



Universidade do Estado do Rio de Janeiro

Centro de Tecnologia e Ciências

Faculdade de Engenharia

Paulo Victor Martins Rosa Huguenin de Lima

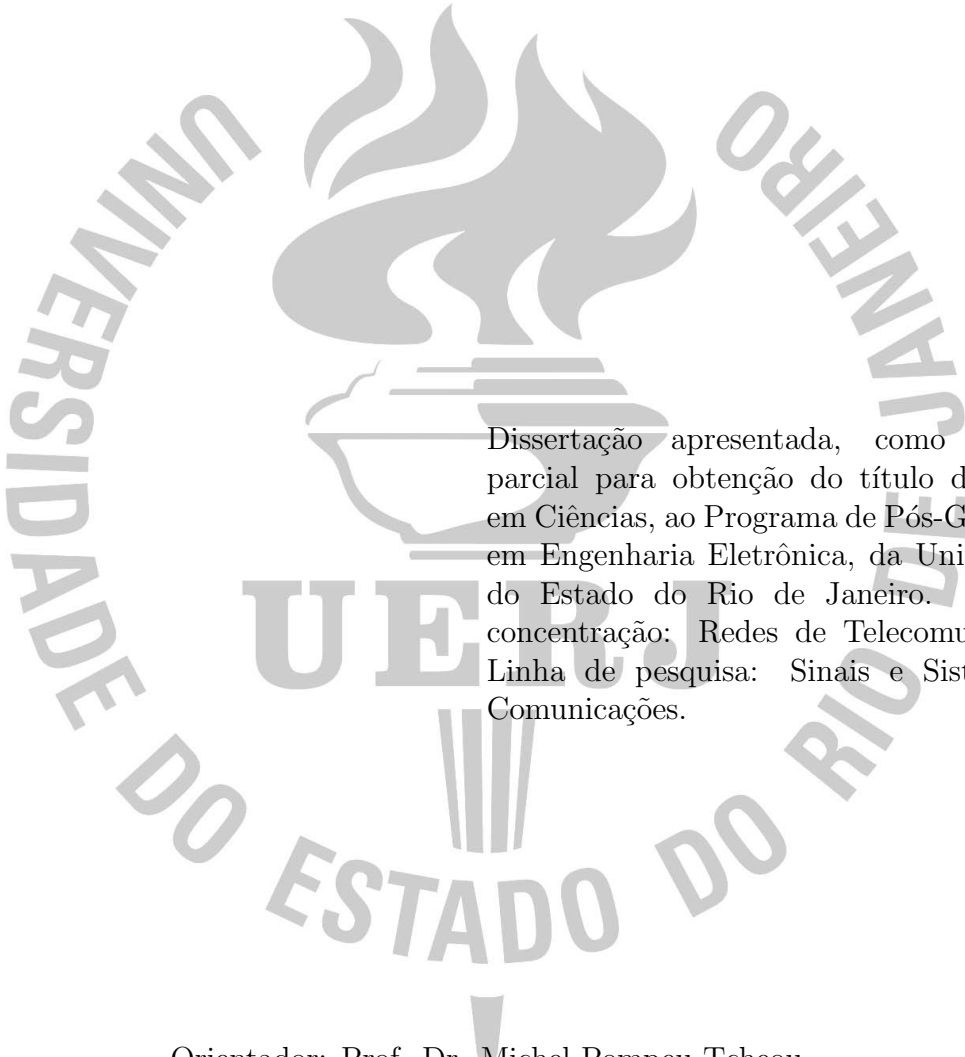
**REPRESENTAÇÃO DE SINAIS DE ATIVIDADE
ELETRODÉRMICA USANDO DECOMPOSIÇÕES
ATÔMICAS COM ACELERAÇÃO VIA REDES NEURAIIS
ARTIFICIAIS**

Rio de Janeiro

2023

Paulo Victor Martins Rosa Huguenin de Lima

**REPRESENTAÇÃO DE SINAIS DE ATIVIDADE ELETRODÉRMICA
USANDO DECOMPOSIÇÕES ATÔMICAS COM ACELERAÇÃO VIA
REDES NEURAIS ARTIFICIAIS**



Dissertação apresentada, como requisito parcial para obtenção do título de Mestre em Ciências, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Redes de Telecomunicações. Linha de pesquisa: Sinais e Sistemas de Comunicações.

Orientador: Prof. Dr. Michel Pompeu Tcheou

Orientador: Prof. Lisandro Lovisolo

Rio de Janeiro

2023

CATALOGAÇÃO NA FONTE

UERJ / REDE SIRIUS / BIBLIOTECA CTC/B

L732 Lima, Paulo Victor Martins Rosa Huguenin de.
Representação de sinais de atividade eletrodérmica usando decomposições atômicas com aceleração via redes neurais artificiais / Paulo Victor Martins Rosa Huguenin de Lima. – 2023. 133 f.

Orientadores: Michel Pompeu Tcheou, Lisandro Lovisolo.
Dissertação (Mestrado) – Universidade do Estado do Rio de Janeiro, Faculdade de Engenharia.

1. Engenharia eletrônica - Teses. 2. Redes neurais (Computação) - Teses. 3. Processamento de linguagem natural (Computação) - Teses. 4. Algoritmos - Teses. I. Tcheou, Michel Pompeu. II. Lovisolo, Lisandro. III. Universidade do Estado do Rio de Janeiro, Faculdade de Engenharia. IV. Título.

CDU 004.032.26

Bibliotecária: Júlia Vieira – CRB7/6022

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta tese, desde que citada a fonte.



Assinatura

25/08/2023

Data

Paulo Victor Martins Rosa Huguenin de Lima

**REPRESENTAÇÃO DE SINAIS DE ATIVIDADE ELETRODÉRMICA
USANDO DECOMPOSIÇÕES ATÔMICAS COM ACELERAÇÃO VIA
REDES NEURAIS ARTIFICIAIS**

Dissertação apresentada, como requisito parcial para obtenção do título de Mestre em Ciências, ao Programa de Pós-Graduação em Engenharia Eletrônica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Redes de Telecomunicações. Linha de pesquisa: Sinais e Sistemas de Comunicações.

Aprovado em: 31 de Maio de 2023

Banca Examinadora:

Prof. Dr. Michel Pompeu Tcheou (Orientador)

Faculdade de Engenharia - UERJ

Prof. Dr. Lisandro Lovisolo (Orientador)

Faculdade de Engenharia - UERJ

Prof. Dr. Tadeu Nagashima Ferreira

Escola de Engenharia - UFF

Prof. Dr. Gabriel Matos Araújo

CEFET-RJ

Rio de Janeiro

2023

AGRADECIMENTO

Primeiramente, gostaria agradecer ao meu pai Paulo Sérgio que apesar de não poder estar mais presente fisicamente comigo nesta fase tão especial, sempre me apoiou e me incentivou durante a vida, a ele todo amor, carinho e a lembrança eterna. Queria agradecer imensamente a minha mãe, Nara Rosa, que sempre esteve ao meu lado me apoiando e me dando motivação para superar cada etapa da vida. Gostaria de agradecer também a toda a minha família, principalmente aos irmãos Marcos Paulo e Paulo Renato, e a minha esposa, Isabella Almeida, que está comigo me dando apoio. Agradecimentos vão para os professores responsáveis por todo o conhecimento trocado e apoio dado, em especial ao professor e orientador Michel Tcheou por toda paciência, ajuda e atenção dada nesse período e grande responsável pela a realização deste trabalho. Agradecimentos também ao professor Lisandro Lovisolo, quem também orientou e fez parte desta caminhada, ao corpo docente do Programa de Pós-graduação em Engenharia Eletrônica (PEL) e a FA-PERJ pelo incentivo financeiro. Por fim, gostaria de agradecer a todos meus amigos que também foram importantes nesta etapa da vida.

A vocês, o meu Muito Obrigado!

Aqui, no entanto, nós não olhamos para trás por muito tempo. Nós continuamos seguindo em frente, abrindo novas portas e fazendo coisas novas porque somos curiosos... e a curiosidade continua nos conduzindo por novos caminhos. Siga em frente.

Walt Disney

RESUMO

LIMA, Paulo Victor Martins Rosa Huguenin. *Representação de Sinais de Atividade Eletrodérmica Usando Decomposições Atômicas Com Aceleração Via Redes Neurais Artificiais*. 134 f. Dissertação (Mestrado em Engenharia Eletrônica) - Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro (UERJ), Rio de Janeiro, 2023.

A profusão de dispositivos embarcando sensores capazes de monitorar e capturar sinais biomédicos em tempo real, dentre eles a atividade eletrodérmica (EDA), demanda técnicas capazes de representar eficientemente o sinal de EDA que permitam ao mesmo tempo analisá-los de forma a prover informações úteis para suas interpretações. Algoritmos vorazes de decomposição adaptativa, como o *Matching Pursuit* (MP), têm sido utilizados para obter aproximações de sinais coerentes, isto é, relacionadas com os fenômenos representados nos sinais. O MP obtém uma soma ponderada de poucas formas de onda (átomos) selecionadas a partir de um dicionário redundante iterativamente. O dicionário contém os átomos de onde selecionam-se quais empregar na aproximação. Quando ele contém múltiplas formas de ondas protótipos a partir das quais os átomos são parametrizados, podemos dizer que o dicionário é composto de subdicionários. Nesta dissertação, investigamos o uso de um tal dicionário para a representação de sinais de EDA via MP, já que tal dicionário pode fornecer decomposições mais esparsas de sinais de EDA. Investigam-se as parametrizações adequadas para os subdicionários contendo átomos similares aos fenômenos representados em sinais EDA. Por outro lado, isso aumenta a complexidade computacional na obtenção da representação. Propõe-se um método de seleção de subdicionário no laço do algoritmo MP usando uma rede neural artificial (RNA) para reduzi-la sem impactar a qualidade da representação obtida. Investigam-se modelos de RNAs capazes de prever com boa acurácia o subdicionário a usar no laço de decomposição MP. Tal abordagem permite dividir a complexidade computacional pela quantidade de subdicionários sem comprometer fortemente a qualidade (medida em função do erro da aproximação) em função da quantidade de átomos (iterações do MP) da representação.

Palavras-chave: Atividade Eletrodérmica. Representações Esparsas. *Matching Pursuit*. Redes Neurais Artificiais.

ABSTRACT

LIMA, Paulo Victor Martins Rosa Huguenin. *Electrodermal Activity Signals Representation Using Atomic Decompositions With Acceleration Via Artificial Neural Networks*. 134 f. Dissertation (Master Electronic Engineering) - Faculty of Engineering, Universidade do Estado do Rio de Janeiro (UERJ), Rio de Janeiro, 2023.

The profusion of devices incorporating sensors capable of monitoring and capturing biomedical signals in real-time, including electrodermal activity (EDA), demands techniques capable of efficiently representing the EDA signal and, at the same time, allowing analysis to provide useful information for interpretation. Greedy adaptive decomposition algorithms, such as *Matching Pursuit* (MP), have been used to obtain coherent approximations of signals, i.e., those related to the phenomena represented in the signals. The MP obtains a weighted sum of a few waveforms (atoms) selected from a redundant dictionary iteratively to represent the signal. The set of candidate atoms is the dictionary from which the ones to use in the approximation are selected. When it contains multiple prototype waveforms, from which the atoms are defined by parameters, we can say that the dictionary is composed of subdictionaries. In this dissertation, we investigate the use of a composite dictionary for the representation of EDA signals via MP, motivated by the fact that such a dictionary can provide more sparse decompositions of EDA signals. Appropriate parameterizations for different subdictionaries containing atoms similar to the phenomena represented in EDA signals are investigated. On the other hand, a composite dictionary entails greater computational complexity to obtain the representation. We propose a subdictionary selection method that acts in the loop of the MP algorithm using an artificial neural network (ANN) to reduce the computational cost of the algorithm without impairing the quality of the representation obtained. ANN models capable of predicting with good accuracy the subdictionary to be used in the MP decomposition loop are investigated. The results indicate that this approach allows dividing the computational complexity by the number of subdictionaries without strongly compromising the quality (measured as a function of the approximation error) as a function of the number of atoms (MP iterations) of the representation thus obtained.

Keywords: Electrodermal Activity. Sparse Representation. Matching Pursuit. Artificial Neural Networks.

LISTA DE FIGURAS

Figura 1 - Dispositivo de captação de atividade eletrodérmica (pulseira Affective Q Sensor).....	24
Figura 2 - Medição de um sinal de EDA.	24
Figura 3 - Diagrama sinóptico das componentes de um sinal de EDA.	25
Figura 4 - Análise de parâmetros de um SCR em um sinal de EDA.	26
Figura 5 - Diagrama de blocos do processo de separação do sinal de EDA em suas componentes fásica e tônica.....	30
Figura 6 - Sinal <i>eda1</i> original ampliado com artefatos da captação.....	37
Figura 7 - Densidade espectral de potência de um sinal EDA.	37
Figura 8 - Sinal <i>eda1</i> ampliado e filtrado com um filtro passa-baixa.	38
Figura 9 - Estimativa da componente tônica pelo método MMQ para diferentes tamanhos de blocos.....	40
Figura 10- Separação das componentes do sinal de EDA usando o algoritmo de otimização convexa (cvxEDA) para estimar a componente tônica.	41
Figura 11- Diagrama de blocos de análise e síntese usando decomposições atômicas .	43
Figura 12- Átomo de <i>Bateman</i>	53
Figura 13- Átomo de Sigmoide-Exponencial	55
Figura 14- Átomo de Chi-quadrado.....	56
Figura 15- Átomo de Gabor.	57
Figura 16- Diagrama de blocos do modelo de decomposição usando <i>Matching Pursuit</i> .	60
Figura 17- Dicionários parametrizados e adaptado de [1].	61
Figura 18- Combinação dos parâmetros ρ e η do dicionário de <i>Bateman</i> e selecionados pelo <i>Matching Pursuit</i>	64
Figura 19- Histograma dos parâmetros ρ e η do dicionário de <i>Bateman</i> para 50 átomos por bloco.	64
Figura 20- Histograma dos parâmetros ρ e η do dicionário de <i>Bateman</i> para 10 átomos por bloco.	65

Figura 21- Exemplo da decomposição realizada pelo MP para um sinal artificial para ilustrar o erro gerado após o algoritmo encontrar o átomo mais relacionado	66
Figura 22- Combinação dos parâmetros ρ (decaimento) e η (subida) do dicionário de Sigmoide-Exponencial e selecionados pelo <i>Matching Pursuit</i>	67
Figura 23- Histograma dos parâmetros ρ e η do dicionário de Sigmoide-Exponencial para 10 átomos por bloco	67
Figura 24- Combinação dos parâmetros χ (grau de liberdade) e s (escala) do dicionário de Chi-quadrado e selecionados pelo <i>Matching Pursuit</i>	68
Figura 25- Histograma dos parâmetros χ (grau de liberdade) e s (escala) do dicionário de Chi-quadrado para 10 átomos por bloco	69
Figura 26- Histograma dos parâmetros s do dicionário de Gabor para 50 átomos por bloco	69
Figura 27- SNR médio da decomposição para os 5 sinais <i>eda1</i> a <i>eda5</i> usando cada sub-dicionário separadamente e o dicionário completo	70
Figura 28- Histograma das frequências dos tipos de subdicionários escolhidos de acordo números de iterações por bloco	71
Figura 29- SNR médio da decomposição para os 5 sinais <i>eda6</i> a <i>eda10</i> usando o dicionário original e o dicionário ajustado.	73
Figura 30- Avaliação da decomposição do sinal de EDA por iteração.	74
Figura 31- Diagrama de blocos do modelo de decomposição atômica com seletor de subdicionários.	75
Figura 32- Modelo Perceptron	78
Figura 33- Função de Ativação Sigmoide.	79
Figura 34- Função de Ativação ReLu.	80
Figura 35- Modelo Perceptron Multicamadas	81
Figura 36- Ilustração do princípio de funcionamento da técnica de <i>Dropout</i>	87
Figura 37- Diagrama de blocos do processo de obtenção do conjunto de dados e do treinamento do modelo.	89
Figura 38- Exemplo da divisão de blocos de 128 (N) amostras usando saltos de 64 ($N/2$) amostras	90
Figura 39- CDF da taxa de aproximação.....	91

Figura 40- Decimação do resíduo por um fator 2	92
Figura 41- Histogramas das classes presentes nos subdicionários para blocos com SCRs centrais.	95
Figura 42- Histogramas das classes presentes nos subdicionários para blocos sem restrição de SCRs centralizados.	96
Figura 43- Ilustração da configuração da técnica <i>k-folds</i>	102
Figura 44- Subdicionários empregados nas decomposições MP de sinais EDA de diferentes pacientes.	102
Figura 45 Matriz de confusão para um classificador binário (de duas classes).....	103
Figura 46- Matrizes de confusão para o classificador dentre três subdicionários na decomposição MP de blocos com SCRs centralizados.....	106
Figura 47- Matriz de confusão dos classificadores binários para pares de subdi- cionários considerando a decomposição MP de blocos com SCRs centrali- zados.....	108
Figura 48- Matrizes de confusão para o classificador dentre três subdicionários na decomposição MP de blocos sem imposição de SCRs centralizados.....	111
Figura 49- Matriz de confusão dos classificadores binários para pares de subdi- cionários considerando a decomposição MP de todos os blocos.	112
Figura 50- Custo computacional do <i>MP</i> com e sem seleção de subdicionários via RNAs em função do tamanho do sinal	113
Figura 51- SNR de decomposição atômica usando o seletor de dicionário no laço MP.	114
Figura 52- SNR de decomposição atômica usando o seletor de dicionário no laço MP e chaveando para o subdicionário de Gabor quando o resíduo se assemelha a ruído de decomposição MP.	115
Figura 53- Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários <i>Bateman</i> e Gabor.	130
Figura 54- Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários <i>Bateman</i> e Sigmoide-Exponencial.	130
Figura 55- Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários Sigmoide-Exponencial e Gabor.	131
Figura 56- Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,44$. .	131

Figura 57- Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,66$. .	131
Figura 58- Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,81$. .	132
Figura 59- Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide Exponencial para $\lambda = 0,9$.	132
Figura 60- Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários <i>Bateman</i> e Gabor.	132
Figura 61- Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários <i>Bateman</i> e Sigmoide-Exponencial.	133
Figura 62- Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários Sigmoide-Exponencial e Gabor.	133
Figura 63- Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,44$.	133
Figura 64- Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,66$.	134
Figura 65- Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide-Exponencial para $\lambda = 0,81$.	134
Figura 66- Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários <i>Bateman</i> , Gabor e Sigmoide Exponencial para $\lambda = 0,9$.	134

LISTA DE TABELAS

Tabela 1 - SNR médio dos sinais de EDA usando MMQ para estimar a componente tônica.	38
Tabela 2 - Discretização de Parâmetros do Dicionário 1 (Dicionário Base).....	62
Tabela 3 - Cardinalidade do Dicionário 1.	63
Tabela 4 - Cardinalidade do Dicionário Ajustado.....	72
Tabela 5 - Cardinalidade dos dados de entrada para diferentes limites de taxas de aproximação.....	91
Tabela 6 - Cardinalidade do conjunto de dados com a aplicação do balanceamento por <i>Undersampling</i> para diferentes percentis de resíduos aplicados.....	94
Tabela 7 - Intervalos para Bayes Search.	100
Tabela 8 - Melhores hiperparâmetros do treinamento usando a decomposição de blocos com SCRs centralizados para o classificador que considera os três subdicionários: Bateman, Gabor e Sigmóide.	105
Tabela 9 - Acurácia do seletor de subdicionário ternário quando treinado com blocos contendo SCRs centralizados.	105
Tabela 10- Melhores hiperparâmetros do treinamento usando a decomposição de blocos com SCRs centralizados para os classificadores binários.	107
Tabela 11- Acurácias de treino, validação e teste do seletor binário de subdicionários considerando resíduos produzidos a partir de blocos com SCRs centralizados.	108
Tabela 12- Melhores hiperparâmetros do treinamento usando a decomposição de blocos sem a imposição de SCRs centralizados para o classificador que considera os três subdicionários Bateman, Gabor e Sigmóide.	109
Tabela 13- Acurácia do seletor de subdicionários ternários quando treinado com os resíduos de todos os blocos – sem a imposição de SCRs centralizados.....	110
Tabela 14- Melhores hiperparâmetros do treinamento usando a decomposição de blocos sem a restrição de SCRs centralizados para os classificadores binários.	111
Tabela 15- Acurácias de treino, validação e teste do seletor binário de subdicionários considerando resíduos produzidos a partir de todos os blocos (sem a restrição de SCRs centralizados).....	111

LISTA DE SIGLAS

EDA	<i>Electrodermal Activity</i> (Atividade Eletrodérmica)
SC	<i>Skin Conductance</i>
SCR	<i>Skin Conductance Response</i>
SCL	<i>Skin Conductance Level</i>
GSR	<i>Galvanic Skin Response</i>
ER-SCR	<i>Event Related Skin Conductance Response</i>
ECG	Eletrocardiograma
EEG	Eletroencefalograma
MP	<i>Matching Pursuit</i>
MMQ	Método Mínimos Quadrado
IRF	<i>Impulse Response Function</i>
IIR	<i>Infinite Impulse Response</i>
SMNA	<i>Sudomotor Nerve Activity</i>
QP	<i>Quadratic Program</i>
FFT	<i>Fast Fourier Transform</i>
DFT	<i>Discrete Fourier Transform</i>
PDF	<i>Probability Density Function</i>
CDF	<i>Cumulative Density Function</i>
RNA	Rede Neural Artificial
MLP	<i>Multi Layer Perceptron</i>
SGD	<i>Stochastic Gradient Descent</i>
CV	<i>Cross-Validation</i>
ReLu	<i>Rectified Linear Unit</i>
SNR	<i>Signal-to-noise Ratio</i>

SUMÁRIO

	INTRODUÇÃO	17
1	PRÉ-PROCESSAMENTO DO SINAL DE EDA	22
1.1	Características da Atividade Eletrodérmica	22
1.2	Componentes do Sinal de EDA	23
1.3	Aquisição de dados	26
1.4	Pré-processamento	27
1.5	Método dos Mínimos Quadrados	28
1.6	Otimização Convexa	31
1.6.1	Modelo	32
1.6.2	<i>Maximum a Posteriori Estimation</i>	34
1.7	Simulações e Resultados	36
1.7.1	Pré-processamento	36
1.7.2	MMQ	37
1.7.3	CVXEDA	38
2	DECOMPOSIÇÕES ATÔMICAS DE SINAIS DE EDA	42
2.1	Representação Atômica de Sinais	42
2.2	Matching Pursuit	45
2.2.1	Redução de Custo Computacional	47
2.3	Dicionários Redundantes Parametrizados	51
2.4	Dicionário de Bateman	52
2.5	Dicionário de Sigmoide-Exponencial	54
2.6	Dicionário de Chi-Quadrado	55
2.7	Dicionário de Gabor	56
2.8	Discretização de Parâmetros	58
2.8.1	Figuras de mérito para avaliação da decomposição	59
2.8.2	Taxa de aproximação	59
2.9	Metodologia e Resultados da Discretização de Parâmetros	60
2.9.1	Resultado da Discretização de <i>Bateman</i>	63

2.9.2	Resultado da Discretização da Sigmoide-Exponencial.....	66
2.9.3	Resultado da Discretização de Chi-Quadrado.....	68
2.9.4	Resultado da Discretização de Gabor	68
2.9.5	Resultado da Discretização de Múltiplos Dicionários.....	70
3	SELEÇÃO DE SUBDICIONÁRIO A USAR EM ITERAÇÕES	
	VIA REDES NEURAIS ARTIFICIAIS.....	75
3.1	Sobre Redes Neurais Artificiais	77
3.1.1	O Neurônio Artificial	77
3.1.2	Perceptron Multicamadas	80
3.1.3	Treinamento de RNAs: <i>Backpropagation</i>	82
3.1.3.1	Gradiente Descendente	83
3.1.3.2	Otimizador Adam.....	84
3.1.4	Heurísticas para a Generalização do Modelo.....	85
3.1.4.1	Dropout.....	86
3.2	Conjunto de Dados Empregado	87
3.2.1	Critério para Seleção de Átomos Coerentes	88
3.2.2	Dados na Entrada da Rede: Resíduos Decimados	91
3.2.3	Balanceamento do Conjunto de Dados	93
3.2.4	Normalização dos Dados.....	97
3.3	Opções e Critérios Aplicados no Projeto do Classificador.....	97
3.3.1	Arquitetura da Rede	98
3.3.2	Busca por Hiperparâmetros	98
3.3.3	Validação Cruzada	100
3.3.4	Critérios de Desempenho	101
3.4	Projeto de Classificadores de Subdicionários para Iterações MP	104
3.4.1	Caso de Blocos com SCRs Centralizados.....	104
3.4.1.1	Classificação Multiclasses.....	105
3.4.1.2	Classificação Binária	107
3.4.2	Caso de Blocos Sem Restrição de Centralização	109
3.4.2.1	Classificação Multiclasses.....	109
3.4.2.2	Classificação Binária	110

3.5	Resultados da Decomposição Usando o Seletor de Subdicionários no Laço do MP	112
	CONCLUSÃO	116
	REFERÊNCIAS	120
	Apêndice A	130
A	CURVAS DE APRENDIZADO DOS SELETORES DE SUBDICONÁRIOS	130

INTRODUÇÃO

Nos últimos anos, a evolução das tecnologias de aquisição de dados, como sensores e dispositivos portáteis que podem coletar dados biomédicos em tempo real capazes de captar e processar sinais fisiológicos provenientes do corpo humano, como a Atividade Eletrodérmica (EDA) e o Eletrocardiograma (ECG), vem chamando a atenção para estudos que buscam maneiras eficientes de representar, analisar, comprimir, transmitir e buscar padrões nesses sinais [1–5]. A capacidade de processar esses grandes conjuntos de dados e extrair informações úteis com o auxílio de técnicas de representações de sinais e aprendizado de máquina tem impulsionado o desenvolvimento de novas ferramentas de monitoramento em tempo real, de diagnóstico e tratamento médico. A variedade de padrões de sinais motiva a utilização de decomposições atômicas adaptativas para a análise de sinais [4–11]. Elas podem permitir a análise e interpretação de sinais, o que é particularmente interessante para sinais gerados por sistemas biológicos. Assim, nesta dissertação, investiga-se a decomposição de sinais EDA através de decomposições atômicas adaptativas.

A atividade eletrodérmica é conhecida por ser modulada pelos ciclos respiratórios e cardíacos [12] e é formada por duas componentes principais: a resposta tônica e resposta fásica. Em geral, essas componentes são avaliadas separadamente, uma vez que apresentam características e representam fenômenos psicofisiológicos distintos [13]. A componente fásica traz consigo as informações/padrões de SCR (*Skin Conductance Response*), por exemplo. Em trabalhos anteriores, modelou-se a EDA matematicamente por meio de funções que descrevem a forma de onda das SCRs nelas contidas. Em [13] desenvolveu-se um modelo que correlaciona o sinal de EDA com sigmoide-exponenciais parametrizáveis. Já em [14] representou-se o formato do SCR através de funções bi-exponenciais e usando filtragem determinística inversa para estimar as rajadas do nervo. Em [15] assumiu-se o sinal de EDA como a convolução de uma função *driver*, refletindo a atividade das glândulas sudoríparas, com uma resposta ao impulso descrevendo os estados de atividade do neurônio. Em [16, 17] desenvolveu-se um modelo causal dinâmico usando inversão Bayesiana para inferir a atividade das glândulas sudoríparas. Cada rajada de atividade sudomotora é modelada como uma função Gaussiana, que serve como uma entrada para uma operação de dupla convolução. Observou-se que esse modelo foi considerado um bom

preditor de ansiedade para discursos em público. Em [18, 19], a decomposição de sinais de EDA é lançada como um problema de otimização convexa. A minimização de uma função de custo quadrático foi usada para estimar as componentes tônica e fásica, representadas por *splines* cúbicas e uma função de *Bateman* biexponencial, respectivamente. Tal modelo é utilizado neste trabalho para obtermos uma aproximação da componente fásica sob a qual atua a decomposição atômica adaptativa.

Como as componentes fásicas ocorrem geralmente de forma esparsa no tempo, em média a cada 2 minutos, nada mais adequado do que lançar mão de métodos de decomposições esparsas para representá-las de maneira eficiente em termos de compressão e extração de características. Para isso, é necessário usar funções base ou átomos que apresentem boas correlações com os fenômenos representados nos sinais [9, 20], que no caso se traduz como boas correlações com as formas de ondas de SCRs [1].

A decomposição atômica é uma técnica importante em processamento de sinais, em particular na análise de sinais não-estacionários. Essa técnica permite a decomposição de um sinal complexo em suas componentes mais simples e básicas, chamadas de átomos, que são funções pré-definidas com uma forma, duração e frequência bem definidas. Isto proporciona uma representação adaptativa, uma vez que os átomos selecionados para um dado sinal dependem do próprio sinal. Diz-se que o dicionário é redundante dado que tem mais átomos do que os necessários para formar uma base do espaço do sinal. Portanto, neste cenário, a decomposição consiste em selecionar as funções e os seus pesos, fornecendo uma combinação linear para aproximar um sinal específico, satisfazendo algum critério de distorção. Esses átomos podem ser escolhidos de diferentes formas, como por exemplo, usando uma base de *wavelets*, de Fourier ou de curvas harmônicas [1, 7, 9].

Através da decomposição atômica, é possível obter uma representação mais compacta do sinal, facilitando sua análise e processamento. Além disso, essa técnica pode ser usada para detectar e identificar padrões e eventos no sinal, como picos, oscilações ou mudanças abruptas de frequência ou amplitude. Aplicações práticas incluem a análise de sinais de áudio e música, o processamento de sinais biomédicos, a detecção de falhas em sistemas mecânicos, a análise de dados financeiros, entre outros.

O algoritmo básico para decomposições adaptativas vorazes usado neste projeto é o *Matching Pursuit* (MP) [21]. O interesse por este tipo de algoritmo advém da sua capacidade de fornecer aproximações esparsas de sinais [8, 9, 22]. Normalmente, representações

de sinais que utilizam menos elementos do que a dimensão do espaço do sinal original e que satisfazem um critério de distorção. O MP é um algoritmo iterativo sub-ótimo para aproximação de sinais. Este método é considerado guloso no sentido em que, em cada iteração, o átomo incluído na representação é selecionado maximizando a sua semelhança com o que ainda falta ser aproximado do sinal original (também denominado resíduo). Embora isto seja ótimo na iteração, pode ser sub-ótimo quando a representação completa é considerada [21–24].

Na literatura, o MP tem sido utilizado com diferentes dicionários para diversas aplicações, alguns exemplos usados para EDA são o Dicionário de Gabor (janelas Gaussianas moduladas) em conjunto com dicionário triangular e de *Bateman* [25]. Em [1] usa-se o dicionário de *Bateman*, dicionário de sigmóides-exponenciais e dicionário de Chi-quadrado. Os átomos pertencentes a cada um destes dicionários têm características diferentes nos seus conteúdos de tempo e frequência tornando a sua utilização mais adequada a diferentes fontes de sinal. Utilizar o dicionário apropriado para uma dada fonte de sinal permite melhorar a coerência (semelhança entre os átomos e os fenômenos representativos do sinal). Consequentemente, o MP é adequado para uma série de sinais, desde que o dicionário seja coerente com o sinal [6, 9, 11, 22]. Além da escolha do dicionário a parametrização do dicionário também é importante para a redução da complexidade computacional mantendo a qualidade da representação [10, 25, 26]. Há algumas estratégias onde subdicionários são criados iterativamente [27–29] ou que usam múltiplos subdicionários para compor o dicionário [25, 26], tentando obter aproximações ainda mais esparsas. O átomo selecionado em cada iteração do MP pode ser associado a algum fenômeno representado no resíduo, mas as propriedades do átomo são definidas pelo seu protótipo, ou seja, o subdicionário a que pertence. No entanto, assumindo um espaço de sinal de dimensão N , quanto maior a cardinalidade do dicionário (a quantidade de elementos), a princípio, maior será o custo computacional da iteração MP, ou seja, a utilização conjunta de vários sub-dicionários agrava o problema da dimensionalidade. Assim, é crucial procurar uma estratégia para reduzir a complexidade do algoritmo de decomposição e, ao mesmo tempo, fornecer representações coerentes do sinal.

Em [30], propôs-se o uso da Transformada Rápida de Fourier a fim de reduzir o custo computacional no cálculo dos produtos internos de acordo com a variação da frequência da função protótipo do dicionário. Em [7], adaptou-se o algoritmo proposto

em [30] de acordo com a variação do deslocamento temporal da função protótipo. Em [26], desenvolveu-se um método de seleção de subdicionário aplicados a sinais de áudio, empregando-se um classificador por meio de redes neurais artificiais (RNA) para inferir o sub-dicionário mais adequado para decompor o resíduo a cada iteração.

Objetivo

Nesta dissertação, é proposto um esquema de seleção de subdicionário usando modelos de redes neurais artificiais, baseado em [26], com o objetivo de reduzir a complexidade do algoritmo MP na escolha do melhor subdicionário a ser usado a cada iteração do algoritmo, permitindo a utilização de um dicionário composto por múltiplos subdicionários sem comprometer a representação do sinal de EDA (*Electrodermal Activity*). Além disso, antes, estuda-se a parametrização mais adequada de diferentes dicionários, como Gabor, *Bateman*, Sigmoide-Exponencial e Chi-quadrado, aos padrões intrínsecos de sinais de EDA com o propósito de reduzir suas dimensionalidades e, conseqüentemente, a complexidade da decomposição. A seleção de subdicionário usando modelos de RNAs a empregar na decomposição MPs de sinais EDA é aplicada para selecionar o subdicionário dentre os presentes no dicionário.

Organização do Texto

Inicialmente, no Capítulo 1, apresentam-se as características dos sinais de EDA, bem como, o processo de obtenção da base de dados de sinais a serem analisados. Além disso, descreve-se a etapa de pré-processamento por meio de métodos de estimação e separação das componentes tônica e fásica.

No Capítulo 2, apresenta-se o processo de decomposição atômica com uso de dicionários redundantes parametrizados baseado no algoritmo voraz de MP. Além disso, estuda-se a discretização dos parâmetros para cada subdicionário, avaliando a cardinalidade usada e qualidade da decomposição.

O Capítulo 3 aborda o método de seleção de dicionário usando RNAs. Neste capítulo, apresenta-se o conhecimento teórico da arquitetura e treinamento de uma rede neural utilizando técnicas de pré-processamento, controle de *overfitting*, busca por hiperparâmetros e métodos de avaliação. Além disso, é realizado um estudo de caso para

analisar o comportamento do modelo usando diferentes critérios de parada por taxa de aproximação e para casos utilizando todos os blocos da decomposição e apenas blocos com SCRs centralizadas. Por fim, apresentam-se os resultados da seleção de dicionários avaliando-se a redução de complexidade e a qualidade da decomposição após a implementação do modelo.

1 PRÉ-PROCESSAMENTO DO SINAL DE EDA

1.1 Características da Atividade Eletrodérmica

A atividade eletrodérmica mede a atividade elétrica na pele e pode ser usada para detectar alterações na atividade simpática, como a sudorese. O controle central da atividade das glândulas sudoríparas é atribuído às áreas hipotalâmicas, especialmente paraventricular e núcleos posteriores [31]. No entanto, outras regiões corticais e subcorticais são conhecidas por estarem envolvidas na modulação de sua ativação [32]. As fibras sudomotoras descem pela via simpática hipotálamo-reticular-espinhal em estreita proximidade com outras fibras simpáticas (por exemplo, eferências vasoconstritoras ou piloerectoras) que terminam nos neurônios sudomotores pré-ganglionares. Os neurônios pré-ganglionares deixam a medula espinhal ipsilateralmente através do córtex lateral que conduz ao tronco simpático, onde são trocados para neurônios pós-ganglionares. Os neurônios pós-ganglionares inervam a parte secretora das glândulas sudoríparas de forma amplamente ramificada. Embora a transmissão simpática pós-ganglionar seja geralmente adrenérgica, a transmissão sudomotora pós-ganglionar é colinérgica, usando acetilcolina como transmissor sináptico. O segmento secretor da glândula sudorípara está localizado no tecido subcutâneo. O suor é descarregado em um ducto de suor que conduz através da derme e epiderme e terminando em um poro na superfície da pele. A densidade das glândulas sudoríparas écrinas varia acentuadamente ao longo da pele. Segundo [33], a densidade é mais alta na sola e dorso do pé, na testa, bochecha, palma e antebraço (200-600 por cm^2). O tempo de condução da ativação central às glândulas sudoríparas nas pontas dos dedos (com distância média de 1,1 m) foi estimado em 1,1 s [34]. O tempo do neuroefetor nas glândulas sudoríparas foi estimado em 348 ms [35]. Cada glândula sudorípara é inervada por muitas fibras diferentes [36] e vice-versa. Cada unidade sudomotora inerva uma área de pele de cerca de $1,28 \text{ cm}^2$ [37]. As glândulas sudoríparas diferem acentuadamente em sua atividade e parecem possuir diferentes limiares de ativação [38]. Estudos envolvendo microneurografia [39] mostram que as fibras sudomotoras pós-ganglionares disparam em forma de explosão com rajadas com duração média de 638 ms [12]. Uma explosão sudomotora corresponde a uma única resposta de condutância da pele. O registro dessas alterações das respostas da condutância da pele é também denominado por seu acrônimo, em inglês, SCR (*Skin Conductance Response*). A amplitude do SCR foi encon-

trada para ser linearmente relacionada com a amplitude da atividade do nervo sudomotor refletindo a frequência dos potenciais de ação [12, 40]. Além disso, a amplitude do SCR mostrou estar relacionada com o número de glândulas sudoríparas recrutadas [38].

A atividade eletrodérmica (EDA) é altamente sensível a uma grande variedade de estados psicofisiológicos (atenção, processamento de informação, respostas emocionais, dor e excitação, etc) sendo capaz de revelar ocorrências de eventos psicologicamente marcantes [41]. Contudo, a EDA registra ainda padrões não psicofisiológicos (atividade física aumentada, umidade, variações de temperatura) e pode ser corrompida ainda por ruído elétrico, e movimentos do dispositivo de medição [42]. Tais artefatos podem impedir avaliações psicofisiológicas por parte de especialistas, se não forem cuidadosamente tratados e removidos [43].

Para uma aferição mais precisa sobre os estímulos emocionais, normalmente, o sinal de EDA é analisado em conjunto com outras atividades biológicas, como o eletrocardiograma (ECG), o eletroencefalografia (EEG), fluxo respiratório, dentre outros. Um bom exemplo de aplicação a fim de verificar as variações nos estímulos emocionais de um conjunto de sinais biomédicos se remete aos polígrafos - mais conhecidos como detectores de mentira, são instrumentos que monitoram determinadas reações de uma pessoa. Eles medem alterações das batidas do coração, pressão arterial, respiração e atividade eletrodérmica. E são essas mudanças fisiológicas que são analisadas pelos especialistas para detectar variações comportamentais que podem indicar mentira, estresse e ansiedade [44].

A aquisição de dados referente a condutividade da pele (por norma em micro-Siemens μS ou micro-Mho $\mu\Omega$) é normalmente realizada através de dois eletrodos em contato com a pele onde é aplicado uma diferença de potencial e, com isso, uma pequena quantidade de corrente elétrica flui no corpo entre dois pontos na superfície da pele, geralmente nas pontas dos dedos ou no pulso. A Figura 1 ilustra a pulseira com eletrodos capazes de medir a atividade eletrodérmica.

1.2 Componentes do Sinal de EDA

Há duas componentes principais presentes na atividade eletrodérmica: a resposta tônica e resposta fásica, que podem ser estudadas/analizadas separadamente, uma vez que apresentam características e propósitos distintos [13]:

- **Componente tônica** (*SCL - Skin Conductance Level*): representa uma variação



Figura 1: Dispositivo de captação de atividade eletrodérmica (pulseira Affective Q Sensor) [45].

mais lenta do sinal, ascendente ou descendente, da condutância da pele (SCL). Normalmente essa medida é pensada como mudanças gerais na excitação autônoma. Os níveis de componente tônica variam normalmente de 2 a 20 μS na maioria dos adultos [41].

- **Componente fásica:** responsável por alterações relativamente rápidas no sinal GSR (*GSR- Galvanic Skin Response*), conhecidas como respostas de condutância da pele (SCRs). As SCRs são flutuações ou picos rápidos que podem ser observados em um sinal GSR. A SCR normalmente indica o estado emocional do indivíduo e é a componente usada por especialistas em suas avaliações psicofisiológicas [43].

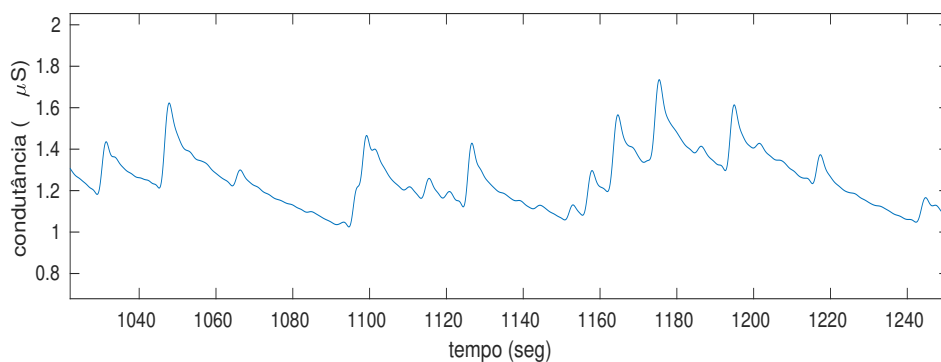


Figura 2: Medição de um sinal de EDA.

A Figura 2 ilustra uma medição de um sinal de EDA. As SCRs podem caracterizar respostas a eventos específicos (um estímulo visual ou uma pergunta inesperada, por exemplo) e são conhecidos como SCRs relacionadas a eventos (*ER-SCR-Event Related Skin Conductance Response*). As ER-SCRs são usadas em pesquisas para relacionar as mudanças na excitação emocional a estímulos específicos.

As capturas também podem aparecer espontaneamente, não estando relacionadas a qualquer evento específico, conhecida como SCR não-específica (NS-SCR). A frequência de NS-SCRs pode variar entre os participantes, ocorrendo em média de 1 a 3 vezes por minuto. Os altos níveis de excitação também são associados a uma maior taxa de NS-SCRs [46]. O diagrama sinóptico da Figura 3 apresenta as principais características das componentes tônica e fásica de sinais de EDA.

