

DISSERTAÇÃO DE MESTRADO Nº 959

**UM ESTUDO SOBRE O AGRUPAMENTO DE SÉRIES TEMPORAIS E SUA
APLICAÇÃO EM CURVAS DE CARGA RESIDENCIAIS**

Pedro Leal Pazzini Da Silva

DATA DA DEFESA: 23/12/2016

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

**UM ESTUDO SOBRE O AGRUPAMENTO DE SÉRIES
TEMPORAIS E SUA APLICAÇÃO EM CURVAS DE CARGA
RESIDENCIAIS**

Pedro Leal Pazzini Da Silva

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do Título de Mestre em Engenharia Elétrica.

Orientador: Prof. Cristiano Leite de Castro

Coorientador: Prof. Frederico Gadelha Guimarães

Belo Horizonte - MG

Dezembro de 2016

S586e

Silva, Pedro Leal Pazzini da.

Um estudo sobre o agrupamento de séries temporais e sua aplicação em curvas de carga residenciais [manuscrito] / Pedro Leal Pazzini da Silva. – 2016.

xviii, 171 f., enc.: il.

Orientador: Cristiano Leite de Castro.

Coorientador: Frederico Gadelha Guimarães.

Dissertação (mestrado) Universidade Federal de Minas Gerais, Escola de Engenharia.

Apêndices: f. 86-171.

Bibliografia: f. 79-85.

1. Engenharia elétrica - Teses. 2. Análise de séries temporais - Teses.
3. Medidas elétricas - Teses. 4. Carga e distribuição elétrica - Teses. I.
Castro, Cristiano Leite de. II. Guimarães, Frederico Gadelha. III.
Universidade Federal de Minas Gerais. Escola de Engenharia. IV.
Título.

CDU: 621.3(043)

**"Um Estudo sobre o Agrupamento de Séries Temporais e
sua Aplicação em Curvas de Carga Residenciais"**

Pedro Leal Pazzini Da Silva

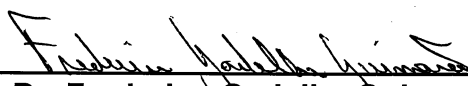
Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Mestre em Engenharia Elétrica.

Aprovada em 23 de dezembro de 2016.

Por:



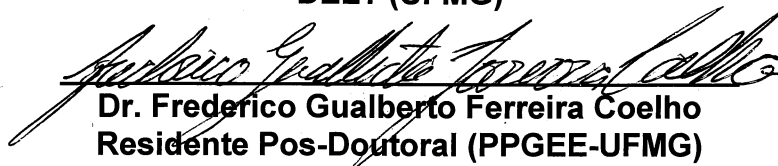
Prof. Dr. Cristiano Leite de Castro
(UFMG) - Orientador



Prof. Dr. Frederico Gadelha Guimarães
DEE (UFMG)



Prof. Dr. André Paim Lemos
DELT (UFMG)



Dr. Frederico Gualberto Ferreira Coelho
Residente Pos-Doutoral (PPGEE-UFMG)

Abstract

The union of computational, measurement and communication systems applied to electrical power systems will form the Smart Grids. One of its main components are the Smart Meters, which are measurement systems that periodically measure the consumed power in electrical loads, and that send the measurements to the electricity companies. The research, and mainly the installation of smart meters is more developed in the north hemisphere countries, although it already started in Brazil. When this measures become ubiquitous in the Brazilian electrical power system, a considerable amount of data will be produced, and the extraction of useful information from it will be necessary. One of the possible data mining tasks that could be applied to this data, and which is the object of this dissertation, is the clustering of load curves, from which is expected to get information and knowledge that can enhance the distribution service, the load forecast, the planning of expansion, and others. As the data of the smart meters installed in the Brazilian electrical system is not publicly available, the load curves of Australian residences were clustered. Since that, in this country, the climatic variations that could influence in the consuming profile are very similar to the climatic Brazilian variations. The problem of clustering load profiles is a time series clustering problem, since that for each measurement there is the temporal component associated. Thus, in this master dissertation was presented, initially, a study of clustering of time series, and this study is focused in the preprocessing, dissimilarity measure and algorithm applied, as well as the method of evaluating the results. Initially, after an empirical study in labeled datasets, the general clustering of time series strategy was proposed, and it is composed by: Z normalization in the preprocessing step, CID-DTW as the dissimilarity the hierarchical clustering algorithm using the average linkage and the silhouette And Calinski-Harabasz index for the evaluation of the obtained partitions. In the case study, with the Australian dataset, such strategy generated highly unbalanced partitions, which motivated the proposition of a multiobjective approach, in which partitions with a good compromise between the validation index values, which is a measure of the desired low variation intragroup and high variance intergroup, and balanced partitions, in which the variance of the number of instances in each cluster is low. The multi-objective approach indicates that the max normalization, the k-means algorithm and the euclidean distance is the most appropriate configuration for the clustering of load profiles generated by the smart meters.

Keywords: Clustering, time series, load curves, smart meters.

Resumo

A união de sistemas computacionais, de medição e de comunicação aplicada aos sistemas elétricos de potência dará forma às chamadas Redes Inteligentes. Um de seus componentes principais são os chamados medidores inteligentes, que são instrumentos que medem periodicamente a potência consumida em cargas elétricas, e enviam estes dados de medição para as distribuidoras de energia. A pesquisa, e principalmente a instalação de medidores inteligentes é mais intensa nos países do hemisfério norte, no entanto ela já se iniciou também no Brasil. A partir do momento em que esses medidores se tornarem onipresentes no sistema elétrico de potência brasileiro, uma massa de dados considerável será gerada, e se fará necessária a extração de informação útil dela. Uma das possíveis tarefas de mineração de dados a ser realizada nesta massa de dados, e que será objeto desta dissertação de mestrado, será o agrupamento das cargas, a partir do qual se espera obter informações e conhecimento que possa aprimorar o serviço de distribuição, a previsão de demanda, o planejamento de expansão da rede, dentre outras possibilidades. Uma vez que ainda não estão disponíveis os dados dos medidores inteligentes instalados no Brasil, foi realizado o agrupamento de curvas de carga medidas em residências australianas, já que neste país, as variações climáticas que podem influenciar no consumo de energia elétrica são bem similares às variações climáticas brasileiras. O problema de agrupamento de curvas de carga, é um problema de agrupamento de séries temporais, já que a cada medição realizada pelos medidores inteligentes existe uma componente temporal associada. Assim, nesta dissertação de mestrado foi apresentado, inicialmente, um estudo das estratégias de agrupamento de séries temporais no que diz respeito ao pré-processamento empregado, métrica de dissimilaridade e algoritmo de agrupamento utilizados, bem como o método de avaliação dos resultados obtidos. Inicialmente, em um estudo empírico em base de dados rotuladas, foi definida uma estratégia genérica para o agrupamento de séries temporais, composta por: normalização *Z* na etapa de pré-processamento, escolha da métrica de dissimilaridade CID-DTW e o algoritmo hierárquico com *linkagem average* para a realização do agrupamento em si, e os índices de validação silhouette e Calinski-Harabasz para a avaliação das partições obtidas. No estudo de caso, com os dados australianos, tal estratégia gerou grupos altamente desbalanceados, o que motivou a proposição de uma abordagem multi-objetiva, na qual se procura obter partições com um bom índice de validação, que por sua vez indica a tão desejada baixa variância intragrupo e alta variância intergrupo, e partições balanceadas, nas quais o número de instâncias em cada grupo possui baixa variância. A abordagem multi-objetiva indica que a normalização *max*, aliada ao algoritmo *k-means*, fazendo uso da distância euclidiana como métrica de dissimilaridade é a configuração mais adequada para o agrupamento de curvas de carga geradas por medidores inteligentes.

Palavras-chave:

Agrupamento, séries temporais, curvas de carga, medidores inteligentes.

Sumário

Agradecimentos	ix
1 Introdução	1
1.1 Objetivos e contribuições	2
1.2 Visão Geral do Texto	5
2 Agrupamento de séries temporais	6
2.1 Pré-processamento de séries temporais	7
2.1.1 Normalização	8
2.1.2 <i>Piecewise aggregate approximation</i> - PAA	9
2.1.3 Transformada discreta de Fourier - DFT	10
2.2 Métricas de dissimilaridade	11
2.2.1 Minkowski	12
2.2.2 <i>Dynamic Time Warping</i> - DTW	14
2.2.3 <i>LB-Keogh</i>	18
2.2.4 <i>Edit distance on real sequence</i> - EDR	21
2.2.5 <i>Dice, Jaccard e Lorentzian</i>	21
2.2.6 <i>Cosine</i>	22
2.2.7 Distância de Mahalanobis	22
2.2.8 Distâncias baseadas na correlação de Pearson	22
2.2.9 Distância baseada na correlação temporal	23
2.2.10 <i>Complex Invariant Distance</i> - CID	24
2.2.11 Discussão geral das métricas de dissimilaridade	26
2.3 Algoritmos de agrupamento	32
2.3.1 Algoritmos particionais	34
2.3.2 Algoritmos hierárquicos	37
2.3.3 Algoritmos baseados em densidades	38
2.4 Índices de validação de agrupamentos	42
2.4.1 Índices de validação interna	43
2.4.2 Índices de validação externa	45
2.5 Conclusões do Capítulo	47

3	Testes em bases de dados <i>benchmark</i>	48
3.1	Testes estatísticos de significância	50
3.2	Primeiro experimento - Métrica de dissimilaridade e algoritmo de agrupamento . . .	51
3.2.1	Algoritmos de agrupamento	51
3.2.2	Métricas de dissimilaridade	52
3.2.3	Comparação entre Métricas Promissoras	53
3.3	Segundo experimento - Índice de validação interno	55
3.3.1	Índice para a definição do número de grupos	56
3.3.2	Índice interno para avaliação de partições com o número de grupos definido .	59
3.4	Conclusões do Capítulo	62
4	Estudo de caso	65
4.1	Metodologia	65
4.2	Agrupamento	66
4.2.1	Estratégia alternativa de agrupamento	68
4.3	Análise dos resultados	69
5	Conclusões e propostas de trabalhos futuros	75
5.1	Propostas de trabalhos futuros	78
	Referências	79
A	Curvas do UCR <i>Time Series</i>	86

Agradecimentos

À minha mãe Maria Tereza pelo incentivo durante toda esta minha empreitada acadêmica assim como pelo exemplo de força e determinação. À minha irmã Sara pela inspiração nos estudos e irmã Débora pelo apoio incondicional durante este desafio. Ao meu cunhado Matheus pelos conselhos e ao meu avô "Bóia".

Ao meu orientador, Prof. Cristiano Leite de Castro, pelos valiosos ensinamentos tanto em sala de aula quanto nas reuniões realizadas em sua sala que guiaram e foram fundamentais para a conclusão deste trabalho, assim como para a escrita de todos os artigos que foram frutos deste. Ao meu coorientador, Prof. Frederico Gadelha Guimarães pelo apoio e orientação e aos demais professores do PPGEe da UFMG.

Aos colegas da graduação em Engenharia Elétrica da UFMG, alguns destes já mestres e doutores que foram inspiração e incentivadores para este meu retorno à Escola de Engenharia. Aos colegas do PPGEe, e em especial aos colegas do laboratório MINDS. Aos meus amigos de infância do Colégio Loyola, apoiadores e incentivadores durante todo o período de elaboração desta dissertação de mestrado.

Finalmente, à Fundação CAPES pelo apoio financeiro provido.

Lista de Ilustrações

2.1	Exemplo de normalização Z. As duas senóides s1 e s2 que possuem comportamento semelhante mas, devido à diferença de <i>offsets</i> e amplitudes, se apresentam significativamente distintas. Após a aplicação da normalização Z em cada uma delas (z-s1 e z-s2), elas são praticamente idênticas, ...	9
2.2	Exemplos de normalização max e min-max.	10
2.3	Exemplo de redução de dimensionalidade pela PAA.	11
2.4	Distância Euclidiana entre as duas curvas é dada pela soma das distâncias euclidianas entre as observações de cada curva no mesmo instante.	13
2.5	Distância Euclidiana entre duas curvas do conjunto de dados ECG200.	15
2.6	DTW sem janela entre duas curvas do conjunto de dados ECG200, na qual foi usada como distância $\delta(x_i, y_j) = (x_i - y_j)^2$	16
2.7	DTW com janela de tamanho 10% do tamanho da série temporal, entre duas curvas do conjunto de dados ECG200, na qual foi usada como distância $\delta(x_i, y_j) = (x_i - y_j)^2$	17
2.8	Duas senóides e uma reta têm suas distâncias comparadas.	18
2.9	Curvas x e y	19
2.10	Curva y envelopada pelas curvas u e l obtidas pela equação 2.2.8.	20
2.11	A raiz do somatório das distâncias em preto fornece a distância LB-Keogh, como descrito pela equação 2.2.9.	20
2.12	Função de sintonia 2.2.22 para diferentes valores de γ	25
2.13	Três curvas diferentes sendo duas complexas e uma simples. A estimativa de complexidade de cada uma bem como as distâncias CID e euclidiana entre elas podem ser vistas na tabela 2.3	27
2.14	À esquerda duas curvas, não normalizadas, que, quando comparadas, se parecem bastante distintas, quando na verdade não o são. À direita um eletrocardiograma, a ser comparado com um sinal de medição contínuo. As duas primeiras comparações têm um bom casamento, mas as subsequentes apresentam valores maiores em uma comparação ponto a ponto devido ao crescente <i>offset</i> que a curva começa a apresentar. Adaptado de (Batista, Wang e E. J. Keogh 2011)	28
2.15	Duas curvas cujo alinhamento ótimo é obtido por meio da DTW. Extraído de (Batista, Wang e E. J. Keogh 2011)	29
2.16	Duas curvas cujo alinhamento ótimo é obtido após a curva ser esticada por um fator $\kappa = 1.41$, que foi obtido por força bruta. Adaptado de (Batista, Wang e E. J. Keogh 2011).	29

2.17	Duas curvas que após sucessivas tentativas de defasamento de uma delas, possuem casamento ótimo. Cada uma delas representa dados de radiação colhidos de uma estrela diferente. Adaptado de (Batista, Wang e E. J. Keogh 2011).	30
2.18	Duas curvas geradas a partir do formato de dois crânios a serem comparados. A falta da região do nariz de um dos crânios pode influenciar significativamente na comparação entre as métricas. Extraído de (Batista, Wang e E. J. Keogh 2011).	31
2.19	Diversas formas geométricas e as séries temporais geradas para cada uma delas. As curvas foram geradas pela distância do ponto central da figura até cada ponto do seu contorno. Extraído de (Batista, Wang e E. J. Keogh 2011).	32
2.20	Agrupamentos hierárquicos com <i>linkagem</i> average realizados nas curvas; à esquerda com a distância euclidiana e à direita com a distância euclidiana corrigida pela CID. Extraído de (Batista, Wang e E. J. Keogh 2011).	33
2.21	Exemplo de um dendrograma resultante do agrupamento hierárquico com a <i>linkagem</i> average e métrica de dissimilaridade euclidiana para o conjunto de dados Iris.	38
2.22	Conjunto de dados sintético formado por dois grupos.	40
2.23	Gráfico <i>k-dist</i> do conjunto de dados da figura 2.22, onde o crescimento abrupto ocorre para a distância 0.086 aproximadamente.	41
2.24	Partição dos dados da figura 2.22 obtida pelo DBSCAN com <i>MinPts</i> =3 e <i>Eps</i> =0.086. O pontos verdes foram considerados <i>outliers</i> pelo algoritmo.	42
3.1	Resultado do teste de Bonferroni-Dunn após o ranqueamento dos valores de VI. A estratégia de controle se encontra centrada no ranking médio da estratégia <i>k-means(random)</i> com distância Euclidiana.	52
3.2	Resultado teste Bonferroni-Dunn após o ranqueamento dos valores de VI, centrado no ranking médio da estratégia padrão (<i>k-means(random)</i> com distância Euclidiana).	53
3.3	Resultado teste Nemenyi após o ranqueamento dos valores de VI.	54
3.4	Resultado teste Nemenyi após o ranqueamento dos valores de tempo de processamento.	55
3.5	Resultado do teste de Nemenyi para o teste do melhor índice interno para apontar o número de grupos.	58
3.6	Resultado do teste de Nemenyi para o teste de coerência dos índices internos com o VI.	63
4.1	Formação dos 6 subconjuntos de dados a partir dos perfis diários.	67
4.2	Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e no verão. Os pontos em azul, na Figura 4.2a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo.	70
4.3	Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e no inverno. Os pontos em azul, na Figura 4.3a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo.	70

4.4	Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e nas estações de transição. Os pontos em azul, na Figura 4.4a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo	71
4.5	Resultado final para os curvas diárias das cargas medidas nos dias de semana e no verão. Os pontos em azul, na Figura 4.5a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo	71
4.6	Resultado final para os curvas diárias das cargas medidas nos dias de semana e no inverno. Os pontos em azul, na Figura 4.6a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo	72
4.7	Resultado final para os curvas diárias das cargas medidas nos dias de semana e na transição. Os pontos em azul, na Figura 4.7a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo	72
A.1	Conjunto de dados '50words'.	87
A.2	Conjunto de dados 'Adiac'.	88
A.3	Conjunto de dados 'ArrowHead'.	89
A.4	Conjunto de dados 'Beef'.	90
A.5	Conjunto de dados 'BeetleFly'.	91
A.6	Conjunto de dados 'BirdChicken'.	92
A.7	Conjunto de dados 'CBF'.	93
A.8	Conjunto de dados 'Car'.	94
A.9	Conjunto de dados 'ChlorineConcentration'.	95
A.10	Conjunto de dados 'CinC_ECG_torso'.	96
A.11	Conjunto de dados 'Coffee'.	97
A.12	Conjunto de dados 'Computers'.	98
A.13	Conjunto de dados 'Cricket_X'.	99
A.14	Conjunto de dados 'Cricket_Y'.	100
A.15	Conjunto de dados 'Cricket_Z'.	101
A.16	Conjunto de dados 'DiatomSizeReduction'.	102
A.17	Conjunto de dados 'DistalPhalanxOutlineAgeGroup'.	103
A.18	Conjunto de dados 'DistalPhalanxOutlineCorrect'.	104
A.19	Conjunto de dados 'DistalPhalanxTW'.	105
A.20	Conjunto de dados 'ECG200'.	106
A.21	Conjunto de dados 'ECG5000'.	107
A.22	Conjunto de dados 'ECGFiveDays'.	108
A.23	Conjunto de dados 'Earthquakes'.	109
A.24	Conjunto de dados 'ElectricDevices'.	110
A.25	Conjunto de dados 'FISH'.	111

A.26 Conjunto de datos 'FaceAll'.	112
A.27 Conjunto de datos 'FaceFour'.	113
A.28 Conjunto de datos 'FacesUCR'.	114
A.29 Conjunto de datos 'FordA'.	115
A.30 Conjunto de datos 'FordB'.	116
A.31 Conjunto de datos 'Gun_Point'.	117
A.32 Conjunto de datos 'Ham'.	118
A.33 Conjunto de datos 'HandOutlines'.	119
A.34 Conjunto de datos 'Haptics'.	120
A.35 Conjunto de datos 'Herring'.	121
A.36 Conjunto de datos 'InlineSkate'.	122
A.37 Conjunto de datos 'InsectWingbeatSound'.	123
A.38 Conjunto de datos 'ItalyPowerDemand'.	124
A.39 Conjunto de datos 'LargeKitchenAppliances'.	125
A.40 Conjunto de datos 'Lighting2'.	126
A.41 Conjunto de datos 'Lighting7'.	127
A.42 Conjunto de datos 'MALLAT'.	128
A.43 Conjunto de datos 'Meat'.	129
A.44 Conjunto de datos 'MedicalImages'.	130
A.45 Conjunto de datos 'MiddlePhalanxOutlineAgeGroup'.	131
A.46 Conjunto de datos 'MiddlePhalanxOutlineCorrect'.	132
A.47 Conjunto de datos 'MiddlePhalanxTW'.	133
A.48 Conjunto de datos 'MoteStrain'.	134
A.49 Conjunto de datos 'NonInvasiveFetalECG_Thorax1'.	135
A.50 Conjunto de datos 'NonInvasiveFetalECG_Thorax2'.	136
A.51 Conjunto de datos 'OSULeaf'.	137
A.52 Conjunto de datos 'OliveOil'.	138
A.53 Conjunto de datos 'PhalangesOutlinesCorrect'.	139
A.54 Conjunto de datos 'Phoneme'.	140
A.55 Conjunto de datos 'Plane'.	141
A.56 Conjunto de datos 'ProximalPhalanxOutlineAgeGroup'.	142
A.57 Conjunto de datos 'ProximalPhalanxOutlineCorrect'.	143
A.58 Conjunto de datos 'ProximalPhalanxTW'.	144
A.59 Conjunto de datos 'RefrigerationDevices'.	145
A.60 Conjunto de datos 'ScreenType'.	146
A.61 Conjunto de datos 'ShapeletSim'.	147
A.62 Conjunto de datos 'ShapesAll'.	148
A.63 Conjunto de datos 'SmallKitchenAppliances'.	149
A.64 Conjunto de datos 'SonyAIBORobotSurface'.	150
A.65 Conjunto de datos 'SonyAIBORobotSurfaceII'.	151
A.66 Conjunto de datos 'StarLightCurves'.	152
A.67 Conjunto de datos 'Strawberry'.	153
A.68 Conjunto de datos 'SwedishLeaf'.	154

A.69 Conjunto de datos 'Symbols'	155
A.70 Conjunto de datos 'ToeSegmentation1'	156
A.71 Conjunto de datos 'ToeSegmentation2'	157
A.72 Conjunto de datos 'Trace'	158
A.73 Conjunto de datos 'TwoLeadECG'	159
A.74 Conjunto de datos 'Two_Patterns'	160
A.75 Conjunto de datos 'UWaveGestureLibraryAll'	161
A.76 Conjunto de datos 'Wine'	162
A.77 Conjunto de datos 'WordsSynonyms'	163
A.78 Conjunto de datos 'Worms'	164
A.79 Conjunto de datos 'WormsTwoClass'	165
A.80 Conjunto de datos 'synthetic_control'	166
A.81 Conjunto de datos 'uWaveGestureLibrary_X'	167
A.82 Conjunto de datos 'uWaveGestureLibrary_Y'	168
A.83 Conjunto de datos 'uWaveGestureLibrary_Z'	169
A.84 Conjunto de datos 'wafer'	170
A.85 Conjunto de datos 'yoga'	171

Lista de Tabelas

2.1	Distâncias Minkowski entre as curvas da figura 2.4 para diversos valores de p .	13
2.2	Distâncias euclidiana e DTW entre as curvas da figura 2.8.	18
2.3	Medida de complexidade, como definida na equação 2.2.25, das curvas da figura 2.13 e distâncias euclidiana e CID-euclidiana entre elas.	26
2.4	Sumário das métricas de dissimilaridades discutidas em 2.2.	32
3.1	Estratégias comparadas no experimento 1.	52
3.2	Estratégias comparadas no experimento 2.	53
3.3	Estratégias do experimento 3.	54
3.4	Principais índices internos de validação disponíveis na literatura	55
3.5	Resultados dos experimentos iterativos para o conjunto de dados <i>Herring</i> , cujo número de grupos é igual a 2. Os valores em negrito representam os melhores valores obtidos em cada índice.	57
3.6	Resultados dos experimentos iterativos para cada conjunto de dados, onde o asterisco corresponde à um sucesso na sugestão do índice para a base de dados. Foi utilizado o algoritmo hierárquico com linkagem <i>average</i> e métrica de dissimilaridade cid-euclidiana.	57
3.7	Rankings dos valores dos índices Para cada uma das partições obtidas variando-se o algoritmo de agrupamento.	60
3.8	Rankings dos valores dos índices da tabela 3.7. Note que valores maiores para o Calinski-Harabasz e o silhouette implicam em rankings melhores, ao passo que para o VI em rankings piores.	60
3.9	Tabela com os valores de Kendall-Tau dos índices em relação ao VI.	60
4.1	Tabela que apresenta as informações das partições escolhidas para cada um dos seis sub-conjuntos gerados.	69

Lista de Abreviaturas e Siglas

REI Redes Eléctricas Inteligentes

MI Medidores Inteligentes

SVD *Singular Value Decomposition* (Decomposição em valores singulares)

DFT *Discrete Fourier Transform* (Transformada discreta de Fourier)

DBI *Davies-Bouldin Index*

DTW *Dynamic Time Warping*

EDR *Edit distance on real sequence*

RAM *Random Access Memory*, ou memória de acesso aleatória, que é a memória principal da maioria dos sistemas computacionais.

DBI *Davies-Bouldin index*, ou índice Davies-Bouldin, que é um índice de validação interna de grupos.

CID *Complex Invariant Distance*

VI *Variation of Information*, ou índice de variação da informação, que é um índice de validação externa de grupos.

ANEEL Agência Nacional de Energia Elétrica

Lista de Símbolos

L_1 Norma L_1 , distância *Manhattan*, ou distância *cityblock*.

L_2 Norma L_2 ou distância Euclidiana.

L_∞ Norma L_∞ ou distância Chebyshev.

$dist(\mathbf{a}, \mathbf{b})$ Métrica de dissimilaridade entre as séries temporais $\mathbf{a}$ e $\mathbf{b}$.

Lista de Algoritmos

2.1	Algoritmo <i>k-means</i> .	35
2.2	Algoritmo <i>k-medoids</i> .	36
2.3	Agrupamento aglomerativo.	39

Capítulo 1

Introdução

Ao conjunto de tecnologias que visam modernizar o sistema elétrico de potência, dá-se o nome de Redes Elétricas Inteligentes (*Smart Grids*), doravante REI. Tal conjunto de tecnologias faz uso de sistemas de medição, computação e comunicações digitais de forma a gerar um fluxo de informações bidirecional entre o fornecedor do sistema de energia e o cliente final. Os maiores impactos da adoção de tais tecnologias ocorrerão na distribuição de energia elétrica e nos pequenos consumidores. Pelo número crescente de investimentos em pesquisa de REI em todo mundo, percebe-se que esta se apresenta como uma forte tendência de modernização dos sistemas elétricos de potência em diversos países (Centro de Gestão e Estudos Estratégicos 2012).

Uma das bases dessa nova tecnologia é a instalação de sensores, conhecidos como *smart meters* ou medidores inteligentes (MI), em diversos pontos da rede, de forma a se obter dados relevantes da mesma. Com a instalação desses sensores em cada consumidor, seja ele residencial, comercial ou industrial, será possível medir periodicamente a potência consumida por ele. De posse dos dados gerados pelos medidores inteligentes, os consumidores podem ser classificados por padrões de consumo e a partir dessa classificação espera-se obter, por exemplo, as seguintes vantagens:

- Realização de uma nova política de tarifas diferenciadas por tipo de consumidor residencial.
- Aprimoramento das técnicas de previsão de demanda por parte dos fornecedores de energia.
- Auxílio dos consumidores na análise do seu consumo de energia para a elaboração de estratégias de redução deste.
- Fomento de um mercado competitivo de fornecimento de energia.
- Identificação automática de falhas e fraudes.

De forma geral, a análise inteligente de dados extraídos dos *smart meters* permitirá às companhias o desenvolvimento de novas capacidades em termos de gerenciamento e planejamento da rede, bem como novos serviços direcionados ao consumidor visando a eficiência energética (Kai-le Zhou 2013).

O grande volume de dados gerados pelos *smart meters* demandarão algoritmos e estratégias capazes de extrair informação útil dessa grande massa de dados. A tarefa de se encontrar sub-grupos em determinado conjunto de dados baseando-se somente nas características dos elementos

do conjunto é denominada agrupamento, que, do ponto de vista da Inteligência Computacional, faz parte do paradigma de aprendizado não-supervisionado. Ou seja, o agrupamento se caracteriza pela necessidade de se formar grupos dos dados, de forma que os dados são mais semelhantes entre si quando contidos no mesmo grupo, ao passo que estes são mais díspares possível dos dados pertencentes a outros grupos.

Por não-supervisionado, entende-se que a tarefa de agrupamento se dará apenas a partir das características dos dados, e que não existe rotulação dos mesmos, ou seja, não se sabe a priori à qual classe, ou grupo, os dados pertencem e, em certos problemas, não se sabe também o número de grupos nos quais os dados se encontram naturalmente divididos.

O problema de se dividir as curvas de carga geradas por MI em grupos é um problema de agrupamento de séries temporais, pois na maioria das aplicações, os dados de consumo são medidos em intervalos regulares, caracterizando, assim, uma série temporal. Ademais, não sabemos de antemão a qual grupo cada carga pertence e nem o número de grupos que contêm os dados, logo devido à ausência de dados rotulados, constitui-se como uma tarefa de agrupamento.

1.1 Objetivos e contribuições

O objetivo geral desta dissertação de mestrado é de investigar as técnicas consideradas estado da arte para o agrupamento de séries temporais, visando à proposta de uma metodologia para agrupamento de curvas de carga de consumidores residenciais. Tal investigação envolveu o estudo teórico-empírico de diferentes formas de normalização de dados, métodos de agrupamento, medidas de dissimilaridade e índices de validação.

A Agência Nacional de Energia Elétrica, ANEEL, em uma resolução de 2010 (Agência Nacional de Energia Elétrica 2010), definiu os critérios de divisão tarifária dos seus consumidores. A primeira divisão que é feita, é baseada na tensão de atendimentos dos consumidores que são divididos em:

- grupo A: tensão de atendimento superior à 2,3 kV, que pode ainda ser subdividido em:
 - subgrupo A1: atendimento em tensão igual ou superior à 230 kV,
 - subgrupo A2: atendimento em tensão de 88 kV a 130 kV,
 - subgrupo A3: atendimento em tensão de 69 kV,
 - subgrupo A4: atendimento em tensão de 2,3 kV a 44 kV,
 - subgrupo AS: atendimento em tensão inferior a 2,3 kV mas por sistema subterrâneo,
- e grupo B: tensão de atendimento igual ou inferior à 2,3kV, que também é subdividido em:
 - subgrupo B1: atendimento residencial,
 - subgrupo B2: atendimento rural,
 - subgrupo B3: atendimento das demais classes,
 - subgrupo B4: atendimento da iluminação pública.

Existem ainda, mais subdivisões, para alguns dos subgrupos do grupo B, e especificamente o subgrupo B1, pode ser dividido, ainda, em:

- atendimento residencial,
- e atendimento residencial de Baixa Renda -TSEE (Tarifa Social de Energia Elétrica).

Esta divisão implica em um cálculo tarifário diferenciado para as famílias de baixa renda, e seus critérios baseiam-se não nos perfis de curva de carga, mas sim em fatores de renda e de consumo, como pode ser visto em (Agência Nacional de Energia Elétrica 2011).

O cálculo tarifário das cargas do grupo A, que são aquelas que demandam uma maior potência, sendo em geral, cargas industriais e comerciais, possuem um cálculo tarifário bem mais complexo do que o cálculo tarifário residencial. As empresas podem contratar uma potência mínima das concessionárias de energia, pagando multas, inclusive, se o seu consumo ultrapassar o valor contratado. Além deste fator, para as cargas do grupo A, o cálculo tarifário varia de acordo com o mês e a hora do dia de consumo, o que motiva as empresas a fazerem um uso mais racional da energia.

Dessa maneira, o objetivo deste trabalho é a realização da proposição de uma técnica que gere uma segmentação maior dos consumidores residenciais, baseada em perfis de consumo e não em critérios de renda, de forma a ser possível a proposição de uma política tarifária diferenciada para os consumidores residenciais, assim como já é feito atualmente para os consumidores do grupo A.

Seria extremamente interessante que os dados a serem utilizados durante o trabalho fossem o mais representativos possível à realidade brasileira, ou seja, sem as influências da grande variação de temperatura que ocorre nos países do hemisfério norte, que é onde se encontra mais avançada a pesquisa e implantação de REI e, conseqüentemente, de instalação de MI e a disponibilidade de dados para pesquisa. Em (Centro de Gestão e Estudos Estratégicos 2012) é feita uma menção à um projeto da CEMIG em Sete Lagoas denominado "Cidade do Futuro". Tal projeto é reconhecido como pioneiro no Brasil no que se refere à pesquisa e instalação de equipamentos de REI no atual sistema elétrico de potência brasileiro. No entanto, estes dados ainda não são de domínio público. Dessa maneira, neste trabalho, optou-se por utilizar dados de cargas residenciais da Austrália, obtidos de (*Smart-Grid SmartCity Customer Trial data* 2016).

Os dados consistem de cerca de três anos e meio de medições do consumo de aproximadamente 13000 residências australianas, com uma taxa de amostragem de trinta minutos. Por questões de privacidade, somente as curvas de demanda foram disponibilizadas, e outros dados que poderiam ser úteis na tarefa de agrupamento como: localização das residências, número e situação econômica dos moradores, entre outros, não foram fornecidos. Dessa maneira, a única informação disponível para a realização do agrupamento das cargas são as curvas de demanda. Existem dados inconsistentes e incompletos o que demandou uma fase de pré-processamento dos mesmos, no qual o número de curvas a serem agrupadas foi reduzido para 427.

Para a realização do agrupamento de curvas de carga fazem-se necessárias as escolhas de um tipo de normalização, de uma métrica de dissimilaridade para se comparar as cargas, de um algoritmo de agrupamento para se obter a partição dos dados, e de um índice de validação que indique a qualidade da partição obtida.

A etapa de normalização tem como objetivos principais reduzir a dimensionalidade das séries temporais e facilitar a comparação entre séries temporais que possuam amplitudes ou *offsets* muito discrepantes. As abordagens mais comuns para a redução de dimensionalidade consistem na *Piecewise aggregate approximation* (Lin et al. 2003) ou na transformada discreta de Fourier (Agrawal, Faloutsos e Swami 1993). Já a eliminação de distorções, tais como amplitude e *offset*, pode ser realizada por meio da normalização Z (Rakthanmanon et al. 2012), ou por meio da normalização *max* como utilizada em (Chicco 2012). Em geral, em problemas de agrupamento de cargas é utilizada a normalização *max* sem nenhum tipo de redução de dimensionalidade, como pode ser visto em (Chicco 2012; Flath et al. 2012; Tsekouras et al. 2008), no entanto, outras abordagens foram testadas.

Uma vez definido a normalização a ser aplicada nos dados, surge a necessidade de se escolher a medida de dissimilaridade a ser utilizada na comparação entre as cargas. A distância euclidiana se apresenta onipresente em estudos de agrupamento de cargas, no entanto, em outras aplicações como, por exemplo, na classificação de séries temporais, medidas de dissimilaridades como a DTW (Berndt e Clifford 1994), corrigidas por métodos de invariância à complexidade (Batista, Wang e E. J. Keogh 2011), têm obtido desempenho superior à distância euclidiana (Giusti e Batista 2013). Isso motivou o presente estudo a verificar a aplicabilidade de tais métricas ao problema de agrupamento de séries temporais em geral e de agrupamento de curvas de carga em específico.

Após o estabelecimento de uma métrica de dissimilaridade para a comparação das cargas, faz-se necessária a escolha de um algoritmo de agrupamento. O algoritmo mais utilizado em tarefas de agrupamento de cargas é o *k-means*, devido à sua simplicidade e praticidade. No entanto, ele possui a deficiência de fazer uso de somente a distância euclidiana (Rajaraman e Ullman 2011, p. 254), o que descarta potenciais ganhos a serem obtidos com a escolha de outras medidas de dissimilaridades. O *k-means* assume que o formato dos grupos é do tipo hiper-esfera (esférico), o que pode não ser verdade para dados multidimensionais como dados de séries temporais. Ademais, existem estudos que mostram que outros algoritmos de agrupamento, como os algoritmos hierárquicos, possuem desempenho superior aos algoritmos particionais, como o *k-means*, no contexto de séries temporais (Paparrizos e Gravano 2016).

A partição dos dados a partir da aplicação do algoritmo escolhido, deve ser avaliada por um índice que indique a qualidade da mesma. Tal índice é útil, também, para comparar os resultados obtidos por diferentes escolhas de normalização, métrica de dissimilaridade e algoritmo de agrupamento. Não existe um índice que seja sistematicamente empregado no agrupamento de curvas de carga, e dessa maneira, os principais índices de validação de agrupamento são discutidos e comparados neste trabalho.

Sendo assim, devido a multiplicidade de caminhos possíveis para a realização da tarefa de agrupamento de séries temporais, a primeira fase do estudo focou em analisar, por meio de conjuntos de dados *benchmark* (rotulados) de séries temporais, uma gama de metodologias de agrupamento propostas na literatura com o intuito de se obter um direcionamento para a aplicação nos dados reais australianos.

Além da tarefa de agrupamento em si, os tempos e custos, no que diz respeito à utilização de processamento, também são objetos de estudo desta pesquisa. Assim, espera-se que este trabalho auxilie no dimensionamento de um sistema computacional a ser utilizado pelas companhias de distribuição de energia elétrica para o agrupamento de curvas de cargas obtidas por MI.

Dentre as principais contribuições desta dissertação, pode-se citar:

- Extensa revisão bibliográfica acerca do problema e das várias opções para a realização de agrupamento de séries temporais.
- Proposição de uma metodologia genérica para a realização de agrupamento de séries temporais embasada em dados empíricos de experimentos realizados com bases de dados rotuladas.
- Proposição de uma metodologia específica para a realização de agrupamento de curvas de cargas residenciais.

1.2 Visão Geral do Texto

O restante do texto desta dissertação de mestrado se apresenta da seguinte maneira:

O Capítulo 2 apresenta uma extensa revisão bibliográfica acerca dos aspectos relevantes durante a tarefa de agrupamento de séries temporais. Nele, são destacadas as principais métricas de dissimilaridade, bem como os principais algoritmos de agrupamento para a realização do agrupamento. Ao final deste capítulo, também são discutidos os métodos de avaliação dos resultados do agrupamento.

Em seguida, no Capítulo 3, são realizados testes em conjuntos de dados rotulados (*benchmark*), com a finalidade de se avaliar os algoritmos de agrupamento, métricas de dissimilaridades e métodos de avaliação discutidos no Capítulo 2. Neste terceiro capítulo, foram realizados três experimentos com a finalidade de se definir a estratégia de agrupamento a ser empregada no estudo de caso com dados reais. O primeiro destes experimentos teve como objetivo definir o algoritmo de agrupamento a ser empregado no referido estudo de caso. Já o segundo procurou definir a métrica de dissimilaridade, enquanto o terceiro visou apontar a melhor maneira de se avaliar o resultado do agrupamento. Em todos os experimentos foram realizados testes estatísticos de significância para se definir a melhor estratégia a ser empregada no agrupamento da base de dados australiana.

Após as definições obtidas a partir dos testes realizados no Capítulo 3, o estudo de caso em si, que consiste no agrupamento das curvas de carga das residências australianas, é descrito no Capítulo 4. Finalmente, no Capítulo 5, são apresentadas as conclusões gerais do trabalho bem como algumas das propostas para sua continuidade.

Capítulo 2

Agrupamento de séries temporais

Por agrupamento, entende-se a tarefa de dividir um conjunto de dados em grupos homogêneos e bem definidos. A partir de determinada métrica de dissimilaridade, o agrupamento busca formar grupos que contêm dados que possuem baixa dissimilaridade entre si, ao passo que estes possuem alta dissimilaridade com os dados contidos nos demais grupos. O agrupamento se faz necessário quando não se possui, *a priori*, a rotulação dos dados, ou seja, não se sabe a qual grupo cada instância pertence, e em alguns casos, não se sabe até mesmo o número de grupos que as instâncias estão naturalmente contidas. A partir da realização de tal tarefa, espera-se adquirir conhecimento acerca da estrutura dos dados bem como extrair informação útil dos mesmos.

O agrupamento pode ser feito em dados textuais, categóricos, binários, numéricos, ordinais, relacionais, temporais, espaço-temporais, ou qualquer combinação destes. Quando o agrupamento é realizado em dados nos quais a observação das grandezas de interesse não variam com o tempo, os dados são considerados estáticos, e caso contrário os dados são ditos dinâmicos. As séries temporais, que são objeto de estudo deste trabalho, são dados dinâmicos. Grande parte da pesquisa de agrupamento se dá na análise de dados estáticos, no entanto, existe uma tendência de crescimento no estudo em séries temporais devido às diversas aplicações nas engenharias, mercado financeiro, marketing, medicina, biologia, meteorologia, astronomia entre outros (Warren Liao 2005).

Formalmente, uma partição $\mathcal{C}$ é a partição de um conjunto de dados X composto por n instâncias, e cada instância possuindo d características, em k sub-conjuntos mutualmente disjuntos, $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$. A esses sub-conjuntos dá-se o nome de grupos. Dessa maneira,

$$\mathcal{C} = \mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k, \text{ tal que } \mathcal{C}_k \cap \mathcal{C}_l = \emptyset \quad (2.0.1)$$

Uma série temporal é definida como um conjunto de observações sequenciais de uma grandeza de interesse, assim, ela é um tipo de dados com valores reais e de alta dimensionalidade. Denomina-se de tamanho da série temporal o número de observações ou amostras que ela contém.

As seções a seguir deste capítulo descrevem as principais estratégias de agrupamento de séries temporais. Por estratégia entende-se a escolha de:

- um pré-processamento dos dados (normalização),
- uma métrica de dissimilaridade para se comparar as instâncias,

- um algoritmo de agrupamento para a realização do agrupamento em si,
- e, finalmente, um índice de validação para avaliar a qualidade da partição resultante.

Este capítulo se encontra estruturado de forma a realizar um estudo de cada um destes aspectos. Inicialmente, na primeira seção, são analisadas as opções de pré-processamento dos dados, que consistem, basicamente em normalizar, e reduzir o tamanho das séries temporais.

Em seguida, são analisadas as métricas de dissimilaridade de séries temporais que podem ser divididas em: métricas baseadas em distâncias ou dissimilaridades, aplicadas no espectro das séries temporais e em modelos estatísticos (Warren Liao 2005). Pelo fato da escolha de uma métrica de dissimilaridade ser um dos aspectos centrais na tarefa de agrupamento de séries temporais as principais métricas apresentadas na literatura são discutidas na maior parte deste capítulo.

A escolha da métrica de dissimilaridade se dá pela definição de um critério, ou métrica, que nos diga o quão similares duas instâncias são. Tal métrica é dependente do objetivo principal da tarefa de agrupamento. Existem determinadas aplicações que realizam o agrupamento baseado no valor da predição da série temporal, ou seja, as curvas são agrupadas em função do próximo valor esperado de cada série, enquanto em outras aplicações, quando se deseja comparar o formato das séries, o uso de métricas ponto a ponto se adequa melhor. Existem ainda outras abordagens que buscam agrupar curvas que possuam o mesmo comportamento ou tendência, ou seja, curvas que se mantêm constante, aumentam ou diminuem o seu valor em períodos de tempo próximos.

Após a discussão das métricas, serão expostos os principais algoritmos de agrupamento de séries temporais disponíveis na literatura. Estes algoritmos podem ser estocásticos como o *k-means*, *k-medoids*, ou de agrupamento hierárquico com diferentes tipos de *linkagem*, tais como: a *average*, *complete* ou *ward*. Existem também os algoritmos de agrupamento baseados em densidade, como o *DBSCAN*, que possuem a vantagem de, não ser necessária a definição, *a priori*, do número de grupos.

Finalmente, na última seção do capítulo, são apresentados os principais índices de validação de agrupamentos disponíveis na literatura. Tais índices são essenciais para se mensurar a qualidade das partições dos dados resultantes dos algoritmos de agrupamento.

Tal revisão bibliográfica é importante para a definição da melhor estratégia, ou seja, da escolha de uma normalização, métrica de dissimilaridade, algoritmo de agrupamento e índice de validação das partições resultantes a ser empregada no agrupamento das curvas de carga.

2.1 Pré-processamento de séries temporais

Antes da tarefa de agrupamento em si, recomenda-se a realização de uma fase de pré-processamento dos dados, nos quais eles são normalizados. A normalização é utilizada para corrigir possíveis distorções que os dados brutos podem apresentar devido ao seu processo de aquisição. Pode-se corrigir distorções relacionadas à escala, translação, rotação, diferença de fase, entre outras.

Além da normalização, métricas de dissimilaridade podem ser aplicadas no espectro das séries temporais, como uma estratégia de redução de dimensionalidade, para a posterior realização da tarefa de agrupamento. Tal redução é desejada, pela ocorrência do "mal da dimensionalidade", que basicamente nos diz que quando o número de dimensões de cada instância de um conjunto de dados

é muito grande, as instâncias tendem a se encontrar espaçadas e equidistantes, fazendo com que seja necessário um número muito grande de instâncias para a construção de uma fronteira de decisão (Duda, Hart e Stork 2000). Dessa maneira, faz-se necessária uma etapa de pré-processamento, e as principais abordagens se encontram detalhadas a seguir.

2.1.1 Normalização

Estudos demonstram que qualquer tarefa de mineração de dados de séries temporais sem algum tipo de normalização é sem sentido (E. Keogh e Kasetty 2003). O tipo de normalização mais comum no contexto de séries temporais é realizado por meio da normalização Z (Rakthanmanon et al. 2012), na qual, a média μ e o desvio padrão σ de cada série temporal de tamanho n é calculada e é feita a seguinte transformação em cada medição da série $\mathbf{x}$

$$\hat{x}_i = \frac{x_i - \mu}{\sigma}, i = 1, \dots, n. \quad (2.1.1)$$

No contexto de séries temporais, a normalização Z é extremamente importante para remover certas imperfeições e diferenças de escala que surgem a partir da geração das séries temporais, sejam estas obtidas por meio de medições ou de algum tipo de transformação de dados. Na figura 2.1 podem ser vistas duas senoides, $s1$ e $s2$, ligeiramente defasadas mas com diferença significativa em amplitude e valor médio, ou *offset*. Na mesma figura, as suas respectivas versões normalizadas podem ser vistas pelas linhas tracejadas. É notável que as curvas tracejadas são bem mais semelhantes entre si, e caso as diferenças de amplitude e valor médio não sejam relevantes para a tarefa de agrupamento ou classificação, o que é geralmente o caso, então a normalização Z é uma etapa de pré-processamento recomendada. Em (Rakthanmanon et al. 2012) é mencionado que em tarefas de classificação com séries temporais não normalizadas e Z-normalizadas, obteve-se uma taxa de erro 50% maior no primeiro caso em relação ao segundo quando é inserido um *offset* de apenas $\pm 5\%$.

Além da normalização Z, podem ser utilizadas também as normalizações max e min-max, nas quais devem ser aplicadas, respectivamente, as seguintes transformações em cada uma das n medições da série $\mathbf{x}$:

$$\hat{x}_i = \frac{x_i}{\max(\mathbf{x})}, i = 1, \dots, n \text{ e} \quad (2.1.2)$$

$$\hat{x}_i = \frac{x_i - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})}, i = 1, \dots, n. \quad (2.1.3)$$

A primeira é utilizada, em geral, quando os valores observados das séries temporais que compõem o conjunto de dados são sempre positivos, e confina a série no intervalo $[0, 1]$ dividindo cada valor pelo maior valor observado em toda série. Já a segunda, também confina a série no intervalo $[0, 1]$, mas necessariamente seu menor valor se torna 0, o que nem sempre ocorre na primeira. Os efeitos de ambas normalizações podem ser vistos na figura 2.2. Como os valores de medição dos medidores inteligentes são sempre positivos, a normalização max é muito utilizada na mineração de dados de Redes Elétricas Inteligentes (Chicco 2012).

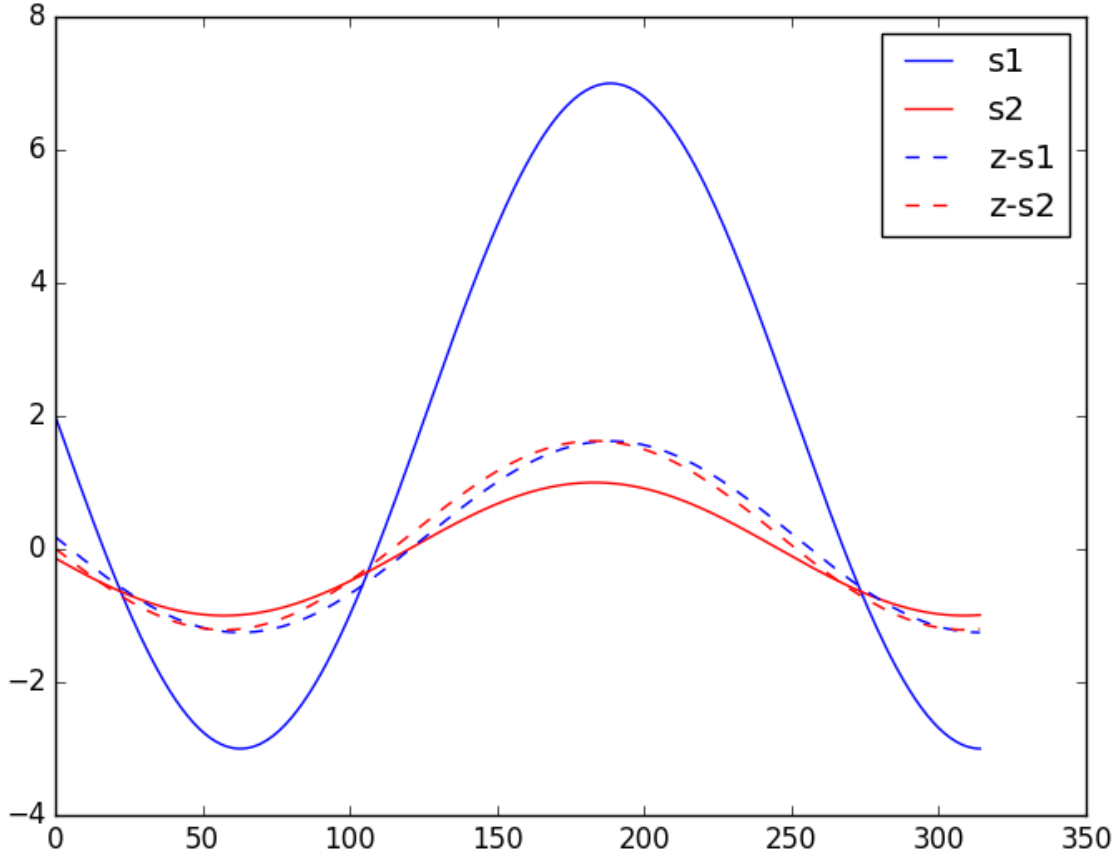


Figura 2.1: Exemplo de normalização Z. As duas senóides s1 e s2 que possuem comportamento semelhante mas, devido à diferença de *offsets* e amplitudes, se apresentam significativamente distintas. Após a aplicação da normalização Z em cada uma delas (z-s1 e z-s2), elas são praticamente idênticas,

2.1.2 *Piecewise aggregate approximation* - PAA

A *Piecewise aggregate approximation* é uma técnica de redução de dimensionalidade na qual, basicamente, a série temporal é dividida em w janelas de tamanhos iguais e o valor da série neste intervalo é definido como o valor médio da série no intervalo compreendido pela janela (Lin et al. 2003). Assim, dada uma série temporal $\mathbf{x} = x_1, x_2, \dots, x_n$ de tamanho n , esta pode ser representada em um espaço w dimensional, sendo $w < n$ e w múltiplo inteiro de n , por um vetor $\bar{\mathbf{x}} = \bar{x}_1, \bar{x}_2, \dots, \bar{x}_w$, tal que seu i -ésimo elemento é dado por

$$\bar{x}_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} x_j. \quad (2.1.4)$$

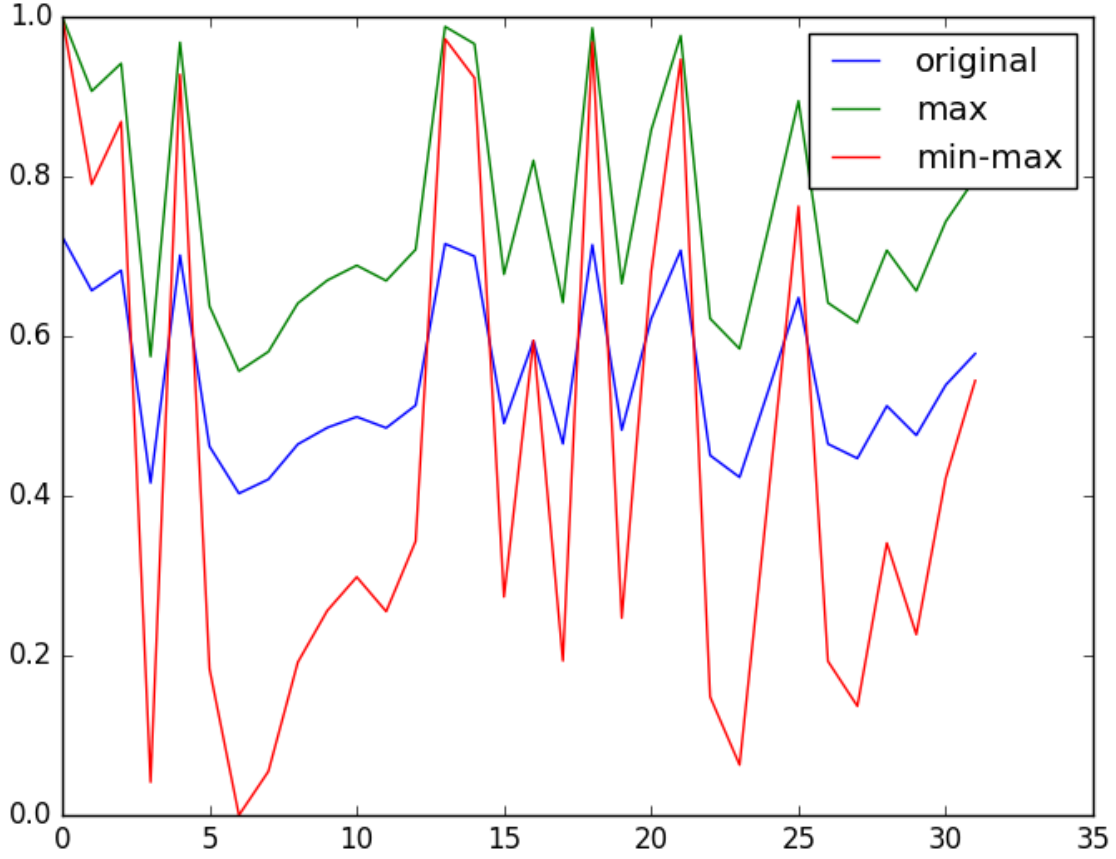


Figura 2.2: Exemplos de normalização max e min-max.

Na figura 2.3 pode ser vista a redução de dimensionalidade da senoide em azul formada por 315 pontos para a curva em vermelho, formada por 19 pontos. Em (Novák et al. 2013) é sugerida uma técnica de redução de dimensionalidade de séries temporais que pode ser vista como uma versão fuzzy da PAA. Basicamente, a ideia é definir uma partição fuzzy para o espaço de entrada e definir um coeficiente para cada valor da partição.

2.1.3 Transformada discreta de Fourier - DFT

Ao se calcular a série de Fourier para os vetores $\mathbf{x}$ e $\mathbf{y}$, a distância entre os dois vetores pode ser obtida pelo cálculo de qualquer umas das métricas de distância a serem discutidas, como a distância euclidiana, entre os coeficientes da série de Fourier discreta truncada de cada vetor

$$dist(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{\theta} (\hat{x}_i - \hat{y}_i)^2 \right)^{\frac{1}{2}}, \quad (2.1.5)$$

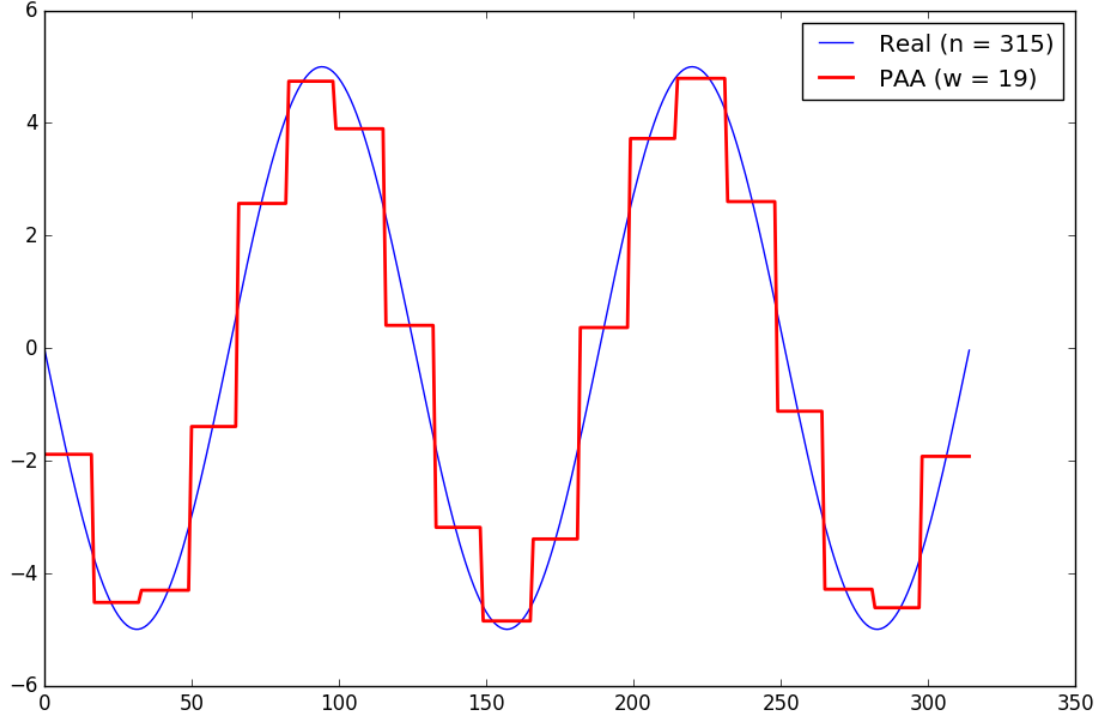


Figura 2.3: Exemplo de redução de dimensionalidade pela PAA.

onde θ é o número de coeficientes complexos a serem utilizados, e $\hat{x}_i$ e $\hat{y}_i$ são os i -ésimos termos da transformada discreta de Fourier, ou *Discrete Fourier Transform* (DFT), dos vetores $\mathbf{x}$ e $\mathbf{y}$ respectivamente.

Segundo (Agrawal, Faloutsos e Swami 1993), são necessários apenas 5 coeficientes ou menos para uma boa aproximação das curvas no domínio da frequência, e a distância euclidiana no domínio da frequência é análoga à distância euclidiana no domínio do tempo pelo teorema de *Parseval*, sendo, portanto, a métrica de distância mais indicada.

O método proposto tem a vantagem de realizar uma redução significativa de dimensionalidade, já que uma série temporal com 300 pontos, por exemplo, pode ser reduzida para uma análise em 5 pontos. Além disso, quando existe a presença de ruídos de alta frequência no sinal, e tais ruídos não são relevantes para o agrupamento, tal abordagem acarreta em resultados melhores.

2.2 Métricas de dissimilaridade

Existem diversas métricas de dissimilaridade para comparar duas séries temporais, e elas divergem entre si principalmente no que diz respeito aos custos computacionais, sensibilidade à ruídos, invariância ao defasamento entre as curvas, capacidade de ser utilizada para séries temporais com

diferentes tamanhos, entre outros. Além disso, as métricas podem ser divididas em dois grandes grupos, sendo o primeiro formado pelas métricas que podem ser consideradas distâncias, e aquelas que não podem. Para determinada métrica ser considerada uma distância ela deve possuir, necessariamente, as seguintes propriedades para vetores $\mathbf{x}, \mathbf{y}$ e $\mathbf{z}$ (Duda, Hart e Stork 2000):

- não-negatividade:

$$dist(\mathbf{x}, \mathbf{y}) > 0, \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n, \quad (2.2.1)$$

- reflexividade:

$$dist(\mathbf{x}, \mathbf{y}) = 0, \text{ se, e somente se, } \mathbf{x} = \mathbf{y}, \quad (2.2.2)$$

- simetria:

$$dist(\mathbf{x}, \mathbf{y}) = dist(\mathbf{y}, \mathbf{x}), \quad (2.2.3)$$

- desigualdade do triângulo:

$$dist(\mathbf{x}, \mathbf{z}) + dist(\mathbf{z}, \mathbf{y}) \geq dist(\mathbf{x}, \mathbf{y}). \quad (2.2.4)$$

Nas subseções seguintes desta seção, encontram-se algumas métricas que atendem à todas ou à algumas dessas propriedades e que serão detalhadas em cada uma das características já expostas e esperadas de uma métrica de similaridade entre duas séries temporais. Ao fim do capítulo se encontra a Tabela 2.4 que sumariza todas as informações discutidas nesta seção.

Vale mencionar, neste momento, que existem métricas de dissimilaridade obtidas por meio de modelos estatísticos das séries temporais que sejam representativos de cada uma das séries temporais a serem comparadas. Supõe-se que elas foram geradas por modelos paramétricos, sendo o modelo ARIMA um dos modelos mais utilizados. A obtenção da medida de dissimilaridade entre as séries se dá a partir da comparação entre os parâmetros desses modelos. As métricas obtidas a partir de modelos estatísticos possuem aplicação maior em agrupamentos de séries temporais baseados na predição delas, não sendo assim, em geral, aplicáveis à problemas de agrupamento de curvas de carga, e por isso não serão abordadas nesta dissertação de mestrado. No entanto, algumas das abordagens neste sentido, assim como outras referências, podem ser encontradas em (Montero e Vilar 2014).

2.2.1 Minkowski

A distância de Minkowski é definida como

$$dist(\mathbf{x}, \mathbf{y}) = \left(\sum_i^n |x_i - y_i|^p \right)^{\frac{1}{p}}. \quad (2.2.5)$$

A métrica de Minkowski é também conhecida como norma L_p , e tem como casos notáveis a distância Euclidiana, ou L_2 , quando $p = 2$, a distância *Manhattan*, ou L_1 , quando $p = 1$ e a distância *Chebyshev*, ou L_∞ , quando $p \rightarrow \infty$.

Na figura 2.4 encontram-se duas séries temporais, na tabela 2.1 os valores de distância obtidos variando-se o valor de p . Pela tabela, pode-se notar que pequenas variações no valor de p

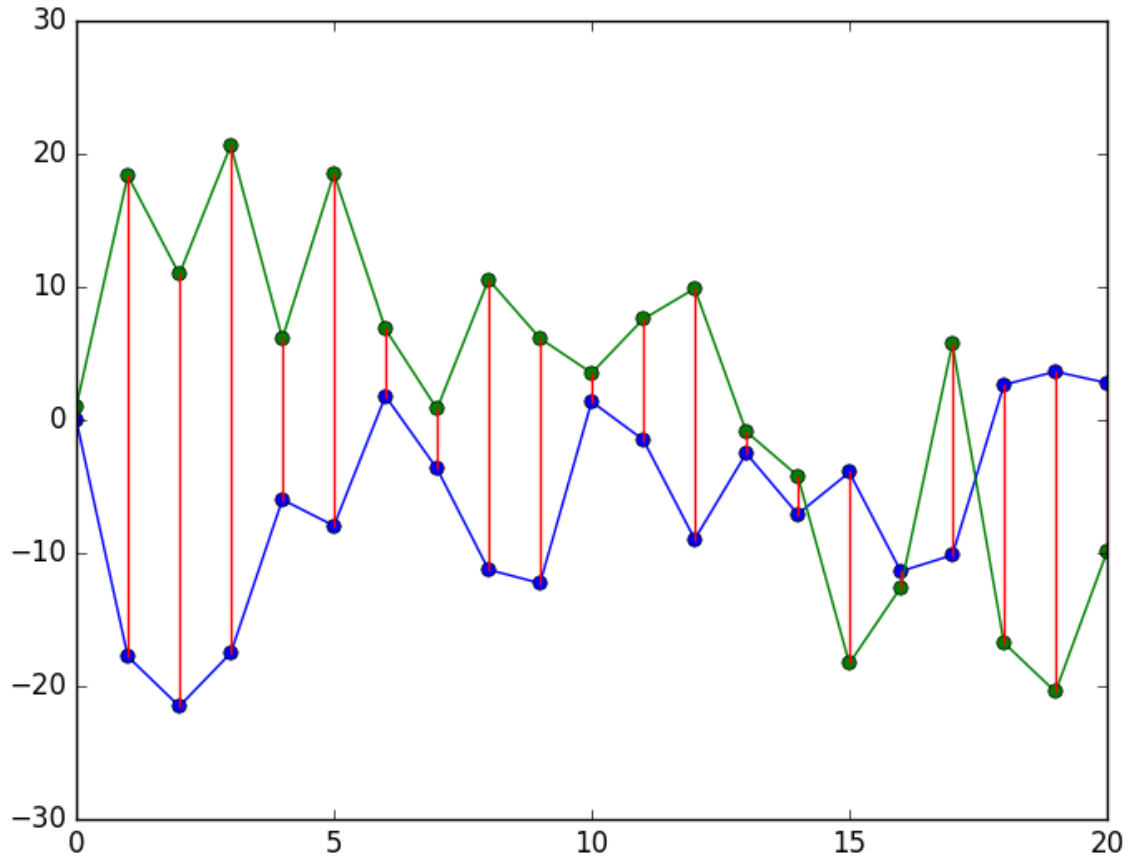


Figura 2.4: Distância Euclidiana entre as duas curvas é dada pela soma das distâncias euclidianas entre as observações de cada curva no mesmo instante.

acarretam em grandes variações no valor da métrica, bem como que este valor diminui ao passo que p aumenta. A métrica de Minkowski atende à todas as propriedades das métricas de distância e, dessa maneira, pode ser considerada uma distância, e o seu custo computacional é $\mathcal{O}(n)$, onde n é o número de amostras ou observações das séries a serem comparadas.

Tabela 2.1: Distâncias Minkowski entre as curvas da figura 2.4 para diversos valores de p .

	p=1	p=2	p=3	p=4	p=5	p→ ∞
distância	318.18	86.88	59.59	50.54	44.03	38.14

2.2.2 *Dynamic Time Warping* - DTW

A *Dynamic Time Warping* é, talvez, a métrica de dissimilaridade mais utilizada para dados de séries temporais. Proposta inicialmente para ser utilizada em problemas de detecção de fala, a DTW se mostrou útil para tarefas de mineração de dados em séries temporais (Berndt e Clifford 1994).

Dada duas séries temporais $\mathbf{x} = x_1, x_1, x_3, \dots, x_i, \dots, x_n$ e $\mathbf{y} = y_1, y_2, y_3, \dots, y_j, \dots, y_m$, suponha que seja feito a construção de um *grid* $n_x m$, no qual na posição (i, j) se encontra o valor resultante de uma métrica de distância $\delta(i, j)$ entre os pontos x_i e y_j . Tal métrica pode ser, por exemplo, a já citada distância Minkowski. A partir deste *grid*, é construído um caminho $W = w_1, w_2, w_3, \dots, w_k$, onde cada w_k corresponde à alguma posição $(i, j)_k$ do grid, tal que a distância DTW entre as séries $\mathbf{x}$ e $\mathbf{y}$, definida por esse caminho, seja minimizada:

$$DTW(\mathbf{x}, \mathbf{y}) = \min_W \left[\sum_{k=1}^p \delta(w_k) \right] \quad (2.2.6)$$

Assim, pode-se definir o cálculo da dissimilaridade DTW por meio da programação dinâmica da variável $\gamma(i, j)$ que representa a soma acumulada da distância escolhida na posição (i, j) do *grid*:

$$\gamma(i, j) = \delta(i, j) + \min[\gamma(i-1, j), \gamma(i-1, j-1), \gamma(i, j-1)]. \quad (2.2.7)$$

O cálculo se inicia na posição (m, n) , e a partir da fórmula recursiva apresentada na equação 2.2.7, constrói-se uma tabela de distâncias acumuladas. O caminho mínimo, que é a distância DTW entre as séries temporais, é obtido realizando o caminho reverso, ou seja, com o valor contido em $\gamma(m, n)$.

A ordem de complexidade do algoritmo é $\mathcal{O}(n * m)$, no entanto, a adição de algumas restrições fazem com que se alcance ganhos computacionais com pouca perda de performance. Uma primeira tentativa, é a criação de uma janela de tamanho ω tal que, $|i_k - j_k| \leq \omega$. Note que para $\omega = 1$, a DTW se torna a distância euclidiana. Em (Giusti e Batista 2013), sugere-se a utilização de uma janela com tamanho de aproximadamente 10% do tamanho da menor série temporal. Além da redução no tempo de processamento, existem estudos que, a partir de experimentos empíricos, afirmam que a utilização de janelas melhoram a performance de classificadores de séries temporais ao não permitirem mapeamentos patológicos entre uma pequena seção de uma curva à uma grande seção da outra (E. Keogh e Ratanamahatana 2005). Na figura 2.7 pode ser vista uma comparação entre as métricas DTW, com e sem janela, e a Euclidiana, para duas curvas do conjunto de dados ECG200 do *UCR Time Series* (Y. Chen et al. 2015).

A DTW não pode ser considerada uma métrica de distância pois não obedece à desigualdade do triângulo, no entanto, diferentemente da distância Minkowski, pode ser aplicada à series temporais de diferentes tamanhos e não é sensível ao deslocamentos entre as séries temporais.

Na literatura, a distância Euclidiana é uma das abordagens mais utilizadas para o agrupamento em geral, no entanto, em alguns casos nos quais os dados são séries temporais, ela pode apresentar resultados incoerentes ao se definir a similaridade entre duas séries temporais. Para discutir tal afirmação, os experimentos realizados em (Minnaar 2014) serão reproduzidos aqui.

