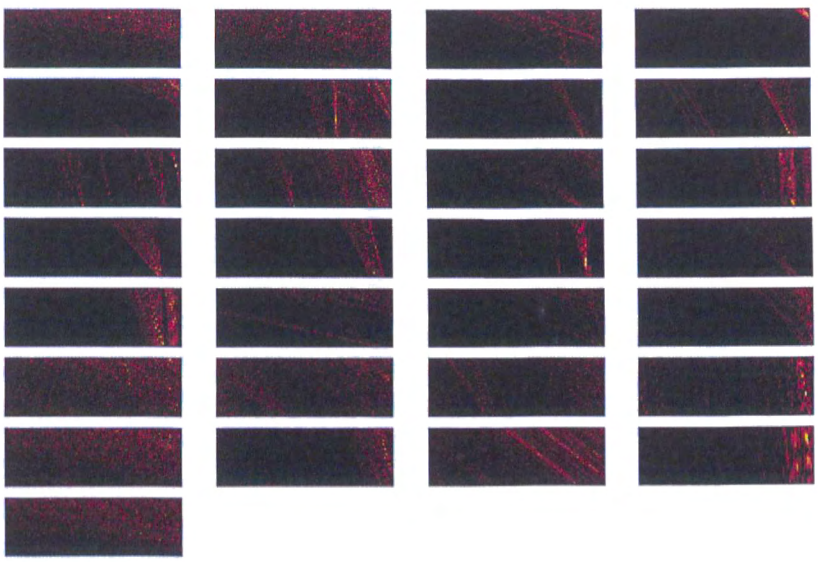


Figura 4.3: Chirpograma, segmentos da amostra do dia 08/02/2015

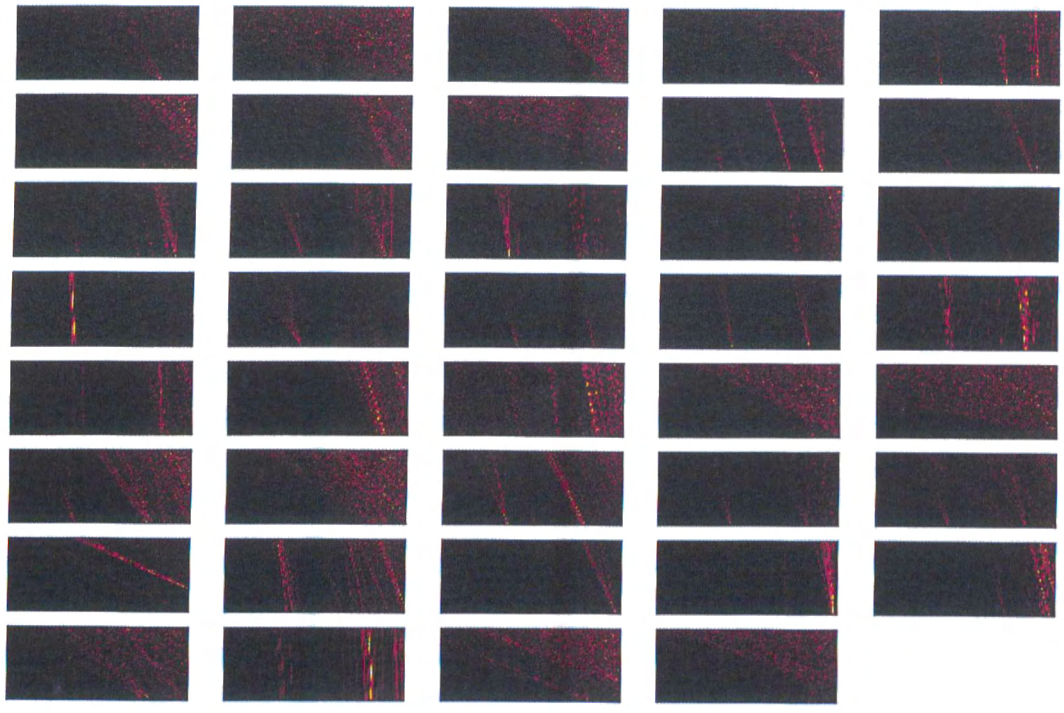


Figura 4.4: Chirpograma, segmentos da amostra do dia 22/02/2015

4.2 Medidas de similaridade

Uma vez escolhido o método de caracterização dos segmentos, é preciso definir uma medida de similaridade (ou, equivalentemente, de dissimilaridade) no espaço das características (*features*).

Para a caracterização do segmento via DFT média, o espaço de chegada será o dos vetores reais não-negativos (por DFT nos referimos à amplitude da transformada discreta de Fourier). Neste espaço, a métrica mais naturalé a norma $\|\cdot\|_2$ (distância euclidiana). Esta norma tem a propriedade de ser contínua e diferenciável, além de ser de fácil implementação.

O problema com a norma euclidiana (ou com qualquer outra norma que compare os vetores pareando suas componentes) no caso da comparação das DFTs reside na seguite observação: imagine dois espectros identicamente nulos exceto em uma faixa de frequência. No primeiro espectro, esta faixa de frequência está entre $50 - 55$ Hz; no segundo, entre $55 - 60$ Hz.

Na maioria das situações práticas, é razoável dizer que os dois espectros acima são bastante semelhantes (ou seja, correspondem a eventos possivelmente parecidos); contudo, a distância euclidiana entre os dois vetores seria alta (a raiz da soma dos quadrados das energias nas faixas respectivas de frequência); um espectro que fosse identicamente nulo, por exemplo, e de acordo com a norma euclidiana, estaria mais próximo de qualquer um desses dois espectros do que eles entre si.

É justamente o reconhecimento dessa limitação nas métricas que trabalham com os vetores pareados (i.e., que comparam as componentes i nos dois vetores) que motiva a definição do Dynamic Time Warping (DTW). A DTW é um método para cálculo de similaridades entre séries de tempo, ou outros conjuntos de dados ordenados em que não se pode assumir a priori que as séries estão perfeitamente alinhadas. As ideias originais que levaram à DTW são da área do reconhecimento de voz (Sakoe e Chiba (1978); Vintsyuk (1968)); porém, na literatura mais recente esta técnica vem sendo aplicada em diferentes domínios, incluindo a análise de sequências de DNA

(H. Skutkova e Provaznik (2013)), o reconhecimento de gestos (ten Holt *et al.* (2007)) e o agrupamento de séries financeiras (P. Tsinaslanidis e Livanis (2014)).

A ideia essencial da DTW é simples: dadas duas sequências s^1 e s^2, de dimensão M e N, respectivamente, construímos uma matriz de distâncias entre as componentes de cada sequência, $D_{M\times N}$. Nesta matriz, a entrada (i, j) representa a distânica unidimensional entre a componente i da sequência s^1 e a componente j da sequência s^2. Ou seja, a matriz D é tal que $d_{ij} = |s_i^1 - s_j^2|$.

Por exemplo, sejam as sequências $s^1 = [0, 1, 1, 0, 0,]$ e $s^2 = [0, 0, 1, 1, 0]$; a distância euclidiana entre essas duas sequências é igual a $\sqrt{2}$, a e norma-1 da diferença é igual a 2. A matriz D obtida para estas duas sequências tem a seguinte forma:

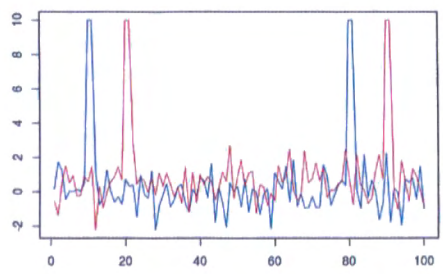
$$D = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Note que o traço dessa matriz é igual à norma-1 da diferença entre os segmentos.

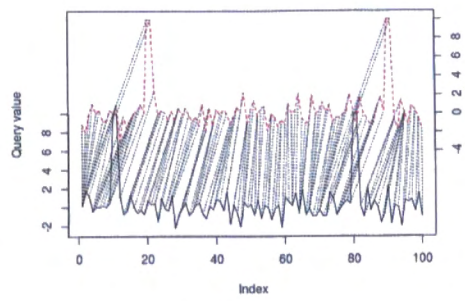
Após a construção da matriz D, encontramos o caminho entre as coordenadas $(1, 1)$ (canto superior esquerdo) e (M, N) (canto inferior direito) da matriz, de tal forma que a soma de distâncias unidimensionais ao longo deste caminho é a menor possível. A distância entre s^1 e s^2 é definida como a soma das distâncias unidimensionais ao longo deste menor caminho.

No exemplo, o caminho seria dado por $(1, 1) \to (1, 2) \to (2, 3) \to (3, 3) \to (3, 4) \to (4, 5) \to (5, 5)$, e a DTW entre as duas sequências seria igual a 0 (refletindo o fato de que um deslocamento temporal permite o perfeito alinhamento das duas sequências).

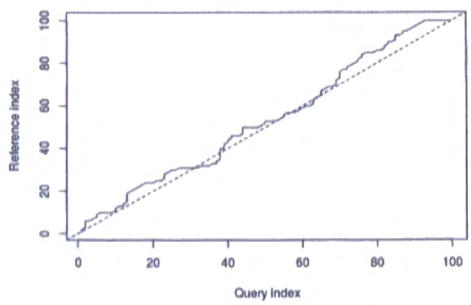
Na figura 4.5 ilustramos o cálculo da DTW para dois sinais simulados de 100 elementos. Os gráficos foram obtidos no R, usando o pacote *dtw* de Giorgino (2009)



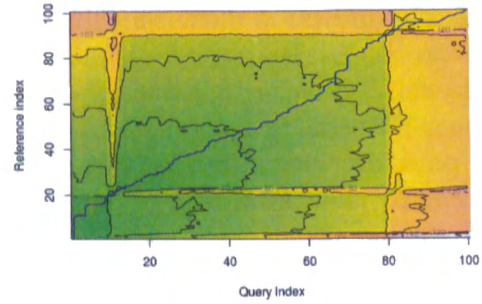
(a) Sinais originais



(b) Alinhamento da DTW



(c) Caminho da DTW



(d) Caminho da DTW com valores de custo

Figura 4.5: Dynamic Time Warping

Assim posto, o problema de encontrar a DTW é um problema de programação dinâmica não-

trivial, com complexidade quadrática no tamanho das sequências. Para acelerar o cálculo da DTW, algumas aproximações foram sugeridas (Salvador e Chan (2007) Gu e Jin (2006)). Tipicamente, as aproximações restringem o tamanho do espaço de busca na geração do caminho entre $(1,1)$ e (M,N), sendo que as restrições mais comuns são o paralelogramo de Itakura (Itakura (1965)) e as bandas de Sakoe-Chiba (Sakoe e Chiba (1978)). Ambas as restrições limitam o caminho na matriz de distâncias unidimensionais, seja forçando-o a permanecer dentro de uma banda de largura fixa em torno da diagonal principal (Sakoe-Chiba), seja forçando-o a permanecer dentro de uma banda de largura variável (paralelogramo de Itakura).

Para a categorização dos segmentos de sinal do OceanPod utilizamos o módulo *fastdtw* do Python, descrito em Salvador e Chan (2007). Este módulo propõe uma aproximação da DTW baseado na ideia de *coarsening and projection*: primeiro, os sinais originais são amostrados de forma a obter novos sinais de resolução temporal menor (*coarsening*). A DTW é obtida para os sinais assim reduzidos, e em seguida projetada (*projection*) para uma resolução maior, onde a DTW é refinada. Neste segundo passo, a matriz de custos é avaliada somente nas células que fazem parte do caminho mínimo encontrado no passo anterior para as séries de dimensão reduzida. O algoritmo de Salvador tem complexidade linear (tanto de tempo de execução quanto de custo de memória) no tamanho original dos sinais.

A DTW, portanto, pode servir de métrica tanto entre as DFTs médias, quanto entre os próprios sinais originais (pois a definição da DTW permite a comparação de sequências de tamanhos distintos). Resta portanto apenas definir as métricas sobre os chirpogramas.

Como vimos na seção anterior, os chirpogramas serão matrizes reais. Neste caso, a métrica mais natural e imediata é a distância de Frobenius $\|\cdot\|_F$ entre os chirpogramas. Contudo, da mesma forma que a distância euclidiana no caso dos vetores unidimensionais, a distância de Frobenius pareia as matrizes componente a componente, o que pode distorcer o cálculo da similaridade em situações onde os chirpogramas exibem padrões similares porém deslocados (ou distorcidos) no espaço $\alpha \times \omega$.

Podemos então aplicar a DTW também aos chirpogramas. No entanto, a generalização da DTW para o caso multidimensional não é imediata nem trivial; é possível provar, inclusive, que o problema exato da DTW é NP-completo no caso multidimensional (Keysers e Unger (2003)). Aproximações, porém, estão disponíveis (M. Shokoohi-Yekta e Keogh (2015); ten Holt *et al.* (2007)); em particular, duas generalizações diretas são as chamadas DTW_I e DTW_D (onde I vem de independente, e D de dependente).

No primeiro caso, DTW_I, dadas duas matrizes de dados X e Y, obtém-se a DTW entre a primeira linha de X e a primeira de Y, a segunda linha de X e a segunda de Y, e assim sucessivamente; ao final, somam-se as DTW assim obtidas para obter a medida conjunta da matriz. Note que, neste caso, o número de linhas em X e Y deve ser igual.