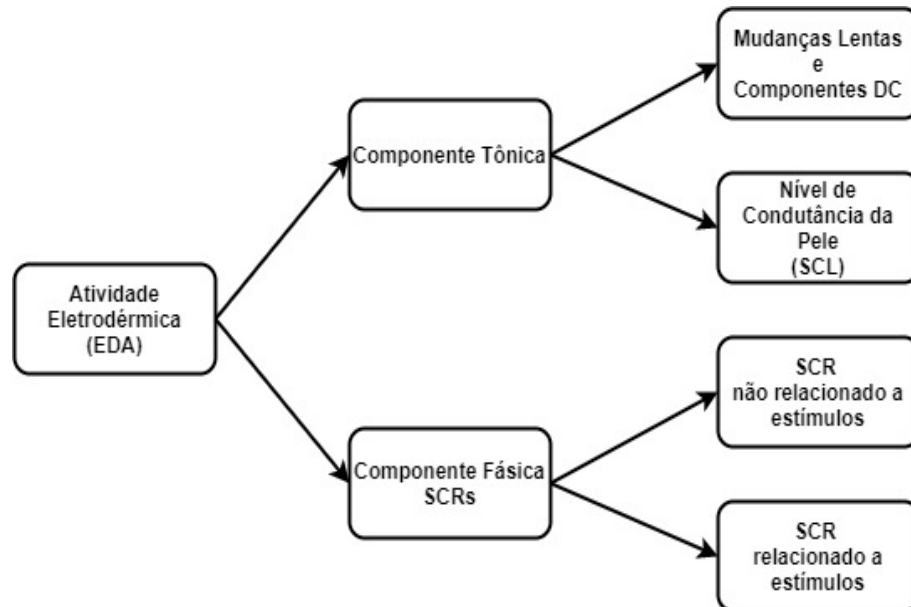


Figura 3: Diagrama sinóptico das componentes de um sinal de EDA.

Em geral, separa-se a componente tônica do sinal de EDA para a análise com o objetivo de reconhecer padrões ligados a estados psicofisiológicos, já que esta componente normalmente não fornece informação clara ou útil para diagnósticos [43]. Assim, em análise de sinais de EDA com o objetivo de identificar padrões psicofisiológicos normalmente considera-se a resposta fásica, que usualmente segue um padrão bem definido, como ilustrado na Figura 4. Esse padrão é composto por três partes: tempo de subida, pico da resposta e meio período de recuperação. O formato da SCR é caracterizado por um aumento acentuado do sinal e uma recuperação lenta [31]. A amplitude de uma SCR

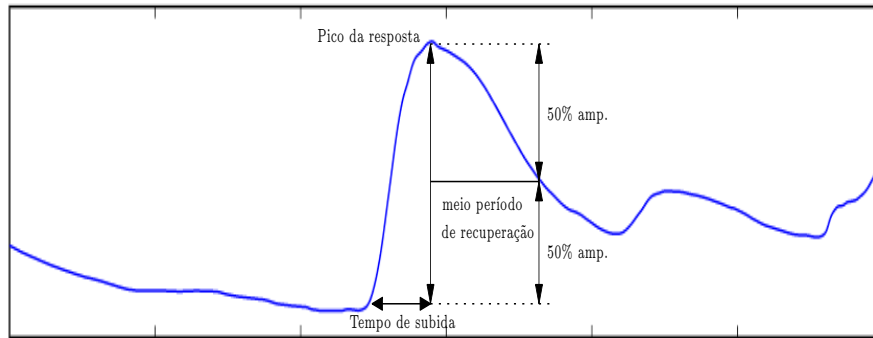


Figura 4: Análise de parâmetros de SCRs em um sinal de EDA.

está relacionada com o aumento de condutância desde o início da resposta ao seu pico com valores típicos que variam entre 0,1 e 1,0 μS [41]. Para adultos saudáveis, o tempo de subida se situa normalmente entre 1 e 3 segundos, e o meio período de recuperação entre 2 e 10 segundos [42]. É possível ocorrerem SCRs compostas, em que uma SCR surge antes da completa recuperação da SCR anterior.

1.3 Aquisição de dados

Neste trabalho, a base de dados de sinais EDA *Physionet* [3] é empregada. Foram extraídos dessa base 20 sinais de EDA com aproximadamente 38 minutos de duração cada, a taxa de amostragem é de 8 Hz.

A configuração experimental para obter os sinais de EDA envolve 20 pessoas, sendo 14 do sexo masculino e 6 do sexo feminino. A estas foram solicitadas o uso de um dispositivo capaz de medir a condutância da pele, ilustrado pela Figura 1, enquanto realizavam determinadas tarefas. O propósito desse experimento foi avaliar como o corpo humano de cada indivíduo se comporta a cada situação. As tarefas foram realizadas na seguinte ordem:

1. **Primeiro relaxamento:** Período de 5 minutos.
2. **Estresse físico:** Ficar de pé por 1 minuto, caminhar na esteira a 2 km/h por 2 minutos, em seguida, caminhar a 5 km/h por mais 2 minutos.
3. **Segundo relaxamento:** Período de 5 minutos.
4. **Estresse cognitivo:** Contar de trás para frente de 7 em 7, começando em 2485

por 3 minutos. Foi realizado também o teste de ler palavras com nomes de cores diferentes das cores apresentadas, proferindo-se, assim, a cor da palavra e não a cor de fato. Em ambos os testes, foi utilizado um alarme para avisar sobre o erro.

5. **Terceiro relaxamento:** Período de 5 minutos.
6. **Estresse emocional:** Assistir a um trecho de um filme de terror por um período de 5 minutos.
7. **Quarto relaxamento:** 5 minutos de duração.

Durante os quatro períodos de relaxamento, os voluntários foram solicitados a sentar-se confortavelmente e ouvir sons específicos usados para meditação ou concentração (*Binaural*). O objetivo do relaxamento entre as tarefas é estabelecer uma linha base para as medidas fisiológicas [3].

1.4 Pré-processamento

Como foi visto, a captação do sinal de EDA pode conter componentes ruidosas. Além disso, também foi verificada a importância da separação da componente fásica do restante do sinal a ser usado nas próximas etapas do projeto. Devido a isso, é necessário realizar um pré-processamento, empregando-se um filtro com a finalidade de remover as componentes de frequências indesejadas na medição [2]. Como é sabido que o sinal de EDA se concentra em uma faixa de baixas frequências, usa-se um filtro passa-baixa para a remoção dessas componentes indesejadas de altas frequências, mantendo-se uma boa qualidade do sinal.

A componente fásica do sinal de EDA contém a informação relevante para reconhecimento de padrões intrínsecos fundamentais para a análise psicofisiológica do paciente. Essa componente, como visto na Seção 1.2, apresenta uma variação mais abrupta e está relacionada à resposta da condutância da pele. Já a componente tônica apresenta uma variação lenta, crescente ou decrescente na condutância da pele ao longo do tempo e normalmente, ao se analisar um sinal de EDA, essas componentes são separadas.

Devido a componente tônica não trazer consigo informação relevante quanto ao diagnóstico psicofisiológico, é interessante lançar mão de métodos capazes de estimar a componente tônica, para posteriormente subtraí-la do sinal, obtendo-se assim, a componente fásica. Neste trabalho, extrai-se a componente tônica usando o Método dos Mínimos

Quadrados (MMQ) [47] e otimização convexa (cvxEDA) [18] para obter as retas ou curvas que a aproximam.

1.5 Método dos Mínimos Quadrados

O método dos mínimos quadrados é um método bastante usado para aproximar soluções de sistemas por meio de regressões, minimizando-se a soma dos quadrados dos resíduos (diferença entre um valor observado e o valor previsto pelo modelo), que corresponde a encontrar uma reta ou curva que melhor se adapta a um conjunto de dados.

Devido ao comportamento lento da componente tônica de um sinal de EDA, contendo variações crescentes ou decrescentes ao longo do tempo é fundamental lançar mão desse método para estimar a componente tônica de um sinal EDA. Um ponto importante na utilização deste método é que não se pode estimar essa reta para o sinal inteiro, em função dessas variações crescentes e decrescentes. Com isso, é necessário dividir o sinal em blocos (janelas), ajustando-se uma reta a cada bloco. Dessa maneira, a componente tônica do sinal é aproximada linearmente por partes. Um cuidado a ser tomado é quanto ao tamanho do bloco. Se o tamanho do bloco for muito pequeno é possível que a regressão estime a componente fásica, o que não é desejado. Em contrapartida, janelas muito grandes podem não gerar uma estimativa coerente.

Na formulação padrão, um conjunto de N pares de observações $\{Y_i, X_i\}$ é usado para encontrar uma função que relaciona o valor da variável dependente (Y) com valores de uma variável independente (X). Com uma variável e uma função linear, a previsão é dada pela seguinte Equação

$$\hat{Y} = aX + b. \quad (1)$$

Esta Equação envolve dois parâmetros livres que especificam o coeficiente angular (a) e coeficiente linear (b) da linha de regressão. O MMQ define a estimativa destes parâmetros como os valores que minimizam a soma dos quadrados. Isto equivale a minimizar a expressão

$$S = \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 \quad (2)$$

Transformando para a forma matricial $\hat{\mathbf{Y}} = \mathbf{X}\boldsymbol{\beta}$ onde $\boldsymbol{\beta} = [a \ b]^T$ é o vetor contendo os

coeficientes linear (b) e angular (a) e substituindo (1) em (2) temos

$$\|\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^\top (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) = \mathbf{Y}^\top \mathbf{Y} - \mathbf{Y}^\top \mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Y} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\beta} \quad (3)$$

Calculando o gradiente na Equação (3) podemos obter

$$\frac{\partial (\mathbf{Y}^\top \mathbf{Y} - \mathbf{Y}^\top \mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Y} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{X}\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -2\mathbf{X}^\top \mathbf{Y} + 2\mathbf{X}^\top \mathbf{X}\boldsymbol{\beta} \quad (4)$$

Igualando-se o gradiente da perda a zero e resolvendo-se $\boldsymbol{\beta}$, obtém-se

$$-2\mathbf{X}^\top \mathbf{Y} + 2\mathbf{X}^\top \mathbf{X}\boldsymbol{\beta} = \mathbf{0} \Rightarrow \mathbf{X}^\top \mathbf{Y} = \mathbf{X}^\top \mathbf{X}\boldsymbol{\beta} \quad (5)$$

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{Y} \quad (6)$$

Seja um sinal \mathbf{x} . Este é dividido em blocos e, em cada bloco, calcula-se a reta que melhor se ajusta ao sinal, sendo que a primeira amostra da reta (numericamente equivalente ao coeficiente linear da reta) corresponde à última amostra da reta obtida para o bloco anterior. O método MMQ realizado a cada bloco do sinal é descrito em Algoritmo 1.

Algoritmo 1 MMQfunção MMQ(\mathbf{X} , \mathbf{Y})**início**

Define tamanho do bloco

numeroDeBlocos = tamanhoDoSinal/tamanhoDoBloco

para $nBloco = 1; nBloco \leq numeroDeBlocos; nBloco = nBloco + 1$ **faça** **se** $nBloco == 1$ **então**

$$\beta = (\mathbf{X}^T * \mathbf{X})^{-1} * (\mathbf{X}^T * \mathbf{Y})$$

$$\hat{\mathbf{Y}} = \mathbf{X} * \beta$$

fim **senão**

$$\beta = (\mathbf{X}^T * \mathbf{X})^{-1} * (\mathbf{X}^T * (\mathbf{Y} - \mathbf{bfinal}))$$

$$\hat{\mathbf{Y}} = \mathbf{X} * \beta$$

$$\hat{\mathbf{Y}} = \hat{\mathbf{Y}} + \mathbf{bfinal}$$

fim $\mathbf{bfinal} = \hat{\mathbf{Y}}[-1]$ // coeficiente linear do final do bloco **fim****fim**

Os valores dos coeficientes linear e angular de cada bloco são armazenados, caso haja a necessidade de reconstrução do sinal após as etapas de decomposição ou uma possível compressão. Em seguida, remove-se a componente tônica, subtraindo-se do sinal original as retas obtidas por regressão linear em cada bloco. A Figura 5 mostra o diagrama de blocos referente ao procedimento de pré-processamento e separação das componentes física e tônica utilizando o método MMQ.



Figura 5: Diagrama de blocos do processo de separação do sinal de EDA em suas componentes física e tônica.

1.6 Otimização Convexa

Nesta seção, é apresentado um método para estimar a atividade eletrodérmica usando uma otimização convexa desenvolvido em [18]. O modelo é baseado em estatísticas Bayesianas e uma representação simples, mas fisiologicamente sólida, do fenômeno observado por meio da SC (*Skin Conductance*) como a soma de três componentes: uma componente tônica considerada lenta; a componente fásica representada por uma saída da convolução entre um IRF (*Impulse Response Function*) e uma sequência esparsa de impulsos não negativos associados, a SMNA (*Sudomotor Nerve Activity*); e um termo de ruído branco Gaussiano incorporando erros de previsão do modelo. Modelou-se o IRF, no qual está relacionado à componente fásica, como uma resposta ao impulso infinita (IIR) por meio de um modelo ARMA (*Autoregressive-Moving-Average*). Comparado a uma resposta ao impulso finita (FIR) ou filtro combinado, o sistema IIR apresenta melhor precisão (por não exigir truncamento do IRF) e maior eficiência computacional, graças a uma representação mais compacta do sistema.

O objetivo da otimização é selecionar o melhor vetor de um subconjunto de \mathbb{R}^n definido por funções de restrição. Um problema de otimização é considerado convexo quando tanto a função objetivo quanto as funções de restrição são convexas [48]. A principal vantagem da convexidade na otimização matemática é que as condições necessárias para o mínimo local são suficientes para a obtenção do mínimo global. Isso significa que se uma função convexa for minimizada, a solução global ótima sempre será obtida. Ao contrário de problemas não convexos, existem algoritmos para certas categorias importantes de problemas de otimização convexa que podem resolver efetivamente problemas em grande escala. A otimização quadrática (QP) é uma subclasse específica de problemas de otimização convexa em que a função objetivo é um polinômio quadrático das variáveis de decisão e as restrições são todas funções afins [48]. Sua formulação base pode ser dada como

$$\min \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{l}^T \mathbf{x} \text{ sujeito à } \begin{cases} \mathbf{C} \mathbf{x} \leq \mathbf{b}, \\ \mathbf{C}_{\text{eq}} \mathbf{x} = \mathbf{b}_{\text{eq}} \end{cases} \quad (7)$$

onde $\mathbf{x} \in \mathbb{R}^n$ é um vetor real com as variáveis a serem minimizadas, $\mathbf{Q} \in \mathbb{R}^{n \times n}$ representa a matriz real simétrica com os termos de segunda ordem das variáveis, $\mathbf{l} \in \mathbb{R}^n$ um vetor com os termos de primeira ordem, $\mathbf{C} \in \mathbb{R}^{m \times n}$ e $\mathbf{b} \in \mathbb{R}^m$ são a matriz e o vetor das m restrições de desigualdade, enquanto que $\mathbf{C}_{\text{eq}} \in \mathbb{R}^{q \times n}$ e $\mathbf{b}_{\text{eq}} \in \mathbb{R}^q$ são a matriz e vetor das

q restrições de igualdade.

A função de custo do algoritmo QP também pode incluir termos de regularização de norma l_2 e norma l_1 que podem ser empregados para evitar o *overfitting* ou para promover soluções esparsas, com um grande número de componentes iguais a zero. Usando suposições fisiologicamente plausíveis, é possível mostrar que o problema de desconvolução de EDA pode ser transformado em um problema de otimização quadrática. Para isso, é necessário fazer algumas suposições com base nas características da atividade eletrodérmica para chegarmos em um modelo matemático que represente o sinal estudado:

1. SCRs são rajadas de impulsos discretas no domínio do tempo geradas pelos nervos sudomotores naturalmente de característica esparsa e não negativa [12, 38].
2. É possível caracterizar o modelo de um sinal EDA como um sistemas linear invariante no tempo, pois há uma relação linear entre a quantidade de glândulas sudoríparas expostas aos sensores e a amplitude de um SCR [38]. Além disso, a resposta de saída do sistema depende apenas do instante em que a entrada do nervo é aplicada. Em outras palavras, o curso de tempo de um única rajada SCR não é influenciada por outras anteriores, mesmo quando seus SCRs se sobreponham [16].
3. O processo de difusão do suor tem uma função específica de resposta ao impulso (IRF) que é relativamente estável para todos os SCRs do mesmo paciente [15].
4. A componente fásica possui espectro de frequência abaixo de 0,05 Hz, sendo assim possível representar amostras espaçadas a cada 10 s [17, 31].

1.6.1 Modelo

Dado um sinal de EDA \mathbf{y} de N amostras como a soma de três componentes: tônica \mathbf{t} , fásica \mathbf{r} e ruído $\boldsymbol{\epsilon}$, tem-se que

$$\mathbf{y} = \mathbf{t} + \mathbf{r} + \boldsymbol{\epsilon}. \quad (8)$$

O termo $\boldsymbol{\epsilon}$ pode ser representado como ruído branco gerado através da diferença entre o sinal \mathbf{y} e as estimativas feitas para as componentes fásica e tônica. A componente tônica é representada como a soma de funções *B-spline* com nós igualmente espaçados a cada

10 segundos (suposição 4), um *offset* (coeficiente linear) e coeficiente angular

$$\mathbf{t} = \mathbf{B}\mathbf{l} + \mathbf{C}\mathbf{d} \quad (9)$$

onde \mathbf{B} é matriz $N \times V$ cuja as colunas são as V funções base da *B-spline*, \mathbf{l} é o vetor $V \times 1$ dos coeficientes das *splines*, \mathbf{C} é uma matriz $N \times 2$ com $C_{i,1} = 1$ e $C_{i,2} = 1/N$, sendo i o índice da linha, \mathbf{d} é um vetor de dimensão 2×1 com o *offset* e o coeficiente angular. A resposta física \mathbf{r} é modelada usando uma função biexponencial, chamada de função *Bateman* (suposições 2 e 3). A principal razão de utilizarmos essa função para representar a componente física é por ela ser considerada a saída de um modelo farmacocinético bicomportamental, representando a difusão do suor através do dutos glandulares [49]. Tal função também será usada na seção de decomposição atômica como subdicionário. A função de *Bateman* pode ser definida como

$$h(t) = (e^{-at} - e^{-bt})u(t) \quad (10)$$

onde a e b são as constantes de decaimento e crescimento respectivamente e $u(t)$ representa o degrau unitário. A transformada de Laplace de (65) é dada como

$$\mathcal{L}\{h(t)\} = \frac{1}{s+a} - \frac{1}{s+b} \quad (11)$$

onde a e b são polos de segunda ordem de um sistema LTI (*Linear Time Invariant*). Aplicando a transformada bilinear no tempo discreto temos $s = \frac{2}{\delta} \frac{z-1}{z+1}$ com δ sendo o intervalo de amostragem. Assim temos o seguimos modelo ARMA dado como

$$\begin{aligned} H(z) &= \frac{(1+z^{-1})^2}{\psi + \theta^{-1} + \zeta^{-2}} \\ \psi &= (a\delta + 2)(b\delta + 2) / (b\delta^2 - a\delta^2) \\ \theta &= (2ab\delta^2 - 8) / (b\delta^2 - a\delta^2) \\ \zeta &= (a\delta - 2)(b\delta - 2) / (b\delta^2 - a\delta^2) \end{aligned} \quad (12)$$

O modelo ARMA pode ser representada na forma de matricial como

$$\mathbf{q} = \mathbf{A}^{-1}\mathbf{p}, \quad \mathbf{r} = \mathbf{M}\mathbf{q} \quad (13)$$

onde \mathbf{p} representa a atividade sudomotora, \mathbf{q} é a variável auxiliar que será usada para calcular-se \mathbf{p} indiretamente, \mathbf{M} é uma matriz tridiagonal com elementos $M_{i,i} = M_{i,i-2} = 1$, $M_{i,i-1} = 2$, $3 \leq i \leq N$; e \mathbf{A} é uma matriz tridiagonal com elementos $A_{i,i} = \psi$, $A_{i,i-1} = \theta$, $A_{i,i-2} = \zeta$, $3 \leq i \leq N$. Com isso, podemos escrever a Equação (8) como

$$\mathbf{y} = \mathbf{M}\mathbf{q} + \mathbf{B}\mathbf{l} + \mathbf{C}\mathbf{d} + \epsilon \quad (14)$$

1.6.2 *Maximum a Posteriori Estimation*

Dada a Equação (14), o objetivo agora é identificar o máximo a posteriori (MAP) de \mathbf{p} e \mathbf{t} parametrizado por $[\mathbf{q}, \mathbf{l}, \mathbf{d}]$, para a condutância da pele \mathbf{y} :

$$[\mathbf{q}, \mathbf{l}, \mathbf{d}] = \arg \max P[\mathbf{q}, \mathbf{l}, \mathbf{d} | \mathbf{y}] \quad (15)$$

assumindo independência entre \mathbf{q} , \mathbf{l} e \mathbf{d} (ou seja, entre a componente fásica e a componente tônica de variação lenta) e aplicando o teorema de Bayes, obtemos

$$P[\mathbf{q}, \mathbf{l}, \mathbf{d} | \mathbf{y}] \propto P[\mathbf{y} | \mathbf{q}, \mathbf{l}, \mathbf{d}]P[\mathbf{q}]P[\mathbf{l}]P[\mathbf{d}] \quad (16)$$

onde $P[\mathbf{y} | \mathbf{q}, \mathbf{l}, \mathbf{d}]$ é a probabilidade de se observar uma série temporal da condutância da pele dados os parâmetros do modelo, enquanto $P[\mathbf{q}]$, $P[\mathbf{l}]$ e $P[\mathbf{d}]$ são as probabilidades dos parâmetros a priori. A evidência $P[\mathbf{y}]$ é omitida uma vez que ela não desempenha nenhum papel na otimização. Para modelar a atividade sudomotora \mathbf{p} representando a entrada (suposição 1) para o sistema LTI, usa-se o sistema mais simples de primeira ordem de trem de impulsos [50], ou seja, uma distribuição de Poisson: onde $\lambda\delta$ é a taxa de disparo esperada por amostra, ou seja, λ é o número médio de picos por unidade de tempo. Para manter a análise tratável, substitui-se a distribuição de Poisson por uma distribuição exponencial de mesma média [50]. Finalmente, como \mathbf{p} e \mathbf{q} estão relacionados por [19], a probabilidade a priori $P[\mathbf{q}]$ se torna

$$P[\mathbf{q}] = \prod_{i=1}^N \frac{1}{\lambda\delta} e^{-\frac{p_i}{\lambda\delta}} \quad (17)$$

$$P[\mathbf{q}] = \prod_{i=1}^N e^{(-\frac{1}{\lambda\delta}(\mathbf{A}\mathbf{q})_i)} \quad (18)$$

Relativamente à componente tônica, fazemos uso da suposição 4 e consideramos um espectro de frequência uniforme no banda 0-0,05 Hz. Porque é usado nós igualmente espaçados a cada $\Delta = 10$ segundos, a frequência de amostragem F_s é exatamente o dobro da limite de banda e os elementos do vetor \mathbf{l} podem ser assumidos IID. É adotado uma distribuição normal para a amplitude em cada nó $l_i \sim \mathcal{N}(0, \sigma_l^2)$. Como resultado, o $P[\mathbf{l}]$ anterior é definido como

$$P[\mathbf{l}] = \prod_{i=1}^Q \frac{1}{\sqrt{2\pi}\sigma_l} e^{\left(-\frac{l_i^2}{2\sigma_l^2}\right)} \quad (19)$$

onde Q é o número de nós (aproximadamente $N\delta/\Delta$). Para os coeficientes de \mathbf{d} , assume-se que $P[\mathbf{d}]$ não seja relevante e é desconsiderado. O termo $P[\mathbf{y} | \mathbf{q}, \mathbf{l}, \mathbf{d}]$ segue imediatamente a Equação (14), com $\epsilon \sim \mathcal{N}(0, \sigma^2)$ da seguinte forma

$$P[\mathbf{y} | \mathbf{q}, \mathbf{l}, \mathbf{d}] = \prod_{i=1}^N \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{(\mathbf{M}\mathbf{q} + \mathbf{B}\mathbf{l} + \mathbf{C}\mathbf{d} - \mathbf{y})_i^2}{2\sigma^2}\right)} \quad (20)$$

Substituindo as equações (18), (19), (20) em (16) e fazendo o logaritmo

$$P[\mathbf{q}, \mathbf{l}, \mathbf{d} | \mathbf{y}] = -\frac{1}{2\sigma^2} \sum_{i=1}^N (\mathbf{M}\mathbf{q} + \mathbf{B}\mathbf{l} + \mathbf{C}\mathbf{d} - \mathbf{y})_i^2 - \frac{1}{\lambda\delta} \sum_{i=1}^N (\mathbf{A}\mathbf{q})_i - \frac{1}{2\sigma_l^2} \sum_{i=1}^Q l_i^2 \quad (21)$$

com $(\mathbf{A}\mathbf{q})_i \geq 0$. Com isso temos a solução MAP descrita pela Equação (15) ao maximizar a Equação (21). Além disso, é possível simplificar a Equação (21) como um problema de minimização restrita na forma matricial para obter uma notação mais compacta se multiplicarmos por σ^2 e substituir $\alpha = \sigma^2/(\lambda\delta)$ e $\gamma = \sigma^2/\sigma_l$, obtendo assim

$$\min -\frac{1}{2} \|\mathbf{M}\mathbf{q} + \mathbf{B}\mathbf{l} + \mathbf{C}\mathbf{d} - \mathbf{y}\|_2^2 + \alpha \|\mathbf{A}\mathbf{q}\|_1 + \frac{\gamma}{2} \|\mathbf{l}\|_2^2 \quad | \mathbf{A}\mathbf{q} \geq 0 \quad (22)$$

Depois de alguma álgebra matricial, este problema de otimização pode ser reescrito na forma de um problema QP (*Quadratic Program*) dado pela Equação (7) padrão e resolvido com o algoritmo de otimização QP. Depois de encontrar o valor ótimo de $[\mathbf{q}, \mathbf{l}, \mathbf{d}]$, a componente tônica \mathbf{t} pode ser derivado de (9), enquanto a atividade do nervo sudomotor conduzindo a componente fásica pode ser facilmente encontrado como $\mathbf{p} = \mathbf{A}\mathbf{q}$. Embora resolver (22) seja estritamente equivalente a maximizar (21), o primeiro tem uma interpretação diferente. A função objetivo a ser minimizada é uma medida quadrática de desajuste entre o previsto e os dados observados. O conhecimento prévio é contabilizado

por meio de termos de regularização aditiva. Por exemplo, a natureza impulsiva da entrada (suposição 1) é aplicada por meio de a penalização da norma l_1 , que é uma forma eficaz de prover esparsidade a um sinal mantendo-se a convexidade [51, 52]. A suavidade da curva tônica (suposição 4) é reforçada pela escolha da base (\mathbf{B}) e através da penalização da norma l_2 dos coeficientes das *splines*. Os dois parâmetros α e γ controlam a intensidade de penalidade para as componentes fásicas e tônicas, respectivamente. Um grande α (regularização l_1 mais forte de \mathbf{p}) produz uma estimativa mais esparsa, mas também mais distorção de sinal (ou seja, atenuação de ativações genuínas). Por outro lado, um pequeno α produz um solução menos distorcida, porém mais ruidosa. Em relação a γ , valores mais altos significam uma penalização mais forte de \mathbf{l} , ou seja, uma curva tônica mais suave.

1.7 Simulações e Resultados

1.7.1 Pré-processamento

Nesta seção, apresenta-se o procedimento para realizar o pré-processamento do sinal de EDA usando as ferramentas computacionais e os métodos abordados neste capítulo. As simulações computacionais foram realizadas através do Matlab[®] (ferramenta computacional para análise de dados, desenvolvimento de algoritmos e criação de modelos) [53]. Inicialmente, é necessário realizar a leitura de todos os sinais disponíveis na base de dados da *Physionet*. A Figura 6 ilustra um segmento de um sinal de EDA captado pelo sensor de medição. Nota-se que é possível visualizar artefatos (“serrilhados”) captados pelo sensor que indicam ruídos externos e ou até mesmo interferências geradas pelo próprio corpo não referentes à atividade sudomotora. Como já mencionado, a atividade eletrodérmica ocupa uma faixa de frequência bem baixa, enquanto artefatos ruidosos normalmente são de alta frequência. A Figura 7 ilustra a densidade espectral de potência de um sinal EDA com maior percentual de energia concentrado de zero até 0,5 Hz. Devido a isso, aplicou-se um filtro passa-baixa com frequência de corte de 0,5 Hz com o objetivo de reduzir o ruído aditivo de medição do sinal de EDA. A Figura 8 apresenta o sinal filtrado em que o artefato de serrilhamento torna-se visualmente imperceptível.

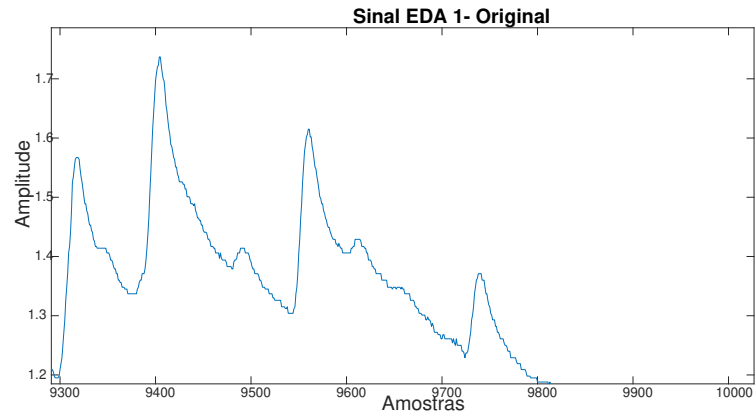


Figura 6: Sinal *eda1* original ampliado com artefatos da captação.

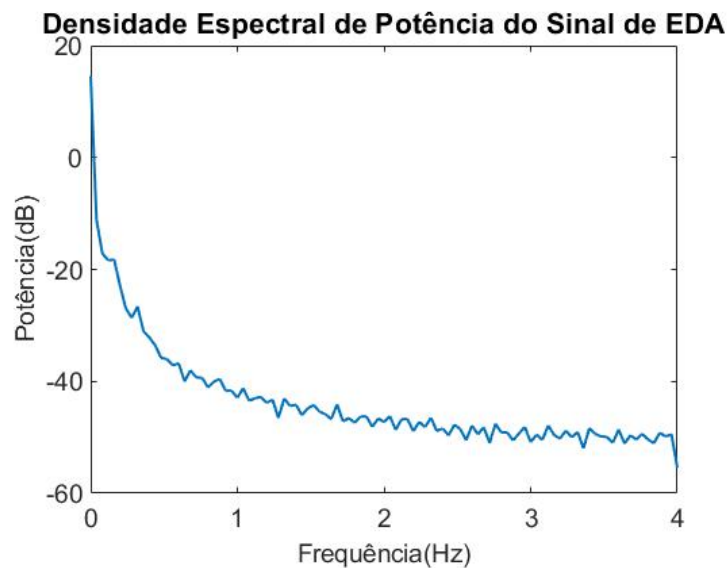


Figura 7: Densidade espectral de potência de um sinal EDA.

1.7.2 MMQ

Após a filtragem do sinal, realiza-se a separação do sinal de EDA em componentes tônica e fásica. Primeiramente, empregou-se o método dos mínimos quadrados (MMQ), onde cada sinal foi subdividido em blocos de tamanhos em potências de 2 (64, 128, 256, 512 e 1024 amostras). Para determinar o tamanho do bloco, foram calculadas as curvas geradas pelo MMQ que compõem cada bloco. Tendo em posse a potência do sinal original e a componente tônica, podemos usar a medida de razão sinal-ruído (SNR) para determinar a qualidade do sinal. A Tabela 1 ilustra o resultado de SNR médio em dB para diversos tamanhos de blocos e para os 20 sinais que compõem a base de dados usando a

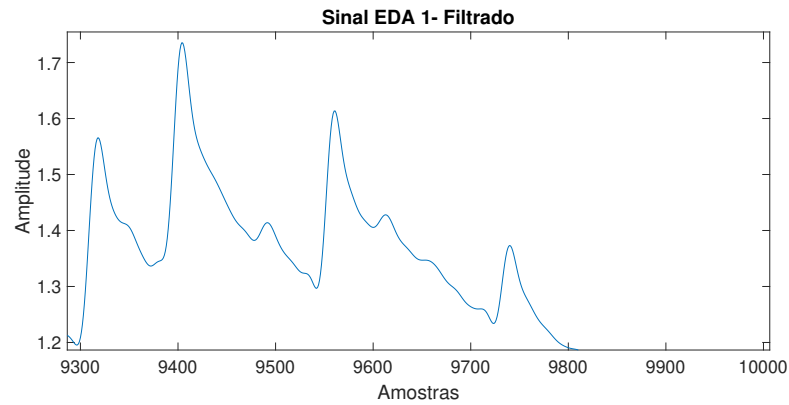


Figura 8: Sinal *eda1* ampliado e filtrado com um filtro passa-baixa.

regressão linear. É possível observar que a formação de blocos com 64 amostras obteve o melhor resultado quanto à qualidade da estimativa da componente tônica dentre os testes realizados. Contudo temos que levar em conta se essa estimativa não está englobando a componente fásica. Pela Figura 9 é possível observar a variação da estimativa da componente tônica conforme a variação do tamanho do bloco. Usando janelas de 64 amostras, observamos que as retas acompanham o sinal original, inclusive algumas variações mais abruptas, as quais são características da componente fásica. Com o aumento do tamanho da janela para 128 e 256 amostras é possível notar que a estimativa da componente tônica se torna melhor quanto a evitar representar as variações abruptas. Contudo, ao longo do sinal, é possível observar alguns momentos de subida rápida não relacionadas ao SCR. Nesses casos, a regressão linear faz uma boa aproximação, gerando erros maiores conforme o aumento do tamanho da janela.

Tabela 1: SNR médio dos sinais de EDA usando MMQ para estimar a componente tônica.

Tamanho do Bloco	SNR médio (dB)
64	$31,34 \pm 8,1$
128	$29,04 \pm 7,8$
256	$27,4 \pm 7,6$
512	$24,7 \pm 7,9$
1024	$20,8 \pm 8,6$

1.7.3 CVXEDA

Agora, aplicando-se o algoritmo “cvxEDA” [18] de código aberto desenvolvido em Matlab[®], é possível estimar as componentes tônica e fásica através de otimização con-

vexa. Quanto à estimativa da componente tônica, o que o difere do método dos mínimos quadrados é que em vez de usarmos retas como método de aproximação, são usadas *splines* cúbicas que normalmente apresentam uma melhor aproximação, convergência e estabilidade. Como parâmetros de entrada do algoritmo temos o próprio sinal EDA y , $\delta = 10s$ como intervalo de tempo entre as amostras, $a_0 = 2$ e $a_1 = 0,7$, coeficientes de subida e descida da função de *Bateman*, respectivamente, $\alpha = 0,0008$ é a penalização para os coeficientes da componente fásica e $\gamma = 0,01$ é a penalização para os coeficientes da componente tônica. Como saída, temos as componentes fásica, tônica e o erro residual dado por $\epsilon = \mathbf{y} - \mathbf{r} - \mathbf{t}$. Para avaliar os resultados de ambos algoritmos vamos comparar visualmente as componentes fásica e tônica de cada método sobrepostas, conforme apresentado pela Figura 10. Pela análise gráfica, podemos observar que a componente tônica gerada pelo método MMQ consegue realizar uma boa estimativa da componente tônica com exceção de momentos que o SCL sofre mudanças rápidas não relacionadas com eventos de SCR. Outro fator que afeta o algoritmo MMQ é caso as amostras nas extremidades da janela forem picos de SCRs, gerará um erro onde a estimativa da componente tônica estaria acima da linha base. Este erro gerado acaba gerando uma componente fásica com valores negativos. Por outro lado, o algoritmo “cvxEDA” utiliza um modelo considerando a componente fásica sendo não negativa, em conjunto com a estimativa tônica modelada por *splines*. Com isso, temos uma componente tônica mais suave, devido ao maior número de graus do polinômio, que considera as variações lentas e rápidas não relacionadas ao SCR. Devido a isso, o método “cvxEDA” apresentou melhores resultados quanto a capacidade de estimar a componente tônica e por considerar um modelo matemático capaz de representar a componente fásica de forma mais fidedigna. Por fim, após definir o método mais adequado, vamos usar a componente fásica resultante do algoritmo “cvxEDA” para a etapa de decomposição atômica.

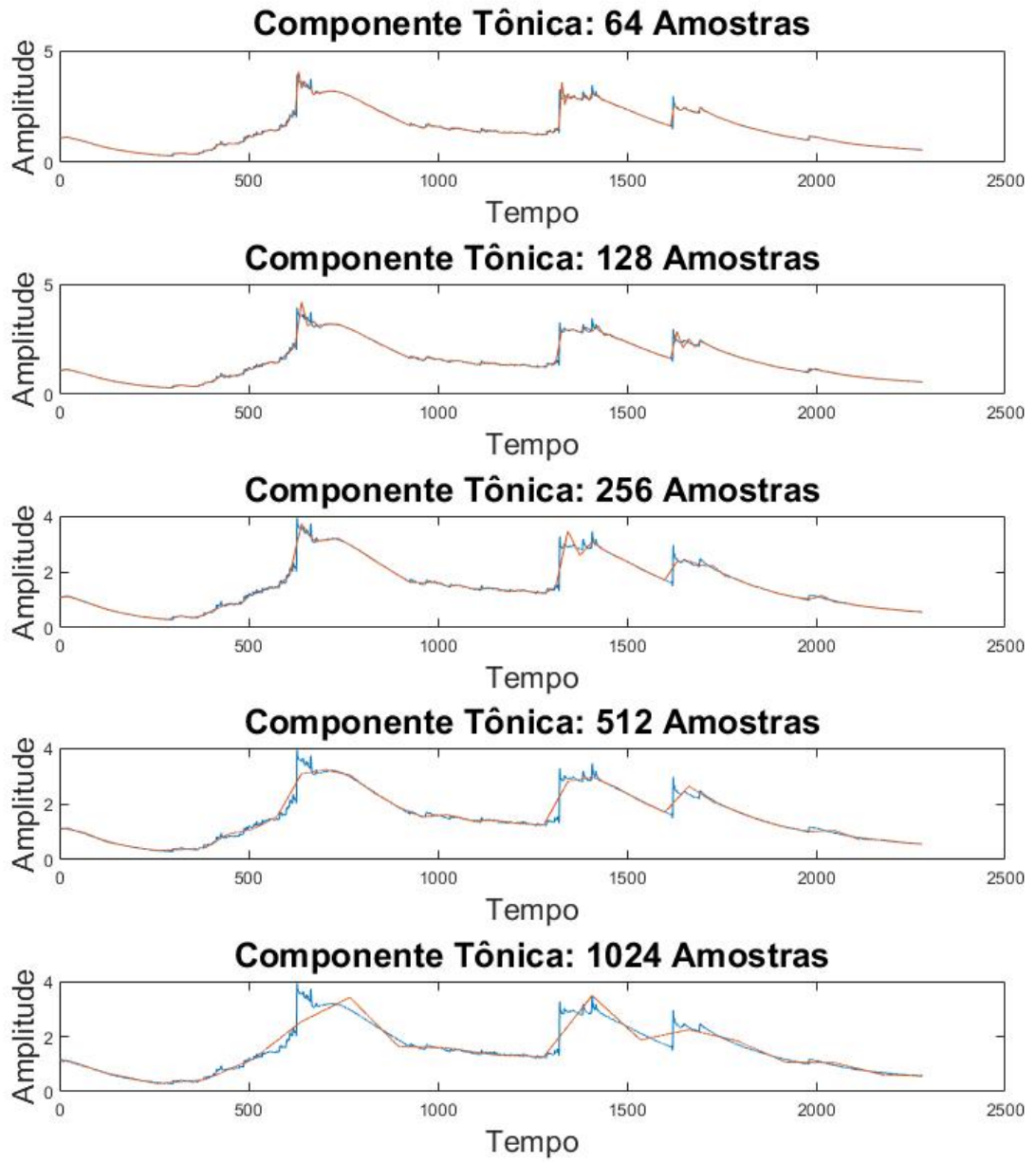


Figura 9: Estimativa da componente tônica pelo método MMQ para diferentes tamanhos de blocos.

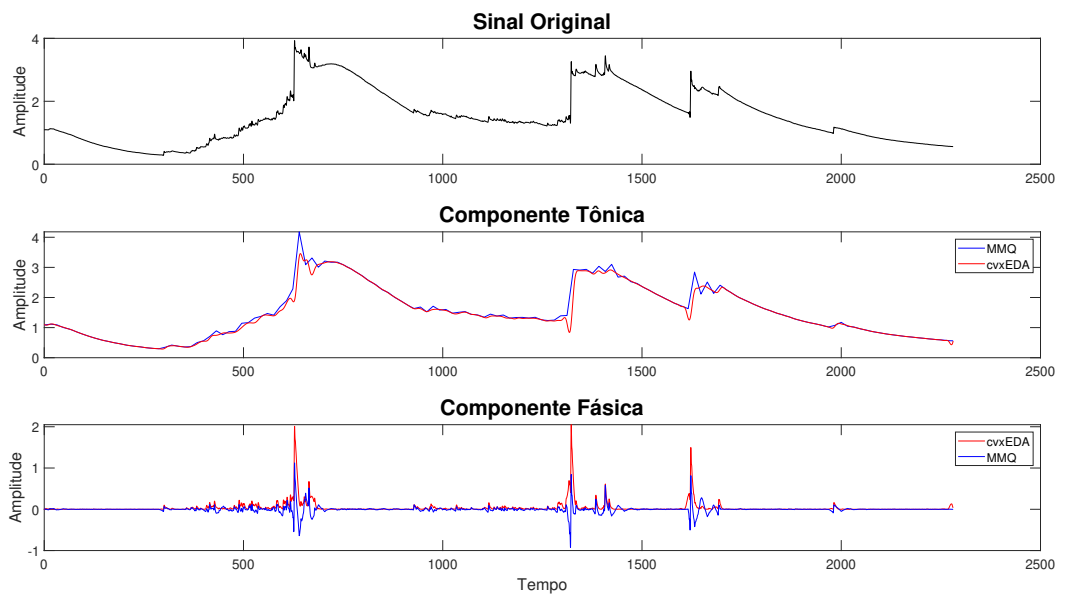


Figura 10: - Separação das componentes do sinal de EDA usando o algoritmo de otimização convexa (cvxEDA) [18] para estimar a componente tônica.

2 DECOMPOSIÇÕES ATÔMICAS DE SINAIS DE EDA

Neste capítulo, apresenta-se o processo de decomposição de sinais de EDA utilizando-se decomposições atômicas com base em dicionário redundante. Esse processo emprega um algoritmo iterativo e voraz conhecido como *Matching Pursuit* (MP) e utiliza um dicionário parametrizado [22] [21] [6] [7]. A cada iteração, seleciona-se um átomo do dicionário para ser adicionado à representação do sinal, assim decomposto em uma soma de átomos. A decomposição atômica assim pode ser empregada em tarefas de processamento de sinais como filtragem e compressão com perdas. Além disso, apresenta-se um teorema que garante uma representação exata do sinal, de probabilidade não-nula, após um número finito de iterações, se o dicionário empregado contiver uma base ortonormal [7].