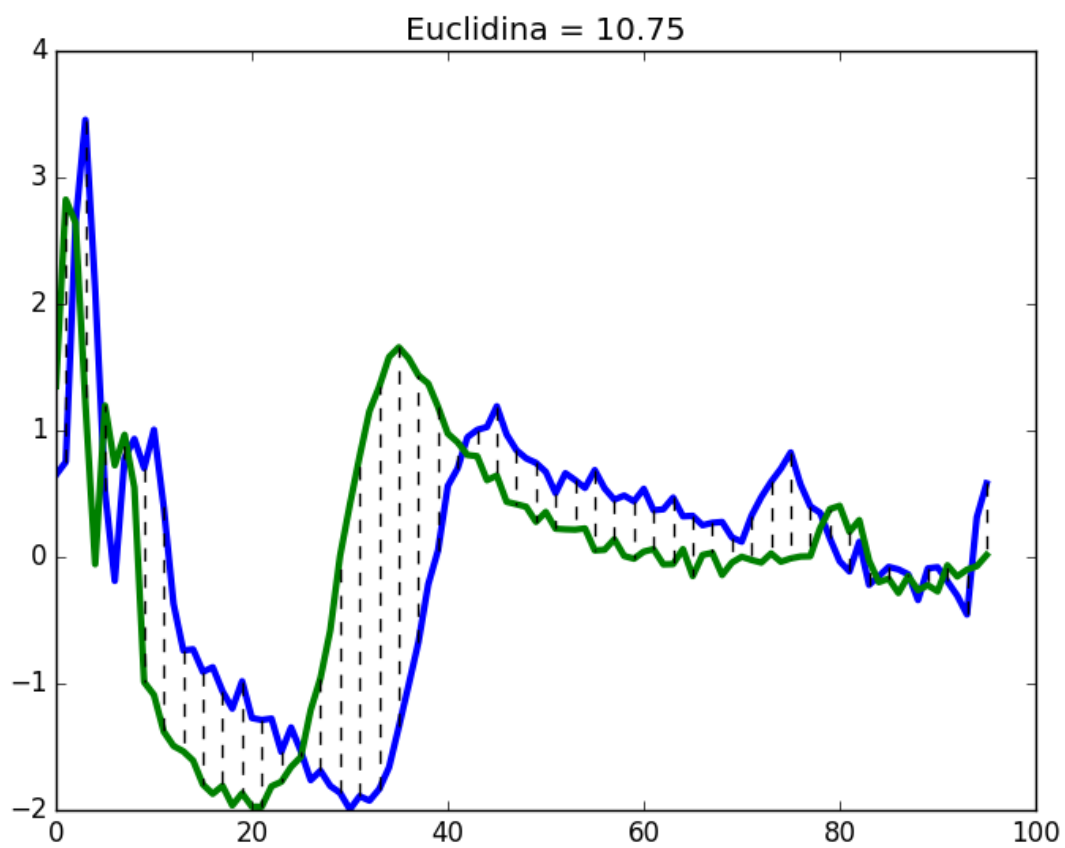


Figura 2.5: Distância Euclidiana entre duas curvas do conjunto de dados ECG200.

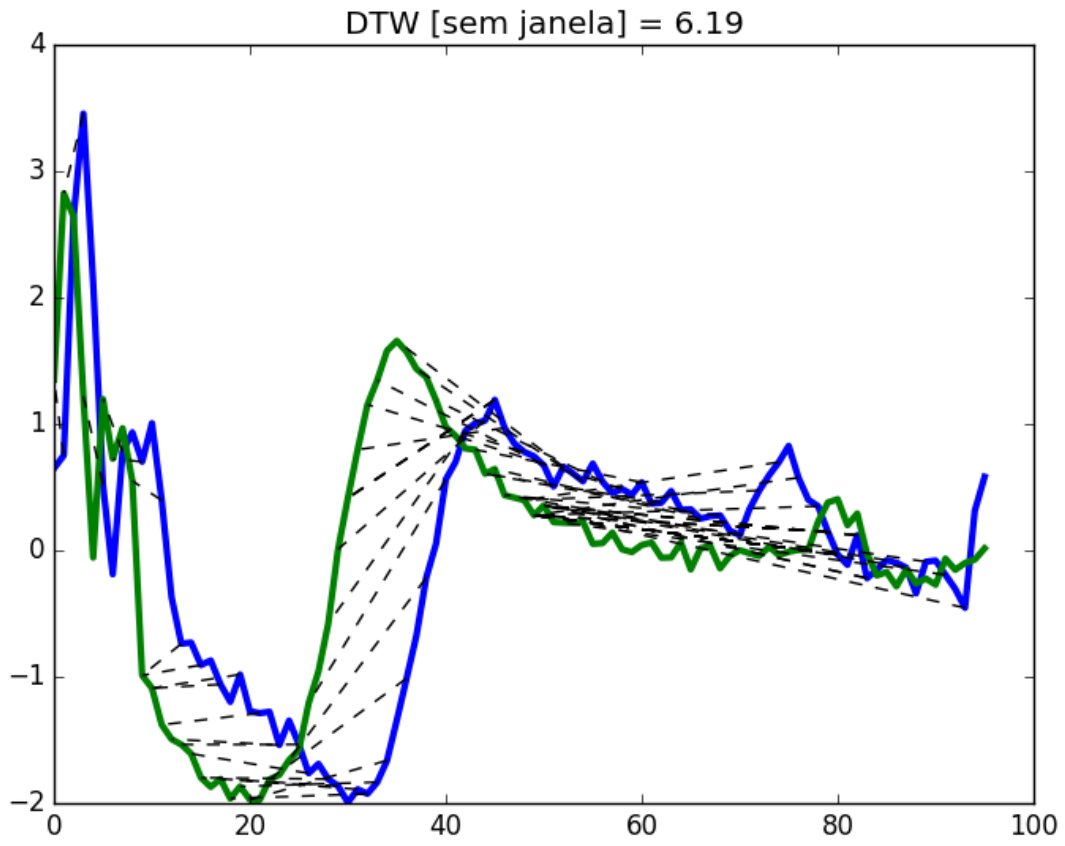


Figura 2.6: DTW sem janela entre duas curvas do conjunto de dados ECG200, na qual foi usada como distância $\delta(x_i, y_j) = (x_i - y_j)^2$.

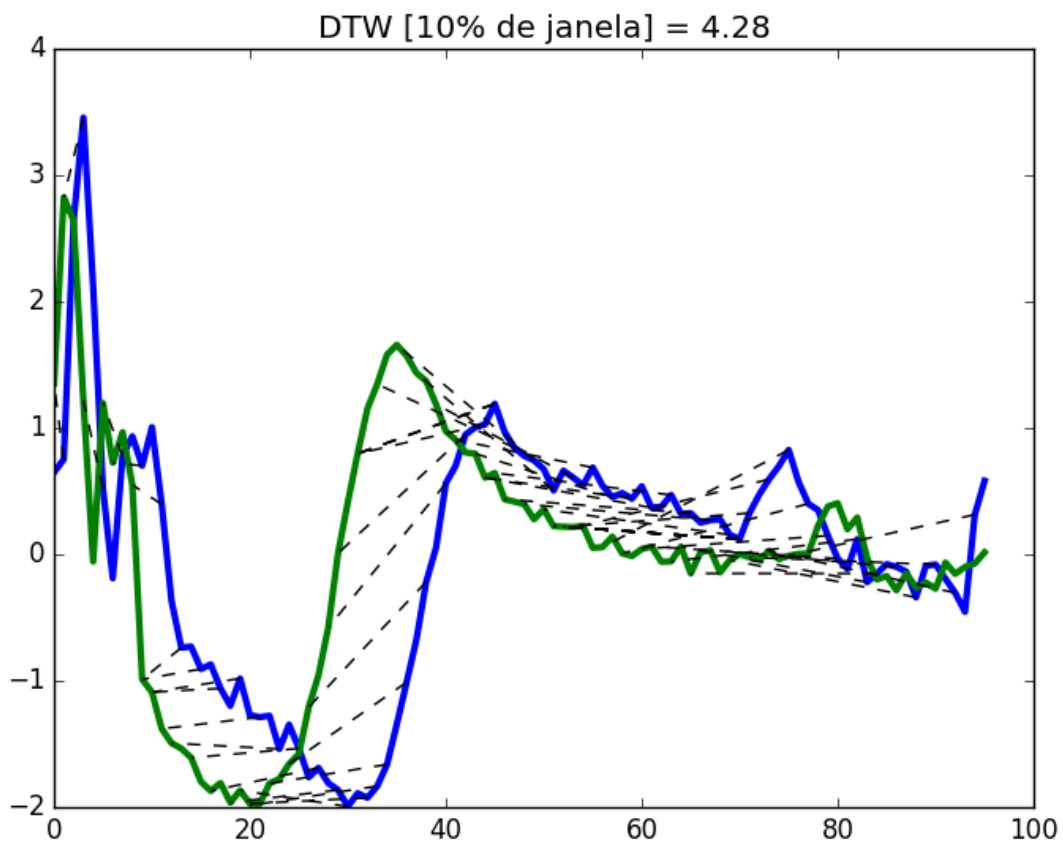


Figura 2.7: DTW com janela de tamanho 10% do tamanho da série temporal, entre duas curvas do conjunto de dados ECG200, na qual foi usada como distância $\delta(x_i, y_j) = (x_i - y_j)^2$.

Tabela 2.2: Distâncias euclidiana e DTW entre as curvas da figura 2.8.

	Euclidiana	DTW
(ts1,ts2)	26.95	1.51
(ts1,ts3)	23.18	1.85

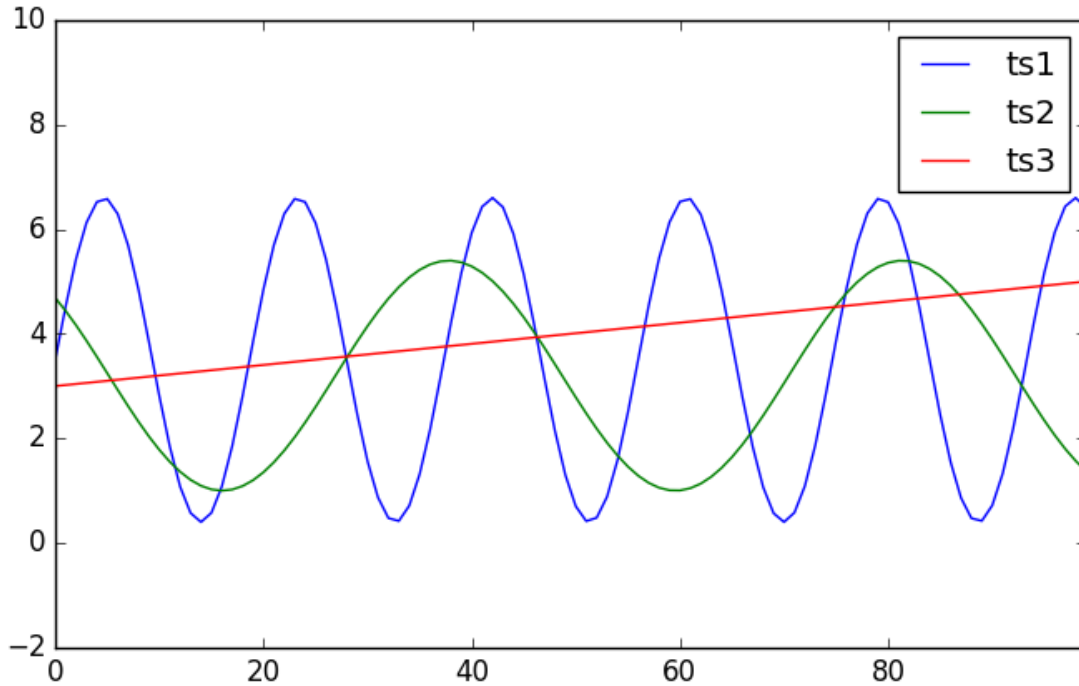


Figura 2.8: Duas senóides e uma reta têm suas distâncias comparadas.

Na figura 2.8, encontram-se três séries temporais e, fixando-se a curva ts1, para compará-la com as outras duas, foram calculadas as distâncias euclidianas e DTW. Os resultados podem ser vistos na tabela 2.2. Pelos dados da tabela, fica claro que a métrica euclidiana falha ao comparar as curvas em análise, já que a distância euclidiana entre a senoide ts1 e a reta ts3 foi menor que a distância euclidiana entre a senoide ts1 e a outra senoide ts2, o que contradiz o nosso senso de similaridade, que é baseado no formato da curva. Por sua vez, a métrica DTW nos diz que a senoide ts1 é mais "próxima" à senoide ts2 do que à reta ts3, o que condiz com nosso senso de similaridade. A vantagem da DTW em relação à Euclidiana, neste caso, está no fato de a DTW ser invariante à distorção de fase que as curvas podem apresentar entre si.

2.2.3 *LB-Keogh*

A DTW é reconhecida como uma das métricas mais robustas na obtenção da similaridade entre séries temporais, no entanto, seu elevado custo computacional faz com que esta seja preterida em

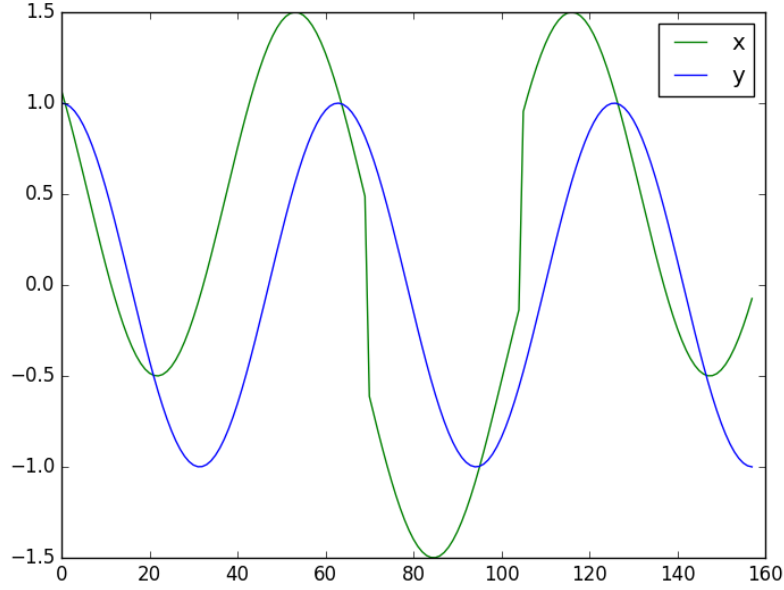


Figura 2.9: Curvas $\mathbf{x}$ e $\mathbf{y}$.

muitas aplicações. Em uma tentativa de se aproximar a DTW com um custo computacional $\mathcal{O}(n)$, em (E. Keogh e Ratanamahatana 2005) foi proposta a métrica LB-Keogh. No trabalho citado, tal métrica teve resultados próximos à DTW para diversas bases de dados e em tempo significativamente menor. Assim, dada duas séries temporais $\mathbf{x}$ e $\mathbf{y}$ de comprimento n , então, define-se as séries $\mathbf{u}$ e $\mathbf{l}$ como:

$$\begin{cases} u_i = \max(y_{i-r} : y_{i+r}), \\ l_i = \min(y_{i-r} : y_{i+r}), \end{cases} \quad (2.2.8)$$

onde r é definido como o alcance da aproximação e $1 \leq r \leq n$. As novas séries $\mathbf{u}$ e $\mathbf{l}$ formam um envelope que contém a série $\mathbf{y}$ sendo o limite superior desta formada pela curva $\mathbf{u}$ e o inferior pela curva $\mathbf{l}$. Dessa maneira, faz-se necessário que $\forall i \ u_i \geq y_i \geq l_i$. Após a definição das duas curvas que formam o envelope, pode-se definir, finalmente, a distância *LB-Keogh* entre as curvas $\mathbf{x}$ e $\mathbf{y}$:

$$dist_{LB-Keogh}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_i^n \begin{cases} (x_i - u_i)^2, & \text{se } x_i > u_i, \\ (x_i - l_i)^2, & \text{se } x_i < l_i, \\ 0, & \text{caso contrário.} \end{cases}} \quad (2.2.9)$$

A métrica LB-Keogh é sempre menor ou igual ao valor obtido pela DTW, e ao contrário desta última, pode ser aplicada somente em séries temporais de mesmo comprimento e também não pode ser considerada uma métrica de distância pois não obedece à desigualdade do triângulo. Um exemplo de cálculo da LB-Keogh pode ser visto nas figuras 2.9, 2.10 e 2.11.

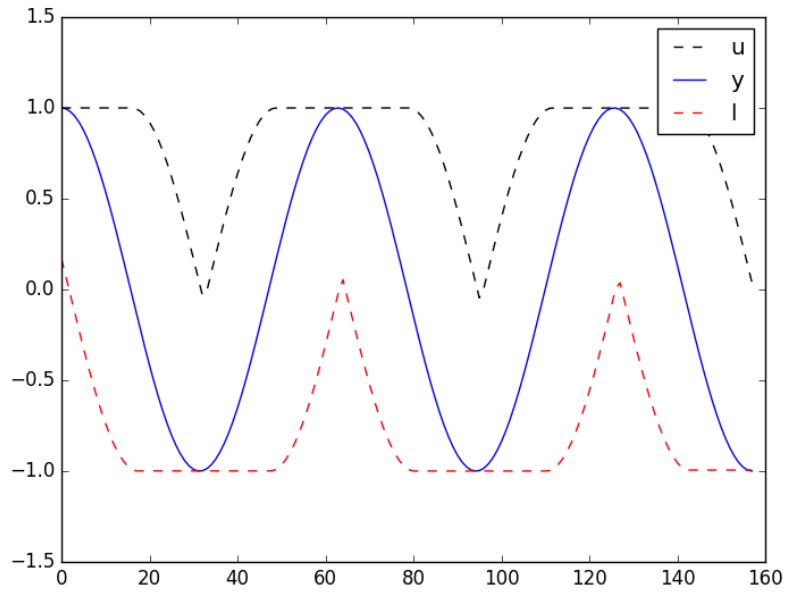


Figura 2.10: Curva y envelopada pelas curvas u e l obtidas pela equação 2.2.8.

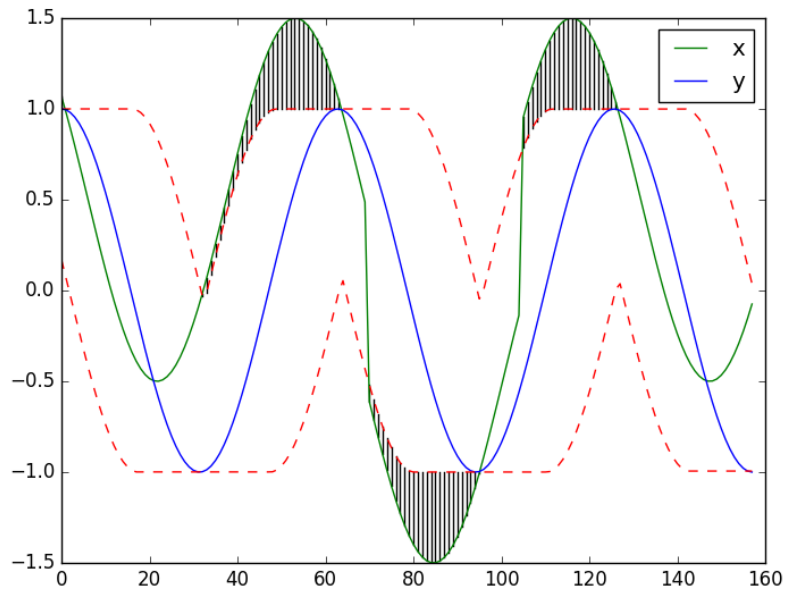


Figura 2.11: A raiz do somatório das distâncias em preto fornece a distância LB-Keogh, como descrito pela equação 2.2.9.

2.2.4 *Edit distance on real sequence* - EDR

Em (L. Chen, Özsu e Oria 2005), é proposta uma nova técnica a ser aplicada para a obtenção de métricas de similaridade entre séries temporais multivariadas, com aplicação na análise de trajetórias de objetos móveis. Neste artigo, é defendido que a *Edit distance on real sequence* é uma métrica mais robusta e menos sensível à ruídos e *outliers* que outras métricas tradicionais, como a DTW e a distância Euclidiana.

A métrica é baseada na *Edit distance*, ou distância de *Levenshtein*, que possui larga aplicação na obtenção de métricas de dissimilaridades entre *strings*. A EDR é uma adaptação desta distância a ser aplicada à uma sequência de números reais, ao invés de ser aplicada à uma sequência de caracteres. Apesar de ter sido proposta inicialmente para séries temporais multivariadas, em (Serrà e Arcos 2014) são feitos experimentos com séries temporais univariadas nos quais se chegou a resultados superiores aos resultados de métricas tradicionais como a DTW e distância Euclidiana. Assim como a DTW, a métrica EDR não é sensível ao deslocamento entre as curvas e pode ser utilizada para curvas de comprimentos diferentes.

A EDR de duas curvas de tamanho m e n pode ser obtida por meio da seguinte programação dinâmica:

$$D_{ij} = \begin{cases} D_{i-1,j-1} & \text{se } c(x_i, y_j) = 1, \\ 1 + \min(D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}) & \text{se } c(x_i, y_j) = 0, \end{cases} \quad (2.2.10)$$

onde $i = 1, \dots, m$ e $j = 1, \dots, n$. A função de casamento $c(x_i, y_j) = \theta(\varepsilon - f(x_i, y_j))$, sendo $\theta(\cdot)$ a função degrau unitário, $\varepsilon \in [0, \infty)$ é um parâmetro a ser escolhido, e sugerido por (L. Chen, Özsu e Oria 2005) como um quarto da variância da série que possui a maior variância das suas observações, e a função $f(\cdot)$ é normalmente definida como o valor absoluto da diferença entre as variáveis, ou seja, $f(x_i, y_j) = |x_i - y_j|$. Além disso, a matriz D deve ser iniciada da seguinte maneira: $D_{i,0} = i$ para $i = 0, 1, \dots, m$ e $D_{0,j} = j$ para $j = 0, 1, \dots, n$, e a distância EDR entre as curvas é dado pelo valor contido em $D_{m,n}$.

A EDR não pode ser considerada uma métrica de distância pois não possui a propriedade de desigualdade do triângulo e tem complexidade computacional $\mathcal{O}(m * n)$. Dado o seu elevado custo computacional, restrições devem ser feitas ao cálculo da dissimilaridade.

2.2.5 *Dice, Jaccard e Lorentzian*

As distâncias *Dice*, *Jaccard* e *Lorentzian* obtiveram performance superior à distância euclidiana em um estudo empírico de classificação de séries temporais realizado em (Giusti e Batista 2013), e por isso são utilizadas neste trabalho e definidas a seguir. Para duas séries temporais $\mathbf{x}$ e $\mathbf{y}$ de comprimento n , temos:

- Dice

$$\text{dist}(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^n (x_i - y_i)^2}{\sum_i^n x_i^2 + \sum_i^n y_i^2} \quad (2.2.11)$$

- Jaccard

$$\text{dist}(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^n (x_i - y_i)^2}{\sum_i^n (x_i^2 + y_i^2 - x_i y_i)} \quad (2.2.12)$$

- Lorentzian

$$dist(\mathbf{x}, \mathbf{y}) = \sum_i^n \ln(1 + |x_i - y_i|) \quad (2.2.13)$$

Todas as três distâncias possuem ordem de complexidade $\mathcal{O}(n)$, e são aplicáveis somente à séries temporais com o mesmo comprimento n .

2.2.6 *Cosine*

A métrica *Cosine* é definida a partir do cosseno do ângulo entre os vetores $\mathbf{x}$ e $\mathbf{y}$ das séries temporais:

$$dist(\mathbf{x}, \mathbf{y}) = 1 - \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2}, \quad (2.2.14)$$

onde $\|\mathbf{x}\|_2$ é a norma euclidiana do vetor $\mathbf{x}$. A distância *Cosine* tem ordem de complexidade $\mathcal{O}(n)$ e só pode ser calculada em entre séries temporais de mesmo tamanho n .

2.2.7 Distância de Mahalanobis

A distância de Mahalanobis é uma distância estatística proposta em (Mahalanobis 1936), que leva em conta a covariância entre as amostras da base de dados. Útil na detecção de *outliers*, a distância de Mahalanobis utiliza a matriz de covariância Σ das amostras, e, para o caso específico no qual $\Sigma = I$, então a métrica é análoga à distância euclidiana. Assim, a distância de Mahalanobis entre os vetores $\mathbf{x}$ e $\mathbf{y}$, pertencentes ao conjunto de dados que gerou a matriz Σ , é:

$$dist(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T \Sigma^{-1} (\mathbf{x} - \mathbf{y})}. \quad (2.2.15)$$

Os maiores custos computacionais da obtenção da distância de Mahalanobis residem no cálculo da matriz de covariância Σ e da inversão da mesma. Além do custo computacional em si, numericamente, a inversão da matriz Σ é problemática pois pode levar a diversos erros numéricos já que, em geral, a matriz Σ é de baixo *rank* para conjuntos de dados de séries temporais. O cálculo e inversão da matriz Σ têm ordem de complexidade $\mathcal{O}(n^2)$, e a métrica só pode ser aplicada à séries temporais de mesmo tamanho.

Em (Prekopcsák e Lemire 2012), foi realizado um estudo de classificação de séries temporais no qual a distância de Mahalanobis foi utilizada como métrica de dissimilaridade entre as curvas. Foram realizadas aproximações da inversa da matriz de covariância dos dados que fizeram com que se tivessem ganhos computacionais relevantes, possuindo, assim, custos computacionais inferiores aos custos da DTW, no entanto, com uma pequena perda de performance na acurácia.

2.2.8 Distâncias baseadas na correlação de Pearson

A correlação de Pearson entre duas séries temporais $\mathbf{x}$ e $\mathbf{y}$ de tamanho n é dada por

$$COR(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i^n (x_i - \bar{x})^2} \sqrt{\sum_i^n (y_i - \bar{y})^2}}, \quad (2.2.16)$$

sendo $\bar{\mathbf{x}}$ e $\bar{\mathbf{y}}$ as média das observações de $\mathbf{x}$ e $\mathbf{y}$ respectivamente. O valor da correlação de Pearson é compreendido no intervalo $[-1, 1]$. Dessa maneira, as seguintes distância baseadas na correlação de Pearson podem ser definidas

$$dist_1(\mathbf{x}, \mathbf{y}) = 1 - COR(\mathbf{x}, \mathbf{y}), \quad (2.2.17)$$

$$dist_2(\mathbf{x}, \mathbf{y}) = \sqrt{2(1 - COR(\mathbf{x}, \mathbf{y}))}, \quad (2.2.18)$$

$$dist_3(\mathbf{x}, \mathbf{y}) = \sqrt{\left(\frac{1 - COR(\mathbf{x}, \mathbf{y})}{1 + COR(\mathbf{x}, \mathbf{y})}\right)^\beta}, \quad (2.2.19)$$

onde β é um parâmetro para regular o decrescimento da distância $dist_3$. Todas as métricas podem ser aplicadas somente entre séries temporais de mesmo comprimento e possuem ordem de complexidade $\mathcal{O}(n)$, onde n é o número de observações das séries temporais em comparação.

2.2.9 Distância baseada na correlação temporal

Uma medida da correlação temporal entre duas séries temporais definida em (Chouakria e Nagabhushan 2007), é equivalente à uma medida do quão similar é o comportamento dinâmico entre as séries, ou, em outras palavras, o quão próximo está o instante de tempo no qual elas crescem ou decrescem. Definida no intervalo $[-1, 1]$ a correlação temporal entre as séries $\mathbf{x}$ e $\mathbf{y}$ de tamanho n pode ser obtida por

$$CORT(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^{n-1} (x_i - x_{i+1})(y_i - y_{i+1})}{\sqrt{\sum_i^{n-1} (x_i - x_{i+1})^2} \sqrt{\sum_i^{n-1} (y_i - y_{i+1})^2}}, \quad (2.2.20)$$

Valores de correlação temporal próximos de 1 indicam que as séries $\mathbf{x}$ e $\mathbf{y}$ crescem e decrescem simultaneamente e com a mesma taxa, enquanto valores próximos de -1 indicam que uma série cresce ao passo que a outra decresce simultaneamente, e isso ocorre com a mesma taxa. Valores próximos de 0 indicam que estatisticamente não se pode afirmar nada a respeito da taxa e do crescimento ou decrescimento de uma curva ao saber o comportamento da outra.

A partir da equação 2.2.20 criar uma métrica que leva em conta tanto aspectos temporais como a correlação temporal quanto a proximidade ponto a ponto entre as séries. Para tal, faz-se uso de uma função de sintonia e de alguma métrica convencional, como a métrica Euclidiana ou DTW, por exemplo. Assim, uma possível métrica de distância que leve em conta o aspecto temporal entre as séries $\mathbf{x}$ e $\mathbf{y}$ seria:

$$dist(\mathbf{x}, \mathbf{y}) = f(CORT(\mathbf{x}, \mathbf{y})).dist_Q(\mathbf{x}, \mathbf{y}) \quad (2.2.21)$$

onde $dist_Q$ é a distância convencional entre as séries e $f(\cdot)$ é a função de sintonia que deve amplificar a distância convencional quando a correlação temporal se encontra no intervalo $[-1, 0[$, reduzi-la quando a correlação temporal se encontra no intervalo $]0, 1]$, e não alterá-la quando a correlação

temporal for igual à 0. A seguinte função, proposta em (Chouakria e Nagabhushan 2007), atende à estes critérios:

$$f(x) = \frac{2}{1 + \exp(\gamma x)}, \quad \gamma \geq 0, \quad (2.2.22)$$

onde γ é um parâmetro de controle que controla o quanto o aspecto temporal deve influenciar na métrica de distância. Valores grandes de γ fazem com que o valor fornecido por $dist_Q$ seja mais penalizado, ou atenuado, quando se obtém valores não nulos de correlação temporal entre as séries. Tal comportamento da função de sintonia bem como a influência do parâmetro γ podem ser vistos na figura 2.12.

Em (Chouakria e Nagabhushan 2007) são feitos experimentos com séries temporais reais e sintéticas, utilizando-se métricas baseadas na correlação de Pearson e na correlação temporal, e como distâncias convencionais foram escolhidas a DTW e a distância Euclidiana. Após a realização do agrupamento das curvas por meio de algoritmos hierárquicos, utilizando-se as métricas citadas anteriormente, chegou-se à resultados superiores com a correlação temporal utilizando a similaridade DTW como distância convencional. Este resultado superior foi alcançado tanto para os conjuntos de dados sintéticos quanto para o conjunto de dados reais.

A correlação temporal não pode ser considerada uma métrica de distância pois não obedece à desigualdade do triângulo e como o cálculo da correlação temporal tem custo $\mathcal{O}(n)$, seu custo computacional se torna, em geral, o mesmo da distância convencional utilizada.

2.2.10 *Complex Invariant Distance - CID*

Em alguns casos, séries temporais "complexas", dependendo da métrica a ser utilizada, podem se tornar mais similares à uma outra série temporal "simples" do que à outra série tão "complexa" quanto ela. Por complexidade, entende-se uma série temporal com muitos picos e vales (Batista, Wang e E. J. Keogh 2011). A *Complex Invariant Distance*, ou CID, leva em conta estimativas de complexidade das séries temporais para realizar uma correção na medida da dissimilaridade e, dessa maneira, para duas séries $\mathbf{x}$ e $\mathbf{y}$

$$dist_{CID} = dist_Q(\mathbf{x}, \mathbf{y}) \cdot CF(\mathbf{x}, \mathbf{y}), \quad (2.2.23)$$

onde $dist_Q(.)$ é qualquer uma das dissimilaridades já vistas, como a DTW ou Euclidiana, e $CF(.)$ é um fator de correção que leva em conta uma estimativa da complexidade relativa entre as séries. Note que, pelo fato do fator de correção ser relativo, duas séries que tiverem complexidades semelhantes tenderão a ter um valor muito próximo de $dist_Q$, ao passo que séries que possuem complexidades distintas, penalizarão a dissimilaridade $dist_Q$.

Em (Batista, Wang e E. J. Keogh 2011), é proposto o fator de correção

$$CF(\mathbf{x}, \mathbf{y}) = \frac{\max(CE(\mathbf{x}), CE(\mathbf{y}))}{\min(CE(\mathbf{x}), CE(\mathbf{y}))}, \quad (2.2.24)$$

onde $CE(.)$ é um estimador de complexidade da série temporal. A complexidade das séries pode ser estimada de diferentes maneiras, como: pela complexidade de Kolmogorov, métricas de entropia,

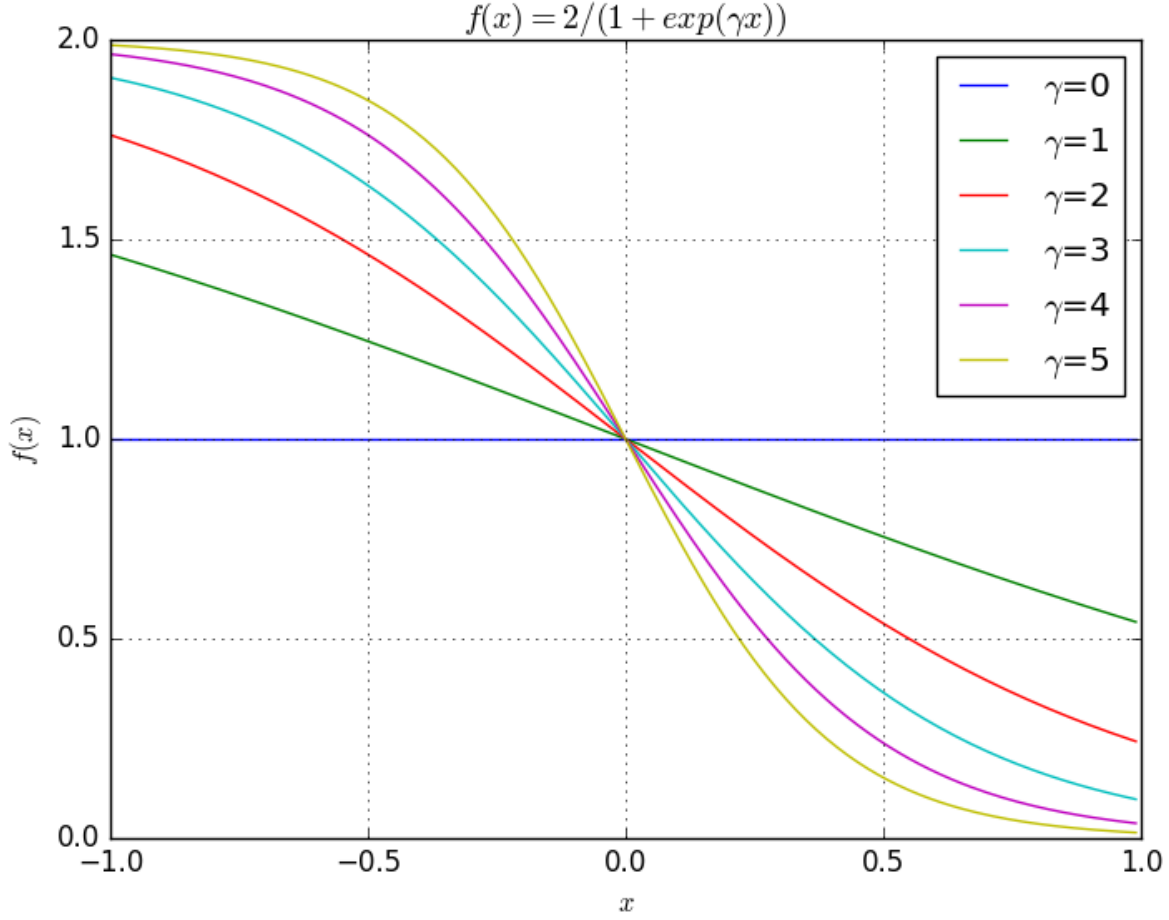


Figura 2.12: Função de sintonia 2.2.22 para diferentes valores de γ .

número de trocas do sinal da derivada, entre outras citadas em (Montero e Vilar 2014). A medida de complexidade a ser utilizada neste trabalho, sugerida em (Batista, Wang e E. J. Keogh 2011), é bastante simples e intuitiva e será obtida pelo comprimento do tamanho da série temporal após esta ser "esticada", e é obtida por

$$CE(\mathbf{x}) = \sqrt{\sum_{i=1}^{n-1} (x_i - x_{i+1})^2}. \quad (2.2.25)$$

O cálculo do fator de correção possui complexidade $\mathcal{O}(n)$, o que faz com que o custo computacional da métrica seja, em geral, igual ao custo da dissimilaridade $dist_Q$, e a pergunta se a dissimilaridade CID pode ou não ser aplicada à séries temporais com diferentes comprimentos, deve ser respondida pela pergunta se a métrica $dist_Q$ o faz.

A seguir é exibido um exemplo no qual se faz necessário levar em conta a complexidade entre as séries temporais. Na figura 2.13 podem ser vistas três séries temporais, sendo uma simples, $ts1$,

e duas complexas, $ts2$ e $ts3$. Foram calculadas as distâncias euclidianas e as distâncias euclidianas corrigidas pelo fator de complexidade da Equação 2.2.24. Além dos valores das distâncias euclidianas e CID-euclidiana, na Tabela 2.3, na primeira coluna, podem ser vistas as complexidades de cada curva.

Note que, como esperado, as curvas com maior ruído, ou com maior número de picos e vales, possuem complexidade aproximadamente quatro vezes maior do que a curva mais suave, ou mais simples. A submatriz de distância euclidiana entre as curvas nos mostra que as curvas complexas, $ts2$ e $ts3$, estão mais distantes entre si do que da curva mais simples, $ts1$. Tal resultado contradiz nossa intuição de proximidade entre as três curvas analisadas. No entanto, após o fator de correção que leva em conta a complexidade, a distância euclidiana foi corrigida, já que a distância CID-euclidiana entre as curvas complexas é menor que entre cada uma e a curva mais simples, $ts1$.

Este resultado é, em geral, o mesmo para outras métricas de dissimilaridade como a DTW. Em (Batista, Wang e E. J. Keogh 2011) foi realizado o agrupamento hierárquico de formas geométricas, após estas serem convertidas para séries temporais (vide Figuras 2.19 e 2.20), e a métrica CID obteve resultados melhores que os resultados com a distância Euclidiana pura, tanto para dados sintéticos como para dados reais. No teste de classificação empírico realizado em (Giusti e Batista 2013), a CID, utilizando a DTW como distância convencional, obteve a melhor acurácia dentre outras 45 métricas utilizada, para 20 conjuntos de dados de séries temporais diferentes.

Tabela 2.3: Medida de complexidade, como definida na equação 2.2.25, das curvas da figura 2.13 e distâncias euclidiana e CID-euclidiana entre elas.

	Complex.	Eucl.(ts1)	Eucl.(ts2)	Eucl.(ts3)	CID.(ts1)	CID.(ts2)	CID.(ts3)
ts1	0.560	0.0	1.408	1.354	0.0	5.090	4.777
ts2	2.025	1.408	0.0	2.027	5.090	0.0	2.078
ts3	1.975	1.354	2.027	0.0	4.777	2.078	0.0

2.2.11 Discussão geral das métricas de dissimilaridade

Após a exposição das diversas métricas de dissimilaridade, faz-se necessária a sumarização de todas as informações nesta seção, que pode ser vista na Tabela 2.4, bem como de uma discussão geral acerca das métricas de dissimilaridade entre séries temporais. A escolha da métrica a ser utilizada no agrupamento depende sensivelmente do domínio do problema e de como a série temporal foi gerada. Para algumas aplicações, em que o formato da série é relevante, métricas que medem a dissimilaridade ponto a ponto são mais indicadas. Em contrapartida, para aplicações onde o interesse é verificar se as séries crescem ou decrescem de forma similar, independentemente do seu formato, métricas baseadas em correlação temporal são mais apropriadas. Além disso, as séries temporais a serem agrupadas, ou classificadas, podem possuir diversas distorções, devido à sua natureza ou pela maneira que elas foram obtidas, e isso deve ser levado em conta na escolha da métrica, uma vez que determinadas métricas são naturalmente invariantes a algumas distorções.

Em problemas de mineração de dados de séries temporais, é desejável que a métrica de dissimilaridade e o pré-processamento escolhidos alcancem uma ou mais das seguintes invariâncias, que

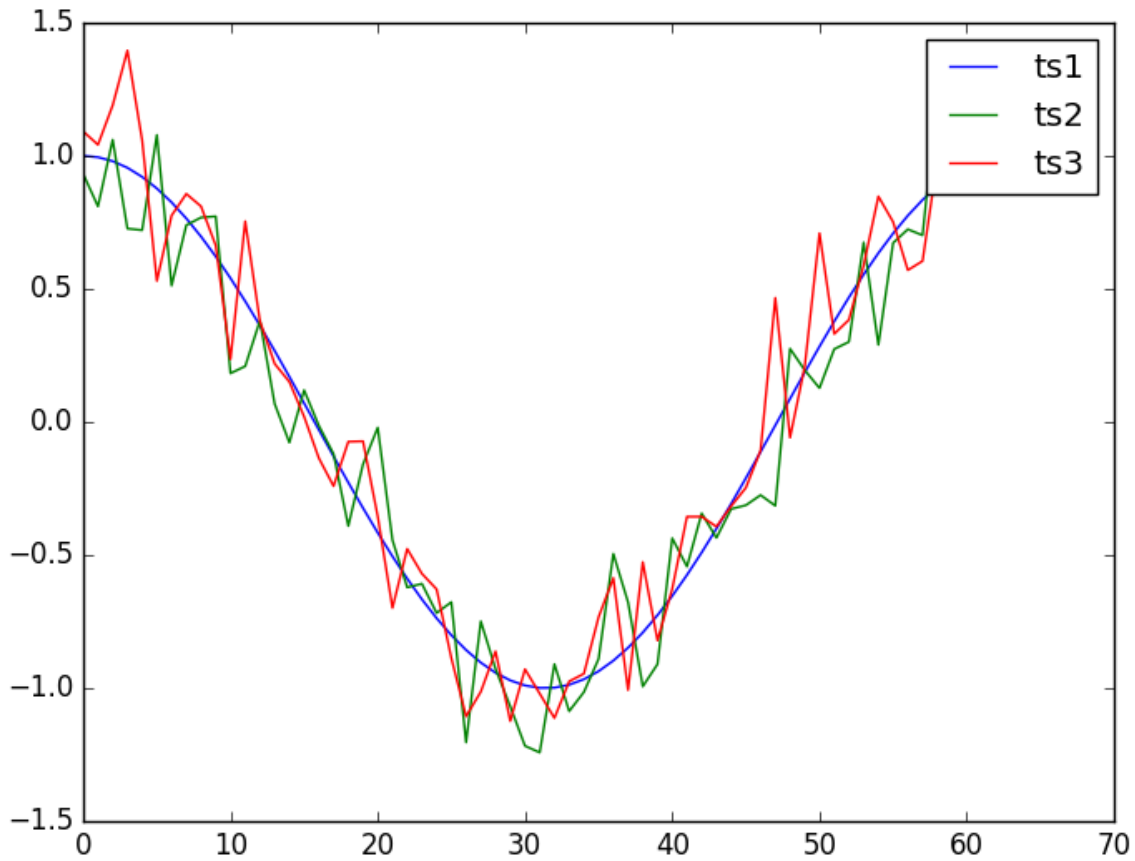


Figura 2.13: Três curvas diferentes sendo duas complexas e uma simples. A estimativa de complexidade de cada uma bem como as distâncias CID e euclidiana entre elas podem ser vistas na tabela [2.3](#)

são detalhadas em (Batista, Wang e E. J. Keogh [2011](#)) e serão brevemente discutidas nas subseções seguintes. Vale ressaltar que ao se tentar remover algumas das distorções em um problema específico, resultados piores podem ser obtidos, de forma que cada conjunto de dados de séries temporais deve ser analisado antes de se definir quais invariâncias são necessárias para a comparação entre as séries temporais.

Invariância à amplitude/*offset*

Quando as séries temporais a serem comparadas são idênticas, mas medidas com diferentes escalas, estas possuem amplitudes distintas, e as métricas de dissimilaridades já discutidas podem apresentar valores muito altos, o que pode indicar erroneamente que as séries são dissimilares. Além da amplitude, séries que são idênticas, mas possuem um valor médio diferente (*offset*),

podem apresentar também valores muito grandes de dissimilaridade. A solução apresentada em (Batista, Wang e E. J. Keogh 2011), e discutida na seção 2.1.1, para se alcançar a invariância à amplitude e *offset* é a utilização da normalização Z nas curvas das séries temporais, em uma etapa de pré-processamento. Exemplos de curvas que devem ser comparadas com invariância à amplitude e *offset* podem ser vistas na Figura 2.14.

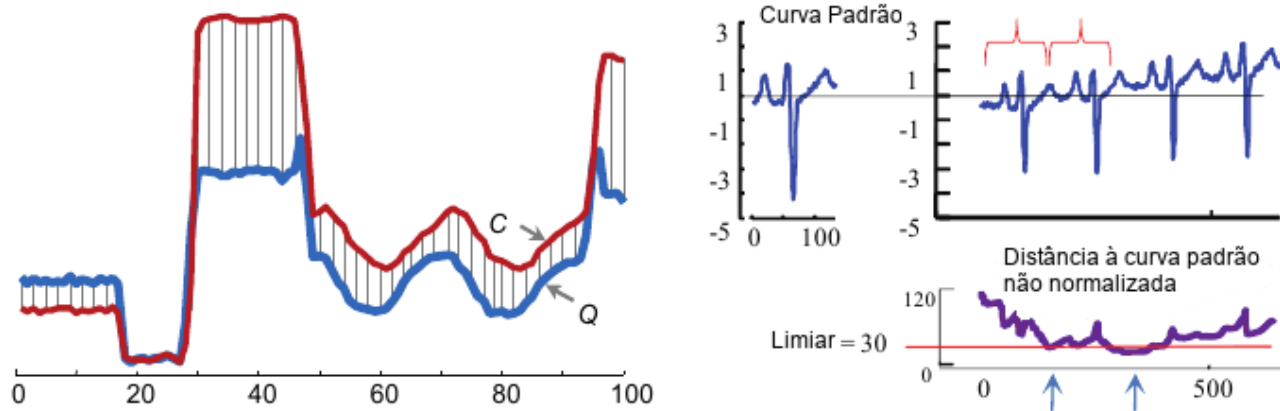


Figura 2.14: À esquerda duas curvas, não normalizadas, que, quando comparadas, se parecem bastante distintas, quando na verdade não o são. À direita um eletrocardiograma, a ser comparado com um sinal de medição contínuo. As duas primeiras comparações têm um bom casamento, mas as subsequentes apresentam valores maiores em uma comparação ponto a ponto devido ao crescente *offset* que a curva começa a apresentar. Adaptado de (Batista, Wang e E. J. Keogh 2011)

Invariância à escala local ou ao deslocamento

Muito comum em sinais biológicos, a invariância à escala local ou ao deslocamento das séries, se manifesta quando o alinhamento ótimo entre as séries ocorre devido à um entortamento local. A dissimilaridade DTW é uma métrica invariante ao deslocamento, e tal propriedade pode ser vista na Figura 2.7

Invariância à escala uniforme

Em alguns casos, duas séries temporais que são idênticas ou da mesma classe mas, por questões de medição, possuem tamanhos distintos, podem ter sua dissimilaridade afetada ao se utilizar alguma métrica de dissimilaridade como a DTW, por exemplo. A solução apresentada em (Batista, Wang e E. J. Keogh 2011) para tal é a multiplicação de uma das curvas por um fator κ , que irá esticar ou reduzir uma das curvas para melhorar o casamento entre elas. A principal dificuldade em se obter a invariância à escala uniforme na comparação de séries temporais é a definição do valor de κ , que na prática, é obtido por meio de diversos testes dentro de determinado intervalo na etapa de pré-processamento. Um alinhamento ótimo obtido por meio da escala uniforme pode ser visto na Figura 2.16.

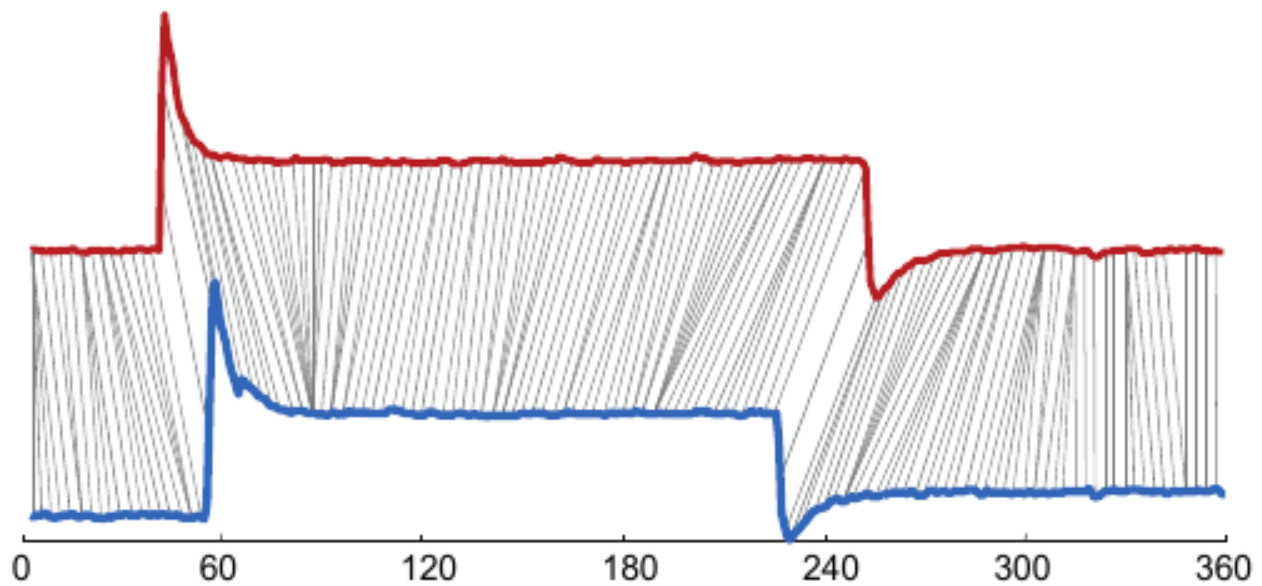


Figura 2.15: Duas curvas cujo alinhamento ótimo é obtido por meio da DTW. Extraído de (Batista, Wang e E. J. Keogh 2011)

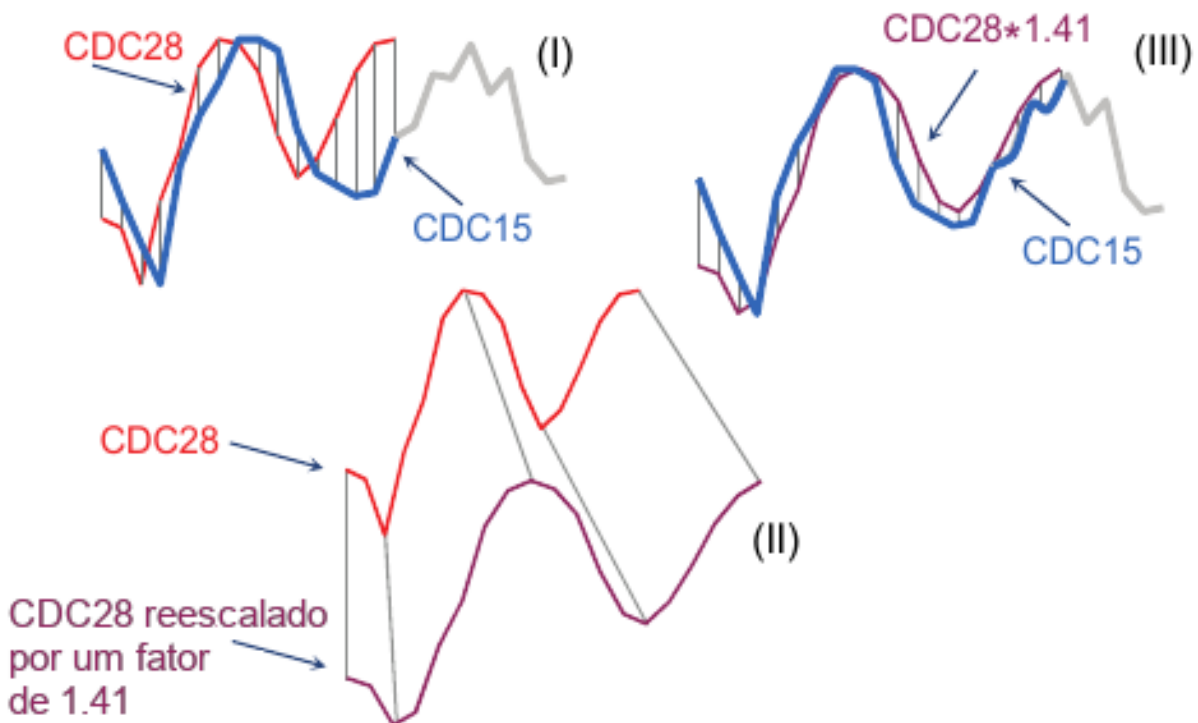


Figura 2.16: Duas curvas cujo alinhamento ótimo é obtido após a curva ser esticada por um fator $\kappa = 1.41$, que foi obtido por força bruta. Adaptado de (Batista, Wang e E. J. Keogh 2011).

Invariância ao defasamento

A invariância ao defasamento é muito importante quando se compara duas séries temporais periódicas. Segundo (Batista, Wang e E. J. Keogh 2011) algumas heurísticas foram propostas para se obter a invariância ao defasamento entre duas séries temporais, mas nenhuma delas é robusta o suficiente, de forma que tal invariância deve ser obtida por meio do teste de todos os alinhamentos possíveis entre as curvas na etapa de pré-processamento. Dessa maneira, este é um problema em aberto na literatura, e nenhuma das dissimilaridades ou técnicas de pré-processamento discutidas alcançam a invariância ao deslocamento. A Figura 2.17 demonstra a obtenção desta invariância para duas séries temporais.

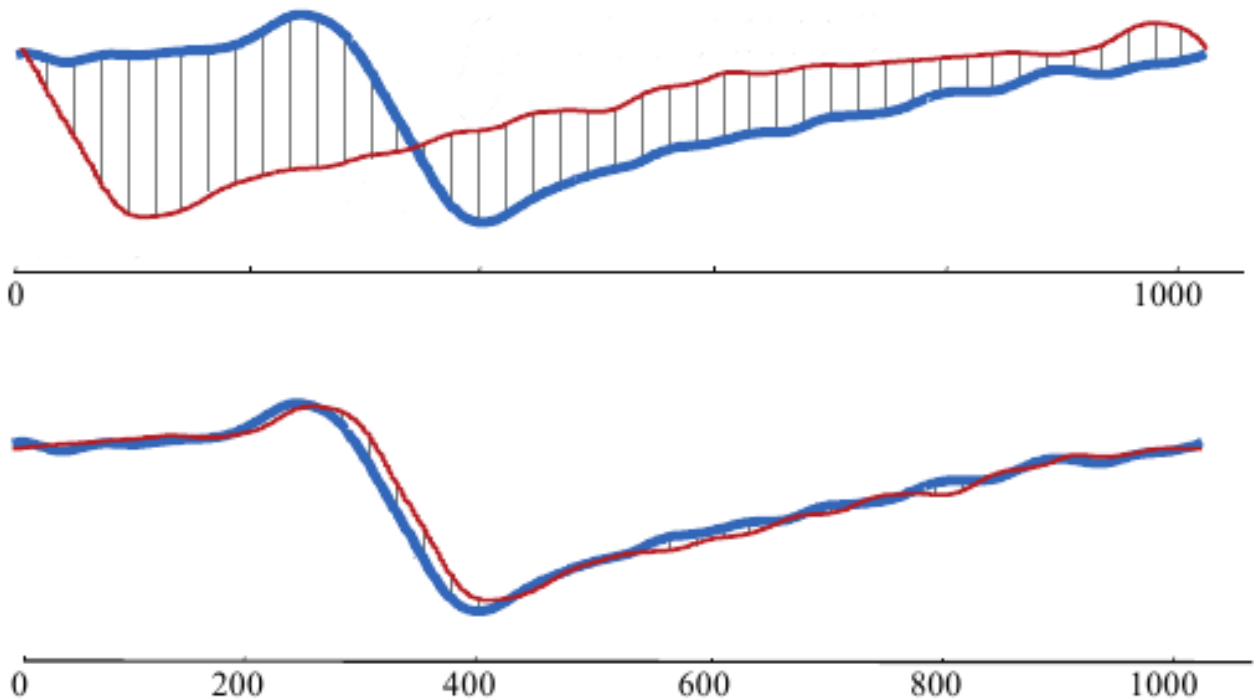


Figura 2.17: Duas curvas que após sucessivas tentativas de defasamento de uma delas, possuem casamento ótimo. Cada uma delas representa dados de radiação colhidos de uma estrela diferente. Adaptado de (Batista, Wang e E. J. Keogh 2011).

Invariância à oclusão

Algumas séries temporais podem apresentar subsequências que não possuem valores observados, por algum erro na medição, ou pela própria característica do problema. A invariância à oclusão pode ser obtida ao se ignorar determinadas partes das séries temporais que não possuem casamento. Algumas métricas de dissimilaridade como a DTW e EDR são mais robustas à oclusão do que a distância Euclidiana. A comparação entre as séries temporais geradas a partir das imagens de dois crânios, na qual se faz necessária a invariância à oclusão, pode ser vistas na Figura 2.18.

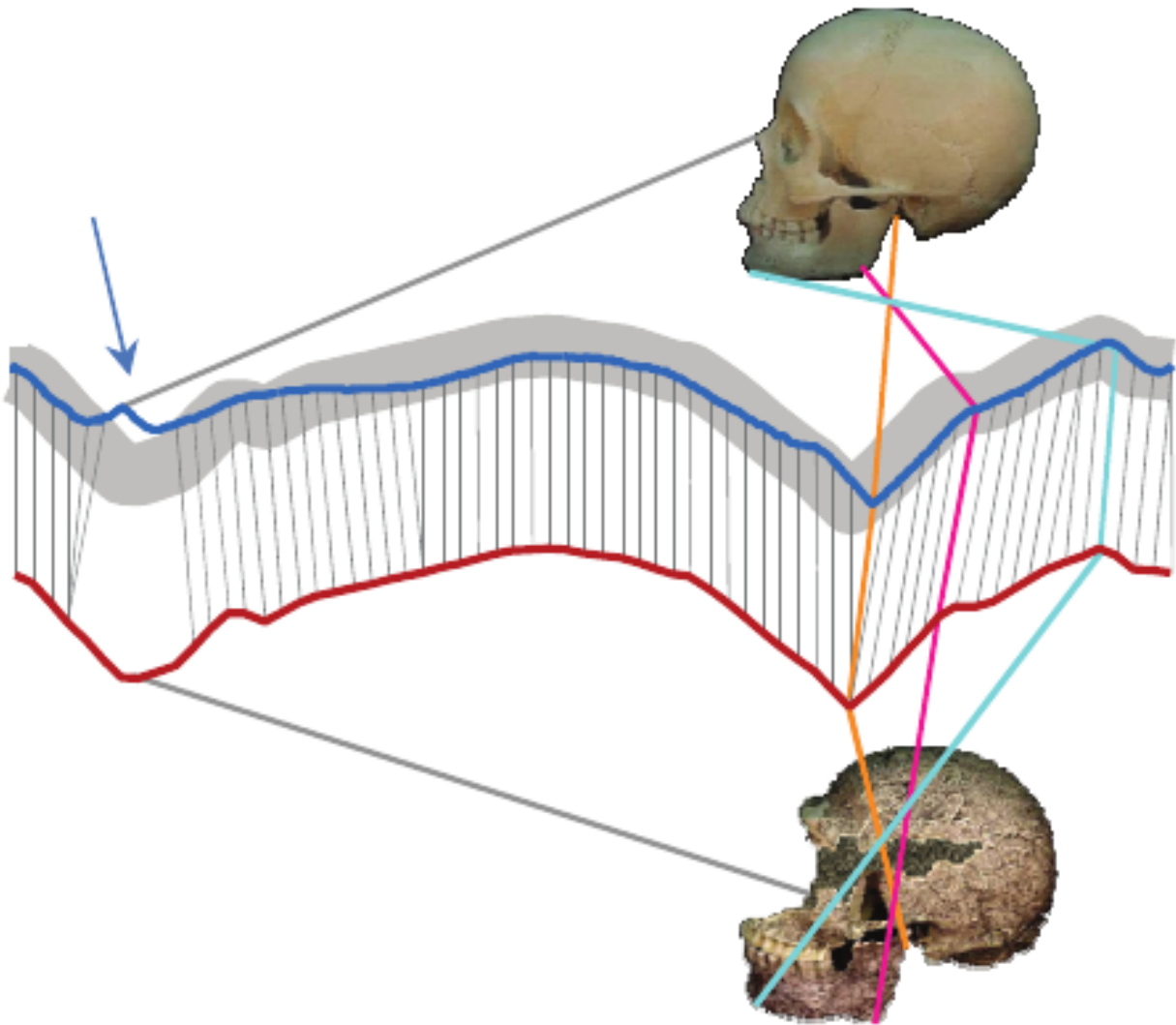


Figura 2.18: Duas curvas geradas a partir do formato de dois crânios a serem comparados. A falta da região do nariz de um dos crânios pode influenciar significativamente na comparação entre as métricas. Extraído de (Batista, Wang e E. J. Keogh 2011).

Invariância à complexidade

A complexidade de uma série temporal, como definida na seção 2.2.10, diz respeito ao número de picos e vales que a série apresenta. Observa-se que em diferentes domínios, instâncias de diferentes classes possuem complexidades distintas, o que torna tal atributo da série temporal relevante para as tarefas de classificação e agrupamento destas. A melhor maneira de se levar em conta as complexidades das séries temporais ao compará-las, é a utilização do fator de correção CID exposto na seção 2.2.10. As figuras 2.19 e 2.20 demonstram um problema no qual a invariância à complexidade influencia os resultados em uma tarefa de agrupamento.

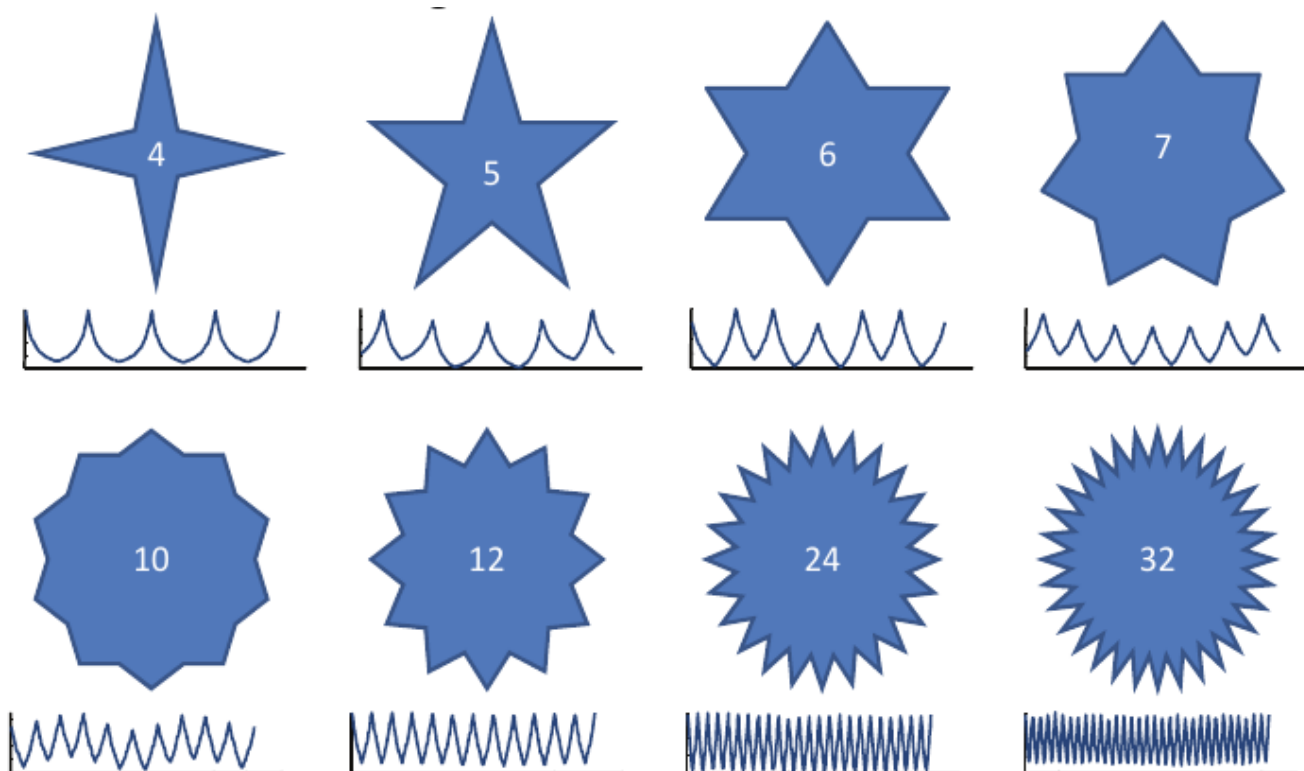


Figura 2.19: Diversas formas geométricas e as séries temporais geradas para cada uma delas. As curvas foram geradas pela distância do ponto central da figura até cada ponto do seu contorno. Extraído de (Batista, Wang e E. J. Keogh 2011).

Tabela 2.4: Sumário das métricas de dissimilaridades discutidas em 2.2.

Métrica	Distância?	Ordem de complexidade	Inv. Oclusão	Inv. Deslocamento	Inv. Complexidade	Ref.
Minkowski	✓	$\mathcal{O}(n)$				(Duda, Hart e Stork 2000)
DTW		$\mathcal{O}(n_1 * n_2)$		✓		(Berndt e Clifford 1994)
LB-Keogh		$\mathcal{O}(n)$		✓		(E. Keogh e Ratanamahatana 2005)
EDR		$\mathcal{O}(n_1 * n_2)$		✓		(L. Chen, Özsu e Oria 2005)
Dice		$\mathcal{O}(n)$				(Giusti e Batista 2013)
Jaccard		$\mathcal{O}(n)$				(Giusti e Batista 2013)
Lorentzian		$\mathcal{O}(n)$				(Giusti e Batista 2013)
Cosine		$\mathcal{O}(n)$				(Giusti e Batista 2013)
Mahalanobis		$\mathcal{O}(n^2)$				(Mahalanobis 1936)
Pearson		$\mathcal{O}(n)$				(Montero e Vilar 2014)
CORT		$\mathcal{O}(n)$				(Chouakria e Nagabhushan 2007)
CID		$\mathcal{O}(n)$			✓	(Batista, Wang e E. J. Keogh 2011)

2.3 Algoritmos de agrupamento

Dada uma métrica de dissimilaridade e um conjunto de dados, existem diferentes algoritmos que realizam a tarefa de agrupamento dos dados, e segundo (Rajaraman e Ullman 2011, p. 243) estes podem ser divididos em:

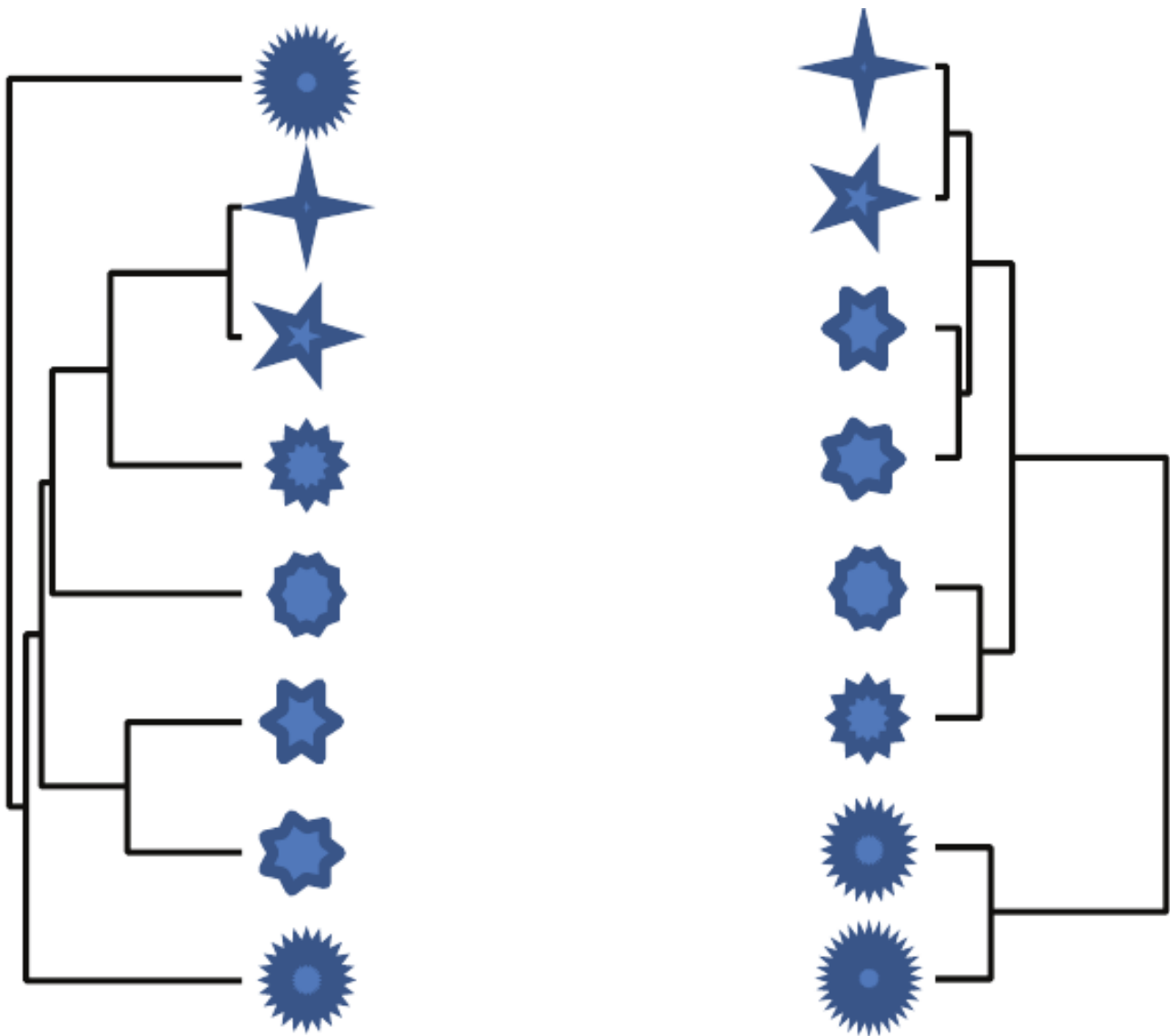


Figura 2.20: Agrupamentos hierárquicos com *linkagem* average realizados nas curvas; à esquerda com a distância euclidiana e à direita com a distância euclidiana corrigida pela CID. Extraído de (Batista, Wang e E. J. Keogh 2011).