No segundo caso, DTW_D, define-se uma medida de distância no espaço das linhas de X e Y, tipicamente utilizando-se a distância euclidiana. Em seguida, é calculada uma nova matriz de custos, como no caso unidimensional, onde agora cada entrada da matriz de custo representa a distância entre as respectivas linhas de X e Y (em oposição ao caso unidimensional, onde cada entrada da matriz de custos representa a distância entre componentes individuais dos sinais). Aqui, tanto o número de linhas quanto o número de colunas deve ser igual nas duas matrizes.

No caso do chirpograma, a aplicação da DTW multidimensional conforme estas definições pode ser feita de duas maneiras: uma, assumindo-se que pode haver distorção dos padrões **no eixo da frequência, para um chirp fixo**; a outra, assumindo-se que pode haver distorção dos padrões **no eixo do chirp, para uma frequência fixa**. No primeiro caso, a DTW vai identificar como semelhantes sinais que possuam alta energia em bandas distintas de frequência, mas para o mesmo valor de chirp; no segundo caso, ao contrário, a DTW vai identificar como semelhantes sinais que possuam valores próximos da energia em bandas distintas do chirp, para as mesmas bandas de frequência.

O primeiro caso implica em obter a DTW multidimensional dos chirpogramas originais, C_i e C_j; no segundo caso, obtemos a DTW multidimensional dos chirpogramas transpostos, C_i^T e C_j^T.

Para combinar esses dois valores, propomos utilizar o valor mínimo, $min(dtw(C_i, C_j), dtw(C_i^T, C_j^T))$.

Desta maneira, permitimos que chirpogramas sejam alinhados tanto vertical quanto horizontalmente.

4.3 Categorização de sinais simulados

Para efetuar uma primeira avaliação das diferentes caracterizações para os segmentos, aplicamos as métricas acima num conjunto de sinais simulados. Simulamos 5 categorias distintas de sinal, onde uma categoria é definida pelos valores de ω e α (frequência fundamental e chirp). O modelo tonal será composto de 2 harmônicos, com amplitudes 1 e $0,5$ e fase idêntica.

A cada sinal obtido do modelo tonal, adicionamos ruído Gaussiano de desvio-padrão $\sigma_r = 2$; para simular a situação em que há diferença de fase (ou em que o momento de início do sinal é diferente em cada segmento), simulamos segmentos de duração total de 4 segundos a uma taxa de amostragem de 11.025 Hz; o sinal obtido do modelo tonal, porém, terá duração de apenas 2 segundos. O instante exato de início do modelo tonal na janela de 4 segundos será selecionado aleatoriamente para cada sinal.

A tabela 4.1 apresenta os parâmetros das cinco categorias simuladas. Nas figuras 4.6 a 4.8 vemos o gráfico da amplitude, a DFT média de 1 segundo e o chirpograma de cada sinal (chirpograma calculado para $\alpha \in [0, 20]$ e $\omega \in [10, 500]$, num reticulado uniforme de 100 pontos para cada dimensão).

Categoria	α	ω
1	0	100
2	10	100
3	0	0
4	6	200
5	0	300

Tabela 4.1: Parâmetros de cada categoria para os segmentos simulados

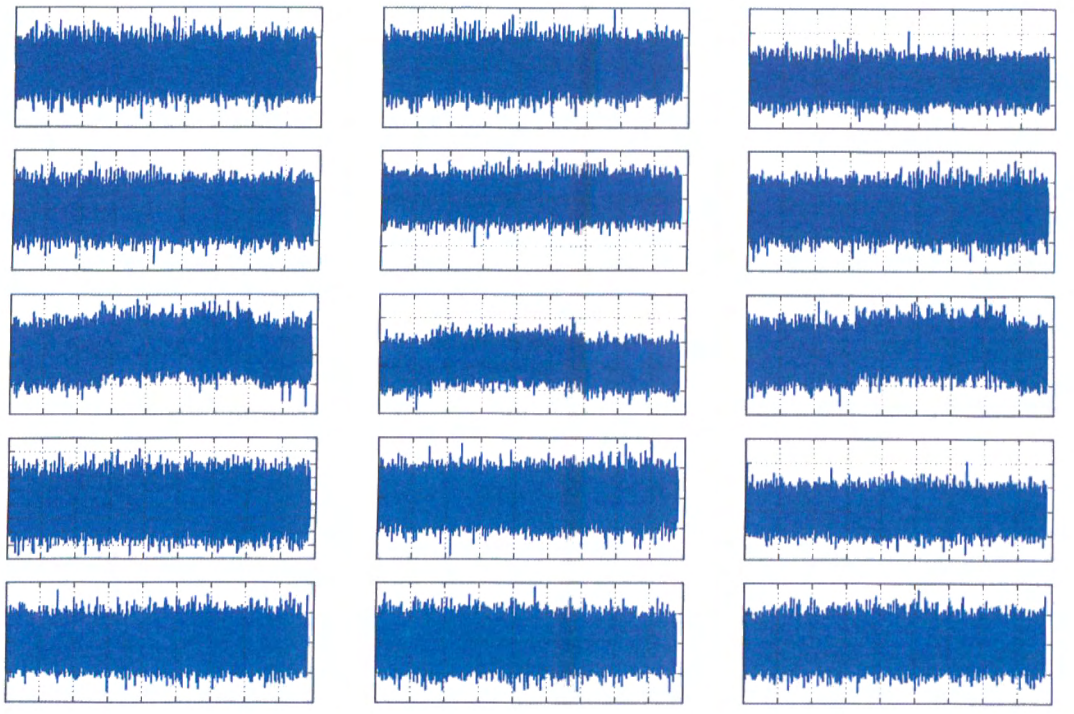


Figura 4.6: Sinais simulados (sinais da mesma categoria na mesma linha)

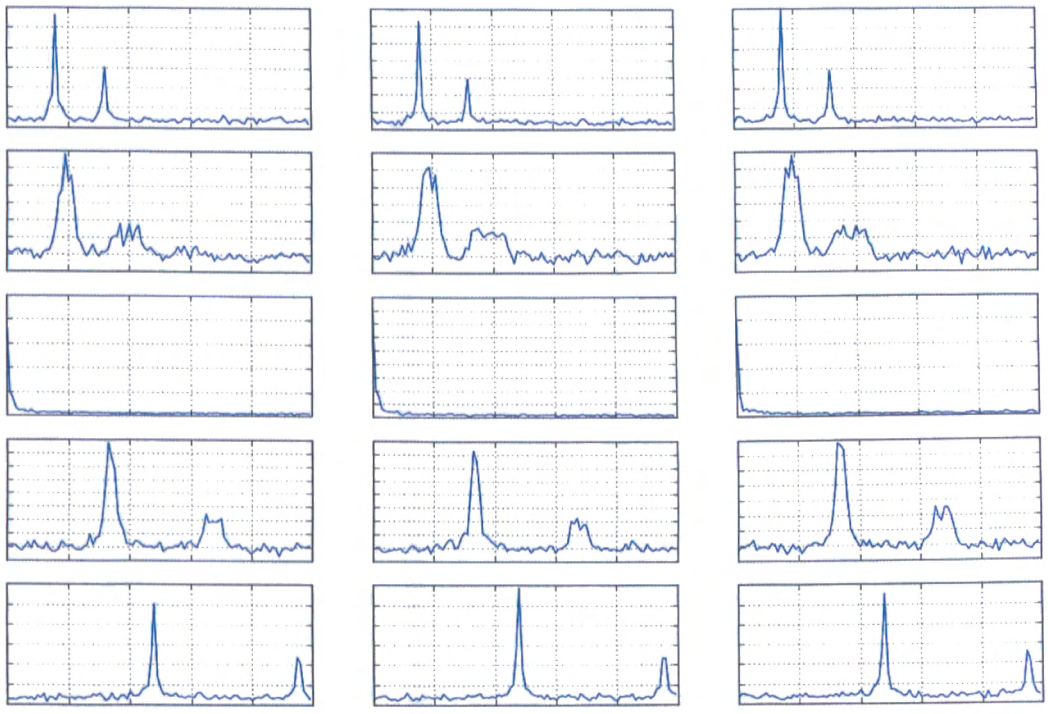


Figura 4.7: DFT média de 1 segundo dos sinais simulados (sinais da mesma categoria na mesma linha)

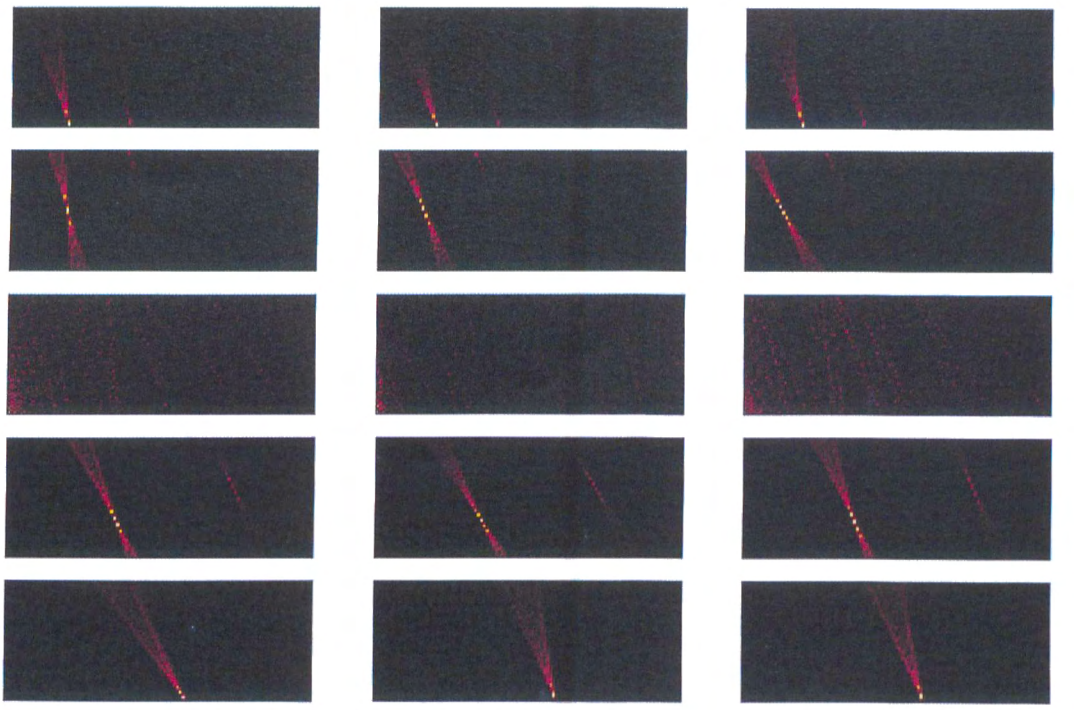


Figura 4.8: Chirpogramas dos sinais simulados (sinais da mesma categoria na mesma linha)

A categoria 3 representa segmentos onde não há nenhum sinal, mas apenas ruído; isto se reflete no espectro médio (energia espalhada uniformemente por todo o espectro, à exceção de um pico correspondente a $\omega = 0$) e no chirpograma (valor espalhados ao longo de todo o espaço paramétrico). As categorias 1 e 5 representam sinais sem chirp; nos espectros, isto se reflete por uma concentração mais alta de energia na faixa mais estreita em torno da frequência fundamental. Nos chirpogramas, a ausência de chirp faz com que os padrões que aparecem na imagem tenham seu ponto de máxima intensidade sobre o eixo $\alpha = 0$.

As categorias 2 e 4, por fim, representam sinais com chirp; esta característica fica evidente principalmente nos chirpogramas, onde o ponto de valor mais alto encontra-se na linha correspondente ao valor do chirp.

Os padrões de forma "gravata borboleta" que observamos nos chirpogramas se devem ao seguinte fato: temos que

$$\omega t + \alpha t^2 = (\omega + \epsilon)t + (\alpha + \delta)t^2 \iff \epsilon t + \delta t^2 = 0 \iff \epsilon = -\delta t$$

Ou seja, sempre que no chirpograma nos deslocamos ϵ para a esquerda (i.e., subtraímos ϵ da frequência), e ao mesmo tempo nos deslocamos δt para cima (i.e., aumentamos o chirp em δt), o valor da função do modelo será o mesmo; logo, surgem os padrões de linhas com inclinação negativa no chirpograma (inclinação dependente de t), que se encontram exatamente em (α, ω) os valores verdadeiros da frequência e do chirp.

Para agrupar os segmentos simulados, vamos obter as matrizes de distância das seguintes maneiras: usando a distância euclidiana entre segmentos, a distância euclidiana entre as DFTs médias, a DTW sobre os segmentos, DTW sobre as DFT médias, distância de Frobenius sobre os chirpogramas, e DTW multidimensional sobre os chirpogramas. No caso da DTW multidimensional,

aplicamos a DTW_I conforme definição da seção anterior, obtendo o mínimo entre $DTW_I(C_i, C_j)$ e $DTW_I(C_i^T, C_j^T)$. Escolhemos a DTW_I em detrimento da DTW_D pois a métrica euclidiana utilizada na definição de DTW_D não faz sentido no contexto da comparação de chirpogramas.