2.1 Representação Atômica de Sinais

A decomposição atômica é uma técnica importante em processamento de sinais, em particular na análise tempo-frequência/escala de sinais não-estacionários. Essa técnica permite a decomposição de um sinal em suas componentes mais simples e básicas, chamadas de átomos, que são funções predefinidas com uma duração e frequência bem definidas. Através da decomposição atômica, é possível obter uma representação mais compacta do sinal, facilitando sua análise e processamento.

Esta análise pode ser exemplificada pelas séries de Fourier, pois estas são capazes de representar funções infinitas e periódicas complexas representativas de processos físicos, na forma de somas de funções trigonométricas simples, senos e cossenos; permitindo assim, simplificar a visualização e manipulação de funções complexas [54]. Um exemplo clássico refere-se a uma onda quadrada que pode ser decomposta em um somatório de senos e cossenos. Já na representação atômica, pode-se representar qualquer sinal \mathbf{x} de suporte finito periódico ou não periódico com formas de ondas preestabelecidas. Essas formas de onda normalmente são dadas por funções matemáticas já conhecidas e parametrizadas. As formas de ondas selecionadas a partir de um conjunto predefinido são chamadas de dicionário. Estes átomos podem ser escolhidos de diferentes formas como por exemplo: *Wavelets*, exponencial, senoides amortecidas [7], Gabor [6], e triangular [25]. A escolha do dicionário mais adequado varia conforme a aplicação e característica de cada sinal a ser estudado.

Podemos representar um sinal \mathbf{x} usando os átomos \mathbf{g}_γ pertencentes a um dicionário D via

$$\mathbf{x} \approx \hat{\mathbf{x}} = \sum_{i=1}^M \alpha_i \mathbf{g}_{\gamma(i)}, \quad \mathbf{g}_{\gamma(i)} \in D \quad (23)$$

Os átomos $\mathbf{g}_{\gamma(i)}$ são indexados por $\gamma(i)$, que é definido como $\gamma : \mathbb{Z}^+ \rightarrow \{1, \dots, \#D\}$, $\#D$ é a cardinalidade – o número de elementos do dicionário D , portanto $\gamma(i) \in \{1, \dots, \#D\}$. O parâmetro α_i é o coeficiente que pondera $\mathbf{g}_{\gamma(i)}$ e M corresponde ao número de átomos selecionados para representar \mathbf{x} .

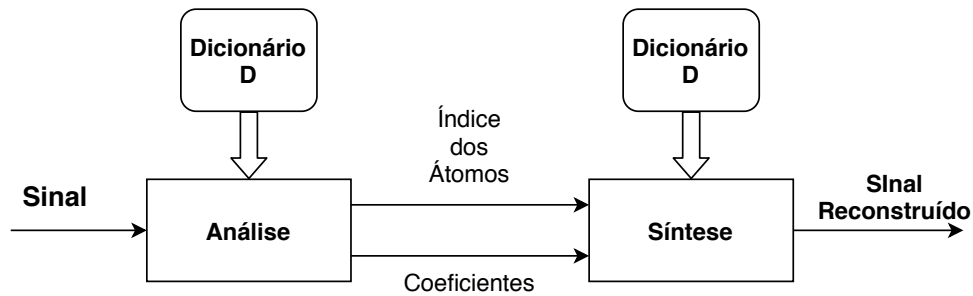


Figura 11: Diagrama de blocos de análise e síntese usando decomposições atômicas.

A Figura 11 ilustra um esquema de análise e síntese usando decomposições atômicas. Enquanto a análise do sinal encontra coeficientes e índices dos átomos com base em um dicionário D , a síntese reconstrói o sinal através da equação (23), considerando o mesmo dicionário D . Existem várias aplicações relacionadas às decomposições atômicas, que têm sido empregadas na filtragem de sinais e supressão de ruído [22], análise de fenômenos físicos intrínsecos dos sinais em conjunto com reconhecimento padrões e modelagem matemática de sinais [55, 56], análise tempo-frequência [22, 56]. A decomposição pode ser empregada também para compressão de sinais [7, 57–59], dado que é capaz de gerar uma representação compacta, característica fundamental para o bom desempenho de um codificador.

A distorção entre a aproximação de M termos na equação (23) e o sinal \mathbf{x} é descrita por

$$d(\mathbf{x}, M, D) = \|\mathbf{x} - \hat{\mathbf{x}}\| = \left\| \mathbf{x} - \sum_{i=1}^M \alpha_i \mathbf{g}_{\gamma(i)} \right\|. \quad (24)$$

Esta distorção depende do:

- número de elementos M usados para representar $\hat{\mathbf{x}}$;
- dos átomos $\mathbf{g}_{\gamma(i)}$ escolhidos para exprimir $\hat{\mathbf{x}}$;

- dos coeficientes α_i dos átomos

A representação mais compacta de \mathbf{x} é aquela que utiliza o menor número de átomos com distorção nula. Contudo, na prática, busca-se a representação mais eficiente, ou seja, aquela que utiliza o menor número de átomos que aproxima \mathbf{x} com a menor distorção aceitável de acordo com a aplicação [9].

Em decomposições atômicas, permite-se o uso de dicionários compostos por elementos linearmente dependentes e eles têm que ser completos para que combinações lineares de M termos formadas com átomos de um dicionário D sejam capazes de representar qualquer sinal $\mathbf{x} \in X$ com distorção $d(\mathbf{x}, M, D)$ arbitrariamente pequena, D tem que ser completo em X . Assim, deverá existir pelo menos uma combinação linear de elementos de D que produza $\hat{\mathbf{x}} = \mathbf{x}$, $\forall \mathbf{x} \in X$, isto é, D tem que gerar o espaço X , o que as distingue de metodologias clássicas baseadas em transformadas, nas quais se utiliza uma base que provê uma representação única do sinal.

A dependência linear do dicionário está associada à existência de mais elementos no dicionário do que o necessário para se gerar um espaço vetorial e, portanto, há mais de uma representação possível do sinal. Nesse caso, o dicionário é denominado redundante. Os dicionários redundantes podem ser formados por uma união de bases ortogonais, por transformadas redundantes (como *wavelet packet* [60] e janelas de Gabor) [61] ou por formas de onda parametrizadas [7, 10].

A redundância resulta em poder representar o sinal de diferentes maneiras utilizando os mesmos elementos do dicionário. Logo, o dicionário redundante corresponde a uma base linearmente dependente. A utilização de dicionários altamente redundantes possibilita a extração direta de uma variedade maior de padrões e fenômenos presentes em sinais, resultando em representações mais compactas e eficientes. Entretanto, a compacidade da representação de sinais não depende somente do nível de redundância, mas também da acurácia do modelo matemático que gera os elementos do dicionário em representar de forma coerente os fenômenos intrínsecos de uma determinada classe de sinais. Os átomos do dicionário devem apresentar um alto grau de similaridade com os padrões existentes da classe. Em processos físicos, considera-se o sinal observado como uma mistura de componentes \mathbf{p}_i , que representam os fenômenos físicos dada por

$$\mathbf{x} = \sum_{i=1} \beta_i \mathbf{p}_i + \mathbf{n}, \quad (25)$$

em que \mathbf{n} é o ruído inerente à observação. Quanto mais parecidos forem os átomos $\mathbf{g}_{\gamma(i)}$ e respectivos coeficientes α_i utilizados na representação de um sinal \mathbf{x} às componentes \mathbf{p}_i e respectivos coeficientes β_i efetivamente presentes em \mathbf{x} , melhor será a representação obtida para modelagem do sinal e reconhecimento de padrões. Portanto, conclui-se que a redundância e a coerência são fatores fundamentais para se atingir um maior nível de compacidade na representação de sinais.

De maneira geral, nota-se que através das decomposições atômicas é possível modelar sinais de forma matematicamente acurada, compacta e coerente, com baixa distorção. Um algoritmo bastante popular utilizado na obtenção dessas decomposições é o *Matching Pursuit* (MP) [7, 9, 22].

2.2 Matching Pursuit

O algoritmo de *Matching Pursuit* (MP) realiza aproximações sucessivas de sinais, de forma adaptativa, buscando em um dicionário redundante D a melhor aproximação a cada iteração. Esse algoritmo é muito utilizado para representação de diversos tipos de sinais (sinais elétricos, áudio, biomédicos), separação de fontes, reconhecimento de padrões, dentre outros [2, 62]. De maneira geral, o algoritmo pode ser usado para representar inúmeros tipos de sinais e os que diferencia é o dicionário que é utilizado para modelar os fenômenos neles presentes [7]. Além disso, este método também é considerado um algoritmo voraz, no qual pode vir a consumir grande parte de suas iterações corrigindo erros cometidos nas primeiras iterações [21, 22].

Considere um sinal \mathbf{x} pertencente ao espaço de Hilbert \mathbb{H} de dimensão finita

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}. \quad (26)$$

Define-se um dicionário redundante $D = (\mathbf{g}_{\gamma})_{\gamma \in \Gamma}$, tal que $\|\mathbf{g}_{\gamma}\| = 1$, em que γ é o índice que define \mathbf{g}_{γ} e Γ é o conjunto de todos os γ 's possíveis. O MP gera uma expansão linear de \mathbf{x} em função de um subconjunto de elementos de D . Isso é feito através de aproximações sucessivas de \mathbf{x} com projeções ortogonais sobre os elementos do dicionário. O sinal \mathbf{x} pode ser decomposto da seguinte maneira

$$\mathbf{x} = \langle \mathbf{x}, \mathbf{g}_{\gamma_0} \rangle \mathbf{g}_{\gamma_0} + \mathbf{r}_{\mathbf{x}} \quad (27)$$

onde \mathbf{r}_x é o resíduo da projeção de \mathbf{x} na direção de \mathbf{g}_{γ_0} . Como \mathbf{g}_{γ_0} é ortogonal a \mathbf{r}_x temos

$$\|\mathbf{x}\|^2 = |\langle \mathbf{x}, \mathbf{g}_{\gamma_0} \rangle|^2 + \|\mathbf{r}_x\|^2 \quad (28)$$

Com o objetivo de minimizar $\|\mathbf{r}_x\|$, a cada passo da decomposição escolhe-se o elemento \mathbf{g}_{γ_0} do dicionário tal que $\langle \mathbf{x}, \mathbf{g}_{\gamma_0} \rangle$ seja máximo. Devido a Γ ser finito, obtém-se um \mathbf{g}_{γ_0} ótimo no seguinte sentido

$$|\langle \mathbf{x}, \mathbf{g}_{\gamma_0} \rangle| \geq \max_{\gamma \in \Gamma, \gamma \neq \gamma_0} |\langle \mathbf{x}, \mathbf{g}_{\gamma} \rangle| \quad (29)$$

Em seguida, o resíduo \mathbf{r}_x é decomposto pela sua projeção no elemento do dicionário que o melhor representa, como foi feito anteriormente. Essa projeção irá gerar um segundo resíduo \mathbf{r}_x^2 , o qual sofrerá o mesmo procedimento, gerando um terceiro resíduo e assim por diante. Dessa forma, obtém-se a seguinte formulação para se calcular o resíduo de ordem n

$$\mathbf{r}_x^n = \langle \mathbf{r}_x^n, \mathbf{g}_{\gamma_n} \rangle \mathbf{g}_{\gamma_n} + \mathbf{r}_x^{n+1} \quad (30)$$

em que $\mathbf{r}_x^0 = \mathbf{x}$, e define-se o resíduo de ordem $n + 1$, \mathbf{r}_x^{n+1} . Ainda que seja não linear, o MP é caracterizado pela conservação de energia, característica intrínseca de decomposições lineares e ortogonais. Para fins de demonstração, dado que \mathbf{r}_x^{n+1} é ortogonal a \mathbf{g}_{γ_n} , tem-se

$$\|\mathbf{r}_x^n\|^2 = |\langle \mathbf{r}_x^n, \mathbf{g}_{\gamma_n} \rangle|^2 + \|\mathbf{r}_x^{n+1}\|^2 \quad (31)$$

Em seguida, escreve-se $\|\mathbf{x}\|^2$ da seguinte forma

$$\|\mathbf{x}\|^2 = \sum_{n=0}^{m-1} \left(\|\mathbf{r}_x^n\|^2 - \|\mathbf{r}_x^{n+1}\|^2 \right) + \|\mathbf{r}_x^m\|^2 \quad (32)$$

Combinando as equações (30) e (32), obtém-se a equação de conservação de energia

$$\|\mathbf{x}\|^2 = \sum_{n=0}^{m-1} |\langle \mathbf{r}_x^n, \mathbf{g}_{\gamma_n} \rangle|^2 + \|\mathbf{r}_x^m\|^2 \quad (33)$$

Decompondo \mathbf{x} até o resíduo de ordem M , obtém-se o seguinte somatório

$$\mathbf{x} = \sum_{n=0}^{m-1} (\mathbf{r}_x^n - \mathbf{r}_x^{n+1}) + \mathbf{r}_x^m \quad (34)$$

Com base na representação da equação (30), tem-se

$$\mathbf{x} = \sum_{n=0}^{m-1} \langle \mathbf{r}_x^n, \mathbf{g}_{\gamma_n} \rangle + \mathbf{r}_x^m \quad (35)$$

Em [21], apresenta-se um teorema que garante a convergência deste método de decomposição, se o dicionário for completo, $\lim_{n \rightarrow \infty} \|\mathbf{r}_x^n\| = 0$. Na prática, esse processo iterativo termina quando o resíduo $\|\mathbf{r}_x^m\|$ for suficientemente pequeno. Ao fim da decomposição, obtém-se uma representação para \mathbf{x} como uma soma dos elementos do dicionário que melhor representam os resíduos; logo, tem-se que

$$\mathbf{x} \approx \sum_{n=0}^{M-1} \alpha_n \mathbf{g}_{\gamma_n} \quad (36)$$

em que M corresponde ao número de passos do MP e também ao número de átomos do dicionário D cuja a combinação linear aproxima \mathbf{x} , e $\alpha_n = \langle \mathbf{r}_x^n, \mathbf{g}_{\gamma_n} \rangle$. Portanto, \mathbf{x} pode ser aproximado pela sequência $(\alpha_n, \gamma_n) \in N$, que pode ser denominada de livro de estruturas [7].

2.2.1 Redução de Custo Computacional

Tradicionalmente, o Matching Pursuit (MP) é considerado um algoritmo de alto custo computacional para aplicação em problemas reais envolvendo sinais de grande dimensão. A implementação numérica típica deste algoritmo apresenta uma complexidade de $\mathcal{O}(N^2)$ (N é o comprimento do sinal), que leva a tempos de execução muito elevados [63]. Esta seção tem o propósito de discutir e apresentar estratégias para a redução da complexidade computacional do MP, em especial, na etapa de cálculo dos produtos internos.

Em [30], é apresentada uma proposição para o cálculo do produto interno entre o sinal e um átomo real do dicionário apresentando fase ótima. Os elementos de um dicionário redundante podem ser gerados, escalonando-se, trasladando-se e modulando-se uma função-janela $g(t)$ contínua no tempo. Aqui, em vez da função contínua $g(t)$, considera-se sua versão discreta $g(n)$, que corresponde à amplitude de $g(t)$ a cada instante nT , onde T representa o período de amostragem. No caso de um dicionário de átomos

reais e discretos, cada átomo $g_{\gamma,\phi}(n)$ é definido como

$$g_{\gamma,\phi}(n) = \frac{g\left(\frac{n-\tau}{s}\right) \cos(\xi n + \phi)}{\|g\left(\frac{n-\tau}{s}\right) \cos(\xi n + \phi)\|}, \quad (37)$$

em que $\gamma = (s, \tau, \xi)$, s é a escala. τ é o deslocamento, ξ é a frequência e θ representa a fase. Em seguida, definem-se os vetores

$$P_{\gamma}(n) = g\left(\frac{n-\tau}{s}\right) \cos(\xi n), \quad (38)$$

$$Q_{\gamma}(n) = g\left(\frac{n-\tau}{s}\right) \text{sen}(\xi n). \quad (39)$$

Então, tem-se

$$\mathbf{g}_{\gamma,\phi} = \frac{\mathbf{P}_{\gamma} \cos \phi - \mathbf{Q}_{\gamma} \text{sen} \phi}{\|\mathbf{P}_{\gamma} \cos \phi - \mathbf{Q}_{\gamma} \text{sen} \phi\|}, \quad (40)$$

com isso, o produto interno pode ser escrito como

$$\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{P}_{\gamma} \rangle \cos \phi - \langle \mathbf{x}, \mathbf{Q}_{\gamma} \rangle \text{sen} \phi}{\|\mathbf{P}_{\gamma} \cos \phi - \mathbf{Q}_{\gamma} \text{sen} \phi\|}. \quad (41)$$

Sejam \mathbf{x} um vetor de sinal, $\gamma = (s, \tau, \xi)$, $\mathbf{P} = \mathbf{P}_{\gamma}$, $\mathbf{Q} = \mathbf{Q}_{\gamma}$, tem-se que

$$a_1 = \langle \mathbf{x}, \mathbf{P} \rangle \|\mathbf{Q}\|^2 - \langle \mathbf{x}, \mathbf{Q} \rangle \langle \mathbf{P}, \mathbf{Q} \rangle, \quad (42)$$

$$b_1 = \langle \mathbf{x}, \mathbf{Q} \rangle \|\mathbf{P}\|^2 - \langle \mathbf{x}, \mathbf{P} \rangle \langle \mathbf{P}, \mathbf{Q} \rangle. \quad (43)$$

Logo, a fase ótima ϕ_0 , para a qual $\max_{\phi \in [0, 2\pi)} \langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle$, é dada por uma das soluções abaixo

1. Se $\xi = 0$

$$\phi_0 = 0 \quad e \quad \langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{P} \rangle}{\|\mathbf{P}\|} \quad (44)$$

2. Se $\xi \neq 0$ e $a_1 = 0$, então

$$\phi_0 = \frac{\pi}{2} \quad e \quad \langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{Q} \rangle}{\|\mathbf{Q}\|} \quad (45)$$

3. Se $\xi \neq 0$ e $a_1 \neq 0$, então

$$\phi_0 = \arctan\left(-\frac{b_1}{a_1}\right) \quad e \quad \langle \mathbf{x}, \mathbf{g}_{\gamma\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{P} \rangle a_1 + \langle \mathbf{x}, \mathbf{Q} \rangle b_1}{\|\mathbf{P}a_1 + \mathbf{Q}b_1\|} \quad (46)$$

Podemos chegar a conclusão que são necessários os seguintes fatores para obter a projeção $\langle \mathbf{x}, \mathbf{g}_{\gamma\phi} \rangle$ do vetor sinal sobre o átomo de fase ótima

$$\langle \mathbf{x}, \mathbf{P} \rangle = \sum_{n=0}^{N-1} x(n)g\left(\frac{n-\tau}{s}\right)\cos(\xi n), \quad (47)$$

$$\langle \mathbf{x}, \mathbf{Q} \rangle = \sum_{n=0}^{N-1} x(n)g\left(\frac{n-\tau}{s}\right)\text{sen}(\xi n), \quad (48)$$

$$\langle \mathbf{P}, \mathbf{P} \rangle = \sum_{n=0}^{N-1} g^2\left(\frac{n-\tau}{s}\right)\cos^2(\xi n), \quad (49)$$

$$\langle \mathbf{Q}, \mathbf{Q} \rangle = \sum_{n=0}^{N-1} x(n)g^2\left(\frac{n-\tau}{s}\right)\text{sen}^2(\xi n), \quad (50)$$

$$\langle \mathbf{Q}, \mathbf{P} \rangle = \sum_{n=0}^{N-1} g^2\left(\frac{n-\tau}{s}\right)\cos(\xi n)\text{sen}(\xi n). \quad (51)$$

Com base no algoritmo *Matching Pursuit* e na estratégia usada em [30], propõe-se uma versão rápida do MP tendo a frequência ξ como parâmetro de livre escolha. Com o objetivo de ter máximo aproveitamento do potencial computacional da transformada Rápida de Fourier (FFT), os sinais devem ter tamanhos em potências de 2, como por exemplo $s = 2^j$ onde $j = 1, 2, \dots, \log_2 N$. Inicialmente para s e ξ , definem-se

$$P_\tau(n) = g\left(\frac{n-\tau}{s}\right)\cos(\xi n), \quad (52)$$

$$Q_\tau(n) = g\left(\frac{n-\tau}{s}\right)\text{sen}(\xi n), \quad (53)$$

Em seguida, calculam-se as projeções do sinal sobre versões deslocadas no tempo de um átomo de forma mais rápida fazendo uso de uma propriedade da DFT referente a convolução circular no tempo [64]. Essa propriedade enuncia que o produto escalar entre as transformadas para o domínio da frequência de dois vetores equivale à convolução

circular entre eles no domínio do tempo, ou seja,

$$\sum_{l=0}^{N-1} z(l)w(n-l) = \sum_{l=0}^{N-1} z(n-l)w(l) \Leftrightarrow Z(k)W(k), \quad (54)$$

de comprimentos N_z e N_w , utilizando a DFT inversa do produto de duas DFTs. Entretanto, deseja-se calcular a convolução linear entre ambas as sequências usando essa propriedade. Para isso, o comprimento deve ser $N_z + N_w - 1$ realizando-se o *zero padding*, que consiste em preencher com zero as amostras adicionais. Sejam $z(n) = x(n)e^{i\xi n}$ e $w(n) = g\left(\frac{N-1-n}{s}\right)$ duas sequências de comprimento N e $z_1(n)$ e $w_1(n)$ suas respectivas sequências de comprimento aumentado $2N - 1$ após o *zero padding*. A convolução linear entre $z(n)$ e $w(n)$ é calculada da seguinte maneira

$$[z * w](n) = DFT^{-1}\{Z_1(k).W_1(k)\}. \quad (55)$$

Observa-se que $w(n)$ corresponde a reversão temporal de $g(n/s)$, pois, dessa forma, através da convolução linear obtém-se a correlação temporal entre as sequências $x(n)e^{i\xi n}$ e $g(n/s)$, que é o desejado. Tendo em mente as equações (47) e (48) e a sequência $[z * w](n)$ obtém-se

$$\langle \mathbf{x}, \mathbf{P}_\tau \rangle = Re\{[z * w](\tau + N - 1)\}, \quad (56)$$

$$\langle \mathbf{x}, \mathbf{Q}_\tau \rangle = Im\{[z * w](\tau + N - 1)\}, \quad (57)$$

Agora, sejam $z(n) = e^{i2\xi n}$ e $w(n) = g^2\left(\frac{N-1-n}{s}\right)$, e $z_1(n)$ e $w_1(n)$ suas respectivas sequências de comprimento aumentado $2N - 1$ com preenchimento de zeros. Obtém-se a convolução linear $[z * w](n)$ como descrito na equação (55). Com base nesse resultado e nas equações (49), (50), (51), obtém-se

$$\langle \mathbf{P}_\tau, \mathbf{P}_\tau \rangle = \frac{1}{2}\{Re\{[z_{\xi=0} * w](\tau + N - 1)\} + Re\{[z * w](\tau + N - 1)\}\}, \quad (58)$$

$$\langle \mathbf{Q}_\tau, \mathbf{Q}_\tau \rangle = \frac{1}{2}\{Re\{[z_{\xi=0} * w](\tau + N - 1)\} - Re\{[z * w](\tau + N - 1)\}\}, \quad (59)$$

$$\langle \mathbf{Q}_\tau, \mathbf{P}_\tau \rangle = \frac{1}{2} \text{Im}\{[z * w](\tau + N - 1)\}, \quad (60)$$

em que $z_{\xi=0}(n) = 1$

O pseudocódigo relativo ao MP rápido apresentado usando a técnica sem restrição de espaçamento frequencial pode ser visto com mais detalhes em [7]. Vale destacar que a complexidade do algoritmo é dada como $\mathcal{O}(12N \log_2 N)$. A grande vantagem deste método é a obtenção dos produtos internos entre o resíduo e N átomos através de cálculos usando FFTs associados aos deslocamentos temporais e não exige espaçamento de frequência obrigatoriamente linear.

2.3 Dicionários Redundantes Parametrizados

Conceitualmente, o dicionário é formado por um conjunto finito de elementos $D = \{g_i\}$, $i \in I$. Neste caso, $g_i \in D$ pode ser indexado através de $i \in I = \{1, \dots, \#D\}$, onde $\#D$ é o número de elementos em D , a sua cardinalidade. Um dicionário completo permite representar qualquer sinal com um erro de aproximação arbitrariamente pequeno. Entretanto, o emprego de um dicionário completo não garante a obtenção de uma representação compacta, tampouco coerente. A probabilidade de encontrar um átomo no dicionário possuidor de grande semelhança com o sinal a ser decomposto aumenta com $\#D$ [6]. Logo, a utilização de dicionários com $\#D$ grande é desejável em algumas aplicações, possibilitando que D contenha átomos semelhantes a todas as componentes potenciais dos sinais a serem decompostos.

Os átomos de um dicionário redundante são gerados escalonando-se, trasladando-se e modulando-se uma função $h(t)$. Supondo $h(t)$ real e centralizada em 0, o átomo complexo $g_\gamma(t)$ é definido por

$$g_\gamma(t) = \frac{1}{\sqrt{s}} h\left(\frac{t-u}{s}\right) e^{i\xi t} \quad (61)$$

em que o fator $\frac{1}{\sqrt{s}}$ torna unitária a energia de g_γ . O índice γ é composto pelos parâmetros de escalonamento s , frequência de modulação ξ e deslocamento no tempo u , ou seja, $\gamma \in \Gamma = \mathbb{R}^+ \times \mathbb{R}^2$ e Γ é conjunto de todos os parâmetros do dicionário. A função $g_\gamma(t)$ possui sua energia concentrada em torno do seu centro u . A transformada de Fourier de

$g_\gamma(t)$ é dada por

$$\hat{g}_\gamma(\omega) = \sqrt{s} \hat{h}(s(\omega - \xi)) e^{i(\omega - \xi)u} \quad (62)$$

dado que $|\hat{h}(\omega)|$ é par, pois $h(t)$ é real, $|\hat{g}_\gamma(\omega)|$ possui sua energia concentrada em torno da frequência central $\omega = \xi$. No caso de sinais reais, utiliza-se um dicionário de átomos reais gerados da seguinte forma

$$\mathbf{g}_{(\gamma, \phi)}(t) = \frac{K_{(\gamma, \phi)}}{\sqrt{s}} h\left(\frac{t - u}{s}\right) \cos(\xi t + \phi) \quad (63)$$

em que $K_{(\gamma, \phi)}$ é um fator de normalização para que $\|\mathbf{g}_{(\gamma, \phi)}\| = 1$, e a fase ϕ pertence ao intervalo $[0, 2\pi)$.

A escolha de funções base para compor o dicionário depende do comportamento e características da classe de sinais alvo. Para isso, deve-se utilizar um dicionário formado por elementos que sejam coerentes com os fenômenos ou padrões existentes no sinal a ser decomposto. No caso do sinal de atividade eletrodérmica, propomos dicionários específicos que levam em conta a estrutura característica do sinal no tempo. Os dicionários incluem átomos fásicos, ou seja, as rápidas flutuações sobrepostas ao sinal, também conhecido como SCRs. Analisando-se a forma de onda de um SCR, foi possível observar suas características gerais de um aumento acentuado de SC e uma lenta recuperação [31]. Houve diferentes tentativas de descrever quantitativa ou qualitativamente o curso do SCR. Abordagens quantitativas procuram funções matemáticas que descrevem o curso do SCR ao longo do tempo. Usamos quatro tipos de funções que podem potencialmente representar a elevação íngreme característica e decaimento lento de SCRs: a função sigmoide-exponencial, [1, 13] a função de *Bateman* [1, 15], bem como uma função Chi-quadrado [1], e função de Gabor [25] sem modulação para decompor o sinal de EDA em um pequeno número de átomos fásicos via *Matching Pursuit*.

2.4 Dicionário de Bateman

Como dito anteriormente, neste projeto, utilizamos o dicionário de *Bateman* para a decomposição da componente fásica dos sinais de EDA. Inspirada nas leis de difusão nos poros da pele, a função de *Bateman* representa a forma de uma resposta ao impulso, que quando convoluída com uma função de driver resulta no sinal EDA. No modelo de [14], o SCR é descrito por uma função biexponencial, que normalmente utilizada para modelar

efeitos de impulsos nervosos individuais sobre a ativação sináptica da membrana neuronal. Um átomo de *Bateman* é composto por uma função bi-exponencial definida da seguinte forma

$$h(t) = (e^{-a(t)} - e^{-b(t)}) \quad (64)$$

onde a e b são as constantes de decaimento e crescimento respectivamente. Aplicando (64) em (61), gera-se o dicionário de *Bateman* definido por:

$$\mathbf{g}_\gamma(t) = \frac{K_\gamma}{\sqrt{s}} ((e^{-a(\frac{t-u}{s})} - e^{-b(\frac{t-u}{s})})) \cos(\xi t + \phi) \quad (65)$$

Considerando que não há modulação de frequência para a função de *Bateman*, ou seja $\xi = 0$ e que o decaimento e crescimento são definidos usando $\rho = a\frac{1}{s}$ e $\eta = b\frac{1}{s}$, tem-se que

$$\mathbf{g}_\gamma(n) = K_\gamma (e^{-\rho(n-\tau)} - e^{-\eta(n-\tau)}) \cos(\phi) [u(n - n^s) - u(n - n^e)] \quad (66)$$

onde γ é o conjunto de parâmetros do átomo, no caso $\gamma = (\rho, \eta, \tau, \phi)$, sendo ρ é o decaimento, η é o crescimento, τ é o deslocamento, ϕ é a fase, K_γ é o fator de normalização, $u(n)$ é função degrau unitária e n^e e n^s definem o suporte do átomo. Na Figura 12, é possível observar um exemplo de forma de onda de um átomo de *Bateman*.

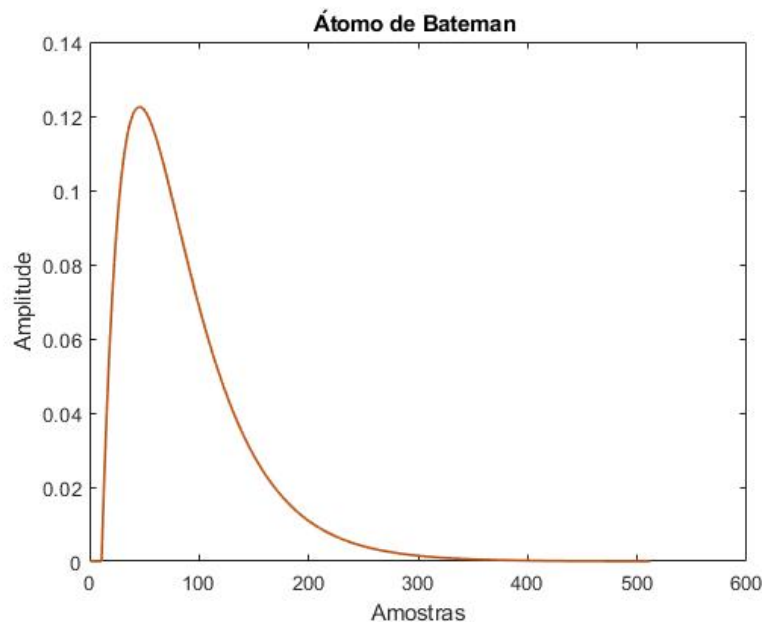


Figura 12: Átomo de *Bateman*.

Os parâmetros ρ e η controlam a inclinação da recuperação e início, respectiva-

mente; quanto mais altos eles são, menos tempo leva para o SCR para fazer a transição para o pico. Outros estudos também mostram que esse dicionário é o mais eficiente em termos de esparsidade para representar o sinal comparativamente a outros dicionários, como de Gabor e Triangular [25]. Mostra-se que os átomos de *Bateman* representam a maior parte das energias da decomposição nas primeiras iterações, destacando assim a sua alta capacidade de representar o sinal de EDA compactamente.

2.5 Dicionário de Sigmoide-Exponencial

Assim como a função de Bateman a Sigmoide-Exponencial ilustrada pela Figura 13 possui forma de onda semelhante a uma onda SCR e é composta por duas funções: sigmoide e exponencial. A parte sigmoide serve como aproximação da ação cumulativa do enchimento das glândulas sudoríparas e é representada pela subida do SCR, enquanto a função de exponencial descendente é pensada para retratar a recuperação [13, 34]. Para cada parâmetro, é fornecida alguma fundamentação que a conecta a processos fisiológicos. Com isso podemos definir que a função sigmoide-exponencial pode ser dada como

$$h(t) = \frac{e^{-a(t)}}{(1 + (b(t))^{-2})^2}, \quad (67)$$

onde a e b , assim como a na função de *Bateman* são as constantes de decaimento e crescimento respectivamente. Aplicando (67) em (61), gera-se o dicionário de Sigmoide-Exponencial definido por

$$g_\gamma(t) = K_\gamma \frac{e^{-a \frac{(t-u)}{s}}}{\left(1 + \left(b \left(\frac{(t-u)}{s}\right)\right)^{-2}\right)^2} \cos(\xi t + \phi). \quad (68)$$

Para o dicionário de Sigmoide-Exponencial iremos considerar modulação de frequência igual a zero $\xi = 0$ e que o decaimento e crescimento são definidos usando $\rho = a \frac{1}{s}$ e $\eta = b \frac{1}{s}$ e K_γ é a constante de normalização. Por fim, podemos definir o dicionário de sigmoide-exponencial no tempo discreto como

$$g_\gamma(n) = K_\gamma \frac{e^{-\rho(n-u)}}{(1 + (\eta(n-u))^{-2})^2} [u(n - n^s) - u(n - n^e)] \quad (69)$$

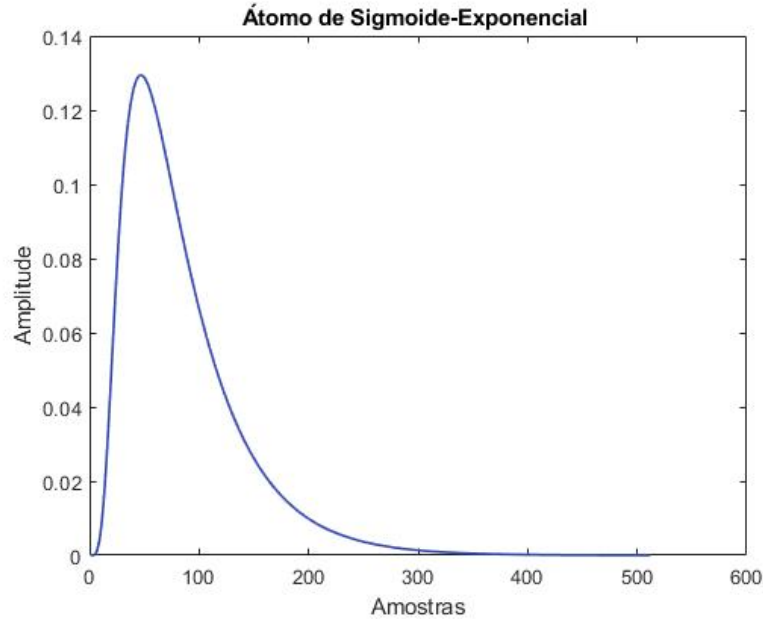


Figura 13: Átomo de Sigmoide-Exponencial.

2.6 Dicionário de Chi-Quadrado

A função de densidade de probabilidade da distribuição do Chi-quadrado, aqui referida como função Chi-quadrado, é introduzida pelo [1] como uma maneira alternativa de representar a forma SCR. Envolve apenas um parâmetro χ , que é definido como grau de liberdade de uma PDF, com grandes valores resultando em SCRs mais largos. Embora nenhuma explicação fisiológica aparente pudesse ser encontrada para o Chi-quadrado, cada átomo de dicionário é codificado com apenas um parâmetro. Tendo em conta o grande tamanho de dicionários em técnicas de representação esparsa, esta função se beneficia de custo computacional e alocação de memória, um fator importante para aplicações de dispositivos móveis. A função Chi-quadrado pode ser definida como

$$h(t, \chi) = \frac{t^{\frac{\chi}{2}-1} e^{-\frac{t}{2}}}{2^{\frac{\chi}{2}} \Gamma\left(\frac{\chi}{2}\right)} \quad (70)$$

onde Γ é a generalização da função fatorial para números complexos positivos. Para valores inteiros a função Γ pode ser definida como

$$\Gamma(n) = (n - 1)! \quad (71)$$

Contudo, para números complexos a parte real sendo um número real positivo, a função $\Gamma(n)$ é definida via integral

$$\Gamma(n) = \int_0^{\infty} t^{n-1} e^{-t} dt. \quad (72)$$

Sendo assim, aplicando (70) em (61) temos

$$g(t) = \frac{K_{\gamma}}{\sqrt{s}} \frac{\left(\frac{t-u}{s}\right)^{\frac{\chi}{2}-1} e^{-\left(\frac{t-u}{2s}\right)}}{2^{\frac{\chi}{2}} \Gamma\left(\frac{\chi}{2}\right)} \cos(\xi t + \phi) \quad (73)$$

Para o dicionário de Chi-quadrado, iremos considerar modulação de frequência igual a zero, ou seja, $\xi = 0$. Por fim, podemos definir o dicionário de Chi-quadrado no tempo discreto como

$$g_{\gamma}(n) = K_{\gamma} \frac{(\rho(n-u))^{\frac{\chi}{2}-1} e^{-\left(\frac{\rho(n-u)}{2}\right)}}{2^{\frac{\chi}{2}} \Gamma\left(\frac{\chi}{2}\right)} [u(n-n^s) - u(n-n^e)] \quad (74)$$

A representação de um átomo deste dicionário é ilustrado pela Figura 14.

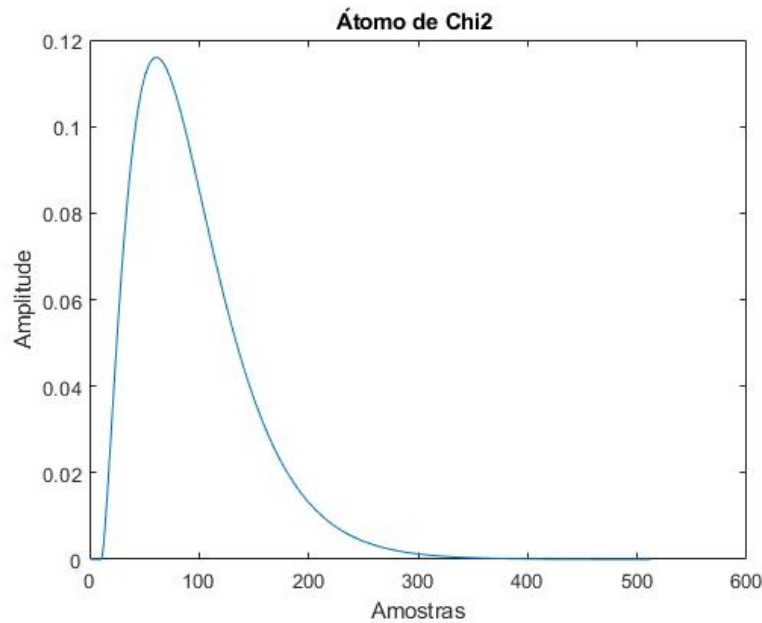


Figura 14: Átomo de Chi-quadrado.

2.7 Dicionário de Gabor

O uso do dicionário de Gabor para decomposições de sinais via *Matching Pursuit* está presente em diversos trabalhos anteriores [6, 7, 9, 21, 22] e se mostrou ser eficiente

para diversas aplicações como análise de sinais elétricos e de áudio [6, 7, 25]. Uma das principais razões do uso desse dicionário é sua boa representação em tempo-frequência por meio de gaussianas moduladas. Contudo, nesta aplicação de sinais eletrodérmicos, estamos interessados apenas na envoltória da gaussiana com $\xi = 0$. A Figura 15 ilustra o átomo de Gabor e diferentemente dos outros dicionários vistos anteriormente, a gaussiana é uma função simétrica, ou seja, o tempo de subida da função é igual ao de subida. Sendo assim, podemos começar a definir o átomo de Gabor sendo

$$h(t) = 2^{\frac{1}{4}} e^{-\pi(t)^2} \quad (75)$$

Através do escalonamento, deslocamento e modulação da Gaussiana apresentada na equação (75) podemos obter o dicionário de Gabor definido como

$$g_\gamma(n) = K_\gamma 2^{\frac{1}{4}} e^{-\pi\left(\frac{n-\tau}{s}\right)^2} \cos(\xi n + \phi) [u(n - n^s) - u(n - n^e)] \quad (76)$$

onde $n \in [0, \dots, N - 1]$, $\gamma = (s, \tau, \xi, \phi, n^s, n^e)$, onde s é a escala, τ é o deslocamento, ξ é a frequência de modulação, ϕ é a fase do átomo, K_γ é fator de normalização, n^s e n^e definem o suporte do átomo, e $u(\cdot)$ é o degrau unitário.

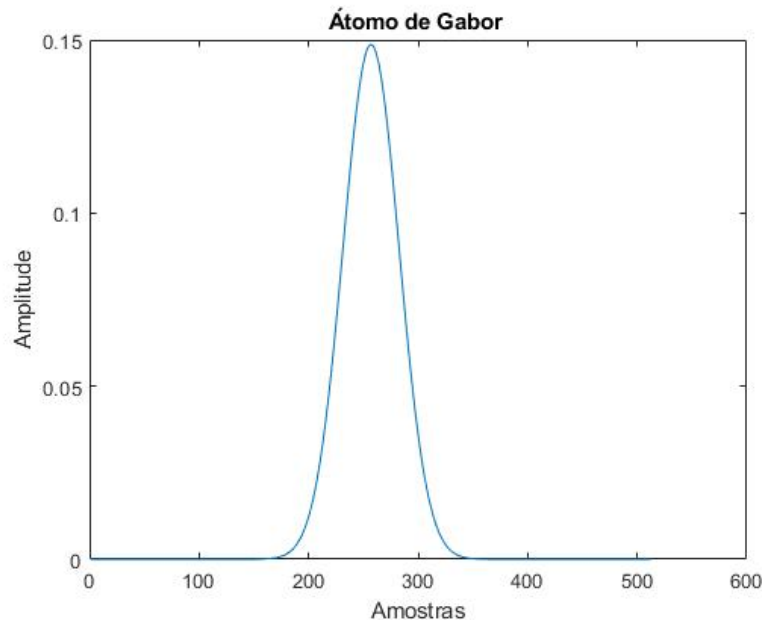


Figura 15: Átomo de Gabor.