1. algoritmos hierárquicos ou aglomerativos, nos quais, inicialmente, cada instância representa um grupo formado por apenas ela mesma, e a partir daí os grupos são combinados sucessivamente a partir de um critério que leva em conta a dissimilaridade entre os grupos,
2. algoritmos particionais, ou de atribuição de pontos, nos quais cada ponto, ou instância, é atribuído inicialmente à um grupo de uma solução inicial. Em seguida, os pontos são iterativamente realocados entre os grupos, até determinado critério de parada para a obtenção da partição final. Quando os pontos são atribuídos a somente um grupo, dá-se o nome de *hard-clustering*, enquanto quando atribui-se graus de pertinência do ponto à cada um dos

grupos, têm-se o *soft-clustering*, ou *fuzzy clustering*.

Os algoritmos de agrupamento podem ser, ainda, classificados em:

1. algoritmos que assumem que as instâncias se encontram no espaço euclidiano, ou seja, que a métrica de dissimilaridade utilizada é a distância Euclidiana. Essa distinção existe pois no espaço euclidiano existe o conceito de centróide, que é o ponto médio das instâncias que compõem o grupo, e este ponto é tido como o ponto mais representativo deste. Note que este ponto mais representativo não é, necessariamente, um ponto do conjunto de dados. Assim, em espaços não euclidianos, tal conceito de um ponto médio não existe e, dessa maneira, deve-se utilizar outras estratégias para se definir um ponto representativo dos grupos.
2. algoritmos que assumem que os dados são pequenos, o suficiente, no que diz respeito ao consumo de memória, para serem armazenados na memória RAM. Caso isso não seja verdade, outras estratégias já partem do pressuposto que é inviável mensurar a dissimilaridade entre todos os pontos, de maneira que diversos algoritmos já são descartados.

Além dos critérios supracitados, os algoritmos de agrupamento podem, ainda, serem classificados em: aqueles baseados em características, nos quais exigem como entrada uma matriz $N \times D$ contendo os dados, que são formados por N instâncias, sendo cada uma composta por D características, e os algoritmos de agrupamento baseados em dissimilaridades, os quais esperam uma matriz simétrica $N \times N$, onde o elemento ij desta matriz contém a dissimilaridade entre o i -ésimo e o j -ésimo elemento da matriz $N \times D$ que contém as instâncias. A seguir seguem breves comentários de alguns dos principais algoritmos de agrupamento utilizados no contexto de agrupamento de séries temporais.

2.3.1 Algoritmos particionais

k-means

O *k-means*, ou k-médias, ou algoritmo de Lloyd, é uma heurística que utiliza o conceito de centróide para a obtenção da partição final. Basicamente, dado um número k de grupos, o objetivo do algoritmo é encontrar uma partição dos dados que minimize o soma do quadrado da distância entre as instâncias de cada grupo e seu centróide. A resolução deste problema é NP-difícil, no entanto em (Lloyd 1982) é proposto um algoritmo de busca local que fez com que o *k-means* se tornasse um dos algoritmos de agrupamento mais populares.

O algoritmo *k-means* se encontra detalhado no pseudocódigo 2.1, e pelo fato de ser um algoritmo de busca local, deve ser executado diversas vezes, para tentar se chegar ao mínimo global. O *k-means* foi o precursor de diversas outras técnicas de agrupamento, que são variações dele ou outras técnicas realmente inspiradas no *k-means*. Dentre as principais podemos destacar: *k-means++* (Arthur e Vassilvitskii 2007), *k-medoids* (Kaufman e P.J. Rousseeuw 1987) e *Fuzzy-k-means* (J. C. Dunn 1973).

Uma das desvantagens do *k-means* é que se deve definir *a priori* o número de grupos da partição. Essa informação nem sempre se encontra disponível de antemão, o que faz com que sejam realizadas

partições com valores iterativos de k , e a partir dos valores dos índices de validação de agrupamento, pode-se ter uma ideia do melhor valor de k , dentro de determinado intervalo.

A utilização do k -means para o agrupamento de séries temporais fica limitada para os casos nos quais a distância euclidiana foi utilizada como métrica de dissimilaridade entre as instâncias, pois somente no espaço euclidiano faz sentido a utilização de pontos médios ou centroides. Em outras palavras, isso significa que várias das métricas promissoras detalhadas na seção 2.2, como a DTW, CID, entre outras, não podem ser utilizadas no agrupamento por meio do algoritmo k -means.

Algoritmo 2.1 Algoritmo k -means.

```

1: Entradas: número  $k$  de grupos, matriz  $X$  de dados com  $n$  instâncias e critério de
   parada  $\epsilon$ 
2: Saída: partição  $\mathcal{G}$  dos dados
3: for  $i \leftarrow 1$  to  $k$  do
4:    $j \leftarrow \text{rand}(0, n)$  // escolhe aleatoriamente um índice da matriz de dados
5:    $C_i \leftarrow x_j$  // define o centróide como a instância escolhida;
6:    $\mathcal{G}_i \leftarrow \emptyset$  // Inicia o  $i$ -ésimo grupo vazio
7: end for
8:  $\phi \leftarrow \epsilon$  // soma das distâncias de cada ponto ao seu centróide
9: while  $\phi \geq \epsilon$  do
10:  for  $p \leftarrow 1$  to  $n$  do
11:    for  $i \leftarrow 1$  to  $k$  do
12:       $d_i \leftarrow \text{dist}(x_p, C_i)$  // calcula a distância do ponto a cada centróide
13:    end for
14:     $s \leftarrow \arg \min_i (d_i)$  // Encontra o centróide mais próximo ao ponto  $p$ 
15:     $\mathcal{G}_s \leftarrow x_p$  // Associa o ponto  $p$  ao grupo do seu centróide mais próximo
16:  end for
17:   $\phi \leftarrow 0$  // soma das distâncias de cada ponto ao seu centróide
18:  for  $i \leftarrow 1$  to  $k$  do
19:     $C_i \leftarrow \text{centroid}(\mathcal{G}_i)$  // Atualiza o centróide do  $i$ -ésimo grupo
20:    for  $j \leftarrow 1$  to  $n$  do
21:      if  $x_j \in \mathcal{G}_i$  then
22:         $\phi \leftarrow \phi + \text{dist}(x_j, C_i)$  // soma das distâncias de cada ponto ao seu centróide
23:      end if
24:    end for
25:  end for
26: end while

```

K-medoids

Essencialmente, o k -medoids é uma variação do k -means, na qual a única diferença é que não se faz o uso de um ponto médio, ou centróide, das instâncias de um mesmo grupo, mas sim de um ponto mediano, ou medóide, que melhor representa as instâncias de cada grupo. O algoritmo se

encontra detalhado no pseudocódigo 2.2, onde pode ser visto que a única diferença em relação ao pseudocódigo 2.1 se encontra na linha 18, ao realizar a atualização dos medóides de cada grupo, ao invés da atualização dos centróides. Dessa maneira, diferentemente do *k-means*, o *k-medoids* consegue trabalhar com qualquer métrica de dissimilaridade, além da distância euclidiana.

O medóide pode ser definido de diversas maneiras, sendo a mais comum como a instância do grupo cuja a soma da distância, ou dissimilaridade, entre ela e as demais instâncias do mesmo grupo é mínima para aquele grupo (Rajaraman e Ullman 2011, p. 253). Então dado um conjunto de dados X composto por n instâncias $x_i, i = 0, \dots, n$, o medóide x_{med} é definido como

$$x_{med} = \arg \min_{x_i} \sum_{j=0}^n diss(x_i, x_j), \quad i = 0, \dots, n \quad (2.3.1)$$

Algoritmo 2.2 Algoritmo *k-medoids*.

```

1: Entradas: número k de grupos, matriz X de dados com n instâncias e critério de
   parada  $\epsilon$ 
2: Saída: partição  $\mathcal{G}$  dos dados
3: for  $i \leftarrow 1$  to  $k$  do
4:    $j \leftarrow rand(0, n)$  // escolhe aleatoriamente um índice da matriz de dados
5:    $C_i \leftarrow x_j$  // define o medóide como a instância escolhida;
6:    $\mathcal{G}_i \leftarrow \emptyset$  // Inicia o i-ésimo grupo vazio
7: end for
8:  $\phi \leftarrow \epsilon$  // soma das distâncias de cada ponto ao seu medóide
9: while  $\phi \geq \epsilon$  do
10:  for  $p \leftarrow 1$  to  $n$  do
11:    for  $i \leftarrow 1$  to  $k$  do
12:       $d_i \leftarrow dist(x_p, C_i)$  // calcula a distância do ponto a cada medóide
13:    end for
14:     $s \leftarrow \arg \min_i (d_i)$  // Encontra o medóide mais próximo ao ponto p
15:     $\mathcal{G}_s \leftarrow x_p$  // Associa o ponto p ao grupo do seu medóide mais próximo
16:  end for
17:   $\phi \leftarrow 0$  // soma das distâncias de cada ponto ao seu medóide
18:  for  $i \leftarrow 1$  to  $k$  do
19:     $C_i \leftarrow medoid(\mathcal{G}_i)$  // Atualiza o medóide do i-ésimo grupo
20:    for  $j \leftarrow 1$  to  $n$  do
21:      if  $x_j \in \mathcal{G}_i$  then
22:         $\phi \leftarrow \phi + dist(x_j, C_i)$  // soma das distâncias de cada ponto ao seu medóide
23:      end if
24:    end for
25:  end for
26: end while

```

2.3.2 Algoritmos hierárquicos

Os algoritmos hierárquicos, possuem a vantagem de não ser necessário informar, *a priori*, o número de grupos que os dados se encontram divididos, no entanto, em geral, possuem ordem de complexidade de $\mathcal{O}(n^2 \log(n))$, onde n é o número de instâncias, e esta complexidade é superior em relação aos algoritmos baseados em atribuição de pontos. Outra característica relevante dos algoritmos de agrupamento hierárquicos é que estes são determinísticos, ou seja, para uma mesma entrada, e mesmos parâmetros, sempre retornam a mesma partição. Eles conseguem trabalhar tanto com matrizes de características como com matrizes de dissimilaridade (Rajaraman e Ullman 2011, p. 245).

Os algoritmos hierárquicos podem ser divididos em algoritmos divisivos ou aglomerativos. Nos primeiros, inicialmente é considerado um grande grupo que contém todos os dados e que é iterativamente particionado até que cada instância esteja contida em grupo formado única e exclusivamente por ela mesma. Já nos algoritmos aglomerativos ocorre o processo inverso, no qual, inicialmente, cada instância está contida única e exclusivamente em seu próprio grupo, e iterativamente os grupos são fundidos para formar um novo grupo, até que se tenha somente um grande grupo contendo todos os dados. O pseudocódigo de um algoritmo hierárquico aglomerativo se encontra detalhado em 2.3.

O resultado de ambas as abordagens, é a estrutura de uma árvore binária chamada dendrograma, na qual as instâncias se encontram nas folhas da árvore e os ramos representam a união sucessiva das instâncias em um mesmo grupo. O tamanho do ramo que une dois grupos representa a distância, ou dissimilaridade, entre eles, de forma que ramos maiores representam uma maior distância entre os grupos. Dessa maneira, visualmente, pela altura dos ramos em um dendrograma, é possível se ter uma ideia do número de grupos que melhor representa a partição dos dados. Um exemplo de dendrograma pode ser visto na figura 2.21, o qual é resultado do agrupamento hierárquico com a *linkagem* average, que será melhor detalhada a seguir, e métrica de dissimilaridade euclidiana para o famoso conjunto de dados Iris (Fisher 1936).

Existem três variações principais na maneira de se definir a dissimilaridade entre os grupos, e cada uma delas pode acarretar em resultados bastante diferentes. A seguir seguem breves comentários de cada uma delas.

Na abordagem *Single link*, a distância entre dois grupos é definida como a menor distância entre duas instâncias de cada grupo. Por sua vez, na *Complete link*, a distância entre dois grupos é a maior distância entre duas instâncias de cada grupo. Assim, se o diâmetro de um grupo é definida como a maior distância entre instâncias contidas no grupo, então a abordagem *complete link* gera partições com grupos bastante compactos mas pouco dispersos, enquanto a estratégia *single link* cria partições com agrupamentos pouco compactos mas muito dispersos entre si.

Em uma tentativa de se alcançar as virtudes de ambos os métodos, foi proposta a *average link*, no qual a distância entre dois grupos é definida como a distância média das distâncias entre todos os pares de instâncias que formam o grupo. Na prática, a abordagem *average link* é a mais utilizada por obter partições que representam um compromisso entre compactação e dispersão dos dados.

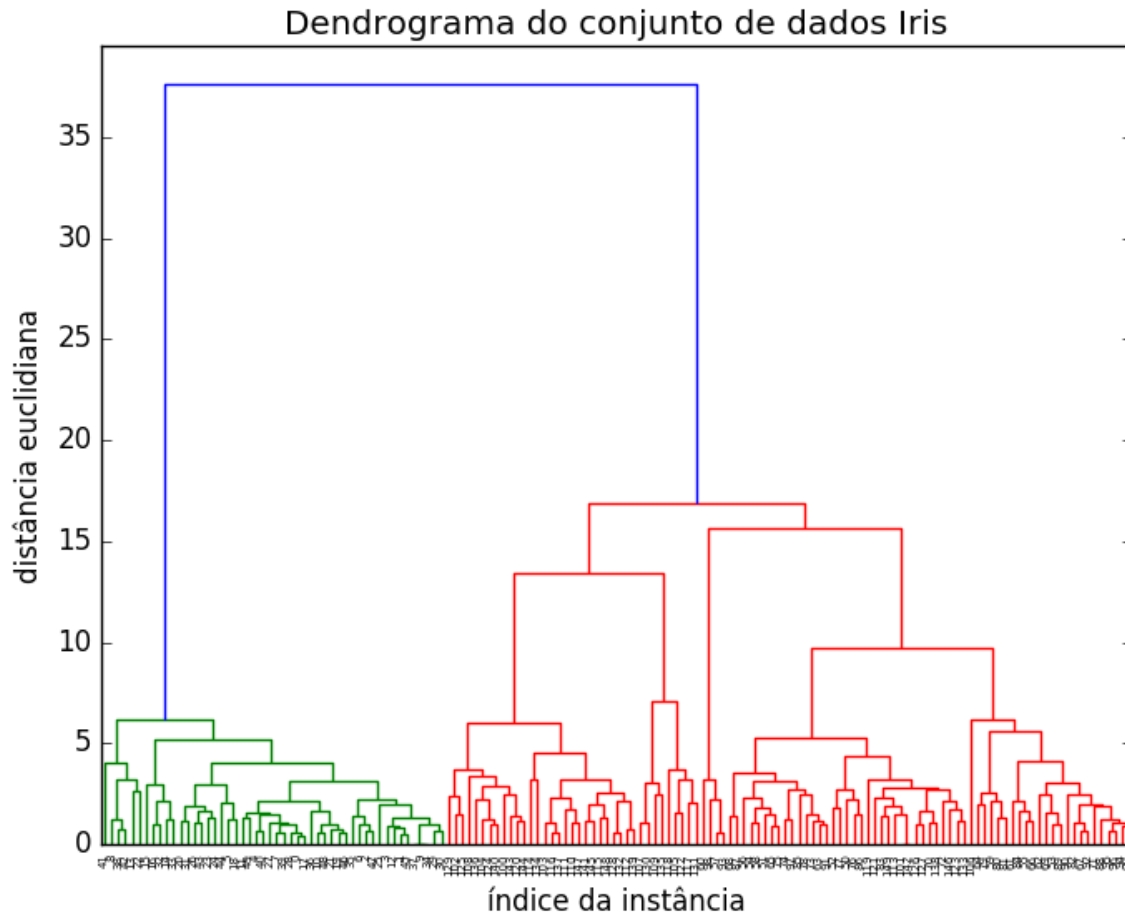


Figura 2.21: Exemplo de um dendrograma resultante do agrupamento hierárquico com a *linkagem* average e métrica de dissimilaridade euclidiana para o conjunto de dados Iris.

2.3.3 Algoritmos baseados em densidades

Os algoritmos particionais possuem a grande desvantagem de ser necessário o conhecimento, *a priori*, do número k de grupos que os dados se encontram naturalmente particionados. Dessa maneira, faz-se necessário rodadas iterativas com diferentes valores de k para se alcançar a partição desejada, o que em muitas aplicações, principalmente para aquelas com muitos dados, é proibitivo. Além disso, os algoritmos particionais têm performance degradada quando o formato dos grupos não é esférico. Por sua vez, os algoritmos hierárquicos possuem a deficiência da definição do ponto de poda do dendrograma para a obtenção dos grupos.

Em uma tentativa de sanar todos esses problemas foram criados os algoritmos baseados em densidades, nos quais regiões com alta densidade de instâncias se encontram os grupos e em regiões de baixa densidade, se encontram as regiões de separação destes. Essa visão é bastante intuitiva e pode ser generalizada para qualquer dimensão, onde a vizinhança de cada ponto do grupo, definida

Algoritmo 2.3 Agrupamento aglomerativo.

```
1: Entradas: matriz  $X$  de dados com  $n$  instâncias
2: Saída: partição  $\mathcal{C}$  dos dados
3: for  $i \leftarrow 1$  to  $n$  do
4:    $\mathcal{C}_i \leftarrow i$  //inicializa cada grupo;
5: end for
6:  $\mathcal{S} \leftarrow \{1, \dots, n\}$ ; // inicializa o conjunto de grupos a serem aglomerados;
7: while  $|\mathcal{S}| > 1$  do
8:    $(j, k) \leftarrow \arg \min_{j, k \in \mathcal{S}} d_{j, k}$ ; //Escolhe os dois grupos mais similares
9:    $\mathcal{C}_l \leftarrow \mathcal{C}_j \cup \mathcal{C}_k$ ; // Cria o novo grupo
10:  if  $\mathcal{C}_l \neq \{1, \dots, n\}$  then
11:     $\mathcal{S} \leftarrow \mathcal{S} \cup \{l\}$ ; // Marca o novo grupo como disponível para aglomeração
12:  end if
13:  foreach  $i \in \mathcal{S}$  do
14:     $d(i, l)$  ; // Atualiza a matriz de similaridade
15:  end for
16: end while
```

por certo raio, deve conter um número mínimo de pontos para este ser considerado pertencente ao grupo. O DBSCAN, *Density-based spatial clustering of applications with noise*, é o precursor desta classe de algoritmos e é detalhado a seguir.

DBSCAN

Proposto em (Ester et al. 1996), o DBSCAN é um algoritmo que consegue descobrir grupos de formas arbitrárias além de não ser necessário informar *a priori* o número de grupos. Os autores do algoritmo advogam que ele é menos sensível à ruídos e mais eficiente que os algoritmos hierárquicos e particionais. Todos esses ganhos foram conquistados com a introdução de dois parâmetros que não são necessários nos demais algoritmos: **Eps**, que é o raio máximo entre duas instâncias para elas serem consideradas da mesma vizinhança e **MinPts**, o número mínimo de pontos que devem estar na vizinhança para um ponto ser considerado pertencente ao grupo. Ambos os parâmetros influenciam fortemente nos resultados do agrupamento e estimativa deles é baseada no conhecimento do problema e dos dados do agrupamento, ou então em análises gráficas como pode ser visto em (Ester et al. 1996).

A abordagem mais comum é fixar o valor de **MinPts** em aproximadamente 4, e a partir daí traçar o gráfico *k-dist* das instâncias do conjunto de dados com $k = \text{MinPts}$. O gráfico *k-dist* é obtido pela ordenação das distância entre cada instância do conjunto de dados a ser agrupado e o seu k -ésimo vizinho mais próximo. A região do gráfico na qual ocorre um crescimento abrupto da curva, contém uma boa sugestão para **Eps**. O exemplo a seguir demonstra este procedimento.

Na figura 2.22 se encontra um conjunto de dados gerado sinteticamente composto por 2500 instâncias, onde se nota a presença de dois grupos distintos. A distância euclidiana entre cada instância e as demais foi calculada, e a distância da terceira instância mais próxima, ou a distância

do terceiro vizinho mais próximo, de cada instância foram armazenadas e em seguida ordenadas em ordem crescente. Ao gráfico obtido por estes valores ordenados dá-se o nome de k -*dist*, que pode ser visualizado na figura 2.23. Nesta figura fica evidente que para uma distância de aproximadamente 0.086, ocorre um ponto de inflexão do gráfico k -*dist*, o que sugere que este é um bom valor para *Eps*. Assim, com os valores de $MinPts=3$ e $Eps=0.086$, os dados foram agrupados pelo DBSCAN, e a partição resultante pode ser vista na figura 2.24. É notável que algumas poucas instâncias, destacadas em verde, foram consideradas anômalas, ou *outliers*, onde pôde-se verificar a robustez do DBSCAN frente a presença de dados ruidosos.

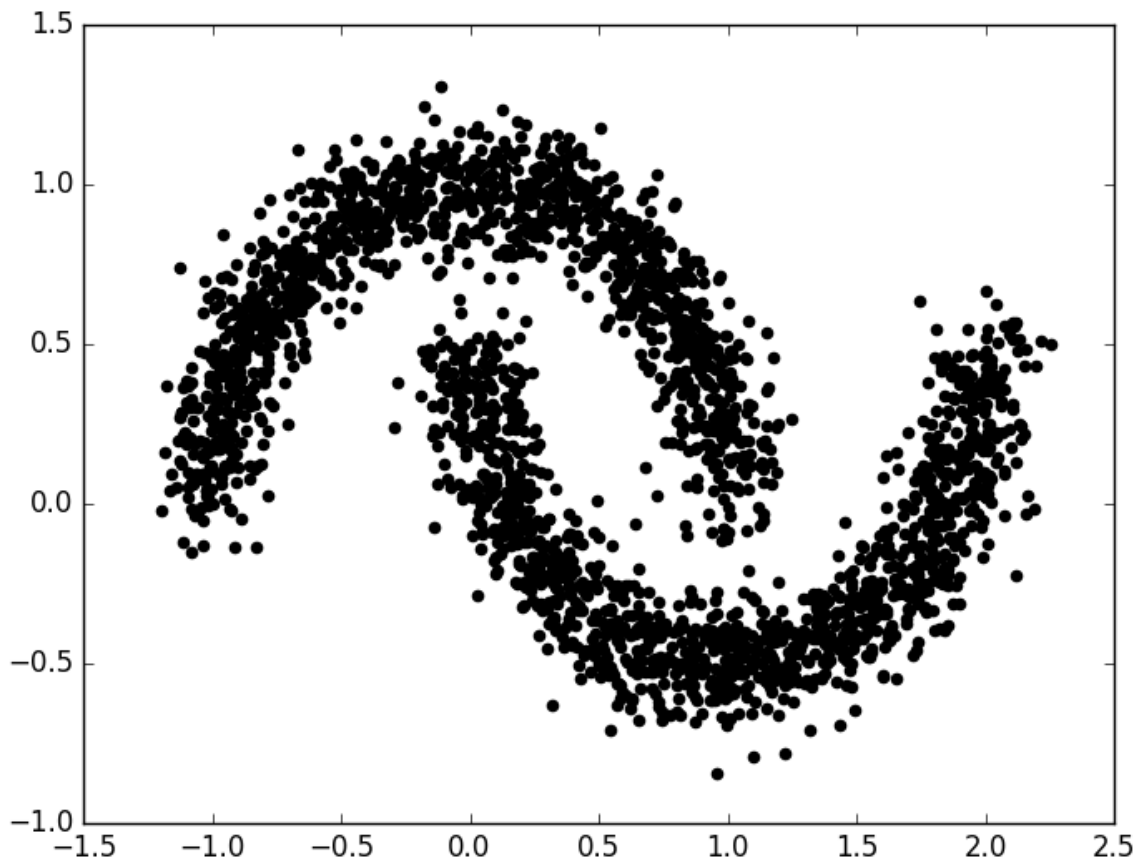


Figura 2.22: Conjunto de dados sintético formado por dois grupos.

Além do método demonstrado, existem outras tentativas para se estimar os melhores valores dos dois parâmetros, de forma automática, como em (Karami e Johansson 2014), no qual os valores são obtidos por meio de algoritmos de evolução diferencial. Nesta abordagem, cada indivíduo da população recebe um valor de **Eps** e **MinPts**, e uma partição por meio do DBSCAN é obtida para cada indivíduo. A função *fitness* do algoritmo de otimização é dada por índices de validação interna dos grupos, a serem detalhados na seção 2.4.1, e o algoritmo continua até um critério de



Figura 2.23: Gráfico k -dist do conjunto de dados da figura 2.22, onde o crescimento abrupto ocorre para a distância 0.086 aproximadamente.

parada. Dessa maneira, espera-se alcançar valores de **Eps** e **MinPts** que geram a partição com os melhores índices, que por sua vez refletem em uma melhor partição. Dessa maneira, pode-se afirmar que a abordagem não é muito diferente da estratégia iterativa de obtenção do valor de k dos algoritmos particionais, nos quais os algoritmos têm que ser executados diversas vezes. Já em (Ankerst et al. 1999), é proposto o algoritmo OPTICS, *Ordering Points to Identify the Clustering Structure*, no qual se faz necessária somente a informação do valor de **MinPts**, sendo o **Eps** estimado pelo próprio algoritmo.

Apesar de no exemplo apresentado a distância euclidiana ter sido utilizada, o DBSCAN pode receber como parâmetro uma matriz de dissimilaridade gerada por qualquer outra dissimilaridade, o que, no contexto de séries temporais, o torna bastante interessante. O custo computacional do DBSCAN é de $\mathcal{O}(n \log n)$.

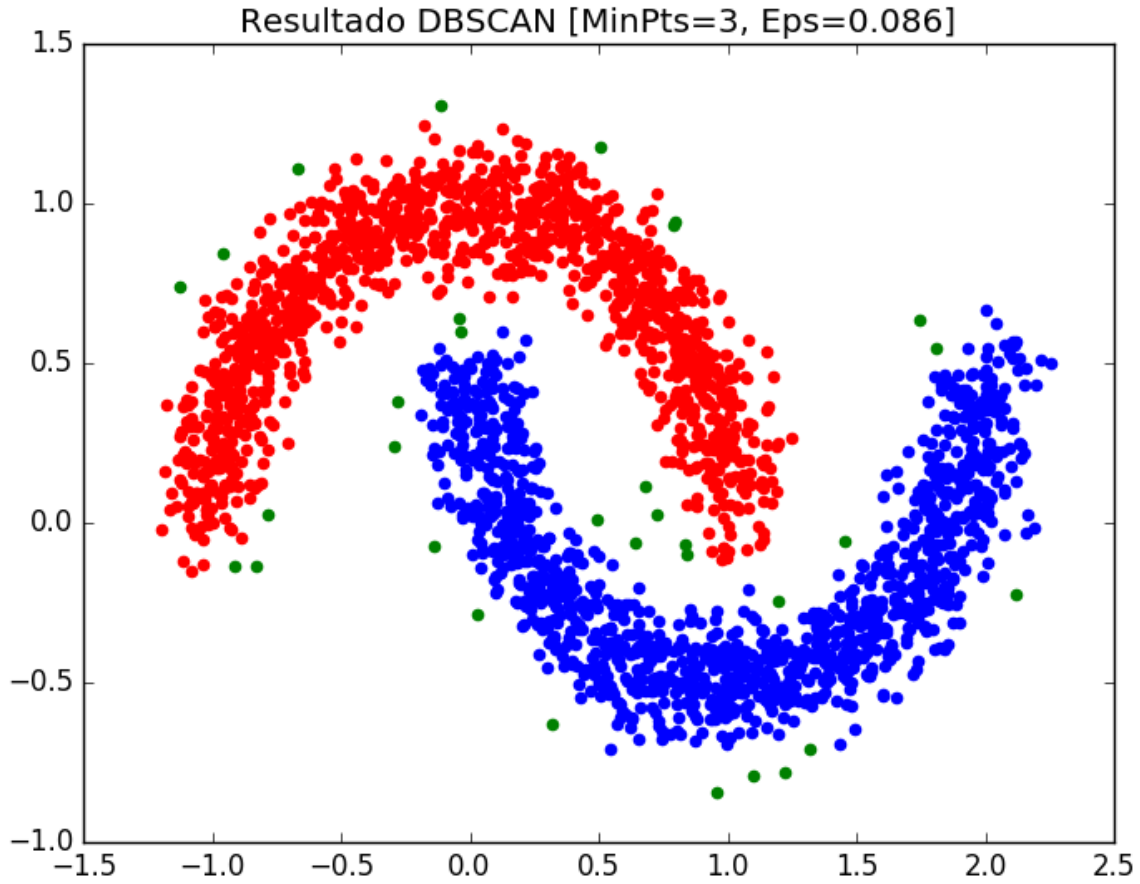


Figura 2.24: Partição dos dados da figura 2.22 obtida pelo DBSCAN com $MinPts=3$ e $Eps=0.086$. O pontos verdes foram considerados *outliers* pelo algoritmo.

2.4 Índices de validação de agrupamentos

Após a definição de uma métrica de dissimilaridade, da escolha de um algoritmo de agrupamento, e da realização do agrupamento em si, faz-se necessária a avaliação da partição obtida. Tal avaliação é realizada pelos índices de validação que podem ser divididos em dois grandes grupos, sendo o primeiro formado pelos índice de validação interna enquanto o segundo contém os índices de validação externa.

Os índices de validação interna são utilizados quando não se dispões de informações externas dos dados, e dessa maneira, ele é obtido apenas pelo resultado do agrupamento e das dissimilaridades entre as instâncias. Por sua vez, os índices de validação externos, utilizam alguma outra informação, além do resultado do agrupamento, no cálculo do índice. Tal informação externa é o resultado esperado do agrupamento, ou seja, os rótulos de cada instância. Uma discussão mais profunda de cada um dos tipos de índices de validação de agrupamentos é realizada a seguir.

2.4.1 Índices de validação interna

Normalmente, nas tarefas de agrupamento de dados, os índices de validação interna são os mais utilizados, pelo fato de não se possuir *a priori* a rotulação das instâncias. Os índices de validação internos podem ser úteis também para a definição de um valor k que represente o número de grupos, no qual os dados se encontram naturalmente distribuídos. O número pode ser aproximado ao se obter diversas partições para valores iterativos de k , e a partição que acarretar no melhor valor do índice de validação interno será a que possui o valor de k recomendado.

Basicamente duas características são mensuradas pelos índices de validação interna: a **compactação** dos grupos, ou seja, o quão próximas, ou similares, as instâncias contidas em um mesmo grupo são entre si, e a **separação** dos grupos, ou seja, o quão distantes, ou dissimilares as instâncias de diferentes grupos são entre si. Cada uma das estratégias descritas a seguir calcula e pondera de formas distintas essas duas características para a obtenção do índice de validação dos resultados do agrupamento.

Silhouette

O silhouette é um índice de validação proposto por (Peter Rousseeuw 1987a) que tenta avaliar o quão compacta, ou quão similar, cada instância se encontra dentro de seu próprio grupo, e, simultaneamente, o quão distante, ou dissimilar, ela é da instância mais próxima, ou mais similar, à ela e que não pertence ao seu mesmo grupo.

Para cada instância x_i do conjunto de dados X contendo n instâncias, agrupada no grupo $\mathcal{C}_k$, devem ser calculados dois valores, sendo o primeiro a média da métrica de dissimilaridade entre x_i e as demais instâncias contidas em $\mathcal{C}_k$, denominado a , e o segundo como o valor mínimo dentre todas as dissimilaridades entre x_i e as demais instâncias não contidas em $\mathcal{C}_k$, denominado b . O valor de silhouette S_i do ponto x_i é definido como:

$$S_i = \frac{(b - a)}{\max(a, b)}, \text{ onde} \quad (2.4.1)$$

$$a = \frac{1}{n} \sum_j^{|C_k|} \text{dist}(x_i, x_j); x_j, x_i \in \mathcal{C}_k, \quad (2.4.2)$$

$$b = \min(\text{dist}(x_i, x_j)); x_i \in \mathcal{C}_k, x_j \notin \mathcal{C}_k, \quad (2.4.3)$$

sendo $\text{dist}(x_i, x_j)$ o valor da métrica de dissimilaridade entre as instâncias x_i e x_j .

Os valores de S_i variam de -1 a 1 , sendo o valor de -1 geralmente quando a atribuição da instância x_i ao seu grupo é equivocada, e 1 quando é acertada. Valores próximos de 0 podem indicar sobreposição de grupos. O valor de silhouette da partição é definido como a média dos valores de cada instância do conjunto de dados:

$$S = \frac{1}{n} \sum_i^n S_i. \quad (2.4.4)$$

Valores próximos de 1 indicam uma boa partição, ao passo que valores próximos de -1 indicam uma partição ruim.

Índice de Dunn

Proposto por (J. C. Dunn 1974), o índice de Dunn é definido como a razão entre a mínima distância intragrupo e a máxima intergrupo. Sendo assim, dada uma partição com K grupos, sendo $\mathcal{C}_k$ o k -ésimo grupo dessa partição, então $dist_{kk'}$ é definida como a menor distância entre cada uma das n_k e $n_{k'}$ instâncias dos grupos $\mathcal{C}_k$ e $\mathcal{C}_{k'}$ respectivamente:

$$dist_{kk'} = \min_{i=1, \dots, n_k, j=1, \dots, n_{k'}} \left(dist(x_i^k, x_j^{k'}) \right), \quad (2.4.5)$$

e d_{min} é a menor destas distâncias:

$$d_{min} = \min_{k \neq k'} dist_{kk'}. \quad (2.4.6)$$

O diâmetro do grupo $\mathcal{C}_k$, é definido como a maior distância entre os distintos n_k pontos contidos pelo grupo $\mathcal{C}_k$:

$$D_k = \max_{x_i, x_j \in \mathcal{C}_k, x_i \neq x_j} dist(x_i, x_j), \quad (2.4.7)$$

e finalmente, o maior diâmetro da partição é definido como d_{max} :

$$d_{max} = \max_{1 \leq k \leq K} D_k. \quad (2.4.8)$$

O índice *Dunn* é definido pela razão entre d_{min} e d_{max} :

$$Dunn = \frac{d_{min}}{d_{max}} \quad (2.4.9)$$

O índice *Dunn* varia de $[0, \infty)$ e valores menores implicam em partições melhores.

Davies-Bouldin Index

Proposto em (Davies e Bouldin 1979), o índice Davies-Bouldin é obtido da seguinte maneira: dada uma partição com K grupos e sendo G_k o ponto definido como o ponto mais representativo do k -ésimo grupo, onde por mais representativo entende-se o centróide no espaço euclidiano e o medóide para outras métricas de dissimilaridade, então a distância média δ_k entre cada instância e seu ponto mais representativo é definida como:

$$\delta_k = \frac{1}{n_k} \sum_i^{n_k} dist(x_i, G_k), \quad (2.4.10)$$

onde n_k é número de instâncias contidas no k -ésimo grupo e x_i é a i -ésima instância do grupo k . Essa métrica de compactação de grupo deve ser calculada para cada uma dos k grupos da partição, e a dispersão entre os grupos é calculada entre cada um dos pares de grupos que formam a partição:

$$\Delta_{kk'} = dist(G_k, G_{k'}). \quad (2.4.11)$$

Finalmente, o índice Davies-Bouldin é definido como:

$$DBI = \frac{1}{K} \sum_{k=1}^K \max_{k \neq k'} \left(\frac{\delta_k + \delta'_k}{\Delta_{kk'}} \right). \quad (2.4.12)$$

O DBI é um índice de validação interna definido em $[0, \infty)$, e quanto maior o seu valor melhor é considerada a partição.

Índice de Calinski-Harabasz

Em (Caliński e Harabasz 1974) é proposto um índice de validação para a identificação de grupos em um espaço euclidiano. Para tal ele define uma métrica que pondera as variâncias intragrupo WGSS (*within-group sum of square distance*), entendida como a variância da distância entre o centróide de um grupo e as suas instâncias, e a dispersão entre os grupos BGSS (*between group sum of square distances*), entendida como a dispersão entre os centróides de cada grupo e o centróide de todo o conjunto de dados. Os valores obtidos são ponderados em função do número de grupos k . Então, o índice de Calinski-Harabasz é definido como

$$CH = \frac{\frac{BGSS}{k-1}}{\frac{WGSS}{n-k}}, \quad (2.4.13)$$

sendo

$$BGSS = \sum_i n_i d^2(c_i, c) \quad (2.4.14)$$

e

$$WGSS = \sum_i \sum_{x \in C_i} d^2(x, c_i), \quad (2.4.15)$$

onde c_i e c são os centróides do i -ésimo grupo e o centróide de todo o conjunto de dados respectivamente, e $d(x_1, x_2)$ é a distância euclidiana entre os pontos x_1 e x_2 .

Apesar de ter sido definido para o espaço euclidiano, os próprios autores em (Caliński e Harabasz 1974) fazem menção à possibilidade de extensão do índice a espaços não-euclidianos. Para tal, basta utilizar outros conceitos como o medóide, ao invés do centróide, e a própria distância não-euclidiana entre os medóides e as instâncias ao invés da distância euclidiana entre os centróides e as instâncias.

Uma partição ideal tem alta dispersão entre os grupos (BGSS), e baixa variância intragrupo (WGSS), de forma que pela equação 2.4.13 fica evidente que valores grandes de CH são desejados. O índice em questão é definido no intervalo $]0, \infty)$.

2.4.2 Índices de validação externa

Em tarefas reais de agrupamento, os índices de validação externa não têm utilidade direta, pois como já dito, em tarefas de agrupamento não se dispõe *a priori* do rótulo das instâncias. Eles são muito úteis para se validar parâmetros e estratégias de agrupamento, ao se obter um valor

numérico que faça com que seja possível comparar o resultado obtido de cada abordagem com o resultado esperado.

Assim, variando-se os algoritmos, métricas, estratégias de pré-processamento e outros parâmetros ou decisões da tarefa de agrupamento de determinado conjunto de dados rotulado, pode-se comparar cada partição obtida com a partição real, ou verdadeira. Ou seja, caso se deseje realizar o agrupamento de dados não rotulados, e não se sabe a melhor estratégia para tal, se, primeiramente, o agrupamento for feito em uma base de dados que seja, de alguma forma, semelhante à base alvo e a primeira esteja rotulada, então os índices de validação externa podem ser úteis na obtenção de uma estratégia mais adequada para a base não rotulada.

Existem diversos índices de validação externa tais como o *Adjusted Rand Index* (ARI) (Hubert e Arabie 1985), *Adjusted Mutual Info* (AMI) (Vinh, Epps e Bailey 2010), *V-measure* (Rosenberg e Hirschberg 2007), *Variation of Information* (VI) (Meilă 2007), entre outros. Neste trabalho, o índice VI foi especialmente utilizado, pelo fato de, diferentemente dos demais, ser uma métrica no espaço de partições. Dada a sua importância, ele se encontra detalhado na seção seguinte.

Variation of Information - VI

Proposto em (Meilă 2007), e baseado em técnicas da teoria da informação, o VI mede a quantidade de informação ganha ou perdida ao se mudar da partição $\mathcal{P}$ para a partição $\mathcal{P}'$. O VI possui a propriedade de ser uma métrica de distância no espaço de partições, ou seja, obedece aos quatro axiomas apresentados no início da seção 2.2. Essa propriedade faz com que seja possível, por exemplo, realizar o agrupamento de partições.

Assim, dada uma partição $\mathcal{P}$ formada por n instâncias divididas em k grupos, onde o k -ésimo grupo contém n_k instâncias, podemos definir uma variável aleatória como a probabilidade de se escolher uma instância do k -ésimo grupo e a entropia da partição como:

$$P(k) = \frac{n_k}{n} \text{ e,} \quad (2.4.16)$$

$$H(\mathcal{P}) = - \sum_i^k P(k) \log P(k). \quad (2.4.17)$$

Sendo $P(k)$ e $P'(k')$ as variáveis aleatórias associadas às partições $\mathcal{P}$ e $\mathcal{P}'$ respectivamente. Então, pode-se definir a probabilidade de uma instância pertencer ao grupo $\mathcal{C}_k$ da partição $\mathcal{P}$ e ao grupo $\mathcal{C}'_{k'}$ da partição $\mathcal{P}'$ como:

$$P(k, k') = \frac{|\mathcal{C}_k \cap \mathcal{C}'_{k'}|}{n}, \quad (2.4.18)$$

e a informação mútua entre as partições $\mathcal{P}$ e $\mathcal{P}'$ como:

$$I(\mathcal{P}, \mathcal{P}') = \sum_i^k \sum_i^{k'} P(k, k') \log \frac{P(k, k')}{P(k)P'(k')}. \quad (2.4.19)$$

Finalmente, a variação da informação entre as partições $\mathcal{P}$ e $\mathcal{P}'$ pode ser definida como:

$$VI(\mathcal{P}, \mathcal{P}') = H(\mathcal{P}) + H(\mathcal{P}') - 2I(\mathcal{P}, \mathcal{P}'). \quad (2.4.20)$$

Pelo fato do índice em questão ser uma métrica no espaço de partições, se soubermos a partição esperada, ou verdadeira, de um conjunto de dados, denominada $\mathcal{P}_v$, e possuímos duas partições de teste $\mathcal{P}_{t1}$ e $\mathcal{P}_{t2}$, a partição de teste que possuir menor VI em relação à partição verdadeira, é considerada uma melhor partição dos dados do que a outra partição de teste. Ou seja,:

$$\begin{cases} \text{se } VI(\mathcal{P}_v, \mathcal{P}_{t1}) < VI(\mathcal{P}_v, \mathcal{P}_{t2}), \text{ então } \mathcal{P}_{t1} \text{ é melhor, ou mais verossímil, que } \mathcal{P}_{t2}, \\ \text{se } VI(\mathcal{P}_v, \mathcal{P}_{t1}) > VI(\mathcal{P}_v, \mathcal{P}_{t2}), \text{ então } \mathcal{P}_{t2} \text{ é melhor, ou mais verossímil, que } \mathcal{P}_{t1}, \\ \text{se } VI(\mathcal{P}_v, \mathcal{P}_{t1}) = VI(\mathcal{P}_v, \mathcal{P}_{t2}), \text{ então } \mathcal{P}_{t2} \text{ e } \mathcal{P}_{t1} \text{ são aproximações equivalentes de } \mathcal{P}_v. \end{cases} \quad (2.4.21)$$

Tais propriedades fazem com que o índice de variação da informação seja extremamente interessante para se construir um ranking entre diversas variações de estratégias de agrupamento para uma mesma base de dados rotulada. Dessa maneira, o VI será utilizado nos testes teóricos a serem realizados no Capítulo 3.

O valores de VI variam de 0, quando $2I(\mathcal{P}, \mathcal{P}') = H(\mathcal{P}) + H(\mathcal{P}')$, o que ocorre somente quando $\mathcal{P} = \mathcal{P}'$, até $H(\mathcal{P}) + H(\mathcal{P}')$, o que ocorre quando $I(\mathcal{P}, \mathcal{P}') = 0$. Dessa maneira, é possível que o índice VI seja normalizado, dividindo o valor obtido pelo seu valor máximo, que como mencionado é a soma da entropia de cada partição, confinando-o, assim, no intervalo $[0, 1]$. No entanto, quando este procedimento é realizado o índice perde a sua característica de métrica no espaço de partições, como provado em (Meilă 2007).

2.5 Conclusões do Capítulo

Neste capítulo foram apresentados as diversas escolhas que podem ser feitas em tarefas de agrupamento de séries temporais. Inicialmente, foram apresentadas as possíveis operações a serem realizadas em uma fase de pré-processamento como a normalização dos dados, redução de dimensionalidade ou transformadas para se trabalhar no domínio da frequência. Em seguida, foram apresentadas diversas métricas de dissimilaridades e em seguida as possíveis invariâncias desejadas em tarefas de mineração de dados de séries temporais foram enumeradas e exemplificadas. Por fim, os algoritmos mais comuns utilizados no agrupamento de séries temporais, e as estratégias de avaliação de partições, que são realizados por meio de índices internos e externos, foram discutidas.

Após esta revisão acerca do tema de mineração de dados de séries temporais, e, especificamente no contexto de agrupamento, tornou-se evidente a complexidade do assunto, dado, principalmente, às diversas possibilidades de se realizar a tarefa de agrupamento que surgem a partir das diversas combinações possíveis de pré-processamento, métrica de dissimilaridade, algoritmo de agrupamento e método de avaliação do resultado. A cada combinação dessas escolhas dá-se o nome de estratégia de agrupamento e, uma vez que tal diversidade de estratégias torna difícil a proposição de uma estratégia de agrupamento genérica, faz-se necessário um estudo detalhado e restrito a um número reduzido de escolhas mais promissoras para avaliar a performance de tais escolhas e a consequente proposição de uma estratégia.

Tal estudo foi realizado e os resultados, comentários e conclusões se encontram no Capítulo seguinte (3).

Capítulo 3

Testes em bases de dados *benchmark*

Neste capítulo são apresentados resultados de experimentos com conjuntos de dados de séries temporais que se encontram rotulados e estão disponíveis em (Y. Chen et al. 2015). Cada uma das 85 bases do conjunto de dados pode ser visualizada no Capítulo A, onde estão detalhados os medóides e centróides de cada grupo que forma a partição real dos dados, obtidos por meio da distância euclidiana. Pela análise de tais figuras, pode-se perceber que existem determinados conjuntos de dados que a distinção entre as instâncias de grupos diferentes é extremamente árdua (vide Figura A.11) ao passo que para outros conjuntos esta é uma tarefa relativamente simples (vide Figura A.6).

A heterogeneidade das bases que compõem o UCR *Time Series* é notável no que diz respeito ao domínio do problema já que existem curvas que são: sinais biológicos, geradas em problemas de visão computacional, sinais de equipamentos elétricos, sinais multivariados, dentre outros. Ademais, a diversidade no que diz respeito ao número de instâncias, grupos e tamanho das séries temporais também é bastante significativa.

Se considerarmos que no Capítulo 2:

- apresentou-se 5 tipos possíveis de pré-processamento: normalização Z, normalização max, normalização min-max, transformada discreta de Fourier e *Piecewise aggregate approximation*, desconsiderando-se as possíveis combinações entre eles, como, por exemplo, *Piecewise aggregate approximation* e normalização Z,
- discutiu-se 12 métricas de dissimilaridades puras (euclidiana, *manhattan*, *chebyshev*, DTW, LB-Keogh, EDR, Lorentzian, *dice*, *jaccard*, *pearson*, *mahalanobis*, *cosine*) além das versões corrigidas pela correlação temporal (CORT) e complexidade (CID) de cada uma delas, existem 36 métricas de dissimilaridades possíveis,
- analisou-se 7 algoritmos diferentes, sendo 3 hierárquicos (cada um com uma das *linkagens* *average*, *complete* e *single*), 3 particionais (*k-means*, *k-means++* e *k-medoids*) e um baseado em densidades (DBSCAN),

então, dentre estas opções, existe um limite de $5 * 36 * 7 = 1260$ maneiras distintas de-se realizar o agrupamento de dados de séries temporais. Este número é, na verdade, menor, devido ao fato de

algoritmo *k-means* fazer uso somente da distância euclidiana e de algumas dissimilaridades como a DTW, EDR, dentre outras serem inaplicáveis após a transformação dos dados pela transformada de Fourier.

No entanto, este número demonstra bem o grande número de caminhos possíveis a serem seguidos e a dificuldade enfrentada por um projetista quando ele deve escolher um destes caminhos para realizar a tarefa de agrupamento de séries temporais. Testes iniciais foram feitos com as diversas possibilidades enumeradas, no entanto a análise dos resultados obtidos se tornou árdua, pois, dado o grande número de estratégias testadas, não ficou claro se alguma delas, ou algum grupo delas, se sobressaiu em relação às demais de forma significativa considerando-se todas as bases de dados. Por isto, o número de possibilidades de agrupamento das bases de dados do UCR *Time Series* neste capítulo é significativamente reduzido, e somente as escolhas mais comuns na literatura serão estudadas a fundo e comparadas entre si.

Pelo fato de todas as curvas do conjunto de dados já se encontrarem normalizadas pela normalização Z, e de estudos afirmarem que, em tarefas de classificação ela é superior às demais (Rakthanmanon et al. 2012), ela é a única escolha de pré-processamento a ser utilizada em todos os experimentos deste Capítulo. No que diz respeito à escolha de métrica de dissimilaridade, a distância euclidiana foi escolhida por ser uma das métricas mais utilizadas em agrupamentos de curva de carga (Chicco 2012), além da DTW por esta ser recorrente na literatura e ter apresentado performance significativamente superior à euclidiana nos testes realizados em (Giusti e Batista 2013). Além das distâncias puras, as respectivas versões que levam em conta a correlação temporal das séries (CORT) e complexidade relativa entre elas (CID) também serão utilizadas para se verificar eventuais ganhos com a aplicação destas.

Dentre os algoritmos estudados, serão testados os algoritmos hierárquicos com *linkagens average* e *complete*, além dos algoritmos particionais *k-means* e *k-medoids*. O *k-means* foi utilizado por ser, assim como a distância euclidiana, onipresente em agrupamento de curvas de carga (Chicco 2012), e a sua variação, o *k-medoids* para que seja possível a realização do agrupamento por meio da combinação de um algoritmo particional e uma métrica de dissimilaridade não-euclidiana como a DTW, por exemplo. Já os algoritmos hierárquicos foram utilizados por apresentarem performance superior aos algoritmos particionais nos experimentos apresentados em (Paparrizos e Gravano 2016). Além da análise das métricas mais promissoras, já mencionadas, todas elas serão confrontadas com a estratégia derivada da combinação do algoritmo *k-means* e distância euclidiana já que, como já dito, esta é a estratégia padrão no agrupamento de curvas de carga.

Os experimentos deste Capítulo foram conduzidos com a finalidade de responder as seguintes perguntas:

- qual métrica de dissimilaridade é a mais apropriada para séries temporais no que diz respeito à qualidade da partição e custo computacional?
- qual é o algoritmo de agrupamento mais apropriado para a realização do agrupamento de séries temporais?
- qual índice de validação interno é capaz de nos indicar as melhores partições de dados no agrupamento de séries temporais?

São realizados dois experimentos, sendo o primeiro destinado a responder às duas primeiras perguntas, enquanto o segundo é destinado a responder à terceira pergunta. Em todos experimentos são realizados os testes estatísticos de significância não-paramétricos de Friedman, Nemenyi e Bonferroni-Dunn para a comparação de diferentes estratégias de agrupamento aplicadas a diversas bases de dados, como sugerido em (Demšar 2006) e detalhado na seção 3.1.

Todos experimentos apresentados nesta dissertação de mestrado foram realizados em uma máquina com CPU Intel(R) Core(TM) i5-2400 3.10GHz e 8GB de RAM. Os códigos foram feitos em python onde foram utilizados, principalmente, os módulos numpy (Walt, Colbert e Varoquaux 2011), scipy (Jones, Oliphant, Peterson et al. 2001), Orange3 (Demšar et al. 2013) e scikit-learn (Pedregosa et al. 2011) e os códigos utilizados podem ser encontrados em (Pazzini 2016).

3.1 Testes estatísticos de significância

No teste de Friedman (Friedman 1940), dado M conjuntos de dados e L algoritmos, ou estratégias, com seus rankings médios R_t , onde $t = 1, \dots, L$, assume-se a hipótese nula H_0 de que os algoritmos, ou estratégias, são equivalentes. Neste caso, a estatística

$$F_F = \frac{(M-1)\chi_F^2}{M(L-1) - \chi_F^2} \quad (3.1.1)$$

é distribuída de acordo com a distribuição F com $L-1$ e $(L-1)(M-1)$ graus de liberdade, onde

$$\chi_F^2 = \frac{12M}{L(L+1)} \left(\sum_t R_t^2 - \frac{L(L+1)^2}{4} \right). \quad (3.1.2)$$

Caso a hipótese nula H_0 seja rejeitada pelo teste de Friedman, os testes *post-hoc* de Nemenyi e Bonferroni-Dunn podem ser aplicados para apontar significância na comparação dos algoritmos, ou estratégias, um-contr-a-um e um-contr-a-todos respectivamente.

O teste de Nemenyi (Nemenyi 1963) é utilizado quando se deseja realizar a comparação entre as estratégias para se verificar quais são equivalentes entre si. A performance entre duas estratégias é significativamente diferente se os rankings médios delas diferirem de, no mínimo,

$$CD = q_\alpha \sqrt{\frac{L(L+1)}{6M}}, \quad (3.1.3)$$

onde o valor crítico q_α é função do número de algoritmos e pode ser consultado em (Demšar 2006).

No entanto, caso se deseje comparar a performance de todas as estratégias em análise, com a performance de uma estratégia de controle, o método de Bonferroni-Dunn (O. J. Dunn 1961) deve ser utilizado, ao invés do teste de Nemenyi. O teste define quais as estratégias possuem performance significativamente equivalente à performance da estratégia de controle, e estas são aquelas estratégias cujo ranking médio se encontra dentro da faixa definida pelo valor de uma diferença crítica do teste de Bonferroni-Dunn centrada no ranking médio da estratégia de controle. Tal diferença crítica é calculada conforme equação 3.1.3, sendo seu valor diferenciado do valor da diferença crítica do teste de Nemenyi apenas pelo valor de q_α , que também pode ser consultado em (Demšar 2006).

3.2 Primeiro experimento - Métrica de dissimilaridade e algoritmo de agrupamento

No contexto de agrupamento de séries temporais, particularmente no agrupamento de curvas de carga, a principal combinação de métrica de dissimilaridade e algoritmo de agrupamento utilizada é a distância euclidiana com o *k-means* (Chicco 2012). O objetivo do experimento a ser realizado nesta seção é de verificar se existe alguma outra combinação que seja mais promissora que a combinação padrão. Para tal, utilizou-se 55 bases de dados do UCR *Time Series*, onde, pelo fato delas estarem rotuladas, o índice VI discutido na Seção 2.4.2 foi utilizado para comparar a partição obtida por determinada escolha de algoritmo e métrica de dissimilaridade com a partição real dos dados.

Todas as séries do repositório UCR *Time Series* já se encontram normalizadas pela normalização Z, e nenhuma estratégia de redução de dimensão dos dados, como a DFT ou o PAA, foi realizada. Além do índice VI, os tempos de processamento de cada estratégia também foram comparados para que fosse possível a comparação das estratégias em relação à este critério. Por tempo de processamento da combinação métrica de dissimilaridade e algoritmo de agrupamento, entende-se como o tempo resultante da soma do tempo de obtenção da matriz de dissimilaridade e do tempo da realização do agrupamento em si pelo algoritmo.

Dado a extensa possibilidade de combinações de algoritmos de agrupamento e métricas de dissimilaridade que podem ser obtidas a partir dos algoritmos e métricas discutidos no Capítulo 2, apenas algumas delas foram analisadas nesta seção. Dessa forma, as métricas DTW e euclidiana, bem como suas variações que levam em conta a complexidade (CID) e correlação temporal (CORT), foram analisadas, já que segundo estudos no contexto de classificação de séries temporais, elas se demonstraram superiores em relação às demais (Giusti e Batista 2013). Os experimentos são detalhados nas seções a seguir.

3.2.1 Algoritmos de agrupamento

Inicialmente, deseja-se verificar o efeito da escolha do algoritmo nos resultados do agrupamento. Assim, em um primeiro experimento, todas as bases serão agrupadas por diferentes algoritmos mas com a dissimilaridade fixa, e definida como a distância euclidiana, já que esta é uma das métricas de dissimilaridade mais comum. Foram escolhidos 4 algoritmos particionais, com diferentes inicializações, e dois algoritmos hierárquicos, com diferentes *linkagens*, e todas as combinações formadas podem ser vistas na Tabela 3.1.

Para as 55 bases de dados e 6 combinações (estratégias), o valor crítico da estatística de Friedman é de 2.24. O valor médio dos *rankings* forneceu uma estatística $F_F = 12.72$ rejeitando, portanto, H_0 . Dessa forma, pode-se afirmar que a escolha do algoritmo de agrupamento tem efeito significativo no desempenho individual de uma dada medida de dissimilaridade.

Finalmente, o teste *post-hoc* de Bonferroni-Dunn foi aplicado, centrado na combinação padrão, *k-means (random)* e distância Euclidiana. Os resultados deste teste podem ser vistos na Figura 3.1), e o traço presente nela, é o valor da diferença crítica do teste de Bonferroni-Dunn, conforme detalhado em 3.1, centrado no ranking médio da estratégia padrão (*k-means(random)* com distância Euclidiana). Os resultados mostram que o algoritmo hierárquico com linkagem *average* possui

	Id	Algoritmo	Dissimilaridade
0	KDR__eucl.	k-medoids(random)	eucl.
1	HC__eucl.	hierarchical-complete	eucl.
2	KDP__eucl.	k-medoids(PAM)	eucl.
3	KMP__eucl.	k-means(++)	eucl.
4	KMR__eucl.	k-means(random)	eucl.
5	HA__eucl.	hierarchical-average	eucl.

Tabela 3.1: Estratégias comparadas no experimento 1.

desempenho significativamente superior a estratégia padrão, ao passo que as demais estratégias não possuem uma performance significativamente distinta dela.

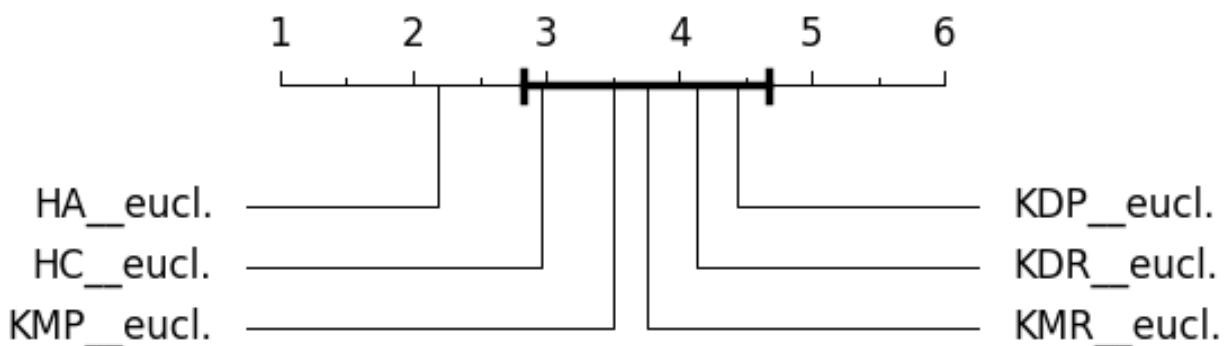


Figura 3.1: Resultado do teste de Bonferroni-Dunn após o ranqueamento dos valores de VI. A estratégia de controle se encontra centrada no ranking médio da estratégia *k-means(random)* com distância Euclidiana.

3.2.2 Métricas de dissimilaridade

O objetivo deste segundo experimento é de comparar o desempenho das diferentes métricas de dissimilaridade. Desde que o algoritmo exerce influência no desempenho individual da métrica, o procedimento usado neste experimento foi fixar o algoritmo de agrupamento e variar as métricas. O algoritmo fixado foi o agrupamento hierárquico com linkagem *average*, por este ter apresentado o melhor resultado no experimento anterior. A combinação *k-means* e distância Euclidiana foi também avaliada neste experimento, por se tratar da combinação padrão adotada na literatura. Todas as combinações deste experimento são mostradas na Tabela 3.2.

Para 55 bases de dados e 5 combinações testadas, o valor crítico da estatística de Friedman é de 2.41. A hipótese H_0 foi descartada, já que obteve-se $F_F = 14.04$. Em seguida, deu-se prosseguimento ao teste *post-hoc* de Bonferroni-Dunn, centrado na estratégia padrão (*k-means(random)*)

	Id	Algoritmo	Similaridade
0	HA__cort-eucl.	hierarchical-average	cort-eucl.
1	HA__DTW	hierarchical-average	DTW
2	HA__cid-eucl.	hierarchical-average	cid-eucl.
3	KMR__eucl.	k-means(random)	eucl.
4	HA__eucl.	hierarchical-average	eucl.

Tabela 3.2: Estratégias comparadas no experimento 2.

com distância Euclidiana). O resultado, ilustrado na Figura 3.2, mostra que a combinação padrão é inferior à todas as demais, e além disso, que o uso da DTW e da CID acarretaram em melhoras significativas em relação à distância Euclidiana e CORT-Euclidiana.

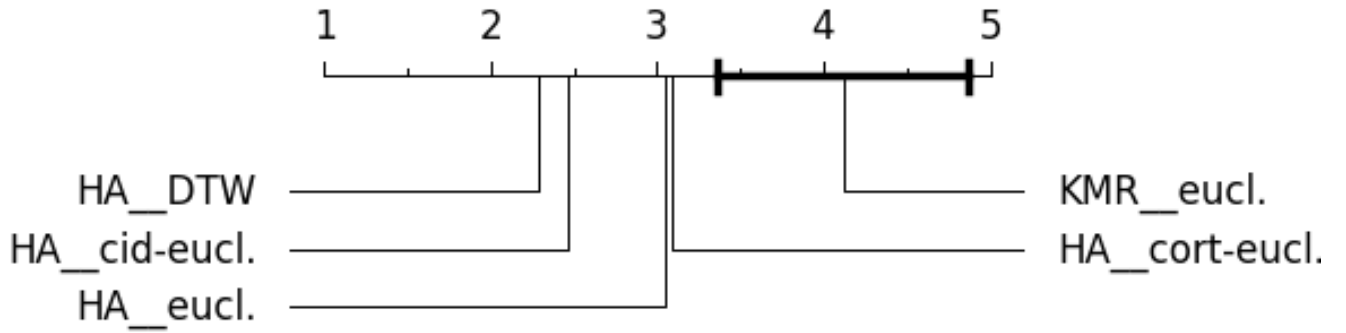


Figura 3.2: Resultado teste Bonferroni-Dunn após o ranqueamento dos valores de VI, centrado no ranking médio da estratégia padrão (*k-means(random)* com distância Euclidiana).

3.2.3 Comparação entre Métricas Promissoras

Os experimentos anteriores mostraram que a adição de métricas robustas a distorções nos dados, tais como DTW e CID pode melhorar significativamente a qualidade das partições obtidas. Com isso em mente, o objetivo deste experimento foi comparar as possíveis combinações envolvendo estas métricas promissoras nos quesitos de qualidade das partições (índice VI) e custo computacional (tempo de processamento). As combinações analisadas neste experimento se encontram na Tabela 3.3. Observe que, novamente, o algoritmo hierárquico com linkagem *average* foi fixado.

O teste de Friedman foi aplicado para testar a hipótese nula de que as estratégias são equivalentes. Para 55 bases de dados e 5 estratégias, o valor crítico é de 2.41. Considerando o quesito tempo de processamento, obteve-se $F_F = 543.03$ e, para o VI, $F_F = 3.48$. Logo, para ambos os casos, a hipótese nula de que os rankings médios são iguais foi descartada. Deu-se então prosseguimento ao teste *post-hoc* de Nemenyi, com o objetivo de verificar quais grupos de estratégias possuem

	Id	Algoritmo	Similaridade
0	HA__cort-eucl.	hierarchical-average	cort-eucl.
1	HA__DTW	hierarchical-average	DTW
2	HA__cid-eucl.	hierarchical-average	cid-ecul.
3	HA__cort-DTW	hierarchical-average	cort-DTW
4	HA__cid-DTW	hierarchical-average	cid-DTW

Tabela 3.3: Estratégias do experimento 3.

desempenhos equivalentes. O diagrama resultante do teste de Nemenyi para o índice VI pode ser visto na Figura 3.3. O diagrama correlato, para o tempo de processamento, é mostrado na Figura 3.4.

Pode-se concluir, com base na Figura 3.3, que todas as combinações envolvendo as métricas DTW e CID são significamente melhores que a combinação envolvendo CORT e Euclidiana (HA__cort-eucl.). A Figura 3.4, por sua vez, mostra que todas as combinações envolvendo DTW (HA-DTW, HA_CID-DTW, HA_cort-DTW) possuem maior custo computacional, sendo estatisticamente inferiores em relação as demais combinações. Uma análise conjunta, envolvendo os dois quesitos, apontam a combinação HA_CID-eucl. como um bom equilíbrio entre a qualidade das partições obtidas e o menor tempo de processamento.

Com relação ao índice VI (Figura 3.3), observa-se ainda que, embora o teste não tenha apontado diferença estatística, os resultados em termos dos *rankings* médios apontam que a correção CID (complexidade) produz efeito mais positivo do que a correção CORT (correlação temporal), independentemente da métrica convencional adotada: Euclidiana ou DTW. Além disso, todas as combinações envolvendo DTW se mostraram melhores que as outras em termos de *rankings* médios, o que sugere a importância de um critério que seja invariante a deslocamentos locais no agrupamento de séries temporais.

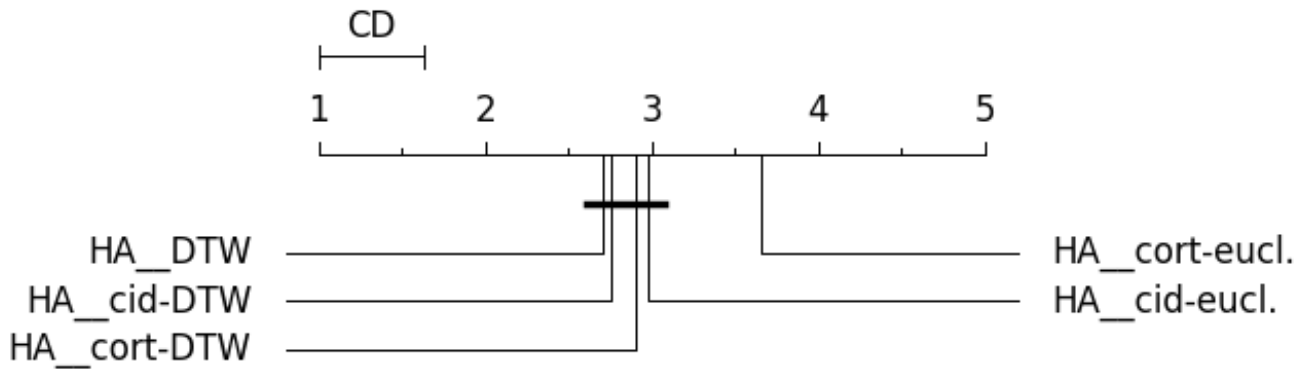


Figura 3.3: Resultado teste Nemenyi após o ranqueamento dos valores de VI.

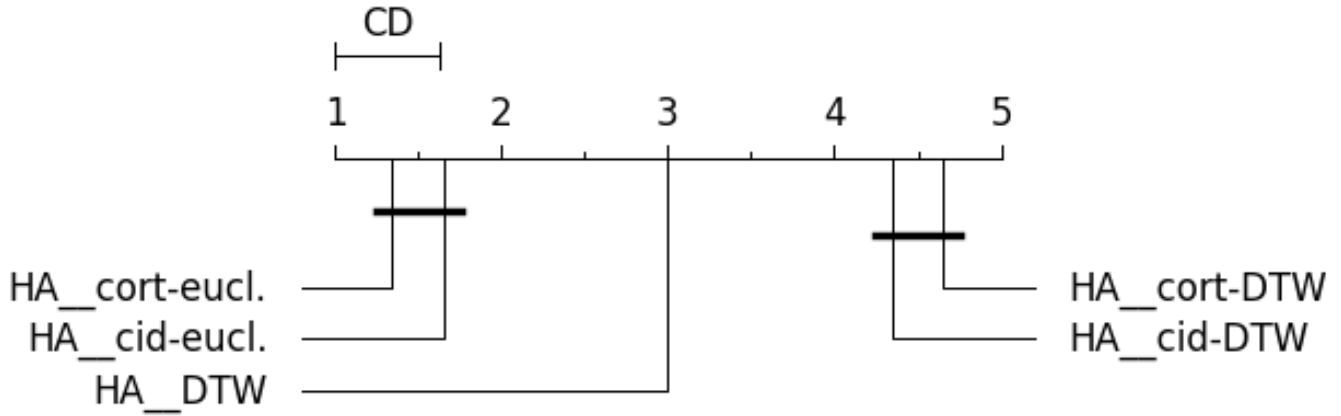


Figura 3.4: Resultado teste Nemenyi após o ranqueamento dos valores de tempo de processamento.

Tabela 3.4: Principais índices internos de validação disponíveis na literatura

Índice	Notação	Definição	Valor ótimo	Ref.
Calinski-Harabasz	CH	$\frac{\sum_i n_i d^2(c_i, c)/(k-1)}{\sum_i \sum_{x \in C_i} d^2(x, c_i)/(n-k)}$	Max	Caliński e Harabasz 1974
I index	I	$\left(\frac{1}{k} \cdot \frac{\sum_{x \in X} d(x, c)}{\sum_i \sum_{x \in C_i} d(x, c_i)} \cdot \max_{i,j} d(c_i, c_j) \right)^p$	Max	Maulik e Bandyopadhyay 2002
Dunn's index	Dunn	$\min_i \left(\min_j \left(\frac{\min_{x \in C_i, y \in C_j} d(x, y)}{\max_k (\max_{x, y \in C_k} d(x, y))} \right) \right)$	Max	J. C. Dunn 1974
Silhouette	silhouette	$\frac{1}{k} \cdot \sum_i \frac{1}{n_i} \sum_{x \in C_i} \frac{b(x) - a(x)}{\max(b(x), a(x))}$	Max	Peter Rousseeuw 1987b
Davies-Bouldin	DB	$\frac{1}{k} \sum_i \max_{j \neq i} \left(\left(\frac{1}{n_i} \sum_{x \in C_i} d(x, c_i) + \frac{1}{n_j} \sum_{x \in C_j} d(x, c_j) \right) / d(c_i, c_j) \right)$	Min	Davies e Bouldin 1979
Xie-Beni	XB	$\frac{\sum_i \sum_{x \in C_i} d^2(x, c_i)}{n \cdot \min_{i,j \neq i} d^2(c_i, c_j)}$	Min	Xie e Beni 1991
SD validity	SD	$Dis(k_{max})Scat(k) + Dis(k)$ $Scat(k) = \frac{1}{k} \frac{\sum_i \sigma(C_i) }{ \sigma(X) }$ $Dis(k) = \frac{\max_{i,j} d(c_i, c_j)}{\min_{i,j \neq i} d(c_i, c_j)} \sum_i (\sum_{j \neq i} d(c_i, c_j))^{-1}$	Min	Halkidi, Vazirgiannis e Batistakis 2000

3.3 Segundo experimento - Índice de validação interno

Neste segundo experimento, tentaremos responder às seguintes perguntas:

- Existe algum índice de validação interno que seja mais adequado para a definição do número de grupos um conjunto de dados de séries temporais?
- Uma vez definido o número de grupos, existe algum índice interno que seja mais indicado para avaliar as partições obtidas por diferentes estratégias de agrupamento?