Em resumo, vamos aplicar 6 metodologias distintas para calcular a similaridade entre os segmentos:

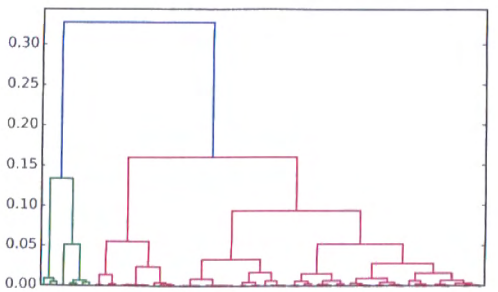
1. Distância euclidiana calculada entre os segmentos (DESeg);
2. Distância euclidiana entre as DFTs médias de 1 segundo (DEDFT);
3. DTW entre os segmentos (DTWSeg);
4. DTW entre as DFTs médias (DTWDFT);
5. Distância de Frobenius sobre os chirpogramas (FROChirp);
6. DTW multidimensional sobre os chirpogramas (DTWChirp).

Uma vez tendo obtido as matrizes de distância, aplicamos uma metodologia de agrupamento hierárquico a cada matriz. A utilização do método hierárquico de agrupamento se deve a dois fatores: primeiro, ao fato de que esses métodos permitem a construção de agrupamentos com base numa matriz de distâncias pré-calculada. O segundo pela possibilidade de obter uma visualização do processo de agrupamento, o dendrograma. Os dendrogramas dos resultados aparecem na figura 4.9.

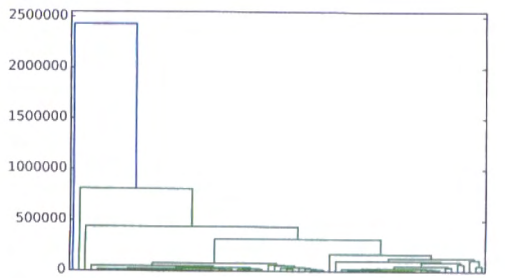
4.4 Categorização de segmentos do OceanPod

Replicamos o exercício da seção anterior para os segmentos de dois arquivos do OceanPod: a amostra de 08/02/2015 e a amostra de 22/02/2015. Ambas totalizam 68 segmentos; as DFTs médias e os chirpogramas desses segmentos são os que aparecem na seção 4.1.

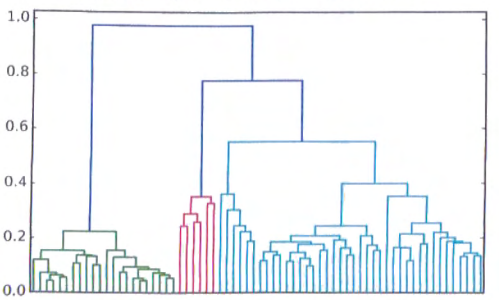
Nos dados reais do OceanPod, não faz sentido aplicar a distância euclidiana entre segmentos, pois estes tem durações distintas. Sendo assim, testamos os outros 5 métodos disponíveis. Os dendrogramas obtidos estão na figura 4.11; eles foram construídos adotando-se a medida do vizinho mais distante para calcular distância entre grupos.



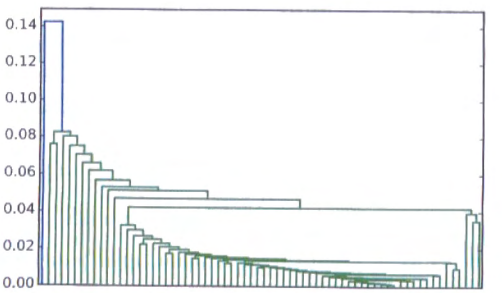
(a) Agrupamento por DEDFT



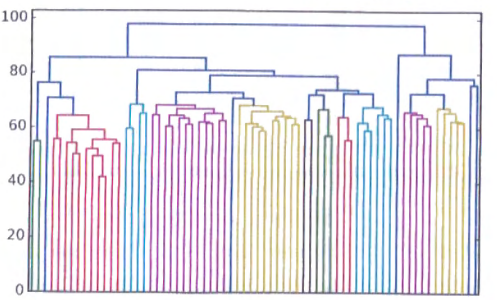
(b) Agrupamento por DTWSeg



(c) Agrupamento por DTWDFT



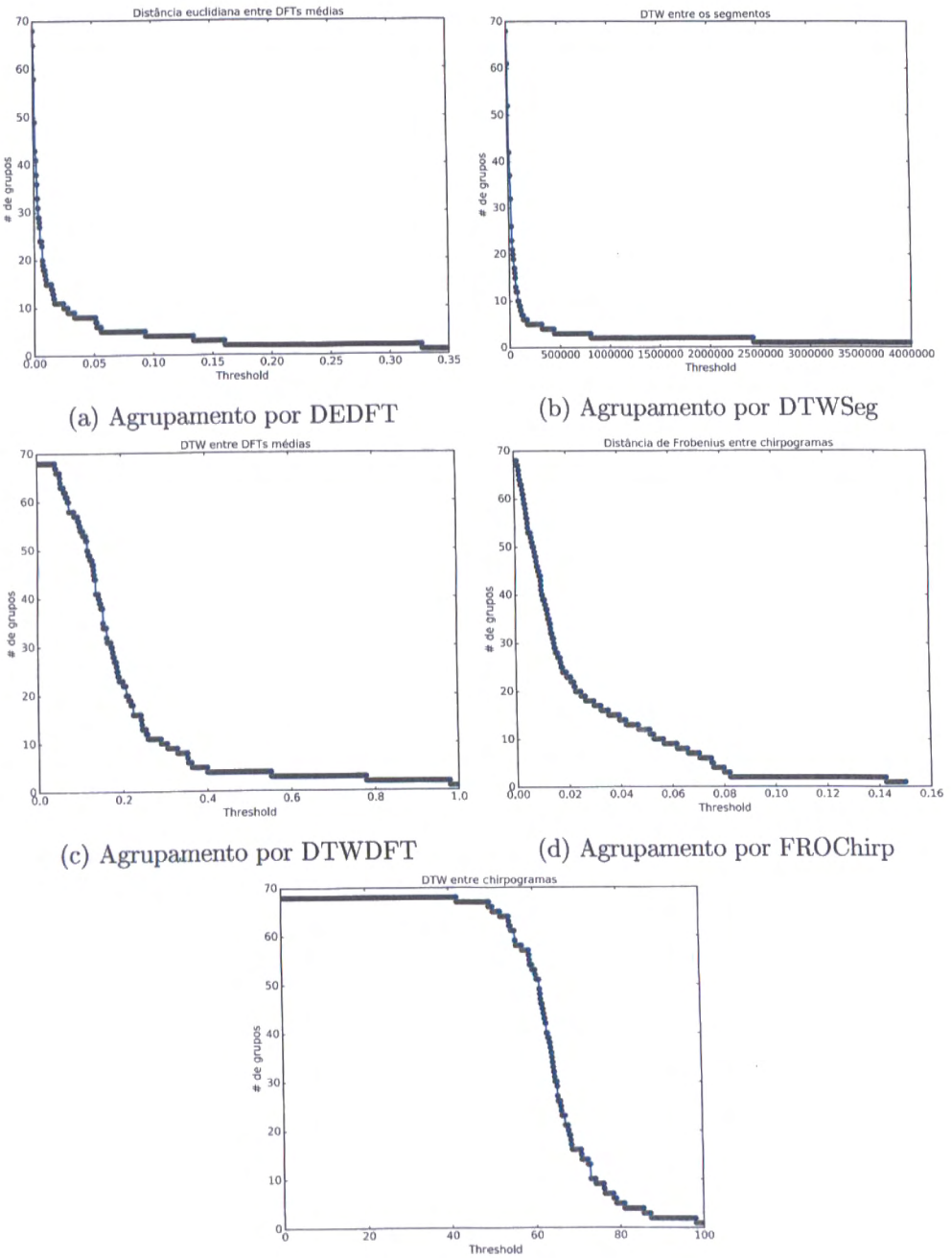
(d) Agrupamento por FROChirp



(e) Agrupamento por DTWChirp

Figura 4.11: Dendrogramas: sinais do OceanPod

Uma vez obtido o dendrograma, para efetivamente agrupar os sinais em categorias é necessário escolher um valor de corte para as distâncias (o eixo vertical do dendrograma). Exibimos na figura 4.12 o número de categorias para diferentes valores do ponto de corte, para cada dendrograma.



(a) Agrupamento por DEDFT

(b) Agrupamento por DTWSeg

(c) Agrupamento por DTWDFT

(d) Agrupamento por FROChirp

(e) Agrupamento por DTWChirp

Figura 4.12: Número de categorias como função do ponto de corte

O método de agrupamento hierárquico funciona a partir de uma ótica *bottom-up*, começando com os indivíduos completamente separados, e agrupando sucessivamente os indivíduos mais próximos. É possível, neste método, que dois pares de indivíduos (ou de categorias) possuam distâncias iguais ou muito próximas; isto significa que o algoritmo hierárquico vai agrupar simultaneamente esses dois pares, de modo que o número total de categorias diminui de k para $k - 2$. Por esta razão identificamos algumas descontinuidades nos gráficos acima, e também no gráfico de silhuetas, já que o método hierárquico não é capaz de gerar agrupamentos com todos os números possíveis de categorias.

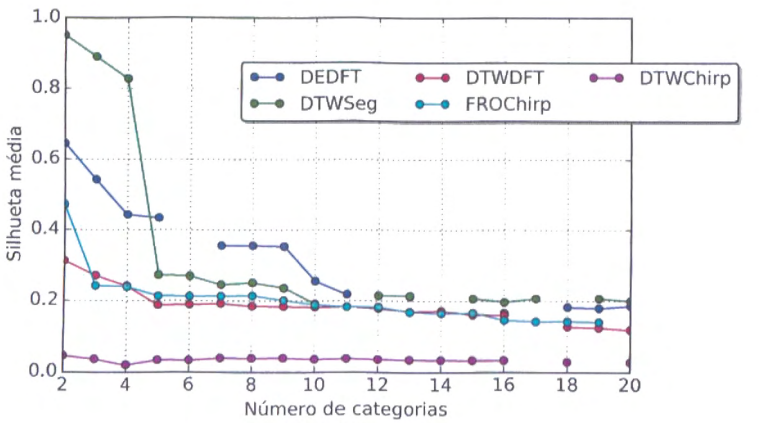


Figura 4.13: Silhuetas médias para os segmentos do OceanPod

Analisando os resultados da silhueta média, vemos que o método de DTW sobre os segmentos teve valores mais altos para $k = 2, 3, 4$, com uma queda brusca para mais do que 4 categorias. Com 5 categorias ou mais, as silhuetas de todos os métodos estão próximas, à exceção de DEDFT com valores mais elevados principalmente para $k = 5, 7, 8, 9$. Além disso, chama a atenção o fato de que o método DTWChirp, que apresentou desempenho ótimo no teste com dados simulados, agora tem valores de silhueta próximos de 0. Na figura 4.14 vemos as silhuetas médias para o método DTWChirp; podemos identificar os valores de $k = 7$ e $k = 9$ como os números ótimos de categorias para este método de agrupamento.

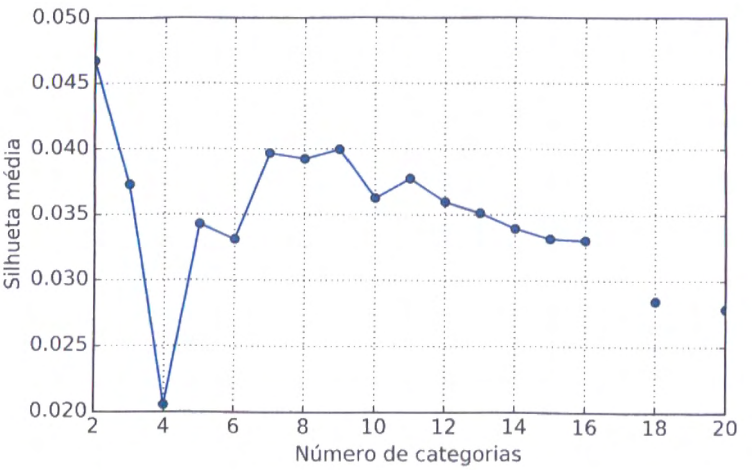


Figura 4.14: Silhuetas médias para os segmentos do OceanPod: DTWChirp

4.4.1 Agrupamento por k-médias e k-medóides

O método de agrupamento hierárquico, utilizado na construção dos dendrogramas, é interessante apenas como uma maneira de estudar as propriedades da matriz de distâncias pré-computada (pois ele é completamente determinado por tal matriz). Para estimar o número correto de categorias, e também para observar melhor a eficiência do agrupamento com as diferentes métricas, é recomendável recorrer a outros métodos.

Um método de agrupamento comum na literatura é o método das *k-médias*. Neste método as categorias são formadas iterativamente; a partir de um agrupamento inicial aleatório, calcula-se a distância de cada indivíduo ao centro de cada categoria (definido como o ponto médio de todos os membros da categoria). A partir dessa distância, os indivíduos são realocados à categoria mais próxima, as médias são recalculadas, e assim sucessivamente, até a convergência.

O método das k-médias é equivalente a um modelo de mistura de Gaussianas, sob a suposição

de isotropia das categorias (i.e., sob a suposição de diagonalidade para as matrizes de covariância das Gaussianas envolvidas na mistura Hastie *et al.* (2001)). Sendo assim, ele funciona bem quando as categorias podem ser bem aproximadas por hiperesferas no espaço das características.