2.8 Discretização de Parâmetros

A discretização de parâmetros é um dos grandes desafios na definição de um dicionário fixo e finito. Se a discretização for muito fina, maior será a capacidade de o dicionário representar mais eficientemente os padrões intrínsecos do sinal. Em contrapartida, há um aumento explosivo da complexidade do algoritmo de decomposição em virtude do aumento da cardinalidade do dicionário. O projeto do dicionário deve levar em consideração as características intrínsecas de um sinal de EDA, como seu comportamento variante no tempo e conteúdo espectral para saber qual faixa de valores escolher na discretização do dicionário [25]. Dessa forma, realizamos um estudo a respeito de algumas possíveis escolhas dos intervalos de valores a serem atribuídos aos parâmetros. A cardinalidade $\#D$ do dicionário afeta os seguintes aspectos:

- **Complexidade computacional:** em geral, o custo computacional dos algoritmos que obtêm representações atômicas está relacionado a $\#D$, de forma tal que quanto maior for $\#D$, maior será, em geral, o custo computacional dos algoritmos.
- **Taxa de codificação:** a taxa (bits/símbolo) necessária para codificar cada um dos átomos utilizados nas representações cresce com $\#D$.

A probabilidade de encontrar um átomo no dicionário possuidor de grande semelhança com o sinal a ser decomposto aumenta com $\#D$ [9]. Logo, a utilização de dicionários com $\#D$ grande é desejável em algumas aplicações, possibilitando que D contenha átomos semelhantes a todas as componentes potenciais dos sinais a serem decompostos. Em contrapartida, quanto maior o dicionário, maior é o tempo de execução do algoritmo de decomposição. Portanto, o ideal é encontrar átomos que sejam mais relevantes para a representação do sinal. Para isso, deve-se escolher bem os parâmetros que geram as funções de cada dicionário. Deve-se considerar que o algoritmo *Matching Pursuit* por ser voraz, acaba corrigindo os erros de aproximação produzidos ao longo das iterações o que induz o emprego de átomos com características diferentes de um SCR, ou com características de impulso. No capítulo 3, será visto que conforme o resíduo passa a ter comportamento de ruído, menos o modelo classificador é capaz de generalizar. Contudo, átomos com essas características são importantes para a convergência da decomposição atômica.

2.8.1 Figuras de mérito para avaliação da decomposição

Para avaliar o desempenho da decomposição, é utilizada uma figura de mérito que mede a razão da potência de um sinal e a potência do ruído sobreposto ao sinal. A razão sinal-ruído compara o nível de um sinal desejado com o nível do ruído de fundo. Tal figura de mérito é chamada de SNR (*Signal-to-Noise Ratio*) definida como

$$SNR = \frac{P_{\mathbf{x}}}{P_{\mathbf{r}}} \quad (77)$$

e que normalmente é medida em (dB) por apresentar uma faixa dinâmica ampla, ou seja,

$$SNR_{dB} = 10 \log \left(\frac{P_{\mathbf{x}}}{P_{\mathbf{r}}} \right) \quad (78)$$

onde a potência do ruído $P_{\mathbf{r}}$ pode ser definida neste caso como

$$P_{\mathbf{r}} = P_{\mathbf{x}} - P_{\hat{\mathbf{x}}} \quad (79)$$

sendo $P_{\mathbf{x}}$ a potência do sinal original e $P_{\hat{\mathbf{x}}}$ a potência do sinal reconstruído. Quanto mais alta for a razão sinal-ruído, menor é o erro de reconstrução.

2.8.2 Taxa de aproximação

Um critério de parada comum para o algoritmo MP se baseia na taxa de aproximação [65, 66] que indica quanto do sinal está sendo aproximado a cada iteração do *Matching Pursuit*. Altas taxas de aproximação estão associadas ao fato de que os átomos escolhidos para a aproximação são coerentes, ou seja, altamente correlacionados com as características intrínsecas do sinal. A taxa de aproximação é definida como

$$\lambda(\mathbf{r}_{\mathbf{x}}^i) = \frac{|\langle \mathbf{r}_{\mathbf{x}}^{i-1}, \mathbf{g}_{\gamma(i)} \rangle|}{\|\mathbf{r}_{\mathbf{x}}^{i-1}\|}. \quad (80)$$

Neste trabalho, $\lambda(\mathbf{r}_{\mathbf{x}}^{i-1})$ é usado como critério de parada na etapa de decomposição. Pois o objetivo é obter átomos mais coerentes possíveis para serem usados no treinamento do classificador que será apresentado com mais detalhes no capítulo 3.

2.9 Metodologia e Resultados da Discretização de Parâmetros

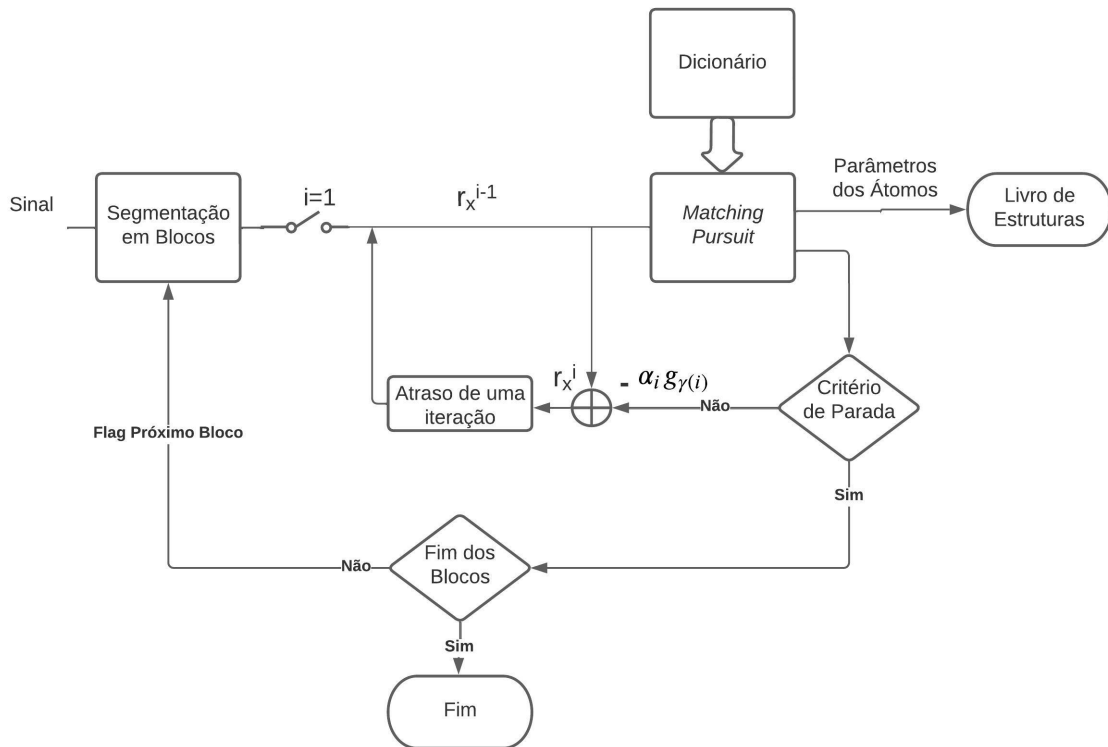


Figura 16: Diagrama de blocos do modelo de decomposição usando *Matching Pursuit*.

Inicialmente, será analisada a decomposição de 5 sinais de EDA com $F_s = 8$ Hz, divididos em blocos de 512 amostras, resultando em blocos de 64 segundos. O tamanho do bloco foi escolhido considerando-se a duração média dos SCRs, garantindo que ele esteja presente dentro das limitações do bloco e também foi considerado a execução da decomposição em tempo computacional viável, visto que o tamanho do bloco influencia diretamente na dimensão do dicionário. Como critério de parada, inicialmente foi considerado um número fixo de 128 átomos por bloco. O modelo do processo de decomposição está ilustrado pelo diagrama de blocos da Figura 16. A chave só é acionada na primeira iteração, ou seja, o algoritmo é alimentado com o trecho do sinal original apenas no primeiro momento. Em iterações seguintes, o algoritmo MP é realimentado com o resíduo $\mathbf{r}_x^i = \mathbf{r}_x^{i-1} - \alpha_i \mathbf{g}_{\gamma(i)}$, onde $\alpha_i \mathbf{g}_{\gamma(i)}$ é o átomo selecionado pelo MP ponderado pelo produto interno. Como resultado da decomposição, temos o nosso livro de estruturas, que seria um documento com a relação de todos átomos escolhidos durante a decomposição

e seus respectivos parâmetros. A partir desses resultados preliminares da decomposição, analisaram-se as ocorrências dos parâmetros referentes aos átomos.

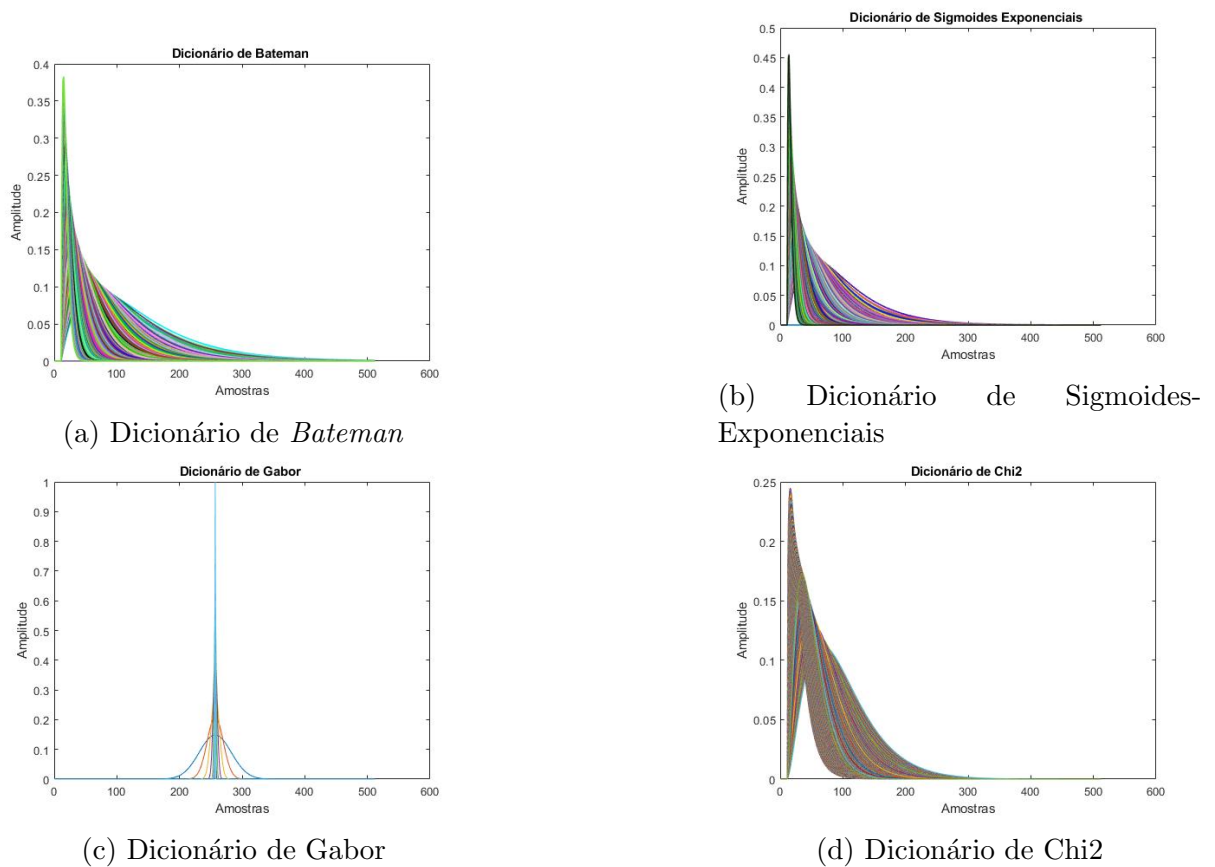


Figura 17: Dicionários parametrizados e adaptado de [1].

Neste primeiro momento, uma das grandes dificuldades deste projeto é definir quais parâmetros de cada sub-dicionário devemos usar com o objetivo de produzir átomos mais bem correlacionados com os fenômenos da atividade eletrodérmica. Devido a isso, devemos relacionar algumas características gerais de um sinal de EDA com as funções que serão usadas para representá-las, além de trabalhar com um dicionário base para análise e estudo para próximas possíveis otimizações/parametrizações e redução da complexidade do dicionário. Sendo assim, considerando que uma SCR dura de 10 a 30 segundos [43] [46] e a fim de se representar a grande variedade de SCRs, os parâmetros dos dicionários foram discretizados inicialmente baseados nos valores usados em [1] para três sub-dicionários (*Bateman*, Sigmoide-Exponencial, e Chi-Quadrado). Para o dicionário de Gabor, foi adotada uma discretização diádica, que resulta em uma base sobrecompleta, permitindo assim, modelar quaisquer fenômenos naturais no domínio tempo-frequência e também sinais com características de ruído branco [21, 22, 61]. Assume-se que o comprimento

do sinal é uma potência de 2, ou seja, $N = 2^J$. Parametriza-se a escala $s = 2^j$, $j \in \{0, 1, 2, \dots, J\}$. Para a frequência de modulação iremos definir que $\xi = 0$, pois desejamos representar somente a envoltória da gaussiana. O deslocamento será o mesmo $\tau = l$, $l \in \{0, 1, \dots, N\}$.

Os parâmetros do dicionário base foram discretizados conforme descrito na Tabela 2. Este dicionário será empregado no algoritmo de decomposição e a partir dos resultados será analisada a relevância dos parâmetros assim definidos.

Tabela 2: Discretização de Parâmetros do Dicionário 1 (Dicionário Base).

Dicionário	Parâmetros	Intervalos
Bateman	ρ	$\{0,2; 0,3; \dots; 2\}$
	η	$\{0,4; 0,6; \dots; 2\}$
	s	$\{0,06; 0,07; \dots; 0,14\}$
	τ	$\{0; 1; \dots; 512\}$
	ϕ	$\{0; \pi\}$
	Condições	$\rho < \eta$
Sigmoide-Exponencial	ρ	$\{0,3; 0,6; \dots; 3\}$
	η	$\{0,5; 0,8; \dots; 2,9\}$
	s	$\{0,06; 0,07; \dots; 0,14\}$
	τ	$\{0; 1; \dots; 512\}$
	ϕ	$\{0; \pi\}$
Chi-Quadrado	χ	$\{2,7; 2,73; \dots; 5,37\}$
	s	$\{0,06; 0,07; \dots; 0,14\}$
	τ	$\{0; 1; \dots; 512\}$
	ϕ	$\{0; \pi\}$
Gabor	s	$\{2^{-6}; 2^{-5}; \dots; 1; \dots; 2^5; 2^6\}$
	τ	$\{0; 1; \dots; 512\}$
	ϕ	$\{0; \pi\}$

Com base nos parâmetros indicados na Tabela 2, podemos definir a cardinalidade de cada sub-dicionário e a cardinalidade total do dicionário. A cardinalidade de cada subdicionário pode ser calculada da seguinte forma

$$\#D = n_\rho n_\eta n_s n_\tau n_\phi n_\xi \quad (81)$$

onde n é a quantidade de elementos de cada parâmetro. A cardinalidade calculada a partir da discretização ilustrada pela Tabela 2 é apresentada pela Tabela 3. A partir

disso, podemos ter a dimensão da complexidade do Dicionário 1 (base), lembrando que a cardinalidade está diretamente relacionada ao custo computacional do algoritmo de decomposição.

Tabela 3: Cardinalidade do Dicionário 1.

Dicionário	Cardinalidade #D
Bateman	6.294.528
Sigmoide-Exponencial	5.829.632
Chi-Quadrado	828.416
Gabor	13.312
Total	12.965.888

Os parâmetros analisados são ρ , η e ξ pois tais parâmetros são responsáveis por definir a forma da curva das funções presentes no dicionário. O deslocamento τ indica a posição em que o átomo aparece e possui uma distribuição uniforme no tempo de acordo com o sinal. Já a fase ϕ com valores sendo 0 ou π indica se o átomo é positivo ou negativo. Com isso, não há necessidade de se analisar as ocorrências desses dois últimos parâmetros. A escala s está implícita nos parâmetros ρ e η para os sub-dicionários de *Bateman* e Sigmoide-Exponencial. A Figura 17 ilustra as formas de ondas contidas nos sub-dicionários usados na decomposição empregando-se a discretização da Tabela 2 para um valor fixo de τ e ϕ , enquanto variam-se somente os parâmetros mais relevantes para a mudança da forma de onda como: s, ρ, η . A análise dos parâmetros é feita para cada subdicionário de forma separada, onde a escolha do subdicionário de cada átomo anterior não interfira, ou enviese o processo de seleção do átomo seguinte.

2.9.1 Resultado da Discretização de *Bateman*

Começando pelo dicionário de *Bateman*, visualizou-se a distribuição dos parâmetros ρ e η escolhidos na fase de decomposição. Pela Figura 18, vemos as combinações dos parâmetros disponíveis a serem selecionados pelo MP à esquerda e à direita os parâmetros do átomos selecionados durante a decomposição. Dessa forma, conclui-se que os valores dos parâmetros de ρ e η estão concentrados na diagonal principal, o que indica que a diferença entre os valores é pequena. Outra faixa que há uma maior concentração está associada a um valor fixo mais alto de η dentro dos limites da discretização.

Complementando os resultados obtidos a Figura 19, contabiliza-se a ocorrência da distribuição dos parâmetros ρ e η para as 50 primeiras iterações de cada bloco. É

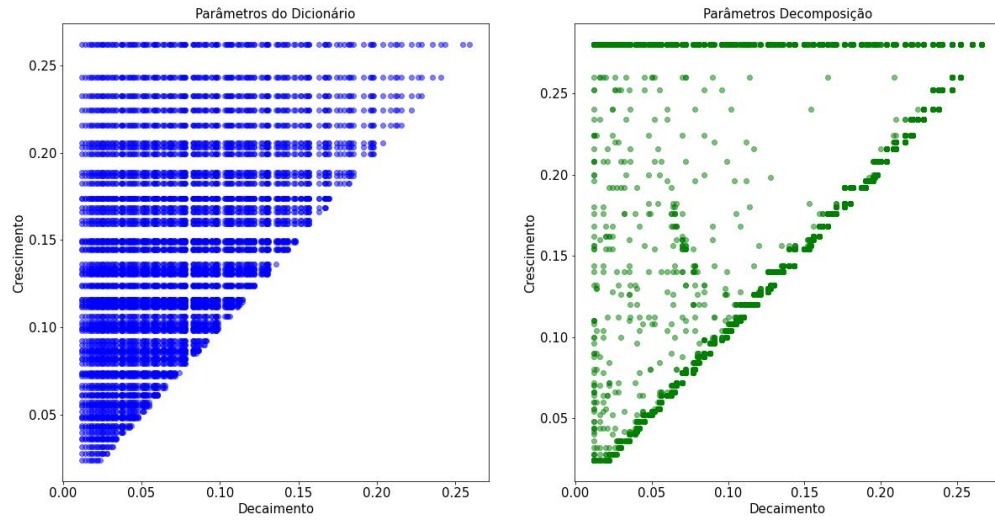


Figura 18: Combinação dos parâmetros ρ e η do dicionário de *Bateman* e selecionados pelo *Matching Pursuit*.

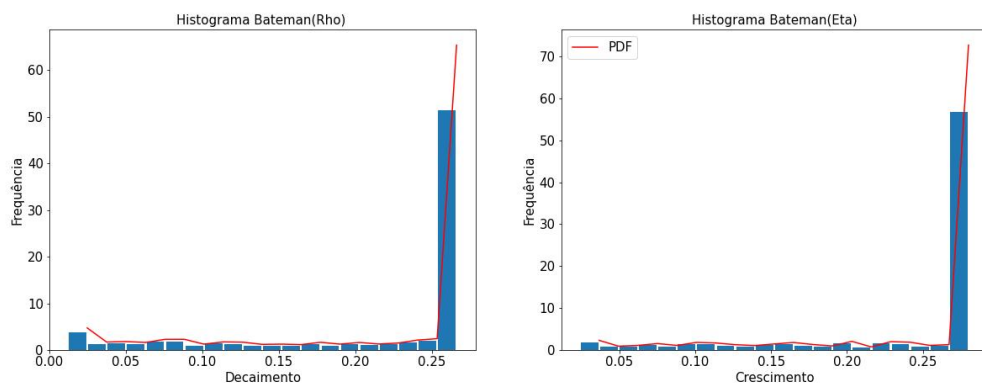


Figura 19: Histograma dos parâmetros ρ e η do dicionário de *Bateman* para 50 átomos por bloco.

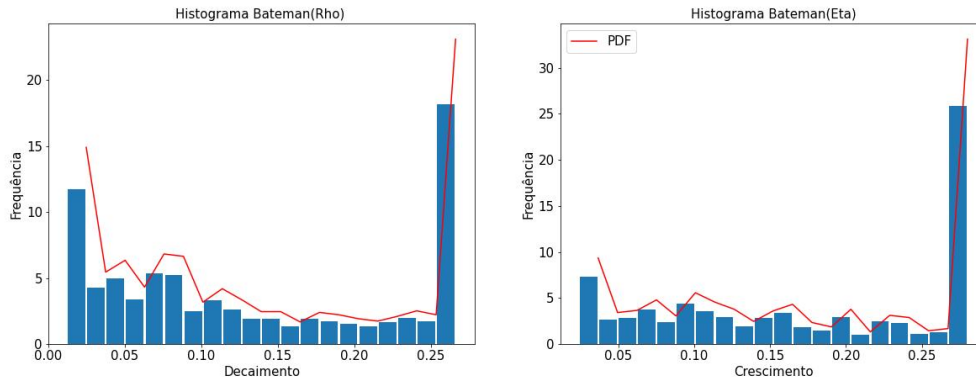


Figura 20: Histograma dos parâmetros ρ e η do dicionário de *Bateman* para 10 átomos por bloco.

possível observar uma predominância de ambos os parâmetros na extremidade superior. Conforme limitamos o número de iterações a 10 por bloco, percebe-se pela Figura 20 uma distribuição mais equilibrada e uma diminuição no percentual de parâmetros mais elevados.

Para facilitar o entendimento destes fenômenos, precisamos analisar o que ocorre com a forma de onda da função ao variarmos os parâmetros ρ e η . Como já vimos, essas variáveis representam o decaimento e o crescimento da função respectivamente e quanto mais alto seu valor mais rápido é o tempo de subida ou de descida da função. Logo, para valores altos de ρ e η , temos funções mais estreitas com características aproximadas de um impulso, enquanto que para valores menores temos uma suavização da curva.

A razão de átomos com características impulsivas aparecerem com mais frequência após um maior número de iterações se dá pelo fato do *Matching Pursuit* tender a corrigir erros de aproximação gerados em iterações anteriores, e o resíduo se aproximar cada vez mais de um ruído, à medida que se passam as iterações. Para exemplificar este fenômeno, simularemos um sinal SCR representado por uma única e simples função de *Bateman* com $\tau = 10,5$. Contudo, a discretização do deslocamento é definida com valores que não cobrem valores reais, tendo como valores mais próximos $\tau = 10$ ou $\tau = 11$.

Após avaliarmos a decomposição deste sinal, ilustrado na Figura 21, podemos observar que uma pequena diferença entre o sinal/resíduo (representado pela curva azul) e o átomo presente no dicionário escolhido a cada iteração (curva pontilhada vermelha) resulta em um resíduo com característica de ruído e nas próximas iterações o algoritmo compensa encontrando átomos com essas características para minimizá-lo.

Uma das soluções para obter um maior número possível de átomos coerentes sem que comprometa a qualidade da decomposição é por meio do uso de critérios de parada com base na razão sinal-ruído ou na taxa de aproximação. Além disso, para os 5 sinais utilizados há muitos átomos disponíveis no dicionário que não foram selecionados. A fim de otimizar a discretização do dicionário e reduzir o custo computacional, vamos compor um novo dicionário com parâmetros que possuam uma probabilidade maior de serem selecionados com base nos resultados prévios.

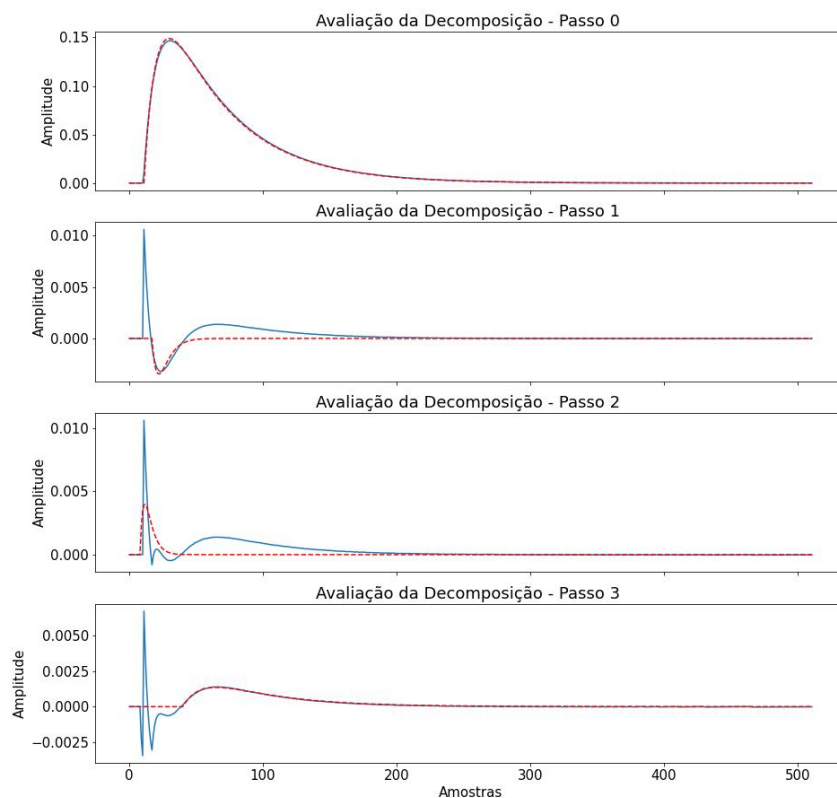


Figura 21: Exemplo da decomposição realizada pelo MP para um sinal artificial para ilustrar o erro gerado após o algoritmo encontrar o átomo mais correlacionado.

2.9.2 Resultado da Discretização da Sigmoide-Exponencial

No caso da sigmoide-exponencial, observa-se o mesmo padrão que ocorreu com o dicionário de *Bateman*. Através da Figura 22, é possível observar as combinações de parâmetros do dicionário de sigmóides-exponenciais e as combinações dos átomos escolhi-

dos durante a decomposição de 5 sinais. Assim como no dicionário de *Bateman*, há vários átomos que não foram selecionados, o que acaba gerando um desperdício de tempo computacional. Na Figura 23, verifica-se um padrão relativamente uniforme entre os parâmetros com exceção das extremidades, que conforme já mencionado, estão representados como átomos de características impulsivas. A fim de otimizar o processo, iremos obter uma nova parametrização com parâmetros com probabilidade maior de serem selecionados.

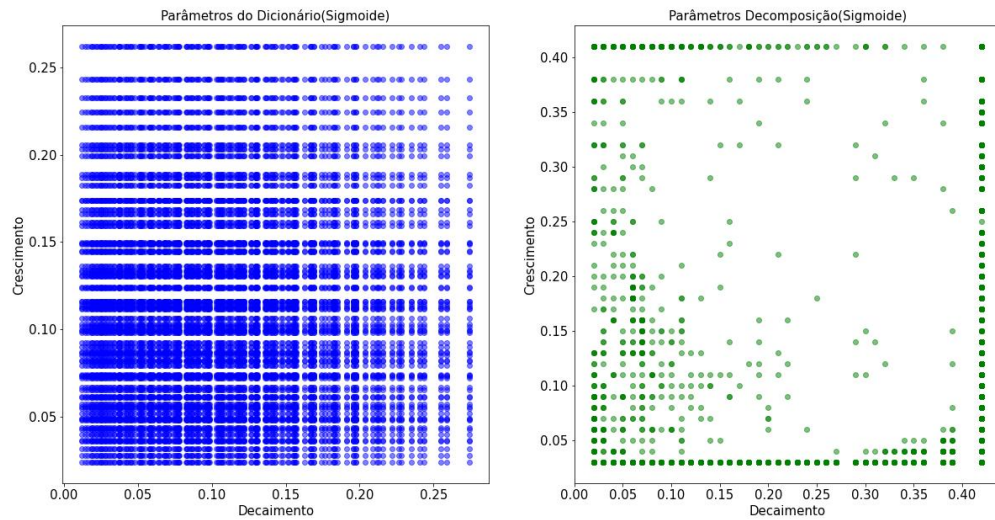


Figura 22: Combinação dos parâmetros ρ (decaimento) e η (subida) do dicionário de Sigmoide-Exponencial e selecionados pelo *Matching Pursuit*.

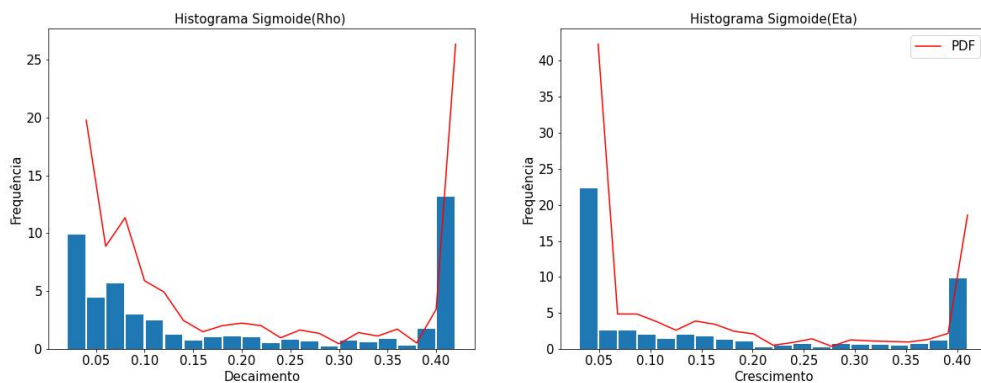


Figura 23: Histograma dos parâmetros ρ (decaimento) e η (subida) do dicionário de Sigmoide-Exponencial para 10 átomos por bloco.

2.9.3 Resultado da Discretização de Chi-Quadrado

Da mesma forma, para o dicionário de Chi-quadrado também têm-se características semelhantes aos dicionários anteriores. Na Figura 24, observam-se as combinações de parâmetros do dicionário de Chi-quadrado e as combinações de parâmetros dos átomos escolhidos durante a decomposição. Nota-se que o padrão da parametrização original é granular demais a ponto de muitos átomos não serem escolhidos durante a decomposição.

Na Figura 25, observa-se que os parâmetros de escala e grau de liberdade dos dez primeiros átomos escolhidos de cada bloco se concentram nas extremidades das faixas de valores possíveis, e apresentam comportamento uniforme no restante das faixas.

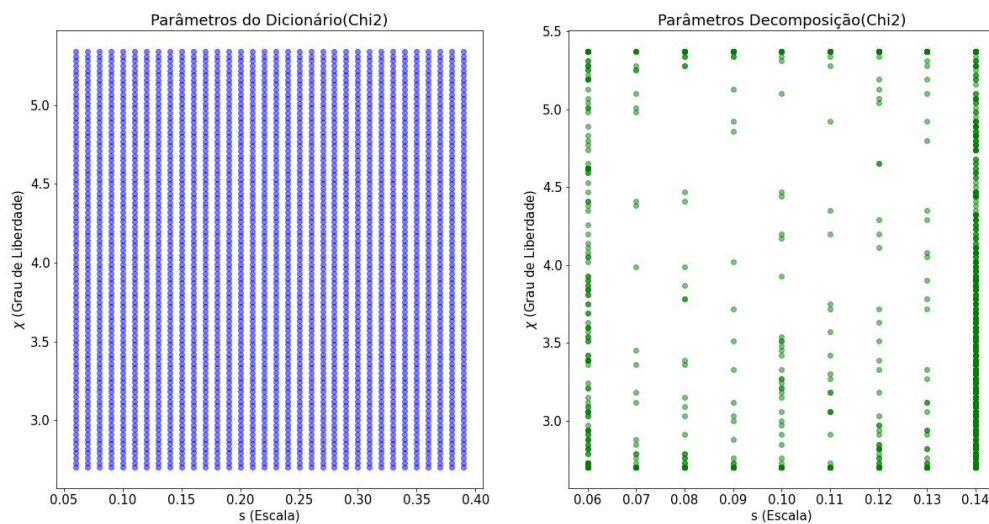


Figura 24: Combinação dos parâmetros χ (grau de liberdade) e s (escala) do dicionário de Chi-quadrado e selecionados pelo *Matching Pursuit*.

2.9.4 Resultado da Discretização de Gabor

Para o dicionário de Gabor, o parâmetro a ser analisado é apenas a escala s , pois neste caso, é o único parâmetro necessário para mudar a forma de onda como vimos na parametrização definida em (76). O uso da discretização diádica para o dicionário de gaussianas não moduladas (de frequência nula) é o que possui menor cardinalidade. A Figura 26 ilustra o histograma dos valores de escala escolhidos durante a decomposição para os 50 primeiros átomos de cada bloco. Observe que há uma menor ocorrência de

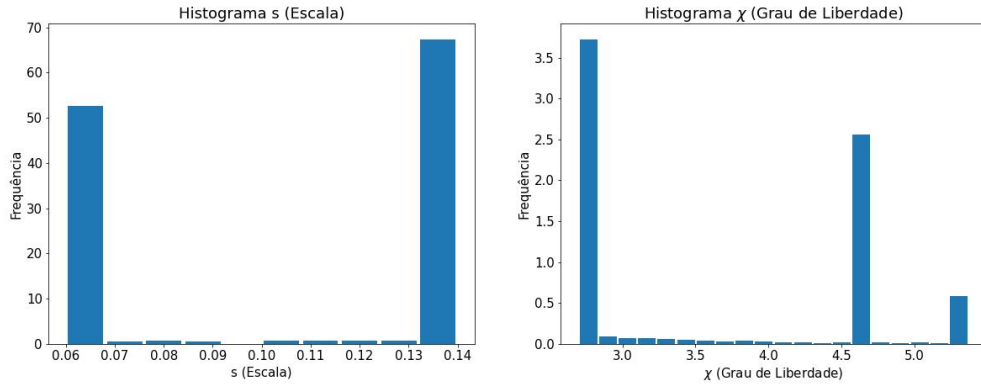


Figura 25: Histograma dos parâmetros χ (grau de liberdade) e s (escala) do dicionário de Chi-quadrado para 10 átomos por bloco.

valores de s menores que 1 e que o valor mínimo de escala foi $s = 0.125$ que está associado a um átomo impulsivo. Com base no histograma, é possível manter a discretização diádica para o dicionário de Gabor com valores mínimo de $s_{min} = 0.125$ e máximo de $s_{max} = 128$.

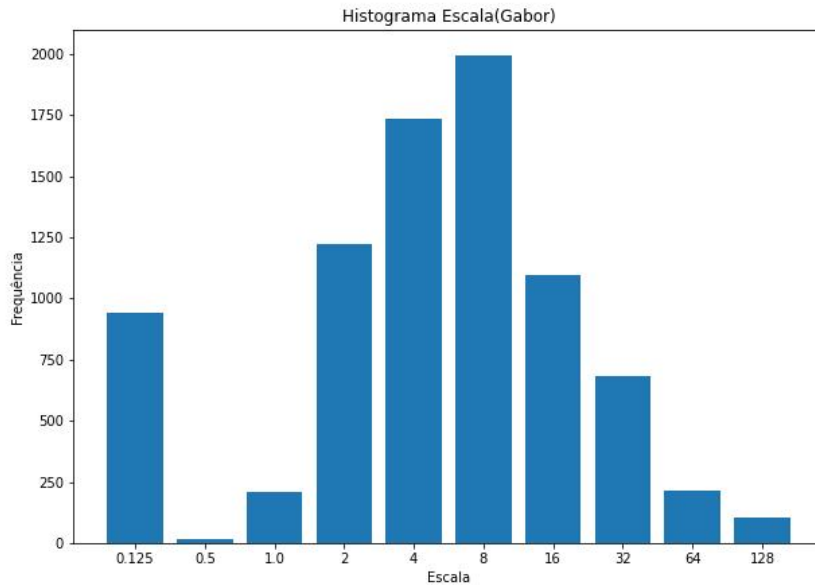


Figura 26: Histograma dos parâmetros s do dicionário de Gabor para 50 átomos por bloco.

2.9.5 Resultado da Discretização de Múltiplos Dicionários

Agora, antes de compor um novo dicionário subamostrado do dicionário base referente à Tabela 2, vamos avaliar o desempenho da decomposição avaliando a razão sinal-ruído para cada subdicionário usado separadamente e a decomposição usando o dicionário completo com todos os subdicionários conjuntamente para os sinais *eda1* a *eda5*, conforme apresentado na Figura 27. É possível notar, pelas curvas do SNR médio a cada iteração do algoritmo MP, a tendência sempre ascendente o que significa que a cada iteração o algoritmo está se aproximando da representação do sinal usando a decomposição atômica.

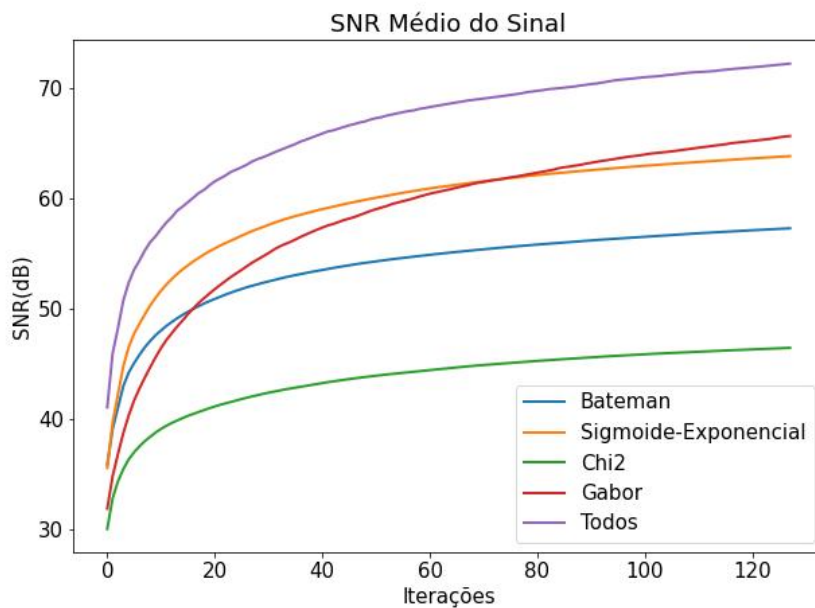


Figura 27: SNR médio da decomposição para os 5 sinais *eda1* a *eda5* usando cada subdicionário separadamente e o dicionário completo.

Primeiramente, vamos comparar os subdicionários separadamente. O Chi-quadrado obteve o pior desempenho quando comparado às outras funções. Os dicionários de *Bateman* e de *Sigmoide-Exponencial* obtiveram desempenhos equivalentes e melhores que o dicionário de *Gabor* para as primeiras iterações, enquanto que ao decorrer das iterações, há uma melhora por parte da *Sigmoide-Exponencial* quando comparada ao *Bateman* e a curva de *Gabor* ascende mais rapidamente no decorrer das iterações, superando os demais. Verifica-se que conforme o resíduo passa a ter característica de ruído, a função de *Gabor* tende a se sobressair perante as outras. Já para as primeiras iterações, onde os resíduos possuem mais energia e SCRs bem definidos, há uma preferência do algoritmo escolher

átomos de *Bateman* ou de Sigmoides-Exponenciais.

Usando todos os subdicionários conjuntamente, é possível representar o sinal de EDA com maior eficiência, em termos de SNR, do que com subdicionários individuais, pois há uma gama maior de possibilidades para escolha do melhor átomo com características mais aderentes ao sinal ao longo do processo de decomposição. Além disso, através do livro de estruturas podemos observar a escolha do tipo de subdicionário com o decorrer das iterações. Na Figura 28, observam-se histogramas que variam conforme o critério de parada de acordo com o número de iterações por bloco, no qual é possível observar primeiro que átomos de chi-quadrado não são escolhidos, dado a discretização estipulada, o que indica que é possível desconsiderar esse subdicionário. Além disso, podemos confirmar uma predominância de átomos de Gabor conforme o aumento do número de iterações.

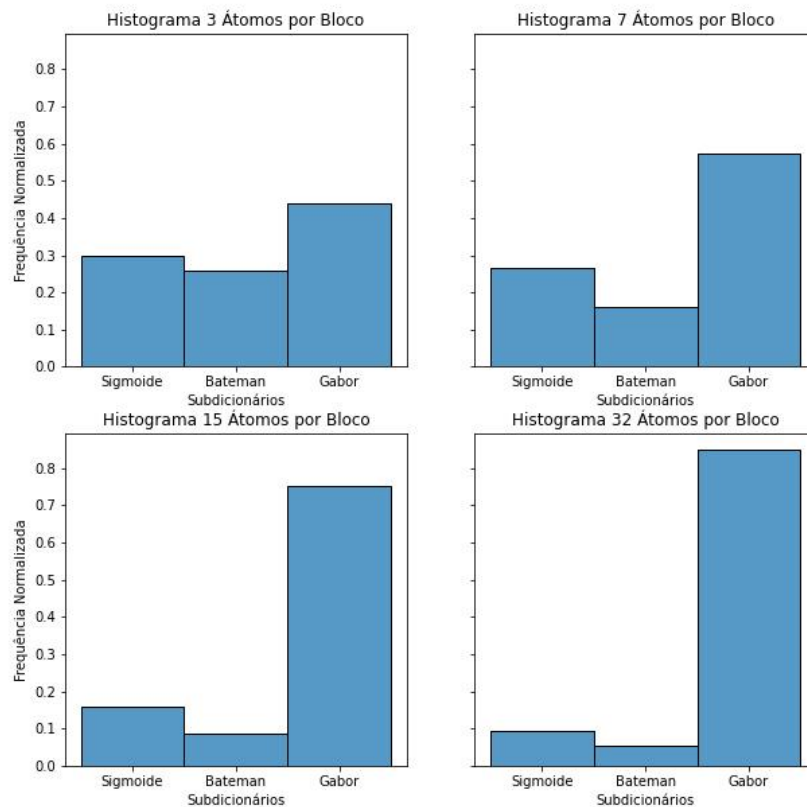


Figura 28: Histograma das frequências dos tipos de subdicionários escolhidos de acordo com o número de iterações por bloco.

Com isso, podemos formar uma nova discretização de parâmetros para o dicionário

otimizado, cujo o objetivo é justamente reduzir a cardinalidade, conseqüentemente o tempo computacional, resultando em um bom aproveitamento dos átomos durante a decomposição. Como a nova discretização foi definida estatisticamente conforme os resultados do dicionário base, não temos como definir os intervalos entre os parâmetros. Em contrapartida, a visualização da nova distribuição de parâmetros para cada subdicionário pode ser feita através das figuras já apresentadas (Figura 18, Figura 22, Figura 24, Figura 26). Além disso é possível definir a cardinalidade de cada subdicionário e a total conforme apresentado na Tabela 4. A cardinalidade do dicionário para a nova configuração é 14 vezes menor que a do dicionário base. Apesar da redução cardinalidade ser significativa, é necessário avaliar o desempenho da decomposição para esta nova configuração. Para isso, vamos realizar a decomposição para novos sinais (*eda6* a *eda10*).