Para responder à ambas perguntas os índices internos exibidos na Tabela [3.4](#) foram comparados nos experimentos desta seção. Além desses índices da tabela, outros três índices farão parte da análise. Eles derivam da combinação de todos os índices da tabela, denominados índices puros, e são definidos como a moda (ou voto majoritário), média e mediana do número de grupos indicado por cada um dos 7 índices.

Na Tabela 3.4 se encontram os índices internos a serem utilizados nesta seção, onde X : é todo o conjunto de dados; c : é o centróide (quando a métrica de dissimilaridade escolhida é a distância euclidiana) ou medóide de X ; C_i : i -ésimo grupo; n_i : número de instâncias no i -ésimo grupo; c_i : centróide ou medóide do i -ésimo grupo; $d(x, y)$: medida de dissimilaridade entre x e y ; n : número de elementos em X ; k : número de grupos em X , $\|v\| = (v^T \cdot v)^{\frac{1}{2}}$ e $\sigma(C_i)$: vetor de variância de todas as instâncias contidas em C_i ; p : valor inteiro, recomendado $p = 2$ (Maulik e Bandyopadhyay 2002), $b(i)$: a menor dissimilaridade média da instância i e cada instância contida em grupos diferentes do grupo de i ; $a(i)$ é a média de dissimilaridade da instância i para todas as outras instâncias contidas no mesmo grupo de i .

3.3.1 Índice para a definição do número de grupos

Em tarefas de agrupamento, quando não se sabe *a priori* o número de grupos nos quais os dados se encontram naturalmente divididos, são necessárias estratégias para se definir este número, já que ele é o principal parâmetro de entrada dos algoritmos particionais como o *k-means* e o *k-medoids*. Mesmo para algoritmos hierárquicos, esta é uma informação importante, pois a interpretação dos dendrogramas muitas vezes não é trivial, já que é difícil a definição do ponto de corte da árvore que define o particionamento dos dados.

Uma estratégia comum é realizar diversos agrupamentos para diferentes valores de k (número de grupos), dentro de determinado intervalo, no qual se espera que o valor ótimo k_{opt} , ou real, de grupos se encontra. Após a avaliação de cada partição resultante por um índice de validação interno, aquela que possuir um valor mais próximo do valor ótimo do índice, será a partição que apresenta o número ótimo ou mais indicado de grupos dentro do intervalo testado.

Dessa maneira, surge a pergunta: qual, ou quais, dos índices internos detalhados na literatura são mais interessante para a tarefa de se definir o número de agrupamentos de um conjunto de dados de séries temporais? Para responder à essa pergunta, iremos utilizar, novamente, o conjunto de dados UCR *Time series* (Y. Chen et al. 2015). Como cada conjunto de dados se encontra rotulado é sabido *a priori* o número de grupos que os dados se encontram divididos em cada conjunto de dados. Assim, dado o valor esperado de determinado conjunto de dados, denominado k_{opt} , foi formado um intervalo $k = k_{opt} - 6, k_{opt} - 5, \dots, k_{opt}, \dots, k_{opt} + 6$, onde caso o primeiro valor seja menor que 2, o intervalo é definido, necessariamente, a partir deste valor, já que em tarefas de agrupamento não faz sentido um valor de k menor do que 2.

Após a escolha de um algoritmo e uma métrica de dissimilaridade como, por exemplo, um algoritmo de agrupamento hierárquico com linkagem *average* e a métrica CID-euclidiana, para cada valor de k no intervalo criado, uma partição é obtida e avaliada com cada um dos índices da Tabela 3.4. A título de exemplo, tal procedimento, com o algoritmo e métrica de dissimilaridade acima mencionado, foi realizado no conjunto de dados *Herring* (vide Figura A.35), que possui $k_{opt} = 2$, e os resultados podem ser vistos na Tabela 3.5. Os índices DBI, XB e silhouette tiveram êxito na sugestão do valor de k_{opt} , ao passo que todos os demais falharam em tal tarefa. A partir da combinação da sugestão de cada um dos 7 índices testados podem-se derivar os seguintes índices: moda dos valores sugeridos, cujo valor é 2; média dos valores sugeridos, que possui valor 3.57 que é truncado para 4 já que este é o valor inteiro mais próximo; e finalmente mediana dos valores sugeridos para k_{opt} , cujo valor é 3. Logo, para o teste em questão, dentre os índices originados pela

combinação dos índices puros, somente a moda teve sucesso na indicação do valor de k_{opt} .

Tabela 3.5: Resultados dos experimentos iterativos para o conjunto de dados *Herring*, cujo número de grupos é igual a 2. Os valores em negrito representam os melhores valores obtidos em cada índice.

k	CH	DBI	Dunn	I	SD	XB	silhouette
2.0	2.025	0.486	0.369	0.031	0.517	0.478	0.409
3.0	20.876	0.492	0.353	15.024	0.414	0.489	0.259
4.0	21.448	0.879	0.262	20.746	0.569	0.614	0.206
5.0	16.315	0.826	0.267	14.124	0.500	0.598	0.183
6.0	14.646	1.184	0.330	12.780	0.806	0.982	0.195
7.0	12.879	1.073	0.334	10.826	0.759	0.961	0.186
8.0	10.987	1.208	0.378	8.506	0.908	1.092	0.184

Os valores em negrito na Tabela 3.5 representam os melhores valores obtidos em cada índice, e quando este valor coincide com o valor de k_{opt} , então um acerto é computado para o índice naquela base. Em seguida, tal procedimento foi repetido para 70 bases de dados, com o objetivo de se construir uma tabela binária, representando os acertos e erros de cada índice em cada base. Ao final da tabela, o número de acertos de cada índice foi somado e dividido pelo número de bases, para se ter uma ideia da porcentagem de vezes que cada índice obteve sucesso. Devido às suas grandes dimensões (70x13), somente parte da tabela binária de tal experimento pode ser vista na Tabela 3.6, já que diversas linhas (datasets) e algumas colunas (índices) da tabela não se encontram presentes.

Tabela 3.6: Resultados dos experimentos iterativos para cada conjunto de dados, onde o asterisco corresponde à um sucesso na sugestão do índice para a base de dados. Foi utilizado o algoritmo hierárquico com linkagem *average* e métrica de dissimilaridade cid-euclidiana.

	Conjunto de dados	CH	DBI	Dunn	I	SD	XB
0	50words						
1	Adiac						
2	ArrowHead	*			*		
26	Herring		*				*
69	yoga	*					
70	Total	0.429	0.214	0.143	0.257	0.171	0.343
		CH	DBI	Dunn	I	SD	XB

O total da tabela binária obtida pela combinação do algoritmo hierárquico com linkagem *average* e métrica de dissimilaridade cid-euclidiana foi armazenado, assim como os totais obtidos por diversas outras combinações. Com todos os totais, obtidos por diferentes combinações de algoritmos de agrupamento e métricas de dissimilaridade, pôde-se formar uma terceira tabela, cuja

as linhas são compostas pelas diferentes combinações de algoritmo e métrica de dissimilaridade, enquanto as colunas são os índices, puros e combinados, utilizados na tarefa iterativa de se apontar o número de agrupamentos. Os valores de tal tabela contêm cada um a porcentagem de vezes que um índice aponta corretamente o número de agrupamentos de um conjunto de dados de séries temporais para determinada combinação de métrica de dissimilaridade e algoritmo de agrupamento.

A tabela resultante possui 78 linhas, ou combinações de algoritmos e métricas de dissimilaridade, e 10 colunas, ou índices testados, o que faz com que o valor crítico para o teste não-paramétrico de Friedman seja $F_{crit} = 1.79$, para uma significância de 5%. Após a obtenção do ranking médio de cada índice e a realização do teste de Friedman, obteve-se o valor $F_F = 258.42$, o que descarta a hipótese nula H_0 de que os índices possuem performance equivalente frente às diferentes estratégias testadas. Dessa maneira, deu-se prosseguimento ao teste *pos-hoc* de Nemenyi, para verificar se existe algum índice, ou grupo de índices, que possua performance superior aos demais. O resultado do teste de Nemenyi pode ser visto na Figura 3.5, onde se destaca o o grupo de índices formado pelo Calinski-Harabasz e a moda da sugestão de todos os índices, já que este grupo possui um ranking médio menor do que os demais grupos.

Logo, conclui-se que o índice Calinski-Harabasz é o mais indicado para a tarefa de se definir o número de grupos em que um conjunto de dados de séries temporais se encontra naturalmente dividido, já que para a obtenção do índice combinado moda, que possui performance equivalente, faz-se necessária a obtenção de todos os demais índices puros além do citado índice, o que, do ponto de vista do custo computacional, não é interessante.

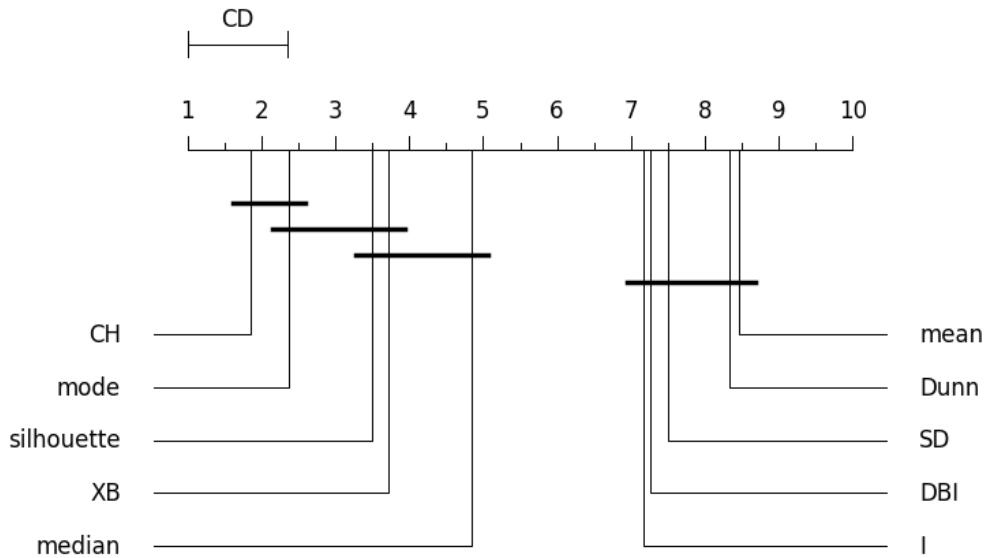


Figura 3.5: Resultado do teste de Nemenyi para o teste do melhor índice interno para apontar o número de grupos.

3.3.2 Índice interno para avaliação de partições com o número de grupos definido

Uma vez definido o número de grupos, surge outra pergunta: qual dos índices internos da Tabela 3.4 é o mais indicado para avaliar a qualidade de diferentes partições? Ou, em outras palavras, qual índice interno é o melhor indicador que houve uma melhora, ou piora, na partição resultante após uma alteração nos parâmetros ou estratégias de agrupamento com exceção do número de grupos? Nos experimentos será utilizada a distância de kendall-tau que é uma medida de dissimilaridade entre dois rankings e definida como a contagem do número de discordâncias entre os dois rankings em análise (Kendall 1938). Vale ressaltar que a medida de Kendall-tau, que varia no intervalo $[-1, 1]$, é uma métrica no espaço de rankings, de forma que valores maiores implicam em rankings mais próximos. Quando dois rankings são idênticos, o valor da métrica Kendall-Tau entre eles é 1.

Com o objetivo de elucidar a proposta deste experimento, foi realizado o seguinte experimento:

- escolheu-se arbitrariamente uma base de dados do UCR *Time Series* (*synthetic_control* que possui 6 grupos, vide Figura A.80),
- escolheu-se arbitrariamente uma métrica de dissimilaridade (distância Euclidiana),
- realizou-se seis agrupamentos do conjunto de dados com os seguintes algoritmos: *hierarchical-average* (HA), *hierarchical-complete* (HC), *hierarchical-single* (HS), *k-means++* (KP), *k-means* (KR), *k-medoids*(PAM)(KP),
- avaliou-se cada uma das seis partições obtidas com os seguintes índices: VI, silhouette, Calinski-Harabasz.

Os resultados numéricos podem ser vistos na Tabela 3.7, bem como os valores ranqueados na Tabela 3.8. Em seguida, calculou-se a distância Kendall-Tau entre o ranking do índice silhouette e o ranking do índice VI, onde obteve-se o valor -0.20, bem como a distância Kendall-Tau entre o ranking do índice Calinski-Harabasz e o ranking do índice VI, onde obteve-se o valor 0.33. Logo, como valores maiores de Kendall-Tau implicam em rankings mais coerentes entre si, pode-se afirmar que o índice Calinski-Harabasz, ao se variar o algoritmo de agrupamento, na base de dados e métrica de dissimilaridade escolhidas, tem comportamento, no que diz respeito a indicar uma piora ou melhora da partição, mais similar ao índice VI do que o índice silhouette.

A partir deste breve experimento, pode-se afirmar que, caso não se tivesse disponível a rotulação dos dados, mas somente o número de grupos, o índice Calinski-Harabasz é mais indicado que o índice silhouette para indicar uma melhora, ou piora, nas partições obtidas com a variação do algoritmo de agrupamento para a base de dados e métrica de dissimilaridade escolhida.

O experimento realizado restringiu-se a uma base de dados, à dois índices internos e à avaliação da variação do algoritmo de agrupamento. Para realizar um estudo mais amplo, foram realizados agrupamentos com um pequeno grupo de 55 combinações de algoritmo de agrupamento e métrica de dissimilaridade e de 71 conjuntos de dados de séries temporais, novamente conjuntos de dados do UCR *Time Series* (Y. Chen et al. 2015). Cada partição obtida foi avaliada por cada um dos índices internos da Tabela 3.4 e pelo índice externo VI exposto na seção 2.4.2. Como o VI é uma métrica no espaço de partições, uma melhora ou piora no seu valor indica uma respectiva melhora

Tabela 3.7: Rankings dos valores dos índices Para cada uma das partições obtidas variando-se o algoritmo de agrupamento.

	HC	HA	HS	KM	KR	KP
VI	1.41	1.39	1.84	0.84	0.73	2.57
Calinski-Harabasz	42.04	49.11	1.02	55.73	55.87	26.27
silhouette	0.23	0.25	-0.01	0.21	0.21	-0.01

Tabela 3.8: Rankings dos valores dos índices da tabela 3.7. Note que valores maiores para o Calinski-Harabasz e o silhouette implicam em rankings melhores, ao passo que para o VI em rankings piores.

	HC	HA	HS	KM	KR	KP
VI	5	4	2	1	3	6
Calinski-Harabasz	5	4	2	1	6	3
silhouette	2	1	5	4	6	3

ou piora na aproximação da partição real em função de uma estratégia de agrupamento melhor ou pior para determinada base de dados. Dessa maneira, espera-se que os índices internos que acompanham a melhora ou piora do VI em função das variações da estratégia de agrupamento sejam os mais indicados para avaliar partições com o número de grupos definido.

Para se comparar a aderência, ou coerência, de cada índice interno ao VI, todos os índices para cada base foram ranqueados em função das estratégias escolhidas. Em seguida, para cada base e para cada índice foi calculada a distância de Kendall-Tau entre os rankings obtidos pelo VI e os rankings obtidos para cada índice. Os resultados podem ser vistos na Tabela 3.9, que contém os valores $1 - \tau$, onde τ é o valor Kendall-Tau, de forma que valores menores indicam uma maior coerência do índice interno com o índice VI.

Tabela 3.9: Tabela com os valores de Kendall-Tau dos índices em relação ao VI.

	dataset	CH	DBI	silhouette	SD	XB	Dunn
0	50words	0.91	1.13	0.91	1.24	1.08	0.93
1	Adiac	0.98	0.68	1.03	0.92	0.74	1.08
2	ArrowHead	1.04	0.95	0.78	1.14	1.02	1.01
3	BeetleFly	0.87	1.11	1.11	1.11	1.29	0.97
4	CBF	1.11	1.15	0.74	0.84	1.05	0.92
5	Car	1.07	1.02	0.95	0.96	1.07	0.84
6	ChlorineConcentration	1.08	1.08	0.66	1.09	0.98	0.81
7	CinC_ECG_torso	0.85	0.94	0.79	1.17	1.00	0.78
8	Coffee	0.82	1.08	0.81	1.00	0.99	1.09
9	Computers	1.02	1.06	0.97	1.15	1.08	0.79
10	DiatomSizeReduction	0.68	1.11	0.58	1.21	1.19	0.60

11	DistalPhalanxOutlineAgeGroup	1.08	0.87	0.66	0.91	0.93	0.82
12	DistalPhalanxOutlineCorrect	1.34	1.44	0.53	1.01	1.40	0.69
13	DistalPhalanxTW	1.01	1.17	0.92	0.90	1.11	0.92
14	ECG200	1.14	1.19	0.63	1.02	1.15	0.77
15	ECG5000	1.17	1.07	0.69	1.02	1.16	0.96
16	ECGFiveDays	1.16	1.05	1.24	0.85	0.77	1.26
17	Earthquakes	1.03	0.79	0.86	1.19	0.84	1.03
18	FISH	1.07	1.07	0.76	0.95	1.23	0.90
19	FaceFour	1.01	1.10	0.93	1.12	1.07	1.00
20	FacesUCR	1.13	1.01	0.82	0.94	1.08	0.99
21	FordA	1.07	0.98	1.03	1.17	0.93	0.83
22	FordB	0.72	0.78	1.12	1.04	0.88	0.82
23	Ham	1.49	1.48	0.66	1.10	1.41	0.61
24	HandOutlines	1.32	1.27	0.61	1.03	1.27	0.61
25	Haptics	1.22	1.22	0.95	0.95	1.31	0.81
26	Herring	1.31	1.38	0.76	0.97	1.35	0.73
27	InlineSkate	1.13	1.30	0.65	0.98	1.23	1.01
28	ItalyPowerDemand	0.95	1.20	0.84	1.15	1.12	0.82
29	LargeKitchenAppliances	1.04	0.88	0.92	0.84	0.93	0.72
30	Lighting2	0.89	1.11	0.88	0.81	1.16	0.93
31	Lighting7	0.96	1.04	0.87	1.02	1.21	0.83
32	MALLAT	0.81	0.94	0.92	0.99	1.01	1.10
33	Meat	0.85	1.30	0.68	1.03	1.25	0.69
34	MedicalImages	1.06	1.11	0.80	1.01	1.03	0.72
35	MiddlePhalanxOutlineAgeGroup	0.74	1.15	0.85	0.87	1.15	1.07
36	MiddlePhalanxOutlineCorrect	1.12	1.10	0.96	0.90	0.93	0.41
37	MiddlePhalanxTW	1.17	1.38	0.83	0.84	1.19	0.64
38	MoteStrain	0.73	0.75	1.10	1.09	0.75	0.79
39	NonInvasiveFatalECG_Thorax1	0.93	1.02	0.96	1.05	0.89	0.98
40	NonInvasiveFatalECG_Thorax2	1.07	0.92	0.85	0.89	0.76	1.03
41	OSULeaf	1.03	1.15	0.82	0.77	1.04	0.99
42	PhalangesOutlinesCorrect	1.37	1.40	0.86	1.30	1.30	0.74
43	Phoneme	0.94	1.04	0.84	1.20	1.03	1.06
44	Plane	0.84	1.02	0.74	1.14	1.06	0.85
45	ProximalPhalanxOutlineAgeGroup	0.98	1.01	0.59	1.18	0.96	0.75
46	ProximalPhalanxOutlineCorrect	1.05	0.72	1.01	1.12	0.95	0.87
47	ProximalPhalanxTW	0.96	1.07	0.79	1.17	1.00	0.71
48	RefrigerationDevices	1.11	1.18	0.91	1.05	1.18	0.87
49	ScreenType	1.14	0.96	0.88	1.15	0.97	1.02
50	ShapesAll	0.64	1.10	1.00	0.99	0.86	0.88
51	SmallKitchenAppliances	1.15	1.03	0.72	0.93	1.06	0.92
52	SonyAIBORobotSurface	1.16	1.01	0.50	0.93	0.88	0.74
53	SonyAIBORobotSurfaceII	0.81	0.94	0.95	1.07	1.12	0.87

54	StarLightCurves	0.72	0.87	0.75	1.12	1.16	1.11
55	Strawberry	1.08	1.08	0.48	0.83	1.03	0.90
56	SwedishLeaf	0.72	1.02	0.96	0.96	0.99	1.01
57	Symbols	0.66	1.08	0.65	0.94	1.24	0.82
58	ToeSegmentation1	0.82	1.14	1.31	1.23	1.13	0.87
59	ToeSegmentation2	0.88	0.98	1.12	0.98	1.04	1.13
60	Trace	0.83	0.81	0.86	0.92	0.93	1.16
61	TwoLeadECG	1.06	1.13	0.78	0.95	1.15	0.61
62	Two_Patterns	0.97	1.14	0.90	1.12	0.96	0.89
63	UWaveGestureLibraryAll	0.82	1.03	0.78	1.09	0.97	0.88
64	Wine	1.39	1.36	0.67	1.15	1.30	0.56
65	WordsSynonyms	0.93	1.14	0.82	0.90	1.25	0.97
66	Worms	0.86	1.12	0.74	0.96	1.08	1.05
67	WormsTwoClass	0.73	0.84	0.87	1.14	0.89	1.09
68	synthetic_control	1.15	1.01	0.58	0.88	1.09	0.85
69	wafer	1.08	1.16	0.80	0.94	1.02	0.94
70	yoga	0.89	0.99	0.66	1.10	1.01	0.85
Rankings médios		3.58	4.42	2.16	3.91	4.23	2.67
		CH	DBI	silhouette	SD	XB	Dunn

Para 70 bases e 6 índices internos, o valor crítico da estatística de Friedman é de $F_{crit} = 2.12$, e como para os rankings médios destacados na Tabela 3.9 foi obtido o valor $F_F = 18.11$, então a hipótese nula H_0 de que todos os índices possuem aderências equivalentes ao índice VI foi descartada. A partir desse resultado, deu-se prosseguimento ao teste de Nemenyi para verificar qual, ou quais, índice interno possui aderência significativamente maior que os demais índices. Os resultados podem ser vistos na Figura 3.6, que mostram que os índices silhouette e Dunn, possuem melhor aderência ao VI que os demais índices internos.

Logo, em uma tarefa de agrupamento real, no qual não se possui a rotulação dos dados e já se decidiu, ou já se sabe *a priori*, o número de grupos que os dados se encontram divididos, os índices silhouette e Dunn são os mais indicados para avaliar as novas partições obtidas por diferentes estratégias de agrupamento.

3.4 Conclusões do Capítulo

Os experimentos deste Capítulo buscaram definir uma estratégia padrão de agrupamento de séries temporais, e as estratégias testadas foram confrontadas com a estratégia padrão, que consiste na realização do agrupamento pelo algoritmo *k-means* e distância Euclidiana. Dessa maneira, foram realizados dois experimentos, sendo o primeiro para estudar as influências da escolha do algoritmo e métrica de dissimilaridade na qualidade da partição de dados de séries temporais, enquanto o segundo experimento teve como objetivo estudar os índices de validação interna na avaliação das partições resultantes do agrupamento de séries temporais.

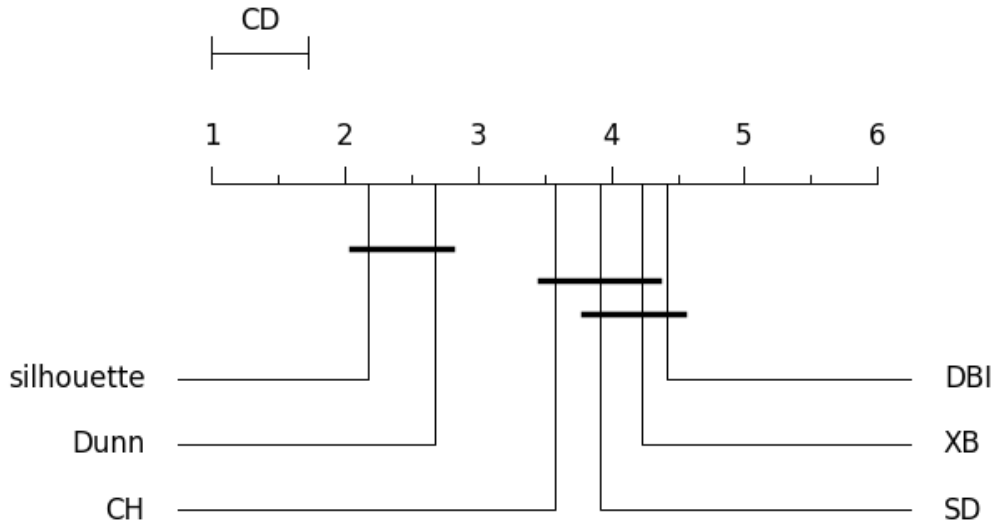


Figura 3.6: Resultado do teste de Nemenyi para o teste de coerência dos índices internos com o VI.

Assim, no primeiro experimento, foram realizados dois testes:

- no primeiro foram testados diversos algoritmos de agrupamento fixando-se a distância Euclidiana como métrica de dissimilaridade e verificou-se a superioridade do algoritmo hierárquico com *linkagem average*,
- já no segundo teste, fixou-se o algoritmo de agrupamento, como o algoritmo hierárquico com *linkagem average*, e variou-se as dissimilaridades. onde verificou-se a superioridade da performance das dissimilaridades DTW e CID-Euclidiana em relação às demais dissimilaridades testadas.

No segundo experimento, também foram realizados dois testes:

- no primeiro, os índices de validação interna foram testados para indicar aquele que é o mais adequado para a tarefa de apontar o número de grupos que as instâncias do conjunto de dados se encontram naturalmente divididas, e, após a análise dos resultados, chegou-se a conclusão que o índice Calinski-Harabasz é o mais indicado para tal,
- no segundo teste, procurou-se encontrar um índice da validação interna que seja mais coerente com o VI e, dessa maneira, um índice que seja um melhor indicador da melhora ou piora da partição dos dados de séries temporais ao se alterar a estratégia de agrupamento, chegou-se a conclusão que os índices Dunn e silhouette são os índices mais coerentes com o VI.

Dessa maneira, partindo-se do pressuposto que não se possui, *a priori*, nenhuma informação útil para a realização do agrupamento, como, por exemplo, o número de grupos que as instâncias

se encontram naturalmente divididas, os experimentos realizados em bases de dados rotuladas sugerem a seguinte abordagem a ser realizada no agrupamento de séries temporais:

- Realizar algum tipo de normalização das curvas. Sugere-se a normalização Z ou normalização max.
- Escolher alguma medida de dissimilaridade. Sugere-se a CID-DTW caso o conjunto de dados não seja muito grande, ou os recursos computacionais escassos. Em qualquer uma dessas duas hipóteses, sugere-se a utilização da CID-Euclidiana.
- Para se definir o número de grupos k , realizar o agrupamento do conjunto de dados já normalizados pelo algoritmo hierárquico com *linkagem average* e medida de dissimilaridade escolhida. Em seguida, obter diferentes partições a partir da poda do dendrograma obtido por valores iterativos de k , em um intervalo no qual se espera que se encontre o valor de k .
- Em seguida, para cada uma das partições obtidas, realizar a avaliação de cada uma delas pelo índice Calinski-Harabasz, e aquela que possuir o melhor valor indicará o valor de k .
- Caso se deseje realizar um refinamento da partição obtida até o momento, uma vez definido o número de grupos k , pode-se realizar outras partições com diferentes escolhas de combinações de algoritmos e métricas de dissimilaridade e avaliá-las pelo índice silhouette. A partição que acarretar no melhor valor do índice é a partição final da tarefa de agrupamento.

A estratégia descrita foi empregada no estudo de caso do Capítulo 4. Vale destacar que esta estratégia não garante resultados ótimos, ela é apenas uma sugestão inicial quando não se dispõe de informações relevantes para o agrupamento acerca dos dados. Nos próprios dados sintéticos do UCR *Time Series*, verificou-se que para alguns conjuntos de dados em específico, a metodologia proposta teve performance inferior à performance da estratégia padrão (*k-means* com distância Euclidiana). No entanto, na média, levando-se em conta todos os conjuntos, os testes estatísticos demonstraram que ela é significativamente superior às demais estratégias, como a estratégia padrão, por exemplo.

Capítulo 4

Estudo de caso

Neste capítulo, é descrita a metodologia para o agrupamento das curvas de carga obtidas pela medição periódica do consumo de energia elétrica das residências australianas. O objetivo deste agrupamento é verificar as influências que existem na potência consumida em cada carga em função dos dias de semana e da estação do ano.

Em geral, tarefas de mineração de dados, independente do domínio do problema, buscam extrair conhecimento inovador, para auxiliar as tomadas de decisões a nível gerencial (CARVALHO [s.d.](#)). Assim, o agrupamento dos perfis de consumo de carga buscam gerar perfis característicos de consumo em diferentes estações do ano, e em diferentes dias da semana. Ademais, espera-se realizar a segmentação dos consumidores para a proposição de uma eventual taxa diferenciada de cobrança para cada grupo de consumidores encontrado. Segundo (Flath et al. [2012](#)), tal faixa diferenciada criada pelas concessionárias de energia a partir da segmentação de seus consumidores residenciais tem o poder de influenciar o consumo destes, podendo, assim, diminuir os picos de consumo.

4.1 Metodologia

O conjunto de dados australianos consiste das medições de 13735 cargas, realizadas de 6/10/2011 a 4/3/2014, no entanto, na avaliação foram utilizadas somente as medições das primeiras 5000 cargas durante um ano compreendido de 21/12/2012 a 21/12/2013. A redução do número de cargas a serem analisadas foi necessária por restrições de memória RAM, já que esta se encontrava limitada em 8GB. Já a restrição e escolha das datas de início e fim do período em análise, se deu pela necessidade de se avaliar as curvas diárias dos medidores em função da variação das estações do ano, assim, o intervalo de análise das medições foi reduzido para um ano, e teve como dia de início e fim o dia do solstício de verão do hemisfério sul, de forma que existe um número igual de medições para cada uma das quatro estações do ano.

A potência consumida foi medida pelos medidores australianos a cada 30 minutos, totalizando 48 medições diárias. Existem dados incompletos (medições faltantes) ou inconsistentes (medições repetidas em um mesmo instante), o que fez com que fosse necessária uma etapa de filtragem dos dados. Nesta etapa, se um medidor apresentasse dados de medição incompletos ou inconsistentes,

ele seria descartado. Isso fez com que o número de medidores utilizados no agrupamento fosse reduzido drasticamente para 427 cargas.

A tarefa de agrupamento foi realizada em seis conjuntos de dados distintos, mas cada conjunto contendo os mesmo medidores. A formação dos conjuntos de dados se deu a partir da combinação das estações do ano; inverno, verão e transição, sendo esta última compreendida pelas estações outono e inverno, com os dois tipos de dia de semana; final de semana e dia de semana. Cada um desses subconjuntos é composto pela curva diária mais representativa de cada medidor nos dias correspondentes ao grupo, portanto, cada subconjunto contém, ao final, 427 curvas. Tal divisão é a mesma realizada em (Flath et al. 2012), onde curvas de carga de consumidores residenciais, comerciais, industriais e rurais da Alemanha foram agrupados.

A formação destes 6 subconjuntos pode ser visualizada na Figura 4.1. A primeira caixa (Perfis diários anuais) contém 155.855 curvas (427×365), e em seguida, cada uma das curvas é direcionada para o segundo nível. As curvas obtidas por medições em dias do final de semana (sábado e domingo) são direcionadas para a caixa à esquerda (Final de semana), enquanto as curvas obtidas por medições em dias de semana (segunda-feira, terça-feira, quarta-feira, quinta-feira e sexta-feira) são direcionadas para a caixa à direita (Dia de semana). Finalmente, em cada uma das caixas, cada curva é novamente direcionada para um novo subconjunto em função da estação do ano que ela foi obtida. Cada um dos subconjuntos do terceiro nível da figura contém todas as medições, no período definido pelo subconjunto, para cada um dos 427 medidores, e, dessa maneira, para a realização do agrupamento, faz-se necessária a obtenção de uma curva representativa para cada um dos 427 medidores em cada um dos 6 subconjuntos.

A curva diária mais representativa de cada medidor foi escolhida como o centróide de todas as medições naquele subconjunto, pois esta escolha, em detrimento do medóide, fez com que as curvas características de cada medidor se tornassem mais suaves, o que facilitou a tarefa de agrupamento e de interpretação dos resultados. Após a obtenção do centróide de cada medidor, em cada um dos seis subconjuntos, obteve-se seis matrizes de 427×48 , já que em um dia foram realizadas 48 medições por cada um dos 427 medidores em cada um dos seis subconjunto de dados. Uma vez que todos os centróides possuíam valores positivos e a normalização max é a mais utilizada no agrupamento de curvas de carga (Chicco 2012), então neste estudo de caso, cada centróide foi normalizado pela normalização max descrita na Seção 2.1.1.

4.2 Agrupamento

Cada um dos seis conjuntos de dados foi agrupado segundo a estratégia definida no Capítulo 3, ou seja, algoritmo hierárquico com *linkagem average* e métrica de dissimilaridade, e avaliação do índice Calinski-Harabasz para a indicação do número de grupos, que foi variado de 2 a 20. Notou-se que os melhores valores desses índices eram obtidos em partições altamente desbalanceadas, ou seja, partições que possuíam um dos grupos contendo cerca de 95% dos medidores. Uma partição com tamanho desbalanceamento não é, na maior parte das vezes, adequada, pois não se pode extrair nenhuma informação útil dela, já que não se pode realizar a segmentação dos consumidores de energia.

A partir dos resultados insatisfatórios obtidos a partir da estratégia indicada no final do 3,

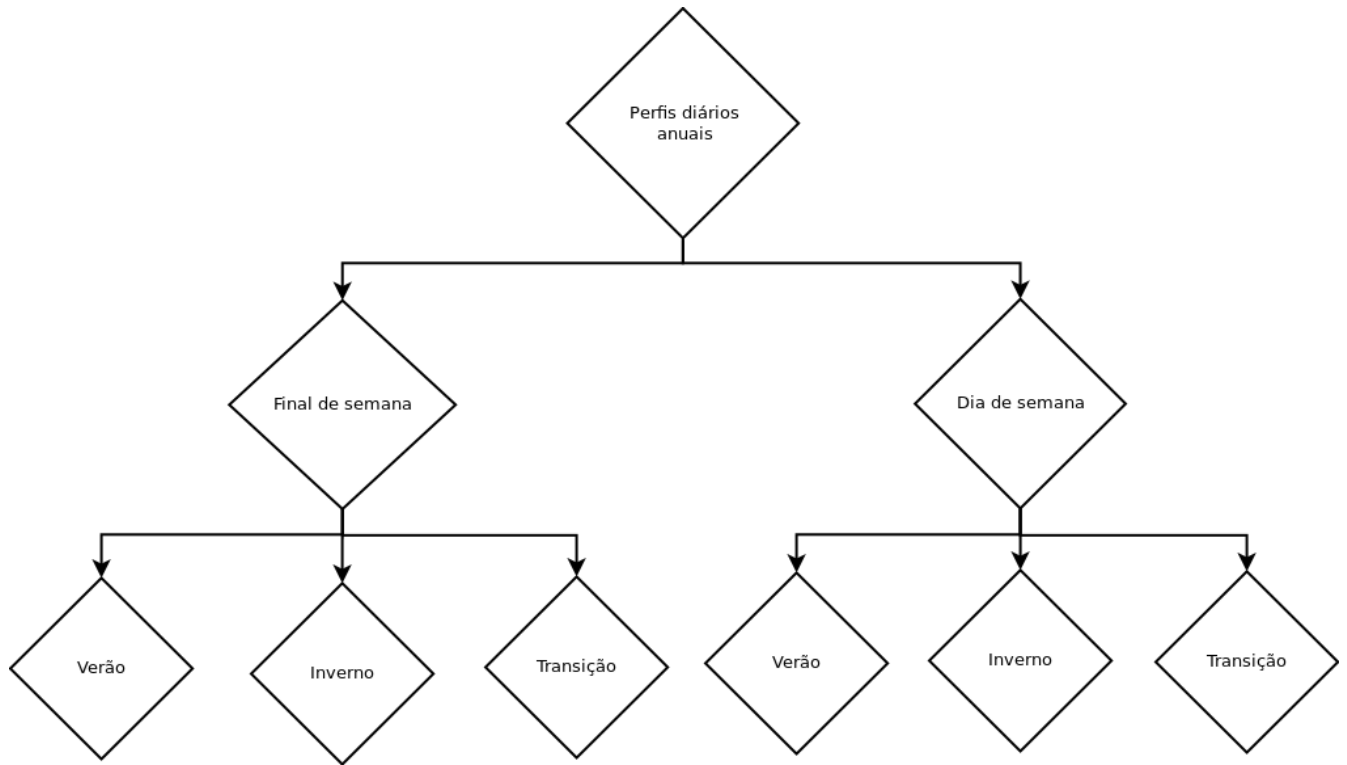


Figura 4.1: Formação dos 6 subconjuntos de dados a partir dos perfis diários.

foram realizados agrupamentos com os seguintes algoritmos: k-means, k-medoids, hierarchical-average . Para cada um dos algoritmos, com exceção do k-means que foi realizado somente para a distância euclidiana, foi realizado um agrupamento com as seguintes dissimilaridades: euclidiana, cid-euclidiana, cort-euclidiana, chebyshev, cid-chebyshev, cort-chebyshev, DTW, cid-DTW, cort-DTW, EDR, cid-EDR, cort-EDR. Para cada combinação possível de algoritmo e dissimilaridade, o valor k do número de grupos foi, novamente, variado de 2 a 20. A partir das combinações de algoritmo, métrica de dissimilaridade e k , formou-se 684 partições diferentes para cada um dos seis conjuntos de dados.

Cada partição foi avaliada pelos índices internos silhouette e Calinski-Harabasz, onde, novamente, todas as soluções apontadas pelos índices se encontravam altamente desbalanceadas. Em uma tentativa de se obter partições mais balanceadas, descartou-se as instâncias que compunham os grupos minoritários, por pensar que estas poderiam ser *outliers*, e o agrupamento foi realizado novamente somente no grupo majoritário, que continha quase a totalidade das instâncias. Os resultados acarretaram, novamente, em partições altamente desbalanceadas. Este procedimento foi repetido reiteradamente, até que se encontra-se um número muito pequeno de cargas, cerca de 10% do número inicial.

Em uma terceira tentativa, para realizar o agrupamento dos subconjuntos, utilizou-se o DBSCAN, devido, principalmente, à sua robustez frente a conjuntos de dados ruidosos. Foi utilizada a métrica CID-Euclidiana, e os critérios de *MinPts* e *Eps*, foram escolhidos pelo gráfico *k-dist*, conforme exemplo da seção 2.3.3. Novamente, o DBSCAN apontou uma partição dos dados

composta por um único grupo majoritário e um número reduzido de instâncias que ele classificou como *outliers*. Assim, como na abordagem com o algoritmo hierárquico, as instâncias que foram consideradas *outliers* foram descartadas e o DBSCAN foi novamente empregado nas instâncias que compunham o grupo majoritário. Uma única partição majoritária foi novamente obtida, com um número reduzido de *outliers*. Assim, chegou-se a conclusão que o grupo de dados australiano é composto, efetivamente, por instâncias compactas, de forma que o agrupamento das mesmas deverá ser realizado por outra técnica não abordada até o momento.

No trabalho realizado em (Flath et al. 2012), que foi inspiração para este estudo de caso, não foram obtidas partições tão desbalanceadas com um procedimento similar ao empregado na base de dados australiana, salvo à utilização do índice DBI ao invés do índice Calinski-Harabasz para a indicação do número de grupos. Dentre as 215 cargas alemãs utilizadas no referido estudo, existiam cargas industriais, residenciais, rurais e comerciais, diferentemente da base de dados australiana que contém somente dados de medidores inteligentes instalados em cargas residenciais. Dessa maneira, dada a própria heterogeneidade dos tipos de cargas a serem agrupadas, no estudo alemão, foi possível a obtenção de grupos mais balanceados com as técnicas empregadas até o momento.

4.2.1 Estratégia alternativa de agrupamento

Após os resultados insatisfatórios obtidos nas partições, deu-se prosseguimento à uma nova estratégia para a escolha da partição de cada um dos seis conjuntos de dados construídos. Calculou-se as porcentagens de cargas contidas em cada grupo, de cada partição obtida. A partir dos valores de porcentagem de cada grupo, obteve-se os valores mínimos, máximos, médios, e o desvio padrão das porcentagens de cada partição. Partições cujo valor máximo excedeu 50%, ou cujo valor mínimo foi inferior à 5% foram descartadas. O outro critério a ser utilizado na escolha da melhor partição de cada grupo foi o índice silhouette. Este índice foi escolhido por ter um limite superior, ou seja, valor máximo igual a 1, o que faz com que seja possível a sua utilização na comparação de duas partições obtidas por diferentes dissimilaridades, por exemplo. Ao se utilizar o índice Calinski-Harabasz, valores muito discrepantes são obtidos para as diferentes partições, e por não possuir um valor máximo, os valores de cada partição não podem ser normalizados para que a sua comparação seja realizada efetivamente.

Observou-se que as partições que possuíam valores maiores do índice silhouette eram as mais desbalanceadas, ou seja, com um alto valor de desvio padrão, ao passo que as partições com um baixo desvio padrão, ou balanceadas, possuíam um valor de silhouette aproximadamente igual a 0, o que indica partições cujo agrupamento foi realizado de forma próxima à aleatória. Assim, dada à característica conflitante observada entre esses dois objetivos; qualidade da partição, associada ao valor de silhouette, e balanceamento da mesma, associada ao desvio padrão das porcentagens de cargas em cada grupo, ficou caracterizado um problema de otimização multiobjetivo. Logo, fez-se necessário encontrar as soluções, ou partições, não dominadas e da escolha de uma solução, ou partição, dentre aquelas soluções que formam a fronteira Pareto-ótima.

Não houve a necessidade de se utilizar um método de otimização multiobjetivo, pois todas as partições já estavam calculadas. O que se fez foi encontrar as soluções não dominadas que formam a fronteira Pareto-ótima. Uma vez encontrada as soluções que formam a fronteira de Pareto-ótima,

Tabela 4.1: Tabela que apresenta as informações das partições escolhidas para cada um dos seis sub-conjuntos gerados.

Norm.	Dia de semana	Estação	Alg.	diss.	k	max %	méd. %	min %	silhouette	std %
max		Transição	k-means	eucl.	9	0.17	0.11	0.07	0.12	0.03
max		Inverno	k-means	eucl.	10	0.14	0.10	0.060	0.11	0.02
max		Verão	k-means	eucl.	8	0.18	0.12	0.06	0.11	0.03
max	✓	Transição	k-medoids	cid-eucl.	3	0.36	0.33	0.31	0.07	0.02
max	✓	Inverno	k-means	eucl.	9	0.15	0.11	0.06	0.13	0.02
max	✓	Verão	k-means	eucl.	3	0.36	0.33	0.30	0.17	0.03

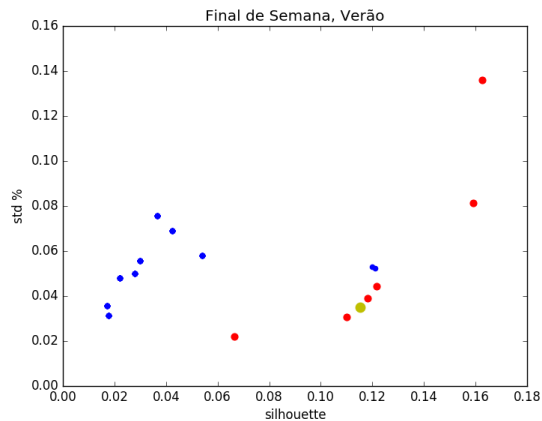
calculou-se a distância da solução ótima $[1, 0]$, ou seja valor de silhouette igual a 1 e desvio padrão igual a 0, a cada uma das soluções da fronteira Pareto-ótima. Aquela que possuía a menor distância foi escolhida como partição adequada para aquele conjunto de dados. Os resultados, ou seja, as partições escolhidas, podem ser encontrados na tabela 4.1.

O espaço de soluções bem como as curvas características de cada curva em cada um dos seis conjuntos de dados podem ser visualizados nas figuras 4.2a, 4.3a, 4.4a, 4.5a, 4.6a e 4.7a. Nestas figuras, cada ponto representa uma partição, e pode-se notar que existem cerca de 20 pontos em cada figura, ao invés das 684 partições inicialmente obtidas. Isso ocorreu devido à restrição imposta ao se descartar partições cujo grupo majoritário contém mais de 50% das instâncias ou o grupo minoritário contém menos de 5% do total de instâncias. Esta redução drástica no número de partições, reforça o caráter compacto do grupo, já que para diferentes escolhas de algoritmo e métrica de dissimilaridade, a maioria das partições obtidas não atendeu à restrição imposta. Os pontos em vermelho nas figuras compõem a fronteira Pareto-ótima, e a solução, ou partição, indicada é representada pelo ponto amarelo.

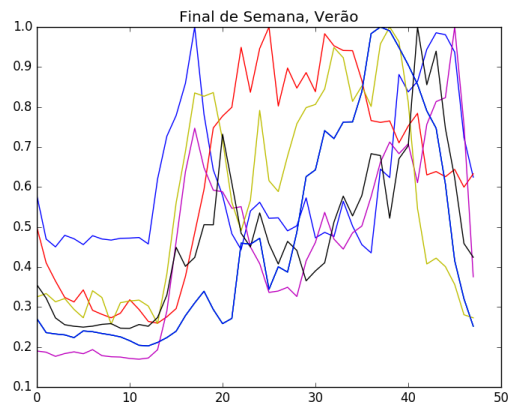
4.3 Análise dos resultados

A abordagem tradicional do agrupamento, com a combinação mais promissora apontada no Capítulo 3, resultou em partições altamente desbalanceadas, o que não é adequado do ponto de vista prático para a descoberta de perfis e segmentação de consumidores residenciais. Foi então proposta uma metodologia que consiga encontrar partições dos dados que consiga estabelecer um compromisso entre o balanceamento da partição, e a qualidade da mesma, ou seja, partições com alta densidade intergrupo e sem a presença de um grupo que contém quase a totalidade das instâncias.

Diferentemente dos resultados obtidos no capítulo 3, onde se tinha disponível a rotulação dos dados, e os melhores resultados foram obtidos com os algoritmos hierárquicos e métrica de dissimilaridade CID-DTW, neste estudo de caso, o conjunto de dados não possui rotulação, as partições obtidas por meio do algoritmo k-means aliado à métrica euclidiana foram os que alcançaram os

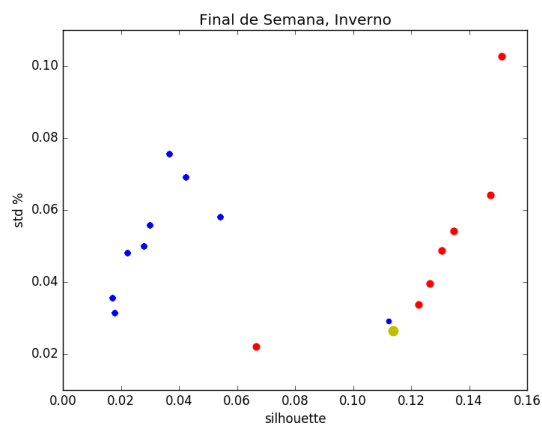


(a) Soluções no espaço de soluções.

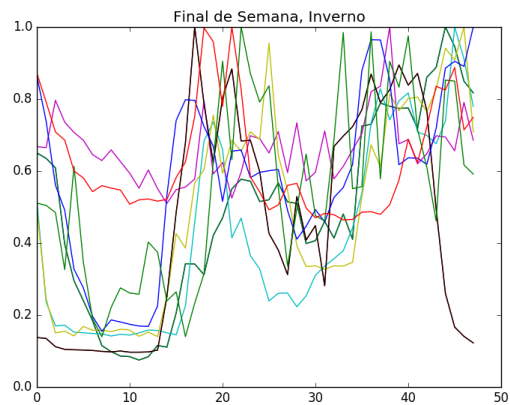


(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.2: Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e no verão. Os pontos em azul, na Figura 4.2a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo.



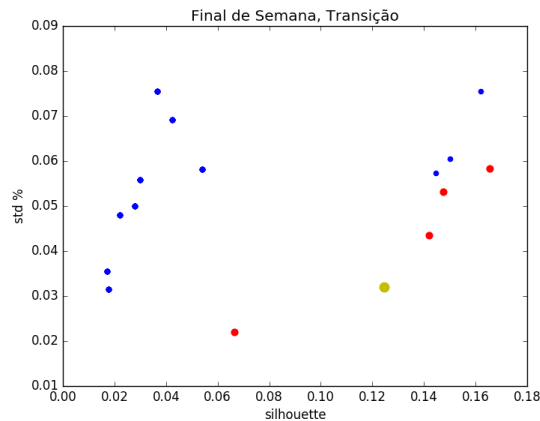
(a) Soluções no espaço de soluções.



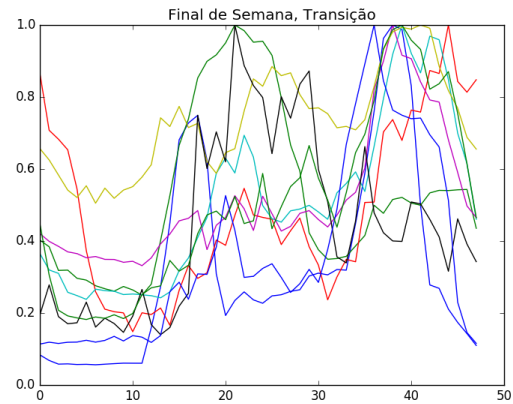
(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.3: Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e no inverno. Os pontos em azul, na Figura 4.3a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo.

resultados mais satisfatórios. As demais combinações testadas de algoritmo e métrica de dissimilaridade acarretaram em partições altamente desbalanceadas. Das 684 partições obtidas ao se variar o algoritmo de agrupamento, a métrica de dissimilaridade e o número de grupos, após a seleção de

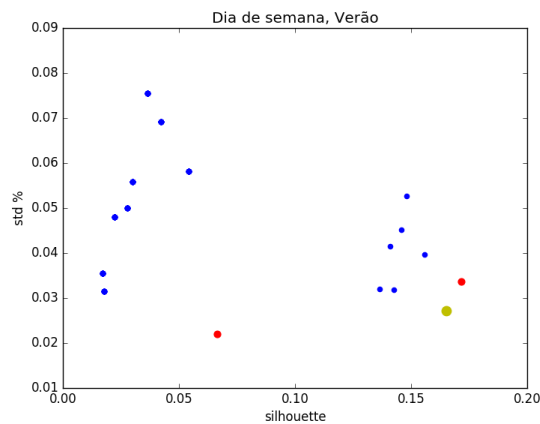


(a) Soluções no espaço de soluções.

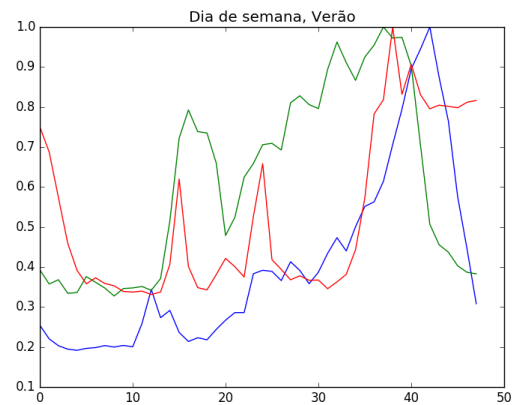


(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.4: Resultado final para os curvas diárias das cargas medidas nos dias do final de semana e nas estações de transição. Os pontos em azul, na Figura 4.4a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo



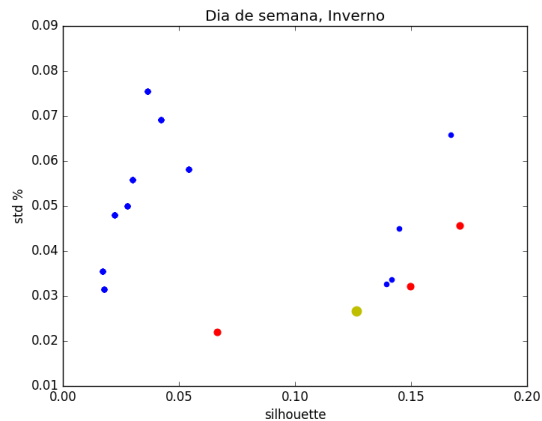
(a) Soluções no espaço de soluções.



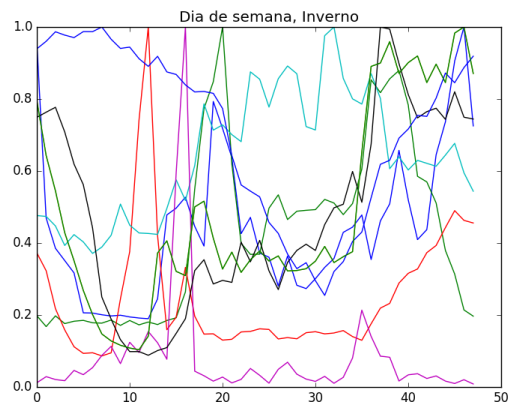
(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.5: Resultado final para os curvas diárias das cargas medidas nos dias de semana e no verão. Os pontos em azul, na Figura 4.5a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo

partições cujo maior grupo representava menos de 50% dos medidores e o menor mais do que 5%, o número de soluções disponíveis foi reduzido drasticamente. Isso pode ser percebido nos gráficos do espaço de soluções 4.2a, 4.3a, 4.4a, 4.5a, 4.6a e 4.7a, já que, em cada um deles, existem

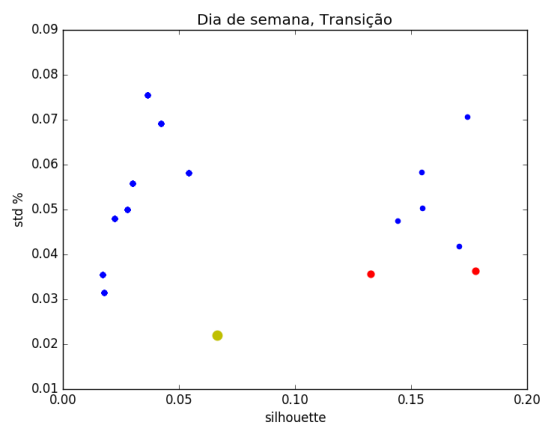


(a) Soluções no espaço de soluções.

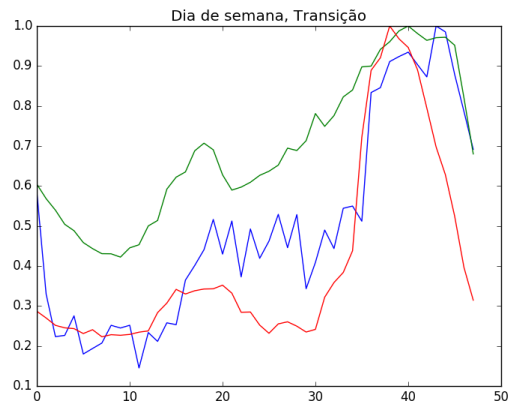


(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.6: Resultado final para os curvas diárias das cargas medidas nos dias de semana e no inverno. Os pontos em azul, na Figura 4.6a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo



(a) Soluções no espaço de soluções.



(b) Medóides de cada um dos grupos da partição escolhida.

Figura 4.7: Resultado final para os curvas diárias das cargas medidas nos dias de semana e na transição. Os pontos em azul, na Figura 4.7a, representam partições dominadas pelas demais, enquanto os pontos vermelhos formam a fronteira de Pareto-ótima. O ponto em amarelo é a partição escolhida por ser a mais próxima do ponto ótimo

aproximadamente 20 soluções possíveis das 684 iniciais.

Assim, pode-se inferir que o algoritmo *k-means* é o mais adequado, no contexto de agrupamento de curvas de carga, quando se deseja obter partições balanceadas. Outro resultado que parece

interessante é o fato do número de grupos encontrado ser maior nos conjuntos de dados obtidos a partir do final de semana. Nestes, o valor de k foi em torno de 9, ao passo que nos conjuntos de dados contendo medições realizadas nos dias de semana, o número k foi igual a 3. Isso leva a crer que durante o final de semana a diversidade de perfis de consumo de carga aumenta em relação aos dias de semana. Além disso, este estudo indica que este fator, o tipo de dia de semana, tem maior impacto no número de grupos k do que a variação das estações do ano. Isto pode ser dito no contexto australiano, pois, caso este estudo fizesse uso de dados medidos em outros países, como no Brasil, por exemplo, os resultados poderiam ter sido bem diferentes.

Em relação aos perfis de consumo em cada subconjunto, pode-se perceber, pelas Figuras 4.2a, 4.3a, 4.4a, 4.5a, 4.6a e 4.7a, que:

- na maioria dos perfis característicos, as curvas têm um crescimento abrupto aproximadamente na décima segunda medição (às 6 horas da manhã),
- na maioria dos perfis característicos, as curvas têm um decrescimento abrupto aproximadamente na quadragésima medição (às 20 horas da noite),
- na maioria dos perfis característicos dos finais de semana, as curvas apresentam um vale centrado na vigésima quarta medição (às 12 horas),
- os perfis de dia de semana no verão (vide Figura 4.5b) e nos meses de transição (vide Figura 4.7b) são bastante similares, o que reforça a suposição de que nas residências australianas a variação da estação do ano não influencia no comportamento de consumo de energia elétrica dos australianos, mas o tipo de dia de semana sim, além de ambos apresentarem uma tendência de crescimento durante o dia,
- no inverno e dia de semana (vide Figura 4.6b), existem alguns perfis que divergem significativamente dos demais, apresentando picos significativos de consumo,

todas essas observações são, de certa forma esperadas para curvas de carga de residências, o que mostra que os resultados são coerentes. No entanto, a interpretação das curvas aliado à outros dados dos consumidores que compõem cada grupo dos subconjuntos em análise, tais como a renda, localização, número de moradores, dentre outros, poderiam gerar informações mais ricas e úteis para as concessionárias de energia.

Há de se destacar que a estratégia implementada neste estudo de caso foi realizada em uma base de dados reduzida, com 427 cargas, e dessa maneira foi possível realizar as 684 partições diferentes de cada um dos seis conjuntos de dados com um esforço computacional moderado. O agrupamento dos dados gerados por medidores inteligentes instalados por uma concessionária de energia elétrica como a CEMIG, que possui milhões de consumidores residenciais (*Relações com investidores - Formulário de Referência* 2016), demandará um custo computacional relevante. Assim, em um cenário real, após a massificação da instalação de medidores inteligentes no sistema elétrico de potência brasileiro, é mister que seja definida uma única abordagem, em relação à escolha do algoritmo e métrica de dissimilaridade.

Assim, pelos resultados obtidos, sugere-se que o agrupamento de curvas de cargas residenciais seja realizado, após a normalização max, por meio do algoritmo *k-means* e métrica de dissimilaridade euclidiana. Esse resultado vai de encontro com a literatura (Chicco 2012). Sugere-se

ainda que caso seja feita a mesma divisão dos dados realizados neste estudo de caso, ou seja, em tipos de dia da semana e em estações do ano, que o parâmetro k do algoritmo *k-means* seja aproximadamente igual à 3 para os conjuntos de dados contendo medições nos dias de semana, e aproximadamente 10 para os dias dos finais de semana.

A indicação algoritmo k-means para a realização do agrupamento de curvas de carga, por gerar partições que possuem um bom compromisso entre balanceamento dos grupos e qualidade da partição, possui ainda a vantagem deste ser um algoritmo altamente escalável, ou em outras palavras, altamente paralelizável (Zhao, Ma e He 2009; Stoffel e Belkoniene 1999; Bahmani et al. 2012), tanto localmente quanto distribuídamente, o que, por sua vez, também é adequado para uma massa de dados de milhões de consumidores.

Capítulo 5

Conclusões e propostas de trabalhos futuros

Este trabalho teve como objetivo a realização de um estudo das estratégias de agrupamento de séries temporais para a posterior aplicação das técnicas estudadas no agrupamento de dados de curvas de consumo de cargas residenciais. Dessa maneira, inicialmente, as possíveis escolhas ao se realizar o agrupamento de séries temporais foram analisadas.

As técnicas de pré-processamento foram discutidas na qual foram expostas algumas técnicas de redução de dimensionalidade como a PAA ou TDF, e se verificou que algum tipo de normalização das curvas deve ser realizada para que a comparação entre as curvas faça sentido. Em seguida, foram expostas as medidas de dissimilaridade entre as curvas, no qual obtiveram destaque algumas medidas como a DTW e EDR, além da correção das dissimilaridades pela CID (Batista, Wang e E. J. Keogh 2011). Logo após, foram apresentados os principais algoritmos encontrados na literatura como os algoritmos *k-means* e hierárquicos. Finalmente, os índices de validação de agrupamentos foram discutidos, dando-se certo destaque para o índice de validação externo de Variação da Informação (VI), pelo fato deste ser uma métrica no espaço de partições.

A partir da revisão bibliográfica realizada no Capítulo 2, no Capítulo 3 realizou-se um estudo empírico de agrupamento de bases de dados rotuladas, onde os resultados foram comparados com o resultado esperado pelo índice VI. Nos testes foram realizados agrupamentos, em cada base de dados, variando-se os principais algoritmos e medidas de dissimilaridade mencionados na literatura. Testes estatísticos de significância apontaram que os algoritmos hierárquicos, principalmente ao se utilizar a *linkagem average*, possuem performance melhor que os algoritmos particionais.

Realizando-se os agrupamentos com o algoritmo hierárquico e *linkagem average* e diferentes métricas de dissimilaridade, verificou-se que as dissimilaridades DTW (Berndt e Clifford 1994) possuem desempenho superior. No entanto, ao se aplicar a CID à dissimilaridades como a euclidiana, estas apresentaram desempenho estatisticamente equivalente ao desempenho da DTW quando também corrigida pela CID. Este resultado é interessante já que existe grande diferença de custos computacionais para se obter a DTW e distância euclidiana entre duas séries temporais, e que este custo para a última é significativamente menor.

Em seguida, foram realizados testes para analisar a performance dos índices de validação interno. Em um primeiro experimento, os índices internos foram testados para indicar o número de

grupos existentes em uma partição e, neste experimento, após as análises de significância, chegou-se a conclusão que o índice Calinski-Harabasz (Caliński e Harabasz 1974) é o mais adequado para a realização de tal tarefa no contexto de séries temporais. Uma vez definido o número de grupos, surge a pergunta de qual índice de validação interno é o mais adequado para avaliar as partições obtidas por diferentes abordagens no que diz respeito às escolhas da técnica de pré-processamento, algoritmo de agrupamento e métrica de dissimilaridade. Para responder à esta pergunta, novos testes foram realizados nas bases de dados rotuladas, e os resultados mostraram que os índices silhouette (Peter Rousseeuw 1987b) e Dunn (J. C. Dunn 1974) possuem performance significativamente superior aos outros quatro índices testados.

Dessa maneira, partindo-se do pressuposto que não se possui, *a priori*, nenhuma informação útil para a realização do agrupamento, como, por exemplo, o número de grupos que as instâncias se encontram naturalmente divididas, os experimentos realizados em bases de dados rotuladas sugerem a seguinte abordagem a ser realizada no agrupamento de séries temporais:

- Realizar algum tipo de normalização das curvas. Sugere-se a normalização Z ou normalização max.
- Escolher alguma medida de dissimilaridade. Sugere-se a CID-DTW caso o conjunto de dados não seja muito grande, ou os recursos computacionais escassos. Em qualquer uma dessas duas hipóteses, sugere-se a utilização da CID-Euclidiana.
- Para se definir o número de grupos k , realizar o agrupamento do conjunto de dados já normalizados pelo algoritmo hierárquico com *linkagem average* e medida de dissimilaridade escolhida. Em seguida, obter diferentes partições a partir da poda do dendrograma obtido por valores iterativos de k , em um intervalo no qual se espera que se encontre o valor de k .
- Em seguida, para cada uma das partições obtidas, realizar a avaliação de cada uma delas pelo índice Calinski-Harabasz, e aquela que possuir o melhor valor indicará o valor de k .
- Caso se deseje realizar um refinamento da partição obtida até o momento, uma vez definido o número de grupos k , pode-se realizar outras partições com diferentes escolhas de combinações de algoritmos e métricas de dissimilaridade e avaliá-las pelo índice silhouette. A partição que acarretar no melhor valor do índice é a partição final da tarefa de agrupamento.

Após a definição dos passos descritos anteriormente, obtidos por meio de testes em bases de dados rotuladas, estes foram aplicados em uma base de dados não rotulada no Capítulo 4. O estudo de caso consistiu no agrupamento dos dados de consumo de cargas residenciais, gerados por medidores inteligentes instalados em residências australianas. As curvas diárias de consumo de cada residência foram divididas em seis subconjuntos, formados cada um pela média das curvas diárias em cada uma das seis possíveis combinações entre as estações do ano; inverno, verão e transição (outono e primavera), e os tipos de dias da semana; dias de final de semana e dias de semana.

Os resultados obtidos, em todos os subconjuntos, por meio dos passos descritos anteriormente, não foram satisfatórios, pois partições altamente desbalanceadas foram obtidas. Por partições altamente desbalanceadas entende-se partições dos dados nas quais um único grupo contém quase

a totalidade dos dados, ao passo que os demais grupos, denominados grupos minoritários, contêm poucas instâncias. Partições com essas características são indesejadas, pois, na maioria das vezes, não é possível identificar padrões relevantes no agrupamento dos dados. Mesmo após a eliminação das cargas contidas nos grupos minoritários, por estas terem sido, inicialmente, consideradas *outliers*, a realização de novos agrupamentos acarretou, novamente, em partições altamente desbalanceadas.

Assim, decidiu-se utilizar uma técnica de avaliação de partição, que pelo menos no estudo realizado nesta dissertação, não foi visto em nenhuma das fontes bibliográficas utilizadas e mencionadas neste trabalho. Na abordagem proposta, cada uma das partições obtidas foram avaliadas por dois critérios: pela qualidade da partição, indicada pelo índice interno silhouette, e pelo balanceamento da mesma, indicado pelo desvio padrão da porcentagem de instâncias nos grupos da partição em análise. Nas partições obtidas por meio de diversas combinações de algoritmo de agrupamento e métrica de dissimilaridade, verificou-se que estes dois critérios são conflitantes, ou seja, partições que apresentavam os melhores valores de silhouette possuíam os piores valores de desvio padrão, ao passo que as partições com os melhores valores de desvio padrão possuíam os piores valores de silhouette.

Dado o caráter conflitante entre as duas características almejadas para a partição final, ficou caracterizado um problema de otimização multi-objetivo, e assim fez-se necessário encontrar as soluções, ou partições não dominadas. A partição final escolhida, foi aquela que pertence à fronteira Pareto-ótima que se encontra mais perto do ponto ótimo, ou seja do ponto $[1, 0]$, no qual a partição possui silhouette igual a 1 e desvio padrão igual a 0. As partições encontradas para cada um dos seis subgrupos formados apresentaram alguns padrões interessantes:

- Todos os subgrupos encontrados utilizaram como algoritmo de agrupamento o *k-means*, apesar de outros também terem sido testados, do que se pode inferir que o *k-means* gera partições que apresentam um bom compromisso entre a qualidade da partição e o balanceamento do número de instâncias nos grupos.
- Os subgrupos que continham dados de curva de carga obtidas nos finais de semana foram particionados em um número de grupos significativamente maior do que os subgrupos contendo curvas obtidas nos dias de semana, o que indica que existe uma maior heterogeneidade de perfis de consumo residenciais nos finais de semana em comparação com os dias de semana.

O fato da estratégia definida a partir dos experimentos realizados em bases de dados rotuladas no Capítulo 3 não ter gerado um resultado satisfatório ao ser aplicada em uma base de dado não rotulada no Capítulo 4 não a desmerece, já que ela deve ser vista como uma sugestão inicial para problemas de agrupamento de séries temporais, e não como uma solução definitiva. Vale destacar que algumas decisões da tarefa de agrupamento de séries temporais, como a escolha da métrica de dissimilaridade, dependem, muitas vezes, das características intrínsecas do problema em questão. No estudo realizado com as bases rotuladas, foram utilizadas séries temporais de diversas aplicações e áreas do conhecimento como sinais biomédicos, problemas de visão computacional, detecção de movimento, sinais de equipamentos elétricos, entre outros. Assim, reforça-se a sua sugestão como uma primeira alternativa para a realização do agrupamento de séries temporais, e caso resultados insatisfatórios sejam obtidos com ela, outras abordagens podem ser adotadas.

No contexto de curvas de cargas obtidas por medidores inteligentes, o estudo de caso realizado no Capítulo 4 indica que o algoritmo *k-means*, aliado à métrica de dissimilaridade euclidiana, fornece resultados satisfatórios. Tais escolhas estão em consonância com a literatura (Chicco 2012) e são interessantes para a realização de agrupamento de grandes massas de dados, como as que serão geradas após a massificação da instalação de medidores inteligentes no sistema elétrico de potência brasileiro, pois a distância euclidiana tem custo computacional significativamente menor que suas principais concorrentes (DTW e EDR), além do algoritmo *k-means* ser altamente escalável (Zhao, Ma e He 2009; Stoffel e Belkoniene 1999; Bahmani et al. 2012).

Assim, pode-se considerar que este contribui na indicação de metodologias (escolhas), no que diz respeito a pré-processamento, algoritmo de agrupamento e métrica de dissimilaridade, dentre as várias existentes na literatura, a serem utilizadas no agrupamento de curvas de carga obtidas por medidores inteligentes.

5.1 Propostas de trabalhos futuros

A dissertação de mestrado apresentada é somente uma fração do estudo de agrupamento de séries temporais e do seu emprego em agrupamentos de curvas de cargas, assim, sugere-se o aprofundamento nos seguintes temas:

- Avaliar, no contexto de agrupamento, a performance da medida de dissimilaridade TWED (*Time Warp Edit Distance*) proposta em (Marteau 2008), que em experimentos realizados em problemas de classificação, expostos em (Serrà e Arcos 2014), apresentou performance superior à distância euclidiana, DTW, EDR e suas respectivas correções pela CID.
- Estudar a variação no cálculo da DTW descrita em (Petitjean, Ketterlin e Gançarski 2011) que faz com que seja possível a utilização dela no algoritmo *k-means*.
- Verificar os métodos descritos em (Morse e Patel 2007) que reduzem os custos computacionais envolvidos no cálculo da DTW e EDR.
- Estudar as técnicas de agrupamento consensuais, que combinam diversas partições para a obtenção de uma partição final, no contexto de séries temporais (Strehl e Ghosh 2003).
- Investigar a robustez da proposta de agrupamento multi-objetiva realizada nesta dissertação para o agrupamento de conjuntos de dados desbalanceados.