Uma vez que esse método utiliza médias aritméticas simples para construir os centros (ou centróides) das categorias, a métrica indicada para avaliar distâncias entre indivíduos é a euclidiana (como se sabe, minimizar a distância euclidiana equivale à maximizar a verossimilhança de um modelo Gaussiano). Sendo assim, aplicamos o método das k-médias à categorização dos segmentos caracterizados pelas DFTs médias, e também para os segmentos caracterizados pelo chirpograma; neste último caso, a utilização do método das k-médias é equivalente à utilização da norma de Frobenius para cálculo da distância entre os chirpogramas.

Para os métodos baseados na DTW, aplicamos o método de categorização dos *k-medóides*; este método é similar ao método das k-médias, mas ao invés de utilizar os pontos médios de cada agrupamento como os centros da categoria (os centróides), um indivíduo representativo é escolhido como o centro da categoria (o medóide). O algoritmo de agrupamento por k-medóides mais utilizado é o PAM (*Partition Around Methods*). Para uma definição detalhada do método PAM, bem como para uma comparação de diferentes métodos de agrupamento, ver por exemplo Hastie *et al.* (2001); Reynolds *et al.* (1992).

Aplicamos portanto o método das k-médias para agrupamento de DFTs médias e chirpogramas, e o método das k-medóides para agrupamento baseado no DTW dos segmentos, no DTW das DFTs médias, e no DTW dos chirpogramas. Na figura 4.15 vemos as silhuetas calculadas para as 5 categorizações, utilizando os algoritmos de k-médias e k-medóides, conforme o caso.

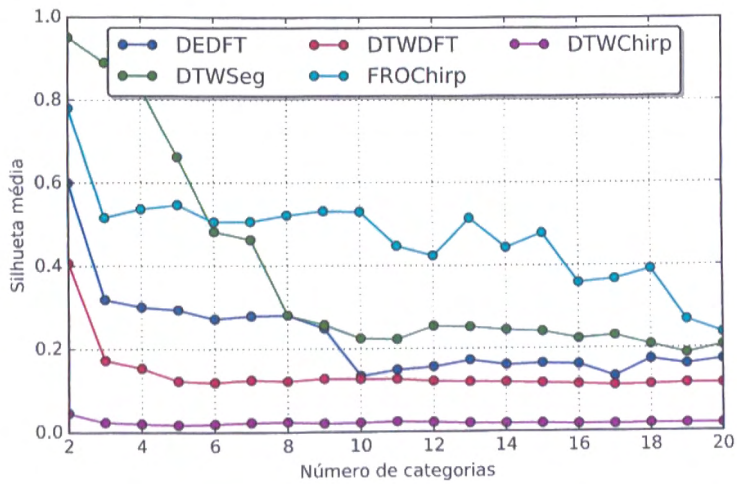


Figura 4.15: Silhuetas médias para os segmentos do OceanPod

O primeiro ponto a se observar é que, para todos os algoritmos e todas as métricas, a melhor silhueta média é obtida quando $k = 2$; isto indica que a divisão do grupo de segmentos em duas classes apresentou, em tese, um bom poder de separação.

Porém, analisando as categorizações para as métricas DTWSEG e FROChirp, temos que quando $k = 2$ apenas um segmento é deslocado para a segunda categoria, permanecendo os demais todos numa categoria só. Para estas duas métricas, o mesmo acontece quando aumentamos o número de categorias: os resultados são sempre desequilibrados, com um grande número de segmentos pertencendo a uma categoria, e diversas categorias com apenas um segmento. Isto explica em parte os valores mais altos da silhueta média para essas duas métricas.

Este desbalanceamento também ocorre com a métrica DEDFT, embora de maneira um pouco menos extrema. Com as métricas DTWDFT e DTWChirp, as categorias resultantes são mais balanceadas, o que indica que estes dois métodos, ambos baseados no algoritmo de k-medóides, podem apresentar melhores resultados.

Por outro lado, o fato destas duas métricas terem produzido agrupamentos com valores de

silhueta próximos de 0 é um indicativo de que pode haver sobreposição entre as categorias; ou seja, os métodos de *hard clustering*, que atribuem uma e apenas uma categoria a cada segmento, podem não ser os mais indicados nesse caso.

Para analisar mais profundamente os resultados dos agrupamentos usando essas duas métricas, vamos avaliar diretamente os medóides resultantes de cada uma. Esta é uma vantagem do algoritmo de k-medóides: como os centros de cada categoria são indivíduos do banco de dados, podermos observar diretamente esses indivíduos para verificar se, de fato, eles possuem características distintas.

Nas figuras 4.16 e 4.17 vemos os espectros médios escolhidos como medóides no caso $k = 2$ e $k = 3$, respectivamente.

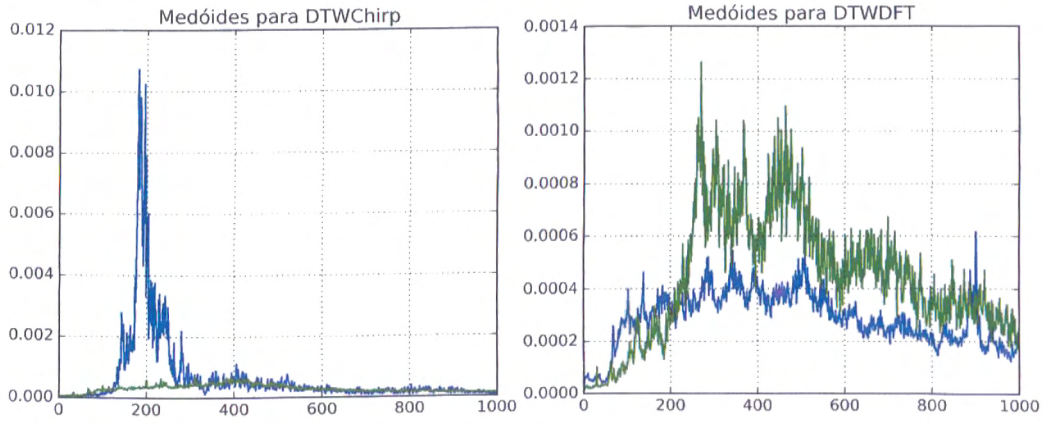


Figura 4.16: Medóides para $k = 2$

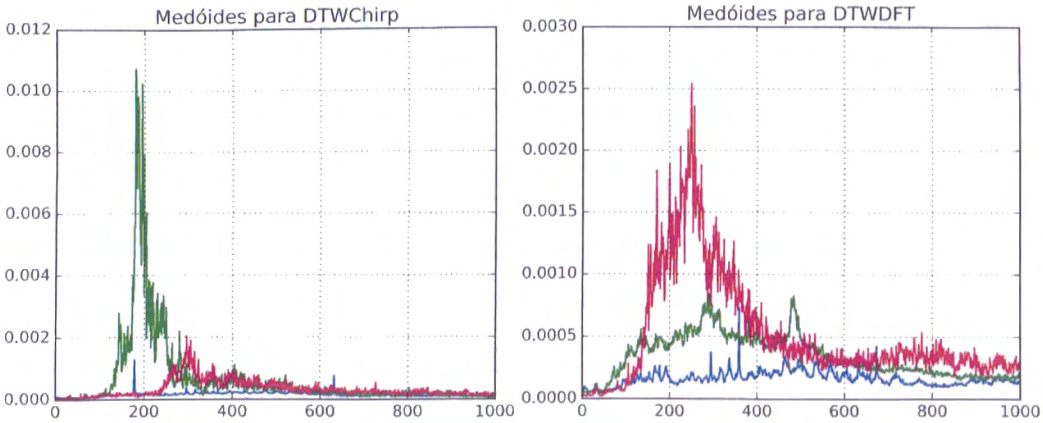


Figura 4.17: Medóides para $k = 3$

Como podemos ver nas figuras, o algoritmo de k-medóides utilizando DTWChirp escolheu como medóides indivíduos com espectros médios bastante distintos (ainda mais distintos do que no caso da DTWDFT), tanto no caso $k = 2$ quanto no caso $k = 3$; para as duas medidas, porém, a escolha de medóides parece ter capturado efetivamente diferenças entre os segmentos. Escutando diretamente os segmentos de áudio representados na figura, verificamos que de fato os dois métodos separaram como medóides segmentos de características bem distintas, principalmente no caso da DTWChirp, onde um segmento contém apenas ruído de fundo, e o outro contém um exemplo bastante evidente de ruído de embarcação próxima ao hidrofone.

Por fim, para uma última comparação entre os métodos de categorização, podemos recorrer à inspeção direta dos 68 segmentos neste conjunto de dados, para identificar manualmente (auditivamente) as categorias adequadas.

Categoria	Segmentos
motor distante	1, 2, 3, 4, 5, 10, 18, 19, 20, 29, 30, 58, 59
motor tipo 1	7, 8, 9, 16, 17, 37, 39, 41, 42, 43, 44, 45, 46, 47, 48, 50, 51, 52, 54, 56
motor tipo 2	11, 13, 14, 25, 27, 40
motor tipo 3	22, 23, 26
motor tipo 4	31, 32, 33, 34, 35, 36
motor tipo 5	60, 61, 62, 63, 65
ruído	0, 6, 12, 15, 21, 24, 28, 38, 49, 53, 55, 57, 64, 66, 67

Tabela 4.2: Categorização por audição

Escutar os segmentos mostra por que a tarefa de categorização (assim como a tarefa anterior de segmentação) não é trivial; não é simples para o ouvido humano destreinado (no reconhecimento de sons subaquáticos e na assinatura acústica de motores) identificar quais segmentos deveriam pertencer a um mesmo grupo, e existem diferenças mais ou menos sutis mesmo entre os segmentos que escolhemos classificar na mesma categoria. Entretanto, o exercício é importante pois no mínimo nos permite avaliar se os algoritmos de categorização são capazes de separar segmentos contendo apenas ruído, de segmentos que contenham outros eventos, quaisquer que sejam eles.

Com esse fim, na tabela 4.2 apresentamos as categorias escolhidas a partir da audição dos 68 segmentos.

A categorização por audição revelou 7 categorias nos segmentos. O número é próximo do valor que produz a silhueta ótima para o DTWChirp, conforme figura 4.18 (e ignorando-se o caso $k = 2$).

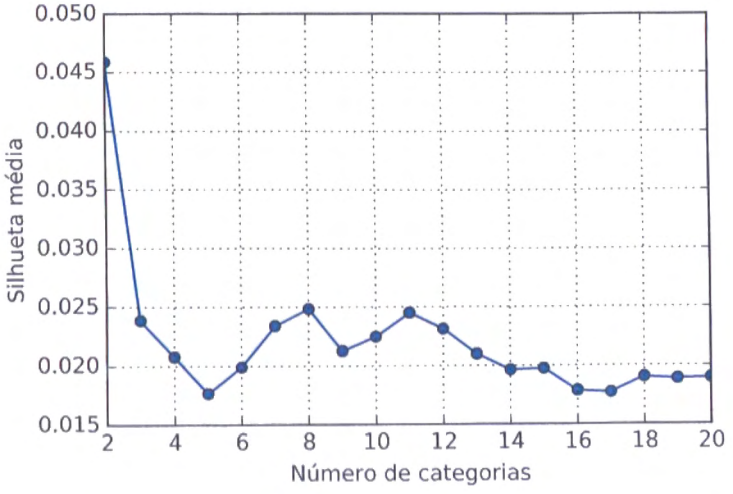


Figura 4.18: Silhuetas médias para os segmentos do OceanPod, DTWChirp

Para comparar as categorizações por DTWDFT e DTWChirp com as categorias manuais, em primeiro lugar calculamos o número distinto de categorias atribuídas ao grupo "ruído" da categorização manual. Quanto maior esse número, maior a fragmentação do grupo de segmentos que contém apenas ruído.

Também podemos calcular, para cada diferente valor de k, o *índice Rand* comparando a categorização manual com a obtida do algoritmo.

O índice Rand é calculado da seguinte maneira: sejam duas categorizações C_1 e C_2 do mesmo conjunto de objetos; definimos

- a: o número de pares de elementos que estão na mesma categoria em C_1 e na mesma categoria em C_2;

- b: o número de pares de elementos que estão em categorias distintas em C_1 e em categorias distintas em C_2;
- c: o número de pares de elementos que estão na mesma categoria em C_1 e em categorias distintas em C_2;
- d: o número de pares de elementos que estão em categorias distintas em C_1 e na mesma categoria em C_2;

Com estas definições, o índice é calculado como

$$R = \frac{a+b}{a+b+c+d}$$

O índice de Rand foi proposto pela primeira vez em Rand (1971). Ele varia entre 0 (nenhuma concordância entre as categorizações) e 1 (concordância perfeita).