O objetivo de realizar a decomposição para um novo conjunto de sinais é para evitar um viés da discretização do dicionário base com relação a discretização do dicionário ajustado. A Figura 29 ilustra a comparação do desempenho entre as duas discretizações para o novo conjunto de sinais. Ambos os dicionários possuem SNRs semelhantes logo nas primeiras iterações e com o decorrer da decomposição o dicionário base tende a apresentar maior qualidade para representar o sinal. Contudo, como já esperado, o novo dicionário ajustado tende a ser inferior ao dicionário original por ser um subconjunto do mesmo. Apesar disto, a diferença entre os SNRs não é significativa, além das vantagens que é ter um dicionário mais compacto que implica em menor complexidade computacional. Com isso, podemos assumir essa discretização que será empregada nas próximas etapas do projeto.

Tabela 4: Cardinalidade do Dicionário Ajustado.

Dicionário	Cardinalidade #D
Bateman	523.264
Sigmoide-Exponencial	376.832
Gabor	6.144
Total	906.240

Na Figura 30, pode-se observar um passo a passo das primeiras iterações da decomposição de um bloco com 128 amostras de um sinal de EDA, utilizando-se o dicionário ajustado. Nota-se que os primeiros átomos escolhidos possuem uma boa correlação com o sinal original (curva azul), com a forma de onda do átomo escolhido (curva pontilhada em vermelho) a se sobrepondo ao sinal original. Contudo, após algumas iterações o resíduo

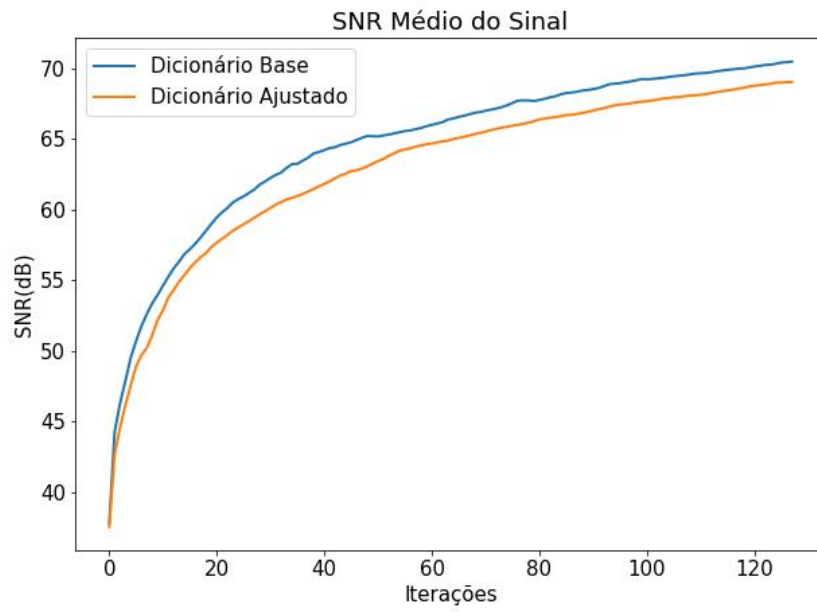


Figura 29: SNR médio da decomposição para os 5 sinais *eda6* a *eda10* usando o dicionário original e o dicionário ajustado.

remanescente possui cada vez mais características impulsivas e ruidosas, reduzindo-se a taxa de aproximação, ou seja, os átomos escolhidos passam a ser menos coerentes com relação ao sinal.

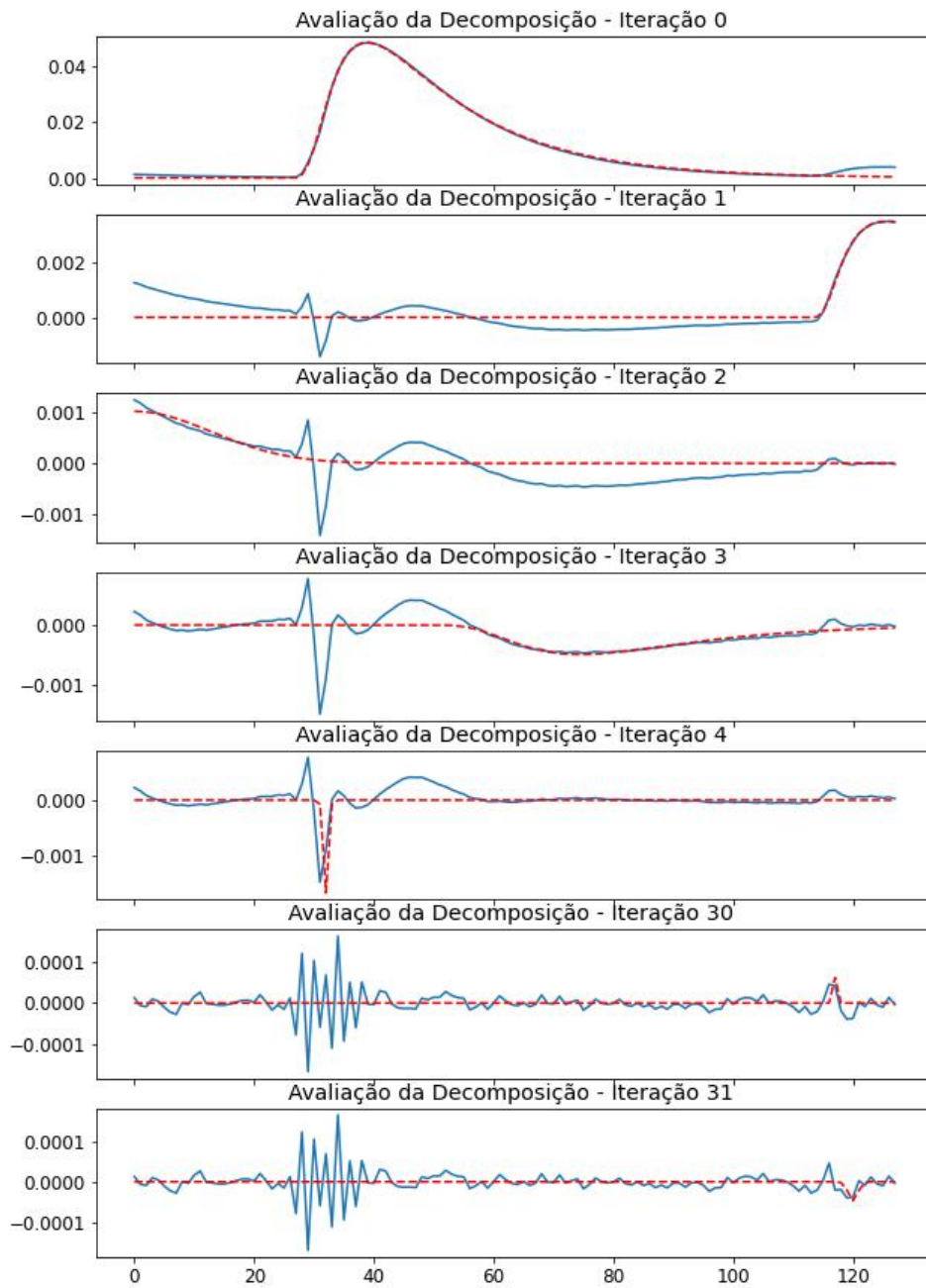


Figura 30: Avaliação da decomposição do sinal de EDA por iteração.

3 SELEÇÃO DE SUBDICIONÁRIO A USAR EM ITERAÇÕES VIA REDES NEURAS ARTIFICIAIS

De maneira geral, o modelo elaborado neste capítulo consiste em uma etapa adicional atrelada à decomposição atômica vista no Capítulo 2. A Figura 31 ilustra o novo modelo de decomposição com a etapa de seleção de dicionário inserida no modelo anteriormente discutido na Figura 16. A arquitetura do modelo classificador responsável por decidir qual o subdicionário mais adequado a ser usado a cada iteração é uma rede neural artificial.

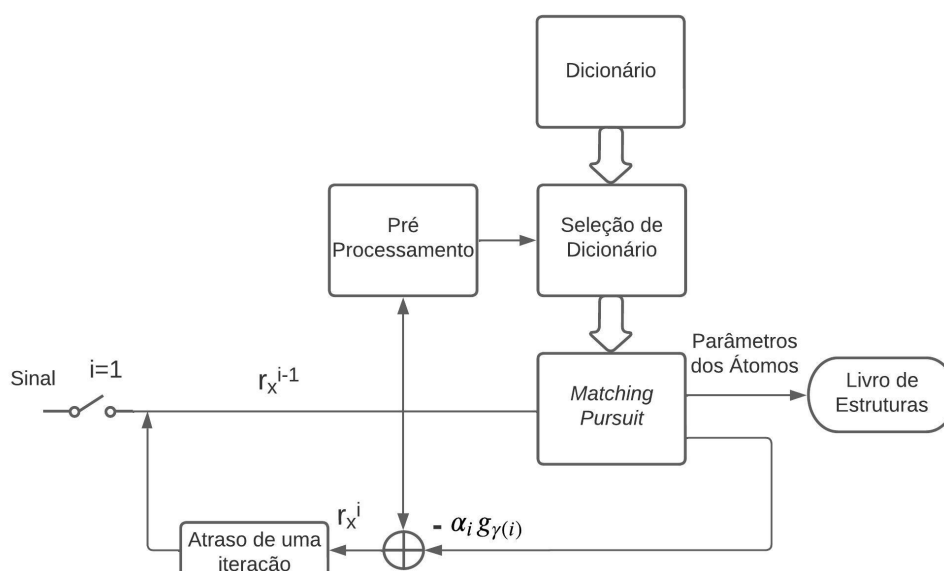


Figura 31: Diagrama de blocos do modelo de decomposição atômica com seletor de subdicionários.

Continua-se a decompor sucessivamente o resíduo de cada iteração do algoritmo MP, porém o módulo de seleção de dicionário escolhe um subconjunto de átomos (um subdicionário) que podem ser empregados na decomposição do resíduo a partir do próprio resíduo pré-processado. Conseqüentemente, a saída do módulo de seleção de dicionário é o tipo de dicionário. O objetivo é reduzir a cardinalidade do dicionário, transformando um problema de múltiplos dicionários para um único dicionário a cada iteração do MP, reduzindo, assim, a complexidade computacional. A desvantagem da utilização de um

dicionário composto por múltiplos subdicionários no MP é o aumento no custo computacional. A seleção da classe de átomos, isto é, do subdicionário, mais adequado para o resíduo corrente é uma forma de redução do encargo computacional.

Uma solução para a escolha de um subdicionário é usar um método de classificação. Métodos de classificação são técnicas utilizadas na área de aprendizado de máquina para treinar um modelo a partir de um conjunto de dados rotulados, com o objetivo de prever a classe ou categoria de novas observações [67]. Esses métodos são amplamente utilizados em diversas áreas, como medicina [68], finanças [69], marketing, processamento de sinais [70], entre outras, para resolver problemas de classificação binária ou multi-classe [67].

Existem vários algoritmos de classificação, cada um com suas vantagens e desvantagens, e a escolha do método mais adequado depende das características dos dados e dos objetivos do projeto. Alguns exemplos de algoritmos de classificação incluem Árvore de Decisão [71], Regressão Logística [72], *K-Nearest Neighbors* [73], *Support Vector Machine* (SVM) [74] e Redes Neurais Artificiais [75]. O sucesso da aplicação de métodos de classificação depende não apenas da escolha do algoritmo mais adequado, mas também da qualidade dos dados, da forma como são pré-processados antes de treinar o modelo e da estratégia de otimização dos hiperparâmetros.

A estratégia proposta neste trabalho utiliza as Redes Neurais Artificiais (RNA). Empregamos uma RNA para adivinhar o subdicionário a ser utilizado para decompor o resíduo atual do sinal. Em cada iteração, os resíduos decimados são apresentados à RNA com o objetivo de escolher o subdicionário mais coerente ou adequado para representar as características do resíduo. Com isso, a cada iteração, o algoritmo MP restringirá a pesquisa pelo átomo mais correlacionado com o resíduo dentro do subdicionário indicado pelo classificador RNA. Dependendo da complexidade do modelo, isto pode resultar em uma redução no custo computacional em comparação à demanda gerada pelo uso do dicionário completo no algoritmo MP. Neste capítulo, são apresentados os procedimentos empregados para o desenvolvimento e a implementação do modelo e a arquitetura da RNA proposta para a classificação de subdicionários aplicadas a decomposições atômicas. Em seguida, são apresentadas as técnicas utilizadas para escolher os hiperparâmetros da RNA, treinar a RNA/modelo (ajustar seus parâmetros ou pesos), validar e testar o modelo utilizando diferentes algoritmos de otimização por gradiente descendente, e controlar o *overfitting*. São abordados também o pré-processamento do resíduo e as métricas

utilizadas para medir o desempenho do modelo.

3.1 Sobre Redes Neurais Artificiais

Uma rede neural artificial (RNA) é um modelo computacional não linear inspirado em sistemas nervosos. A RNA emprega um conjunto de neurônios interconectados procurando reproduzir propriedades de aprendizado, associação e generalização [76]. Uma RNA pode ser definida como um processador massivo e paralelamente distribuído, modelado com base no funcionamento simplificado de neurônios, que possui a habilidade de aprender e, conseqüentemente, generalizar. A generalização está relacionada à RNA ser capaz de estimar saídas coerentes para dados de entrada que não foram apresentados durante o processo de aprendizagem [77]. Devido a isso, RNAs são capazes de resolver problemas complexos em diversas áreas de aplicação como processamento de sinais, aproximação de funções, previsão de séries temporais, otimização, classificação e reconhecimento de padrões, podendo ser considerada como aproximadores universais [78].

3.1.1 O Neurônio Artificial

O modelo mais simples de RNA emprega apenas um único neurônio, ilustrado na Figura 32, é chamado de Perceptron, e foi proposto em [75]. Nesse modelo, identificamos o sinal de entrada \mathbf{x} conectado aos seus respectivos pesos sinápticos \mathbf{w} . O combinador linear é usado para somar as entradas ponderadas pelas sinapses dos neurônios produzindo

$$v_k = \sum_{j=1}^m w_{kj}x_j + b_k. \quad (82)$$

O *bias* ou viés b_k é um parâmetro adicional do neurônio com o intuito de aumentar os graus de liberdade da função por ele implementada, que é usado para ajustar a saída em conjunto com a soma ponderada das entradas do neurônio, ou seja, o *bias* é uma constante que permite ajustar melhor a saída ao valor desejado para a entrada fornecida.

A função de ativação $\varphi(\cdot)$ do neurônio é responsável por restringir a amplitude de saída introduzindo não-linearidade no processamento realizado pelo neurônio. De maneira geral, a Equação de saída de um neurônio após aplicar a função de ativação genérica é

$$y_k = \varphi(v_k). \quad (83)$$

Tendo em vista a simplicidade desejada para as unidades processadoras de uma RNA (o neurônio), geralmente, aplica-se uma função algébrica ao valor v_k , independentemente de valores passados. Geralmente, esta função é monotonicamente não-decrescente e apresenta uma não-linearidade associada ao efeito da saturação.

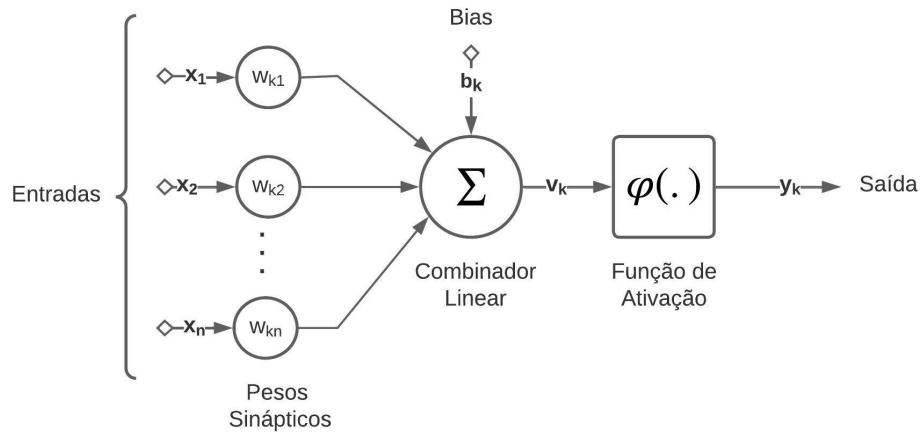


Figura 32: Modelo Perceptron.

Devido ao mapeamento não linear entre entrada e saída nos neurônios, uma RNA pode aprender mais do que relações lineares entre as variáveis dependentes e independentes. Isso é especialmente importante nas camadas escondidas de uma RNA. Nas camadas de saída, as funções de ativação podem ter finalidades específicas, dependendo do problema que a rede neural está tentando resolver [77]. Uma característica importante de uma função de ativação é que deve ser diferenciável para que possamos implementar uma estratégia de otimização de propagação para calcular os erros ou perdas com relação a pesos e eventualmente otimizar pesos usando a descida do gradiente ou qualquer outra técnica de otimização para reduzir erros [79].

Dentre as funções de ativação mais comuns, têm-se a função linear, sigmoide, a tangente hiperbólica, ReLU (unidade linear retificada) e SoftMax [79,80]. A mais básica é a função linear pois ela não altera a saída de um neurônio, apenas multiplica o valor na entrada por uma constante para produzir a saída. Geralmente, ela é utilizada nas camadas de saída em RNAs usadas em problemas de regressão. A função de ativação sigmoide

$$\varphi(z) = \frac{1}{1 + e^{-z}} \quad (84)$$

é comumente utilizada em RNAs com propagação positiva (*Feedforward*) e que empregam apenas números positivos, em RNAs multicamadas e com valores densamente quantizados. A função sigmoide aparece nas camadas de saída das arquiteturas RNA e são usadas para prever probabilidades baseadas na saída e foi aplicada com sucesso em problemas de classificação binária, modelagem de tarefas de regressão logística, bem como outros domínios de rede neural [81]. Entretanto, algumas de suas desvantagens, que incluem gradientes acentuados durante a retropropagação de camadas mais profundas para as camadas de entrada, saturação de gradiente, convergência lenta e saída centralizada diferente de zero, o que faz com o que a atualização dos gradientes se propaguem em diferentes direções [81]. Além disso, como se pode observar na Figura 33, a função sigmoide não é centrada em zero. Ainda, ela apresenta platôs para valores de entrada muito altos ou muito baixos, o que faz com que a derivada nessas regiões se aproxime de zero.

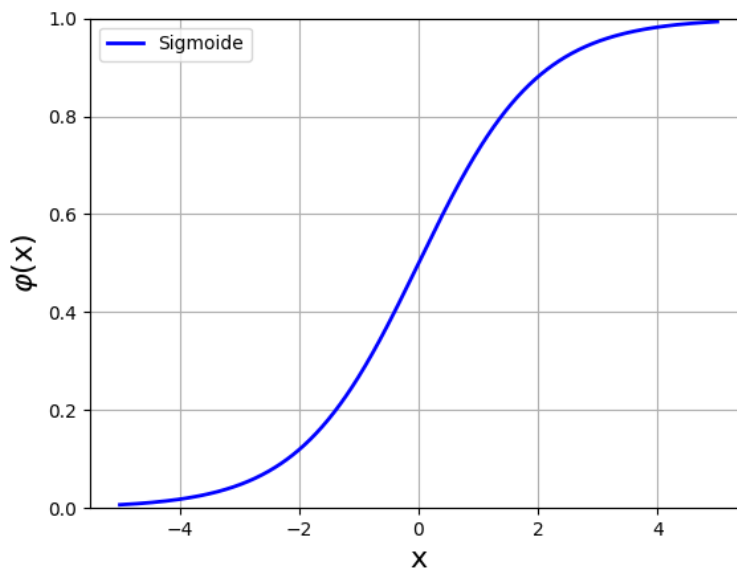


Figura 33: Função de Ativação Sigmoide.

Outra função de ativação muito utilizada é a ReLU (*Rectified Linear Unit*), ou unidade linear retificada, Figura 34. Ela produz resultados no intervalo $[0, \infty[$. A função ReLU retorna 0 para todos os valores negativos, e o próprio valor para valores positivos. É uma função computacionalmente leve, entretanto não é centrada em zero. A vantagem de usar a função ReLU é que todos os neurônios não são ativados ao mesmo tempo. Isso implica que um neurônio será desativado somente quando a saída de transformação linear for zero.

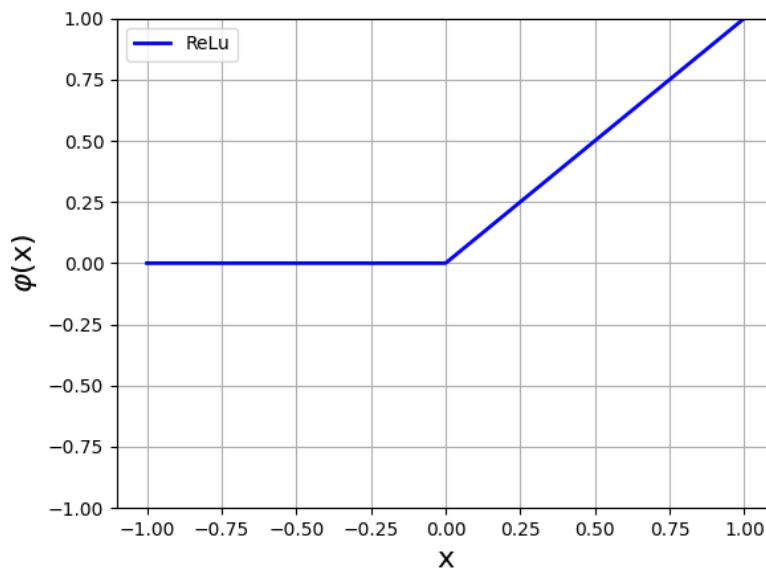


Figura 34: Função de Ativação ReLu.

3.1.2 Perceptron Multicamadas

Para problemas mais complexos, são necessários diversos neurônios dispostos e conectados em camadas, formando uma RNA. As RNAs do tipo MLP (*Multilayer Perceptron*) têm sido utilizadas com sucesso para a solução de vários problemas envolvendo altos graus de não-linearidade. RNAs de múltiplas camadas são estruturas constituídas por uma entrada, uma ou n camadas ocultas e uma camada de saída, onde as unidades de camadas vizinhas são conectadas uma às outras, mas não há conexões entre unidades na mesma camada.

Cada camada pode ter um número diferente de neurônios, como ilustrado pela Figura 35 [82]. Seja uma RNA de três camadas, a primeira camada recebe o vetor de entrada $\mathbf{x} = [x_1, \dots, x_M]^T$, a camada intermediária tem L neurônios, podemos escrever a saída de uma unidade (um neurônio) da camada de saída como

$$y_k(\mathbf{x}, \mathbf{w}, \mathbf{b}) = \varphi^{(2)} \left(\sum_{l=1}^L w_{kl}^{(2)} \varphi^{(1)} \left(\sum_{i=1}^M w_{li}^{(1)} x_i + b_j^{(1)} \right) + b_k^{(2)} \right) \quad (85)$$

Anteriormente, mencionamos algumas funções de ativação. Na camada de saída, em geral, a função sigmoide é útil em problemas de classificação binária, já que seus resultados, em $[0, 1]$, podem ser interpretados como a probabilidade de determinada instância

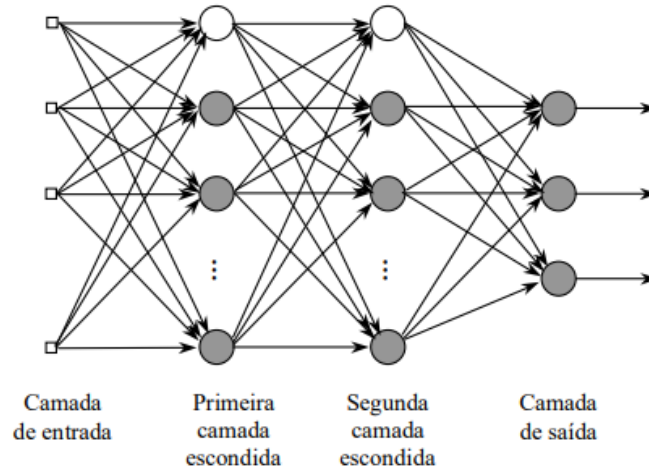


Figura 35: Modelo Perceptron Multicamadas.

pertencer ou não a determinada classe. A função Softmax é uma generalização da função sigmoide para casos não-binários (multi-classe), na qual é possível representar a saída com uma densidade de probabilidades [80]. A função softmax é

$$\varphi(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}, \quad (86)$$

na qual, a variável $\mathbf{z} = [z_1, z_2, \dots, z_K]^T$ designa o vetor de neurônios de saída, i representa o índice da saída e j representa os índices de todos os neurônios da camada. Vale notar que a função de ativação softmax é calculada de uma forma diferente das demais apresentadas, uma vez que a saída de um neurônio depende dos outros neurônios da camada. Ela produz valores no intervalo $[0, 1]$, que somadas se igualam a 1. Dessa forma, num problema com 3 classes, por exemplo, a função softmax vai produzir 3 valores, que somam 1, onde cada valor representa a probabilidade da instância pertencer a uma das 3 possíveis classes. A maior probabilidade entre elas define a classe escolhida.

O objetivo da rede neural é minimizar a diferença entre a saída produzida para o conjunto de entrada \mathbf{x} e a saída da rede. Isso deve ser feito para um conjunto de amostras de entrada-saída $\{\mathbf{x}_n, \mathbf{y}_n\}, n = 1 \dots N$. Para isso devemos treinar a rede, atualizando os pesos \mathbf{w} e os vieses \mathbf{b} para que o erro seja mínimo, segundo o critério desejado. Este treinamento é do tipo supervisionado, pois conhecem-se as saídas que devem ser produzidas pelas instâncias de entrada. Um algoritmo muito popular para isso é o (*Backpropagation*) com otimização do erro usando descida do gradiente estocástico ou variações deste [83,84].

3.1.3 Treinamento de RNAs: *Backpropagation*

O algoritmo *Backpropagation* é baseado numa regra de aprendizagem que “corrige” o erro durante o treinamento, apresenta-se a entrada, calcula-se o erro para a saída desejada, e a partir dos erros, processam-se e corrigem-se os pesos na direção oposta, atualizando-os da saída para a entrada [85]. Basicamente, o processo de retropropagação do erro é constituído de duas fases: uma fase de propagação do sinal funcional (*feed-forward*) e uma de retropropagação do erro (*backpropagation*). Na fase positiva, um vetor \mathbf{x}_n é aplicado nas unidades de entrada, e seu efeito se propaga pela rede, camada a camada. Finalmente, um conjunto de saídas $\hat{\mathbf{y}}_n$ é produzido como resposta da rede. Durante a fase positiva, os pesos das conexões são mantidos fixos. O erro é obtido a partir de uma função de perda (*loss function*) entre a saída produzida $\hat{\mathbf{y}}_n$ e a desejada \mathbf{y}_n . Na retropropagação do erro, os pesos são ajustados usando uma regra de correção do erro [85].

Em aplicações práticas do algoritmo de retropropagação, o aprendizado é resultado de apresentações repetidas de todas as amostras do conjunto de treinamento ao MLP. Cada apresentação de todo o conjunto de treinamento durante o processo de aprendizagem é chamada de época. O processo de aprendizagem é repetido época após época, até que o conjunto de pesos e limiares estabilize e o erro quadrático médio do conjunto de treinamento convirja para um valor mínimo. É uma boa prática fazer com que a ordem de apresentação das amostras seja feita aleatoriamente de uma época para a outra. Esta aleatoriedade tende a fazer com que a busca no espaço de pesos tenha um caráter estocástico ao longo dos ciclos de treinamento.

Para um dado conjunto de treinamento, o aprendizado por retropropagação pode ser feito atualizando-se os pesos em lotes. Um dos hiperparâmetros a serem definidos no modelo é o tamanho do lote (*batch size*). O tamanho do lote determina a quantidade de amostras de treinamento que constituem uma época. O uso do tamanho do lote mínimo torna a busca no espaço de conexões estocástica, reduzindo as possibilidades do algoritmo ficar preso no mínimo local. Em contrapartida, ao utilizar-se o tamanho máximo é possível obter uma estimativa mais precisa do gradiente [77].

3.1.3.1 Gradiente Descendente

A descida de gradiente (ou gradiente descendente) é um algoritmo de otimização iterativa de primeira ordem para encontrar o mínimo de uma função, geralmente denotada como função de custo, e definida via

$$e = \| y_k - \hat{y}_k \|^2, \quad (87)$$

onde \hat{y}_k é a saída k da rede e y_k é a saída correspondente esperada. Com o objetivo de minimizar o erro, calculam-se as derivadas parciais de e com relação aos pesos \mathbf{w} e os vieses \mathbf{b} usando a regra da cadeia

$$\frac{\partial e}{\partial w_{ij}^c} = \sum \frac{\partial e}{\partial \hat{y}_n^c} \frac{\partial \hat{y}_n^c}{\partial w_{ij}^c}. \quad (88)$$

Solucionando as derivadas temos a atualização dos pesos definidos como

$$w_{k+1}^c = w_k^c - \eta \nabla e(w_k), \quad (89)$$

$$b_{k+1}^c = b_k^c - \eta \nabla e(b_k), \quad (90)$$

onde w_k e b_k são os pesos e os vieses a cada iteração, $\nabla e(w_k) = \frac{\partial e}{\partial w_{ij}^c}$ é o gradiente do erro e η é a taxa de aprendizagem, que indica a velocidade com que os pesos são atualizados. Ela pode ser fixa ou alterada de modo adaptativo e é um hiperparâmetro a ser definido para o treinamento.

Uma das funções de custo comumente usada para classificação binária e multi-classes é a perda por entropia cruzada (*cross-entropy loss*), pois incentiva o modelo a prever a probabilidade correta para a classe verdadeira e a diminuir a probabilidade para as outras classes [86]. Para um modelo de classificação binária, a fórmula da entropia cruzada para classificação é

$$e(y, \hat{y}) = -\frac{1}{N} \left(\sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_1) \log(1 - \hat{y}_1) \right). \quad (91)$$

Para um modelo de classificação multiclass, é realizada a soma da entropia cruzada para

cada classe resultando em

$$e(\hat{y}, y) = -\frac{1}{N} \sum_{i=1}^N \hat{y}_i \log(y_i), \quad (92)$$

onde y é a classe verdadeira e \hat{y} é a classe prevista pelo modelo e N é o conjunto de dados.

3.1.3.2 Otimizador Adam

No contexto de treinamento de RNAs, o otimizador é um algoritmo que ajuda a minimizar (ou maximizar) uma função. Normalmente, é utilizado para minimizar a função de custo, uma função matemática dependente dos parâmetros aprendíveis internos do modelo, que são utilizados no cálculo dos valores de saída \mathbf{y} a partir do conjunto de entradas \mathbf{x} . Por exemplo, tem-se que os parâmetros aprendíveis de uma rede neural são os valores de pesos \mathbf{w} e vieses \mathbf{b} , que são utilizados para computar os valores das saídas. Esses parâmetros são aprendidos ou atualizados na direção que minimiza a função de perda conforme treina-se o modelo. Os parâmetros internos de uma RNA (quantidades deles e seus valores) definem se a mesma é eficiente e eficaz. É por isso que são aplicadas várias estratégias e algoritmos de otimização para calcular as atualizações dos pesos. O otimizador influencia no processo de aprendizado do modelo e conseqüentemente no resultado final.

Neste trabalho, usaremos o otimizador Adam (derivado de *Adaptive Moment estimation*). Ele é uma versão estendida da descida do gradiente estocástico (SGD - *Stochastic Gradient Descent*) no qual utilizam-se estimativas de primeiro e segundo momentos do gradiente para adaptar a taxa de aprendizagem para cada peso da rede neural. As amostras são selecionadas aleatoriamente durante o treinamento levando a uma convergência mais rápida de forma ruidosa devido à sua característica estocástica [87]. Já o algoritmo Adam é proposto como a otimização estocástica mais eficiente que requer apenas gradientes de primeira ordem onde a necessidade de memória é muito pequena [83]. Antes do Adam, muitas técnicas de otimização adaptativas como AdaGrad, RMSP, que têm bom desempenho em relação ao SGD, foram introduzidas; mas, em alguns casos, elas apresentam desvantagens, tais como a generalização do desempenho, que é pior do que a do SGD, em alguns casos [88]. O Adam quando introduzido mostrou-se melhor em termos de generalização do desempenho. Além disso, no Adam, os hiperparâmetros têm interpretações intuitivas e, portanto, exigem menos esforço de ajuste [89].

De maneira geral, Adam é um método que atua sobre o componente do gradiente utilizando a média móvel exponencial do gradiente como também na taxa de aprendizado. O peso e os vieses são atualizados via

$$w_{i+1} = w_i - \frac{\eta \hat{m}_i}{(\sqrt{\hat{v}_i} + \epsilon)} \quad (93)$$

$$v_i = \beta_2 v_{i-1} + (1 - \beta_2) \frac{\partial C}{\partial w_i}, \quad (94)$$

onde

$$\hat{v}_i = \frac{v_i}{(1 - \beta_2^i)} \quad (95)$$

$$\hat{m}_i = \frac{m_i}{(1 - \beta_1^i)} \quad (96)$$

$$m_i = \beta_1 m_{i-1} + (1 - \beta_1) \frac{\partial C}{\partial w_i}, \quad (97)$$

com m_i e v_i inicializados em zero. Vemos que m_i é a média exponencial ponderada dos gradientes passados, v_i é a média exponencial ponderada dos quadrados dos gradientes passados, β_1 e β_2 são hiperparâmetros para serem otimizados, e ϵ é um valor muito pequeno para evitar divisão por zero (de regularização).

3.1.4 Heurísticas para a Generalização do Modelo

No processo de aprendizado de um modelo, o conjunto de dados usados para o treinamento costuma ser iterado várias vezes, na tentativa de fazer com que o modelo consiga capturar melhor a tendência e a variância dos dados. Cada iteração é chamada de época. Um dos problemas encontrados no aprendizado de máquina é saber exatamente por quantas épocas o modelo deve ser treinado para encontrar o resultado otimizado, pois caso seja treinado por poucas iterações pode ocorrer *underfitting* e se treinado demais, acontece o *overfitting*.

O *underfitting* ocorre quando o modelo não consegue capturar a tendência subjacente dos dados. A falta de complexidade da função que mapeia a entrada em uma saída destrói a precisão do modelo de aprendizado de máquina. Nesses casos, as regras do modelo treinado são muito fáceis e flexíveis.

O *overfitting* é um erro de modelagem que ocorre quando uma função é muito

próxima de um conjunto limitado de pontos de dados. Isso significa que se o treinamento produzir *overfitting*, o modelo construído será muito bom em classificar o conjunto de dados usado para o aprendizado, mas desempenhará mal para dados ainda não vistos, ou seja, não generalizará os padrões bem o suficiente.

Para evitar *underfitting* é necessário aumentar a complexidade da função que mapeia a entrada na saída. Já para evitar o *overfitting*, é necessário utilizar algumas técnicas um pouco mais complexas. Uma maneira de controlar o *overfitting* é interromper o processo de treinamento após o desempenho para um conjunto de dados de validação parar de melhorar. O *early stopping* é implementado definindo-se um número máximo de épocas de treinamento. Após cada época, o desempenho do modelo no conjunto de validação é medido e comparado com o desempenho na época anterior. Se o desempenho no conjunto de validação começar a piorar após algumas épocas, o treinamento é interrompido e o modelo é salvo. O modelo salvo é geralmente o que obteve o melhor desempenho no conjunto de validação durante o treinamento.

3.1.4.1 Dropout

O *dropout* [90] é uma técnica em que neurônios são ignorados aleatoriamente durante o treino. Eles são “abandonados” aleatoriamente e temporariamente como ilustrado na Figura 36. Isto significa que a sua contribuição para a ativação dos neurônios é temporariamente removida no sentido direto da propagação, e qualquer atualização de peso não é aplicada a eles. Heuristicamente, quando abandonamos diferentes conjuntos de neurônios, é como se estivéssemos treinando redes neurais diferentes. E, assim, o procedimento de eliminação equivale a calcular a média dos efeitos de um grande número de redes diferentes que de outra forma custaria muito computacionalmente. As diferentes redes se adaptarão de diferentes maneiras, e assim, esperançosamente, o efeito líquido do *Dropout* será reduzir o *overfitting* [80].

Durante o treinamento, os pesos dos neurônios são ajustados para características específicas, o que confere alguma especialização. Os neurônios vizinhos confiam nessa especialização. Esta dependência do contexto para um neurônio durante o treino é referida como co-adaptação complexa. Porém, se levada demasiadamente longe, essa especialização pode resultar num modelo demasiadamente especializado para os dados de treino. Pode-se imaginar que se os neurônios forem abandonados (ou apagados) aleatoriamente

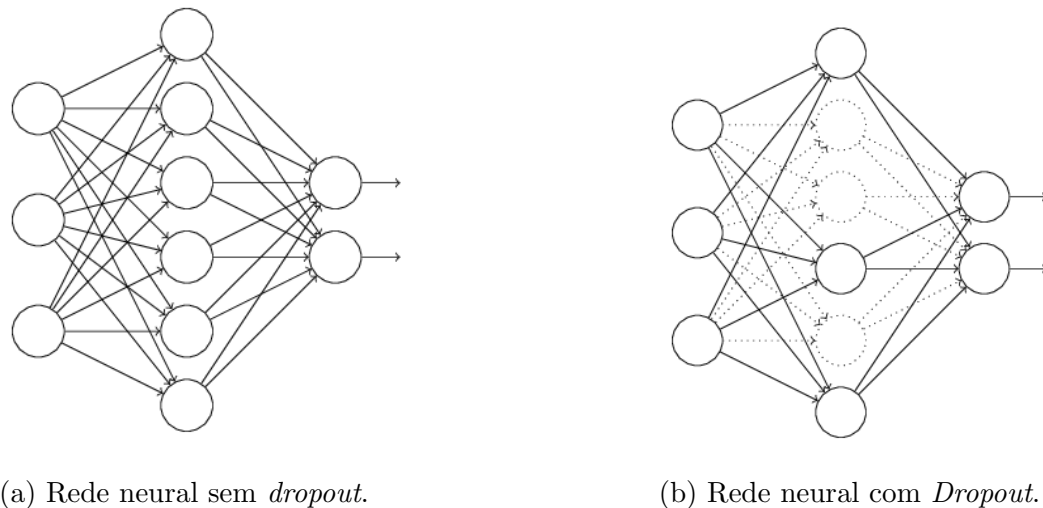


Figura 36: Ilustração do princípio de funcionamento *Dropout*, Fonte: [91]

durante o treino, outros neurônios terão de intervir e lidar com a representação necessária para fazer as previsões originalmente realizadas pelos faltantes [90].

Assim, o efeito do *dropout* é tornar a rede menos sensível a neurônios específicos (e seus pesos). Isto, por sua vez, resulta numa rede capaz de melhor generalização e menos suscetível a *overfitting*. Um hiperparâmetro a ser definido utilizando tal técnica é a quantidade de neurônios que serão desligados. A quantidade usualmente é definida por uma taxa de *dropout* que nada mais é que um percentual do número de neurônios presentes nas camadas escondidas.

3.2 Conjunto de Dados Empregado

O diagrama de blocos ilustrado pela Figura 37 apresenta as etapas do treinamento do modelo seletor de subdicionários.

Para treinar o modelo de classificação de subdicionário, inicialmente devemos obter resíduos resultantes da decomposição atômica de sinais de EDA. A partir da taxa de aproximação, selecionam-se apenas átomos fortemente coerentes com o resíduo.

Para treinamento e validação do modelo, divide-se o conjunto de dados obtidos em dois conjuntos, um de treinamento e outro de teste; o que é seguido do balanceamento e normalização. Aplica-se a validação cruzada para buscar os hiperparâmetros mais adequados para o modelo. O desempenho do modelo é avaliado no conjunto de teste.

A entrada do classificador é o resíduo decimado, isso reduz o número de neurônios

de entrada da rede.

O modelo foi desenvolvido em linguagem *Python* com auxílio de bibliotecas como: *Pandas* [92], *Numpy* [93], *Scikit-Learn* [94], *Keras* [95]. A seguir, descrevemos os procedimentos adotados.

Decompomos 20 sinais divididos em blocos de $N=128$ amostras (em média, têm-se 290 blocos por sinal) com um critério de parada inicial de 32 iterações/átomos por bloco. Os blocos dos sinais contêm superposição de $N/2$ amostras, como é ilustrado pela Figura 38, resultando em conjunto de resíduos (e átomos selecionados para eles) com aproximadamente 185.600 amostras. Esse é o conjunto de dados usados para treinamento e teste do modelo.

Um fator que pode afetar a capacidade de generalização da rede é a existência de blocos com SCRs cortados nas extremidades, enquanto em outros, os SCRs estão centralizados. Os SCRs nas extremidades não são, à princípio, bem modelados pelos átomos considerados e podem assim resultar em escolhas de átomos que, apesar de aproximarem o sinal, não representam bem os fenômenos nele presentes. A superposição entre blocos permite mitigar esse efeito no treinamento do modelo.

Pode-se treinar e avaliar o modelo classificador formando duas configurações de testes. A primeira treina o modelo usando todas as amostras do conjunto de dados, i.e., com átomos empregados nas decomposições de todos os blocos, incluindo os superpostos. Na segunda, empregam-se somente os blocos que possuem a maior parte da energia dos SCRs centralizadas na janela. Para obter somente os blocos com SCRs centralizados, desconsideram-se blocos cujas amostras no início ou no fim da janela correspondam a pelo menos 50% do valor máximo das amostras do bloco. Comparamos os desempenhos dos classificadores obtidos com as duas configurações.