Referências

- [16a] *Relações com investidores - Formulário de Referência*. <http://cemig.infoinvest.com.br/ptb/13695/Set>. de 2016.
- [16b] *Smart-Grid SmartCity Customer Trial data*. http://ckan.biocenit.cat/ko_KR/dataset/energy-consumption-demo. Acesso: Ago. 2016. 2016.
- [AFS93] Rakesh Agrawal, Christos Faloutsos e Arun N. Swami. “Efficient Similarity Search In Sequence Databases”. Em: *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*. FODO '93. London, UK, UK: Springer-Verlag, 1993, pp. 69–84. ISBN: 3-540-57301-1. URL: <http://dl.acm.org/citation.cfm?id=645415.652239>.
- [Agê10] Agência Nacional de Energia Elétrica. *Estrutura tarifária para o serviço de distribuição de energia elétrica*. [http://www2.aneel.gov.br/arquivos/PDF/SumárioExecutivo\(2\).pdf](http://www2.aneel.gov.br/arquivos/PDF/SumárioExecutivo(2).pdf). Dez. de 2010.
- [Agê11] Agência Nacional de Energia Elétrica. *Tarifa Social de Energia Elétrica - TSEE*. <http://www.aneel.gov.br/tarifa-social-baixa-renda>. Dez. de 2011.
- [Ank+99] Mihael Ankerst et al. “OPTICS: Ordering Points to Identify the Clustering Structure”. Em: *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*. SIGMOD '99. Philadelphia, Pennsylvania, USA: ACM, 1999, pp. 49–60. ISBN: 1-58113-084-8. DOI: [10.1145/304182.304187](https://doi.org/10.1145/304182.304187). URL: <http://doi.acm.org/10.1145/304182.304187>.
- [AV07] David Arthur e Sergei Vassilvitskii. “K-means++: The Advantages of Careful Seeding”. Em: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '07. New Orleans, Louisiana: Society for Industrial e Applied Mathematics, 2007, pp. 1027–1035. ISBN: 978-0-898716-24-5. URL: <http://dl.acm.org/citation.cfm?id=1283383.1283494>.
- [Bah+12] Bahman Bahmani et al. “Scalable k-means++”. Em: *Proceedings of the VLDB Endowment* 5.7 (2012), pp. 622–633.
- [BC94] Donald J. Berndt e James Clifford. “Using Dynamic Time Warping to Find Patterns in Time Series.” Em: *KDD Workshop*. Ed. por Usama M. Fayyad e Ramasamy Uthurusamy. AAAI Press, 1994, pp. 359–370. ISBN: 0-929280-73-3. URL: <http://dblp.uni-trier.de/db/conf/kdd/kdd94.html#BerndtC94>.

- [BWK11] Gustavo E. A. P. A. Batista, Xiaoyue Wang e Eamonn J. Keogh. “A Complexity-Invariant Distance Measure for Time Series.” Em: *SDM*. SIAM / Omnipress, 2011, pp. 699–710. ISBN: 978-1-61197-281-8. URL: <http://dblp.uni-trier.de/db/conf/sdm/sdm2011.html#BatistaWK11>.
- [CAR] A. CARVALHO. *INTELIGENCIA ARTIFICIAL: UMA ABORDAGEM DE APRENDIZADO DE MAQUINA*. LTC. ISBN: 9788521618805. URL: <https://books.google.com.br/books?id=4DwelAEACAAJ>.
- [Cen12] Centro de Gestão e Estudos Estratégicos. *Redes Elétricas Inteligentes: contexto nacional*. Rel. téc. 2012.
- [CH74] T. Caliński e J. Harabasz. “A dendrite method for cluster analysis”. Em: *Communications in Statistics-Simulation and Computation* 3.1 (1974), pp. 1–27.
- [Che+15] Yanping Chen et al. *The UCR Time Series Classification Archive*. www.cs.ucr.edu/~eamonn/time_series_data/. Jul. de 2015.
- [Chi12] Gianfranco Chicco. “Overview and performance assessment of the clustering methods for electrical load pattern grouping”. Em: *Energy* 42.1 (2012), pp. 68–80. URL: <http://EconPapers.repec.org/RePEc:eee:energy:v:42:y:2012:i:1:p:68-80>.
- [CN07] Ahlame Douzal Chouakria e Panduranga Naidu Nagabhushan. “Adaptive dissimilarity index for measuring time series proximity.” Em: *Adv. Data Analysis and Classification* 1.1 (2007), pp. 5–21. URL: <http://dblp.uni-trier.de/db/journals/adac/adac1.html#ChouakriaN07>.
- [CÖO05] Lei Chen, M. Tamer Özsu e Vincent Oria. “Robust and Fast Similarity Search for Moving Object Trajectories”. Em: *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*. SIGMOD '05. Baltimore, Maryland: ACM, 2005, pp. 491–502. ISBN: 1-59593-060-4. DOI: [10.1145/1066157.1066213](https://doi.org/10.1145/1066157.1066213). URL: <http://doi.acm.org/10.1145/1066157.1066213>.
- [DB79] David L. Davies e Donald W. Bouldin. “A Cluster Separation Measure”. Em: *IEEE Trans. Pattern Anal. Mach. Intell.* 1.2 (fev. de 1979), pp. 224–227. ISSN: 0162-8828. DOI: [10.1109/TPAMI.1979.4766909](https://doi.org/10.1109/TPAMI.1979.4766909). URL: <http://dx.doi.org/10.1109/TPAMI.1979.4766909>.
- [Dem+13] Janez Demšar et al. “Orange: Data Mining Toolbox in Python”. Em: *Journal of Machine Learning Research* 14 (2013), pp. 2349–2353. URL: <http://jmlr.org/papers/v14/demsar13a.html>.
- [Dem06] Janez Demšar. “Statistical Comparisons of Classifiers over Multiple Data Sets”. Em: *J. Mach. Learn. Res.* 7 (dez. de 2006), pp. 1–30.
- [DHS00] Richard O. Duda, Peter E. Hart e David G. Stork. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000. ISBN: 0471056693.

- [Dun61] Olive Jean Dunn. “Multiple Comparisons among Means”. Em: *Journal of the American Statistical Association* 56.293 (1961), pp. 52–64. DOI: [10.1080/01621459.1961.10482090](https://doi.org/10.1080/01621459.1961.10482090). eprint: <http://www.tandfonline.com/doi/pdf/10.1080/01621459.1961.10482090>. URL: <http://www.tandfonline.com/doi/abs/10.1080/01621459.1961.10482090>.
- [Dun73] J. C. Dunn. “A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters”. Em: *Journal of Cybernetics* 3.3 (1973), pp. 32–57. DOI: [10.1080/01969727308546046](https://doi.org/10.1080/01969727308546046). eprint: <http://dx.doi.org/10.1080/01969727308546046>. URL: <http://dx.doi.org/10.1080/01969727308546046>.
- [Dun74] J. C. Dunn. “Well separated clusters and optimal fuzzy-partitions”. Em: *Journal of Cybernetics* 4 (1974), pp. 95–104.
- [Est+96] Martin Ester et al. “A density-based algorithm for discovering clusters in large spatial databases with noise”. Em: AAAI Press, 1996, pp. 226–231.
- [Fis36] R. A. Fisher. “The Use of Multiple Measurements in Taxonomic Problems”. Em: *Annals of Eugenics* 7.7 (1936), pp. 179–188.
- [Fla+12] Christoph Flath et al. “Cluster Analysis of Smart Metering Data”. Em: *WIRTSCHAFTSINFORMATIK* 54.1 (2012), pp. 33–42. ISSN: 1861-8936. DOI: [10.1007/s11576-011-0309-8](https://doi.org/10.1007/s11576-011-0309-8). URL: <http://dx.doi.org/10.1007/s11576-011-0309-8>.
- [Fri40] Milton Friedman. “A Comparison of Alternative Tests of Significance for the Problem of m Rankings”. Em: *The Annals of Mathematical Statistics* 11.1 (1940), pp. 86–92. ISSN: 00034851. DOI: [10.2307/2235971](https://doi.org/10.2307/2235971). URL: <http://dx.doi.org/10.2307/2235971>.
- [GB13] Rafael Giusti e Gustavo E. A. P. A. Batista. “An Empirical Comparison of Dissimilarity Measures for Time Series Classification”. Em: *Proceedings of the 2013 Brazilian Conference on Intelligent Systems*. BRACIS ’13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 82–88. ISBN: 978-0-7695-5092-3. DOI: [10.1109/BRACIS.2013.22](https://doi.org/10.1109/BRACIS.2013.22). URL: <http://dx.doi.org/10.1109/BRACIS.2013.22>.
- [HA85] Lawrence Hubert e Phipps Arabie. “Comparing partitions”. Em: *Journal of Classification* 2.1 (1985), pp. 193–218. ISSN: 1432-1343. DOI: [10.1007/BF01908075](https://doi.org/10.1007/BF01908075). URL: <http://dx.doi.org/10.1007/BF01908075>.
- [HVB00] Maria Halkidi, Michalis Vazirgiannis e Yannis Batistakis. “Quality Scheme Assessment in the Clustering Process”. Em: *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*. PKDD ’00. London, UK, UK: Springer-Verlag, 2000, pp. 265–276. ISBN: 3-540-41066-X. URL: <http://dl.acm.org/citation.cfm?id=645804.669820>.
- [J+01] Eric Jones, Travis Oliphant, Pearu Peterson et al. *SciPy: Open source scientific tools for Python*. [Online; accessed <today>]. 2001. URL: <http://www.scipy.org/>.
- [Kai13] Shan-lin Yang e Chao Shen Kai-le Zhou. “A review of electric load classification in smart grid environment”. Em: *SIAM J. Numer. Anal.* 24 (2013), pp. 103–110.

- [Ken38] M. G. Kendall. “A New Measure of Rank Correlation”. Em: *Biometrika* 30.1/2 (1938), pp. 81–93. ISSN: 00063444. URL: <http://www.jstor.org/stable/2332226>.
- [KJ14] Amin Karami e Ronnie Johansson. “Article: Choosing DBSCAN Parameters Automatically using Differential Evolution”. Em: *International Journal of Computer Applications* 91.7 (abr. de 2014). Full text available, pp. 1–11.
- [KK03] Eamonn Keogh e Shruti Kasetty. “On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration”. Em: *Data Min. Knowl. Discov.* 7.4 (out. de 2003), pp. 349–371. ISSN: 1384-5810. DOI: [10.1023/A:1024988512476](https://doi.org/10.1023/A:1024988512476). URL: <http://dx.doi.org/10.1023/A:1024988512476>.
- [KR05] Eamonn Keogh e Chotirat Ann Ratanamahatana. “Exact Indexing of Dynamic Time Warping”. Em: *Knowl. Inf. Syst.* 7.3 (mar. de 2005), pp. 358–386. ISSN: 0219-1377. DOI: [10.1007/s10115-004-0154-9](https://doi.org/10.1007/s10115-004-0154-9). URL: <http://dx.doi.org/10.1007/s10115-004-0154-9>.
- [KR87] L. Kaufman e P.J. Rousseeuw. *Clustering by Means of Medoids*. Reports of the Faculty of Mathematics and Informatics. Faculty of Mathematics e Informatics, 1987. URL: <https://books.google.com.br/books?id=HK-4GwAACAAJ>.
- [Lin+03] Jessica Lin et al. “A Symbolic Representation of Time Series, with Implications for Streaming Algorithms”. Em: *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. DMKD '03. San Diego, California: ACM, 2003, pp. 2–11. DOI: [10.1145/882082.882086](https://doi.org/10.1145/882082.882086). URL: <http://doi.acm.org/10.1145/882082.882086>.
- [Llo82] S. Lloyd. “Least Squares Quantization in PCM”. Em: *IEEE Trans. Inf. Theor.* 28.2 (set. de 1982), pp. 129–137. ISSN: 0018-9448. DOI: [10.1109/TIT.1982.1056489](https://doi.org/10.1109/TIT.1982.1056489). URL: <http://dx.doi.org/10.1109/TIT.1982.1056489>.
- [Mah36] Prasanta Chandra Mahalanobis. “On the generalized distance in statistics”. Em: *Proceedings of the National Institute of Sciences (Calcutta)* 2 (1936), pp. 49–55.
- [Mar08] Pierre-François Marteau. “Time Warp Edit Distance with Stiffness Adjustment for Time Series Matching”. Em: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 31.undefined (2008), pp. 306–318. ISSN: 0162-8828. DOI: [doi.ieeecomputersociety.org/10.1109/TPAMI.2008.76](https://doi.org/10.1109/TPAMI.2008.76).
- [MB02] U. Maulik e S. Bandyopadhyay. “Performance evaluation of some clustering algorithms and validity indices”. Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.12 (dez. de 2002), pp. 1650–1654. ISSN: 0162-8828. DOI: [10.1109/TPAMI.2002.1114856](https://doi.org/10.1109/TPAMI.2002.1114856).
- [Mei07] Marina Meilă. “Comparing Clusterings—an Information Based Distance”. Em: *J. Multivar. Anal.* 98.5 (maio de 2007), pp. 873–895. ISSN: 0047-259X. DOI: [10.1016/j.jmva.2006.11.013](https://doi.org/10.1016/j.jmva.2006.11.013). URL: <http://dx.doi.org/10.1016/j.jmva.2006.11.013>.

- [Min14] Alex Minnaar. *BWorld Robot Control Software*. <http://nbviewer.ipython.org/github/alexminnaar/time-series-classification-and-clustering/blob/master/Time%20Series%20Classification%20and%20Clustering.ipynb>. Online; acessado 02-Dezembro-2015. 2014.
- [MP07] Michael D. Morse e Jignesh M. Patel. “An Efficient and Accurate Method for Evaluating Time Series Similarity”. Em: *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. SIGMOD '07. Beijing, China: ACM, 2007, pp. 569–580. ISBN: 978-1-59593-686-8. DOI: [10.1145/1247480.1247544](https://doi.org/10.1145/1247480.1247544). URL: <http://doi.acm.org/10.1145/1247480.1247544>.
- [MV14] Pablo Montero e José A. Vilar. “TSclust: An R Package for Time Series Clustering”. Em: *Journal of Statistical Software* 62.1 (2014), pp. 1–43. URL: <http://www.jstatsoft.org/v62/i01/>.
- [Nem63] Peter Nemenyi. “Distribution-free multiple comparisons”. Tese de doutorado. New Jersey, USA: Princeton University, 1963.
- [Nov+13] Vilém Novák et al. “F-transform and Fuzzy Natural logic in Time Series Analysis.” Em: *EUSFLAT Conf.* Atlantis Press, 2013. ISBN: 978-90786-77-78-9. URL: <http://dblp.uni-trier.de/db/conf/eusflat/eusflat2013.html#NovakPPS13>.
- [Paz16] Pedro Pazzini. *Códigos fontes utilizados no desenvolvimento da dissertação de mestrado*. http://github.com/pedropazzini/sg_data. Online; acessado 02-Dezembro-2016. 2016.
- [Ped+11] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. Em: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.
- [PG16] John Paparrizos e Luis Gravano. “k-Shape: Efficient and Accurate Clustering of Time Series”. Em: *SIGMOD Rec.* 45.1 (jun. de 2016), pp. 69–76. ISSN: 0163-5808. DOI: [10.1145/2949741.2949758](https://doi.org/10.1145/2949741.2949758). URL: <http://doi.acm.org/10.1145/2949741.2949758>.
- [PKG11] François Petitjean, Alain Ketterlin e Pierre Gançarski. “A Global Averaging Method for Dynamic Time Warping, with Applications to Clustering”. Em: *Pattern Recogn.* 44.3 (mar. de 2011), pp. 678–693. ISSN: 0031-3203. DOI: [10.1016/j.patcog.2010.09.013](https://doi.org/10.1016/j.patcog.2010.09.013). URL: <http://dx.doi.org/10.1016/j.patcog.2010.09.013>.
- [PL12] Zoltán Prekopcsák e Daniel Lemire. “Time series classification by class-specific Mahalanobis distance measures.” Em: *Adv. Data Analysis and Classification* 6.3 (2012), pp. 185–200. URL: <http://dblp.uni-trier.de/db/journals/adac/adac6.html#PrekopcsakL12>.
- [Rak+12] Thanawin Rakthanmanon et al. “Searching and Mining Trillions of Time Series Subsequences Under Dynamic Time Warping”. Em: *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '12. Beijing, China: ACM, 2012, pp. 262–270. ISBN: 978-1-4503-1462-6. DOI: [10.1145/2339530.2339576](https://doi.org/10.1145/2339530.2339576). URL: <http://doi.acm.org/10.1145/2339530.2339576>.

- [RH07] Andrew Rosenberg e Julia Hirschberg. “V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure”. Em: *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. 2007, pp. 410–420.
- [Rou87a] Peter Rousseeuw. “Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis”. Em: *J. Comput. Appl. Math.* 20.1 (nov. de 1987), pp. 53–65. ISSN: 0377-0427. DOI: [10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7). URL: [http://dx.doi.org/10.1016/0377-0427\(87\)90125-7](http://dx.doi.org/10.1016/0377-0427(87)90125-7).
- [Rou87b] Peter Rousseeuw. “Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis”. Em: *J. Comput. Appl. Math.* 20.1 (nov. de 1987), pp. 53–65. ISSN: 0377-0427. DOI: [10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7). URL: [http://dx.doi.org/10.1016/0377-0427\(87\)90125-7](http://dx.doi.org/10.1016/0377-0427(87)90125-7).
- [RU11] Anand Rajaraman e Jeffrey David Ullman. *Mining of Massive Datasets*. New York, NY, USA: Cambridge University Press, 2011.
- [SA14] Joan Serrà e Josep Lluís Arcos. “An Empirical Evaluation of Similarity Measures for Time Series Classification”. Em: *CoRR* abs/1401.3973 (2014). URL: <http://arxiv.org/abs/1401.3973>.
- [SB99] Kilian Stoffel e Abdelkader Belkoniene. “Parallel k/h-means clustering for large data sets”. Em: *European Conference on Parallel Processing*. Springer. 1999, pp. 1451–1454.
- [SG03] Alexander Strehl e Joydeep Ghosh. “Cluster Ensembles — a Knowledge Reuse Framework for Combining Multiple Partitions”. Em: *J. Mach. Learn. Res.* 3 (mar. de 2003), pp. 583–617. ISSN: 1532-4435. DOI: [10.1162/153244303321897735](https://doi.org/10.1162/153244303321897735). URL: <http://dx.doi.org/10.1162/153244303321897735>.
- [Tse+08] G.J. Tsekouras et al. “A pattern recognition methodology for evaluation of load profiles and typical days of large electricity customers”. Em: *Electric Power Systems Research* 78.9 (2008), pp. 1494–1510. ISSN: 0378-7796. DOI: [http://dx.doi.org/10.1016/j.epsr.2008.01.010](https://doi.org/10.1016/j.epsr.2008.01.010). URL: <http://www.sciencedirect.com/science/article/pii/S0378779608000278>.
- [VEB10] Nguyen Xuan Vinh, Julien Epps e James Bailey. “Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance”. Em: *J. Mach. Learn. Res.* 11 (dez. de 2010), pp. 2837–2854. ISSN: 1532-4435. URL: <http://dl.acm.org/citation.cfm?id=1756006.1953024>.
- [War05] T. Warren Liao. “Clustering of Time Series Data-a Survey”. Em: *Pattern Recogn.* 38.11 (nov. de 2005), pp. 1857–1874. ISSN: 0031-3203. DOI: [10.1016/j.patcog.2005.01.025](https://doi.org/10.1016/j.patcog.2005.01.025). URL: <http://dx.doi.org/10.1016/j.patcog.2005.01.025>.
- [WCV11] S. van der Walt, S. C. Colbert e G. Varoquaux. “The NumPy Array: A Structure for Efficient Numerical Computation”. Em: *Computing in Science Engineering* 13.2 (mar. de 2011), pp. 22–30. ISSN: 1521-9615. DOI: [10.1109/MCSE.2011.37](https://doi.org/10.1109/MCSE.2011.37).

- [XB91] X. L. Xie e G. Beni. “A validity measure for fuzzy clustering”. Em: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.8 (ago. de 1991), pp. 841–847. ISSN: 0162-8828. DOI: [10.1109/34.85677](https://doi.org/10.1109/34.85677).
- [ZMH09] Weizhong Zhao, Huifang Ma e Qing He. “Parallel k-means clustering based on mapreduce”. Em: *IEEE International Conference on Cloud Computing*. Springer. 2009, pp. 674–679.

Apêndice A

Curvas do UCR *Time Series*

A seguir seguem as curvas disponibilizadas em (Y. Chen et al. 2015) e utilizadas nos experimentos teóricos.

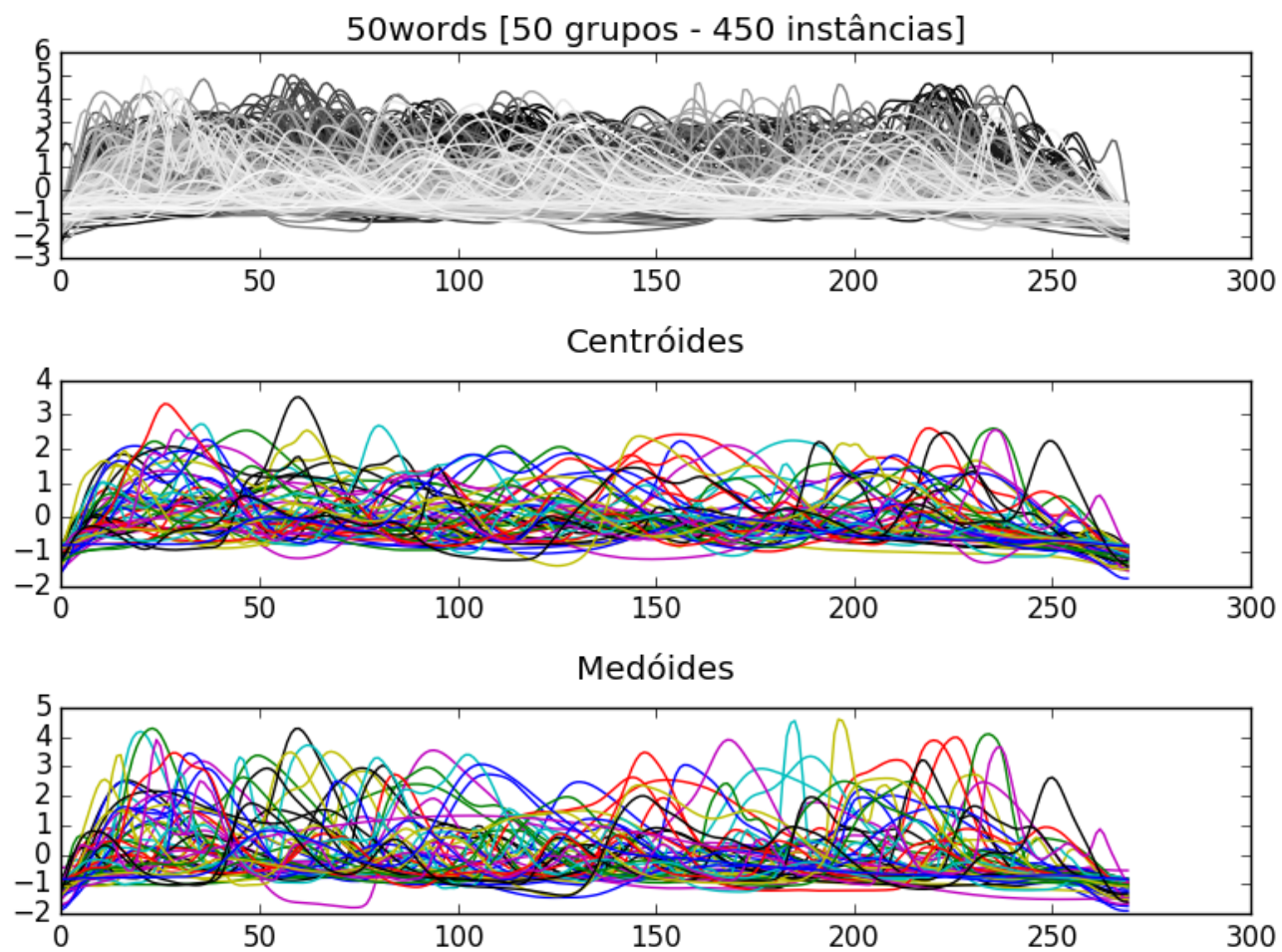


Figura A.1: Conjunto de dados '50words'.

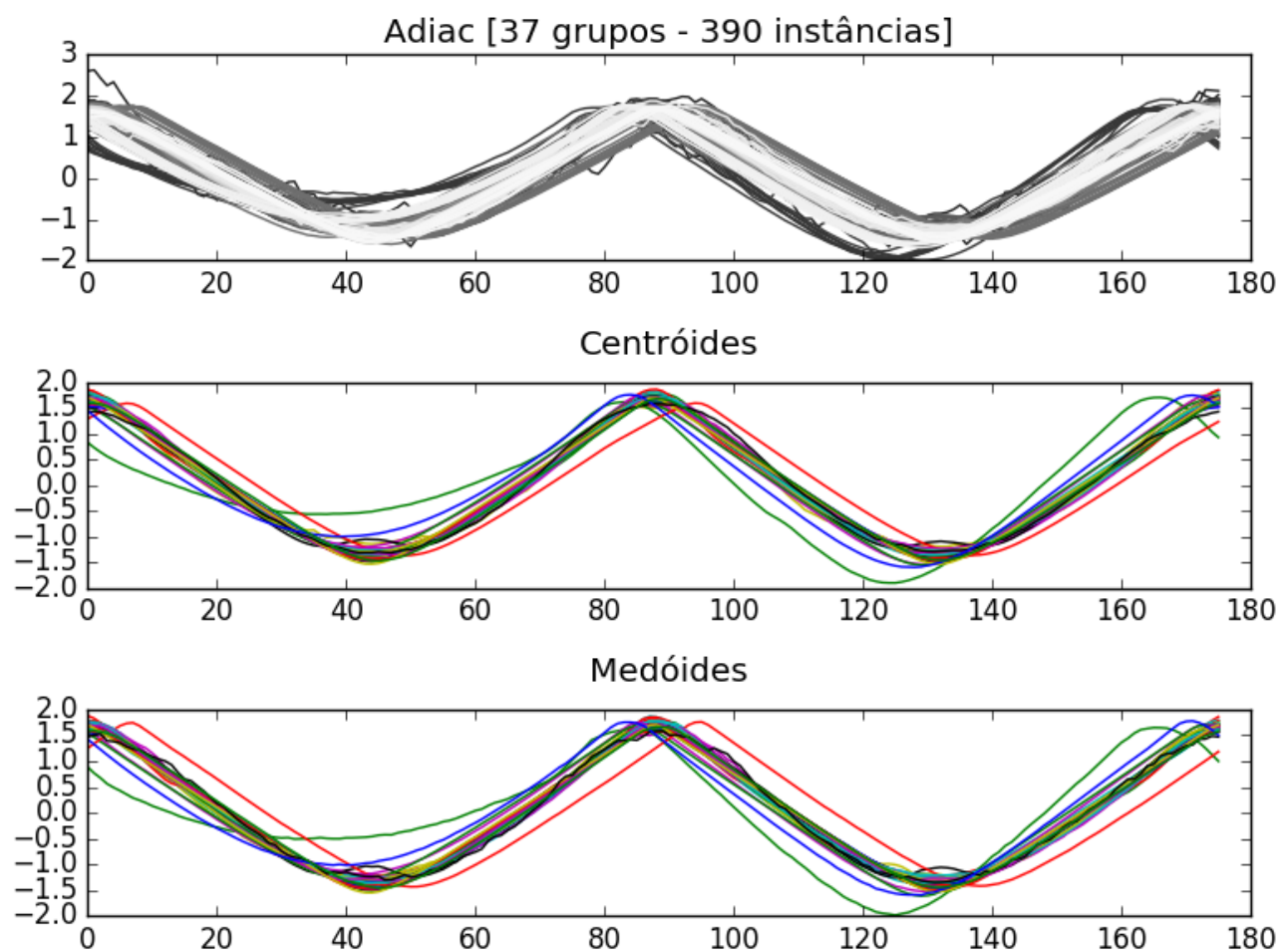


Figura A.2: Conjunto de dados 'Adiac'.

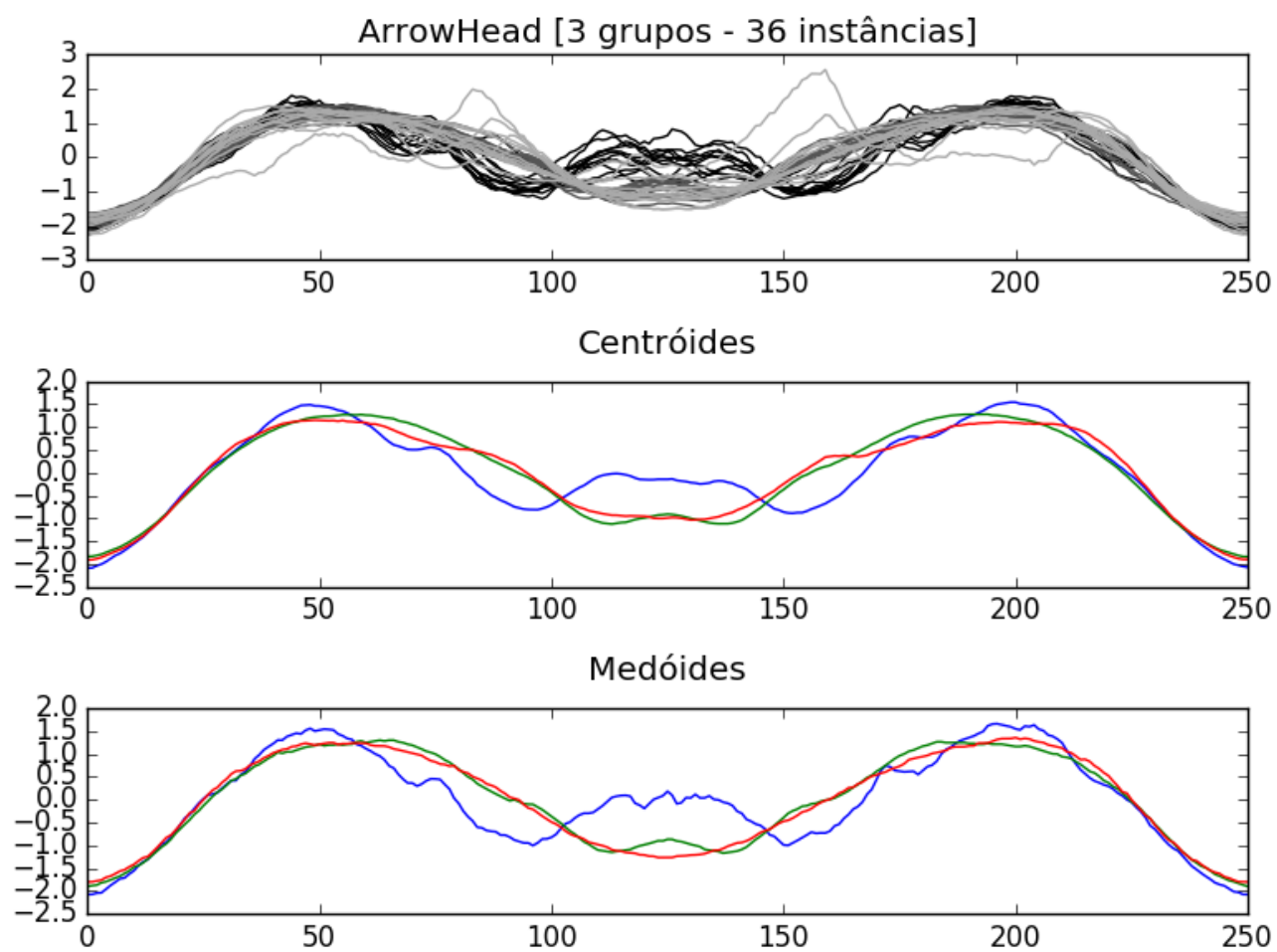


Figura A.3: Conjunto de dados 'ArrowHead'.

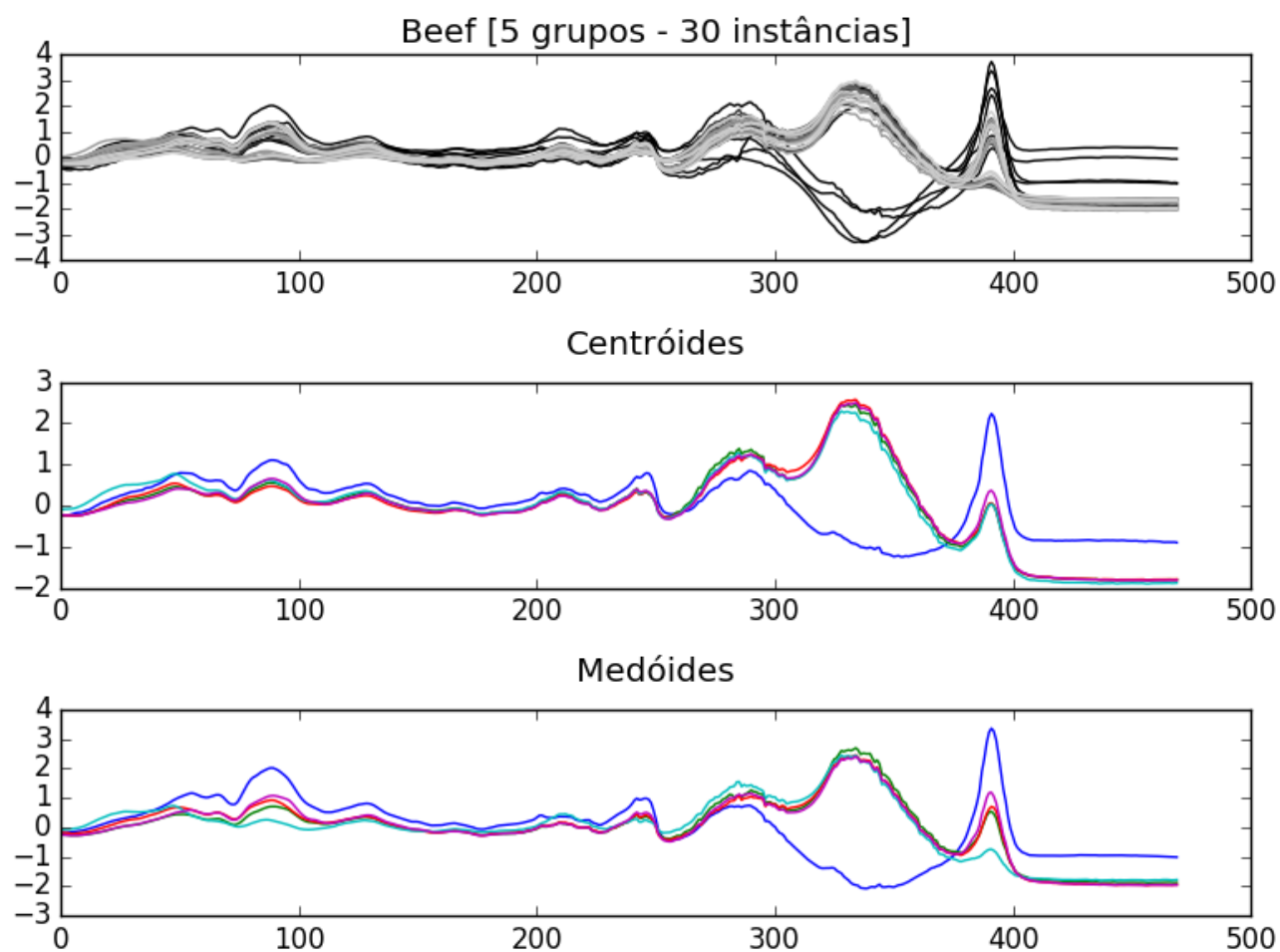


Figura A.4: Conjunto de dados 'Beef'.

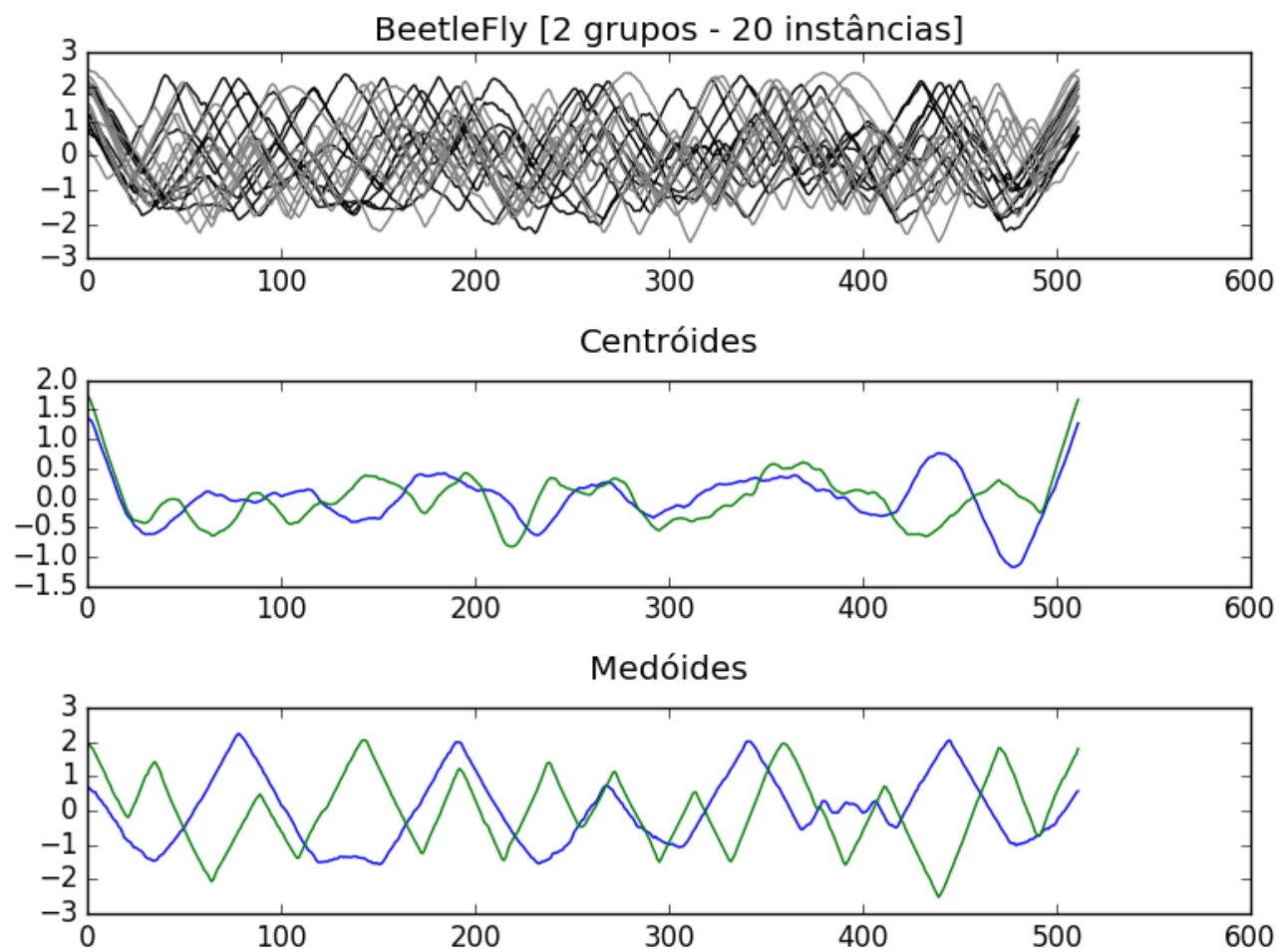


Figura A.5: Conjunto de dados 'BeetleFly'.

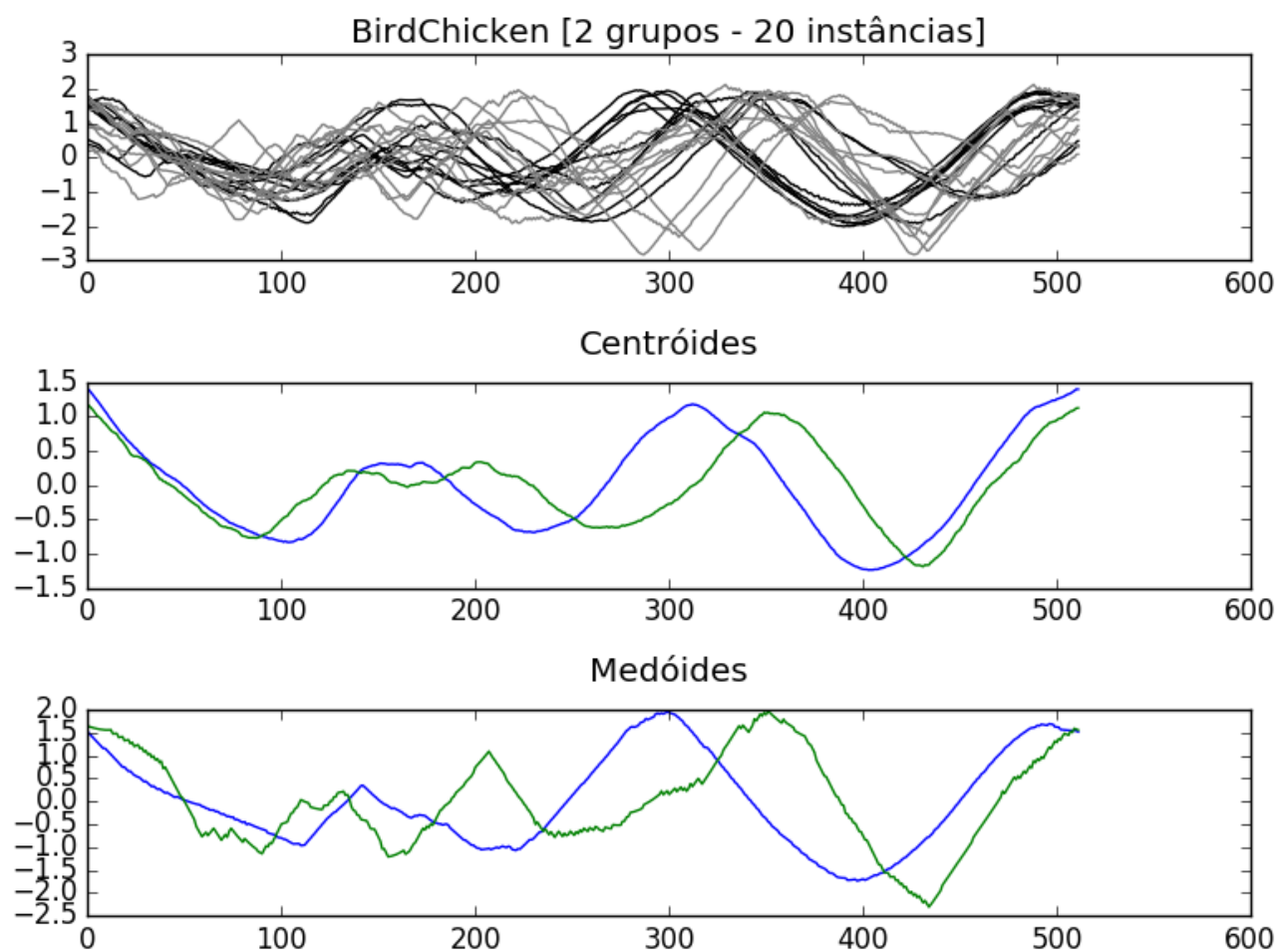


Figura A.6: Conjunto de dados 'BirdChicken'.

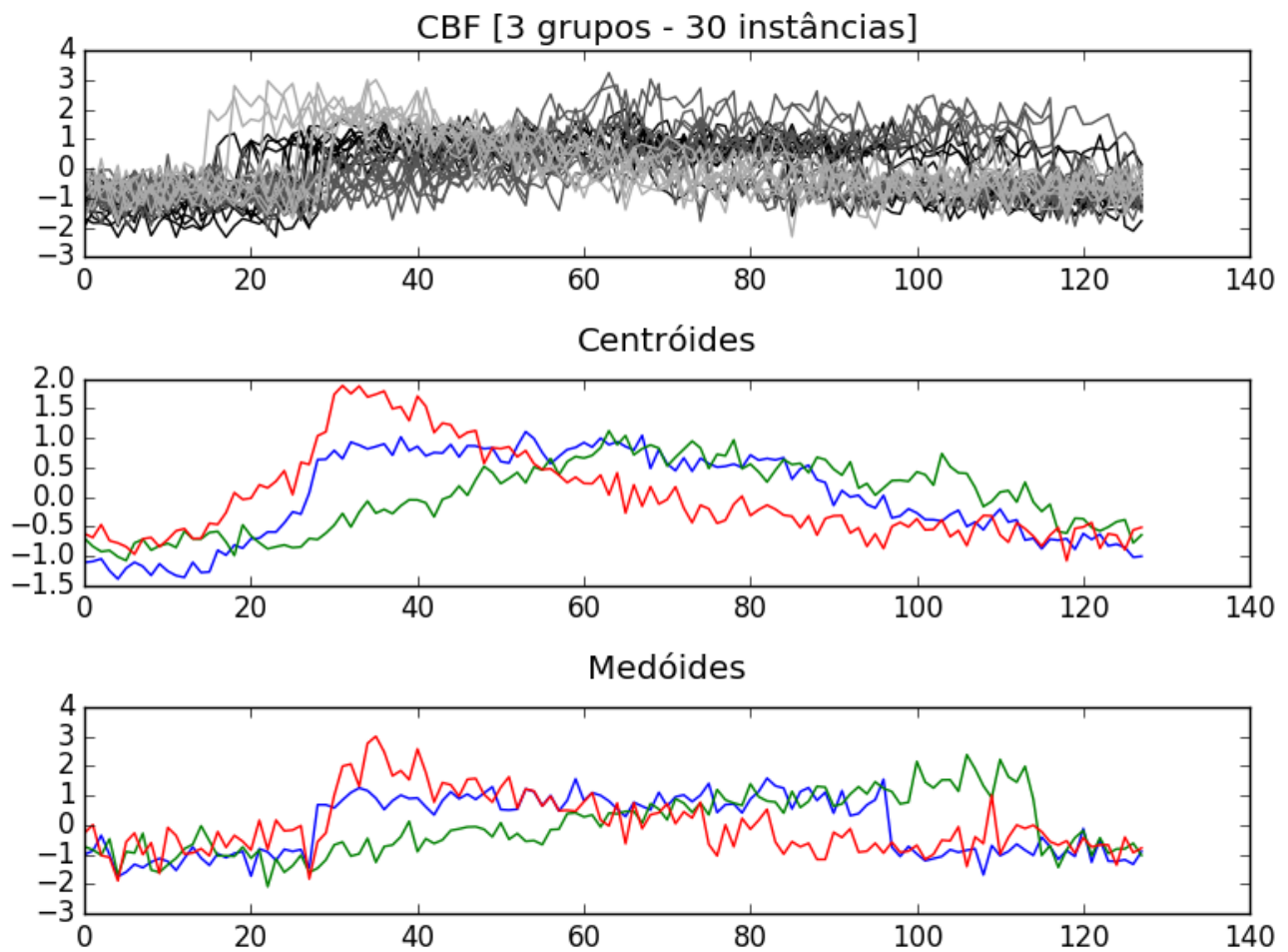


Figura A.7: Conjunto de dados 'CBF'.

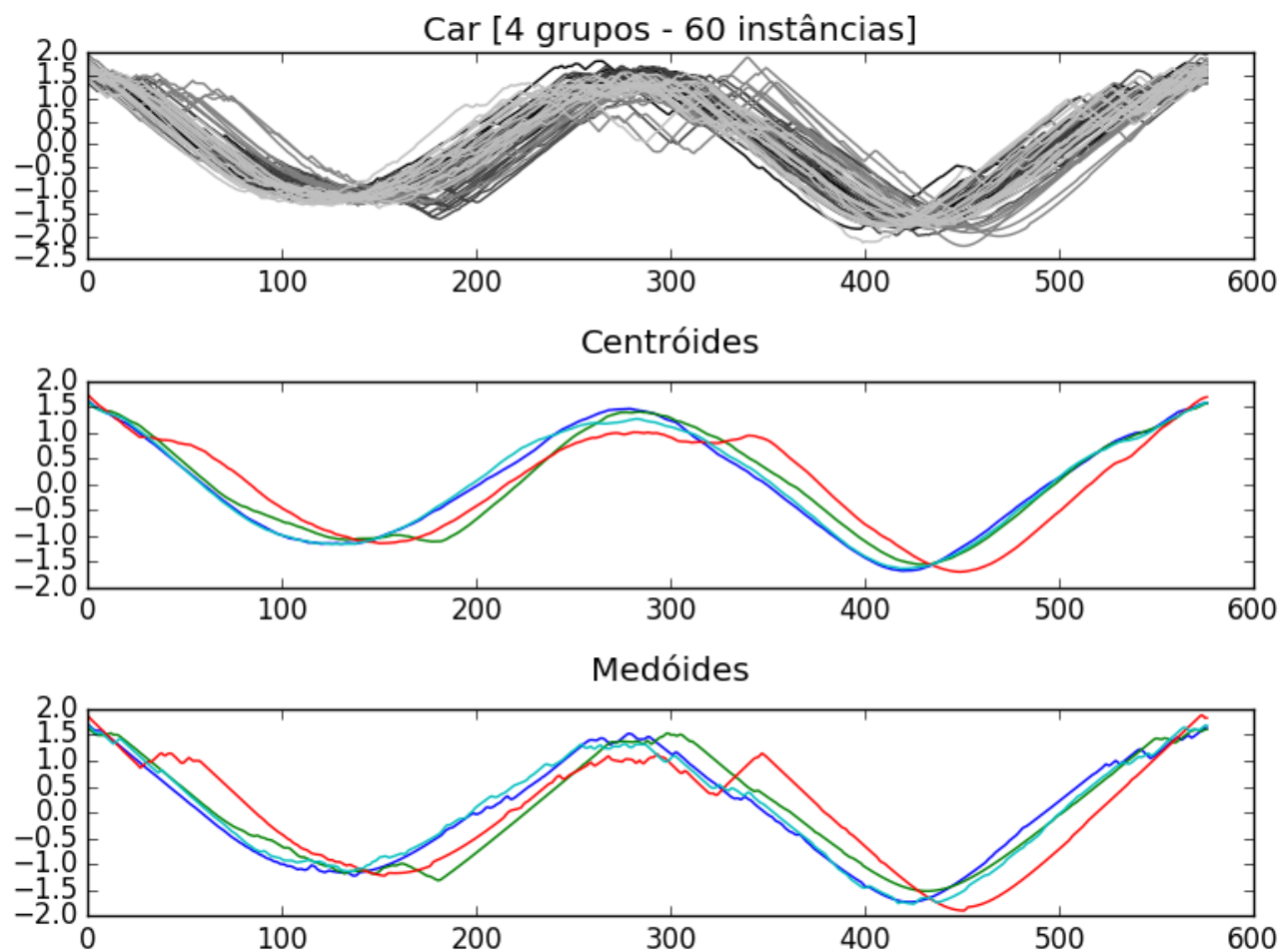


Figura A.8: Conjunto de dados 'Car'.

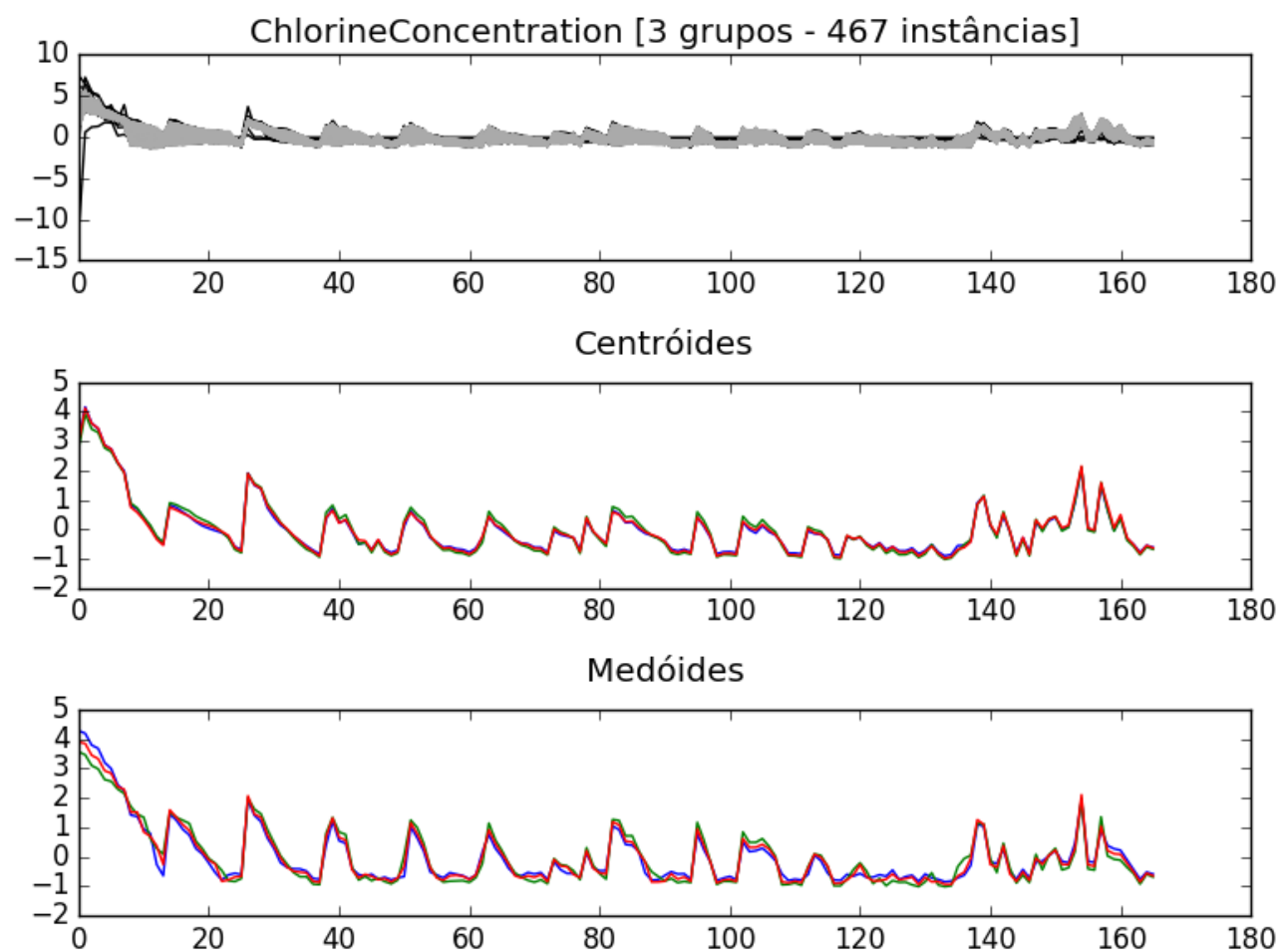


Figura A.9: Conjunto de dados 'ChlorineConcentration'.

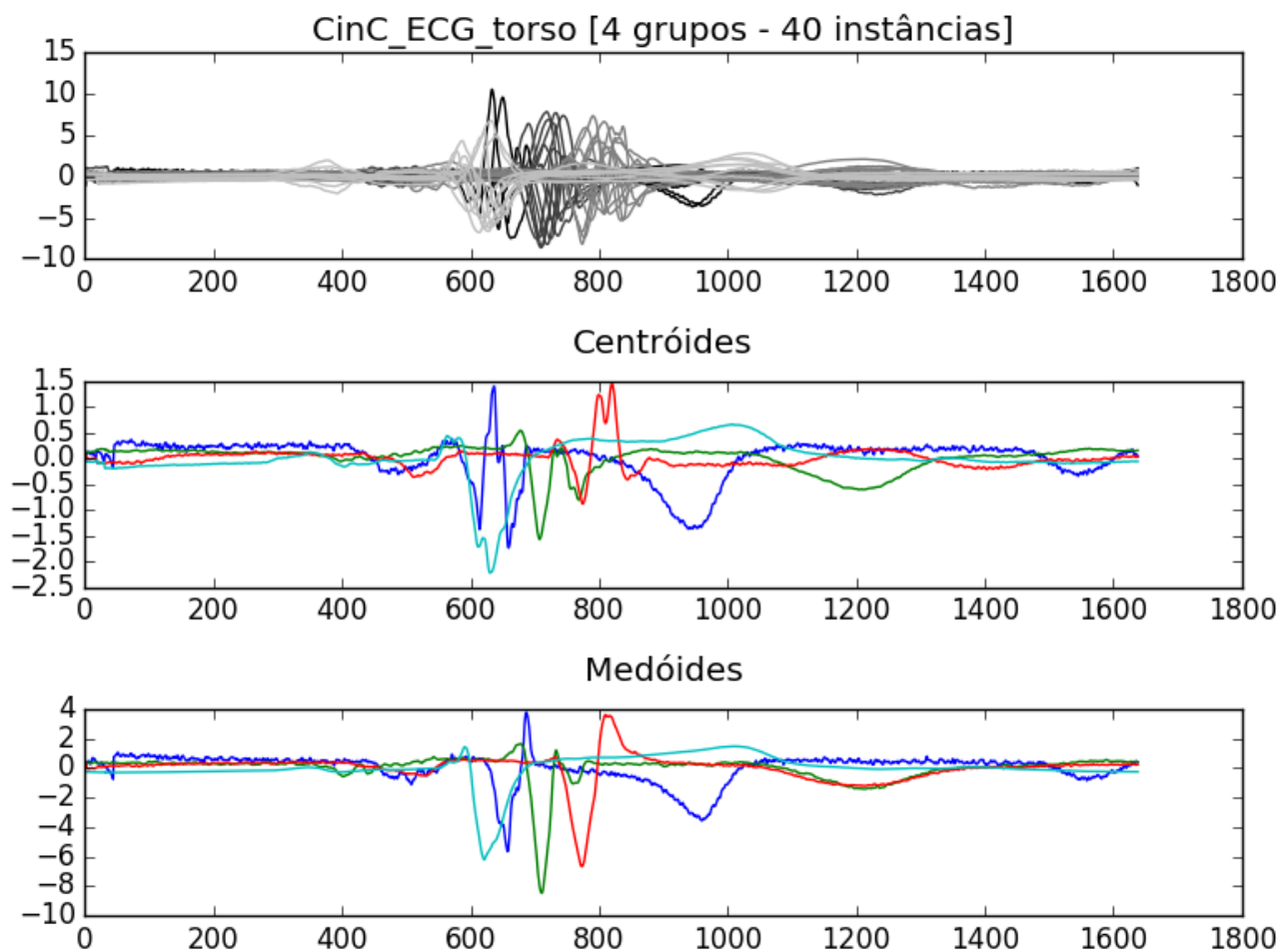


Figura A.10: Conjunto de dados 'CinC_ECG_torso'.

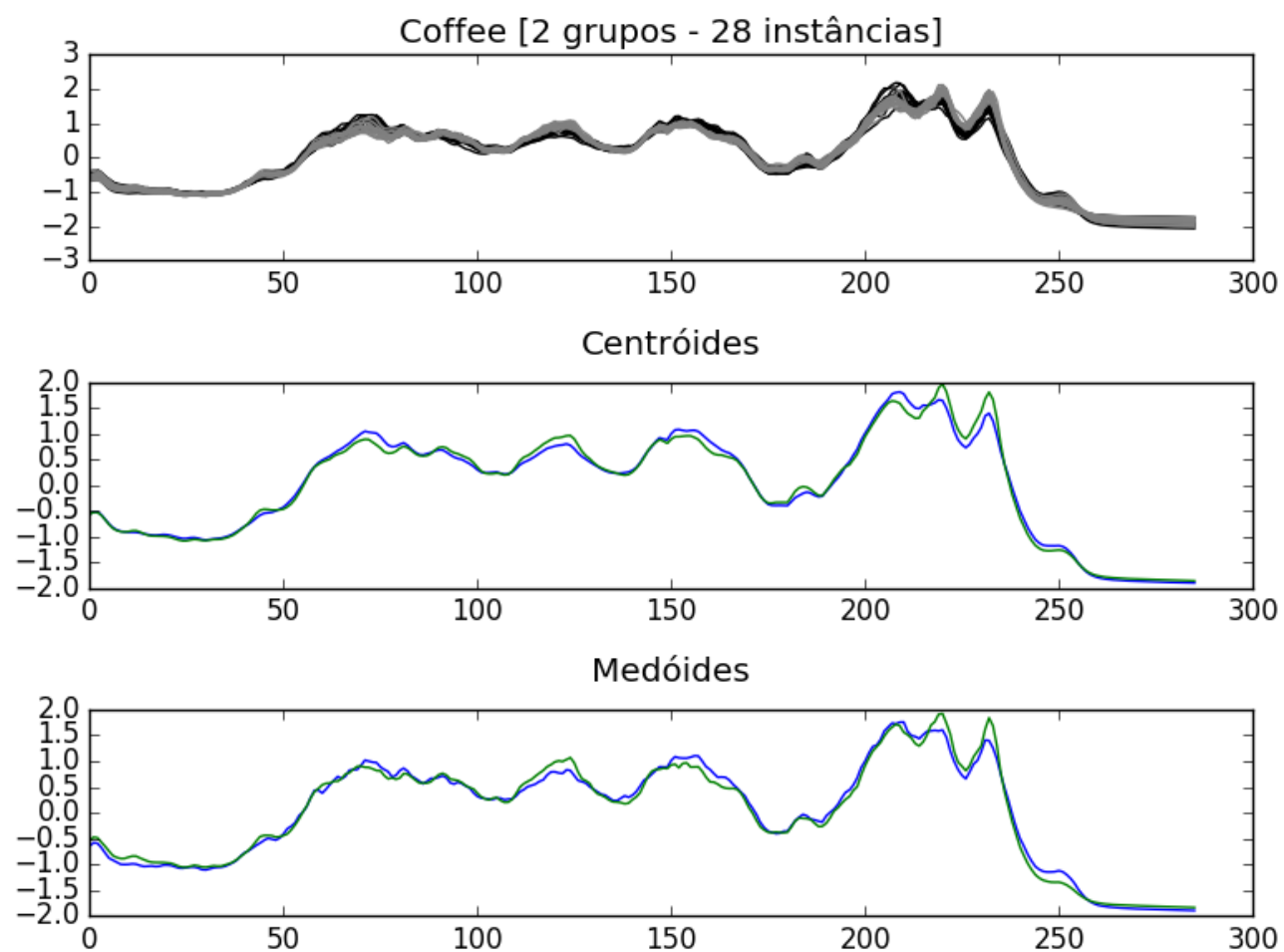


Figura A.11: Conjunto de dados 'Coffee'.

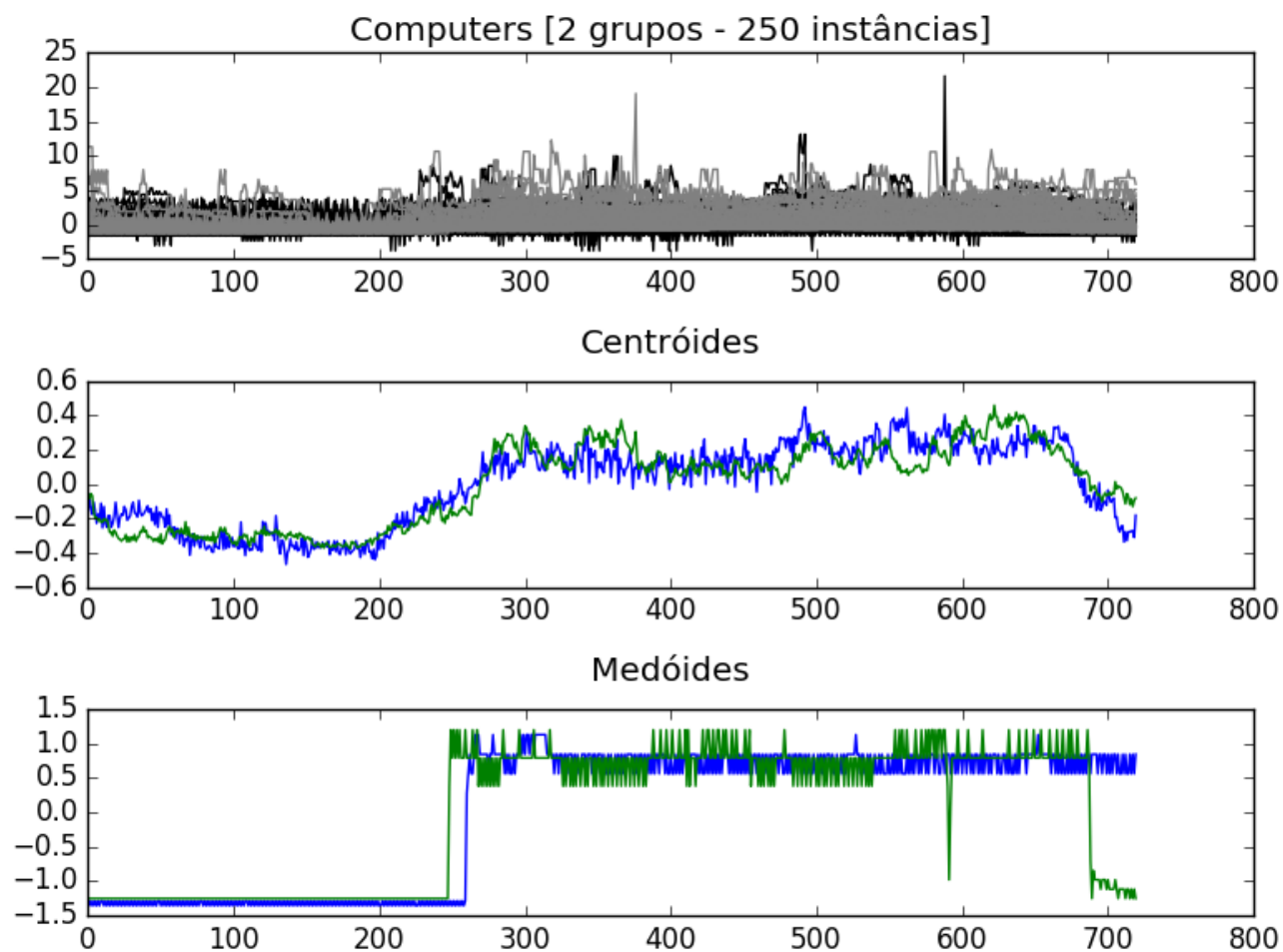


Figura A.12: Conjunto de dados 'Computers'.

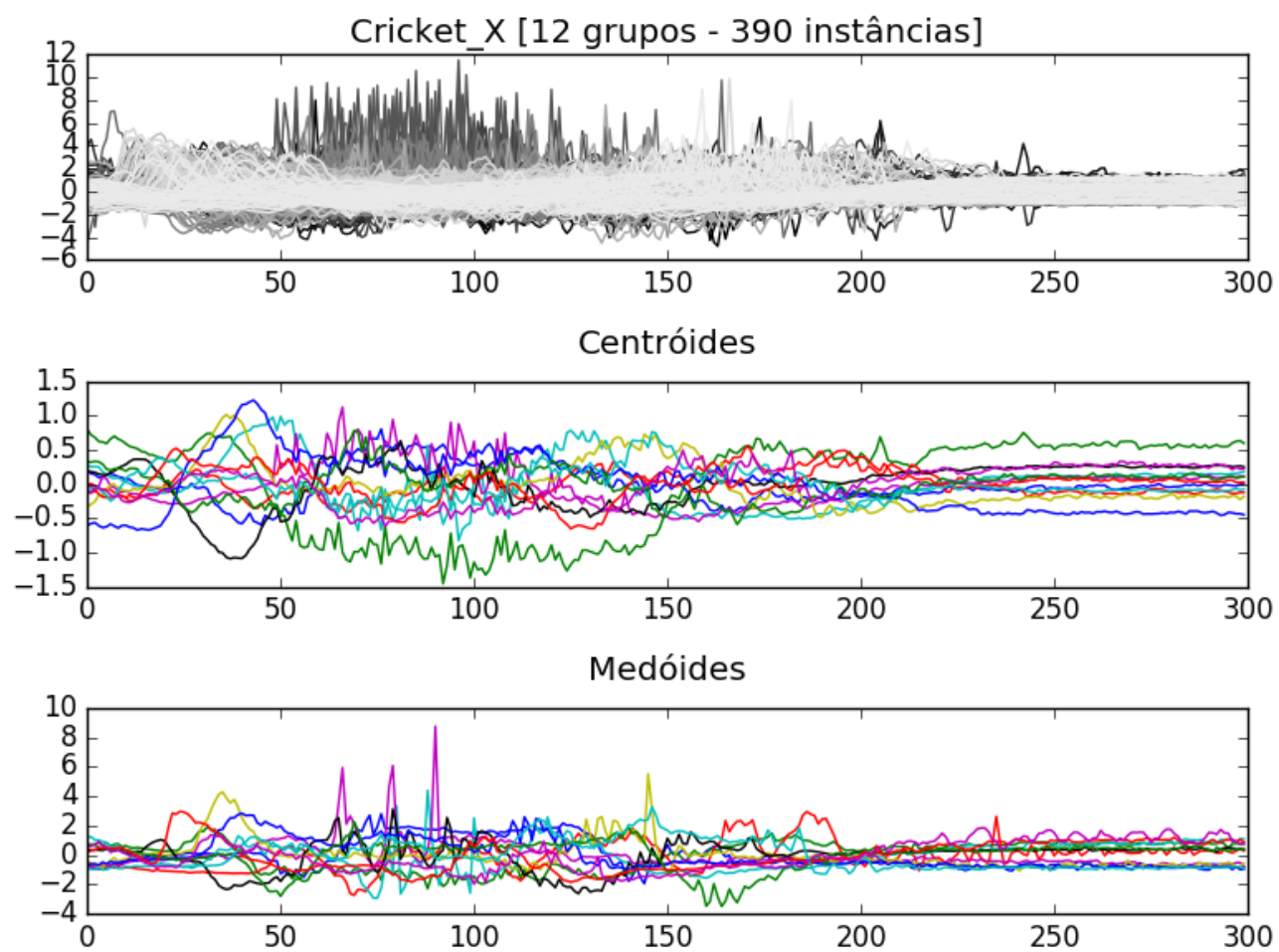


Figura A.13: Conjunto de dados 'Cricket_X'.