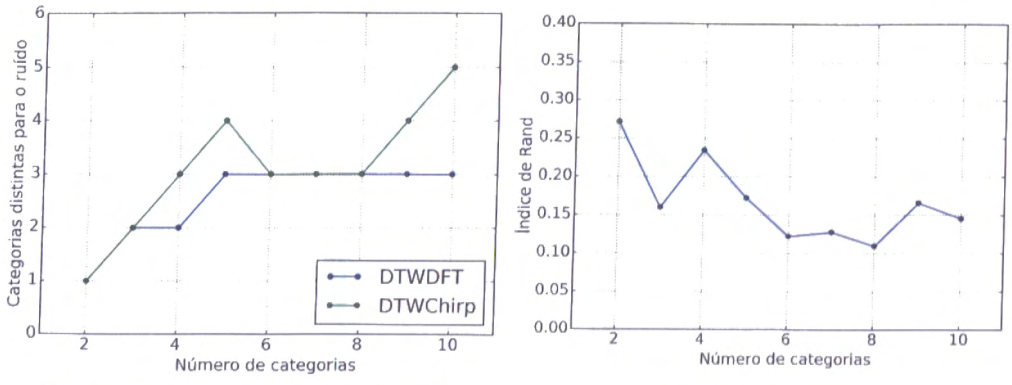
Este índice, porém, não leva em consideração o fato de que mesmo duas categorizações aleatórias devem apresentar uma certa proporção de pares coincidentes; para corrigir essa distorção, Hubert e Arabie (1985) propuseram o índice de Rand ajustado. O ajuste é feito calculando-se o índice de Rand esperado para uma categorização completamente aleatória, R_e, assumindo-se um modelo hipergeométrico generalizado para o número de individuos em cada categoria (i.e., as categorizações são selecionadas aleatoriamente mas mantendo-se fixo o número de elementos em cada categoria). Em seguida, calcula-se o índice de Rand máximo (ainda mantendo fixos os números de elementos em cada categoria), R_{max}, e por fim o índice de Rand ajustado é definido por

$$R_{adj} = \frac{R - R_e}{R_{max} - R_e} \tag{4.4}$$

O índice de Rand ajustado tem valor 0 quando o índice de Rand da categorização é exatamente igual ao valor esperado do índice; pode assumir valores negativos, caso $R < R_e$, e tem valor máximo igual a 1 quando $R = R_{max}$.

Para a categorização por DTWDFT com $k = 7$, o índice de Rand ajustado calculado na comparação com a categorização manual é de 0,13; no caso da medida DTWChirp, o índice é de 0,12. Os resultados, portanto, são bastante semelhantes utilizando essas duas medidas, e mostram uma categorização de baixa sobreposição com a categorização manual.

Na figura 4.19 temos, no lado esquerdo, o número de categorias distintas para o grupo "ruído" da categorização manual; no lado direito temos os índices de Rand ajustado comparando as categorizações por DTWDFT e DTWChirp, para os diferentes números de categorias.



(a) Fragmentação do grupo 'ruído'

(b) Índice Rand

Figura 4.19: Comparação DTWDFT e DTWChirp

O método das k-medóides baseado na medida DTWDFT fragmentou menos o grupo de "ruído": a partir de $k = 5$, não há fragmentação subsequente deste grupo. Já a medida DTWChirp continuou a fragmentar o grupo de "ruído" para $k > 8$.

O índice de Rand máximo obtido na comparação entre os agrupamentos por DTWDFT e DTW-Chirp foi obtido no caso $k = 2$, quando ficou em torno de $0,27$; quando $k = 7$, o índice de Rand ajustado foi igual a $0,13$, indicativo de uma sobreposição fraca entre os dois agrupamentos.

Estes resultados indicam que há diferenças entre a utilização da DTW sobre as DFTs médias e sobre os chirpogramas; por outro lado, na comparação com a categorização manual, as duas métricas apresentaram resultados bastante semelhantes. Os valores relativamente baixos dos índices de Rand ajustados indicam que nenhum dos dois métodos concordou inteiramente com a categorização manual. Como mencionamos acima, porém, a categorização manual não pode ser tomada como a verdade definitiva para esses dados, e a melhor prova para a categorização é a audição dos segmentos. Este teste, que infelizmente não pode ser incluído no texto, mostra que a categorização por DTW sobre as DFTs médias foi particularmente capaz de produzir categorias de segmentos bastante semelhantes entre si. As diferenças entre a categorização manual e a algorítmica se dão principalmente em segmentos em que o evento acústico é de baixa energia, ou quando a segmentação não é perfeita (de modo que há por exemplo alguma extensão de ruído num segmento que também capturou algum evento significativo porém de curta duração).

4.5 Conclusão

Apresentamos neste capítulo diferentes maneiras de categorizar os segmentos de sinal obtidos do algoritmo de segmentação do capítulo anterior.

Esses métodos podem trabalhar diretamente com o segmento (caso da distância euclidiana, DESeg, e do DTW, DTWSeg), com as DFTs médias de 1 segundo (DEDFT e DTWDFT), e com os chirpogramas obtidos num reticulado pré-definido (FROChirp e DTWChirp).

Aplicamos dois algoritmos diferentes de agrupamento, as k-médias e k-medóides, às diferentes combinações de caracterização e cálculo de dissimilaridade entre os segmentos. Além disso, analisamos os dendrogramas obtidos da categorização de sinais simulados.

A aplicação dos métodos aos sinais simulados mostrou que a medida DTW sobre os chirpogramas tem propriedades interessantes, especialmente se o reticulado onde ela é calculada representa uma banda suficientemente ampla de frequência e chirp (no caso dos sinais simulados, sabíamos exatamente os valores de frequência e chirp de cada categoria, o que nos permitiu definir o reticulado de maneira ótima).

Por outro lado, quando categorizamos um conjunto de segmentos obtidos dos dados do OceanPod para o Parque da Laje, os resultados utilizando DTWChirp não foram tão eficientes; neste caso, o DTW aplicado aos DFTs médios de 1 segundo mostrou resultados superiores.

Uma parcela dessa diferença de desempenho do método DTWChirp pode ser devida às condições de obtenção do chirpograma; nos dados reais do OceanPod a banda de frequência dos eventos é desconhecida a priori, e bastante ampla se considerarmos todos os possíveis eventos acústicos que podem ser em tese capturados pelo hidrofone. Sendo assim, para utilização deste método se faz necessária a obtenção do chirpograma em um reticulado de banda maior; neste caso, para manter a resolução adequada, será preciso avaliar mais vezes a função que retorna o chirpograma para valores de (α, ω) fixo, e isto terá um impacto computacional importante.

Sendo assim, pode ser interessante aplicar também um método mais barato computacionalmente; neste caso, o método que mostrou resultados melhores foi o de DTW aplicado sobre as DFTs médias.

Por fim, verificou-se que há indícios de que os métodos de *hard clustering* aplicados podem não ser ideais. Um caminho para trabalhos futuros envolve a investigação de métodos de categorização *soft*, como os modelos de misturas de Gaussianas, por exemplo.

Enfim, no próximo e último capítulo da tese, avaliamos as metodologias de segmentação e categorização de sinais aplicadas a um sinal de longa duração obtido do OceanPod.

Capítulo 5

Aplicação: análise do sinal do Parque Estadual da Laje de Santos

Neste capítulo, aplicamos os métodos de segmentação e categorização de segmentos à análise de um período mais longo do sinal obtido do OceanPod no Parque Estadual da Laje de Santos.

A amostra que vamos analisar corresponde a um período total de 31 dias de gravação contínua, entre 30/01/2015 e 01/03/2015, inclusive. O sinal está dividido em 2.963 arquivos de 15 minutos cada.

Os parâmetros adotados no algoritmo de segmentação encontram-se na tabela 5.1.

Descrição	Parâmetro	Valor
Tamanho mínimo de segmento	minlen	11.025
Resolução temporal	tres	11.025
Iterações MCMC	mciter	20.000
Queima MCMC	mcburn	20.000
Ponto de corte para função de decisão	α	0,100000
Valor inicial para β	β_0	0,000010
Incremento no valor de β	Δ_β	0,000001
Valor máximo para β	β_{max}	0,000100
Número de segmentações com resultado igual	N_{min}	6

Tabela 5.1: Parâmetros utilizados na análise

Para o algoritmo de segmentação, adotamos um passo extra, que consiste no teste da diferença de variância entre o último segmento de um dado arquivo, e o primeiro segmento do arquivo subsequente. Este passo é necessário, uma vez que a segmentação é feita sobre os arquivos de 15 minutos, separados arbitrariamente.

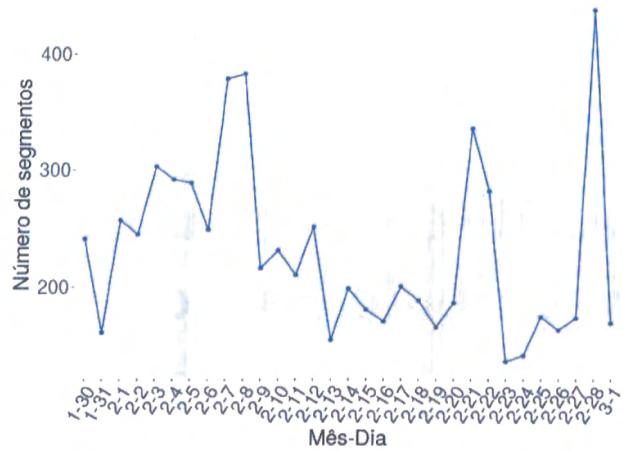
Para o passo de categorização, rodamos dois algoritmos: a DEDFT, baseado na distância euclidiana entre DFTs médias de 1 segundo, e o DTWChirp, baseado no DTW dos chirpogramas. Os chirpogramas foram calculados num grid de 100×100 pontos, para o valor do coeficiente de chirp entre 0 e 20, e o valor da frequência fundamental entre 0 e 500 Hz.

5.1 Resultados da segmentação

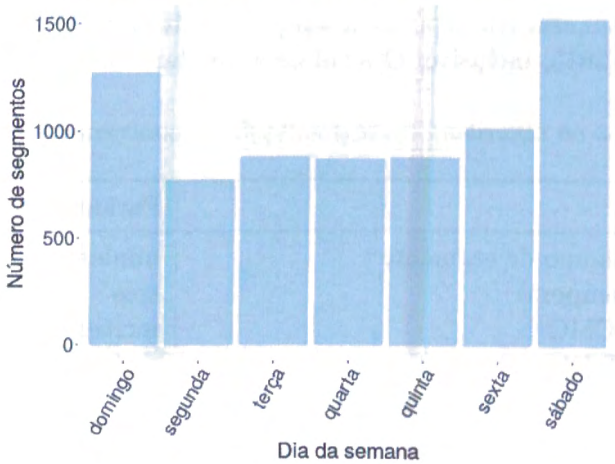
A aplicação do algoritmo de segmentação resultou em 7161 segmentos, com duração média de 372,1 segundos e desvio-padrão de 308,3 segundos. A duração mínima foi de 1 segundo, e a máxima de 66,1 minutos.

O número de segmentos encontrados em finais de semana foi mais elevado do que durante a semana; isso era esperado, pois sabemos que nos finais de semana ocorrem expedições de mergulho

na região do parque. Logo, há diversas embarcações a motor passando perto do hidrofone nesses dias, e o algoritmo de segmentação identificou os eventos correspondentes.



(a) Número de segmentos ao longo do tempo



(b) Número de segmentos por dia da semana

O histograma do número de segmentos por hora de início mostra uma concentração de segmentos entre 12 e 15 horas, o que novamente é compatível com a atividade intensa de embarcações de mergulho.

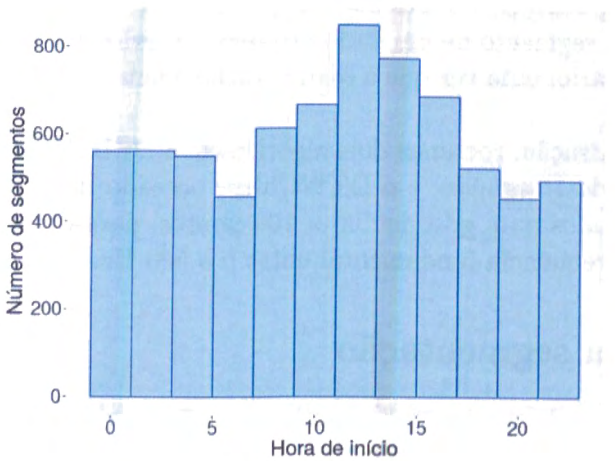


Figura 5.2: Número de segmentos por hora de início

Apesar da média de duração dos segmentos de 285 segundos, os eventos mais curtos são também

mais frequentes, conforme o histograma abaixo (o histograma exclui os segmentos com duração de 15 minutos).

Um total de 1100 segmentos com duração igual a 15 minutos foram encontrados; estes segmentos representam arquivos onde não houve segmentação, mas que não foram agrupados nem com o último segmento do arquivo anterior, nem com o primeiro do arquivo posterior.

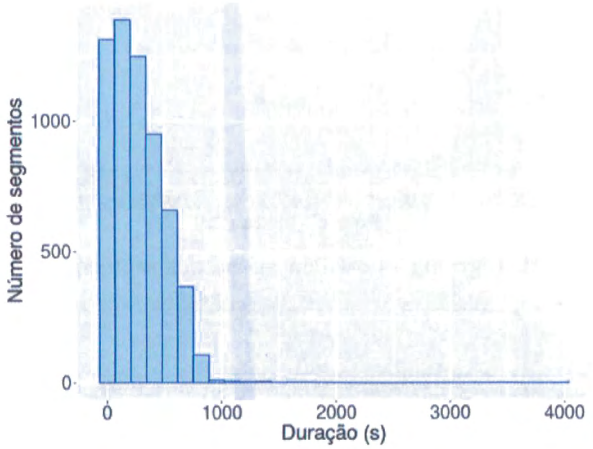


Figura 5.3: Histograma - duração dos segmentos

Os eventos que ocorrem no meio do dia (entre 11 e 17 horas principalmente) tem uma duração média menor, conforme os gráficos de *boxplot* abaixo. Esta observação, em conjunto com as anteriores, aponta para a seguinte situação: encontramos um número grande de segmentos aos finais de semana; dentre estes, encontramos muitos segmentos de curta duração, ocorrendo tipicamente entre 10 e 17 horas. Isto nos leva a crer que muitos dos segmentos produzidos pelo algoritmo vão conter exemplos de ruídos de embarcações, especialmente as utilizadas nas expedições de mergulho.