3.2.1 Critério para Seleção de Átomos Coerentes

Outro fator que pode ser relevante para generalização do modelo pretendido é que, durante a decomposição do sinal por meio do MP, o resíduo se comporta como ruído branco gaussiano à medida que o número da iteração aumenta [9, 66]. Um resíduo com energia pouco concentrada no tempo e na frequência torna mais difícil para o modelo selecionar o subdicionário do qual escolhe-se o átomo para aproximá-lo. Uma maneira de lidar com este problema é treinar a rede com átomos coerentes, ou seja, átomos altamente

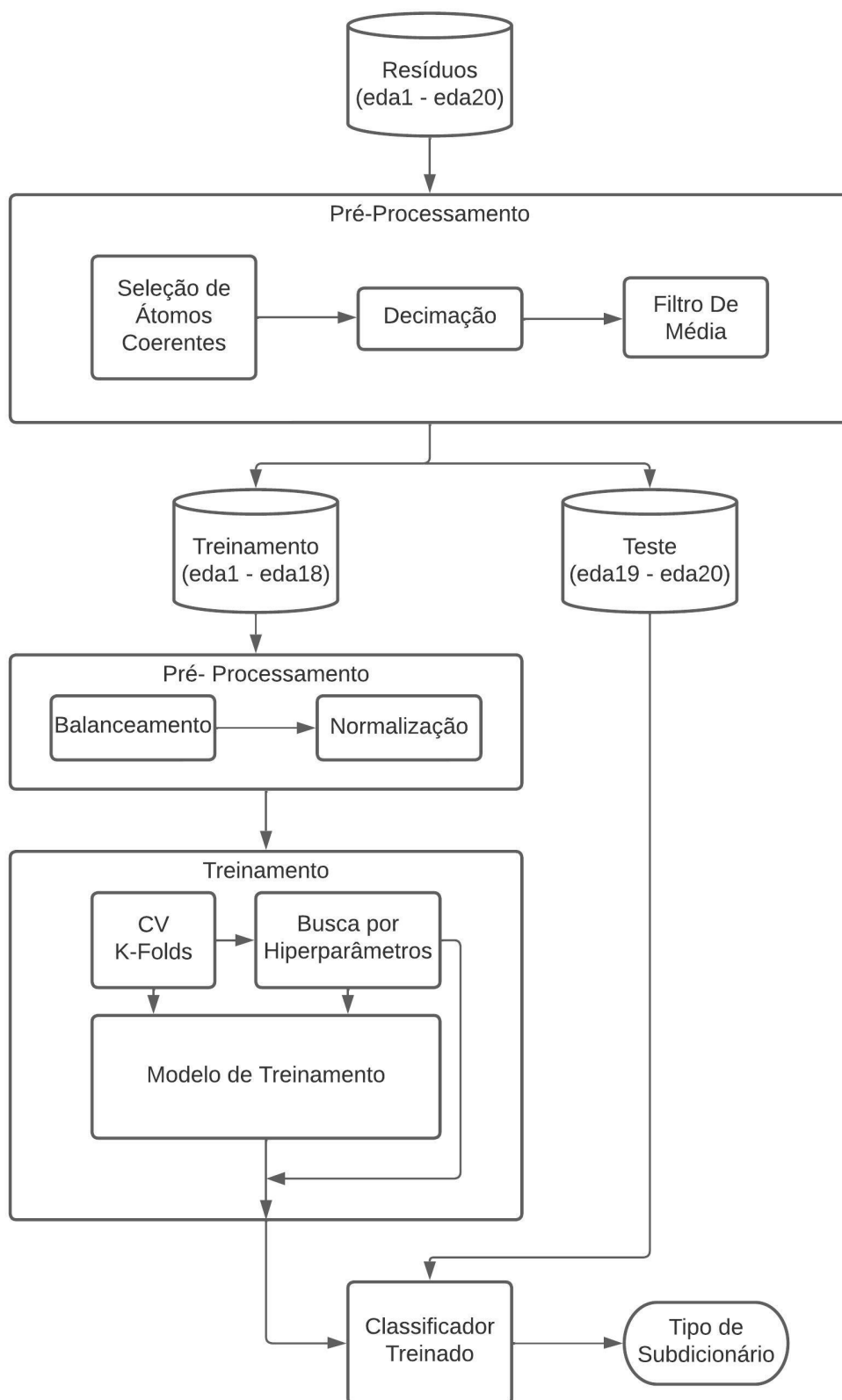


Figura 37: Diagrama de blocos do processo de obtenção do conjunto de dados e do treinamento do modelo.

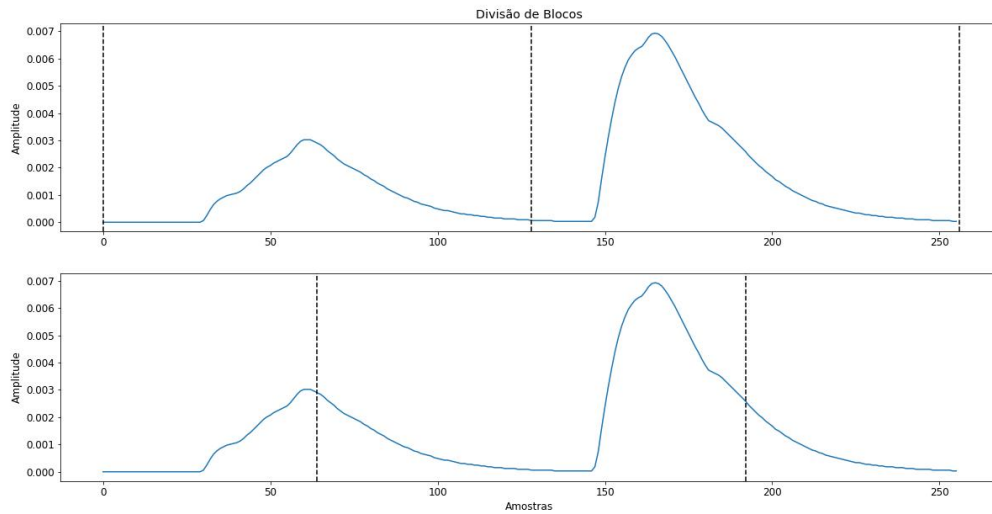


Figura 38: Exemplo da divisão de blocos de 128 (N) amostras usando saltos de 64 ($N/2$) amostras.

correlacionados com os sinais, que a priori, modelam bem as características intrínsecas do sinal [6]. Para isso, usa-se a taxa de aproximação do resíduo na iteração, com base na qual selecionam-se os átomos coerentes.

Empregamos como critério de seleção dos átomos coerentes um percentil dos átomos usados nas representações. Obtemos a função distribuição cumulativa (CDF) da taxa de aproximação considerando os sinais decompostos, conforme ilustrada na Figura 39. A partir dessa CDF, podemos definir um conjunto de resíduos para formar o conjunto de dados usado para obter o modelo. Definimos o conjunto de dados a partir de um percentil da CDF. A Tabela 5 ilustra os limiares de percentis testados e a cardinalidade do conjunto de dados resultante para a obtenção do modelo.

Na Tabela 5, é possível ver que, conforme esperado, ao aumentar-se o percentil, a quantidade de dados disponível reduz. Quanto maior o conjunto de dados disponível, a princípio, maior é a abrangência do que a rede neural pode aprender, reduzindo a possibilidade de *overfitting*.

Quanto maior a cardinalidade do conjunto de dados empregado, maior é a sua representatividade e complexidade, e o modelo pode, a princípio, aprender melhor. Por outro lado, um conjunto de treinamento maior pode requerer mais camadas e neurônios (maior complexidade [80]) para ser capaz de capturar todos os padrões nele presentes. Por

outro lado, se o conjunto de dados é relativamente pequeno, espera-se que ele seja também relativamente simples e uma rede neural com muitas camadas e neurônios possa levar ao *overfitting*, quando a rede neural se ajusta muito bem aos dados de treinamento, mas não é capaz de generalizar para entradas desconhecidas. Assim, espera-se que o ajuste ou o critério de seleção do conjunto de entrada impacte no êxito do modelo.

Uma vez selecionado o conjunto de dados usado no projeto do modelo, temos uma etapa de pré-processamento das amostras. Essa etapa visa melhorar a qualidade dos dados empregados no projeto do modelo.

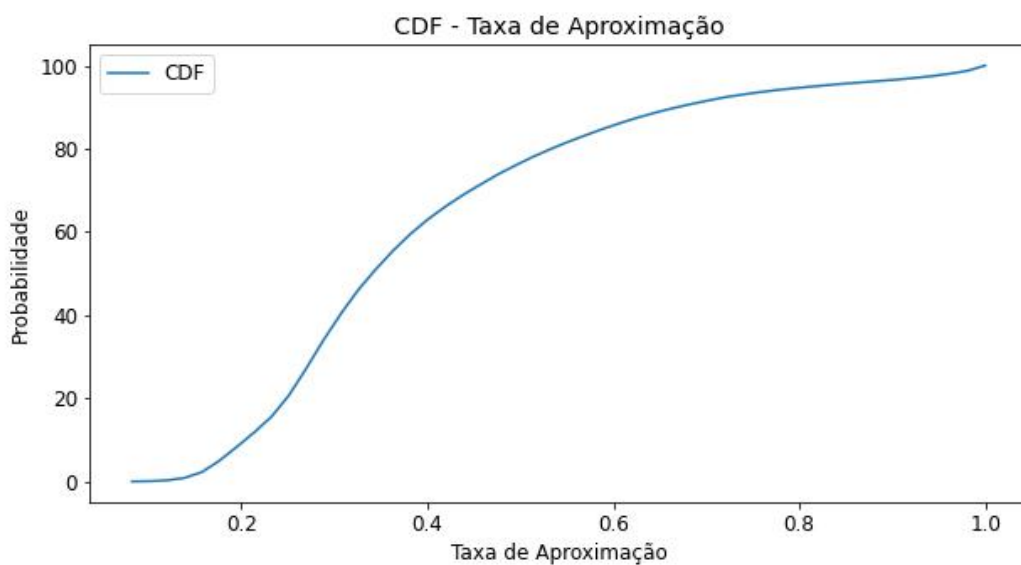


Figura 39: CDF da taxa de aproximação.

Tabela 5: Cardinalidade dos dados de entrada para diferentes limites de taxas de aproximação.

CDF	Cardinalidade		Lim. App Ratio
	Centralizado	Todos os Blocos	
70%	14892	41254	0,44
90%	4683	13770	0,66
95%	2416	6802	0,81
97%	1868	4490	0,9

3.2.2 Dados na Entrada da Rede: Resíduos Decimados

O modelo pretendido deve selecionar um subdicionário no qual a iteração MP deve procurar pelo átomo para decompor o resíduo. As funções protótipo que definem cada

subdicionário são suaves. Assim, numa tentativa de redução do tamanho da entrada do modelo e da complexidade do mesmo, reduzimos a dimensão do resíduo apresentado à rede, decimando-o.

A decimação consiste em sub-amostrar um sinal, eliminando $M - 1$ amostras intercaladas a cada M amostras do sinal original – M é o fator de decimação. Isso requer aplicar antes da decimação um filtro *anti-aliasing*, já que a sub-amostragem reduz a taxa de amostragem em M vezes, e, conseqüentemente, a frequência de Nyquist também se torna M vezes menor [96].

Empregamos um fator de decimação $M = 2$ e, conseqüentemente, a quantidade de entradas da RNA, recebendo os sinais “encurtados” conforme exemplificado na Figura 40. É possível observar que mesmo com metade das amostras do resíduo original não percebemos nenhuma descaracterização do sinal.

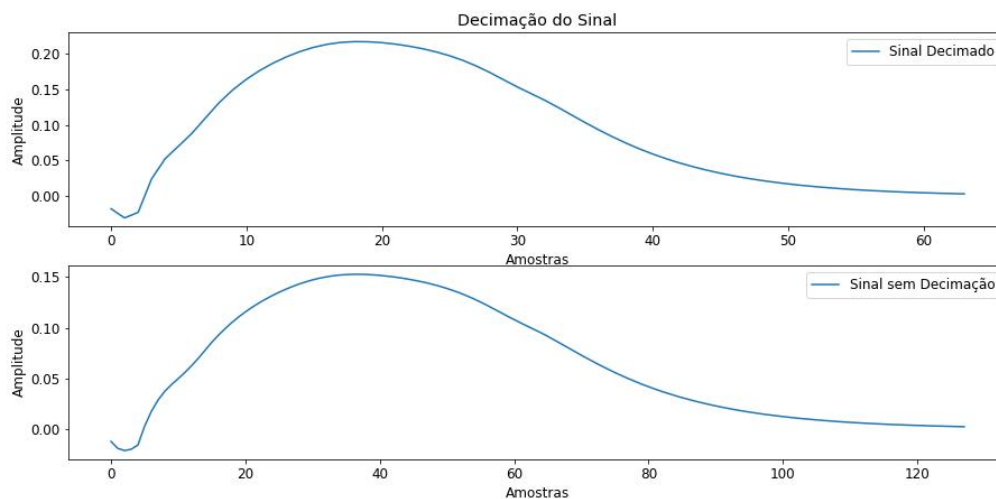


Figura 40: Decimação do resíduo por um fator 2.

Se considerarmos a quantidade de multiplicações para avaliar a complexidade, a complexidade de um classificador RNA é aproximadamente $\frac{N}{M}N_h + N_hN_o + N_oN_d$, onde N/M é a quantidade de entradas da rede (o comprimento do sinal N dividido pelo fator de decimação M), N_h e N_o são os números de neurônios das camadas oculta e de saída, respectivamente, e N_d é o número de categorias na saída do classificador (a quantidade de subdicionários).

3.2.3 Balanceamento do Conjunto de Dados

Definido o conjunto de dados empregado X com L elementos de comprimento 64 e os tipos de átomos escolhidos pelo MP para cada uma das L instâncias, temos que dividir o conjunto entre treinamento e teste. Essa divisão deve levar em conta os sinais e blocos de onde vêm os resíduos. Caso haja resíduos de um mesmo bloco (de um mesmo sinal) tanto no conjunto de treinamento como no de teste, mascara-se o desempenho da rede.

Assim, separam-se os conjuntos de forma que contenham instâncias produzidas por sinais diferentes. A metodologia aplicada consiste em, a partir dos 20 sinais disponíveis (nomeados de *eda1* a *eda20*), reservar 2 deles (*eda19* e *eda20*), cerca de 10%, para os testes e os outros 18 sinais (*eda1* - *eda18*) são utilizados para treinamento da rede. Quando definimos o subconjunto dos dados usado para validação durante o treinamento, toma-se o mesmo cuidado, separando-se para a validação resíduos advindos da decomposição de sinais não usados no ajuste do pesos da rede (treinamento).

Após a divisão dos dados, é preciso avaliar se temos um conjunto razoavelmente balanceado, isto é, cujas saídas alvo (classes) ocorrem em quantidades similares. Um conjunto de dados desbalanceado pode levar a um viés na saída da rede neural, fazendo com que ela se concentre mais nas classes que apresentam um número maior de amostras, em detrimento das classes que apresentam um número menor de amostras. Como resultado, a rede pode ter um desempenho inferior em prever as classes menos representadas.

O balanceamento de dados visa equalizar a distribuição de amostras em cada classe e mitigar os efeitos acima discutidos. Ao balancearem-se os dados, estamos garantindo que a rede neural terá acesso a um número semelhante de amostras em cada classe. Entre as técnicas mais usadas para balanceamento de dados temos o *oversampling* - aumenta-se o número de amostras das classes menos representadas, e o *undersampling* - diminui-se o número de amostras das classes mais representadas, ou uma combinação de ambas [97]. A Figura 41 e a Figura 42 apresentam as distribuições das classes para os conjuntos de dados resultantes dos diferentes percentis de resíduos e com a filtragem de blocos com SCRs no centro (Figura 41) e todos os blocos (Figura 42). Abaixo de cada gráfico, temos o valor da taxa de aproximação (que resulta do percentil escolhido) aplicada na seleção.

Para a configuração de blocos com SCRs centralizados, Figura 41, pode-se observar que há uma troca da classe majoritária, conforme aumenta-se o percentil que define os átomos que compõem o dicionário (equivalentemente, a taxa de aproximação λ). Há

casos em que uma classe é majoritária, Gabor para $\lambda = 0,44$, e torna-se a classe minoritária, Gabor para $\lambda = 0,81$. A queda na ocorrência de átomos de Gabor que levou a troca de classe condiz com a análise feita no capítulo 2, pelo fato do algoritmo *Matching Pursuit* escolher átomos com características de minimização de ruído, no qual prevalece o subdicionário de Gabor. Conforme aumentamos o limite do critério de parada, há uma tendência maior da razão de átomos mais correlacionados com o sinal.

Por outro lado, quando não restringimos aos blocos que contêm SCRs centralizados, vide Figura 42, a classe majoritária é sempre o dicionário de Gabor; Entretanto, é possível observar que o conjunto de dados é razoavelmente balanceado para $\lambda = 0,81$ e $\lambda = 0,9$, neste caso.

Para evitar que o desbalanceamento do conjunto de dados produza um viés na saída do classificador, aplicamos o *undersampling* para balancear as classes de tal forma que todas apresentem as mesmas quantidades de ocorrências que as da minoritária. Dessa forma, dispensam-se amostras de outras classes (que não a minoritária) até que elas tenham a mesma quantidade de ocorrências que a classe minoritária. A desvantagem do balanceamento por *undersampling* é a perda de informações importantes para o treinamento da rede, o que pode comprometer a capacidade de sua generalização, mas em compensação evitamos o viés [98]. Contudo, este problema pode ser minimizado de maneira que as amostras das classes majoritárias a serem descartadas é dada pelo valor da taxa de aproximação dos resíduos. Desta forma, permanecemos com os resíduos mais correlacionados com o sinal e descartamos resíduos com características ruidosas. Dessa maneira, as amostras com menores λ serão descartadas primeiro até que o número de amostras das diferentes classes majoritárias seja igual ao da classe minoritária. Devido ao *undersampling* a cardinalidade do conjunto de dados é reduzida, a Tabela 6 apresenta as cardinalidades resultantes, que vemos ser menores que as do conjuntos de dados obtidos somente pela aplicação do limiar, Tabela 5.

Tabela 6: Cardinalidade do conjunto de dados com a aplicação do balanceamento por *Undersampling* para diferentes percentis de resíduos aplicados.

CDF	Cardinalidade		Lim. App Ratio
	Centralizado	Todos os Blocos	
70%	5097	12504	0,44
90%	2997	82117	0,66
95%	2148	5997	0,81
97%	1341	3957	0,9

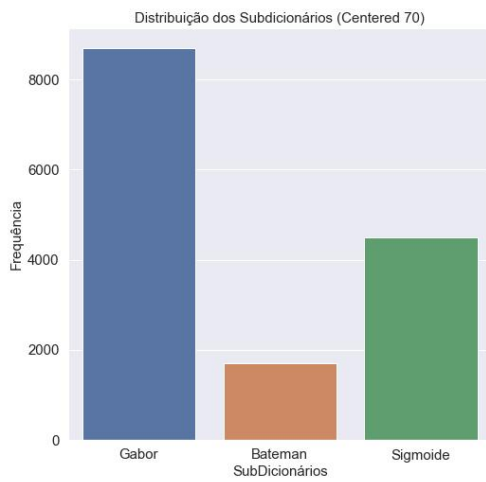
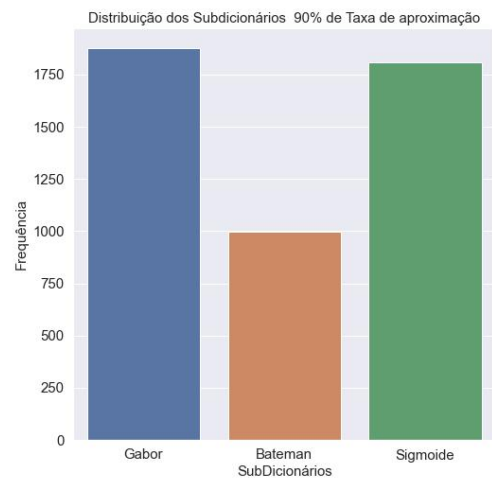
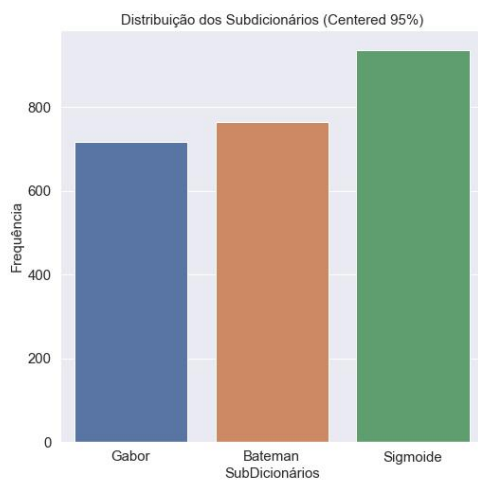
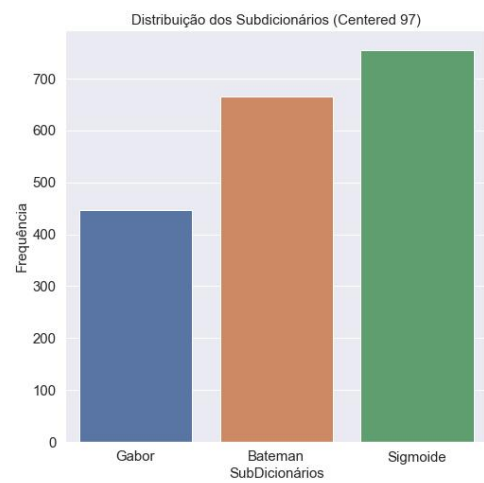
(a) $\lambda = 0,44$ (b) $\lambda = 0,66$ (c) $\lambda = 0,81$ (d) $\lambda = 0,9$

Figura 41: Histogramas das classes presentes nos subdicionários para blocos com SCRs centralizados.

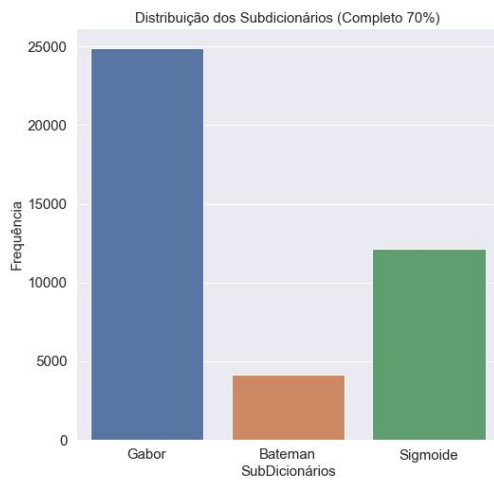
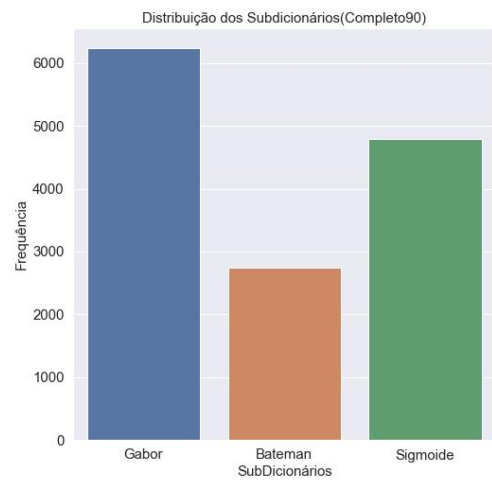
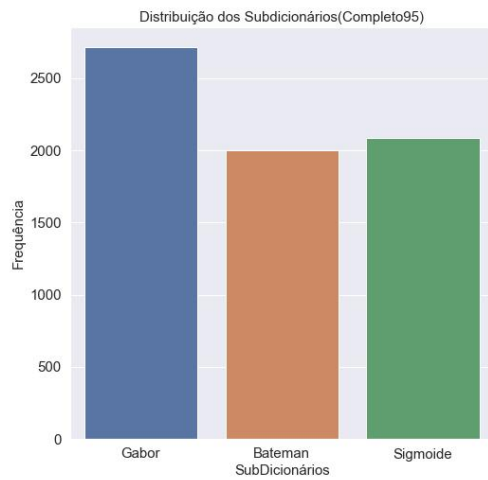
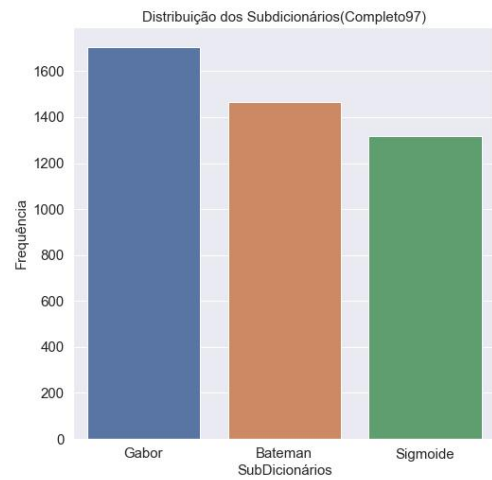
(a) $\lambda = 0,44$ (b) $\lambda = 0,66$ (c) $\lambda = 0,81$ (d) $\lambda = 0,9$

Figura 42: Histogramas das classes presentes nos subdicionários para blocos sem restrição de SCRs centralizados.

3.2.4 Normalização dos Dados

As variáveis envolvidas no problema, e portanto, constantes do conjunto de dados, podem apresentar escalas diferentes. Se não for tomado o correto cuidado, elas podem não contribuir igualmente para o ajuste do modelo, pois suas faixas de valores distintas podem afetar o treinamento. Assim, como uma heurística para evitar esse problema potencial, aplica-se a normalização visando com que as diferentes variáveis do problema apresentem faixas de valores congruentes.

É comum aplicar a padronização (ou normalização) dos recursos, para que tenham médias nulas ($\mu = 0$) e variâncias unitárias ($\sigma^2 = 1$), antes do ajuste do modelo. Aplica-se na entrada Z_i da RNA, a versão normalizada de X_i , via

$$Z_i = \frac{X_i - \mu_X}{\sigma_X} \quad (98)$$

onde μ_X é a média e σ_X o desvio padrão da variável aleatória X_i .

A normalização torna o treinamento menos sensível à escala dos recursos, permitindo ajustar melhor os coeficientes do modelo. Espera-se assim que as atualizações dos pesos apresentem faixas (e taxas) semelhantes, durante o processo de ajuste (*backpropagation* via Adam). Isso, a princípio, facilita o ajuste e a convergência do modelo.

3.3 Opções e Critérios Aplicados no Projeto do Classificador

Neste trabalho, é usado um modelo de rede neural com uma camada escondida para o problema de classificação multi-classes que consiste em indicar o subdicionário mais adequado sobre o qual aplicar o MP para decompor resíduos MP de sinais EDA.

A quantidade de neurônios na camada oculta é um dos hiperparâmetros da rede a ser definido. Esse hiperparâmetro e os demais, como o tamanho do lote, a taxa de aprendizado do otimizador e a taxa de *dropout*, são definidos através de busca Bayesiana. Durante essa busca, aplica-se a técnica de validação cruzada.

Uma vez encontrados os melhores hiperparâmetros, o modelo é retreinado e testado para avaliar seu desempenho e capacidade de generalização. Nesta seção, revisitam-se algumas das técnicas acima mencionadas.

3.3.1 Arquitetura da Rede

A camada de entrada da RNA recebe o resíduo decimado e contém 64 neurônios. A de saída contém tantos neurônios quanto classes, e, no caso em tela, ela tem 3 neurônios já que as saídas possíveis são os tipos de subdicionários (Gabor, Bateman e Sigmoide-Exponencial) usados na decomposição atômica. O número de neurônios presentes na camada escondida da rede é definido pela busca de hiperparâmetros assim como a taxa de *dropout* aplicada a ela.

A função de ativação dos neurônios de saída é a SoftMax e nos demais neurônios, a função de ativação utilizada é a ReLu. O otimizador da descida do gradiente utilizado é o Adam com a função custo sendo a entropia cruzada.

3.3.2 Busca por Hiperparâmetros

Hiperparâmetros são valores definidos que determinam o processo de aprendizagem – o ajuste dos parâmetros da rede, seus pesos. O número de camadas escondidas, o número de neurônios por camada e as funções de ativação neles empregadas definem a arquitetura da RNA. Vale ressaltar que uma quantidade baixa de neurônios leva a baixa capacidade preditiva (*underfitting*) e muitos neurônios levam a alto custo computacional e uma possível perda da capacidade de generalização (*overfitting*). Assim, as ocorrências de *underfitting* e *overfitting* devem ser controladas não só com o uso de um banco de dados adequado no processo de aprendizado, mas também pela arquitetura da rede, a fim de que se tenha uma boa capacidade de generalização e classificação [99]. Por outro lado, a função de ativação introduz a não linearidade que permite que a rede implemente mapeamentos não-lineares entre entrada e saída.

Numa RNA, além da definição da estrutura da rede (neste trabalho, a quantidade de neurônios na camada escondida e as funções de ativação) temos hiperparâmetros que definem o cômputo de atualizações e suas aplicações aos pesos da rede. Suas escolhas podem impactar o desempenho do modelo. O número de épocas refere-se a quantas vezes o algoritmo itera por todo o conjunto de treinamento. Com mais épocas, há mais oportunidades de aprendizado, mas muitas podem acarretar sobre especialização. O tamanho do lote (*batch size*) define quantos exemplos são empregados por iteração SGD. Lotes pequenos fazem os pesos oscilarem muito e podem tornar o aprendizado muito lento, lotes

grandes podem mascarar padrões menos frequentes nos dados. A taxa de aprendizado controla o quanto os pesos são alterados em cada *batch* do processo de treinamento. A taxa de *dropout* induz o percentual de neurônios que são abandonados a cada atualização de uma camada durante o treinamento com uma “visualização” diferente da camada configurada. Uma baixa taxa de *dropout* requer uma quantidade grande de neurônios na camada oculta que desacelera o treinamento e leva ao *underfitting*. Já para uma taxa de *dropout* elevada pode não ser suficiente para evitar o *overfitting* [90].

Encontrar o conjunto de hiperparâmetros ótimos - aquele que leva ao melhor desempenho do modelo - é um desafio comum no campo de aprendizado de máquina, pois os hiperparâmetros afetam significativamente o desempenho e a generalização do modelo. O ajuste manual desses hiperparâmetros pode consumir muito tempo e requer conhecimento e experiência especializados. Uma solução promissora para este problema é o uso de técnicas de busca por hiperparâmetros. Uma abordagem popular é procurar hiperparâmetros usando o algoritmo conhecido como *Bayes Search*, a busca Bayesiana [100]. Essa técnica de otimização probabilística usa uma combinação de métodos estatísticos e modelos matemáticos para selecionar automaticamente os melhores hiperparâmetros para o modelo.

A busca Bayesiana é particularmente útil quando o espaço de hiperparâmetros é grande e complexo e pode levar a melhorias significativas no desempenho e generalização do modelo [100]. O processo de busca Bayesiana envolve a construção de um modelo probabilístico para descrever a relação entre hiperparâmetros e o desempenho do modelo que é atualizado a cada iteração (treinamento do modelo classificador) com base nos resultados da avaliação do desempenho do modelo classificador para um ponto do vetor (conjunto) de hiperparâmetros. O objetivo é encontrar um conjunto de hiperparâmetros que maximize o desempenho do modelo classificador e minimize o número de avaliações necessárias. Uma das vantagens do *Bayes Search* comparado aos outros métodos como *Grid Search* (discretiza-se o espaço de hiperparâmetros e testam-se os pontos resultantes) e *Random Search* (busca aleatória sem nenhum ordenamento ou discretização pré-definida) é sua capacidade de equilibrar a exploração do espaço de hiperparâmetros com a exploração de regiões promissoras.

Na busca Bayesiana, emprega-se um modelo probabilístico dos hiperparâmetros, a partir do qual seleciona-se a próxima configuração de hiperparâmetros (valor do vetor)

a ser testada, aquela que pelo modelo corrente acarretaria em um bom desempenho do modelo. Tal modelo usa informações sobre o desempenho do modelo classificador em avaliações anteriores para atualizar a distribuição de probabilidade dos hiperparâmetros. A cada novo ponto testado, esse modelo probabilístico é atualizado. Esse método já se mostrou eficaz, com ganhos de mais de 100 vezes na eficiência do processo de seleção de hiperparâmetros em comparação com o *Random Search* [101]. Em resumo, a busca por hiperparâmetros usando *Bayes Search* é uma técnica promissora para otimizar automaticamente hiperparâmetros de modelos de aprendizado de máquina.

Neste trabalho, aplicamos a busca Bayesiana para otimizar os hiperparâmetros do modelo. A Tabela 7 apresenta os intervalos do espaço amostral dos hiperparâmetros. Por outro lado, aplica-se *early stopping*, considerando até 2000 épocas como limite máximo e uma espera de 200 épocas para evitar a parada precoce.

Tabela 7: Intervalos para Bayes Search.

Hiperparâmetros	Intervalo
Neurônios	[3 -100] (Inteiros)
Dropout	[0,1 - 0,3] (Real)
Taxa de Aprendizado	[0,001 - 0,1] (Real, em escala log)
Batch Size (tamanho do lote)	[30 - 110] (Inteiro)

3.3.3 Validação Cruzada

A validação cruzada (*Cross Validation*) é uma técnica usada para avaliar a generalização de um modelo de aprendizado de máquina [80]. Isso é particularmente útil quando o conjunto de dados é limitado e se deseja maximizar o número de amostras empregadas no treinamento e na validação do modelo. Ela fornece uma boa estimativa do erro de generalização do classificador ao dividir o conjunto de dados em partições e aplicar o processo de treinamento-validação sobre diferentes arranjos dessas partições [102].

O modelo é então treinado nos dados de treinamento e avaliado no conjunto de validação. A técnica de validação cruzada consiste em dividir o conjunto de treinamento em k partes iguais (*k-fold*), uma parte é usada como conjunto de validação, enquanto as outras $k-1$ partes são usadas para treinar o modelo. Este processo é repetido k vezes, alterando a partição usada como conjunto de validação [103]. Após o *k-fold*, dentro da busca bayesiana, temos os melhores parâmetros. Então treina-se novamente com o

conjunto de treinamento completo e só então se utiliza o conjunto de teste. Para avaliar o modelo, obtém-se a média dos desempenhos dos k modelos resultantes das k repetições do processo. O uso da validação cruzada confere maior confiabilidade à avaliação de desempenho, pois testa-se o modelo em diferentes conjuntos de treinamento e validação, reduzindo assim a possibilidade de *overfitting*. Além disso, como utilizam-se todos os dados disponíveis para treinamento e validação, maximiza-se seu uso na construção do modelo.

Uma das desvantagens da validação cruzada é a sensibilidade na forma em que os dados são divididos em k partições, o que pode levar a resultados inconsistentes se as distribuições de classes forem diferentes em cada partição. Como já mencionado, os dados de entrada da rede são resíduos de decomposições MP e em resíduos decorrentes da decomposição de um mesmo bloco espera-se uma razoável correlação. Portanto, para evitar que haja um viés, amostras de um mesmo bloco ou até de um mesmo sinal não podem estar presentes ao mesmo tempo no conjunto de treinamento e no de validação.

Na prática, como o conjunto de dados consiste em 18 sinais de EDA é possível dividir esse conjunto de dados em 6 *folds* (partições) com os resíduos de três sinais em cada; assim, alternando os *folds*, têm-se os sinais advindos de 3 sujeitos diferentes no conjunto de validação e o complemento no de treinamento. A Figura 43 ilustra esse arranjo para o caso sem distinção de centralização de SCRs usando $\lambda = 0,81$. As primeiras seis linhas indicam a divisão do conjunto de dados em treinamento e validação por pacientes em cada *fold* (partição). É possível observar que a divisão do conjunto de validação está alinhada com as duas linhas subsequentes, nas quais indicam os subdicionários empregados nas decomposições de cada um dos pacientes, cujas quantidades de iterações (e de resíduos) são indicadas pelo eixo horizontal da Figura 43. A Figura 44 oferece uma ampliação da distribuição dos subdicionários escolhidos na decomposição para cada sinal/paciente para facilitar a compreensão.

3.3.4 Critérios de Desempenho

O desempenho do modelo classificador é avaliado considerando o erro e a acurácia. Outras métricas como a sensibilidade, precisão e *F1-score* são muito utilizadas também, principalmente quando há um desbalanceamento do conjunto de dados e quando há preocupação com a relação entre os casos de falso positivo e falso negativo [104]. Como os

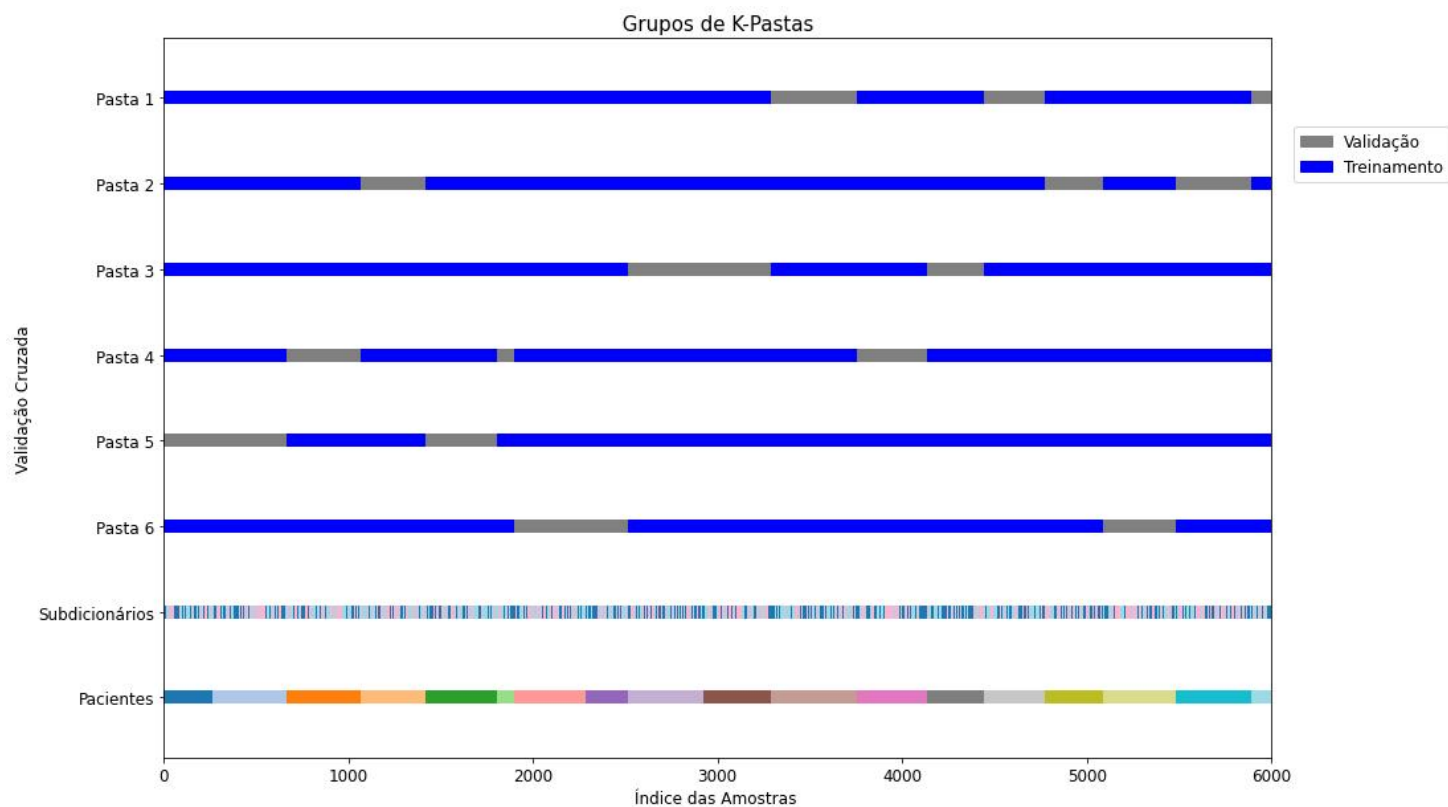


Figura 43: Ilustração da configuração da técnica k -folds.

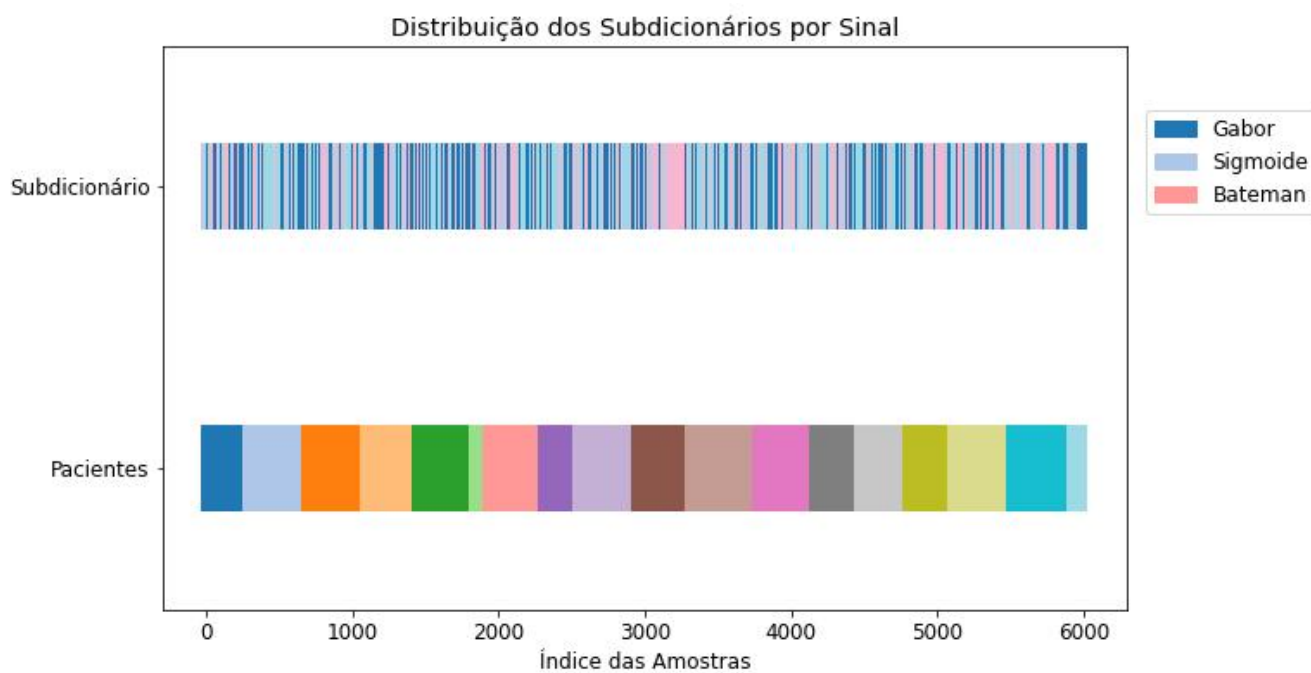


Figura 44: Subdicionários empregados nas decomposições MP de sinais EDA de diferentes pacientes.

dados passaram por um processo de balanceamento, e como para a aplicação desejada não cabe uma diferenciação entre falsos positivos e negativos, não há a necessidade de calculá-las. Repara-se que, devido à estratégia de decomposição MP, mesmo se um subdicionário for erroneamente escolhido, ainda assim, teremos uma redução do resíduo, mesmo que sub-ótima na iteração do MP. Trataremos deste ponto com mais detalhes na Seção 3.5.

Uma ferramenta útil para avaliar o desempenho de um modelo de classificação e entender quais tipos de erros este está cometendo é a matriz de confusão ilustrada na Figura 45. Na qual VP é o número de casos de verdadeiros positivos (acertos da classe, isto é, do subdicionário escolhido), VN é o número verdadeiros negativos (a classe é indicada como inadequada quando o é), FP é o número de falsos positivos (a classe é indicada pela rede como adequada quando não o é), e FN é o número de falsos negativos (a classe é adequada, mas indicada pelo classificador como não o sendo). Com base nela, é possível avaliar o desempenho do modelo e observar os erros da classificação.