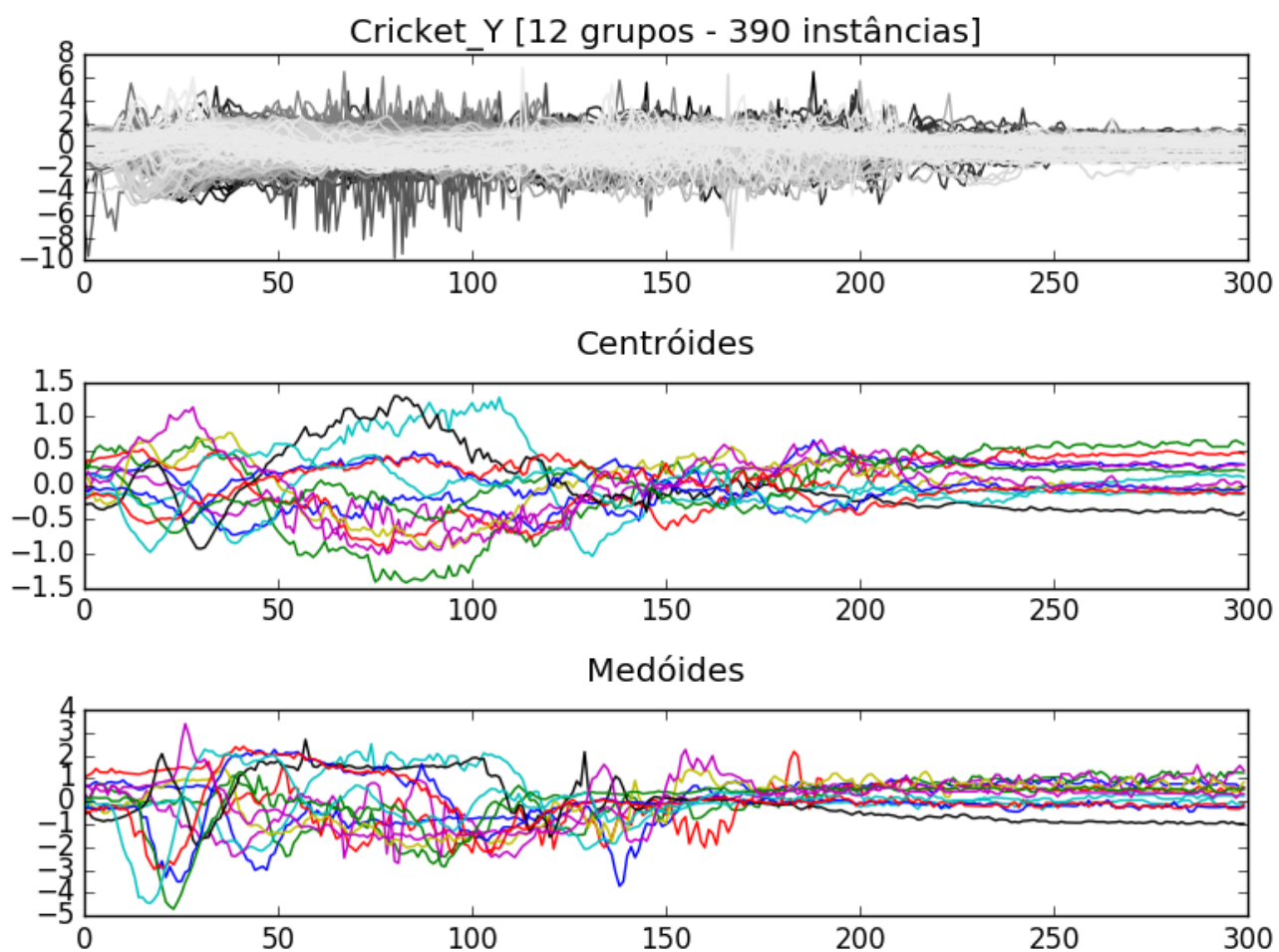


Figura A.14: Conjunto de dados 'Cricket_Y'.

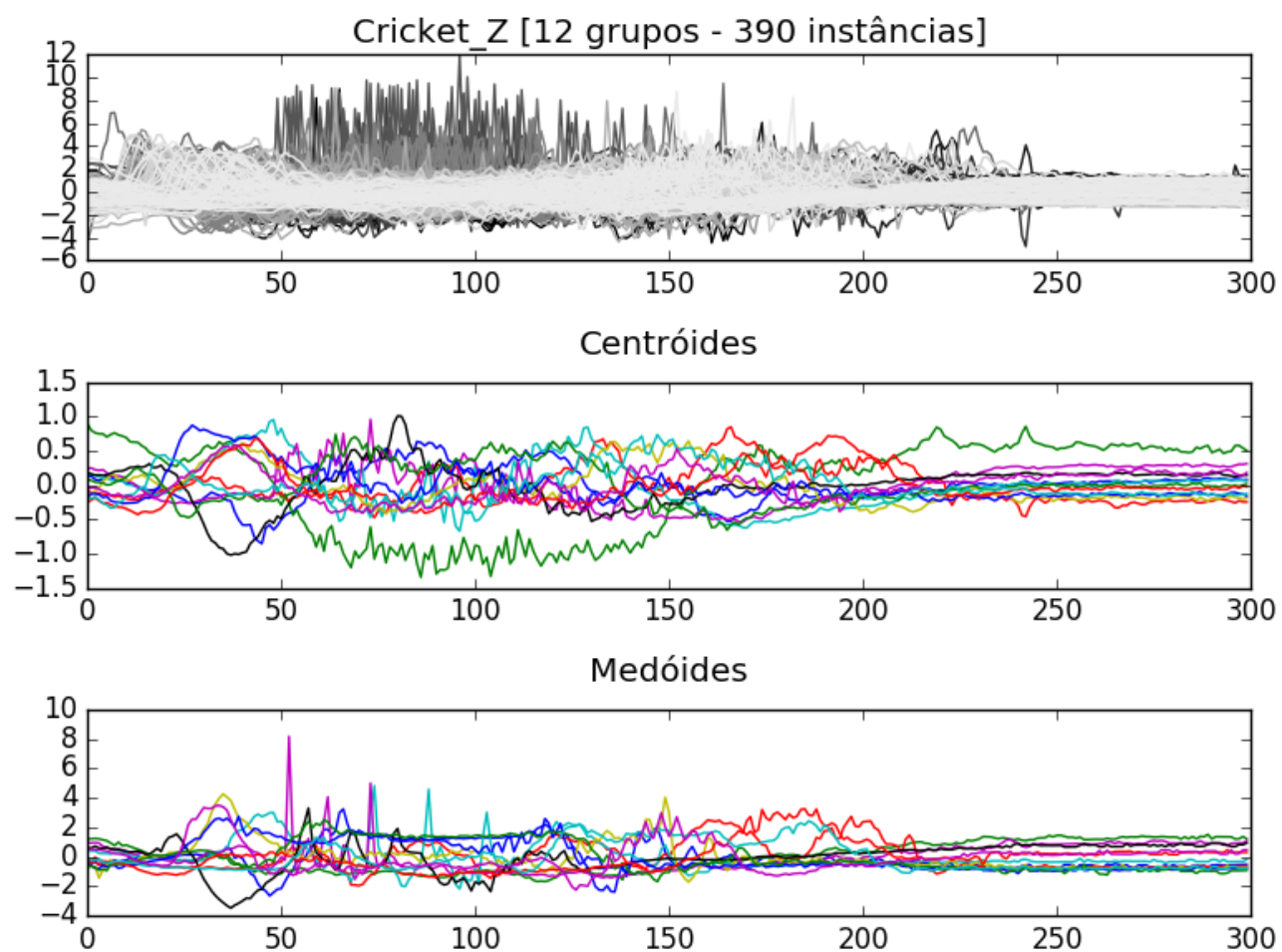


Figura A.15: Conjunto de dados 'Cricket_Z'.

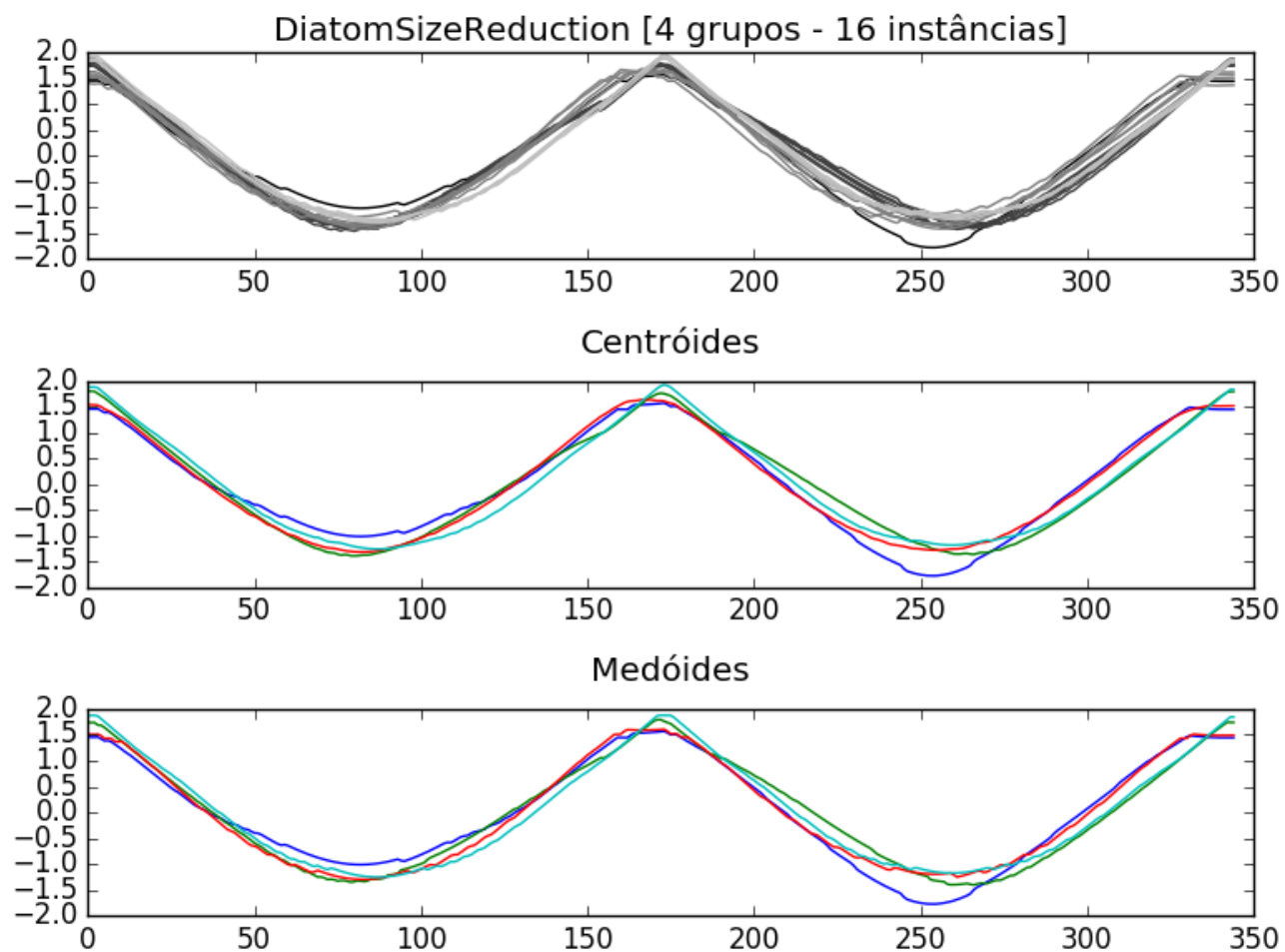


Figura A.16: Conjunto de dados 'DiatomSizeReduction'.

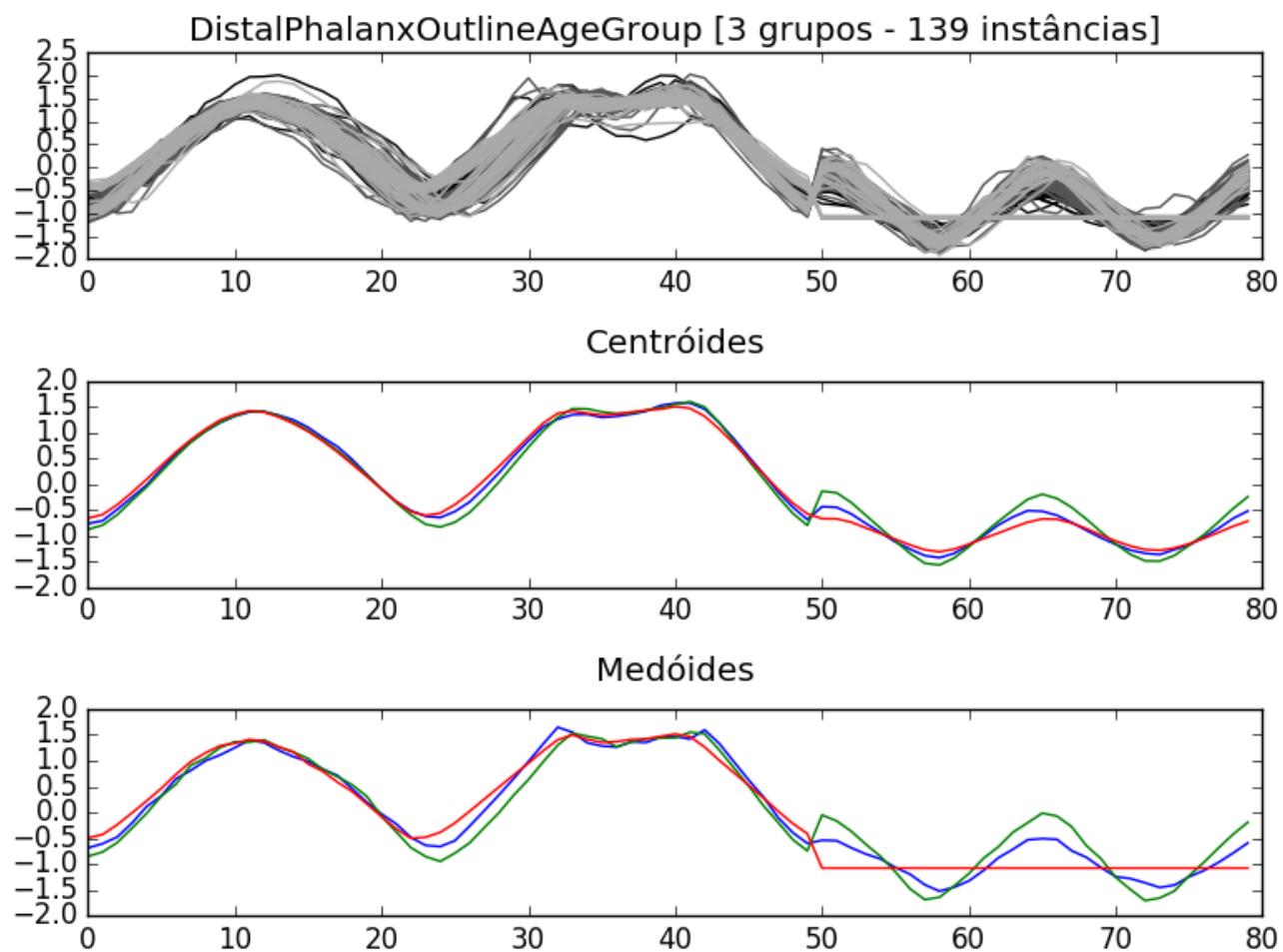


Figura A.17: Conjunto de dados 'DistalPhalanxOutlineAgeGroup'.

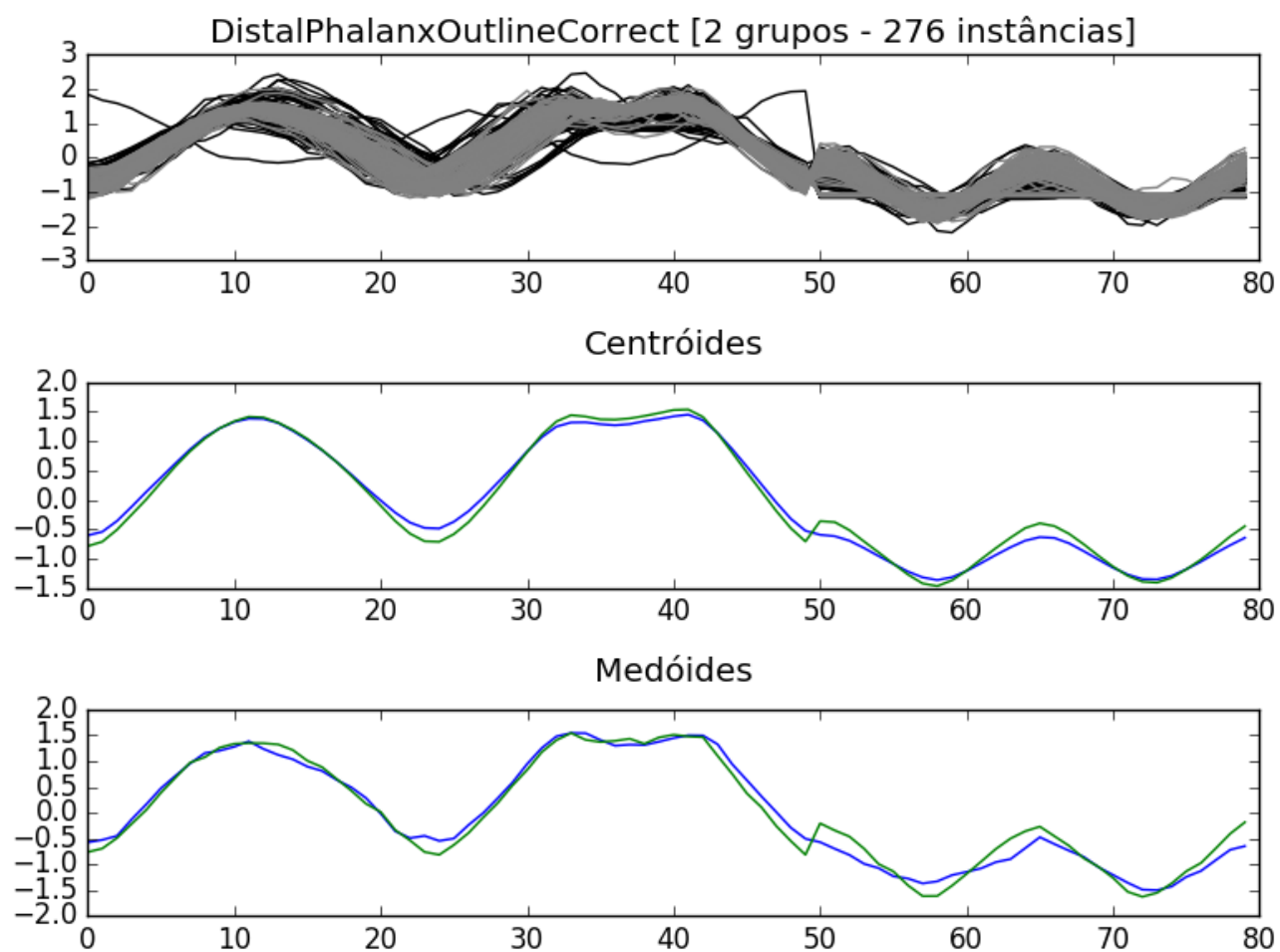


Figura A.18: Conjunto de dados 'DistalPhalanxOutlineCorrect'.

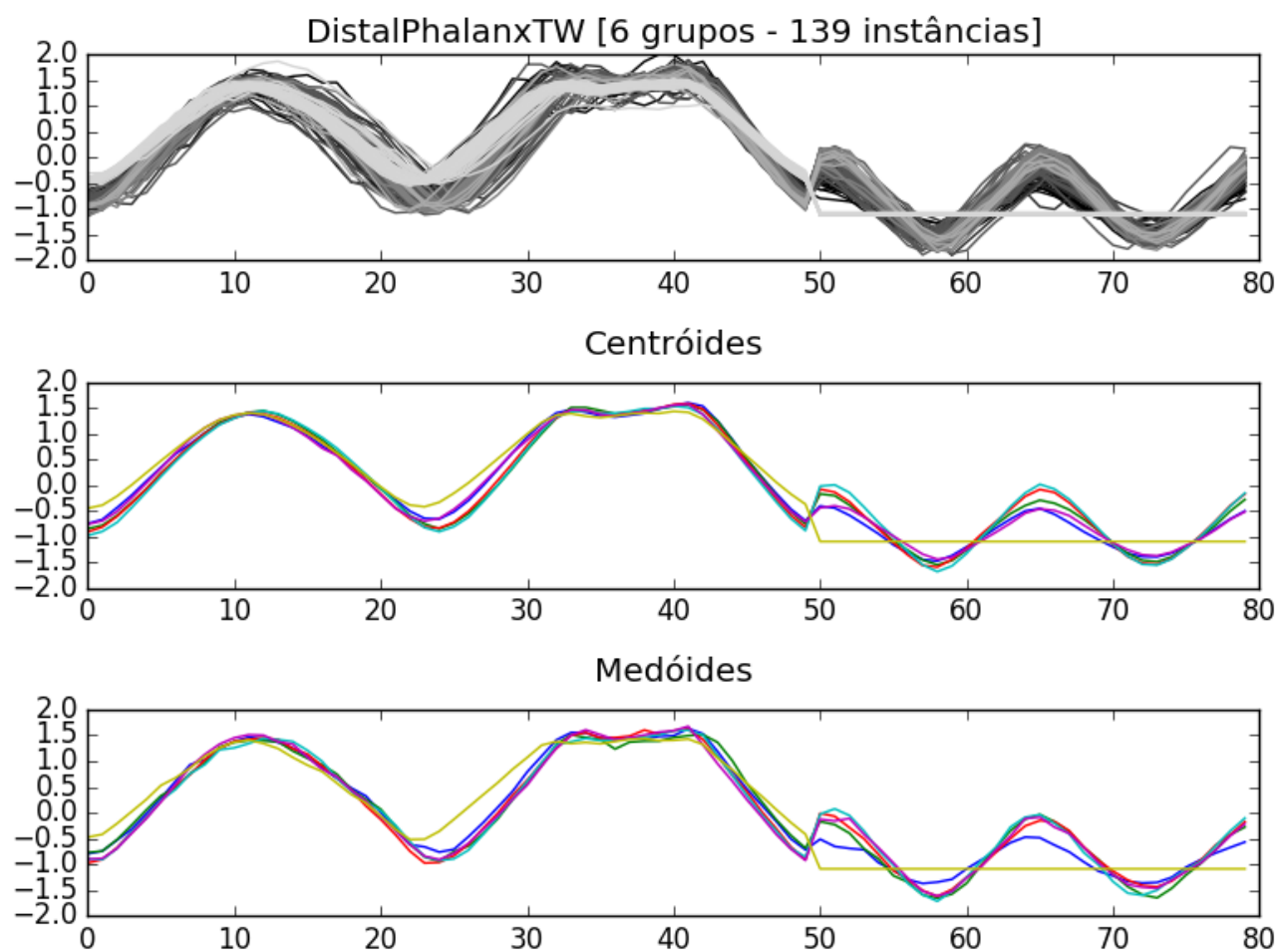


Figura A.19: Conjunto de dados 'DistalPhalanxTW'.

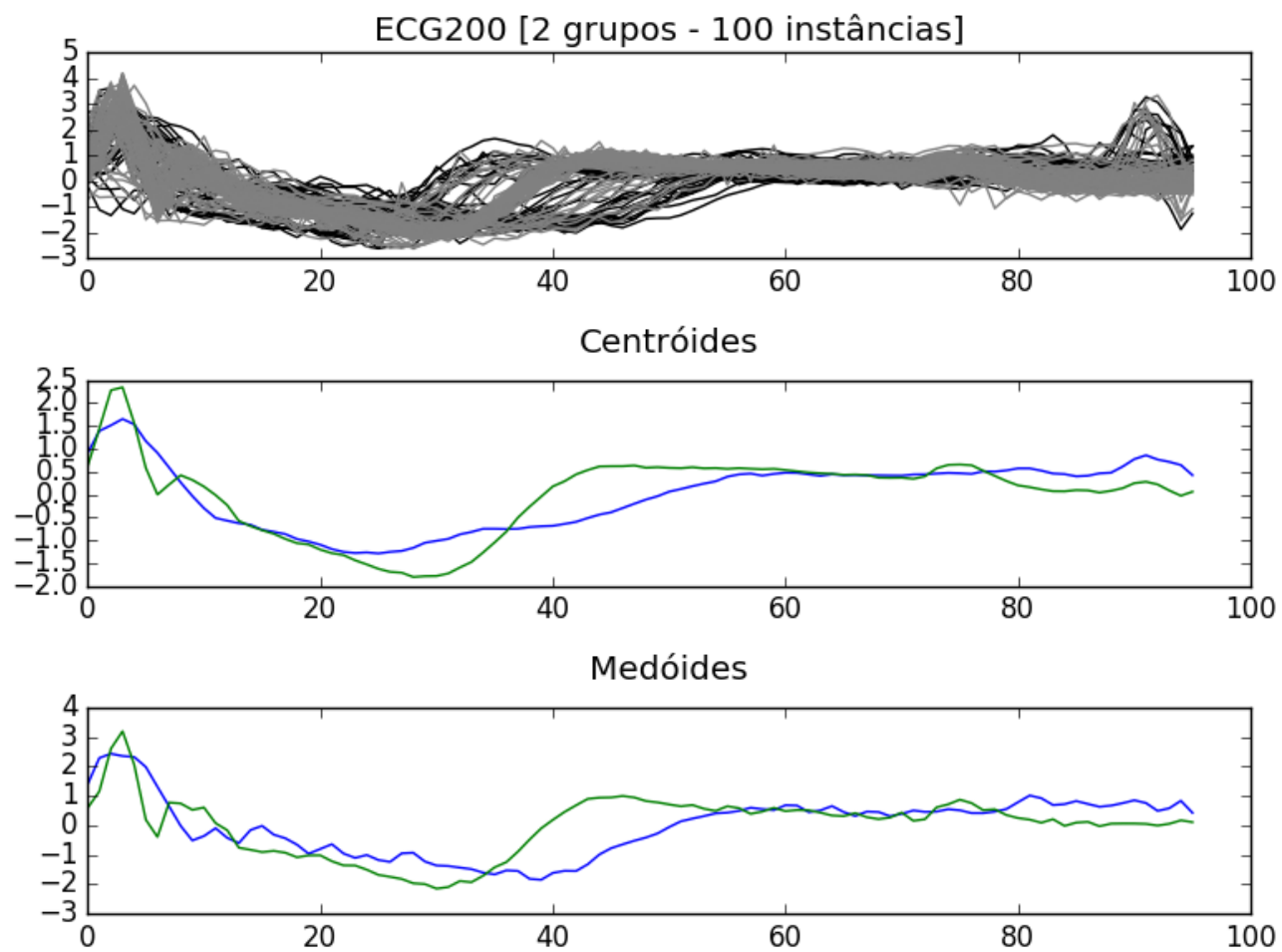


Figura A.20: Conjunto de dados 'ECG200'.

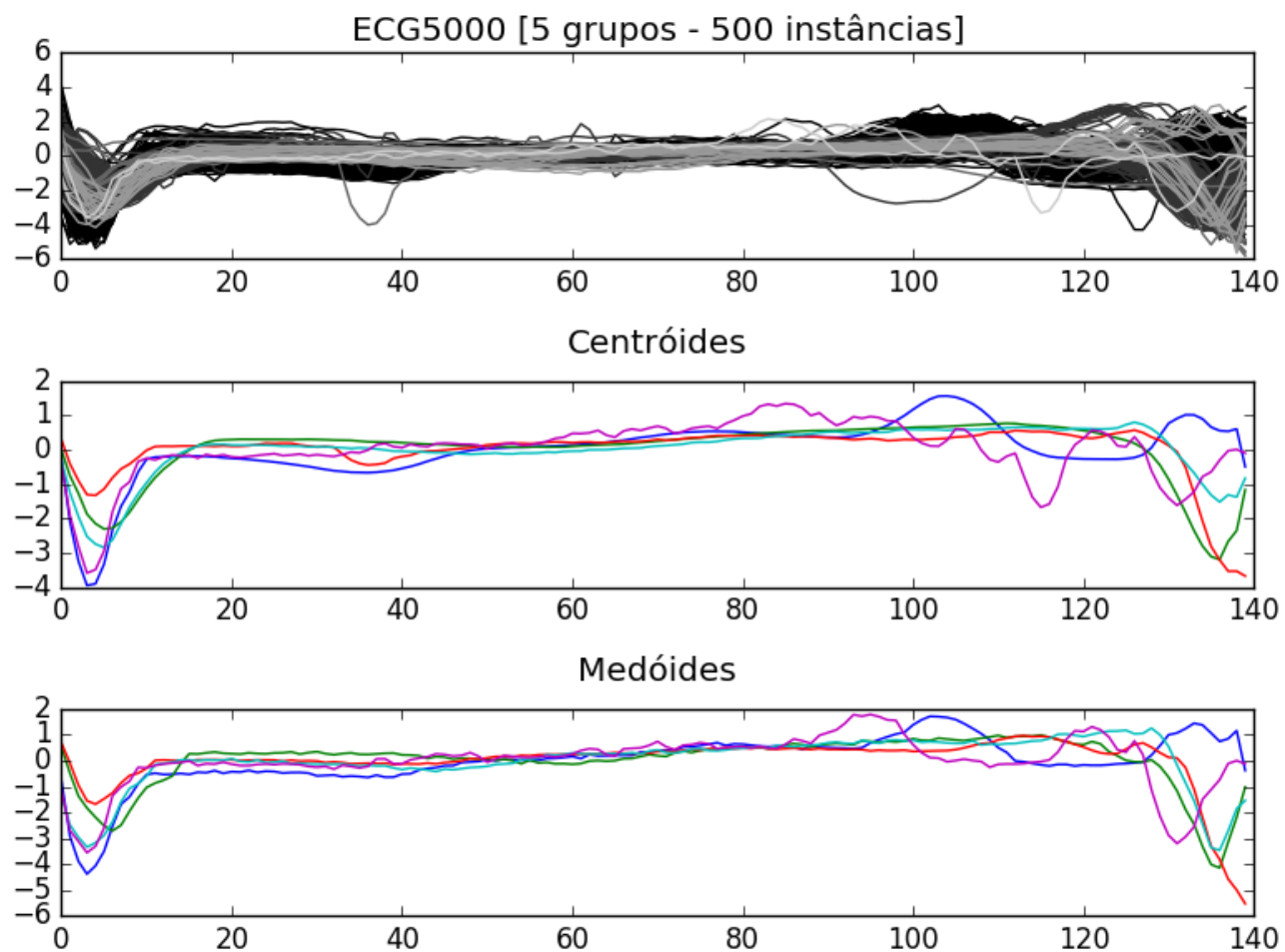


Figura A.21: Conjunto de dados 'ECG5000'.

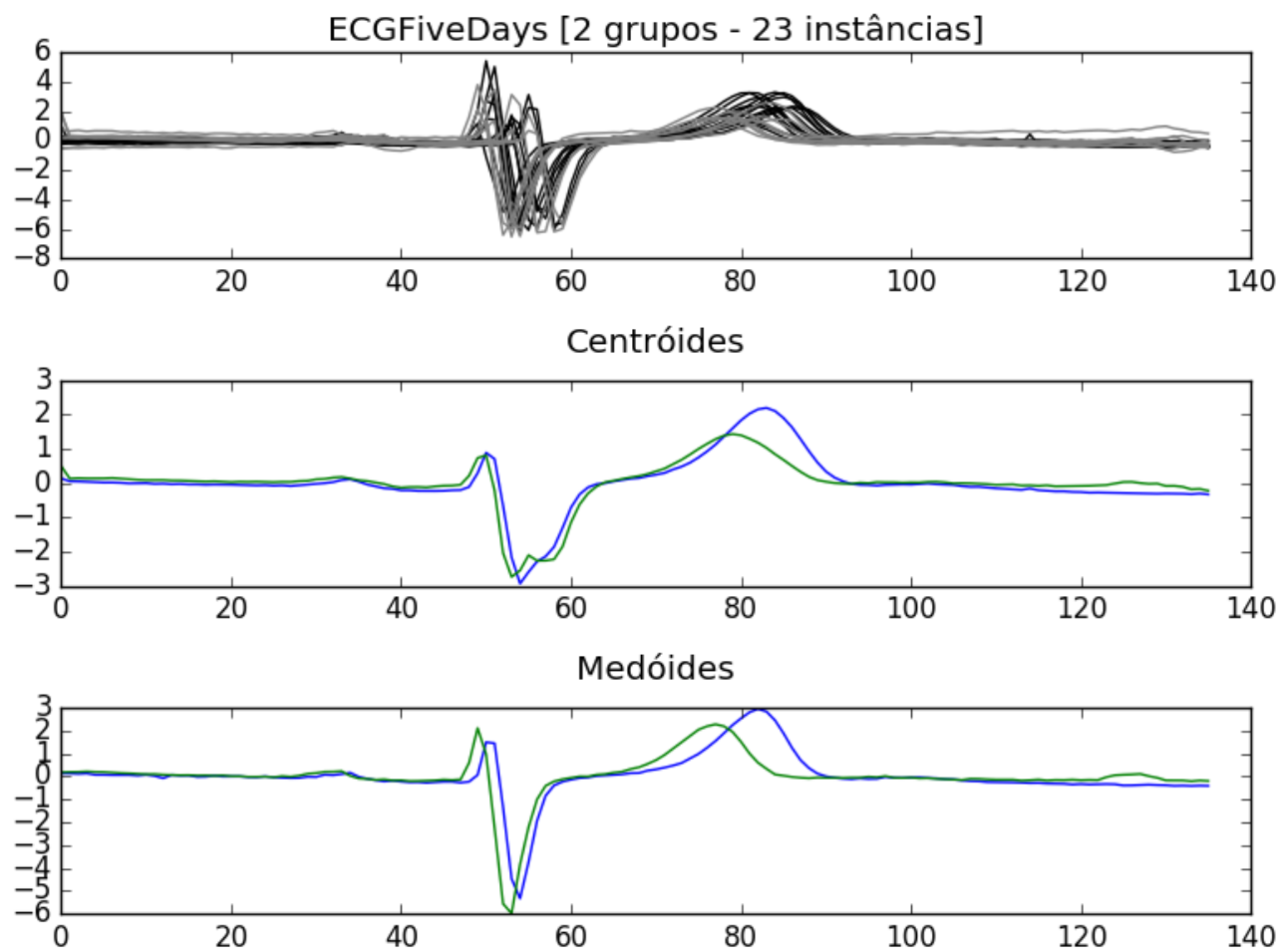


Figura A.22: Conjunto de dados 'ECGFiveDays'.

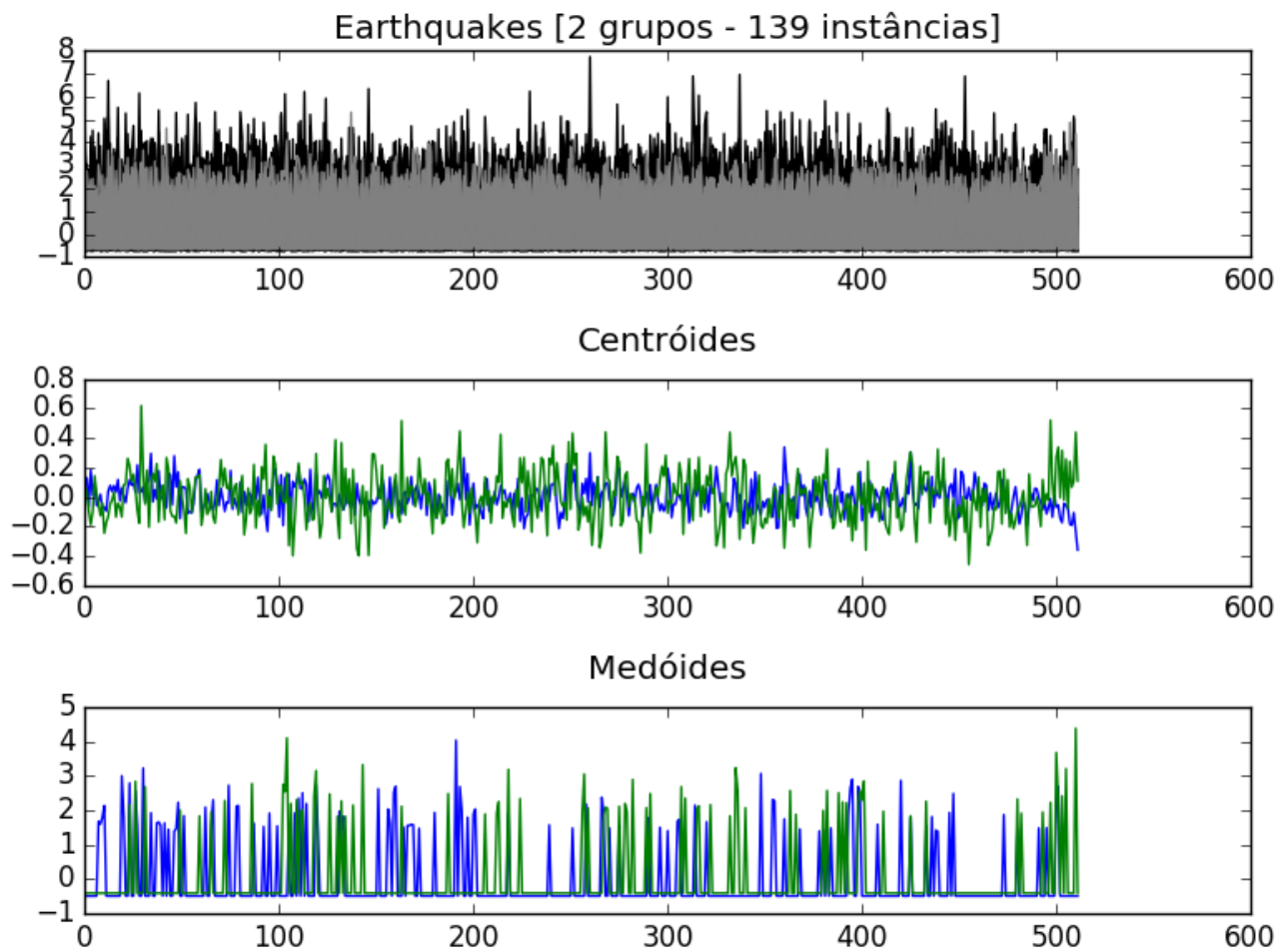


Figura A.23: Conjunto de dados 'Earthquakes'.

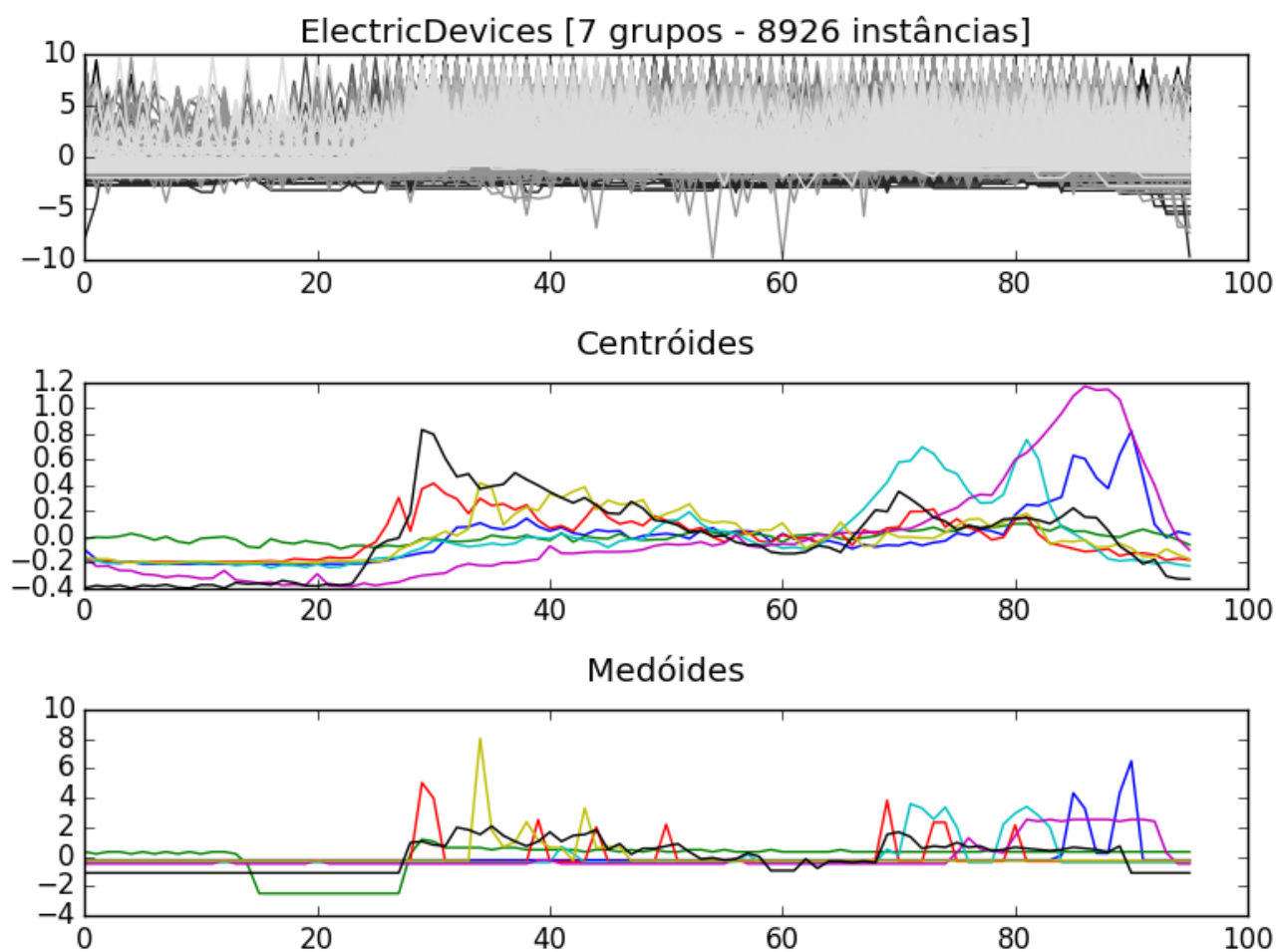


Figura A.24: Conjunto de dados 'ElectricDevices'.

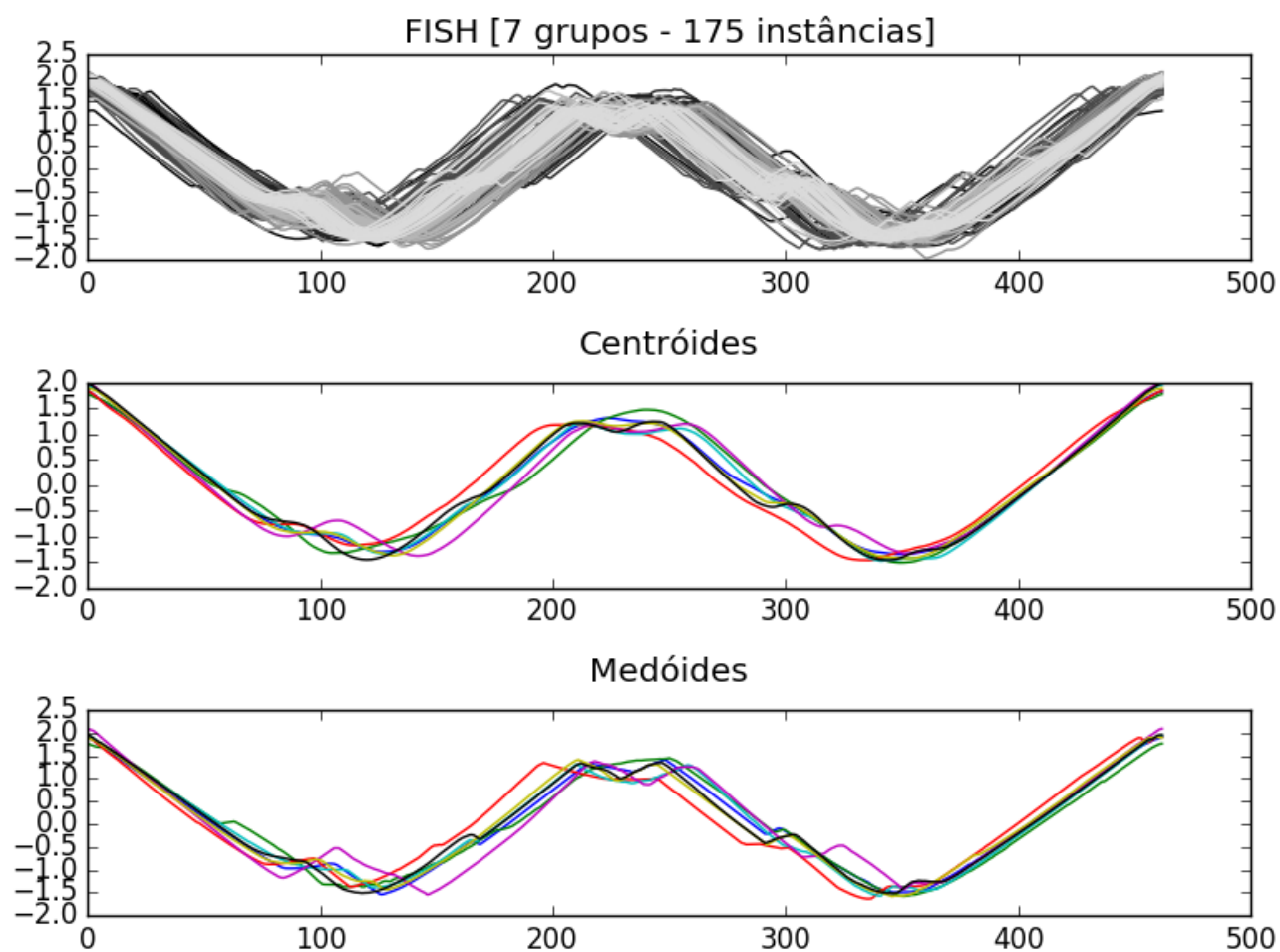


Figura A.25: Conjunto de dados 'FISH'.

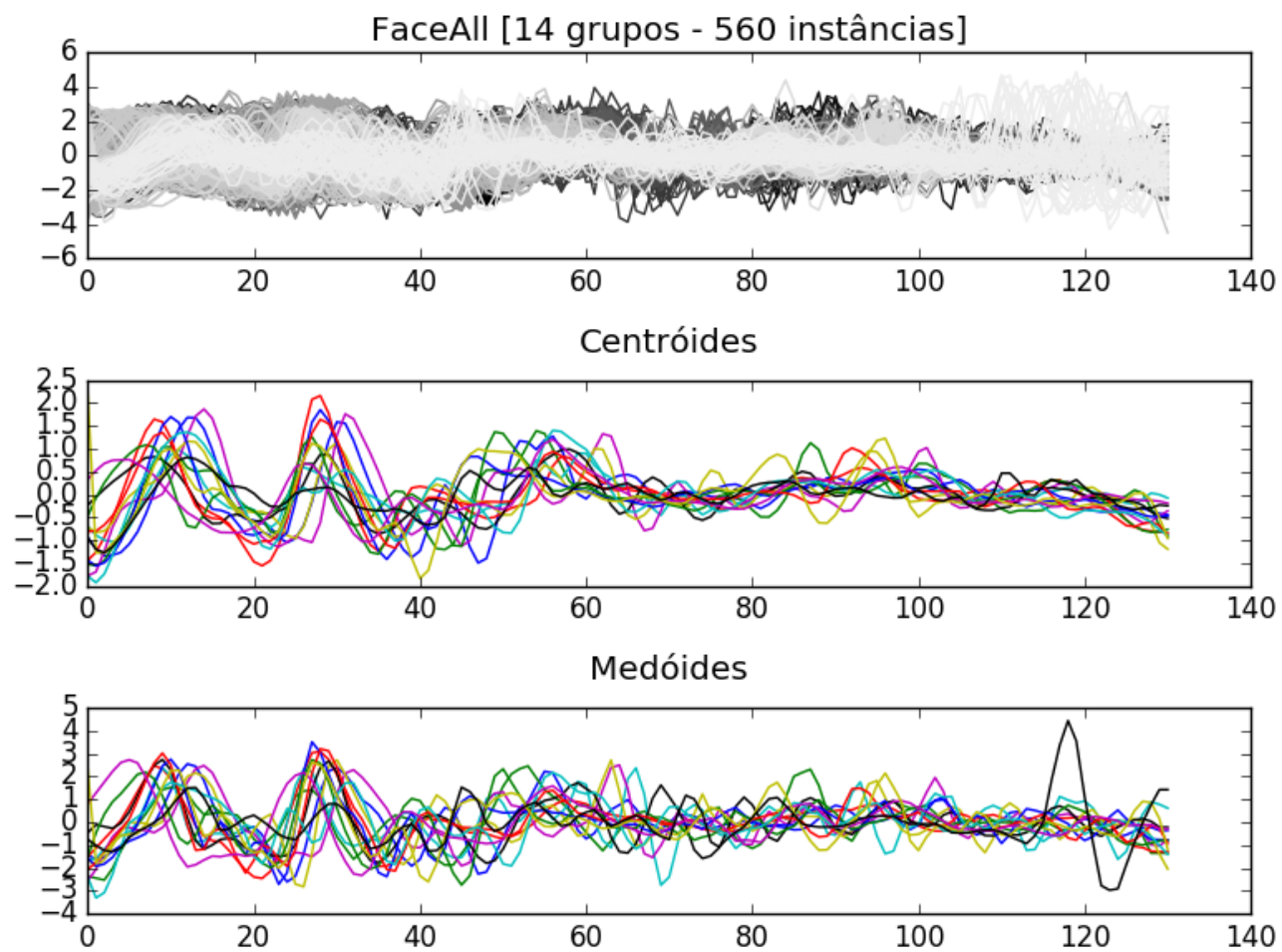


Figura A.26: Conjunto de dados 'FaceAll'.

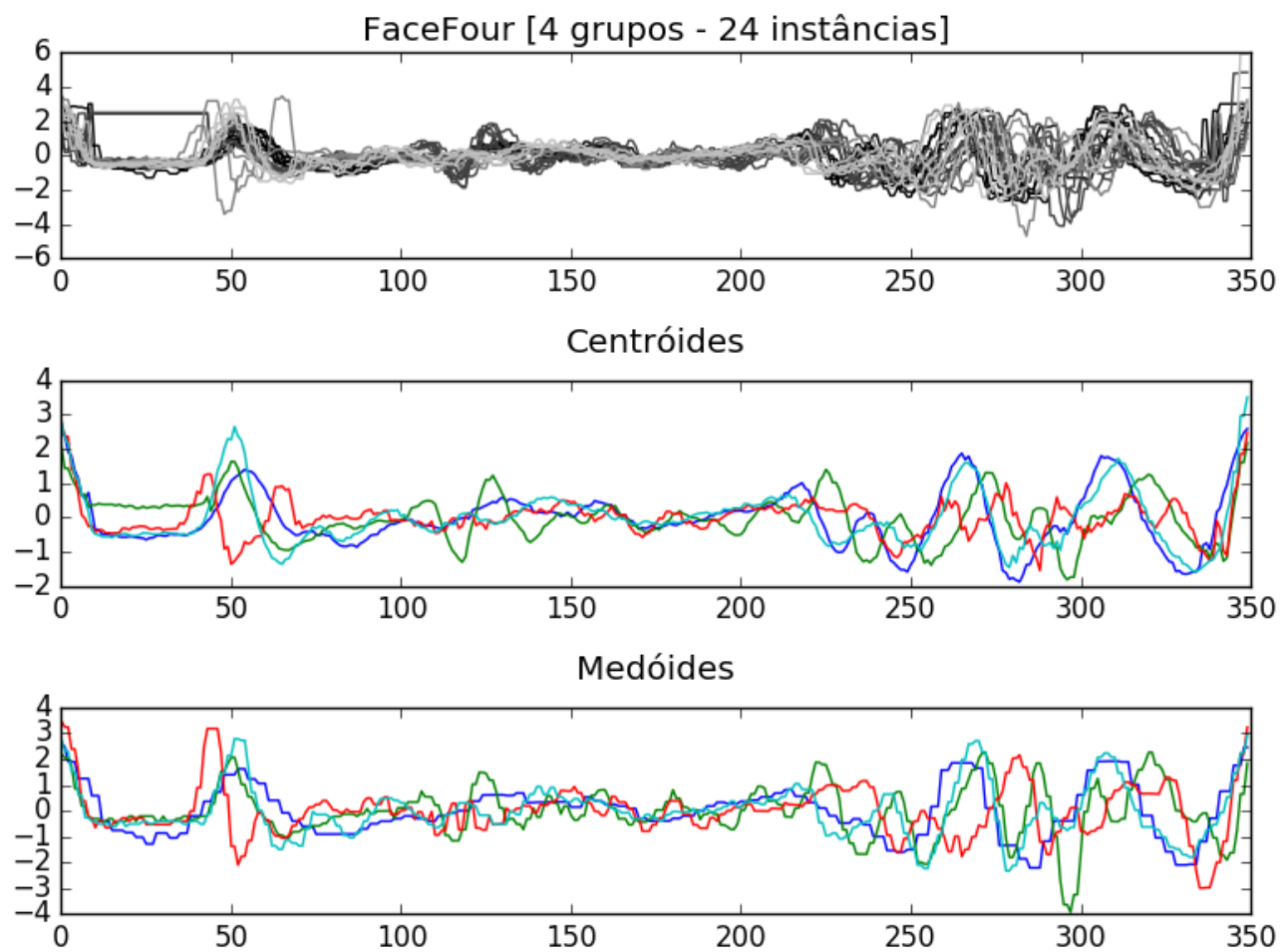


Figura A.27: Conjunto de dados 'FaceFour'.

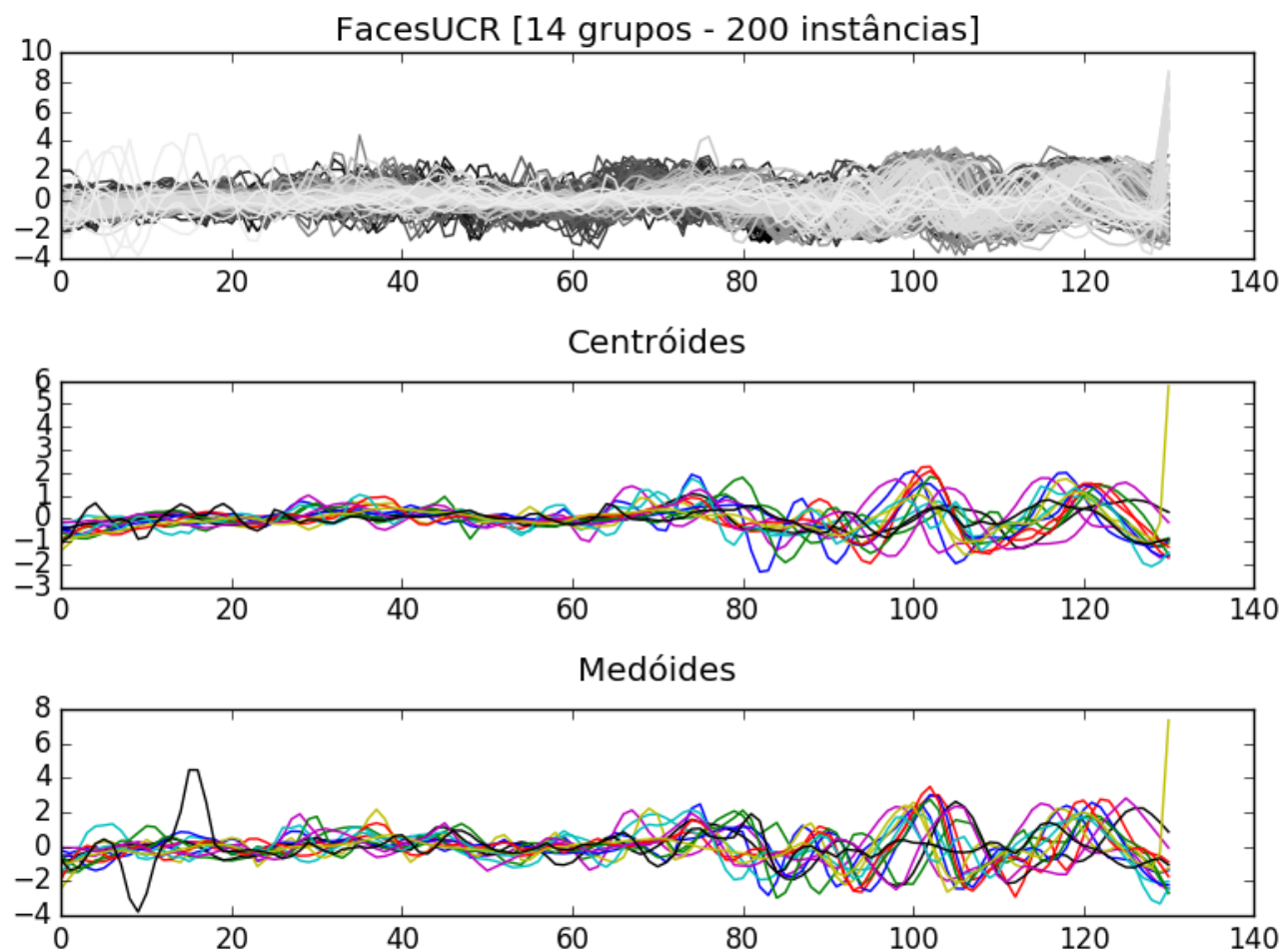


Figura A.28: Conjunto de dados 'FacesUCR'.

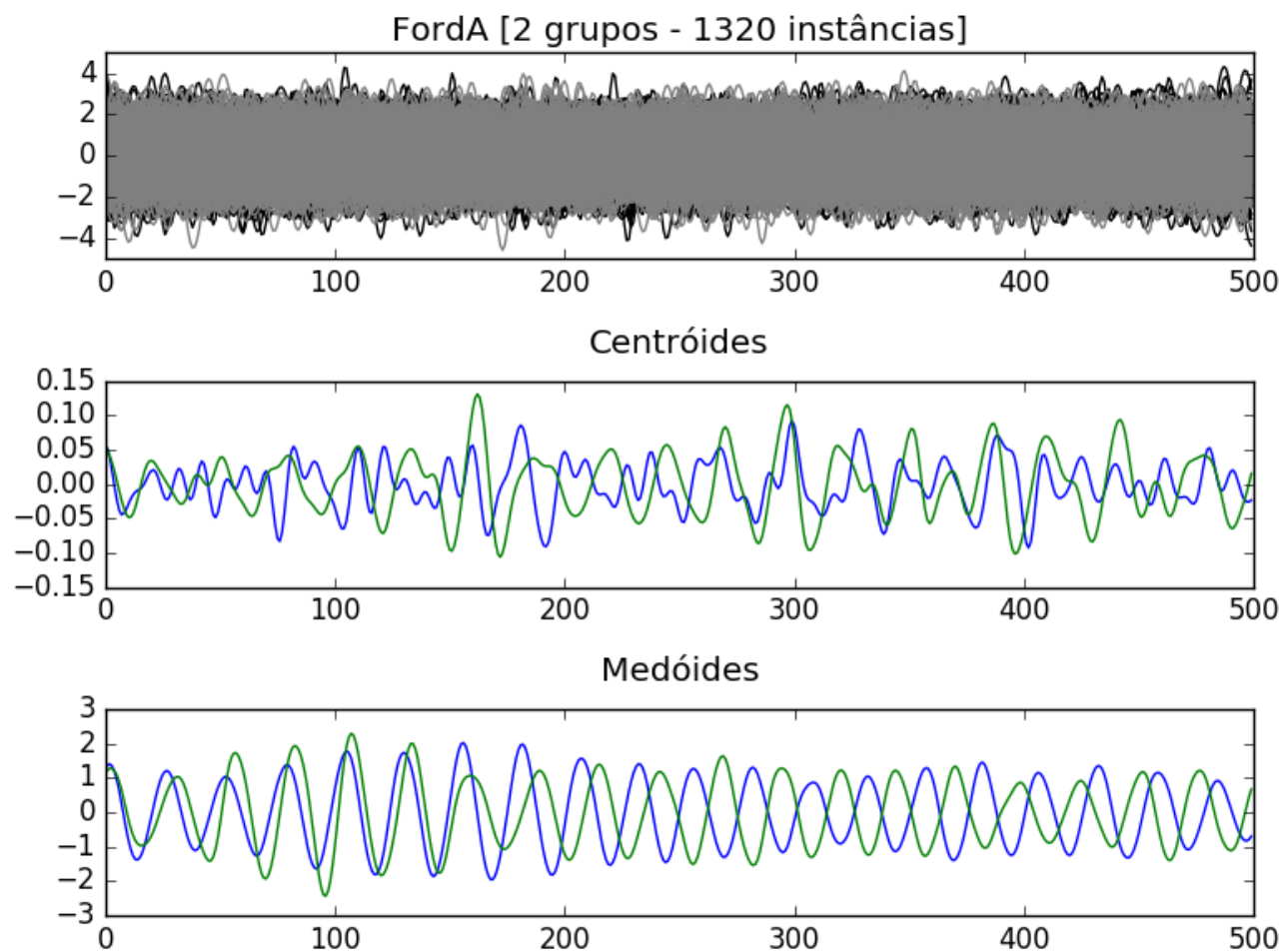


Figura A.29: Conjunto de dados 'FordA'.

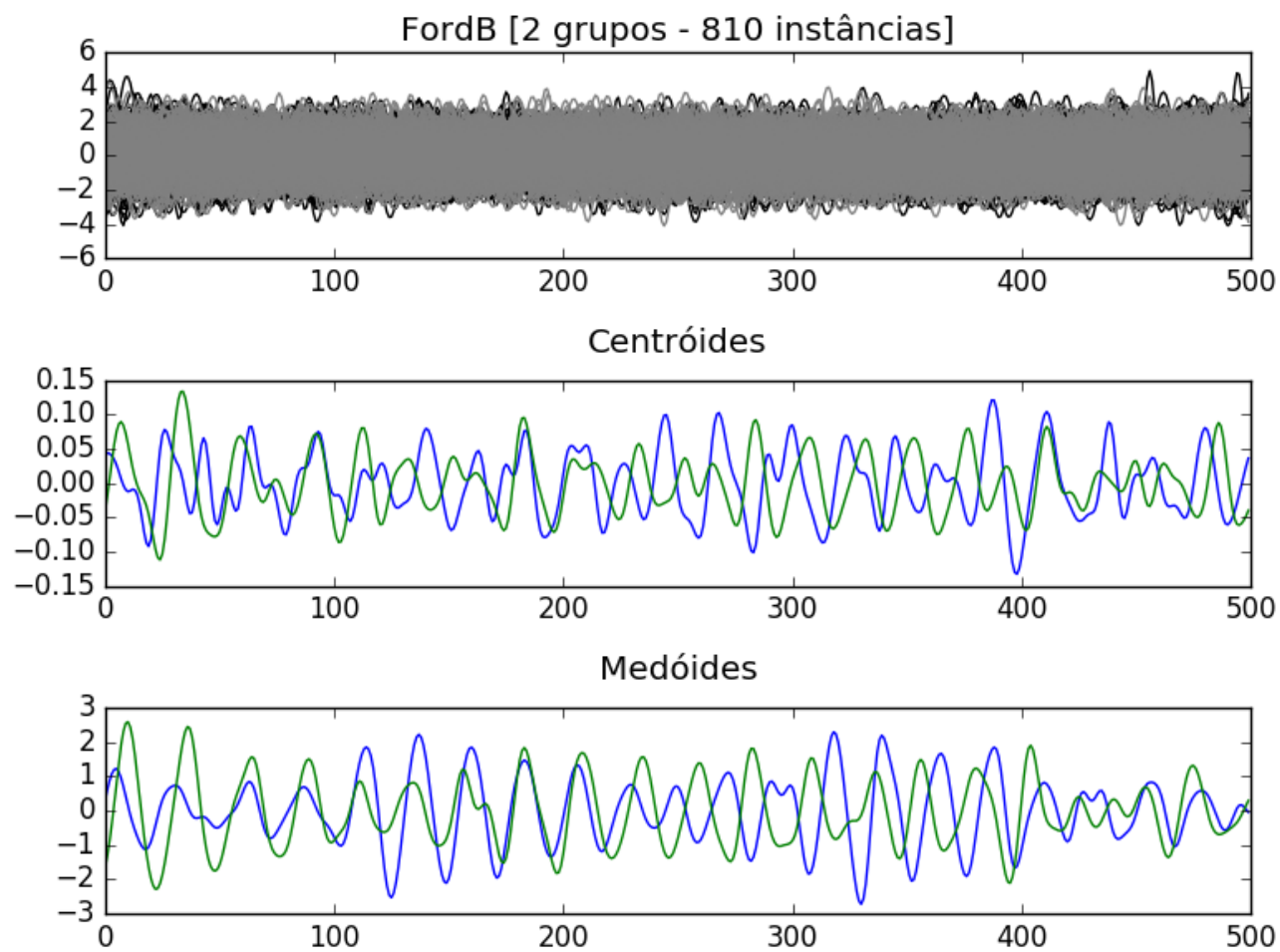


Figura A.30: Conjunto de dados 'FordB'.

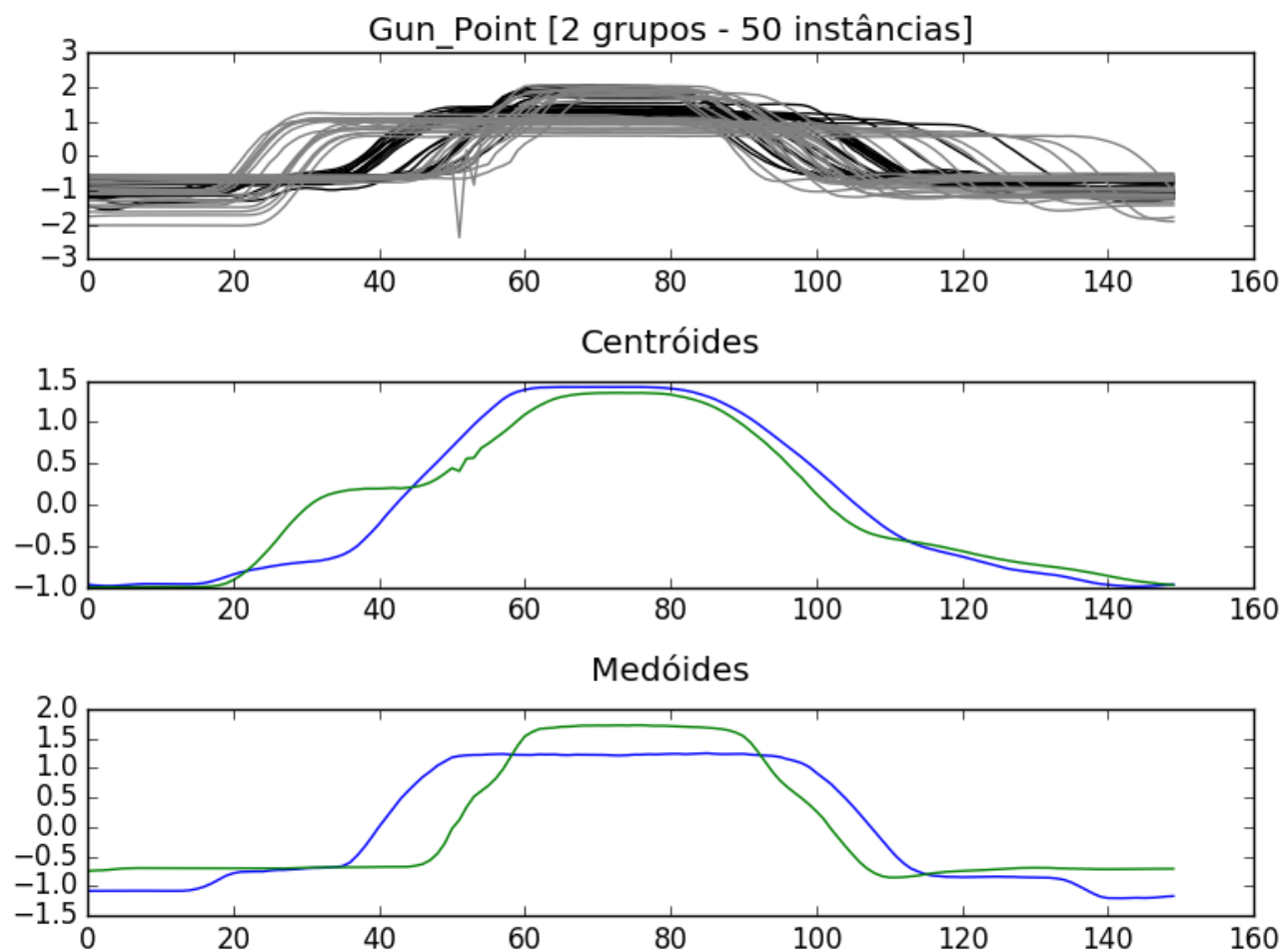


Figura A.31: Conjunto de dados 'Gun_Point'.