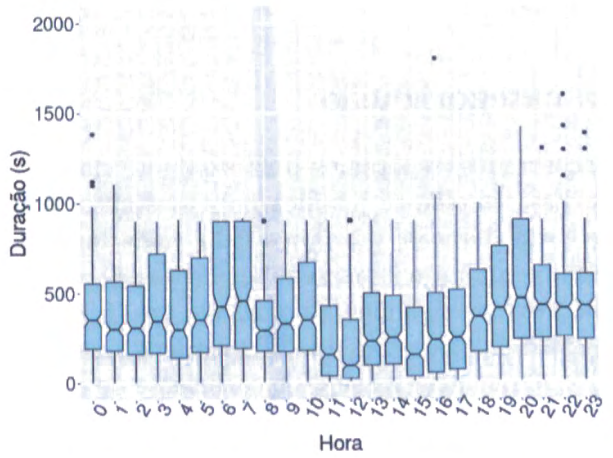
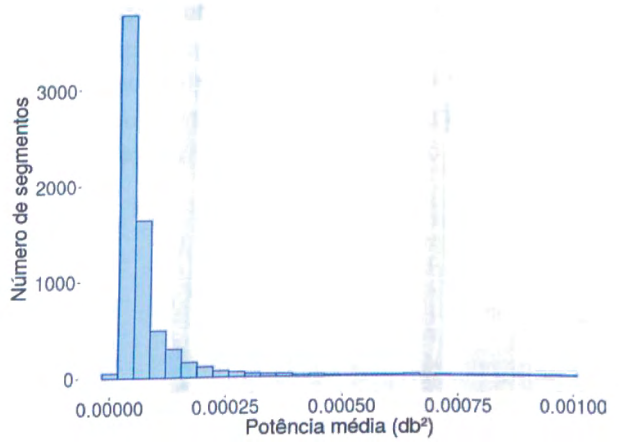
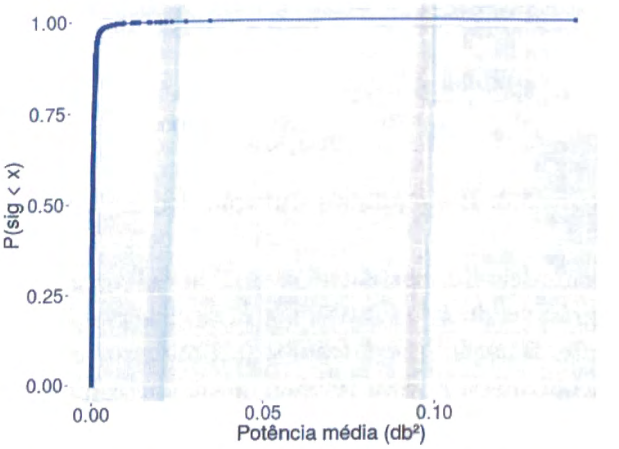


Figura 5.4: Duração dos segmentos por hora de início

Os segmentos obtidos pelo algoritmo SeqSeg podem tanto conter ocorrências de eventos (embarcações, peixes, etc) quanto períodos de silêncio. Nosso modelo inicial assumiu que a diferença fundamental entre essas situações é a potência média dos segmentos. Na figura abaixo, temos o histograma das potências médias obtidas para cada segmento e o gráfico da função distribuição acumulada empírica. Esses gráficos sugerem uma distribuição fortemente concentrada em valores mais baixos de potência, mas com uma cauda bastante grossa. A média da potência dos segmentos foi de $2,3 \times 10^{-4} db^2$, com desvio padrão de $2,0 \times 10^{-3} db^2$; o segmento de mínima potência apresentou um valor de $9,13 \times 10^{-6} db^2$, e o segmento de potência máxima apresentou um valor de $1,4 \times 10^{-1} db^2$.



(a) Histograma - potência média dos segmentos



(b) Função distribuição acumulada - potência média dos segmentos

5.2 Resultados da categorização

Dos 7.161 segmentos encontrados, separamos o subconjunto dos 399 segmentos com duração menor do que 10 segundos. Essa escolha se justifica pois os eventos de maior interesse no banco de dados são os eventos de curta duração; por outro lado, nada impede que outros subconjuntos sejam categorizados, ou mesmo o conjunto inteiro dos segmentos. O custo de processamento, porém, aumenta com segmentos maiores, principalmente no cálculo dos chirpogramas e dos DTW.

Para cada um desses segmentos, obtivemos a DFT média de 1 segundo, e o chirpograma calculado num reticulado de 200×200 pontos, variando o coeficiente de chirp entre 0 e 20 Hz, e o valor da frequência fundamental entre 0 e 1000 Hz.

Aplicamos em seguida o algoritmo de k-medóides, utilizando como medidas de dissimilaridade a DTW das DFTs médias, e a DTW multidimensional dos chirpogramas. As silhuetas médias para um número de categorias entre 2 e 20 aparecem no gráfico 5.6.

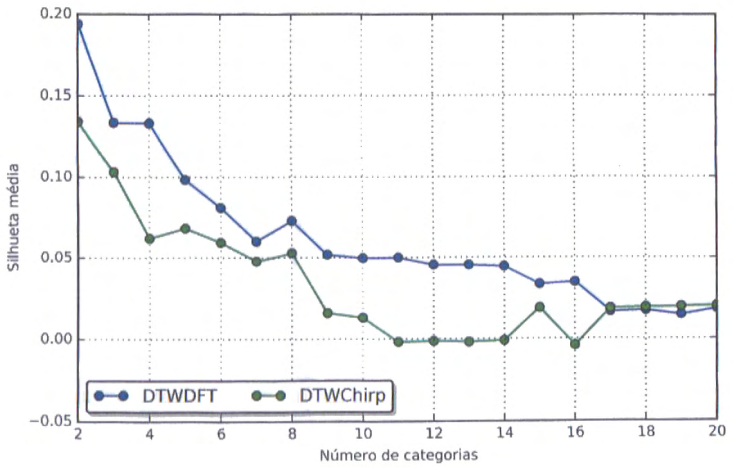


Figura 5.6: Silhuetas médias para os segmentos do Parque da Laje

Novamente, assim como no capítulo anterior, a silhueta média teve valor máximo quando $k = 2$, para os dois métodos, e decresce quase monotonicamente a partir daí. A quebra de monotonicidade ocorre, nos dois casos, entre $k = 7$ e $k = 8$.

O índice de Rand ajustado calculado para a comparação entre as duas categorizações é de $0,54$ para $k = 2$, e cresce com o valor de k. A velocidade de crescimento, porém, diminui após $k = 7$. A figura 5.7 mostra estes valores para k entre 2 e 20.

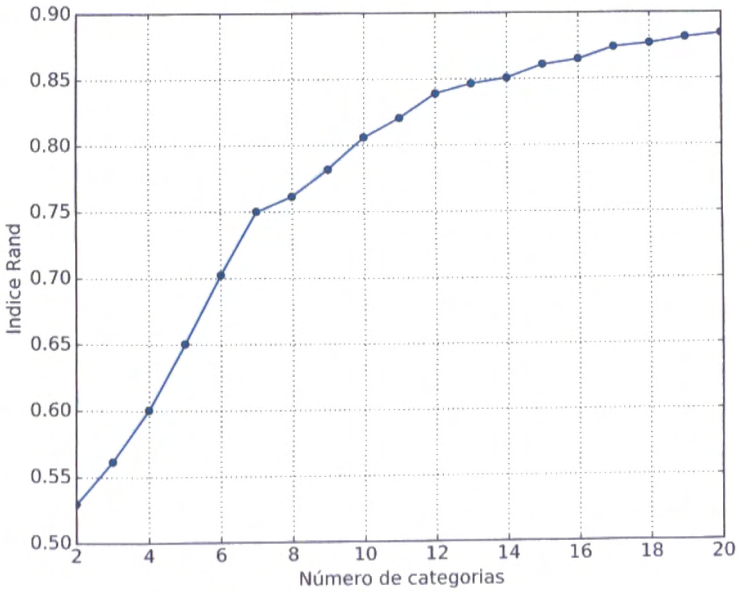


Figura 5.7: Índice de Rand para DTWDFT e DTWChirp

Uma vez que tanto a análise da silhueta média quanto do índice de Rand aponta para uma mudança importante no comportamento dos algoritmos entre $k = 7$ e $k = 8$, fixamos o número de categorias em 8 e obtivemos as categorizações de acordo com os dois métodos.

O algoritmo nos dois casos é o dos k-medóides.

Em primeiro lugar, observamos na tabela 5.2 o número de elementos por categoria, conforme a caracterização utilizada para os segmentos. Na tabela, os rótulos na coluna da esquerda não representam necessariamente a mesma categoria, num e noutro método, mas foram atribuídos sequencialmente, da categoria com mais elementos até a categoria com menos elementos em cada caso.

Categoria	Segmentos (DTWDFT)	Segmentos (DTWChirp)
1	92	72
2	65	72
3	65	69
4	55	61
5	49	42
6	27	34
7	25	25
8	21	24

Tabela 5.2: Número de segmentos por categoria

Nas figuras 5.8 e 5.9 temos as DFTs médias dos medóides de cada categoria, para os dois métodos de caracterização, e nas figuras 5.10 e 5.11 temos os chirpogramas dos medóides de cada categoria, igualmente para os dois métodos de caracterização.