		Valor Previsto	
		positivo	negativo
Valor Real	positivo	VP	FN
	negativo	FP	VN

Figura 45: Matriz de confusão para um classificador binário (de duas classes)

A acurácia é definida como a proporção de amostras classificadas corretamente por classe pelo modelo em relação ao número total de amostras de cada classe. Em outras palavras, é a taxa de acerto do modelo e pode ser definida como

$$\text{Acc} = \frac{VP_{classe} + VN_{classe}}{VP_{classe} + FN_{classe} + VN_{classe} + FP_{classe}}. \quad (99)$$

Além da acurácia, é possível avaliar o desempenho monitorando-se a função perda. Como visto anteriormente, neste projeto, a rede tem como objetivo minimizar a entropia

cruzada, que foi definida nas Equações (91) e (92).

3.4 Projeto de Classificadores de Subdicionários para Iterações MP

Nesta seção, apresentamos os modelos e os desempenhos de classificadores de subdicionários para decomposições de resíduos em iterações MP de sinais EDA. Como visto no capítulo 2, o sinal é subdividido em blocos e dependendo dos limites da janela, é possível que informações relevantes, como o pico de um impulso SCR seja cortado por esses limites. Acredita-se que isso possa afetar na escolha dos subdicionários de maneira que casos com SCRs cortadas pelos limites dos blocos podem ter subdicionários diferentes a serem escolhidos caso a mesma SCR com energia majoritariamente estivesse dentro da janela, o que poderia ser mais complexo para o modelo diferenciar essas características. Devido a isso, avaliam-se dois casos. Em um, aplica-se a restrição de treinar o modelo somente com resíduos que possuam “SCRs centralizadas” e o caso geral, onde não há distinção entre os blocos.

Para cada um dos dois casos, que definem conjuntos de dados distintos, também são avaliados os desempenhos do modelo usando diferentes critérios de coerência entre átomos e resíduos a partir da taxa de aproximação definida pelos percentis.

Além disso, com o objetivo de identificar a influência de cada subdicionário na capacidade de generalização do modelo, avaliou-se além da classificação ternária, a binária em pares de subdicionários (Bateman/Gabor, Bateman/Sigmoide-Exponencial, Gabor/Sigmoide-Exponencial).

Por fim, seleciona-se o modelo mais adequado a ser usado como seletor de subdicionários na etapa de decomposição e compara-se o desempenho da decomposição atômica com e sem o modelo classificador, na Seção 3.5.

O Apêndice A apresenta as curvas de aprendizado de cada classificador projetado. A seguir, discutem-se os resultados relevantes.

3.4.1 Caso de Blocos com SCRs Centralizados

Considerando que os blocos sejam tais que não contenham SCRs nas extremidades, treinamos o modelo classificador para selecionar dentre os 3 subdicionários aquele que deveria ser aplicado para decompor o resíduo. Isso é feito para os diferentes percentis

de seleção (que resultam em diferentes taxas de aproximação) para a seleção dos átomos coerentes.

Para observar com mais detalhe os problemas enfrentados pelo classificador ternário, testam-se ainda os classificadores binários para cada par de dicionário. Projetam-se assim quatro classificadores (um ternário e três binários).

3.4.1.1 Classificação Multiclasses

A configuração dos hiperparâmetros para o caso de classificação multiclasses usando os três subdicionários, escolhidas pela busca Bayesiana com parada antecipada em número de épocas, são apresentadas na Tabela 8. Com a rede treinada, avaliamos seu desempenho através da acurácia apresentadas na Tabela 9 e matrizes de confusão ilustradas pela Figura 46.

Tabela 8: Melhores hiperparâmetros do treinamento usando a decomposição de blocos com SCRs centralizados para o classificador que considera os três subdicionários: Bateman, Gabor e Sigmóide.

Subdics.	Hiperparâmetros						
	λ	CDF	Neurônios	Dropout	Tx. Aprend.	Batch size	Épocas
Gabor	0,44	70%	100	0,2	0,002	30	211
Bateman	0,66	90%	100	0,1	0,0025	110	234
Sigmoide	0,81	95%	88	0,1	0,002	110	202
	0,9	97%	100	0,1	0,003	110	666

Tabela 9: Acurácia do seletor de subdicionário ternário quando treinado com blocos contendo SCRs centralizados.

Subdics.	Acurácia				
	λ	CDF	Treino	Validação	Teste
Gabor Bateman Sigmoide	0,44	70%	81,80%	63,00%	54%
	0,66	90%	84,30%	70,90%	61,60%
	0,81	95%	85,80%	78,60%	67,00%
	0,9	97%	91,50%	77,20%	71,40%

Os resultados na Tabela 9 indicam que para $\lambda = 0,44$ o modelo não obteve uma boa generalização; obtendo a pior acurácia no treinamento e uma acurácia de 54% no teste. Pela matriz de confusão na Figura 46, podemos ver que há uma maior incidência de erros quando o subdicionário alvo é o de Gabor, foi previsto o dicionário de Bateman. Isto é, apesar de para $\lambda = 0,44$ ter-se o maior conjunto de dados, o modelo não generaliza bem a partir deles.

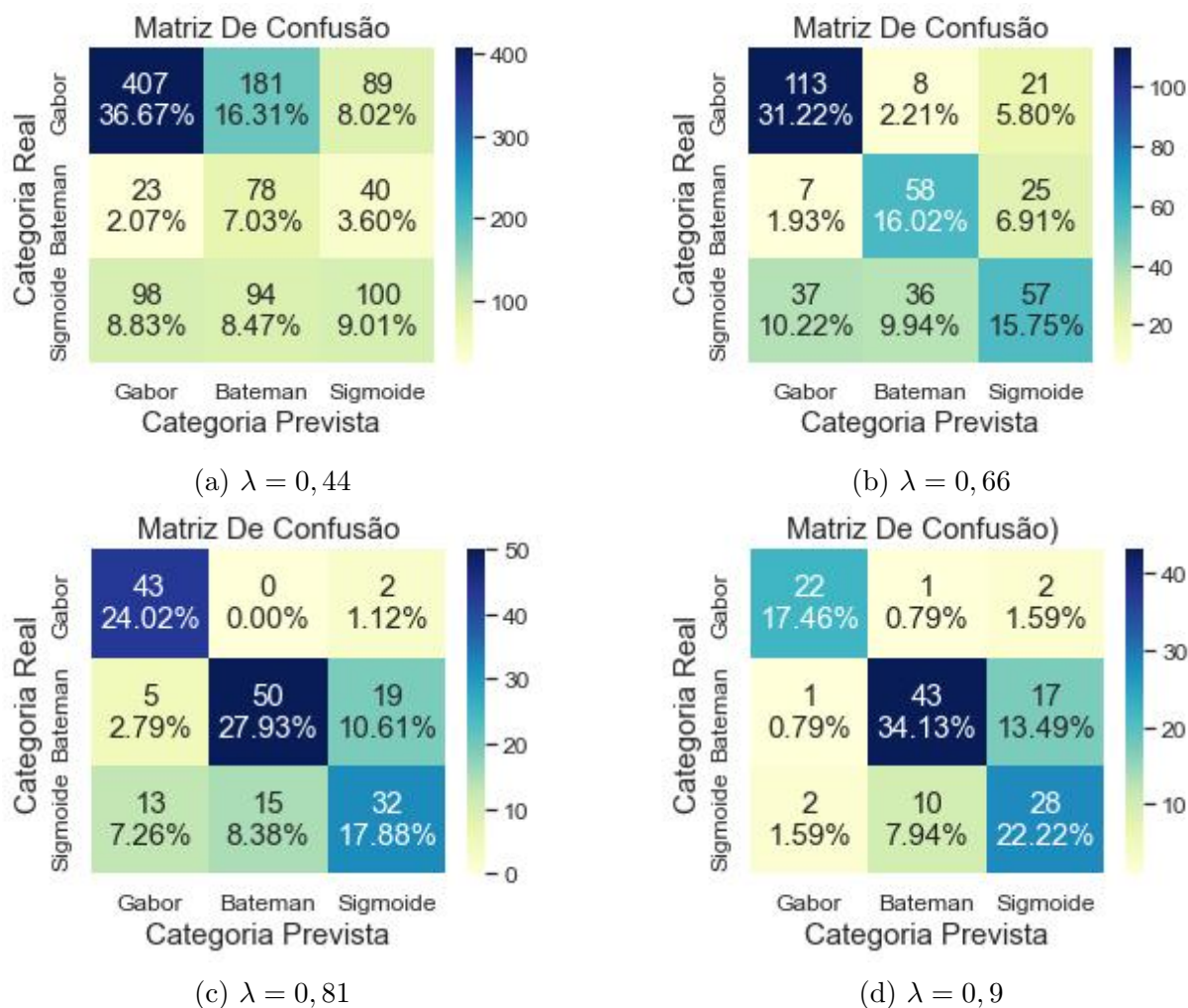


Figura 46: Matrizes de confusão para o classificador dentre três subdicionários na decomposição MP de blocos com SCRs centralizados.

Conforme o limite da taxa de aproximação aumenta, usam-se menos átomos, porém mais coerentes com os resíduos, e vemos dos resultados na Tabela 9 que a acurácia do modelo também aumenta, atingindo o máximo de 71,4% quando $\lambda = 0,9$. Isto é, quando os átomos estão mais fortemente correlacionados com as características intrínsecas do sinal, melhor é a capacidade do modelo para reconhecer padrões. Ainda analisando a matriz de confusão na Figura 46, podemos observar para os modelos com melhor desempenho ($\lambda = 0,81$ e $\lambda = 0,9$) que os erros de classificação concentram-se no discernimento entre os protótipos dos subdicionários de Bateman e Sigmoide-Exponencial. Assim, partimos a seguir para a análise da classificação binária OaO (*One-against-One*) [105] entre os subdicionários separadamente visando entender a influência de cada classe na classificação multiclases.

3.4.1.2 Classificação Binária

Treinamos modelos classificadores considerando apenas pares de subdicionários; o objetivo é comparar a capacidade de generalização da rede para cada par de subdicionários e assim avaliar se o desempenho do classificador multiclases está longe do que seria atingido em casos mais simples. Logo, considera-se um modelo de rede neural com a mesma arquitetura obtida pelo de três classes, mas com entradas e saídas limitadas, contendo somente pares de classes: Bateman/Gabor, Bateman/Sigmoide-Exponencial e Gabor/Sigmoide-Exponencial. Usamos os conjuntos de dados gerados ao aplicar o critério de parada da decomposição com a taxa de aproximação $\lambda = 0,81$ (95% da CDF). A Tabela 10 apresenta a configuração dos hiperparâmetros escolhidos pela busca Bayesiana para os classificadores binários.

Tabela 10: Melhores hiperparâmetros do treinamento usando a decomposição de blocos com SCRs centralizados para os classificadores binários.

Subdics.	Hiperparâmetros						
	λ	CDF	Neurônios	Dropout	Tx. Aprend.	Batch size	Épocas
Gabor/Bateman	0,81	95%	83	0,1	0,002	30	541
Bateman/Sigmoide	0,81	95%	100	0,1	0,001	30	310
Gabor/Sigmoide	0,81	95%	108	0,3	0,002	500	400

A Tabela 11 apresenta a acurácia de treino, validação e teste, enquanto a Figura 47 apresenta a matriz de confusão para os três classificadores obtidos. É possível observar que, neste casos, os classificadores obtêm desempenho superior aos anteriores. Isto é, as redes generalizaram melhor. As redes têm melhor desempenho ao identificar pelo resíduo o

dicionário mais adequado para decompô-lo ao considerarem-se apenas dois subdicionários, em vez de três. Principalmente, quando um deles é o de Gabor. Vemos que o caso crítico ocorre quando o par considerado contém os subdicionários de Bateman e Sigmoide-Exponencial, com uma queda na acurácia que atinge em torno de 65%. A queda de acurácia para esta combinação faz sentido se observamos que os átomos de Bateman e Sigmoide-Exponencial possuem formas e características muito parecidas, dificultando que o modelo generalize, dadas as restrições de complexidade que eles apresentam e do conjunto de dados neste trabalho.

Tabela 11: Acurácias de treino, validação e teste do seletor binário de subdicionários considerando resíduos produzidos a partir de blocos com SCRs centralizados.

Subdics.	λ	CDF	Acurácia		
			Treino	Validação	Teste
Gabor/Bateman	0,81	95%	98,9%	96,3%	95,8%
Bateman/Sigmoide	0,81	95%	88,1%	75,1%	65,77%
Gabor/Sigmoide	0,81	95%	96,6%	90,3%	87,6%

A análise destes classificadores binários justifica os resultados obtidos usando os três subdicionários, quando observamos que a maior taxa de erros ocorre entre os subdicionários de Bateman e Sigmoide. Em contrapartida, tanto a combinação Gabor/Bateman e Gabor/Sigmoide-Exponencial alcançaram resultados bastante satisfatórios com acurácias razoavelmente grandes. Conclui-se que a escolha das funções que definem os subdicionários podem impactar o desempenho do seletor de subdicionário a ser empregado na iteração MP.

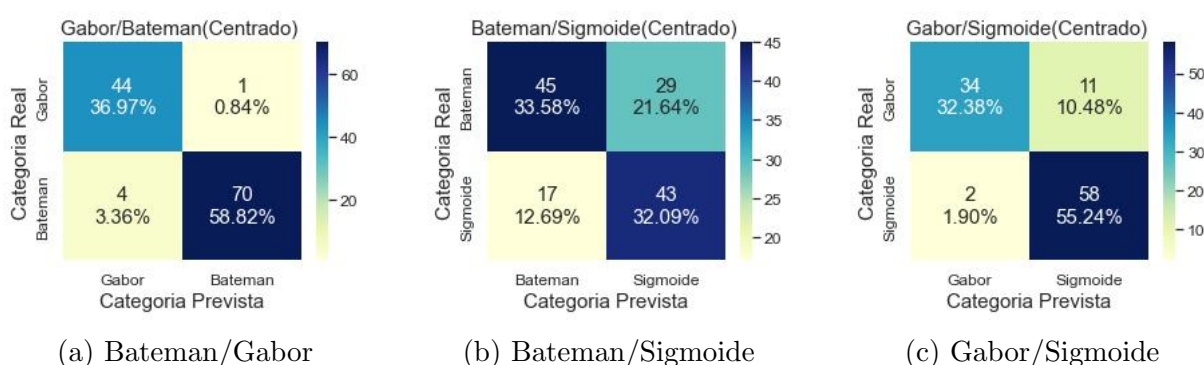


Figura 47: Matriz de confusão dos classificadores binários para pares de subdicionários considerando a decomposição MP de blocos com SCRs centralizados.

3.4.2 Caso de Blocos Sem Restrição de Centralização

Repetimos os procedimentos da subseção anterior para a base de dados mais geral, que considera todos os blocos e os átomos resultantes de suas decomposições sem fazer a distinção de que os blocos devem apresentar SCRs centralizados. Aplica-se a mesma metodologia da subseção anterior para avaliar e comparar os modelos obtidos.

3.4.2.1 Classificação Multiclasses

Para o caso de multiclasses, considerando todos os blocos, a busca Bayesiana retorna os hiperparâmetros apresentados na Tabela 12. Já a Tabela 13 contém a acurácia para treino, validação e teste para os diferentes valores de λ aplicados para determinar os resíduos incluídos no banco de dados. Os resultados são semelhantes aos obtidos quando restringe-se o banco de dados a blocos com SCRs centralizados; vemos um desempenho ruim para $\lambda = 0,44$ e $\lambda = 0,66$, com acurácias de 41% e 63% no conjunto de teste respectivamente. O modelo começa apresentar um melhor desempenho para $\lambda = 0,81$ e $\lambda = 0,9$ com acurácias de 72,8% e 76,9% no conjunto de teste, respectivamente. Quando comparados com o primeiro caso, podemos observar que há uma diferença em torno de 5 a 6 pontos percentuais a mais de acurácia para o caso geral considerando todos os blocos, sem a distinção de SCRs centralizados.

Há algumas diferenças de desempenho dos modelos obtidos com e sem a restrição de SCRs centralizados. Entretanto, podemos afirmar que em ambos os casos a rede foi capaz de generalizar. O caso particular (que se restringe a blocos com SCRs centralizados) emprega um subconjunto dos dados usados no caso geral, conseqüentemente, emprega menos dados no treinamento do modelo; por outro lado, no caso particular, os átomos selecionados nas primeiras iterações devem apresentar maior coerência com os resíduos.

Tabela 12: Melhores hiperparâmetros do treinamento usando a decomposição de blocos sem a imposição de SCRs centralizados para o classificador que considera os três subdivisórios Bateman, Gabor e Sigmóide.

Subdics.	Hiperparâmetros						
	λ	CDF	Neurônios	Dropout	Tx. Aprend.	Batch size	Épocas
Gabor Bateman Sigmoide	0,44	70%	100	0,1	0,003	64	205
	0,66	90%	100	0,2	0,001	30	250
	0,81	95%	100	0,2	0,002	110	255
	0,9	97%	100	0,3	0,001	30	345

Tabela 13: Acurácia do seletor de subdicionários ternários quando treinado com os resíduos de todos os blocos – sem a imposição de SCRs centralizados.

Subdics.	λ	CDF	Acurácia		
			Treino	Validação	Teste
Gabor Bateman Sigmoide	0,44	70%	73,50%	69,50%	41,00%
	0,66	90%	82,50%	74,50%	63,60%
	0,81	95%	86,30%	80,00%	72,80%
	0,9	97%	87,80%	81,20%	76,90%

A Figura 48 apresenta as matrizes de confusão desses classificadores. Assim como ocorreu para o caso envolvendo blocos com SCRs centralizadas, para $\lambda = 0,44$ verificaram-se muitos falsos positivos e negativos para todas as classes, chegando a valores maiores que os verdadeiros. Com o aumento do limite da taxa de aproximação, vemos o desempenho do modelo aumentar com maior concentração dos valores na diagonal principal. Ainda assim, percebemos uma dificuldade maior do classificador diferenciar átomos de *Bateman* e Sigmoide-exponencial apesar do percentil de erro entre esses dois subdicionários ser menor, quando comparado ao caso particular. Em compensação, o caso geral apresentou percentuais maiores de falsos negativos para os subdicionários de Gabor e Sigmoide.

3.4.2.2 Classificação Binária

Assim como foi feito para o caso particular, com blocos com SCRs centralizados, obtêm-se os hiperparâmetros que levam aos melhores desempenhos, que são apresentados na Tabela 14, enquanto a Tabela 15 apresenta as acurácias de treino, validação e teste e a Figura 49 apresenta a matriz de confusão para os três classificadores treinados. Os resultados são semelhantes aos obtidos com o banco de dados gerado usando somente blocos contendo SCRs centralizados. Comparando os resultados obtidos no treinamento, podemos observar uma acurácia um pouco superior para o seletor entre Gabor e Bateman que supera em 5 pontos percentuais e em apenas 1 ponto percentual os seletores entre Bateman e Sigmoide e entre Gabor e Sigmoide-Exponencial, respectivamente. Já no teste o desempenho do seletor entre Bateman e Sigmoide decai. Comparando-se os resultados com os dois bancos de dados, vemos que em alguns casos a acurácia diminui (Gabor/Bateman) e em outros aumenta (Gabor/Sigmoide).

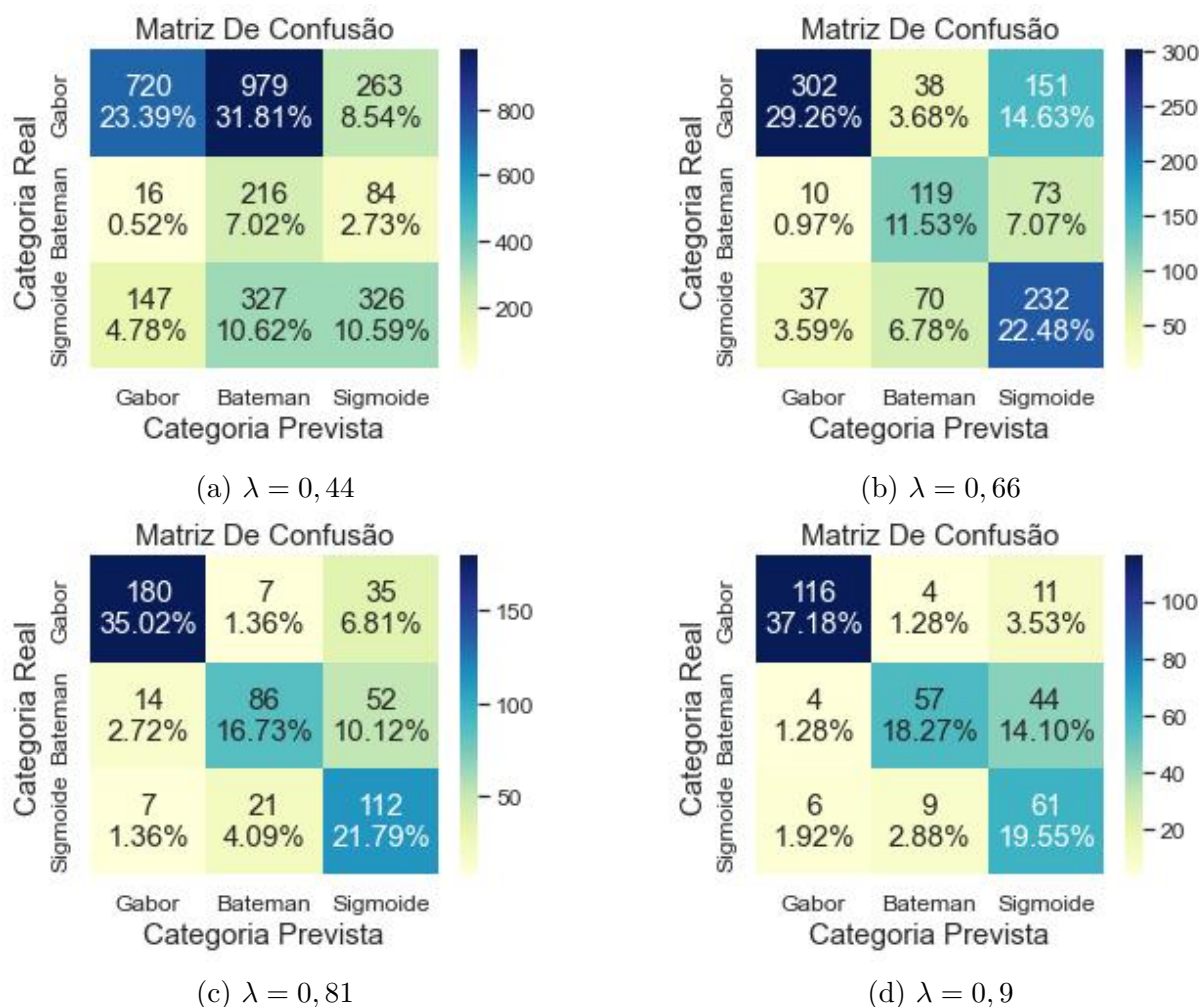


Figura 48: Matrizes de confusão para o classificador dentre três subdicionários na decomposição MP de blocos sem imposição de SCRs centralizados.

Tabela 14: Melhores hiperparâmetros do treinamento usando a decomposição de blocos sem a restrição de SCRs centralizados para os classificadores binários.

Subdics.	Hiperparâmetros						
	λ	CDF	Neurônios	Dropout	Tx. Aprend.	Batch size	Épocas
Gabor/Bateman	0,81	95%	100	0,2	0,001	82	340
Bateman/Sigmoide	0,81	95%	81	0,2	0,001	31	380
Gabor/Sigmoide	0,81	95%	100	0,3	0,001	30	436

Tabela 15: Acurácias de treino, validação e teste do seletor binário de subdicionários considerando resíduos produzidos a partir de todos os blocos (sem a restrição de SCRs centralizados).

Subdics.	λ	CDF	Acurácia		
			Treino	Validação	Teste
Gabor/Bateman	0,81	95%	97,3%	92,2%	90,9%
Bateman/Sigmoide	0,81	95%	87,1%	79,0%	72,6%
Gabor/Sigmoide	0,81	95%	96,2%	90,5%	86,7%

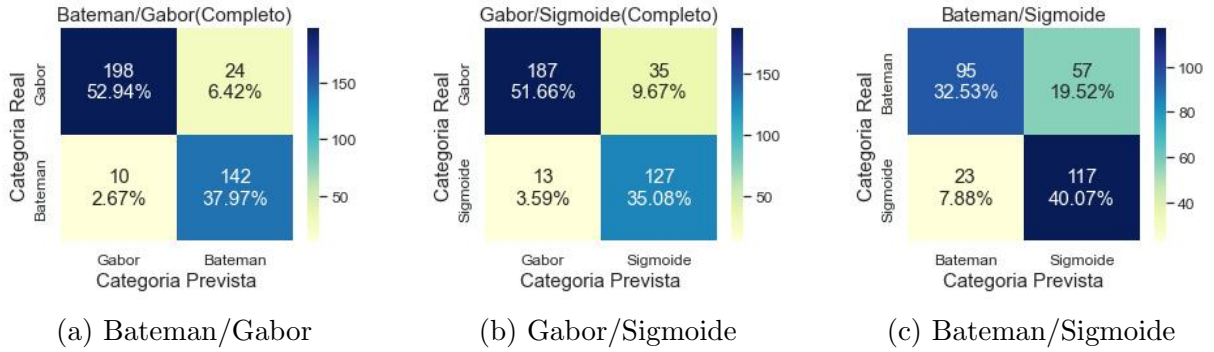


Figura 49: Matriz de confusão dos classificadores binários para pares de subdicionários considerando a decomposição MP de todos os blocos.

3.5 Resultados da Decomposição Usando o Seletor de Subdicionários no Laço do MP

Após treinado, validado e testado o modelo classificador, o empregamos no laço do algoritmo de decomposição para indicar o subdicionário no qual o MP deve procurar pelo átomo para aproximar o resíduo. Nota-se que se os três dicionários tem a mesma quantidade de elementos, essa estratégia reduz a complexidade da busca por três.

Vimos que a complexidade do classificador após treinado é $\mathcal{O}\left(\frac{N}{M}N_h + N_hN_o + N_o\right)$. Se cada subdicionário contém K átomos, a procura em cada um requer $KN^2 + N - 1$ operações. Com isso, podemos assumir que a complexidade computacional utilizando seleção de subdicionários via RNAs para N_o classes é $\mathcal{O}\left(\frac{N}{M}N_h + N_hN_o + N_o + KN^2 + N - 1\right)$, enquanto que a complexidade computacional sem a seleção usando 3 subdicionários é: $\mathcal{O}(3(KN^2 + N - 1))$. Através da Figura 50 é possível observar o custo computacional do MP com e sem a seleção de subdicionário via RNAs para 3 subdicionários em função do tamanho do sinal (N). Com a decimação e a limitação do número de neurônios na camada escondida (de N_h a 100), observa-se uma redução de complexidade da busca a quase um terço da anterior. Além disso, em casos com mais subdicionários, o ganho na redução computacional aumentaria, o custo computacional é dividido pela quantidade de subdicionários aproximadamente.

Observa-se que mesmo em casos em que o subdicionário apontado pelo classificador for incorreto, ainda assim o laço MP irá produzir uma redução de energia do resíduo. É verdade que essa redução não será a maior possível considerando o dicionário completo, entretanto, a energia do resíduo ainda assim reduz. Isto é, esse erro não é crítico para o processo, ainda mais se ocorrer em iterações do MP mais avançadas.

Para avaliar o desempenho do uso do classificador na decomposição, compara-se

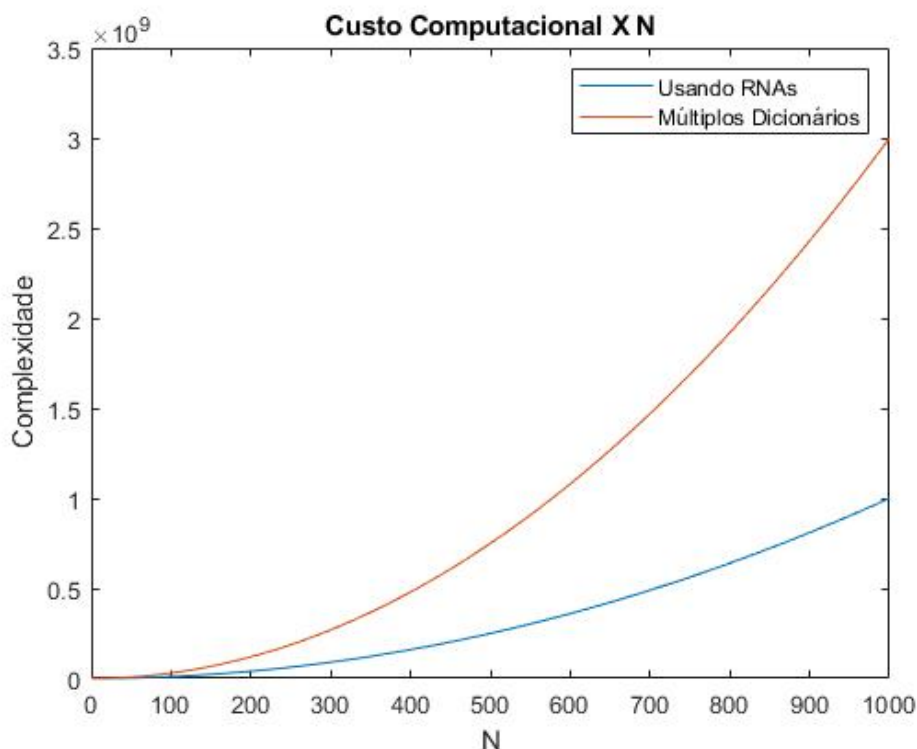


Figura 50: Custo computacional do *MP* com e sem seleção de subdicionários via RNAs em função do tamanho do sinal.

o desempenho de diferentes modelos contra dois casos de referência. O primeiro caso considera o melhor cenário possível usando o *MP* padrão visto no Capítulo 2, no qual o algoritmo percorre todo o dicionário para encontrar o átomo e seus parâmetros a serem usados para aproximar o resíduo. O segundo caso de referência seleciona aleatoriamente o subdicionário a ser usado em cada iteração do *MP*.

Na Figura 51, é possível observar os SNRs médios das decomposições obtidas para os dois sinais reservados para teste (não usados no treinamento e validação da RNA), empregando-se até 32 iterações para diferentes classificadores ternários considerando todos os blocos e os casos de referência acima discutidos. Quando se usa um classificador como seletor de subdicionário a perda relativamente ao *MP* padrão com base no dicionário completo existe mas é substancialmente menor que aquela produzida pela seleção aleatória do subdicionário. Vemos ainda que o melhor desempenho neste teste é obtido pelo classificador treinado com resíduos selecionados com o percentil (intermediário) igual a 90% ($\lambda = 0,66$). Ao reduzirmos o tamanho do conjunto de treinamento, restringindo-o de forma que contenha átomos mais coerentes com os resíduos, observa-se uma pequena perda de desempenho; porém, ao se empregar átomos menos coerentes com os resíduos

no treinamento ($\lambda = 0,44$), o desempenho decai. Entretanto, quando usam-se poucas iterações, a diferença entre os desempenhos quando diferentes classificadores são usados no laço do MP é pequena.

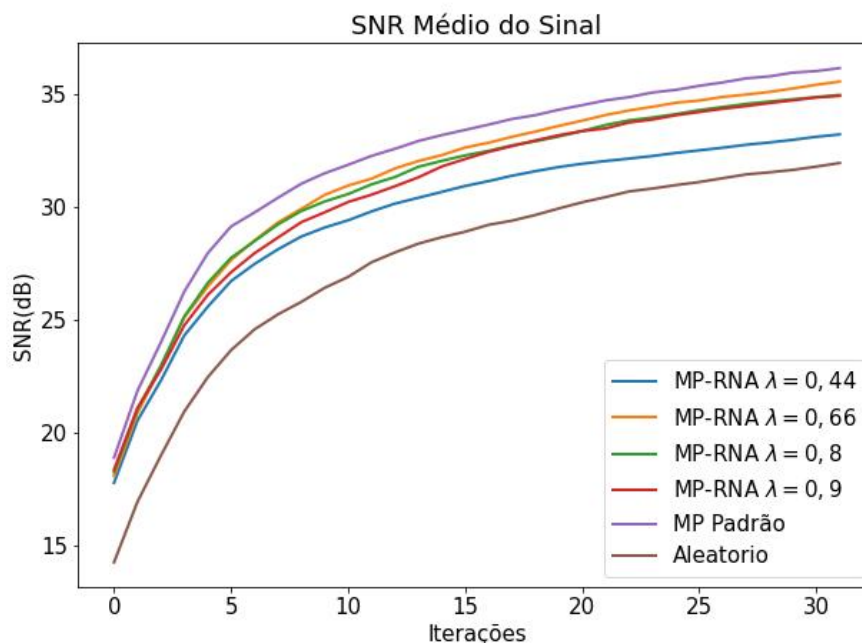


Figura 51: SNR de decomposição atômica usando o seletor de dicionário no laço MP.

Assim, uma forma de se empregar os classificadores de subdicionários no laço do MP seria utilizar o modelo até um dado limite da taxa de aproximação e após ele ser alcançado utilizar um subdicionário mais adequado para a modelagem de resíduos “ruidosos”. Isto é, emprega-se o seletor de dicionário usando RNAs durante as primeiras iterações do MP e após o limiar ser atingido podemos restringir a busca apenas no subdicionário de Gabor, já que este é mais adequado (pelas suas propriedades tempo-frequenciais) para a aproximação do ruído gerado pela decomposição que os demais subdicionários.

A Figura 52 compara o desempenho decorrente desse chaveamento do processo de escolha do subdicionário para os dois sinais destinados para teste usando todos os blocos (caso geral). Para avaliar o desempenho do MP com seletor de subdicionário usando chaveamento, compara-se o SNR pelas iterações do MP quando o subdicionário é selecionado sempre pela rede treinada e quando o seletor restringe a busca ao subdicionário de Gabor para os resíduos considerados como ruídos. O limiar do chaveamento é dado pela taxa

de aproximação utilizada para o treinamento da rede. Dessa maneira, a decomposição começa usando o seletor de subdicionários por meio de RNAs treinadas para uma determinada taxa de aproximação λ . Quando a taxa de aproximação do resíduo da iteração atual atinge esse limiar, o chaveamento é acionado para a escolher somente subdicionário de Gabor, simplificando a decomposição usando apenas um subdicionário em vez de múltiplos. É possível observar pela Figura 52 o mesmo desempenho nas primeiras iterações entre os modelos usando RNAs com e sem chaveamento. Isso é esperado já que ambos usam o mesmo seletor durante as primeiras iterações; entretanto, vemos que quando o MP fica restrito apenas ao subdicionário de Gabor, à medida que o número da iteração aumenta, obtém-se um desempenho superior.

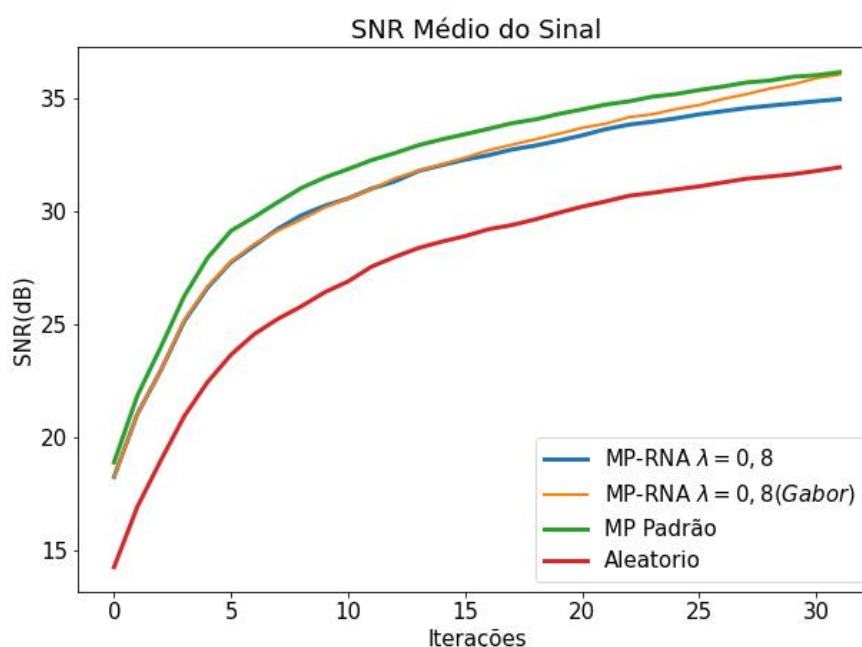


Figura 52: SNR de decomposição atômica usando o seletor de dicionário no laço MP e chaveando para o subdicionário de Gabor quando o resíduo se assemelha a ruído de decomposição MP.

De maneira geral, o uso de redes neurais para seleção do subdicionário no laço do algoritmo de MP, se mostrou eficaz reduzindo o custo computacional sem impactar em grandes perdas de desempenho, em termos de SNR, na capacidade de aproximação do sinal EDA.

CONCLUSÃO

Neste trabalho, foi proposto um modelo de representação de sinais de atividade eletrodérmica usando decomposições atômicas via algoritmo de *Matching Pursuit* com múltiplos dicionários redundantes (subdicionários). Com o objetivo de reduzir o custo computacional gerado pelo algoritmo, propôs-se empregar um modelo classificador no laço de decomposição utilizando redes neurais artificiais para selecionar o subdicionário mais adequado a cada iteração a partir dos próprios resíduos gerados pelo MP. Os resultados obtidos indicam que essa abordagem é eficaz em reduzir a complexidade computacional sem impactar fortemente a qualidade da decomposição.

Pré-Processamento do Sinal de Atividade Eletrodérmica

Inicialmente, apresentaram-se as características da atividade eletrodérmica e seu uso para monitoramento e diagnóstico de estados psicofisiológicos. Analisando o espectro de frequência do sinal EDA, vimos componentes de alta frequência de baixa energia, que no domínio do tempo, correspondem às características de serrilhamento presentes no sinal. Vimos que foi possível eliminá-las aplicando-se um filtro passa-baixa com frequência de corte de 0,5 Hz. Além disso, estudamos as duas componentes principais que compõe o sinal de EDA: fásica e tônica. Vimos que a componente fásica possui maior relevância para análise dos estados psicológicos. Devido a isso, lançamos mão de métodos para estimar ambas as componentes e separá-las.

Primeiramente, utilizou-se o MMQ (método dos mínimos quadrados) para estimar a componente tônica do sinal e, em seguida, obter a componente fásica subtraindo a componente tônica do sinal original. Alternativamente, aplicou-se o algoritmo de otimização convexa *cvxEDA* [18] para estimar a componente tônica através de B-splines e a fásica por uma convolução entre um IRF e uma sequência esparsa de impulsos não negativos associados. Com base nos resultados, vimos que ambos os métodos são capazes de estimar as componentes, contudo o método usando *splines* mostrou-se mais eficiente em comparação ao MMQ quanto à estimativa da componente tônica. Dentre as desvantagens e problemas do método MMQ para estimar a componente tônica, podemos mencionar o surgimento de erros decorrentes de variações abruptas na SCL (*Skin Conductance Level*) e de compo-

nentes físicas com valores negativos e efeitos derivados do tamanho de bloco fixo. Com isso, definiu-se o método cvxEDA como a melhor opção para obter a componente física a ser utilizada na etapa de decomposição atômica.

Decomposições Atômicas

Em seguida, abordou-se a fundamentação teórica a respeito da decomposição atômica. Estudou-se o algoritmo voraz *Matching Pursuit* e as técnicas para reduzir o seu custo computacional. Neste trabalho, foi proposta a decomposição usando átomos de múltiplos dicionários; permitindo usar átomos definidos a partir de quatro (4) formas de onda protótipo – funções matemáticas parametrizáveis, para representar a componente física da atividade eletrodérmica. Os protótipos usados foram as funções de *Bateman*, Sigmoidal-Exponencial, Chi-quadrado e Gaussiana (dicionário de Gabor).

A princípio, para compor o dicionário discretizaram-se os parâmetros de cada dicionário a partir das indicações da literatura. Analisou-se a distribuição dos átomos encontrados na decomposição para cada subdicionário. Com objetivo de reduzir a cardinalidade do dicionário e, conseqüentemente, reduzir o custo computacional sem incorrer em perdas significativas, dicionários subamostrados (no espaço de parâmetros que definem os átomos) foram definidos com cardinalidade 14 vezes menores que os originais com base na probabilidade dos átomos serem escolhidos. Com base nesses resultados, concluiu-se que o dicionário Chi-quadrado obteve um desempenho inferior aos demais subdicionários e não se identificou a escolha do mesmo quando todos os subdicionários compunham o dicionário. Por esta razão, desconsiderou-se este subdicionário no restante do trabalho. Nas primeiras iterações do algoritmo, observou-se uma probabilidade maior dos átomos serem protótipos de *Bateman* e Sigmoidal-Exponencial. Conforme a decomposição avança e a taxa de aproximação decai, observou-se uma predominância de átomos de Gabor. Apesar da redução significativa da complexidade por meio apenas da redução de cardinalidade, vimos que nas primeiras iterações a decomposição para ambos dicionários (completo e subamostrado) apresentam desempenhos similares e, com o decorrer das iterações, o dicionário subamostrado tem uma pequena perda de desempenho de no máximo 3 dB em termos de razão sinal-ruído do sinal (SNR) do sinal reconstruído. Assim, é possível compor um dicionário compacto e coerente reduzindo a cardinalidade e, conseqüentemente, o custo computacional do MP sem que haja perdas relevantes comparativamente ao uso do

dicionário completo.

Seleção de Subdicionários no Laço MP

Empregamos decomposições de sinais de EDA para treinar um modelo classificador utilizando redes neurais artificiais para escolher o subdicionário no qual buscar pelo átomo a aplicar para decompor o resíduo de iterações do MP. Esse classificador emprega um modelo de rede neural artificial, cuja entrada é o resíduo decimado. Ela foi treinada via *backpropagation* usando o algoritmo Adam, balanceamento para evitar viés na classificação, definição do hiperparâmetros através de busca Bayesiana e validação cruzada via *K-Folds*. Visando mitigar possíveis efeitos da definição dos blocos de sinais EDA, aplicou-se a metodologia de projeto de tal classificador considerando blocos com os SCRs centralizados ou não (os SCRs podem ou não estar centralizados no bloco). Avaliou-se o projeto do classificador variando-se o conjunto de treinamento a partir da força da medida de coerência (com base no percentil da taxa de aproximação) entre o resíduo e o átomo selecionado; para isso, foram usados quatro (4) percentis (que definem 4 taxas de aproximação) distintas. O melhor modelo resulta ao escolher a taxa de aproximação 0,9 (percentil 97%), com acurácia de 76,9%. Esta mostrou-se satisfatória quando realizamos a decomposição de outros sinais de teste (não vistos pelo modelo no treinamento) quando o modelo seletor de subdicionário é inserido no laço do MP. Vimos ainda que quando consideramos usar essa estratégia (o seletor de subdicionário) até a taxa de aproximação usada para definir o conjunto de treinamento do modelo e chavear para subdicionário de Gabor para as iterações seguintes, conseguimos obter uma aproximação do sinal bastante similar à obtida utilizando a busca (MP) no dicionário composto por todos os subdicionários. De maneira geral, o modelo proposto se apresentou promissor para ser utilizado em decomposições atômicas usando múltiplos dicionários, reduzindo consideravelmente o custo do computacional em tantas vezes quanto subdicionários houver.