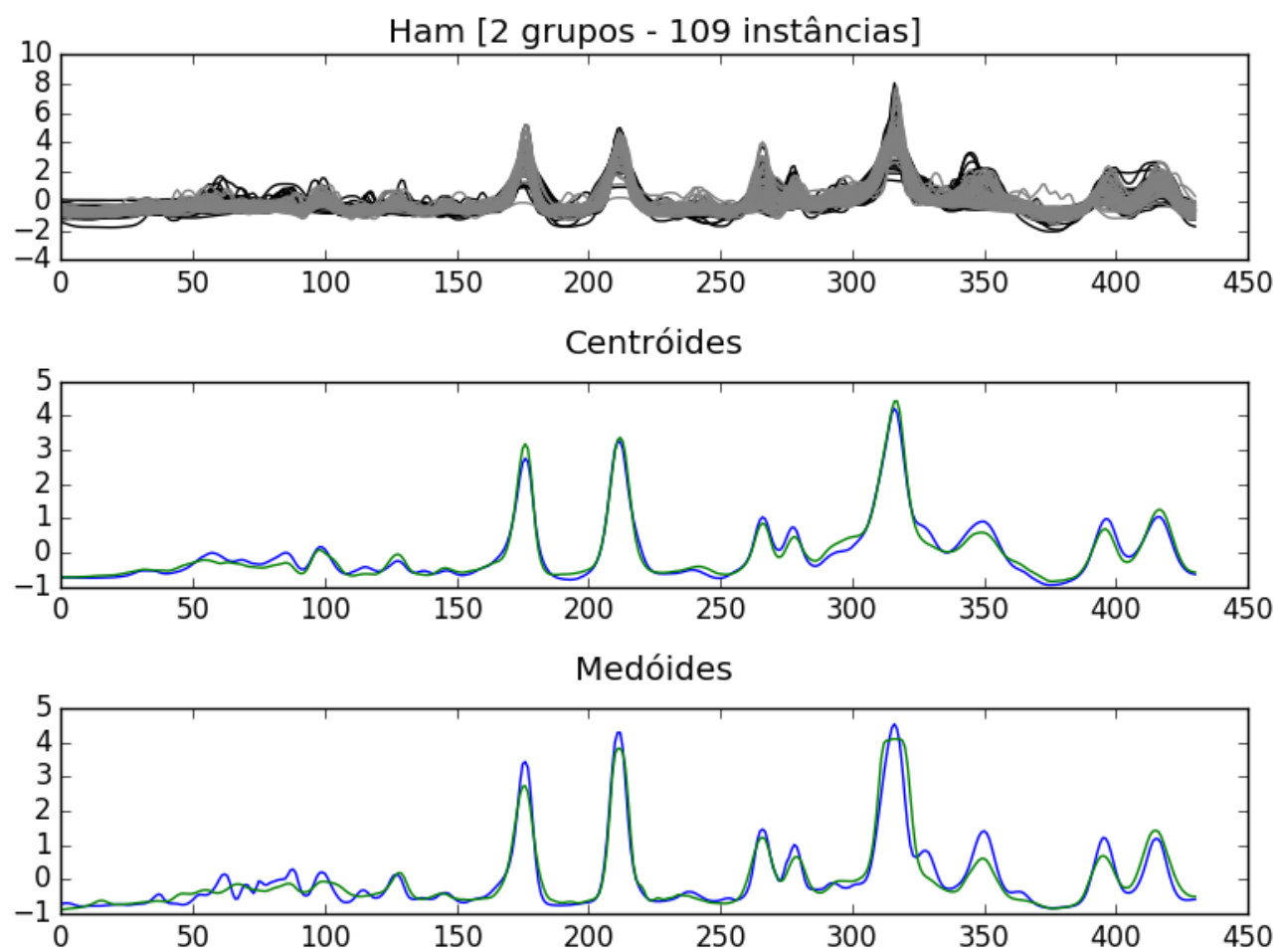


Figura A.32: Conjunto de dados 'Ham'.

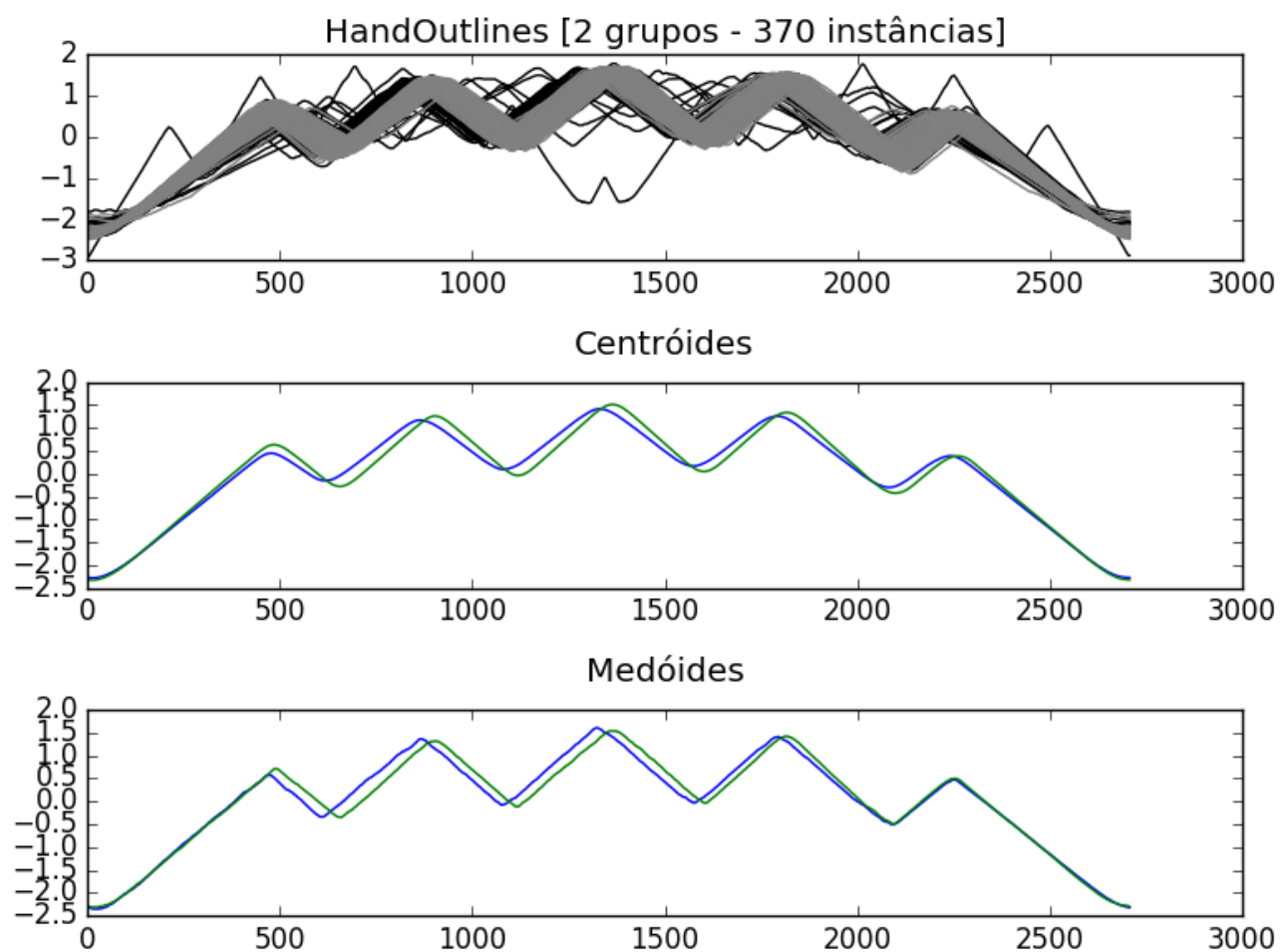


Figura A.33: Conjunto de dados 'HandOutlines'.

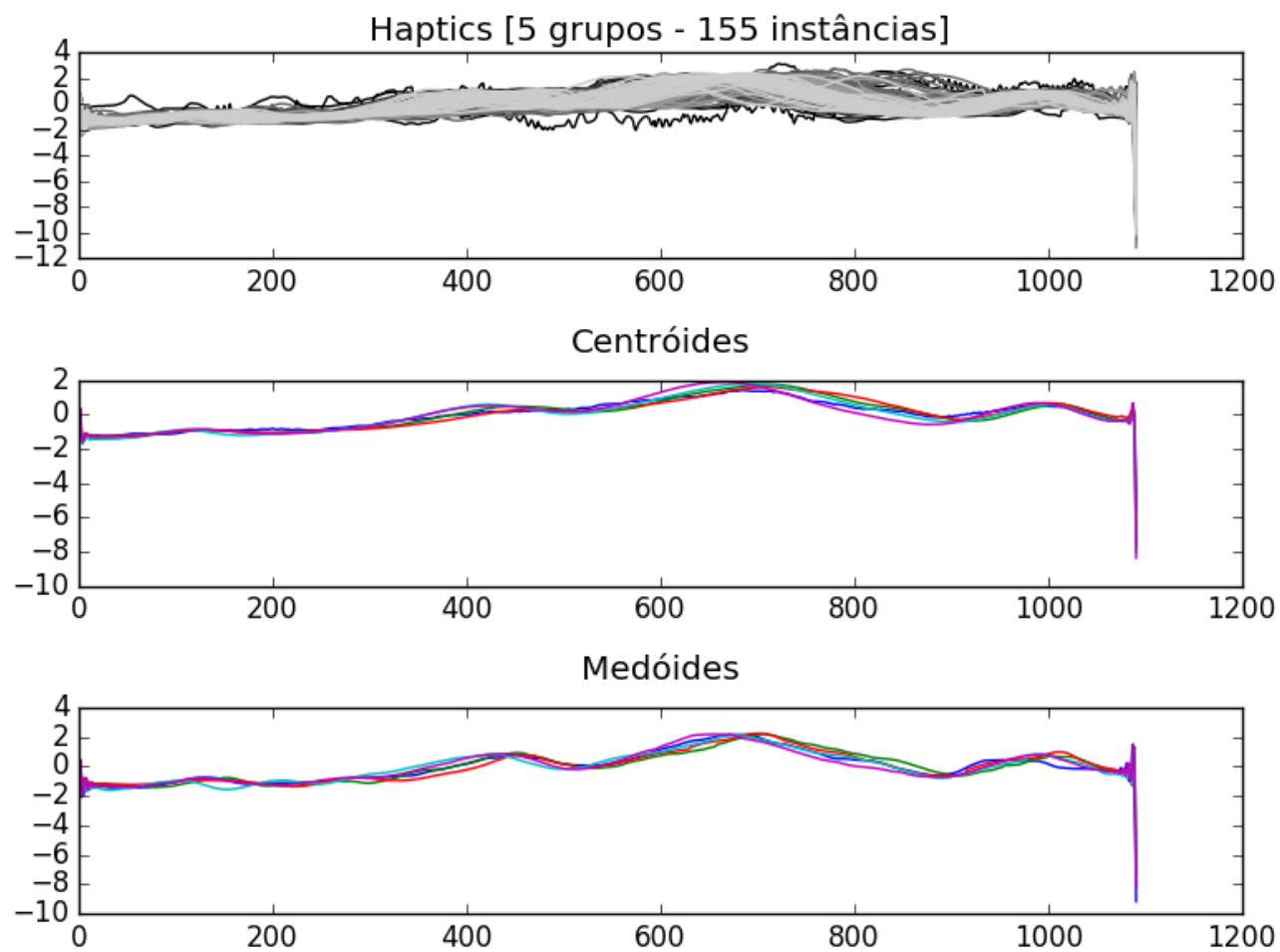


Figura A.34: Conjunto de dados 'Haptics'.

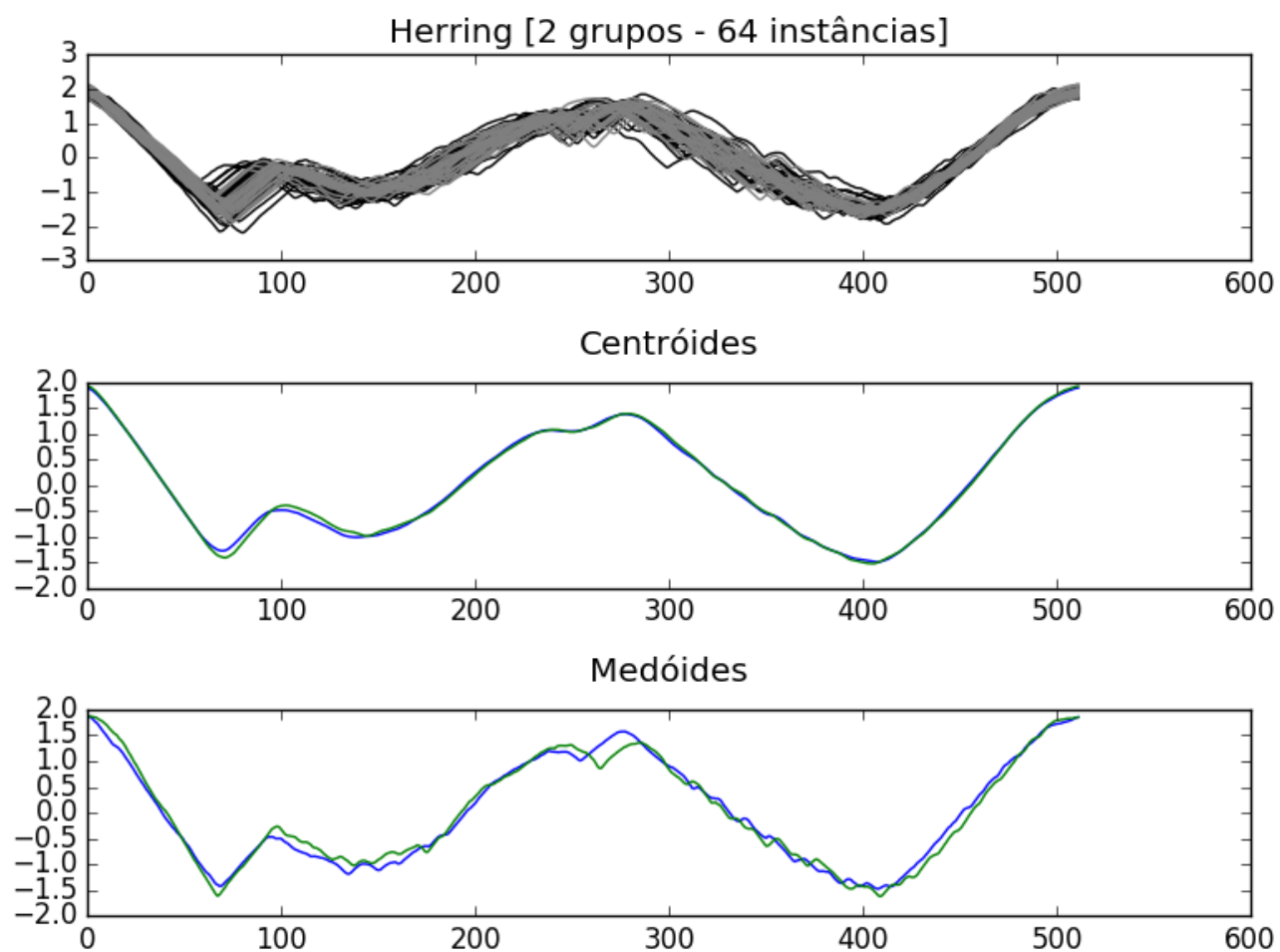


Figura A.35: Conjunto de dados 'Herring'.

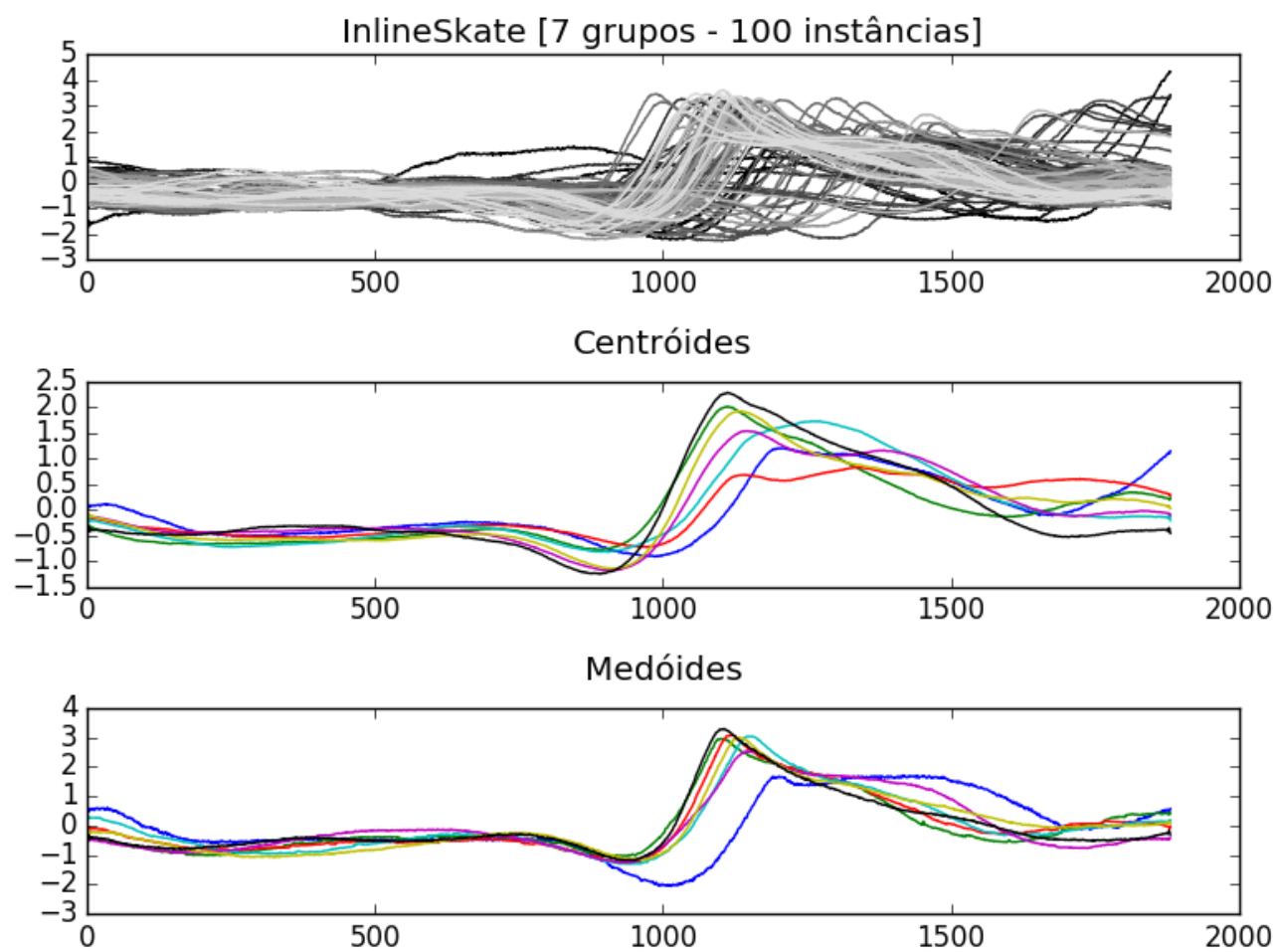


Figura A.36: Conjunto de dados 'InlineSkate'.

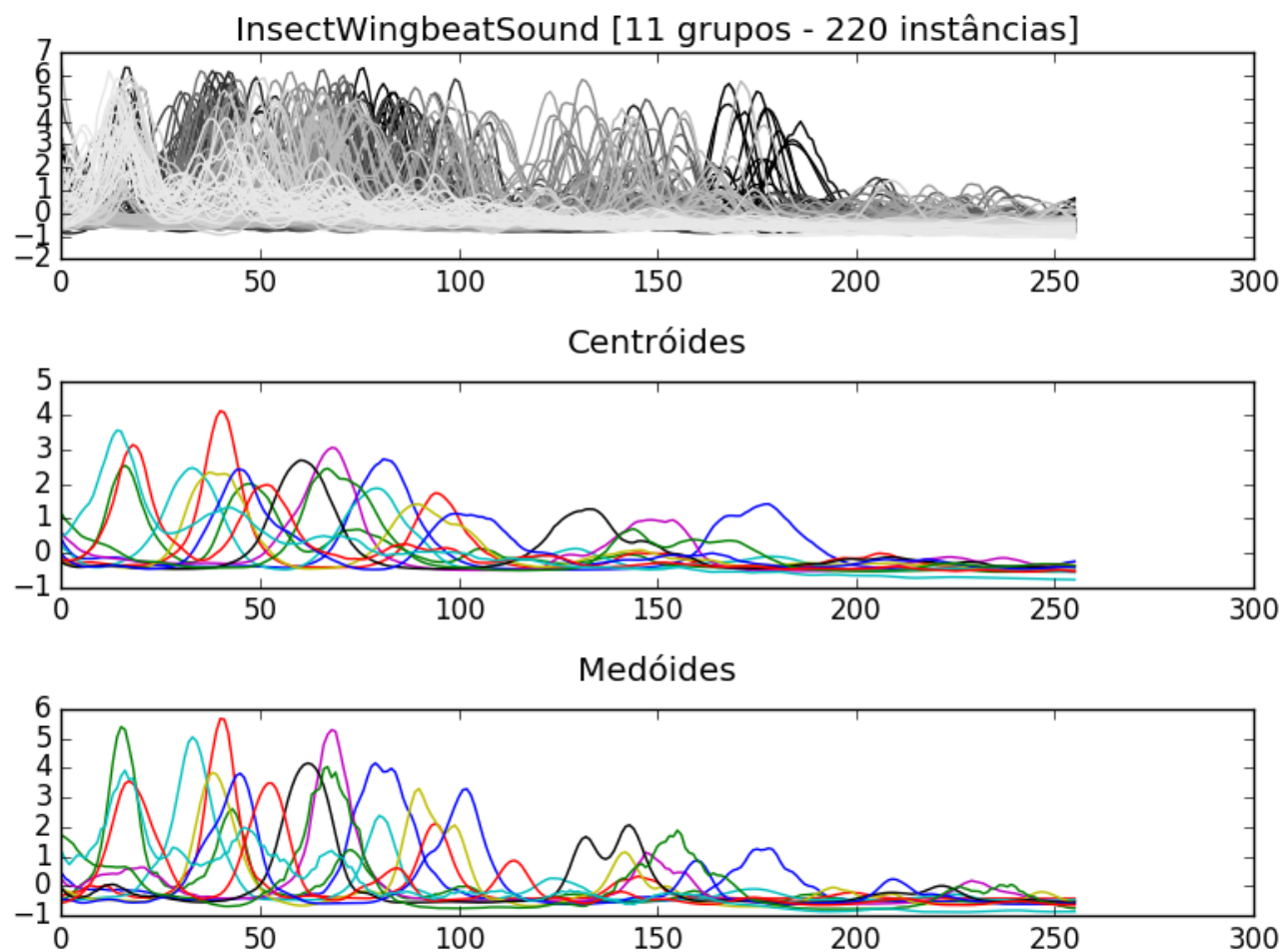


Figura A.37: Conjunto de dados 'InsectWingbeatSound'.

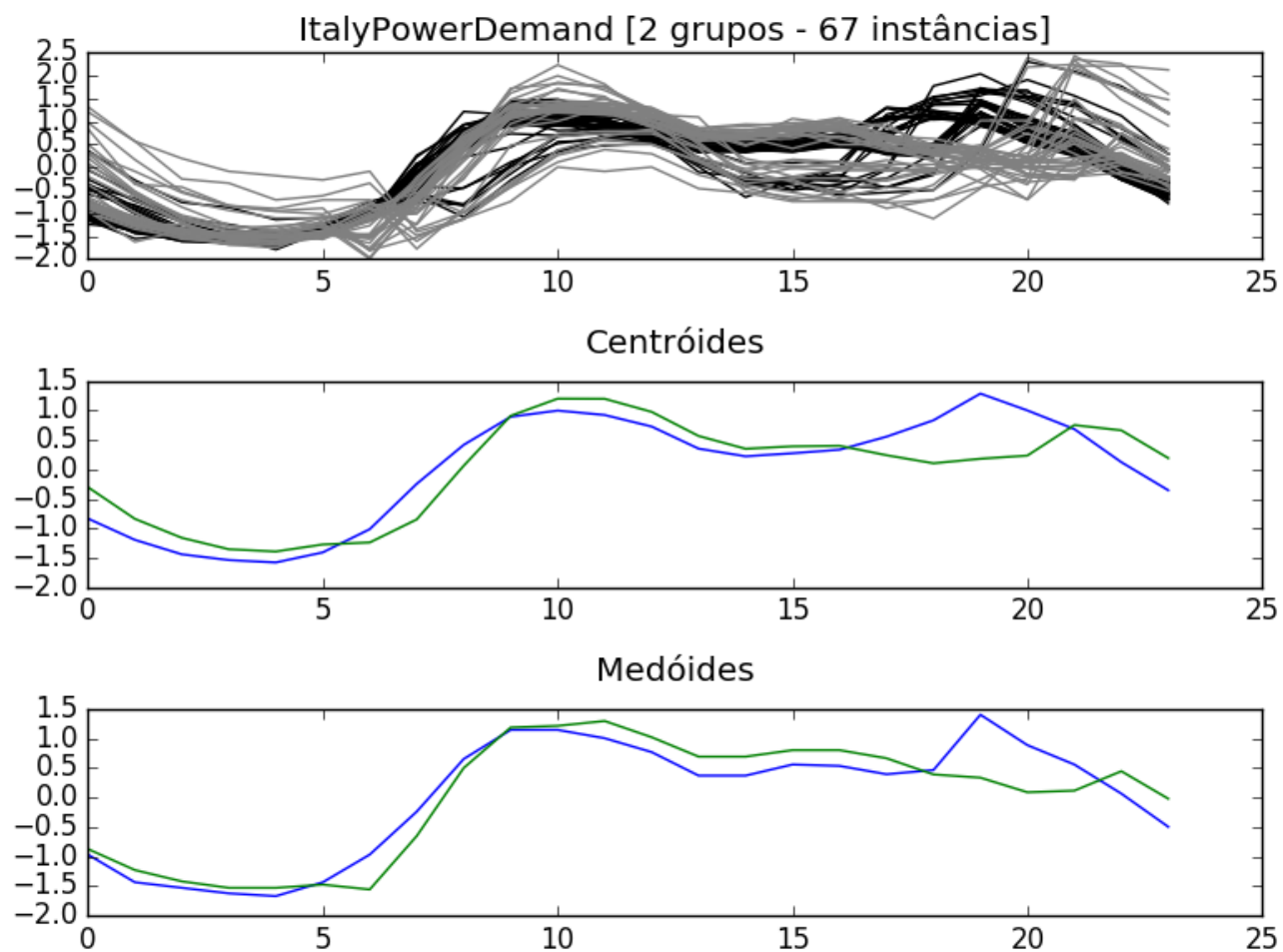


Figura A.38: Conjunto de dados 'ItalyPowerDemand'.

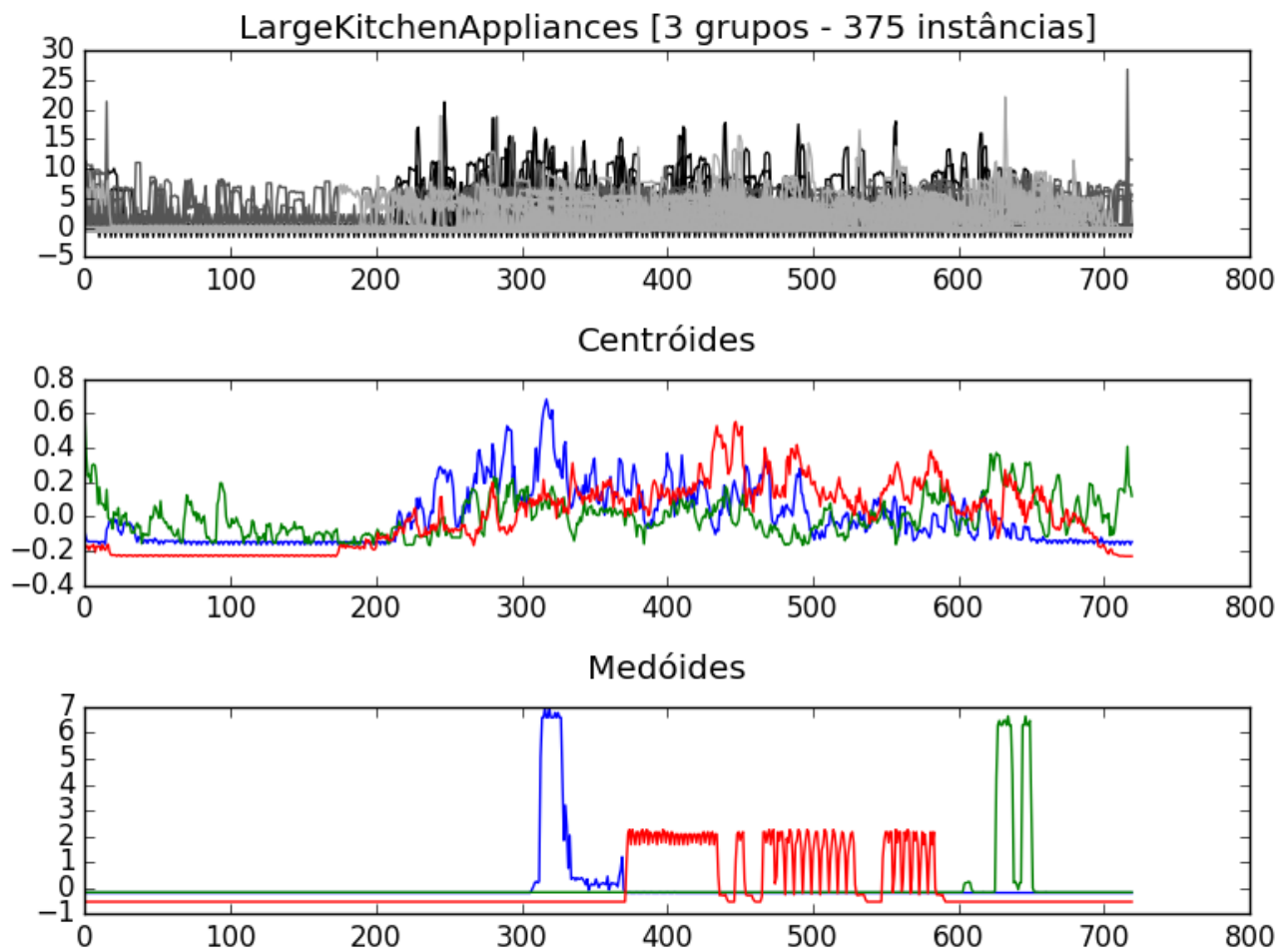


Figura A.39: Conjunto de dados 'LargeKitchenAppliances'.

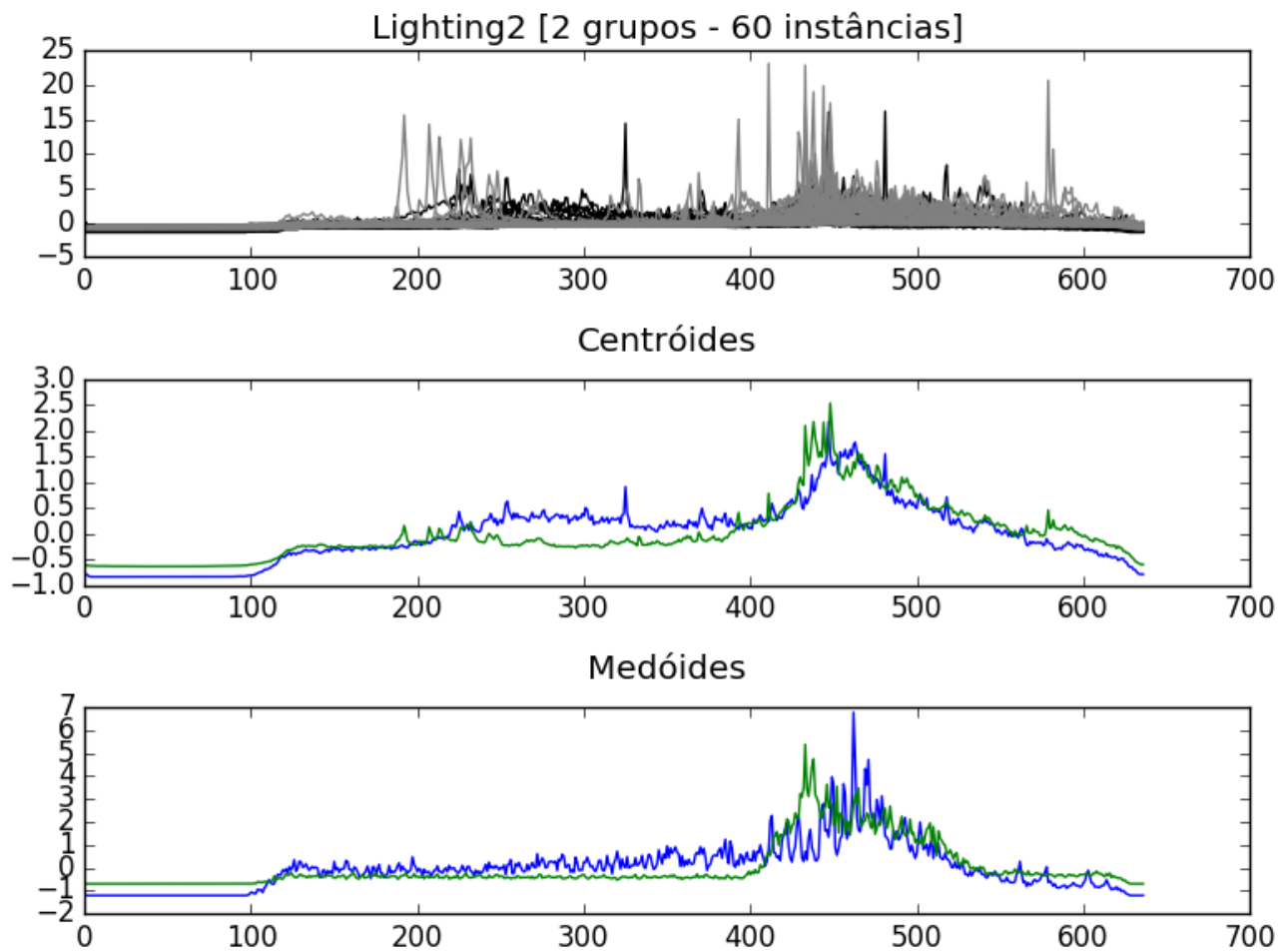


Figura A.40: Conjunto de dados 'Lighting2'.

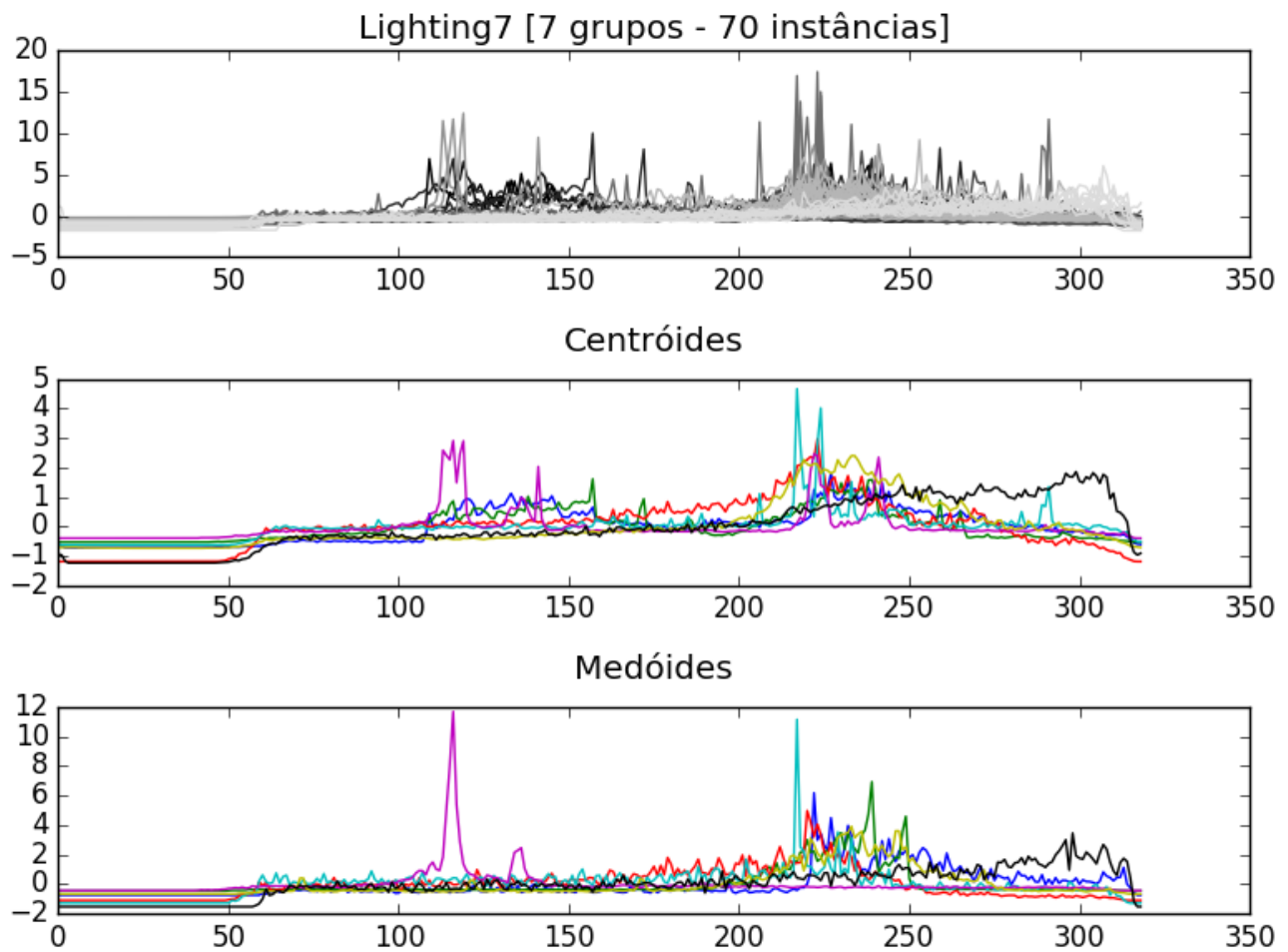


Figura A.41: Conjunto de dados 'Lighting7'.

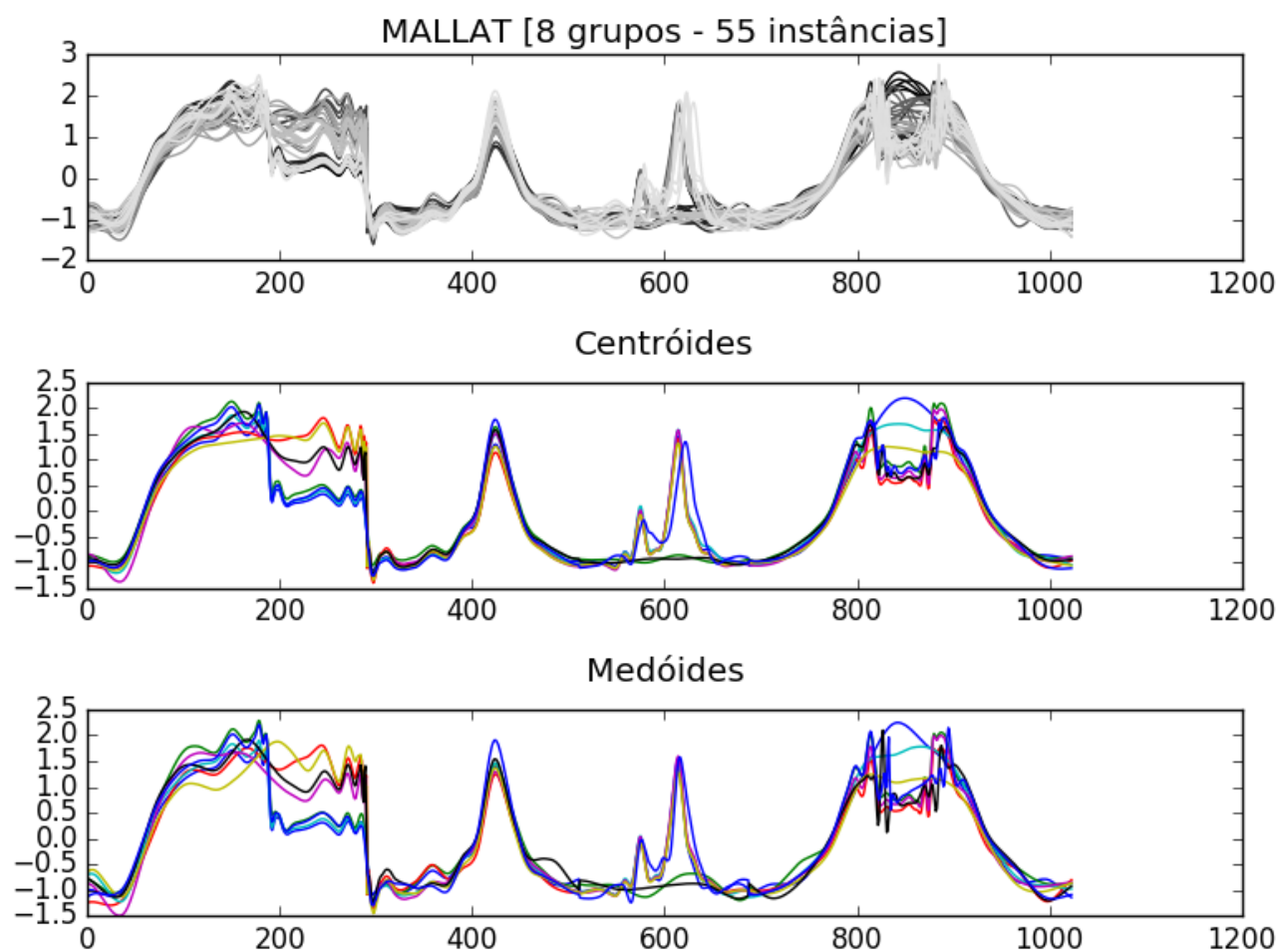


Figura A.42: Conjunto de dados 'MALLAT'.

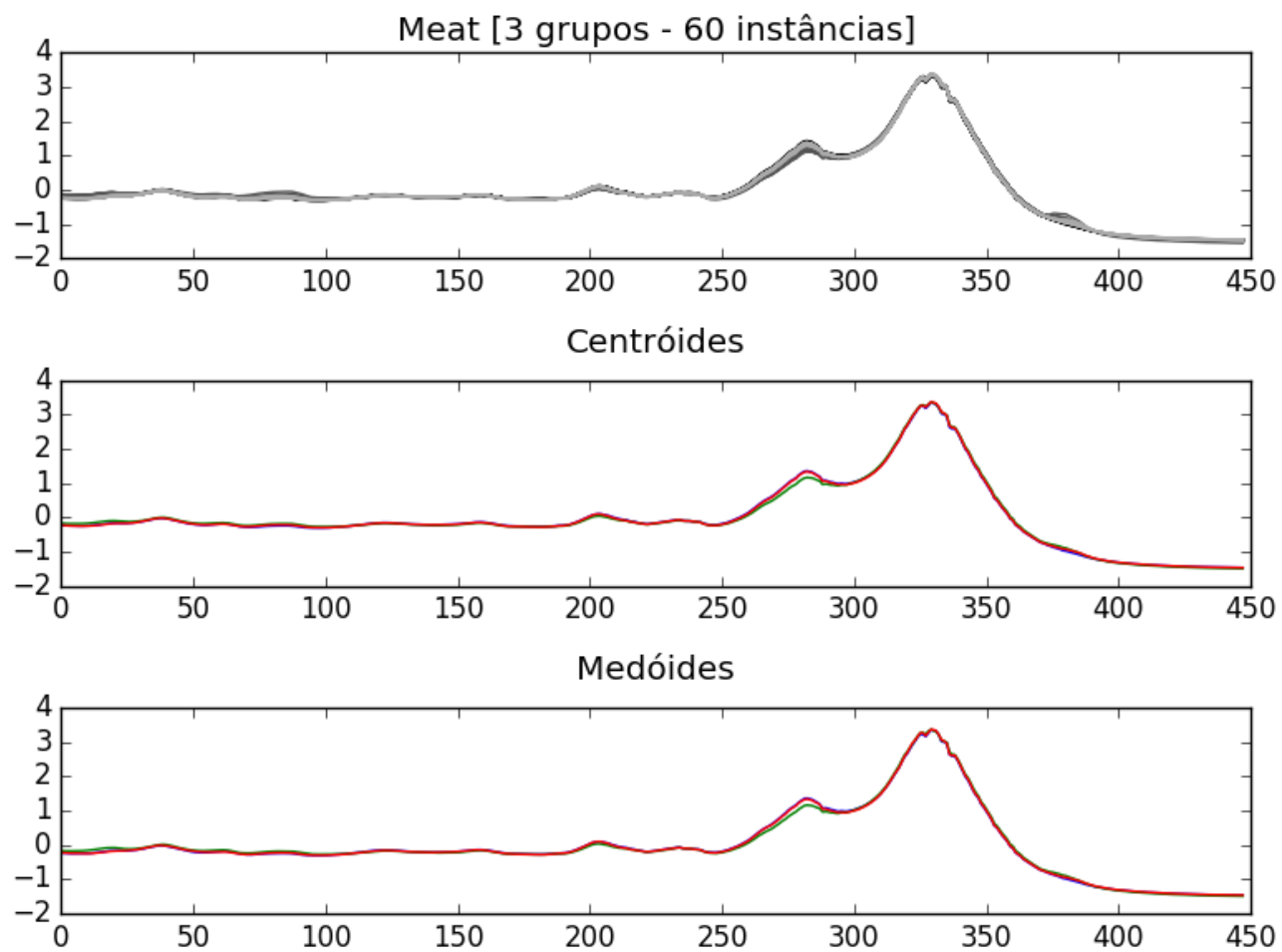


Figura A.43: Conjunto de dados 'Meat'.

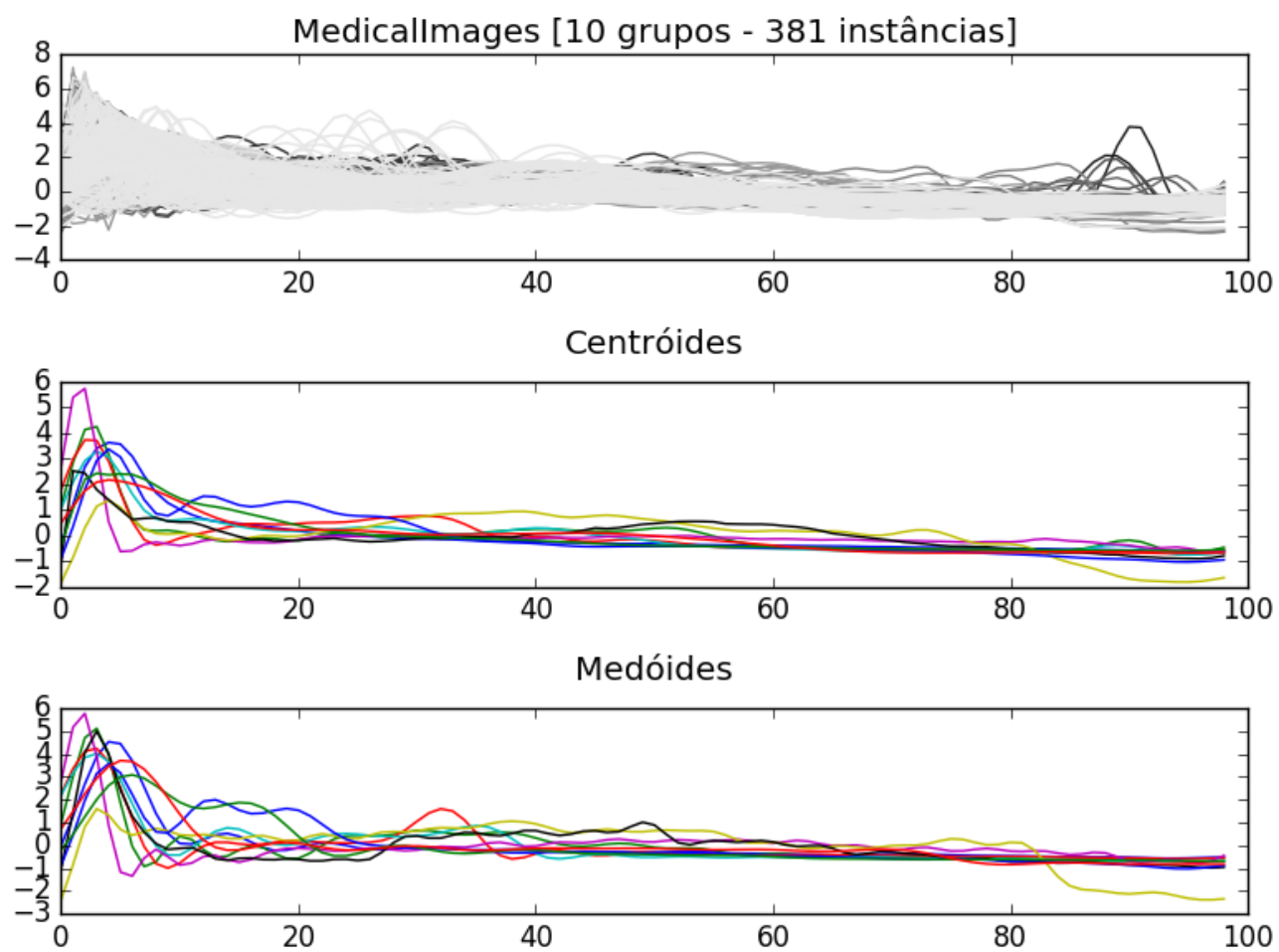


Figura A.44: Conjunto de dados 'MedicalImages'.

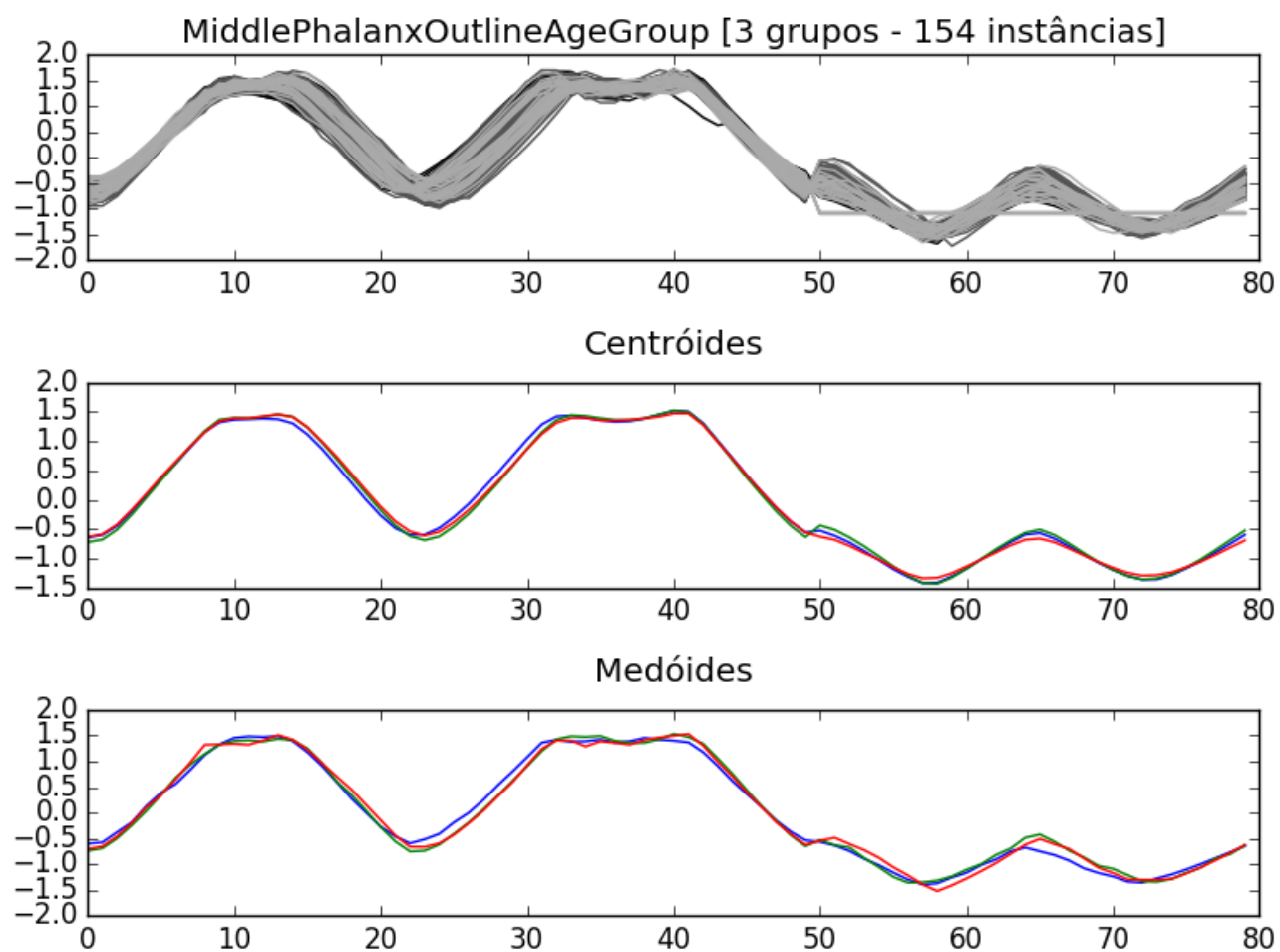


Figura A.45: Conjunto de dados 'MiddlePhalanxOutlineAgeGroup'.

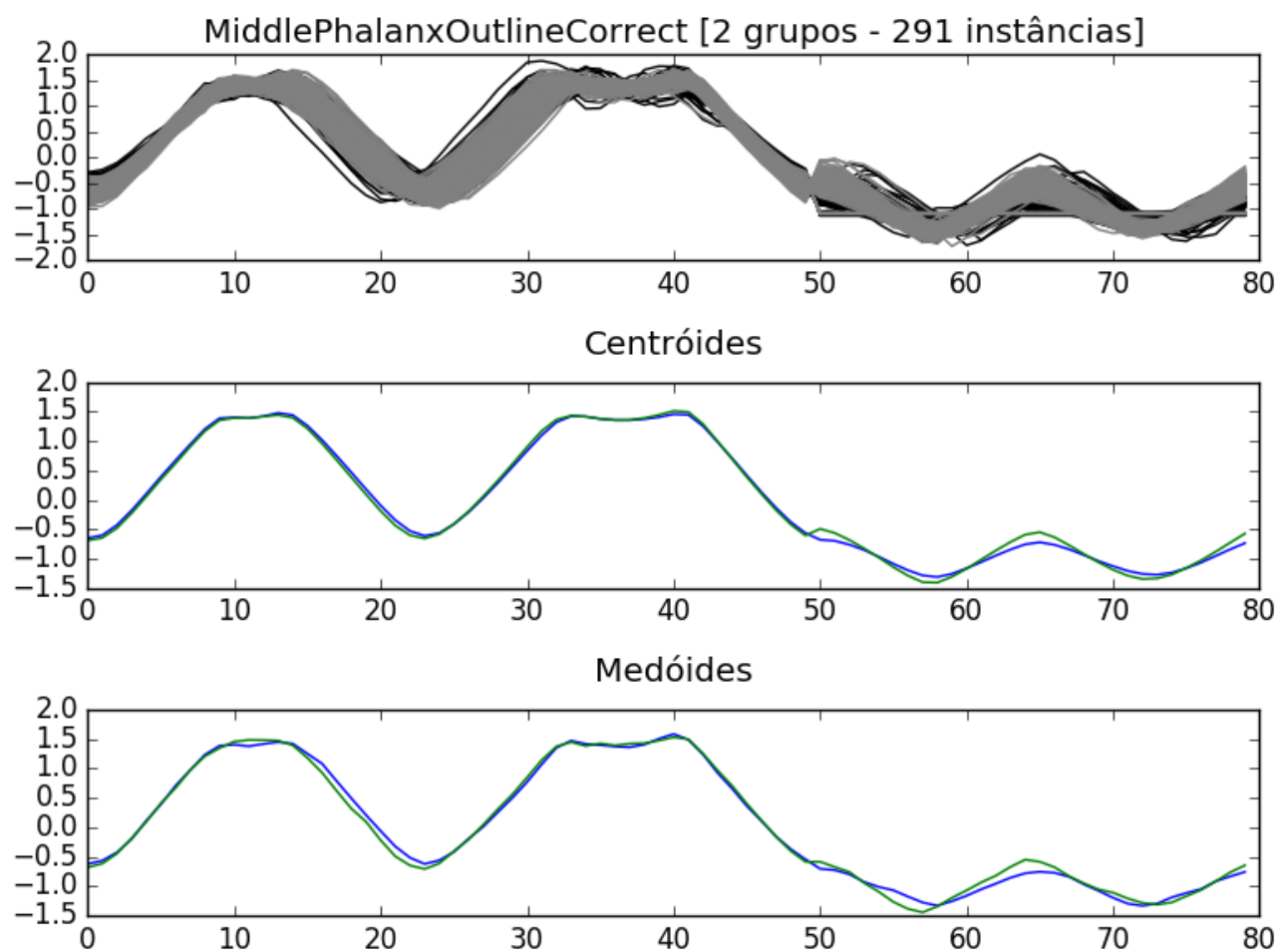


Figura A.46: Conjunto de dados 'MiddlePhalanxOutlineCorrect'.

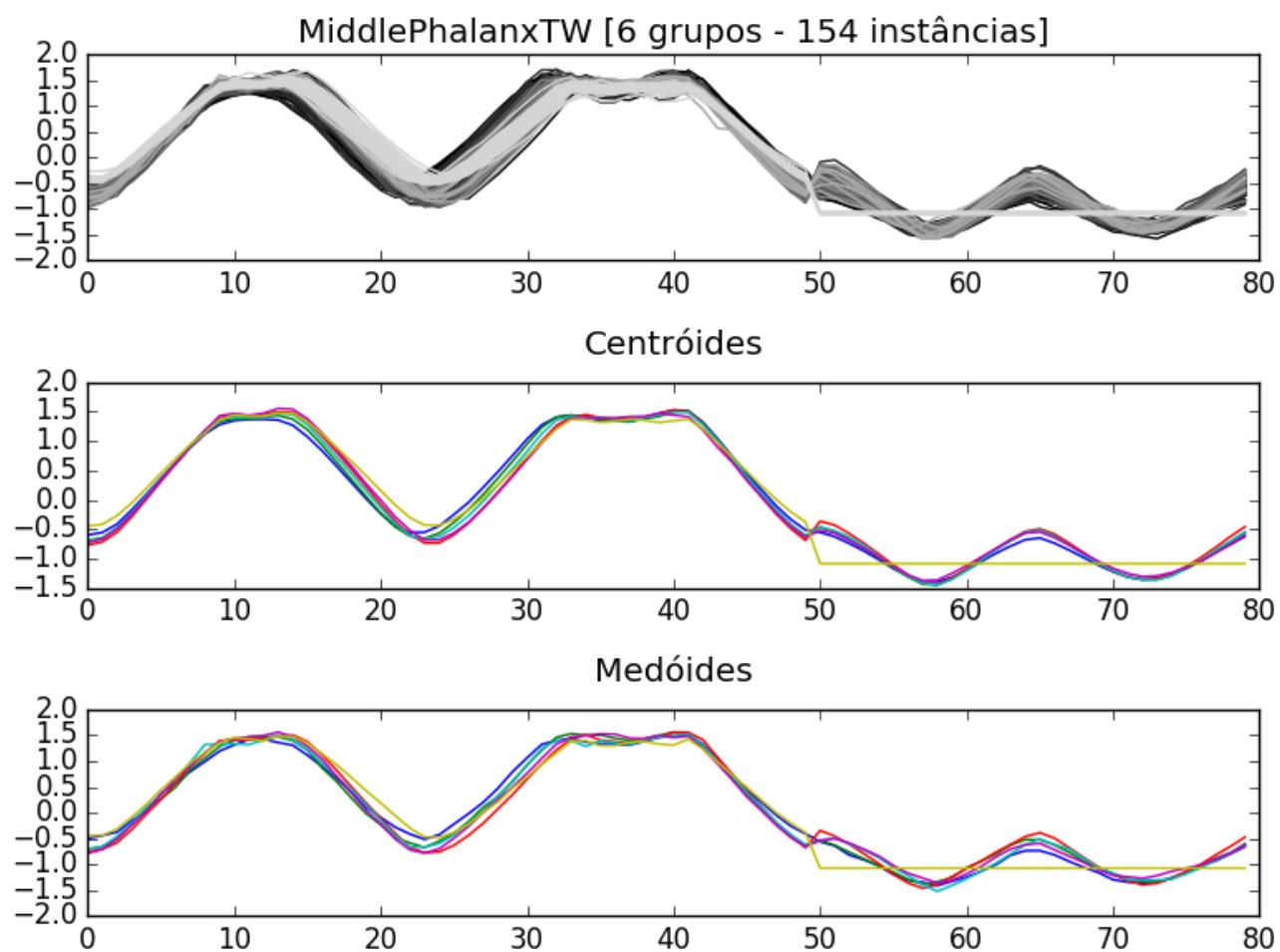


Figura A.47: Conjunto de dados 'MiddlePhalanxTW'.

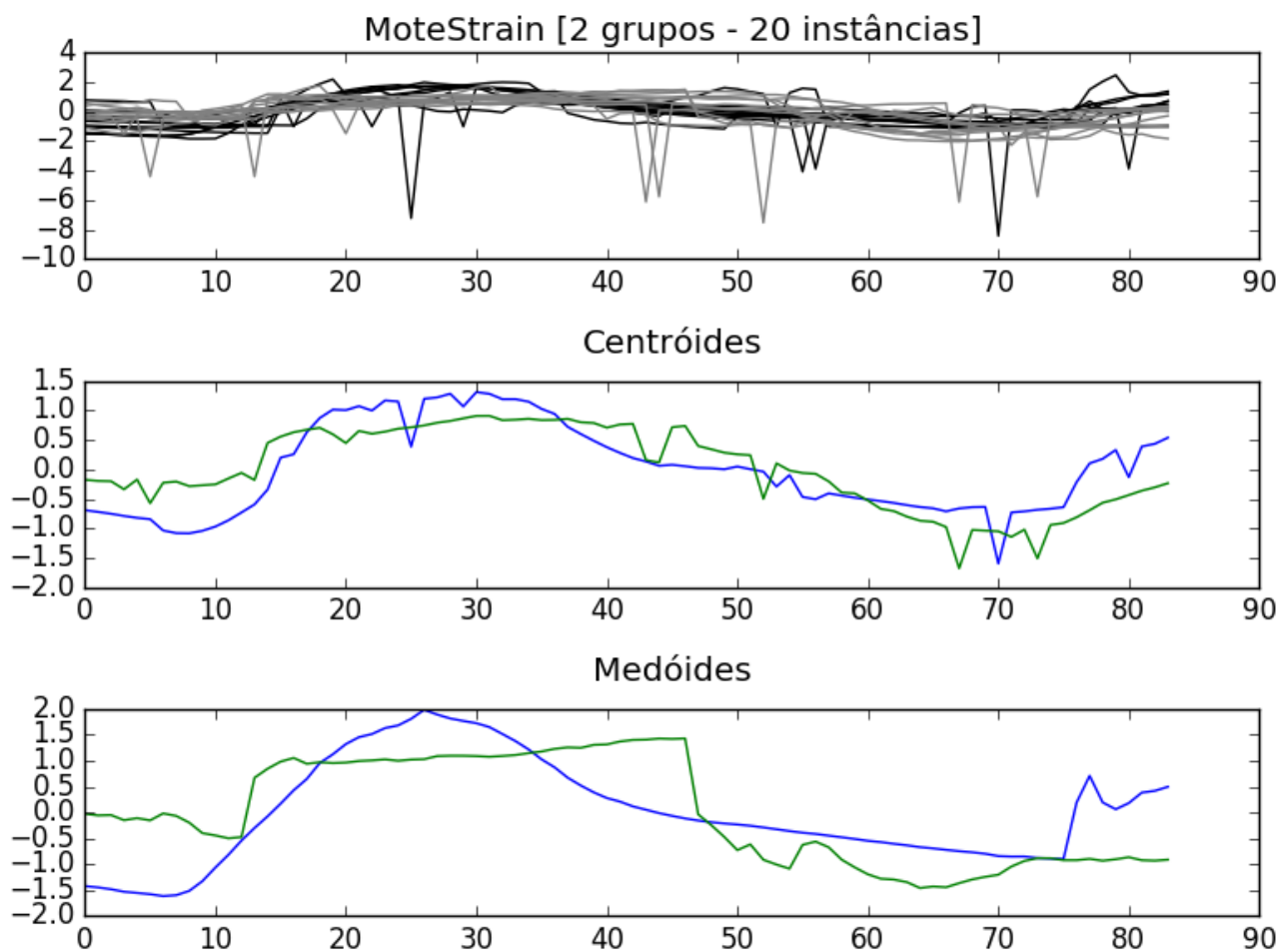


Figura A.48: Conjunto de dados 'MoteStrain'.

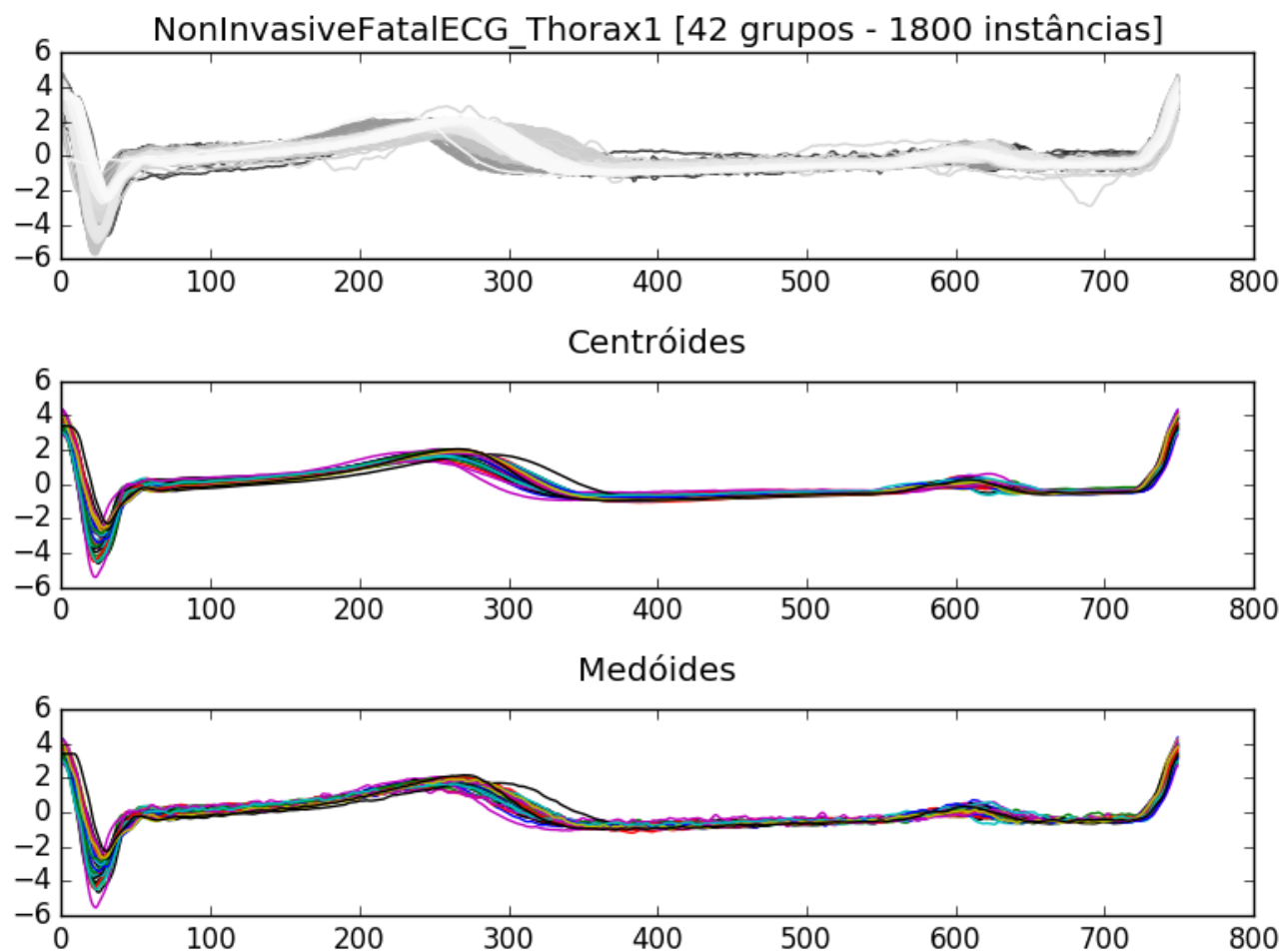


Figura A.49: Conjunto de dados 'NonInvasiveFatalECG_Thorax1'.

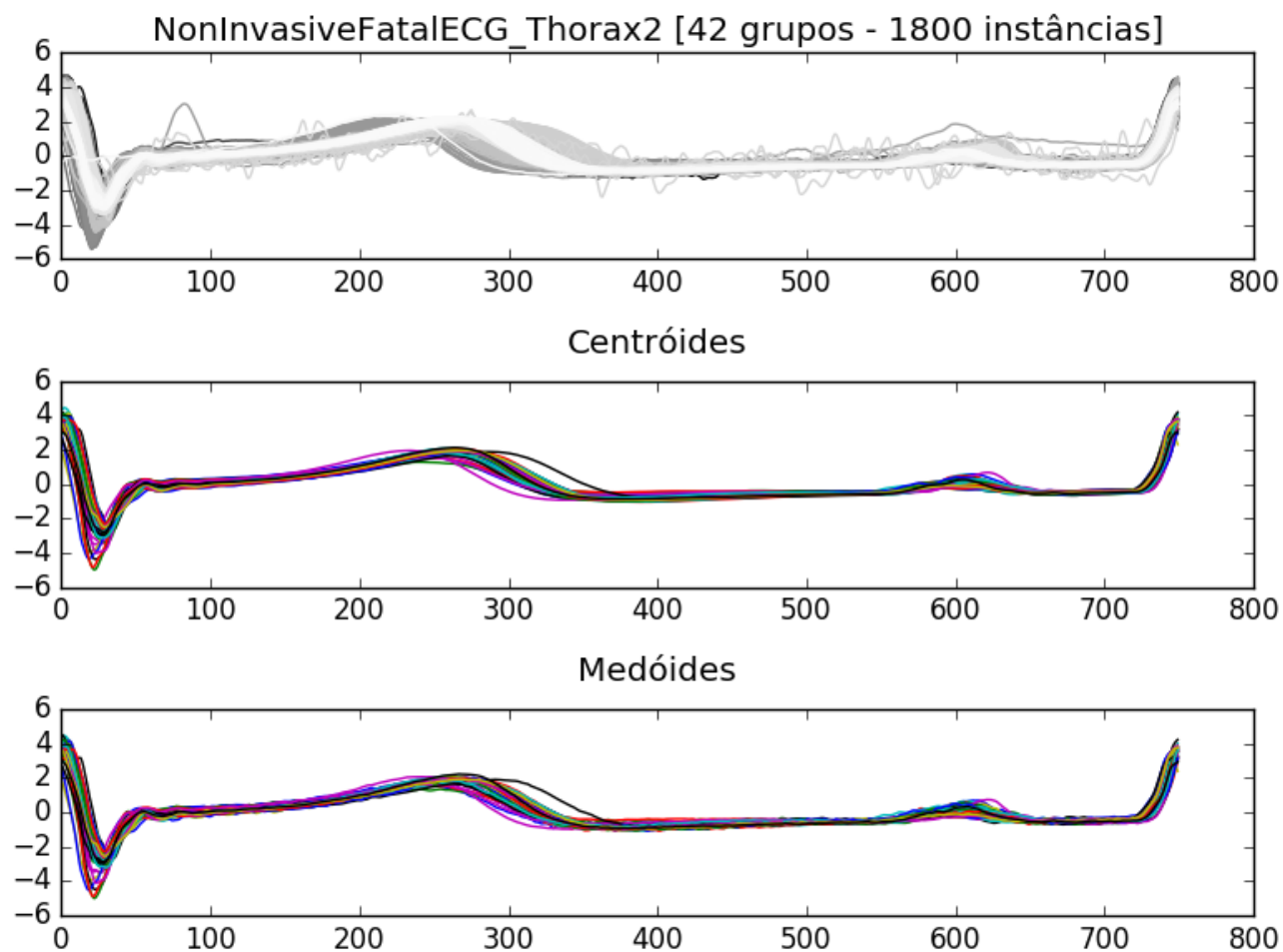


Figura A.50: Conjunto de dados 'NonInvasiveFataleCG_Thorax2'.