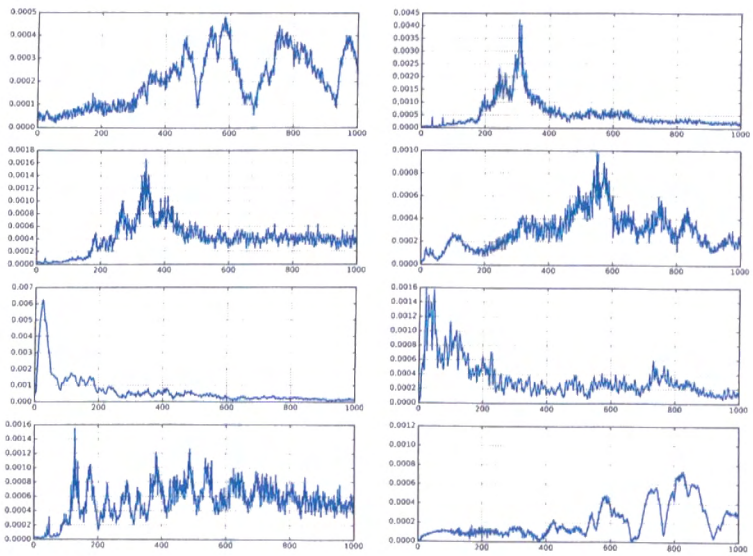


Figura 5.8: DFTs dos medóides, $k = 8$, método DTWDFT

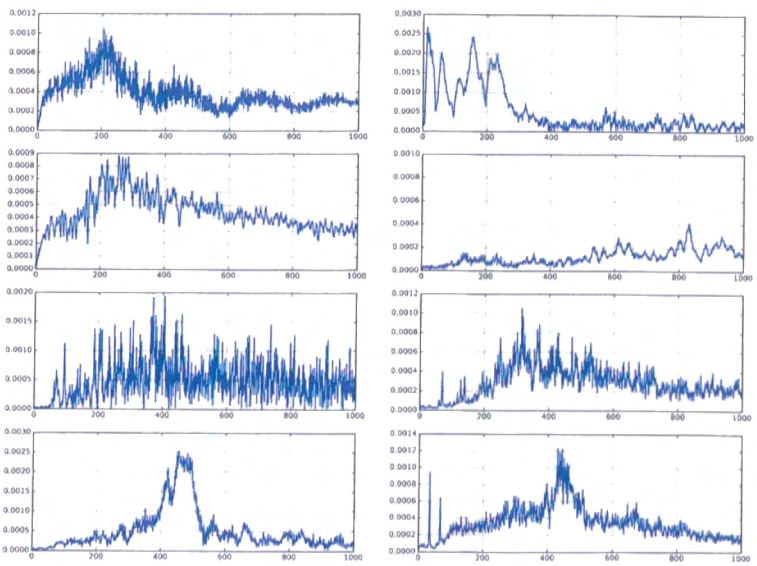


Figura 5.9: DFTs dos medóides, $k = 8$, método DTWChirp

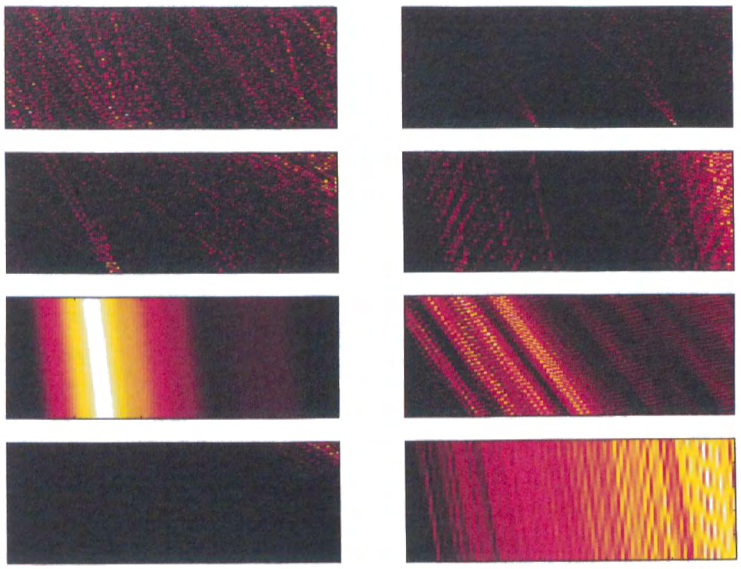


Figura 5.10: Chirpogramas dos medóides, $k = 8$, método DTWDFT

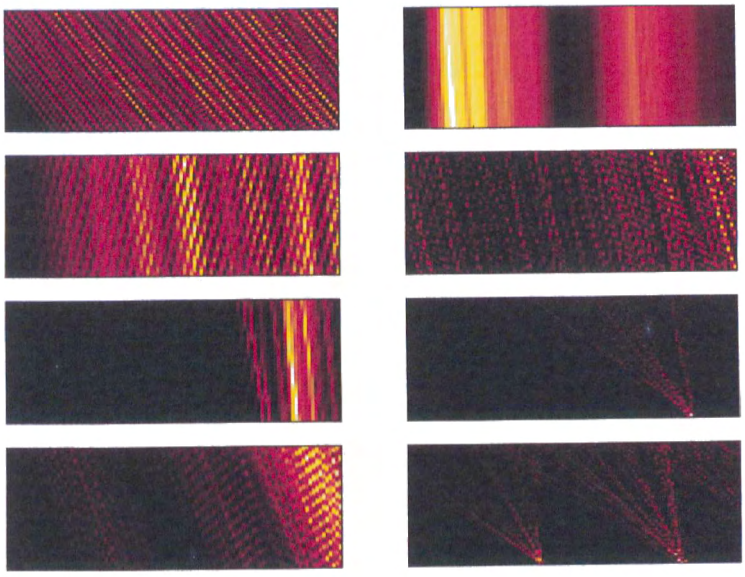


Figura 5.11: Chirpogramas dos medóides, $k = 8$, método DTWChirp

Nos chirpogramas dos medóides de cada método identificamos alguns padrões em comum aos dois algoritmos: o padrão de chirp 0, com dois harmônicos de potência mais alta (o chirpograma da linha superior à esquerda na figura 5.11, e à esquerda na terceira linha na figura 5.10), o padrão de rúido (à direita na segunda linha da figura 5.11, e à esquerda na segunda linha da figura 5.10), o padrão de linhas paralelas com inclinação negativa (direita, quarta linha da figura 5.11, e direita, segunda linha da figura 5.10) e o padrão da barra vertical com frequência fixa (esquerda, segunda linha da figura 5.11 e à direita na quarta linha da figura 5.10).

Ao comparar esses mesmos pares de medóides com respeito aos espectros médios, a semelhança permanece, muito embora a diferença na escala dos espectros torne um pouco mais difícil enxergá-la. Ainda assim, a análise dos medóides parece indicar resultados próximos nos dois algoritmos, o que era de se esperar dado o valor do índice de Rand ajustado, que é igual a $0,76$ quando $k = 8$.

Para analisar a potência e duração dos segmentos em cada categoria e em cada um dos algoritmos de categorização, obtivemos os gráficos de *box-plot* de cada uma das duas variáveis (figuras 5.12 a 5.15).

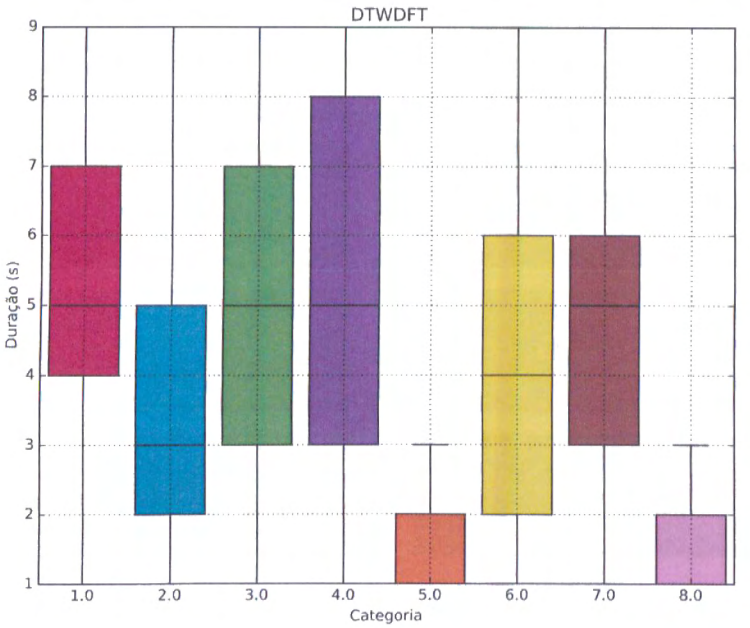


Figura 5.12: Duração dos segmentos por categoria, DTWDFT

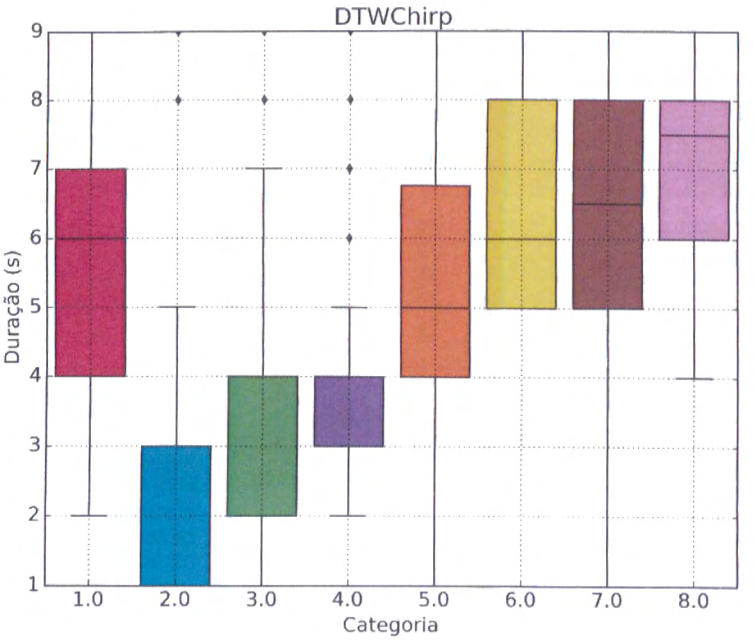


Figura 5.13: Duração dos segmentos por categoria, DTWChirp

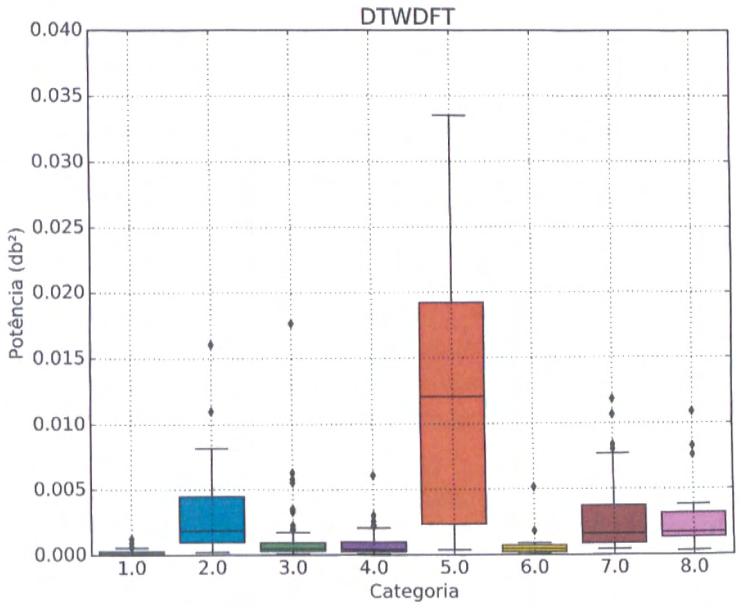


Figura 5.14: Potência dos segmentos por categoria, DTWDFT

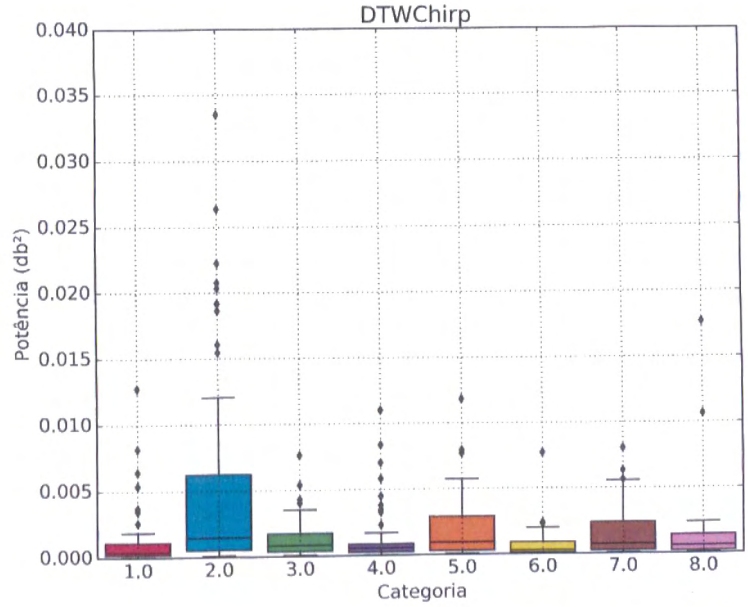


Figura 5.15: Potência dos segmentos por categoria, DTWChirp

Em termos de duração dos segmentos, a DTWDFT produziu duas categorias (5 e 8) com segmentos mais curtos; as demais categorias incluem segmentos com durações menos homogêneas. A DTWChirp, por outro lado, produziu três categorias com segmentos mais curtos (categorias 2, 3 e 4); por outro lado, as três categorias incluem segmentos com a duração máxima de 9 segundos.

Com respeito à potência, a DTWDFT formou uma categoria (categoria 5) que compreende todos os segmentos de maior potência. As categorias 1 e 6 foram as com menor potência média, o que indica que estas categorias devem reunir os segmentos representativos do ruído de fundo. A DTWChirp também separou em uma categoria, a categoria 2, os segmentos de maior potência. As categorias 4 e 6 têm a menor potência média, muito embora também contenham segmentos com potência mais alta.

Ao final, uma vez que o perfil das categorias obtidas de acordo com a caracterização de segmentos por DFTs médias (espectros) e por chirpogramas não é o mesmo, concluímos que as duas

caracterizações de fato capturam propriedades distintas dos segmentos. Num caso, essas propriedades estão ligadas apenas ao espectro do sinal; no outro, a variação desse espectro ao longo do tempo também é levada em conta.

Sendo assim, a escolha por um ou outro método depende do tipo de evento acústico que se pretende separar. Se o objetivo é construir categorias homogêneas à audição, o uso das DFTs médias se mostra mais adequado, conforme se pode comprovar pela escuta dos segmentos categorizados por esta métrica. A categorização pela análise dos chirpogramas, por outro lado, privilegia não o perfil de frequência do segmento, mas sim o modo como este varia. Por isso, o uso desta métrica resultou em segmentos que não representam necessariamente eventos acústicos com as mesmas fontes (como pode ser verificado pela escuta direta dos segmentos), mas sim fontes com comportamento temporal similar. Em algumas situações, a categorização pelo chirpograma agrupou eventos bastante distintos à audição, mas que representam por exemplo situações em que há aumento de frequência em momentos semelhantes.

Isto posto, e uma vez que o objetivo principal da análise do sinal acústico obtido do OceanPod é gerar amostras de eventos acústicos de causas semelhantes, o método de categorização pelas DFTs médias mostrou-se o mais indicado.

5.3 Conclusão

Na análise do sinal subaquático real obtido do OceanPod, verificamos em primeiro lugar que o algoritmo de segmentação foi capaz de revelar alguns padrões da ocorrência de eventos acústicos na região do Parque Estadual da Laje de Santos. Esses padrões permitem a análise da intensidade e diversidade da atividade acústica na área do parque.

Em segundo lugar, a categorização dos segmentos, principalmente utilizando a caracterização pelas DFTs médias, foi capaz de separar grupos de segmentos com características acústicas bastante semelhantes, o que é um indício de que tais segmentos podem ser tomados como amostras de sinais gerados por causas da mesma natureza.

De um modo geral, o sistema composto pelos algoritmos de segmentação e categorização mostrou-se uma ferramenta valiosa na análise não-supervisionada de sinais acústicos de longa duração. Uma aplicação imediata desse sistema é na criação de amostras anotadas de eventos acústicos, amostras estas que podem ser posteriormente utilizadas na construção de algoritmos de classificação. Outra aplicação está na análise direta dos sinais categorizados, para auxiliar a compreensão da atividade humana e / ou animal em ambientes marítimos.