Trabalhos Futuros

Em trabalhos futuros, deseja-se avaliar o desempenho de outros modelos classificadores como SVM, Árvores de decisão, KNN, AdaBoost, redes convolucionais usando espectrogramas dos resíduos como entrada ou redes neurais profundas. Deseja-se, inclu-

sive, fazer uso dos modelos de aprendizado de máquina para escolher outros parâmetros como a escala e/ou deslocamento, reduzindo ainda mais a complexidade da busca pelos átomos no dicionário.

Deseja-se ainda investigar como realizar a classificação de estados psicofisiológicos com base nos átomos escolhidos na decomposição MP. O empecilho reside na falta de uma base anotada referente ao diagnóstico dos pacientes.

REFERÊNCIAS

- [1] CHASPARI, T. et al. Sparse representation of electrodermal activity with knowledge-driven dictionaries. *IEEE Transactions on Biomedical Engineering*, v. 62, n. 3, p. 960–971, March 2015. ISSN 1558-2531.
- [2] KELSEY, M. et al. Applications of sparse recovery and dictionary learning to enhance analysis of ambulatory electrodermal activity data. *Biomedical Signal Processing and Control*, v. 40, p. 58–70, 02 2018.
- [3] BIRJANDTALAB, J. et al. A non-EEG biosignals dataset for assessment and visualization of neurological status. p. 110–114, Oct 2016. ISSN 2374-7390.
- [4] GUIMARÃES, F. dos S. et al. On the compression of ECG records employing triangular elements and analysis-by-synthesis modeling. In: IEEE. *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. [S.l.], 2010. p. 3084–3087.
- [5] HERNANDO-RAMIRO, C. et al. Consistent quality control in ECG compression by means of direct metrics. *Physiological measurement*, IOP Publishing, v. 36, n. 9, p. 1981, 2015.
- [6] LOVISOLO, L. *Representações Coerentes de Sinais Elétricos*. Dissertação (Dissertação de Mestrado) — PEE/COPPE, UFRJ, Rio de Janeiro, Brasil, 2001.
- [7] TCHEOU, M. P. *Compressão de Sinais Usando Decomposições Atômicas*. Tese (Tese de Doutorado) — PEE/COPPE, UFRJ, Rio de Janeiro, Brasil, 2011.
- [8] LOVISOLO, L. et al. Coherent decompositions of power systems signals using damped sinusoids with applications to denoising. In: IEEE. *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No. 02CH37353)*. [S.l.], 2002. v. 5, p. V–V.
- [9] LOVISOLO, L. et al. Efficient coherent adaptive representations of monitored electric signals in power systems using damped sinusoids. *IEEE Transactions on Signal Processing*, IEEE, v. 53, n. 10, p. 3831–3846, 2005.

- [10] TCHEOU, M. P. et al. Optimum rate-distortion dictionary selection for compression of atomic decompositions of electric disturbance signals. *IEEE Signal Processing Letters*, IEEE, v. 14, n. 2, p. 81–84, 2007.
- [11] RUBINSTEIN, R.; BRUCKSTEIN, A. M.; ELAD, M. Dictionaries for sparse representation modeling. *Proceedings of the IEEE*, IEEE, v. 98, n. 6, p. 1045–1057, 2010.
- [12] MACEFIELD, V. G.; WALLIN, B. G. The discharge behaviour of single sympathetic neurones supplying human sweat glands. *Journal of the autonomic nervous system*, Elsevier, v. 61, n. 3, p. 277–286, 1996.
- [13] LIM, C. L. et al. Decomposing skin conductance into tonic and phasic components. *International Journal of Psychophysiology*, Elsevier, v. 25, n. 2, p. 97–109, 1997.
- [14] ALEXANDER, D. et al. Separating individual skin conductance responses in a short interstimulus-interval paradigm. *Journal of Neuroscience Methods*, v. 146, n. 1, p. 116 – 123, 2005. ISSN 0165-0270. Disponible em: <<http://www.sciencedirect.com/science/article/pii/S0165027005000464>>.
- [15] BENEDEK, M.; KAERNBACH, C. Decomposition of skin conductance data by means of nonnegative deconvolution. *Psychophysiology*, v. 47, n. 4, p. 647–658, 2010. Disponible em: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.2009.00972.x>>.
- [16] BACH, D. R. et al. Modelling event-related skin conductance responses. *International Journal of Psychophysiology*, Elsevier, v. 75, n. 3, p. 349–356, 2010.
- [17] BACH, D. R.; FRISTON, K. J.; DOLAN, R. J. An improved algorithm for model-based analysis of evoked skin conductance responses. *Biological psychology*, Elsevier, v. 94, n. 3, p. 490–497, 2013.
- [18] GRECO, A. et al. cvxEDA: A convex optimization approach to electrodermal activity processing. *IEEE Transactions on Biomedical Engineering*, IEEE, v. 63, n. 4, p. 797–804, 2015.
- [19] GRECO, A. et al. Electrodermal activity in bipolar patients during affective elicitation. *IEEE Journal of Biomedical and health Informatics*, IEEE, v. 18, n. 6, p. 1865–1873, 2014.

- [20] TCHEOU, M. P. et al. How far can one compress digital fault records? analysis of a matching pursuit-based algorithm. *Digital Signal Processing*, Elsevier, v. 22, n. 2, p. 288–297, 2012.
- [21] MALLAT, S.; ZHANG, Z. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, v. 41, n. 12, p. 3397–3415, Dez. 1993.
- [22] MALLAT, S. *A Wavelet Tour of Signal Processing*. 2. ed. California, USA: Academic Press, 1998.
- [23] DEVORE, R. A.; TEMLYAKOV, V. N. Some remarks on greedy algorithms. *Advances in computational Mathematics*, Springer, v. 5, n. 1, p. 173–187, 1996.
- [24] STURM, B. L.; SHYNK, J. J. Sparse approximation and the pursuit of meaningful signal models with interference adaptation. *IEEE transactions on audio, speech, and language processing*, IEEE, v. 18, n. 3, p. 461–472, 2009.
- [25] OLIVEIRA, G.; TCHEOU, M. P.; LOVISOLO, L. Análise da decomposição de sinais de atividade eletrodérmica em múltiplos dicionários usando algoritmos vorazes. *SBRT2018*, v. 1, p. 5, 2018.
- [26] OLIVEIRA, G. A. de; TCHEOU, M. P.; LOVISOLO, L. Artificial neural networks for dictionary selection in adaptive greedy decomposition algorithms with reduced complexity. In: IEEE. *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2018. p. 1–8.
- [27] ZANTALIS, D. *Guided matching pursuit and its application to sound source separation*. Tese (Doutorado) — University of York, 2016.
- [28] MOUSSALLAM, M.; DAUDET, L.; RICHARD, G. Matching pursuits with random sequential subdictionaries. *Signal Processing*, Elsevier, v. 92, n. 10, p. 2532–2544, 2012.
- [29] MARSOUSI, M. et al. Multi-stage omp sparse coding using local matching pursuit atoms selection. In: IEEE. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. [S.l.], 2013. p. 1783–1787.
- [30] FERRANDO, S. E.; KOLASA, L. A.; KOVAČEVIĆ, N. Algorithm 820: a flexible implementation of matching pursuit for gabor functions on the interval. *ACM Transactions on Mathematical Software (TOMS)*, ACM, v. 28, n. 3, p. 337–353, 2002.

- [31] BOUCSEIN, W. *Electrodermal activity*. [S.l.]: Springer Science & Business Media, 2012.
- [32] CRITCHLEY, H. D. et al. Neural activity relating to generation and representation of galvanic skin conductance responses: a functional magnetic resonance imaging study. *Journal of Neuroscience*, Soc Neuroscience, v. 20, n. 8, p. 3033–3040, 2000.
- [33] MILLINGTON, P. F.; WILKINSON, R. *Skin*. [S.l.]: Cambridge University Press, 1983.
- [34] LIM, C. et al. Sudomotor nerve conduction velocity and central processing time of the skin conductance response. *Clinical Neurophysiology*, Elsevier, v. 114, n. 11, p. 2172–2180, 2003.
- [35] KUNIMOTO, M. et al. Neuroeffector characteristics of sweat glands in the human hand activated by regular neural stimuli. *The Journal of physiology*, Wiley Online Library, v. 442, n. 1, p. 391–411, 1991.
- [36] KENNEDY, W. R.; WENDELSCHAFER-CRABB, G.; BRELJE, T. C. Innervation and vasculature of human sweat glands: an immunohistochemistry-laser scanning confocal fluorescence microscopy study. *Journal of Neuroscience*, Soc Neuroscience, v. 14, n. 11, p. 6825–6833, 1994.
- [37] SCHMELZ, M. et al. Innervation territories of single sympathetic c fibers in human skin. *Journal of neurophysiology*, American Physiological Society Bethesda, MD, v. 79, n. 4, p. 1653–1660, 1998.
- [38] NISHIYAMA, T. et al. Irregular activation of individual sweat glands in human sole observed by a videomicroscopy. *Autonomic Neuroscience*, Elsevier, v. 88, n. 1-2, p. 117–126, 2001.
- [39] VALLBO, A. B.; HAGBARTH, K.-E.; WALLIN, B. G. Microneurography: how the technique developed and its role in the investigation of the sympathetic nervous system. *Journal of applied physiology*, American Physiological Society, v. 96, n. 4, p. 1262–1269, 2004.

- [40] BINI, G. et al. Thermoregulatory and rhythm-generating mechanisms governing the sudomotor and vasoconstrictor outflow in human cutaneous nerves. *The Journal of physiology*, Wiley Online Library, v. 306, n. 1, p. 537–552, 1980.
- [41] HANDBOOK of Psychophysiology. 3. ed. [S.l.]: Cambridge University Press, 2007.
- [42] ROTH, W. T.; DAWSON, M. E.; FILION, D. L. Publication recommendations for electrodermal measurements. *Psychophysiology*, Wiley Online Library, v. 49, n. 8, p. 1017–1034, 2012.
- [43] DAWSON, M. E.; SCHELL, A. M.; FILION, D. L. The electrodermal system. Cambridge University Press, 2017.
- [44] BONSOR, K. *How lie detectors work*. [S.l.]: www.people.howstuffworks.com/lie-detector1.html, 2018.
- [45] GOLDBERGER, A. L. et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, American Heart Association, Inc., v. 101, n. 23, p. e215–e220, June 2000. ISSN 0009-7322. Disponível em: <http://circ.ahajournals.org/content/101/23/e215>.
- [46] BRAITHWAITE, J. J. et al. A guide for analysing electrodermal activity (eda) & skin conductance responses (scrs) for psychological experiments. *Psychophysiology*, v. 49, n. 1, p. 1017–1034, 2013.
- [47] GOLUB, G. H.; LOAN, C. F. V. *Matrix Computations (3rd Ed.)*. Baltimore, MD, USA: Johns Hopkins University Press, 1996. ISBN 0-8018-5414-8.
- [48] BOYD, S. P.; VANDENBERGHE, L. *Convex optimization*. [S.l.]: Cambridge university press, 2004.
- [49] SCHNEIDER, R. A mathematical-model of human-skin conductance. In: SOC PSYCHOPHYSIOL RES 1010 VERMONT AVE NW SUITE 1100, WASHINGTON, DC 20005. *Psychophysiology*. [S.l.], 1987. v. 24, n. 5, p. 610–610.
- [50] VOGELSTEIN, J. T. et al. Fast nonnegative deconvolution for spike train inference from population calcium imaging. *Journal of neurophysiology*, American Physiological Society Bethesda, MD, v. 104, n. 6, p. 3691–3704, 2010.

- [51] O'BRIEN, M. S.; SINCLAIR, A. N.; KRAMER, S. M. Recovery of a sparse spike time series by l_1 deconvolution. *IEEE Transactions on Signal Processing*, IEEE, v. 42, n. 12, p. 3353–3365, 1994.
- [52] ROOI, J. de; EILERS, P. Deconvolution of pulse trains with the l_0 penalty. *Analytica chimica acta*, Elsevier, v. 705, n. 1-2, p. 218–226, 2011.
- [53] MATLAB. *9.7.0.1190202 (R2019b)*. Natick, Massachusetts: The MathWorks Inc., 2018.
- [54] LATHI, B. *Sinais e Sistemas Lineares - 2.ed.* Bookman, 2007. ISBN 9788560031139. Disponível em: <<https://books.google.com.br/books?id=ySxoo2TVeeYC>>.
- [55] GOODWIN, M. M. *Adaptive signal models: Theory, algorithms, and audio applications*. [S.l.]: Springer Science & Business Media, 2012.
- [56] GOODWIN, M. M.; VETTERLI, M. Matching pursuit and atomic signal models based on recursive filter banks. *IEEE Transactions on Signal Processing*, IEEE, v. 47, n. 7, p. 1890–1902, 1999.
- [57] NEFF, R.; ZAKHOR, A. Modulus quantization for matching-pursuit video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, IEEE, v. 10, n. 6, p. 895–912, 2000.
- [58] DONOHO, D. L. et al. Data compression and harmonic analysis. *IEEE transactions on information theory*, IEEE, v. 44, n. 6, p. 2435–2476, 1998.
- [59] CAETANO, R.; SILVA, E. da; CIANCIO, A. G. Matching pursuits video coding using generalized bit-planes. In: IEEE. *Proceedings. International Conference on Image Processing*. [S.l.], 2002. v. 3, p. III–III.
- [60] MALLAT, S. *A wavelet tour of signal processing: the sparse way*. [S.l.]: Academic press, 2008.
- [61] LOVISOLO, L.; SILVA, E. A. da. Frames in signal processing. In: *Academic Press Library in Signal Processing*. [S.l.]: Elsevier, 2014. v. 1, p. 561–590.

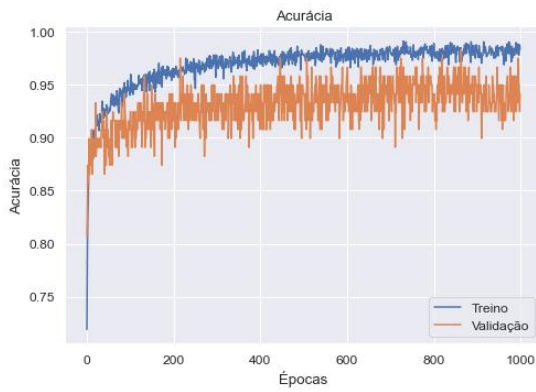
- [62] KELSEY, M. et al. Dictionary learning and sparse recovery for electrodermal activity analysis. In: SPIE. *Compressive Sensing V: From Diverse Modalities to Big Data Analytics*. [S.l.], 2016. v. 9857, p. 126–143.
- [63] KRSTULOVIC, S.; GRIBONVAL, R. MPTK: Matching pursuit made tractable. In: IEEE. *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. [S.l.], 2006. v. 3, p. III–III.
- [64] DINIZ, P. S.; SILVA, E. A. da; NETTO, S. L. *Processamento digital de sinais: Projeto e análise de sistemas*. [S.l.]: Bookman Editora, 2014.
- [65] DAVIS, G.; MALLAT, S.; AVELLANEDA, M. Adaptive greedy approximations. *Constructive approximation*, Springer, v. 13, n. 1, p. 57–98, 1997.
- [66] LOVISOLO, L.; SILVA, E. A. D.; DINIZ, P. S. On the statistics of matching pursuit angles. *Signal processing*, Elsevier, v. 90, n. 12, p. 3164–3184, 2010.
- [67] KOTSIANTIS, S. B. et al. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, Amsterdam, v. 160, n. 1, p. 3–24, 2007.
- [68] DEO, R. C. Machine learning in medicine. *Circulation*, Am Heart Assoc, v. 132, n. 20, p. 1920–1930, 2015.
- [69] AHMED, S. et al. Artificial intelligence and machine learning in finance: A bibliometric review. *Research in International Business and Finance*, Elsevier, v. 61, p. 101646, 2022.
- [70] HOSSEINI, M.-P.; HOSSEINI, A.; AHI, K. A review on machine learning for EEG signal processing in bioengineering. *IEEE reviews in biomedical engineering*, IEEE, v. 14, p. 204–218, 2020.
- [71] KOTSIANTIS, S. B. Decision trees: a recent overview. *Artificial Intelligence Review*, Springer, v. 39, p. 261–283, 2013.
- [72] DREISEITL, S.; OHNO-MACHADO, L. Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, Elsevier, v. 35, n. 5-6, p. 352–359, 2002.

- [73] PETERSON, L. E. K-nearest neighbor. *Scholarpedia*, v. 4, n. 2, p. 1883, 2009.
- [74] HEARST, M. A. et al. Support vector machines. *IEEE Intelligent Systems and their applications*, IEEE, v. 13, n. 4, p. 18–28, 1998.
- [75] MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, p. 115–133, 1943.
- [76] CROSS, S.; HARRISON, R.; KENNEDY, R. *Introduction to neural networks "The Lancet"*; vol. 346, no. 8982. [S.l.]: October, 1995.
- [77] HAYKIN, S. *Neural networks: a comprehensive foundation*. [S.l.]: Prentice Hall PTR, 1998.
- [78] FLECK, L. et al. Redes neurais artificiais: Princípios básicos. *Revista Eletrônica Científica Inovação e Tecnologia*, v. 1, n. 13, p. 47–57, 2016.
- [79] SHARMA, S.; SHARMA, S.; ATHAIYA, A. Activation functions in neural networks. *Towards Data Sci*, v. 6, n. 12, p. 310–316, 2017.
- [80] GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016.
- [81] NWANKPA, C. et al. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*, 2018.
- [82] HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. [S.l.]: Prentice Hall PTR, 1994.
- [83] KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [84] AMARI, S.-i. Backpropagation and stochastic gradient descent method. *Neurocomputing*, Elsevier, v. 5, n. 4-5, p. 185–196, 1993.
- [85] HECHT-NIELSEN, R. Theory of the backpropagation neural network. In: *Neural networks for perception*. [S.l.]: Elsevier, 1992. p. 65–93.
- [86] RUBY, U.; YENDAPALLI, V. Binary cross entropy with deep learning technique for image classification. *Int. J. Adv. Trends Comput. Sci. Eng*, v. 9, n. 10, 2020.

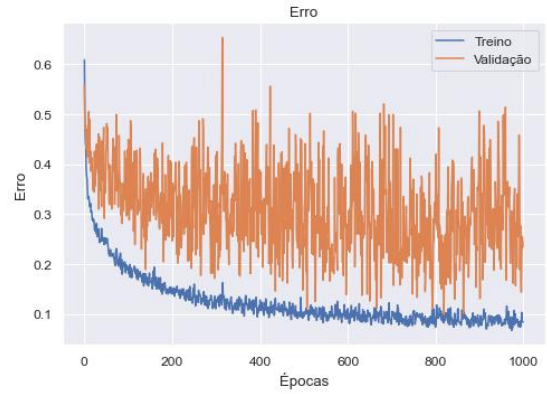
- [87] ROBBINS, H.; MONRO, S. A stochastic approximation method. *The annals of mathematical statistics*, JSTOR, p. 400–407, 1951.
- [88] RUDER, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.
- [89] LYDIA, A.; FRANCIS, S. Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci.*, v. 6, n. 5, p. 566–568, 2019.
- [90] SRIVASTAVA, N. et al. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, JMLR.org, v. 15, n. 1, p. 1929–1958, 2014.
- [91] CAPÍTULO 23 – Como Funciona o Dropout? Data Science Academy, Dec 2022. Disponível em: <<https://www.deeplearningbook.com.br/>>.
- [92] MCKINNEY, W. Pandas, python data analysis library. URL <http://pandas.pydata.org>, p. 3–15, 2015.
- [93] HARRIS, C. R. et al. Array programming with numpy. *Nature*, Nature Publishing Group UK London, v. 585, n. 7825, p. 357–362, 2020.
- [94] PEDREGOSA, F. et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, v. 12, n. Oct, p. 2825–2830, 2011.
- [95] CHOLLET, F. et al. *Keras*. GitHub, 2015. Disponível em: <<https://github.com/fchollet/keras>>.
- [96] CROCHIERE, R. E.; RABINER, L. R. Interpolation and decimation of digital signals—a tutorial review. *Proceedings of the IEEE*, IEEE, v. 69, n. 3, p. 300–331, 1981.
- [97] LONGADGE, R.; DONGRE, S. Class imbalance problem in data mining review. *arXiv preprint arXiv:1305.1707*, 2013.
- [98] LIU, X.-Y.; WU, J.; ZHOU, Z.-H. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 39, n. 2, p. 539–550, 2008.

- [99] LING, C. X. Overfitting and generalization in learning discrete patterns. *Neurocomputing*, Elsevier, v. 8, n. 3, p. 341–347, 1995.
- [100] WU, J. et al. Hyperparameter optimization for machine learning models based on bayesian optimization. *Journal of Electronic Science and Technology*, Elsevier, v. 17, n. 1, p. 26–40, 2019.
- [101] TURNER, R. et al. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In: PMLR. *NeurIPS 2020 Competition and Demonstration Track*. [S.l.], 2021. p. 3–26.
- [102] WITTEN, I. H. et al. Practical machine learning tools and techniques. In: *Data Mining*. [S.l.: s.n.], 2005. v. 2, n. 4.
- [103] LIU, L.; ÖZSU, M. T. *Encyclopedia of database systems*. [S.l.]: Springer, 2009.
- [104] HOSSIN, M.; SULAIMAN, M. N. A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, Academy & Industry Research Collaboration Center (AIRCC), v. 5, n. 2, p. 1, 2015.
- [105] KANG, S.; CHO, S.; KANG, P. Constructing a multi-class classifier using one-against-one approach with different binary classifiers. *Neurocomputing*, Elsevier, v. 149, p. 677–682, 2015.

A CURVAS DE APRENDIZADO DOS SELETORES DE SUBDICIONÁRIOS

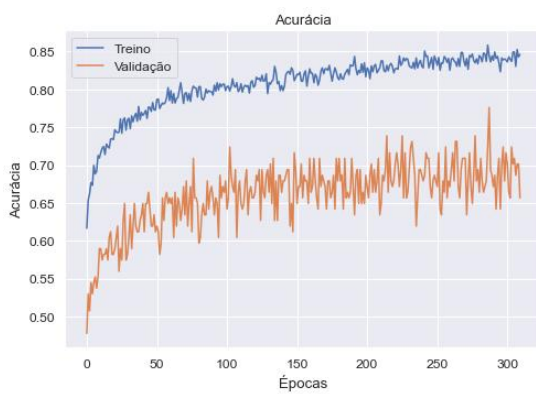


(a) Acurácia X Épocas

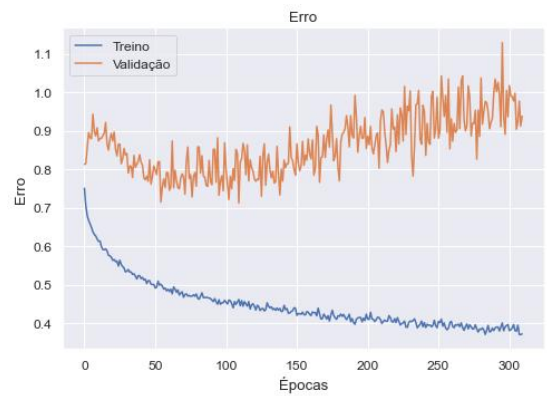


(b) Erro X Épocas

Figura 53: Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários *Bateman* e Gabor.

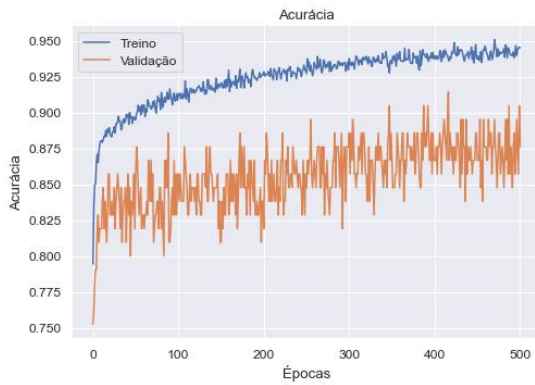


(a) Acurácia X Épocas

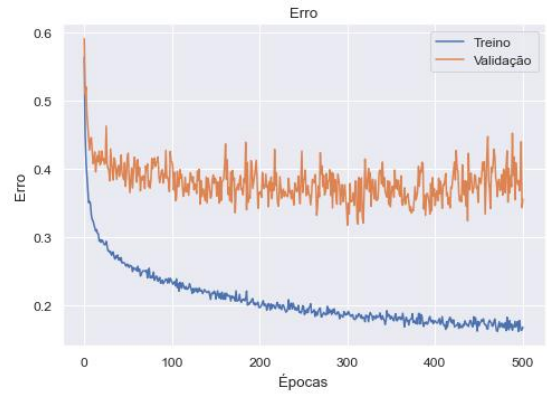


(b) Erro X Épocas

Figura 54: Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários *Bateman* e Sigmoide-Exponencial.

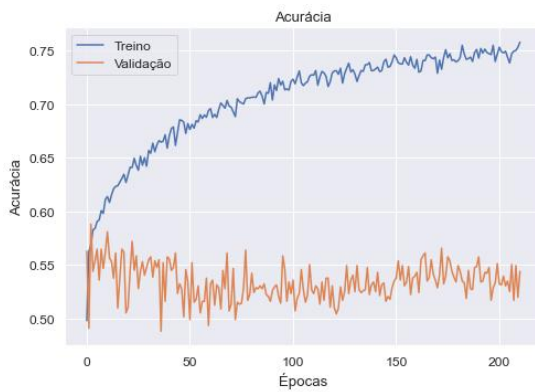


(a) Acurácia X Épocas

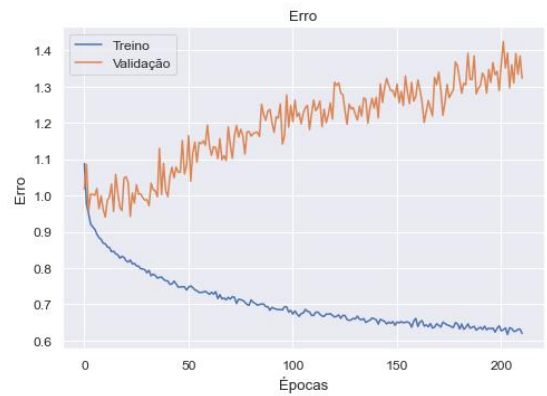


(b) Erro X Épocas

Figura 55: Curva de Aprendizado para o caso com SCRs centralizados utilizando 2 subdicionários Sigmoid-Exponencial e Gabor.

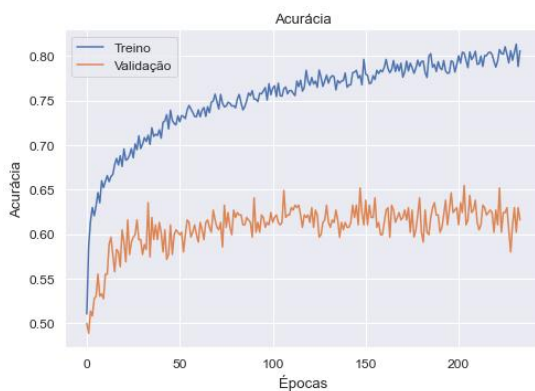


(a) Acurácia X Épocas

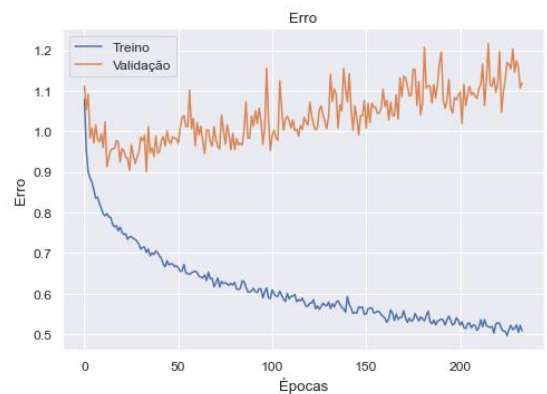


(b) Erro X Épocas

Figura 56: Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoid-Exponencial para $\lambda = 0,44$.

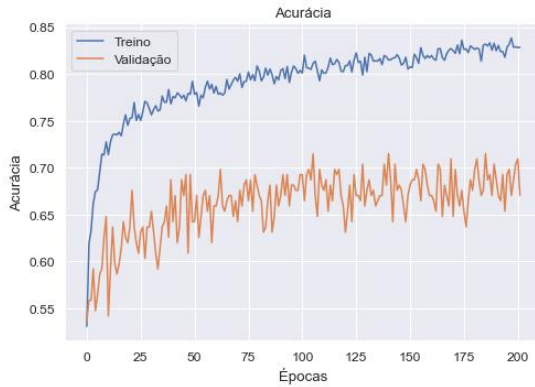


(a) Acurácia X Épocas

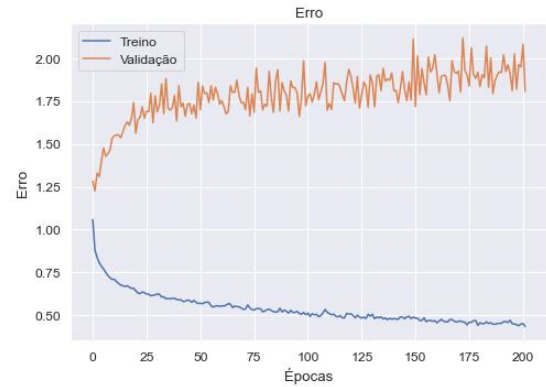


(b) Erro X Épocas

Figura 57: Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoid-Exponencial para $\lambda = 0,66$.

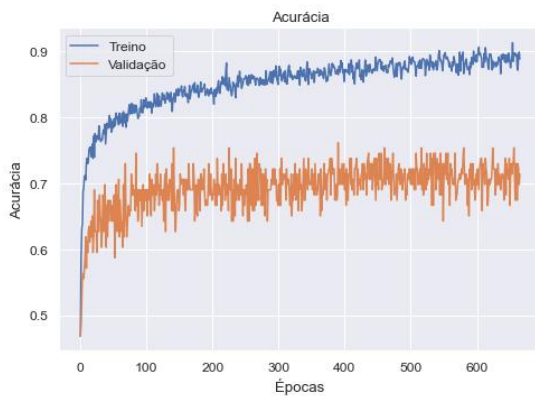


(a) Acurácia X Épocas

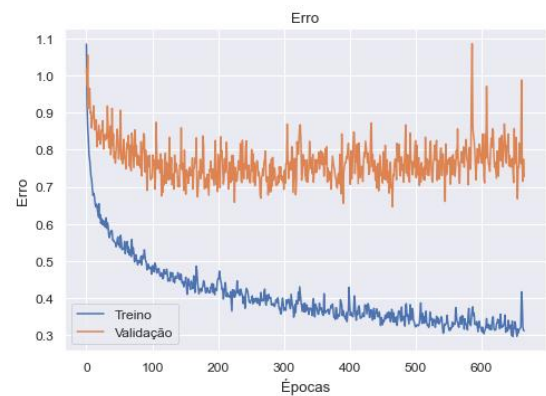


(b) Erro X Épocas

Figura 58: Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários *Bateman*, *Gabor* e *Sigmoide-Exponencial* para $\lambda = 0,81$.

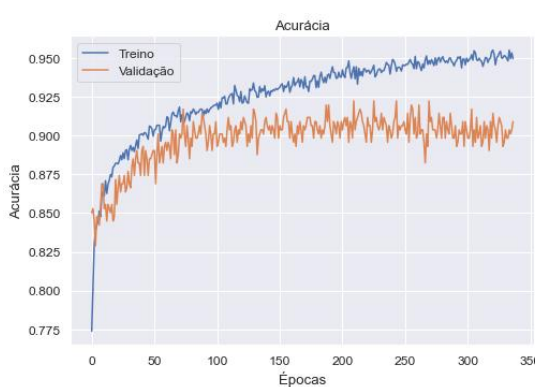


(a) Acurácia X Épocas

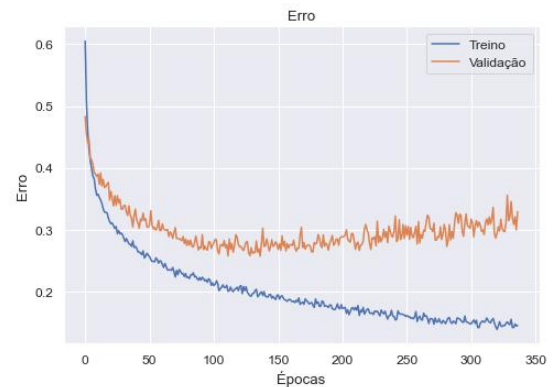


(b) Erro X Épocas

Figura 59: Curva de Aprendizado para o caso com SCRs centralizados utilizando os 3 subdicionários *Bateman*, *Gabor* e *Sigmoide Exponencial* para $\lambda = 0,9$.

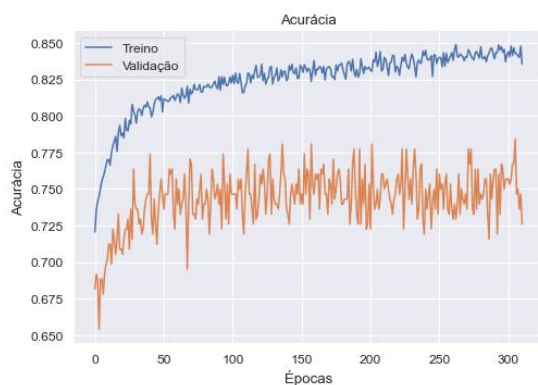


(a) Acurácia X Épocas

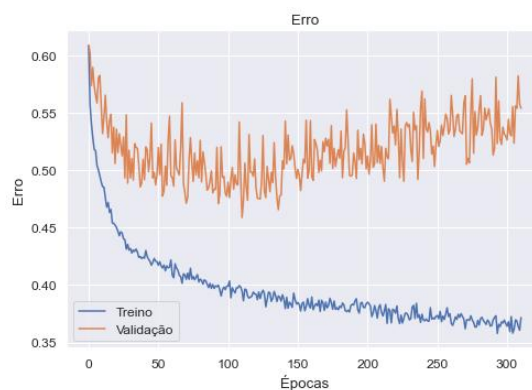


(b) Erro X Épocas

Figura 60: Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários *Bateman* e *Gabor*.

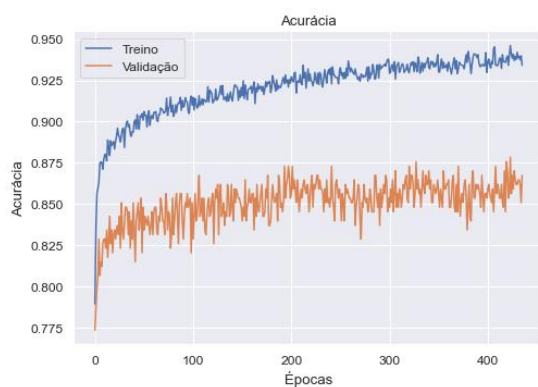


(a) Acurácia X Épocas

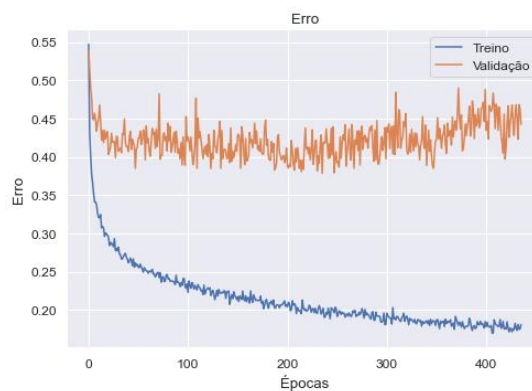


(b) Erro X Épocas

Figura 61: Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários *Bateman* e Sigmoid-Exponencial.

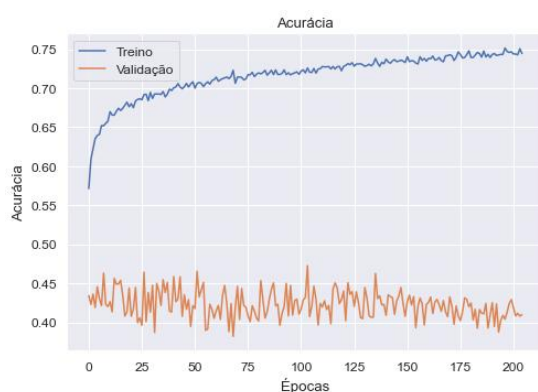


(a) Acurácia X Épocas

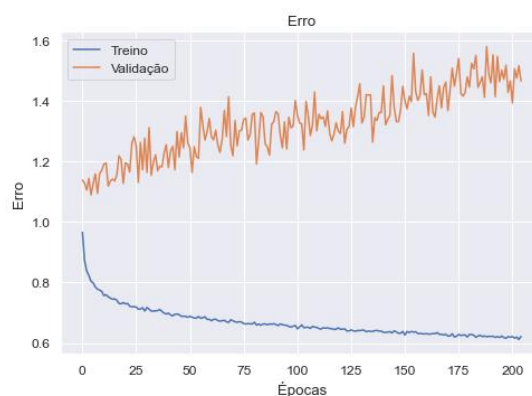


(b) Erro X Épocas

Figura 62: Curva de Aprendizado para o caso sem centralização de SCRs utilizando 2 subdicionários Sigmoid-Exponencial e Gabor.

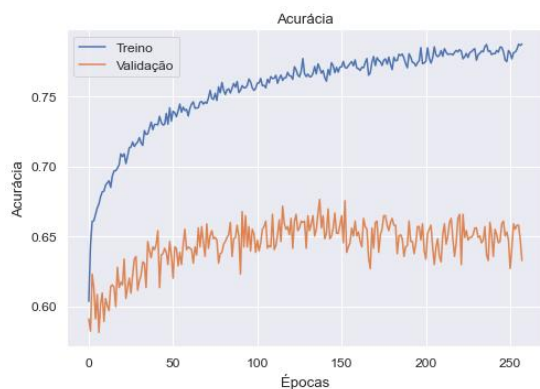


(a) Acurácia X Épocas

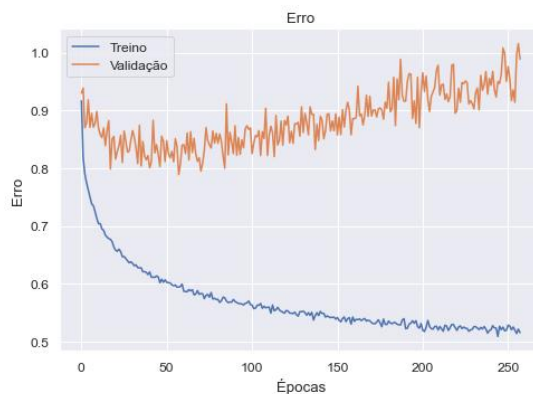


(b) Erro X Épocas

Figura 63: Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoid-Exponencial para $\lambda = 0,44$.

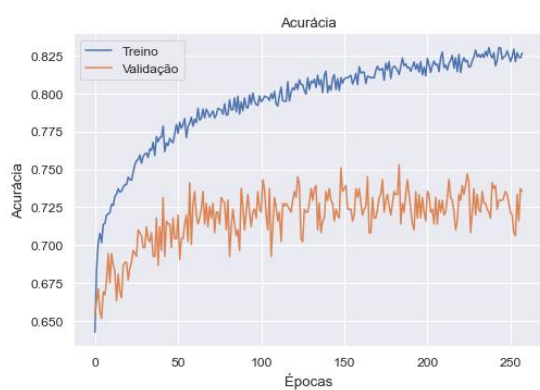


(a) Acurácia X Épocas

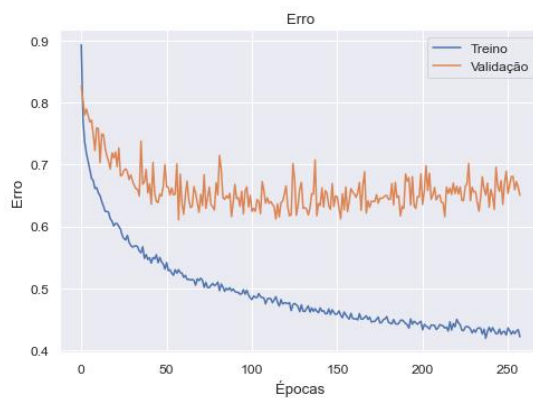


(b) Erro X Épocas

Figura 64: Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoide-Exponencial para $\lambda = 0,66$.

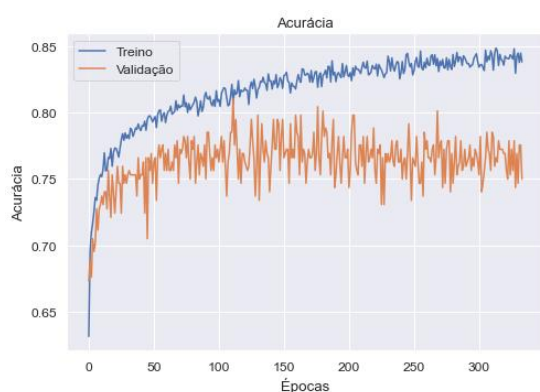


(a) Acurácia X Épocas

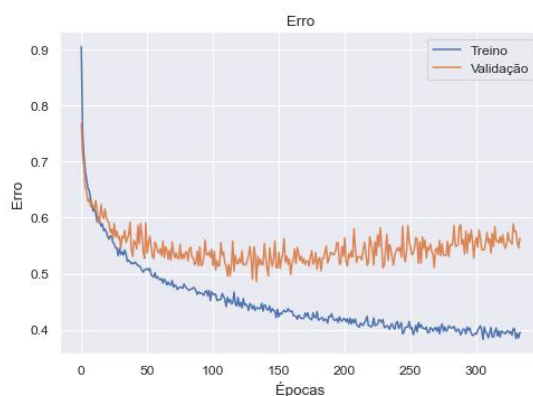


(b) Erro X Épocas

Figura 65: Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoide-Exponencial para $\lambda = 0,81$.



(a) Acurácia X Épocas



(b) Erro X Épocas

Figura 66: Curva de Aprendizado para o caso sem centralização de SCRs utilizando os 3 subdicionários *Bateman*, Gabor e Sigmoide Exponencial para $\lambda = 0,9$.