Referências Bibliográficas

Arveson e Vendittis(2000) P. T. Arveson e D. J. Vendittis. Radiated noise characteristics of a modern cargo ship. *The Journal of the Acoustic Society of America*, 107:118–129. Citado na pág. 18

Bretthorst(1990a) L. Bretthorst. Bayesian analysis. i. parameter estimation using quadrature nmr models. *Journal of Magnetic Resonance*, 88:533–551. Citado na pág. 2, 5

Bretthorst(1990b) L. Bretthorst. Bayesian analysis. ii. signal detection and model selection. *Journal of Magnetic Resonance*, 88:552–570. Citado na pág. 2, 6, 10, 11

Bretthorst(1990c) L. Bretthorst. Bayesian analysis. iii. applications to nmr signal detection, model selection and parameter estimation. *Journal of Magnetic Resonance*, 88:571–595. Citado na pág. 2

Bretthorst(1991) L. Bretthorst. Bayesian analysis. iv. noise and computing time considerations. *Journal of Magnetic Resonance*, 93:369–394. Citado na pág. 2

Bretthorst(1992) L. Bretthorst. Bayesian analysis v: Amplitude estimation, multiple well separated sinusoids. *Journal of Magnetic Resonance*, 98:501–523. Citado na pág. 2, 9, 21

Bretthorst(1988) L. Bretthorst. *Bayesian Spectrum Analysis and Parameter Estimation.* Springer-Verlag. Citado na pág. 2, 5, 7, 9

Brooks e Gelman(1998) S. P. Brooks e A. Gelman. *General Methods for Monitoring Convergence of Iterative Simulations.* Journal of Computation and Graphical Statistics. URL https://doi.org/10.1080/10618600.1998.10474787. Citado na pág. 63

Chakrabarty(2017) D. Chakrabarty. A new bayesian test for the intractability-countering hypothesis. *Journal of the American Statistical Association*, 112. Citado na pág. 14

Cox(1977) D. R. Cox. The role of significance tests. *Scandinavian Journal of Statistics*, 4:49–70. Citado na pág. 33

Daly e Rushforth(1965) R. F. Daly e C. K. Rushforth. Nonparametric detection of a signal of known form in additive noise. *IEEE Transactions on Information Theory*, 11(1):70–76. Citado na pág. 13

Diniz *et al.*(2012) M. Diniz, C. A. B. Pereira e J. M. Stern. Unit roots: Bayesian significance test. *Communications in Statistics - Theory and Methods*, 40(23):4200–4213. Citado na pág. 14

Gamerman e Lopes(2006) D. Gamerman e H. F. Lopes. *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference.* Chapman and Hall CRC. Citado na pág. 16

Giorgino(2009) T. Giorgino. Computing and visualizing dynamic time warping alignments in r: The dtw package. *Journal of Statistical Software*, 31(7). Citado na pág. 81

G.O. Roberts e Gelman(1996) W. R. Gilks G.O. Roberts e A. Gelman. Efficient metropolis jumping rules. Em J. M. Bernardo, J. O. Berger, A. F. David e A. F. M. Smith, editors, *Bayesian Statistics V.* Oxford University Press, Oxford, USA. Citado na pág. 63

Good(1992) I. J. Good. The bayes/non-bayes compromise: a brief review. *Journal of the American Statistical Association*, 87(419):597–606. Citado na pág. 33, 34

Gregory(2001) P. C. Gregory. A Bayesian revolution in spectral analysis. *American Institute of Physics Proceedings*, 568:557. Citado na pág. 2

Gu e Jin(2006) J. Gu e X. Jin. A simple approximation for dynamic time warping search in large time series database. Em V. Botti E. Corchado, H. Yin e C. Fyfe, editors, *Intelligent Data Engineering and Automated Learning - IDEAL 2006. Lecture Notes in Computer Science*, volume 4224. Springer, Berlin, Heidelberg. Citado na pág. 82

H. Skutkova e Provaznik(2013) P. Babula R. Kizek H. Skutkova, M. Vitek e I. Provaznik. Classification of genomic signals using dynamic time warping. *BMC Bioinformatics*, 14. Citado na pág. 81

Haario *et al.***(2001)** H. Haario, E. Saksman e J. Tamminen. An adaptive metropolis algorithm. *Bernoulli*. doi: https://doi.org/10.2307/3318737. Citado na pág. 62, 63

Hastie *et al.***(2001)** T. Hastie, R. Tibshirani e J. Friedman. *The Elements of Statistical Learning*. Springer New York Inc., New York, NY, USA. Citado na pág. 93

Hooke e Jeeves(1961) R. Hooke e T. A. Jeeves. "direct search"solution of numerical and statistical problems. *Journal of the Association for Computing Machinery*, 8(2). Citado na pág. 21

Hubert e Arabie(1985) L. Hubert e P. Arabie. Comparing partitions. *Journal of Classification*, 2(1). Citado na pág. 96

Hubert *et al.***(2009)** P. Hubert, M. Lauretto e J. M. Stern. Fbst for the generalized poisson distribution. *AIP Conference Proceedings*, 1193(210). Citado na pág. 14

Hubert *et al.***(2017)** P. Hubert, L. Padovese e J. M. Stern. Full bayesian approach for signal detection with an application to boat detection on underwater soundscape data. *37th Maximum Entropy Methods in Science and Engineering*. Citado na pág. 23

Hubert *et al.***(2018a)** P. Hubert, L. Padovese e J. M. Stern. A sequential algorithm for signal segmentation. *Entropy*, 20(1):44. Citado na pág. 73

Hubert *et al.***(2018b)** P. Hubert, L. Padovese e J. M. Stern. Fast implementation of a Bayesian unsupervised segmentation algorithm. 2018b. URL https://arxiv.org/abs/1803.01801. Citado na pág. 73

Itakura(1965) F. Itakura. Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 23(1):67–72. Citado na pág. 82

Jaynes(2003) E. Jaynes. *Probability theory*. Cambridge University Press, Cambridge. Citado na pág. 2, 4

Jaynes(1982) E. T. Jaynes. On the rationale of maximum-entropy methods. *Proceedings of the IEEE*, 70(9):939–952. Citado na pág. 2, 4

Jaynes(1987) E. T. Jaynes. Bayesian spectrum and chirp analysis. Em C.R. Smith e G.J. Erickson, editors, *Maximum-Entropy and Bayesian Spectral Analysis and Estimation Problems*. D. Reidel Publishing Co. Citado na pág. 2, 4, 9, 20, 78, 79

Jeffreys(1939) H. Jeffreys. *Theory of Probability*. Oxford University Press, Londres. Citado na pág. 6

Keysers e Unger(2003) D. Keysers e W. Unger. Elastic image matching is np-complete. *Pattern Recognition Letters*, 24:445–453. Citado na pág. 82

Kozaczka e Grelowska(2011) E. Kozaczka e G. Grelowska. Shipping low frequency noise and its propagation in shallow water. *Acta Physica Polonica A*, 119:1009–1012. Citado na pág. 18

Kuntamalla e Reddy(2014) S. Kuntamalla e L. Ram Gopal Reddy. An efficient and automatic systolic peak detection algorithm for photoplethysmographic signals. *International Journal of Computer Applications*, 97:19. Citado na pág. 25

L. Pericchi(2016) C. A. B. Pereira L. Pericchi. Adaptative significance levels using optimal decision rules: Balancing the error probabilities. *Brazilian Journal of Probability and Statistics*, 30(1):70–90. Citado na pág. 33, 34

Levene(1960) H. Levene. Robust tests for equality of variances. Em H. Hotteling e I. Olkin, editors, *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, páginas 278–292. Stanford University Press, Stanford. Citado na pág. 32

Loredo(1990) T.J. Loredo. From laplace to supernova sn 1987a: Bayesian inference in astrophysics. Em P.F. Fougere, editor, *Maximum Entropy and Bayesian Methods. Fundamental Theories of Physics vol. 39*, páginas 81–142, Dordrecht. Springer. Citado na pág. 79

M. Shokoohi-Yekta e Keogh(2015) J. Wang M. Shokoohi-Yekta e E. Keogh. On the non-trivial generalization of dynamic time warping to the multidimensional case. Em *Proceedings of the 2015 SIAM International Conference on Data Mining*, páginas 39–48. Citado na pág. 82

Madruga e L. G. Esteves(2001) M. R. Madruga e S. Wechsler L. G. Esteves. On the bayesianity of pereira-stern tests. *Test*, 10(2):291–299. Citado na pág. 36

Makowsky e Hossa(2014) R. Makowsky e R. Hossa. Automatic speech signal segmentation based on the innovation adaptive filter. *Int. J. Appl. Math. Comput. Sci.*, 24(2):259–270. Citado na pág. 25

McKenna *et al.*(2012) M. F. McKenna, D. Ross, S. M. Wiggins e J. A. Hildebrand. Underwater radiated noise from modern commercial ships. *The Journal of the Acoustical Society of America*, 131:92–103. Citado na pág. 19

Ogden *et al.*(2011) G. L. Ogden, L. M. Zurk, M. E. Jones e M. E. Peterson. Extraction of small boat harmonic signatures from passive sonar. *The Journal of the Acoustical Society of America*, 129:3768–3776. Citado na pág. 19

P. Tsinaslanidis e Livanis(2014) A. Zapranis P. Tsinaslanidis, A. Alexandridis e E. Livanis. Dynamic time warping as a similarity measure: Applications in finance. Em *Hellenic Finance and Accounting Association*. Citado na pág. 81

Palshikar(2009) G. K. Palshikar. Simple algorithms for peak detection in time series. Em *1st Int. Conf. Advanced Data Analysis, Business Analytics and Intelligence*. Citado na pág. 53, 54

Pereira e Stern(1999) C. A. B. Pereira e J. M. Stern. Evidence and credibility: full Bayesian significance test for precise hypotheses. *Entropy*, 1:99–110. Citado na pág. 14, 15, 33, 35

Pereira e Wechsler(1993) C. A. B. Pereira e S. Wechsler. On the concept of p-value. *Revista Brasileira de Probabilidade e Estatï¿½stica*, 7:159–177. Citado na pág. 14, 33

Pereira *et al.*(2017) C. A. B. Pereira, E. Y. Nakano, V. Fossaluza, L. G. Esteves, M. A. Gannon e A. Polpo. Hypothesis tests for binomial experiments: Ordering the sample space by Bayes factors and using adaptive significance levels for decisions. *Entropy*, 19(12). Citado na pág. 34

Perez e Pericchi(2014) M. Perez e L. R. Pericchi. Changing statistical significance with the amount of information: The adaptative α significance level. *Statistics & Probability Letters*, 85: 20–24. Citado na pág. 33, 34

Rand(1971) W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846–850. Citado na pág. 96

Reynolds *et al.***(1992)** A. Reynolds, G. Richards, B. de la Iglesia e V. J. Rayward-Smith. Clustering rules: a comparison of partitioning and hierarchical clustering algorithms. *Journal of Mathematical Modelling and Algorithms*, 5:475–504. Citado na pág. 93

Rousseeuw(1987) P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65. Citado na pág. 89

Ruanaidh e Fitzgerald(1996) J. J. K. Ruanaidh e W. J. Fitzgerald. *Numerical Bayesian Methods Applied to Signal Processing*. Springer, Nova York. Citado na pág. 2

Sakoe e Chiba(1978) H. Sakoe e S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26(1): 43–49. Citado na pág. 80, 82

Salvador e Chan(2007) S. Salvador e P. Chan. Toward accurate dynamic time warping in linear time and space. *Journal of Intelligent Data Analysis*, 11(5):561–580. Citado na pág. 82

Sanchez-Gendriz e Padovese(2016) I. Sanchez-Gendriz e L. Padovese. Underwater soundscape of marine protected areas in the south brazilian coast. *Marine Pollution Bulletin*, 105:65–72. Citado na pág. 1

Schwartzman *et al.***(2011)** A. Schwartzman, Y. Gavrilov e R. J. Adler. Multiple testing of local maxima for detection of peaks in 1d. *The Annals of Statistics*, 39(6):3290–3319. Citado na pág. 25

Seljebotn(2009) D. S. Seljebotn. Fast numerical computations with cython. Em *Proceedings of the 8th Python in Science Conference*, páginas 15–22. Citado na pág. 61

Shannon(2001) C. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55. Citado na pág. 1

Stern(2008) J. M. Stern. *Cognitive Constructivism and the Epistemic Significance of Sharp Hypothesis*. Citado na pág. 33

ten Holt *et al.***(2007)** G.A. ten Holt, M.J.T. Reinders e E.A. Hendriks. Multi-dimensional dynamic time warping for gesture recognition. Em *Thirteenth annual conference of the Advanced School for Computing and Imaging*. Citado na pág. 81, 82

Thedorou *et al.***(2014)** T. Thedorou, I. Mporas e N. Fakotakis. An overview of automatic audio segmentation. *I.J. Information Technology and Computer Science*, 11:1–9. Citado na pág. 25

Ukil e Zivanovic(2006) A. Ukil e R. Zivanovic. *Automatic Signal Segmentation based on Abrupt Change Detection for Power Systems Applications*. Power India Conference. Citado na pág. 25

Vintsyuk(1968) T. K. Vintsyuk. Speech discrimination by dynamic programming. *Kibernetika*, 4(1):81–88. Citado na pág. 80

Vrugt(2016) J. A. Vrugt. Markov chain monte carlo simulation using the dream software package: Theory, concepts and matlab implementation. *Environmental Modelling & Software*, 75:273–316. Citado na pág. 17, 18

Vrugt *et al.***(2009)** J. A. Vrugt, C. J. F. ter Braak, C. G. H. Diks, B. A. Robinson, J. M. Hyman e D. Higdon. Accelerating markov chain monte carlo simulation by differential evolution with self-adaptive randomized subspace sampÄ°ing. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10(3):273–290. Citado na pág. 17, 18