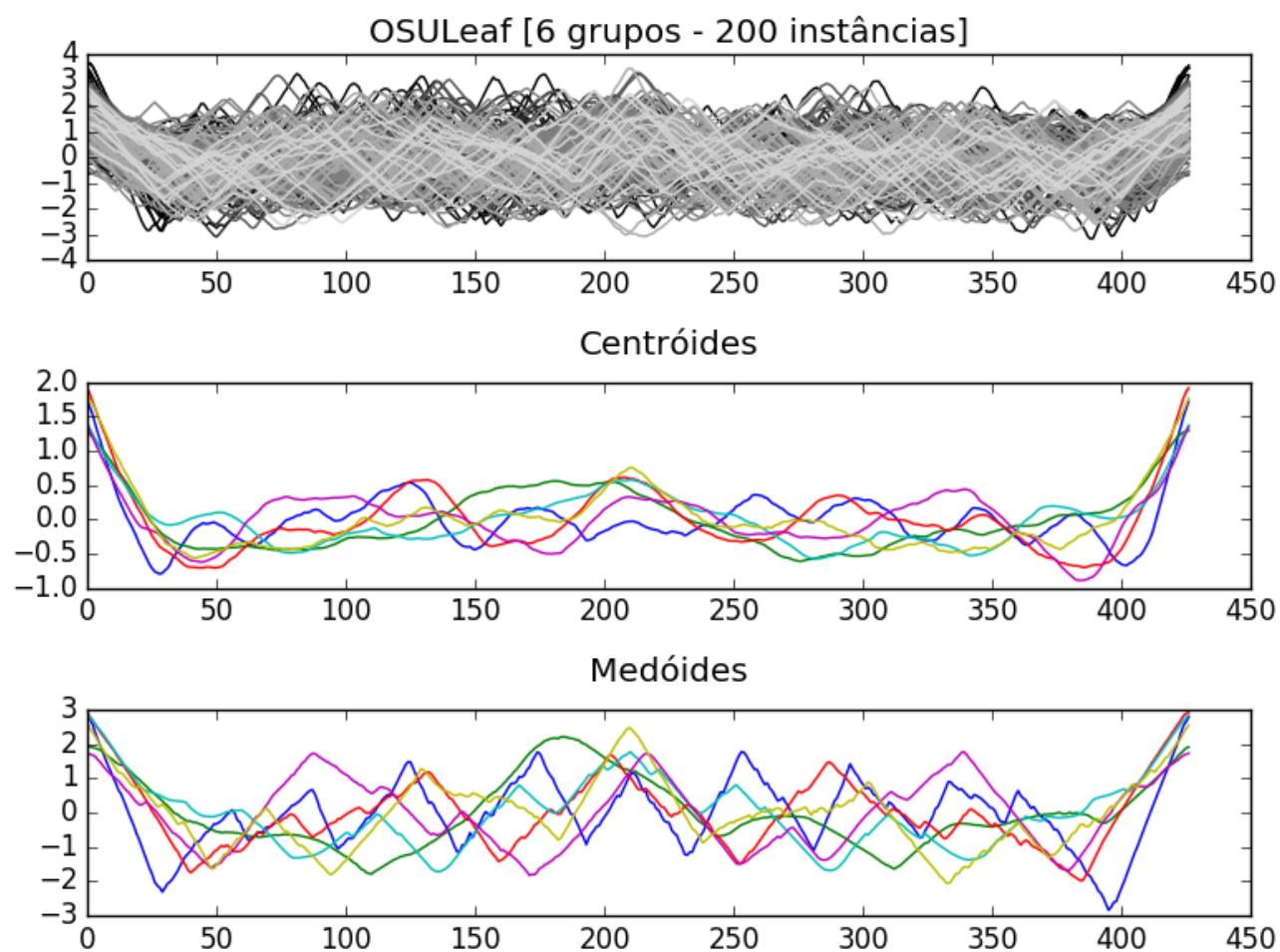


Figura A.51: Conjunto de dados 'OSULeaf'.

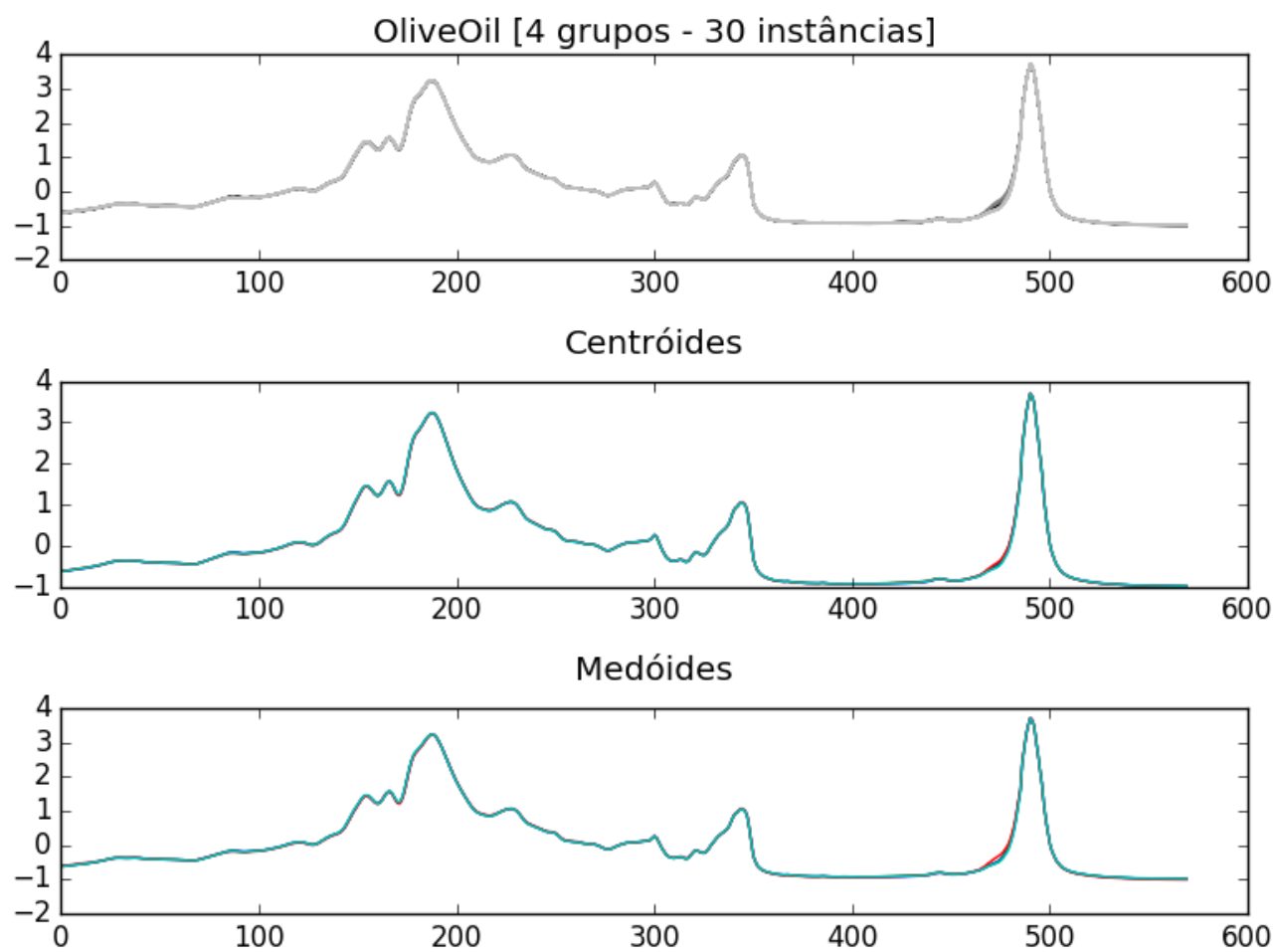


Figura A.52: Conjunto de dados 'OliveOil'.

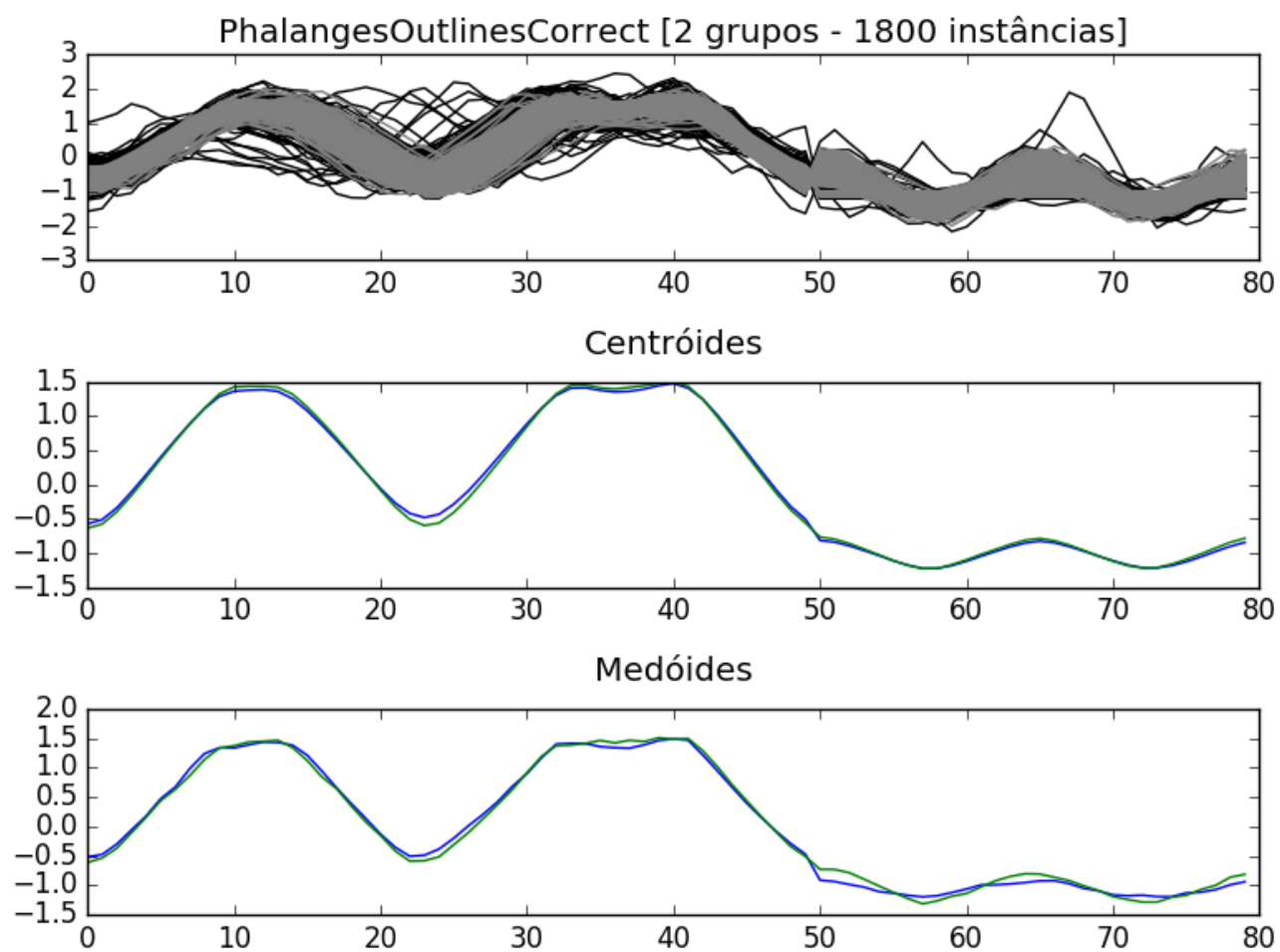


Figura A.53: Conjunto de dados 'PhalangesOutlinesCorrect'.

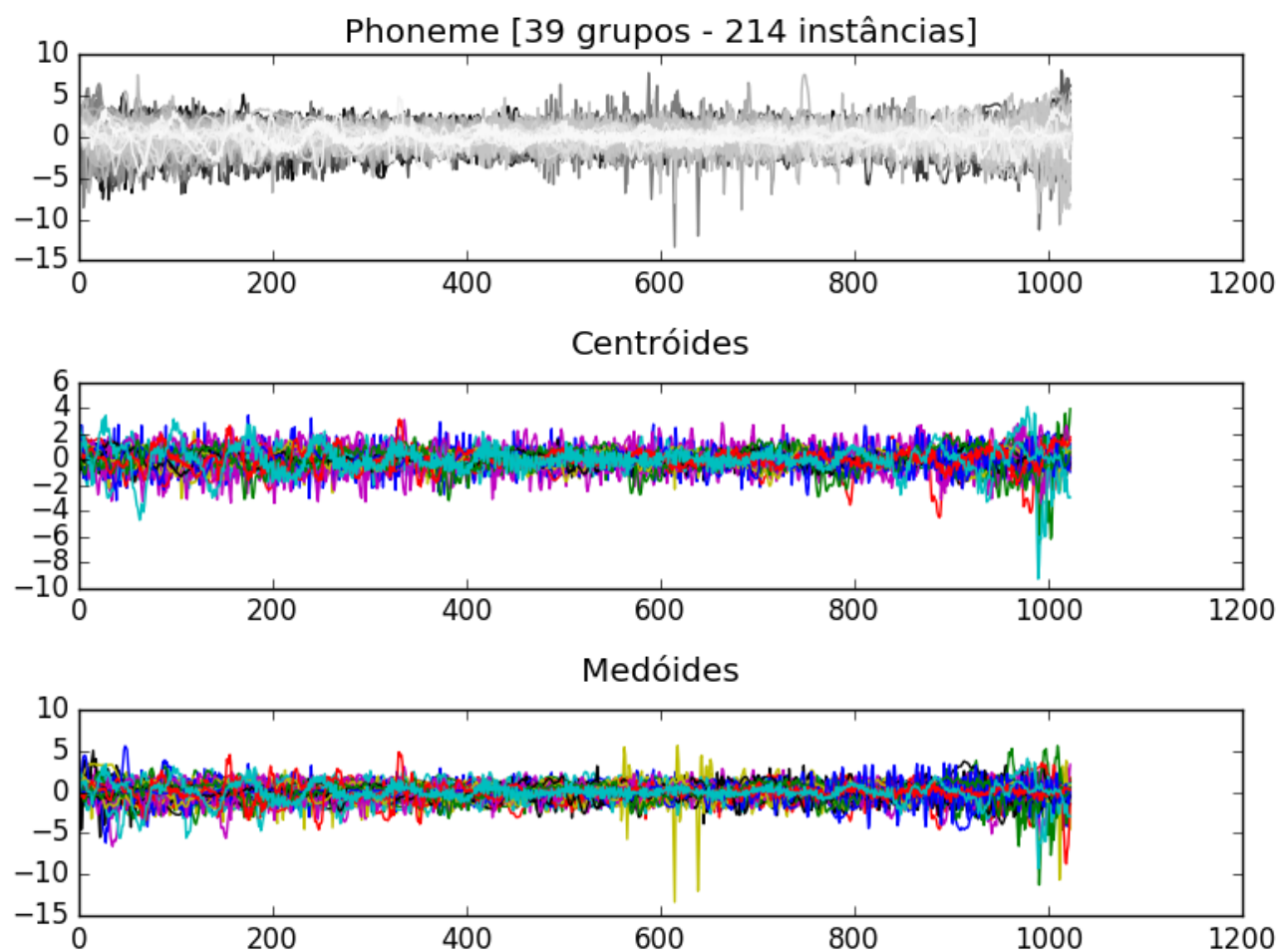


Figura A.54: Conjunto de dados 'Phoneme'.

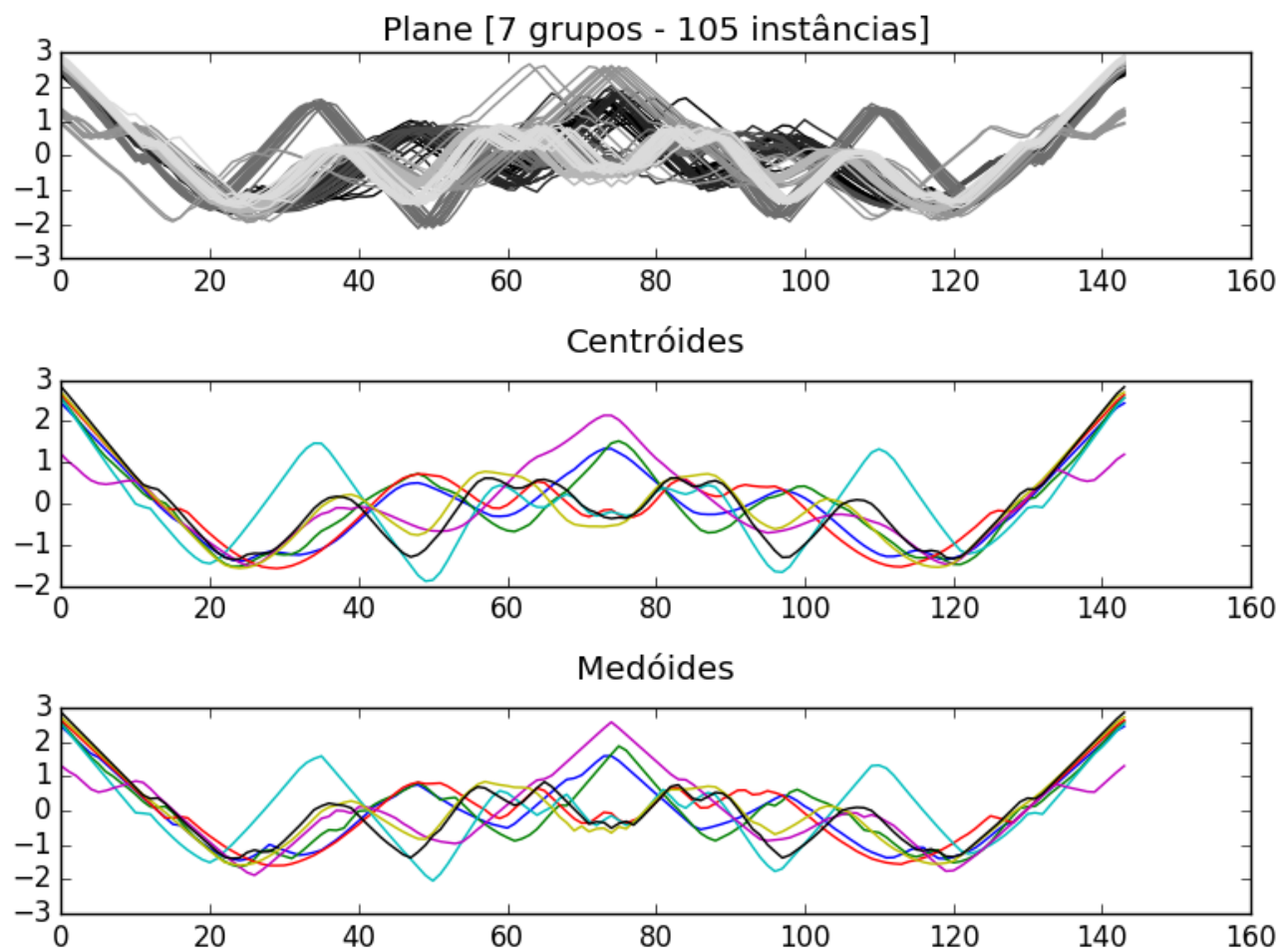


Figura A.55: Conjunto de dados 'Plane'.

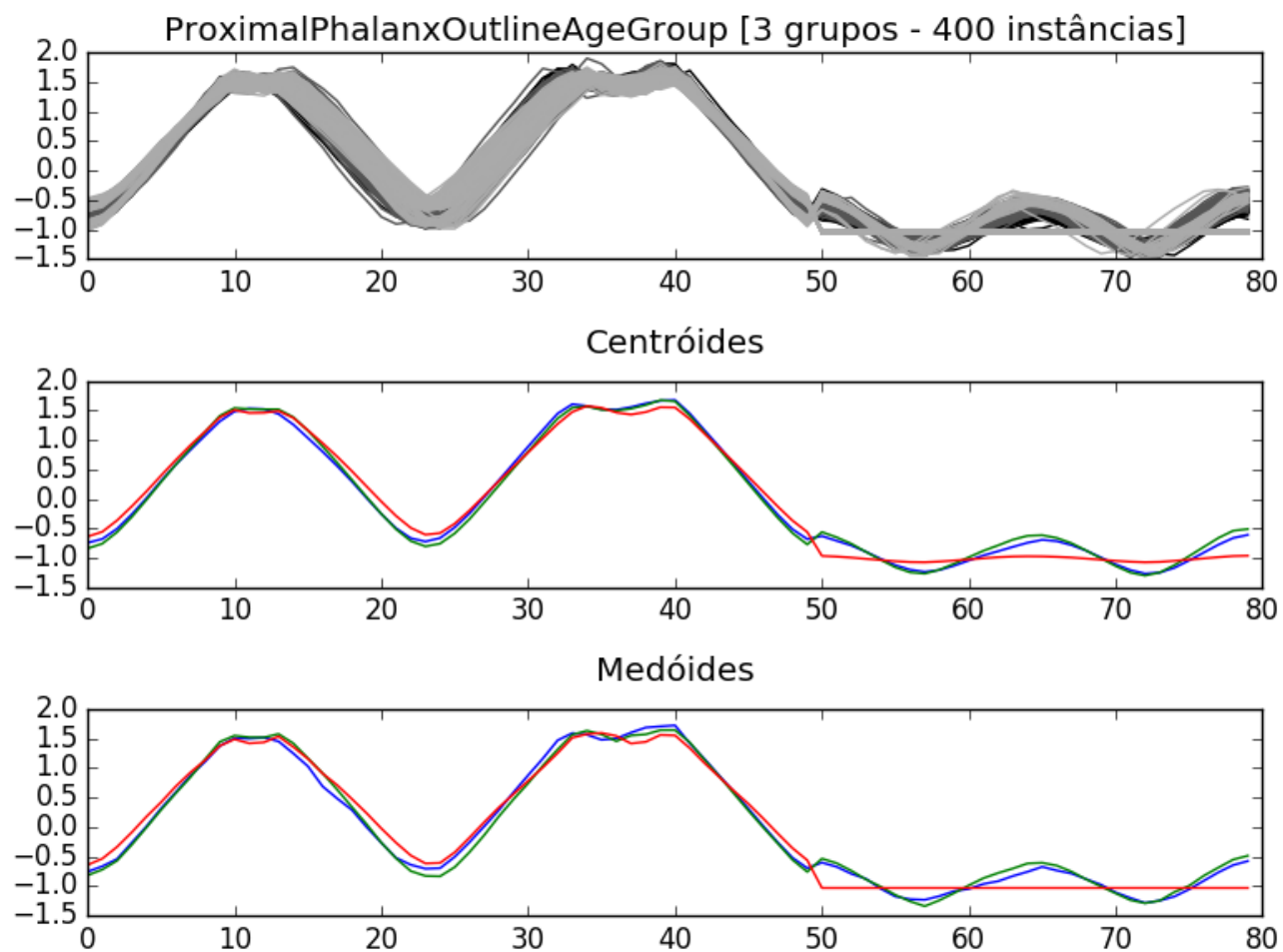


Figura A.56: Conjunto de dados 'ProximalPhalanxOutlineAgeGroup'.

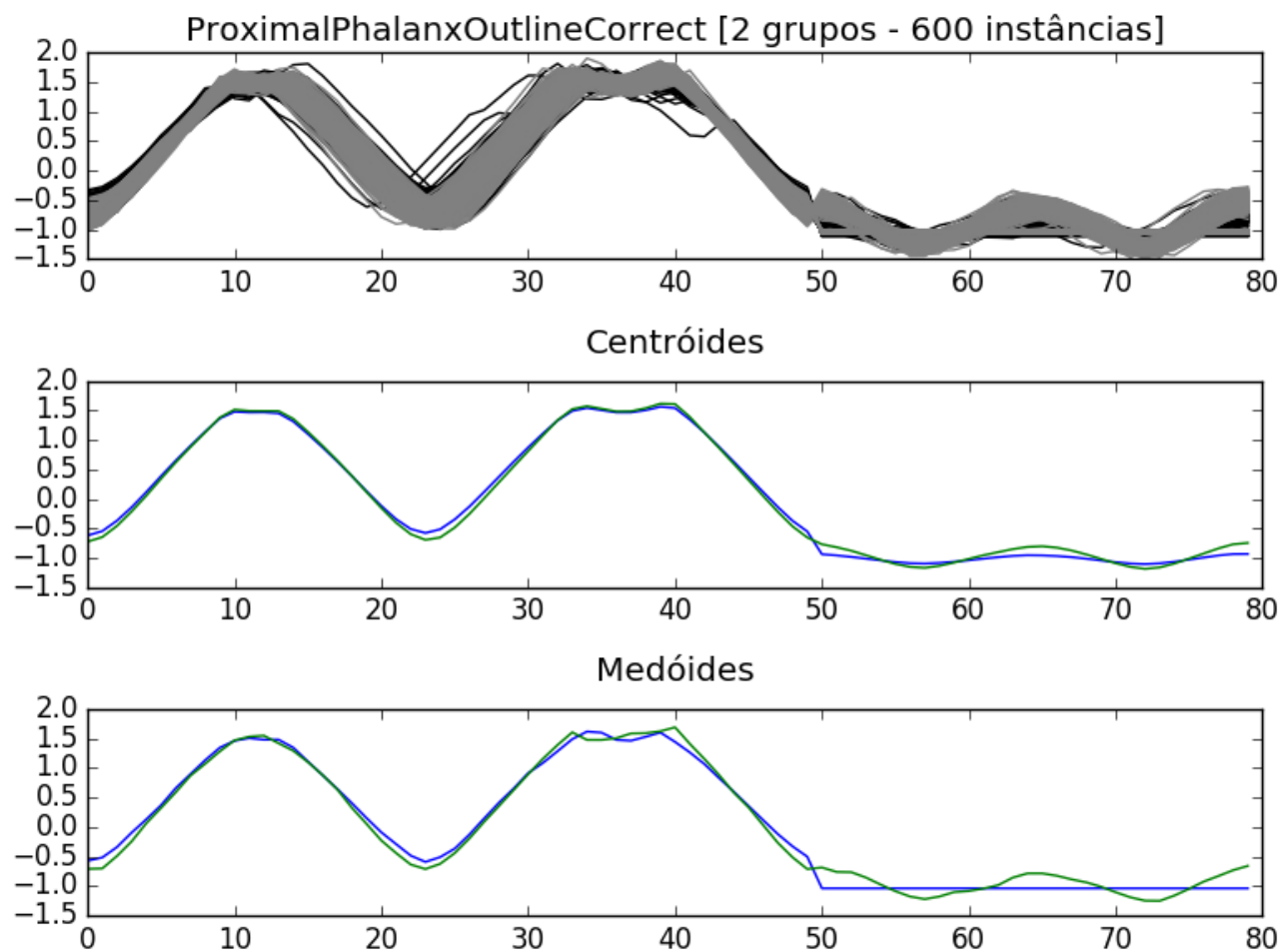


Figura A.57: Conjunto de dados 'ProximalPhalanxOutlineCorrect'.

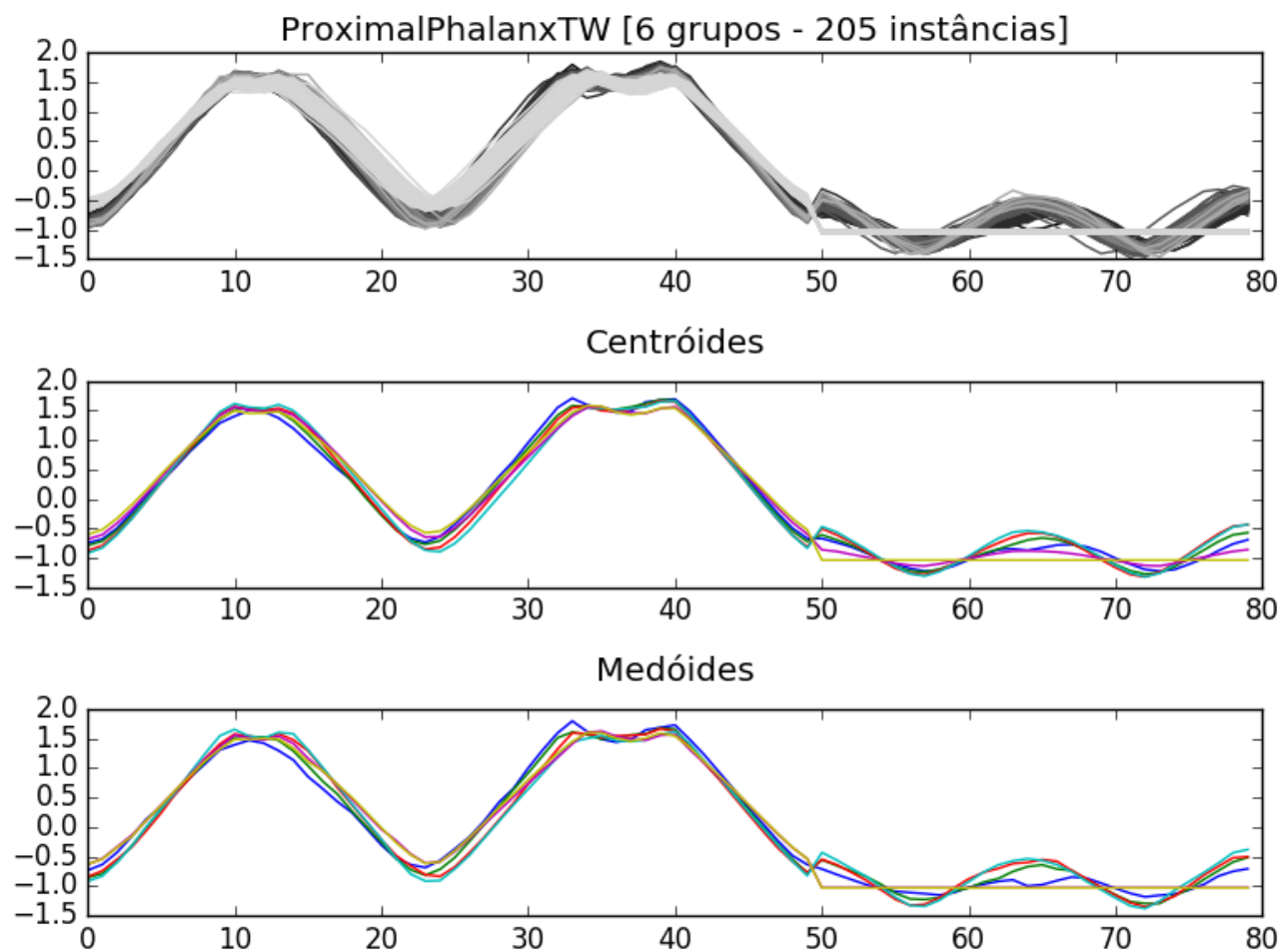


Figura A.58: Conjunto de dados 'ProximalPhalanxTW'.

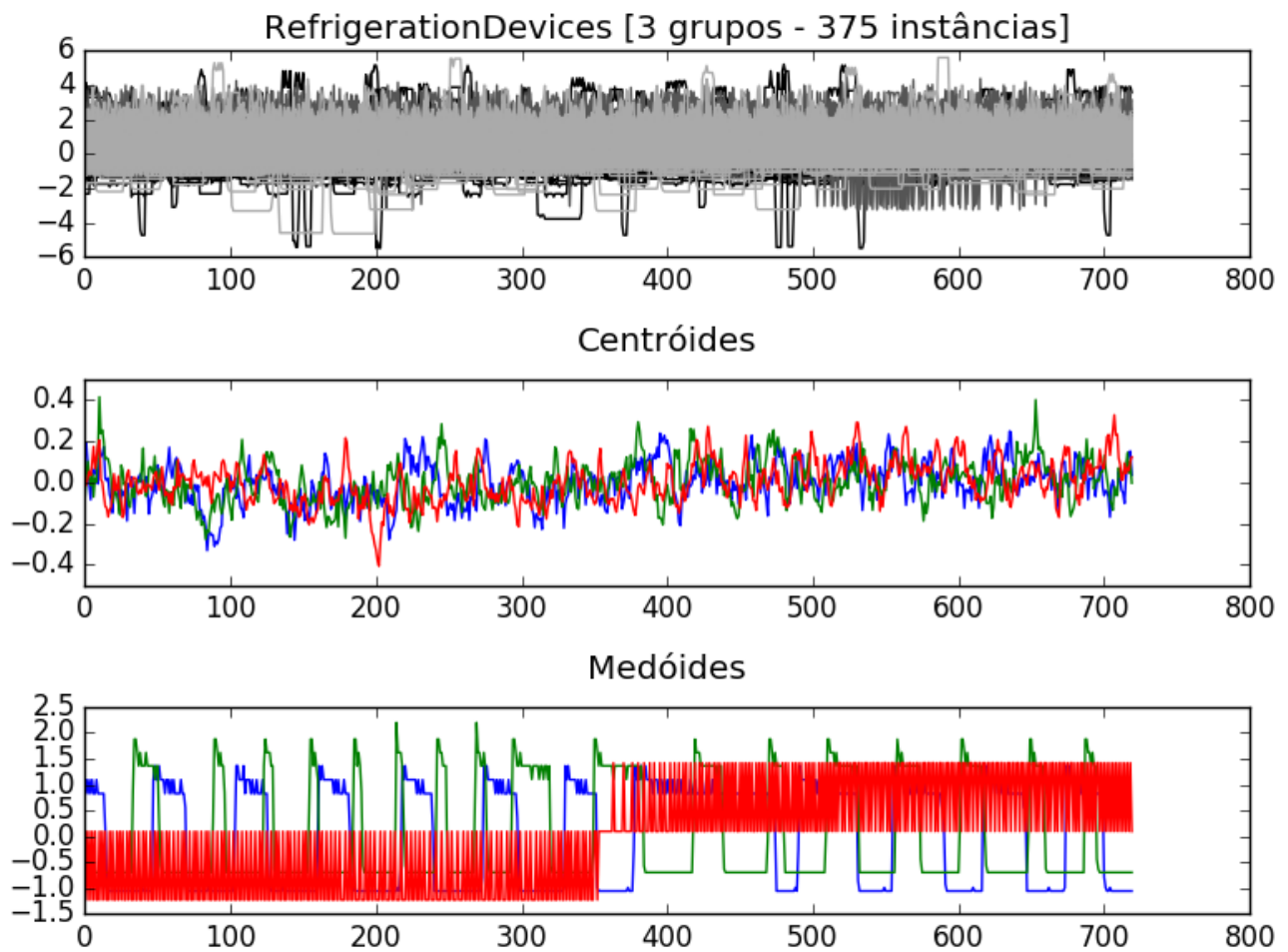


Figura A.59: Conjunto de dados 'RefrigerationDevices'.

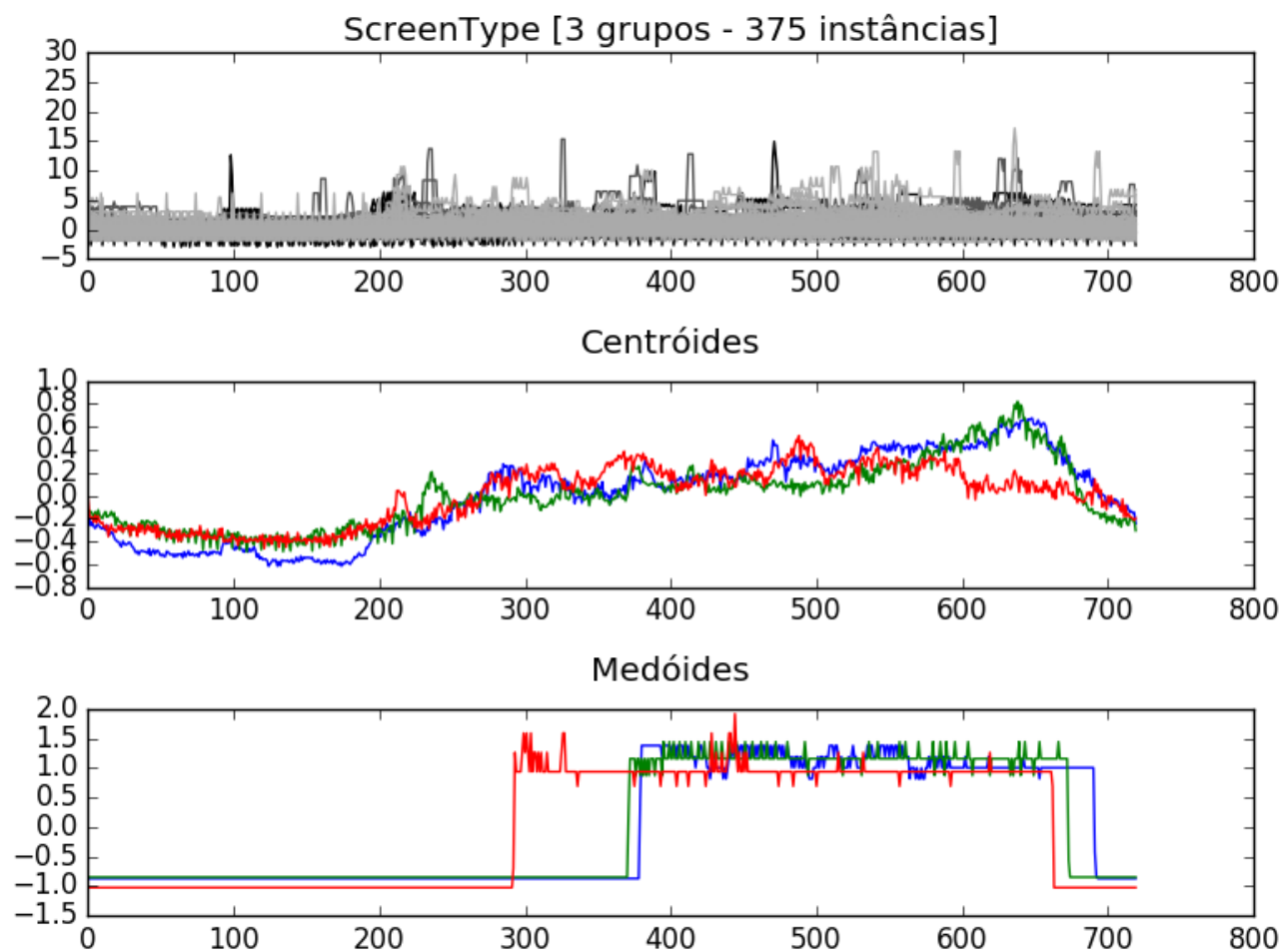


Figura A.60: Conjunto de dados 'ScreenType'.

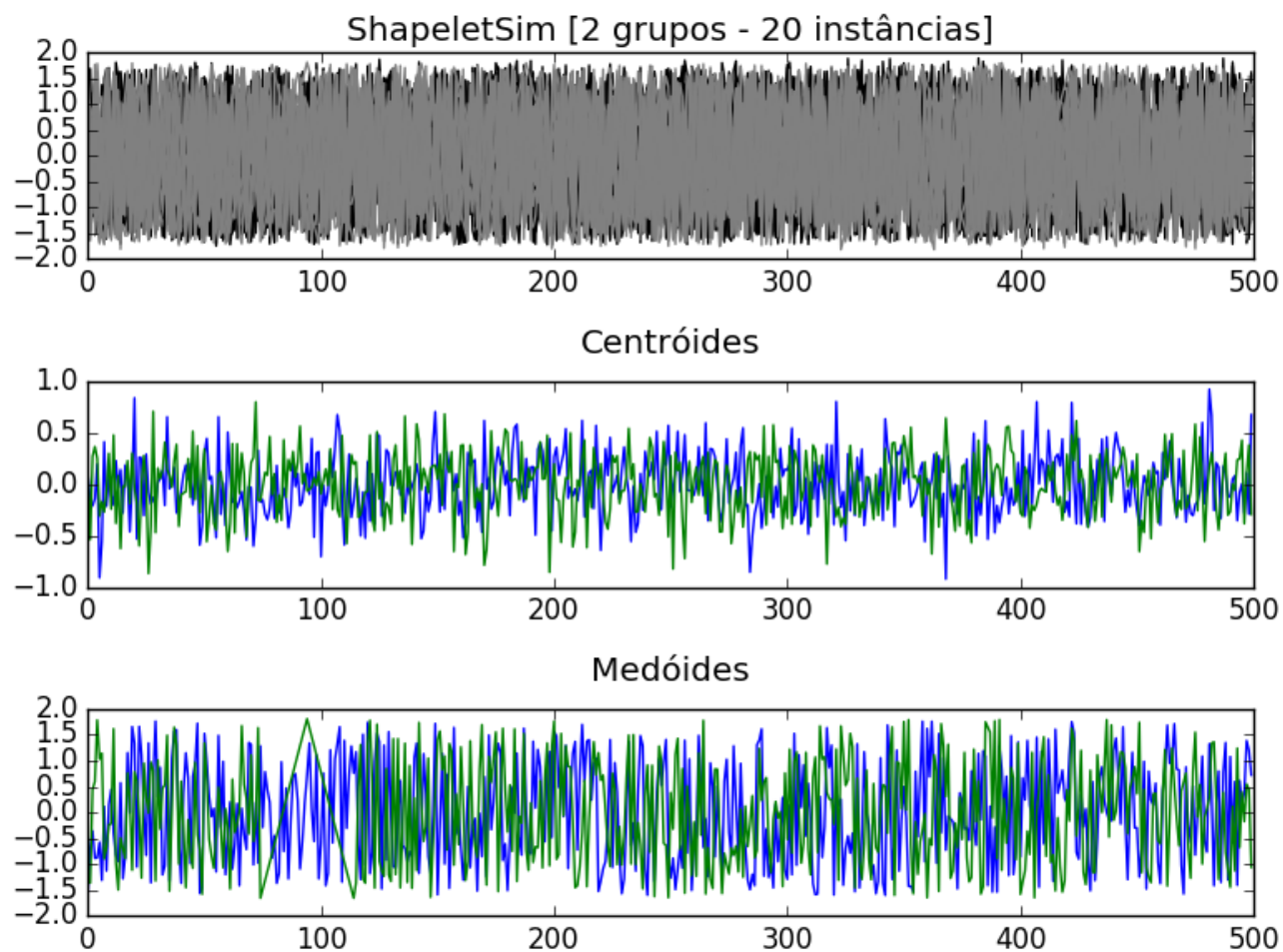


Figura A.61: Conjunto de dados 'ShapeletSim'.

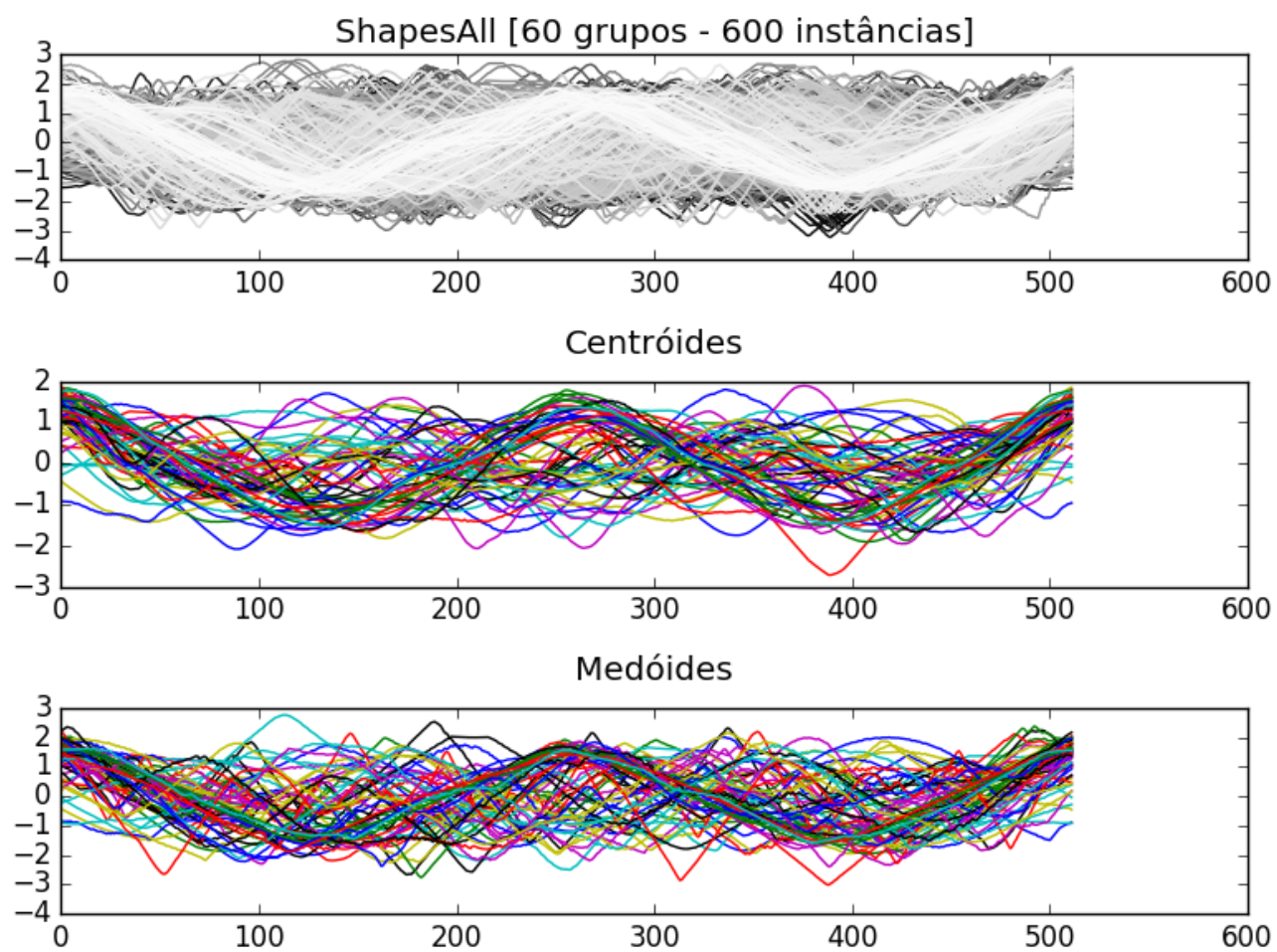


Figura A.62: Conjunto de dados 'ShapesAll'.

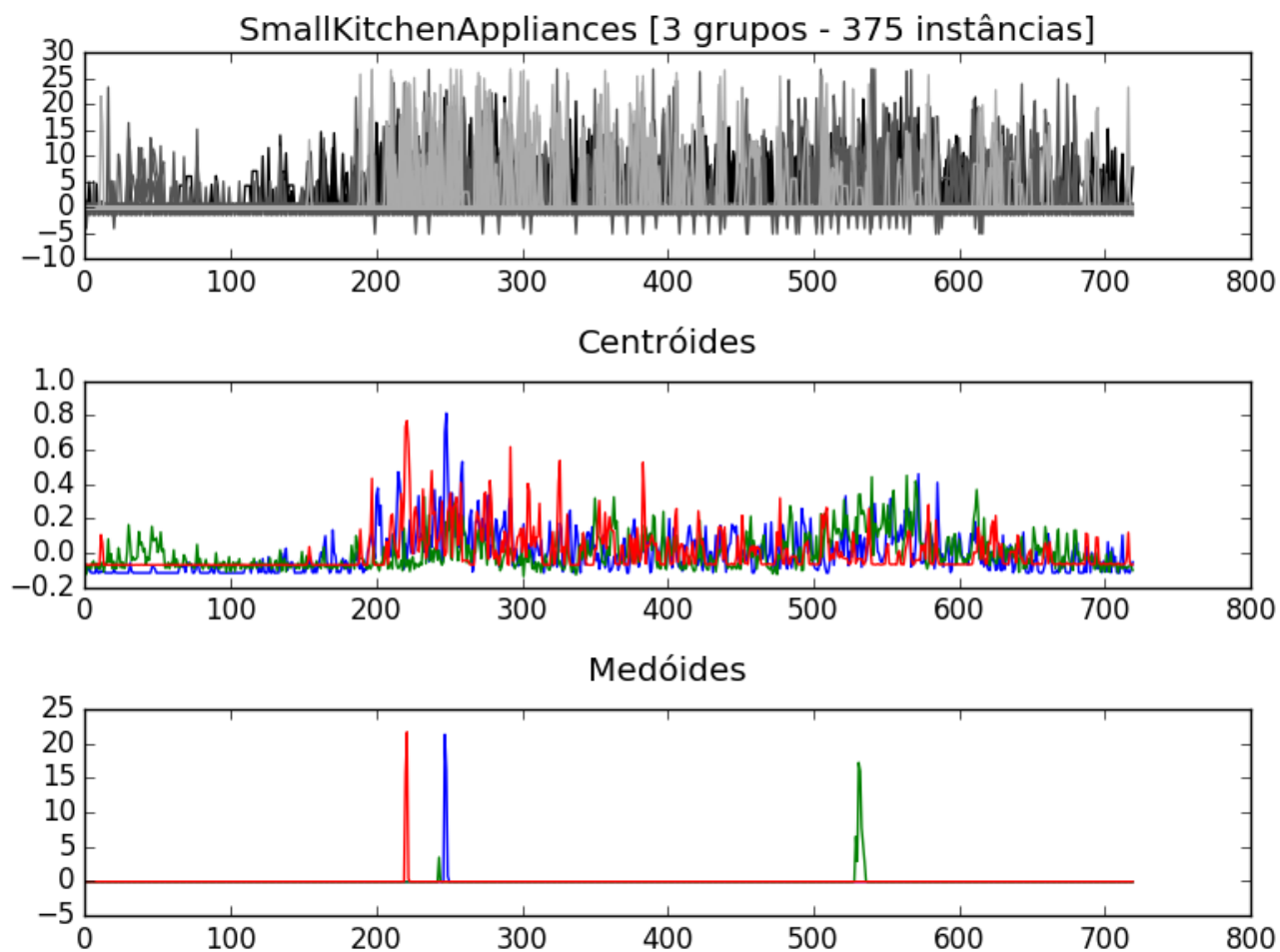


Figura A.63: Conjunto de dados 'SmallKitchenAppliances'.

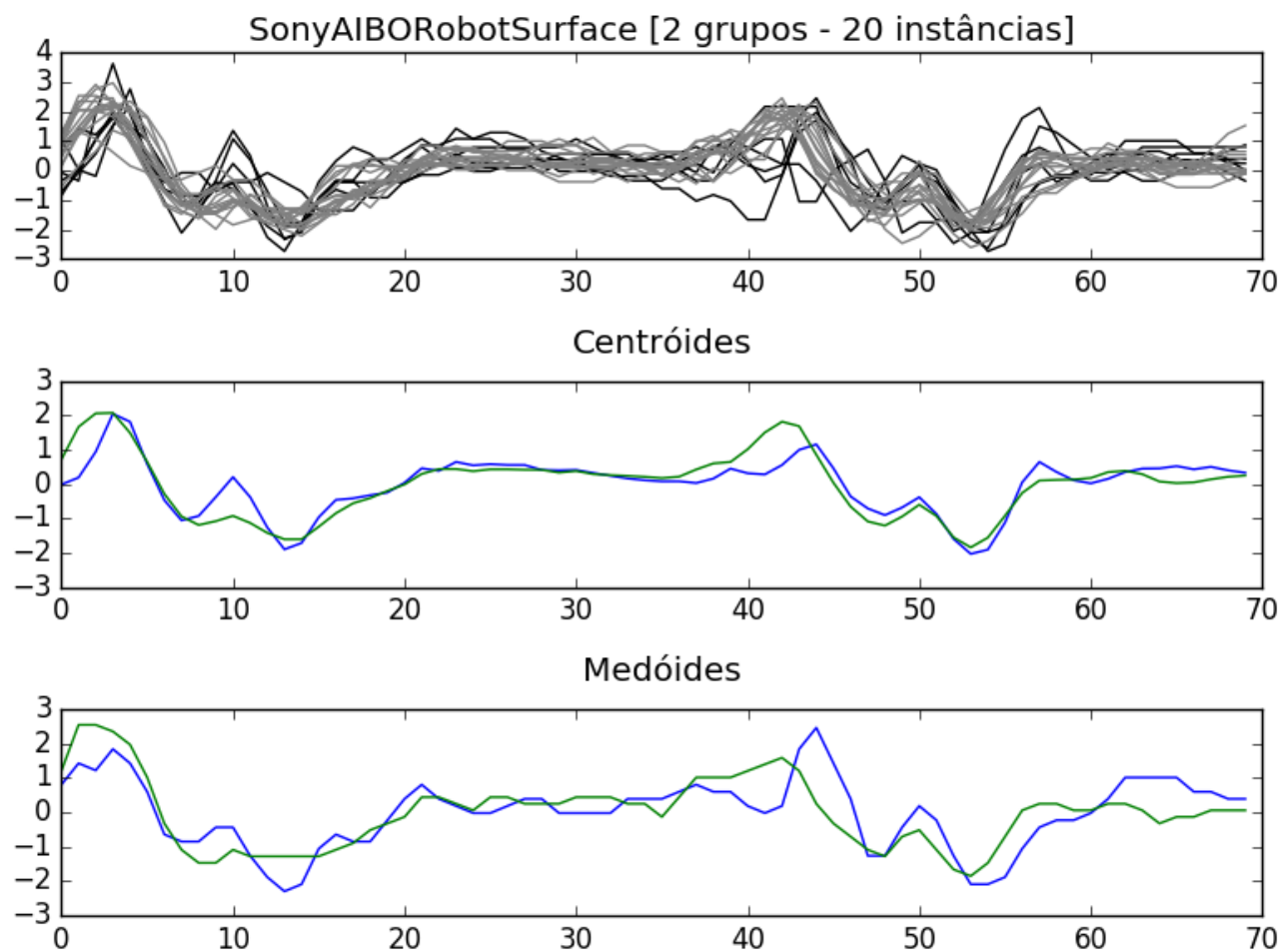


Figura A.64: Conjunto de dados 'SonyAIBORobotSurface'.

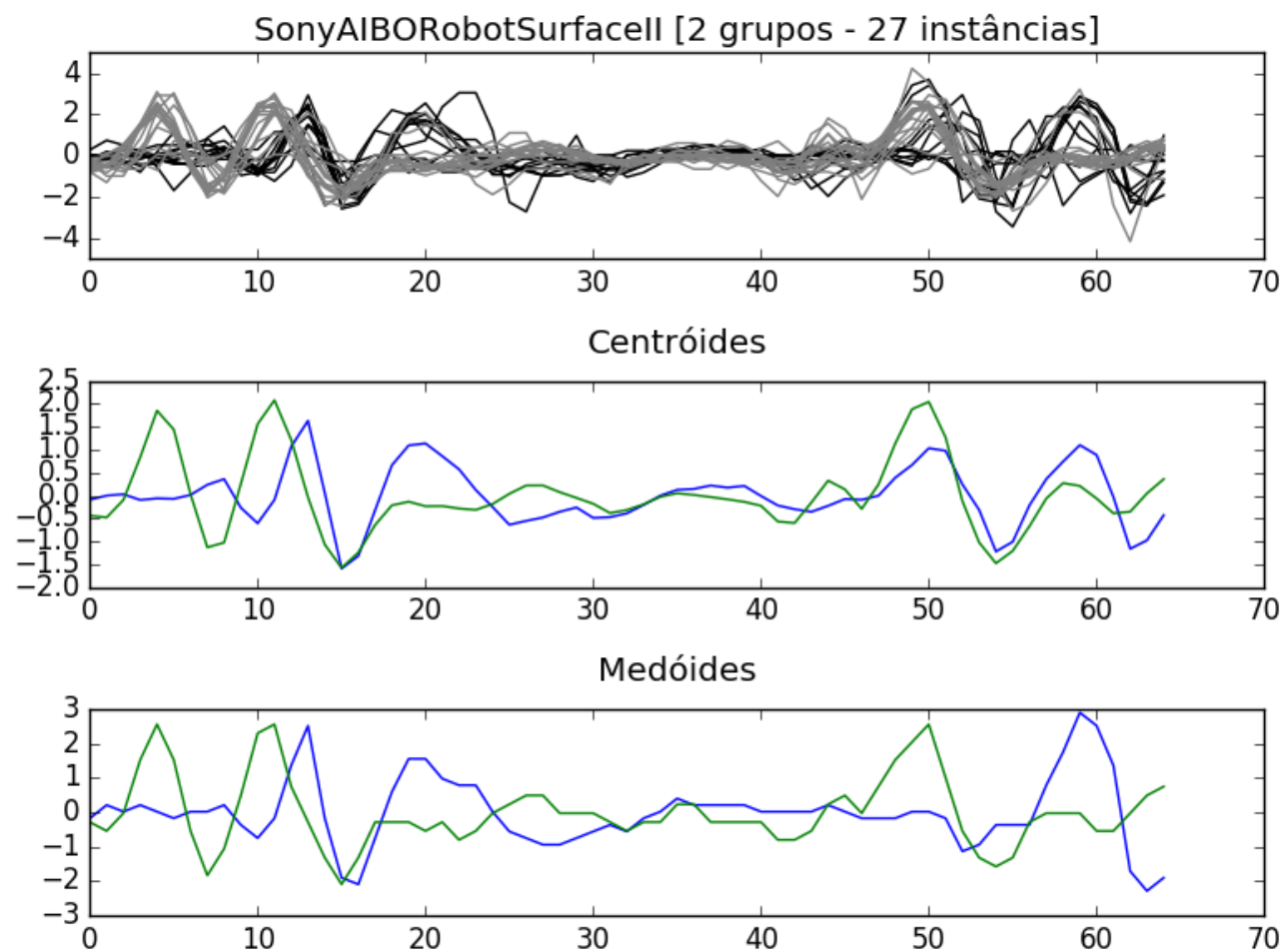


Figura A.65: Conjunto de dados 'SonyAIBORobotSurfaceII'.

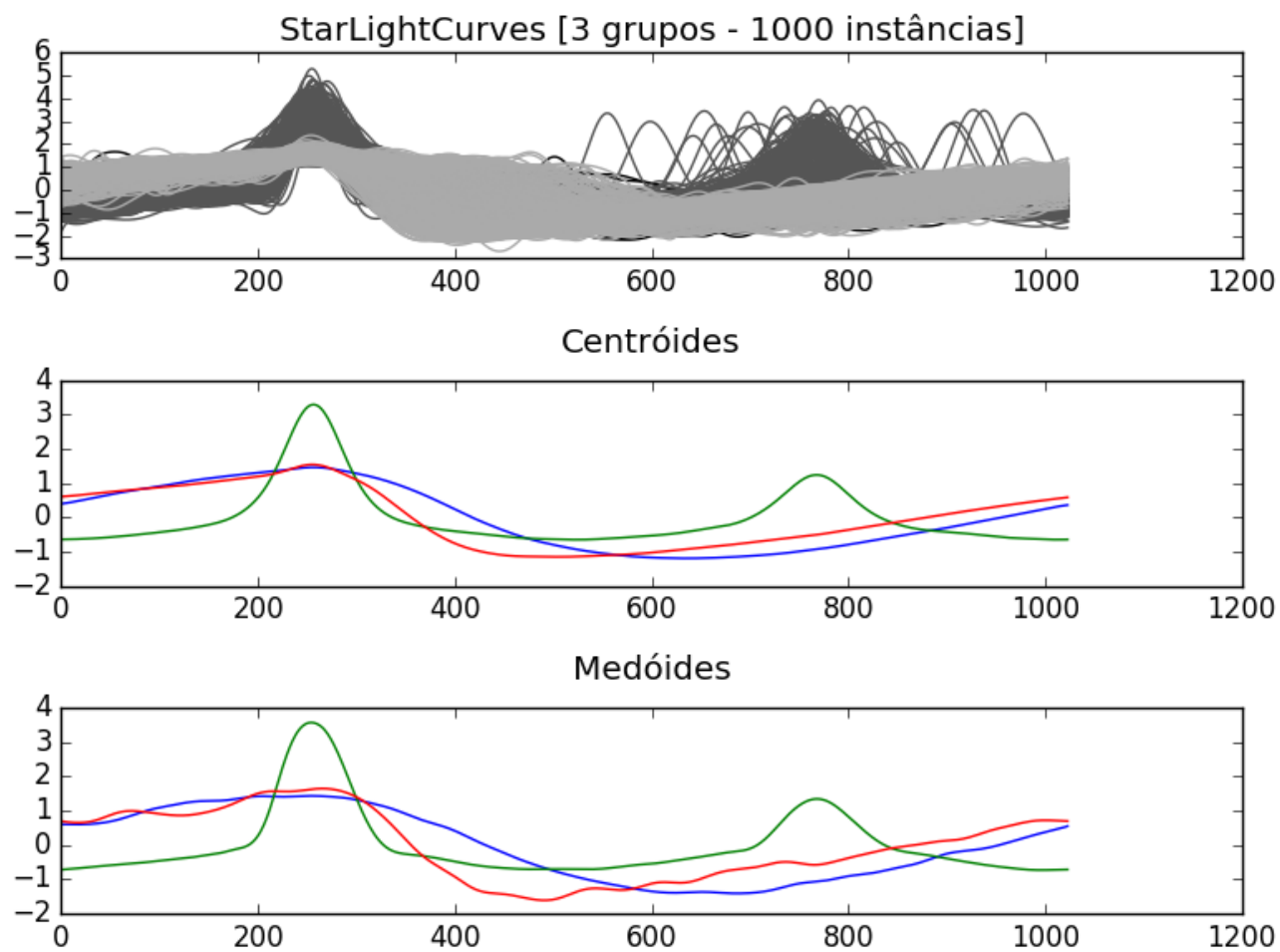


Figura A.66: Conjunto de dados 'StarLightCurves'.

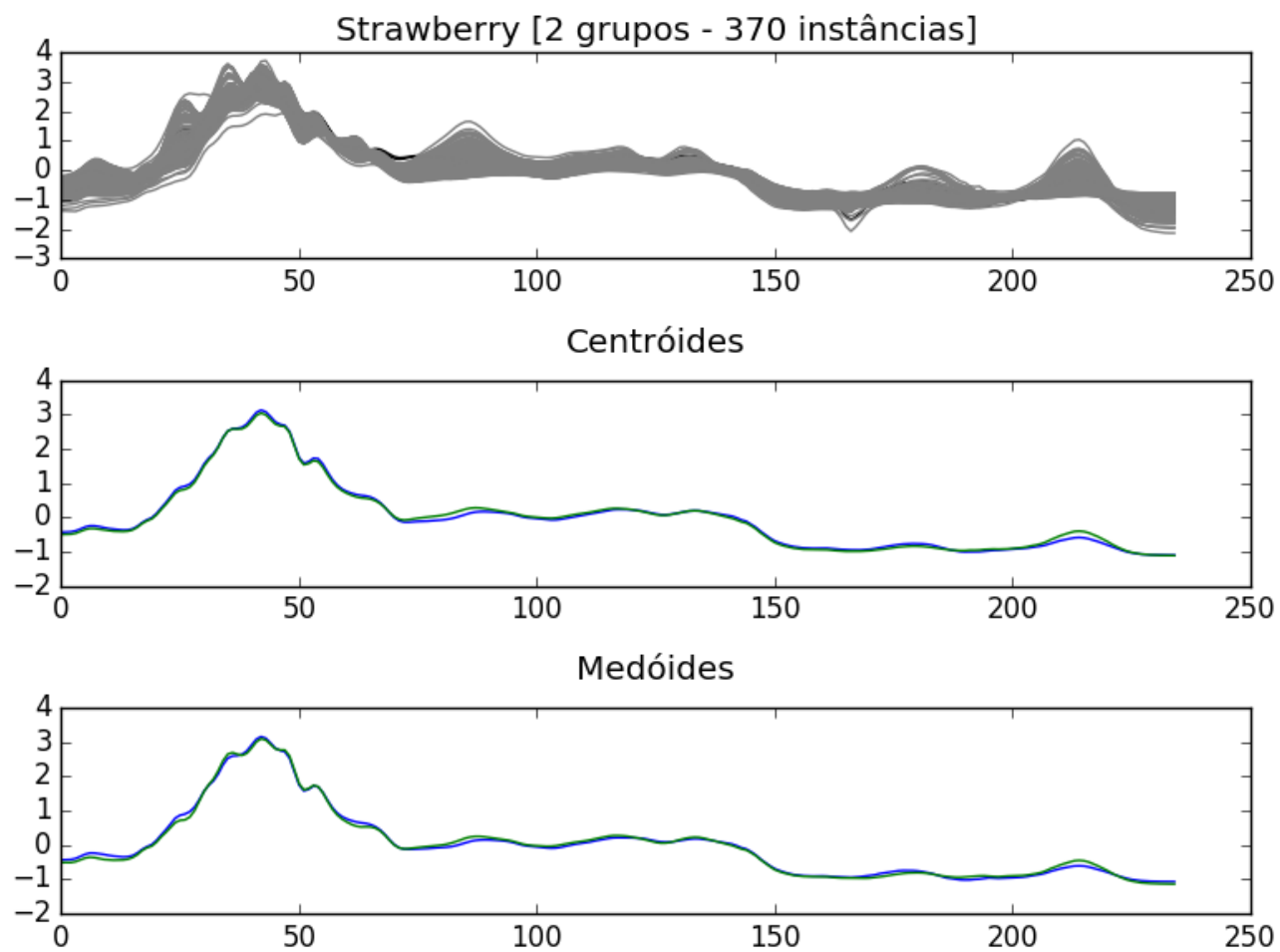


Figura A.67: Conjunto de dados 'Strawberry'.

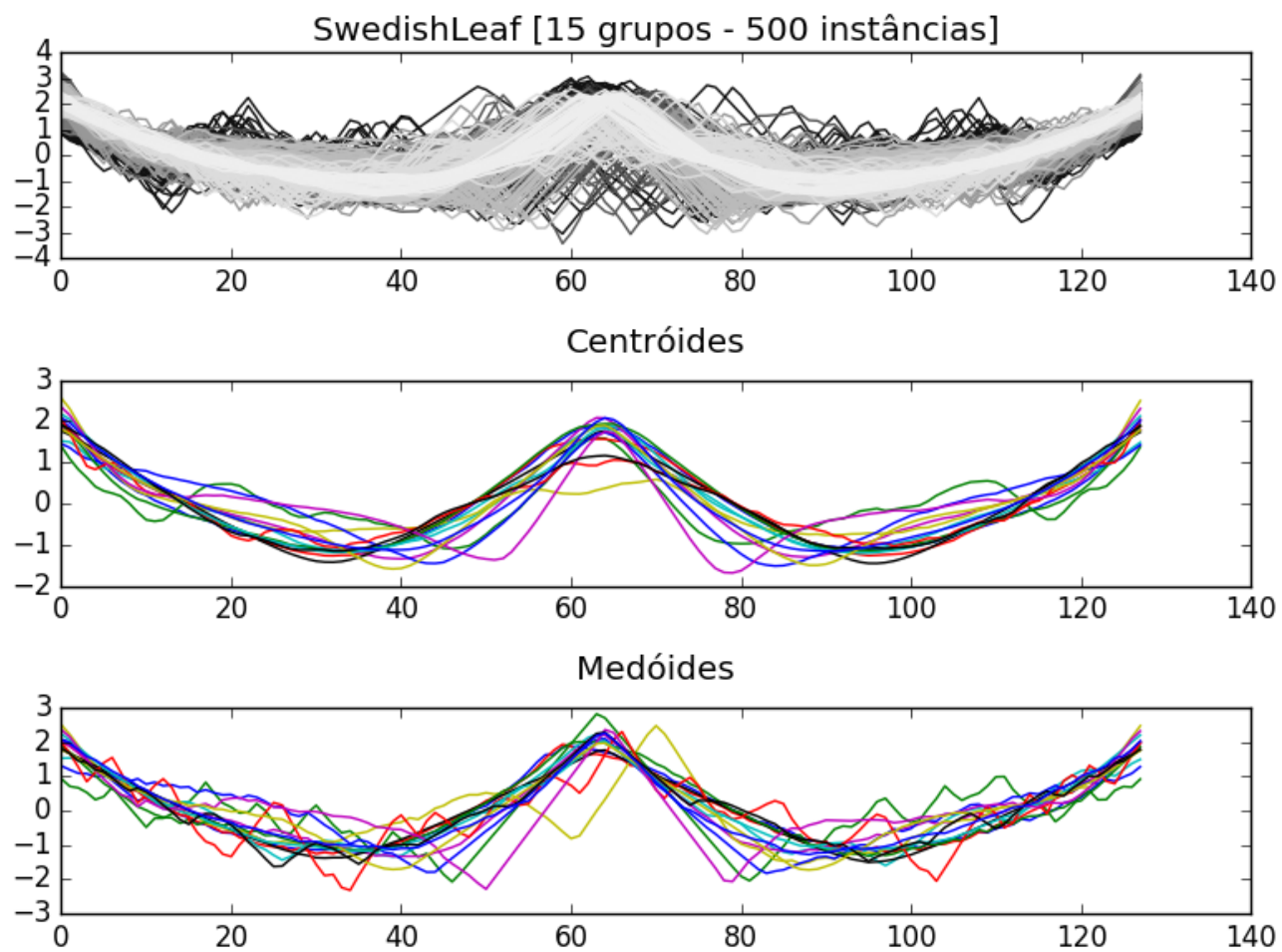


Figura A.68: Conjunto de dados 'SwedishLeaf'.

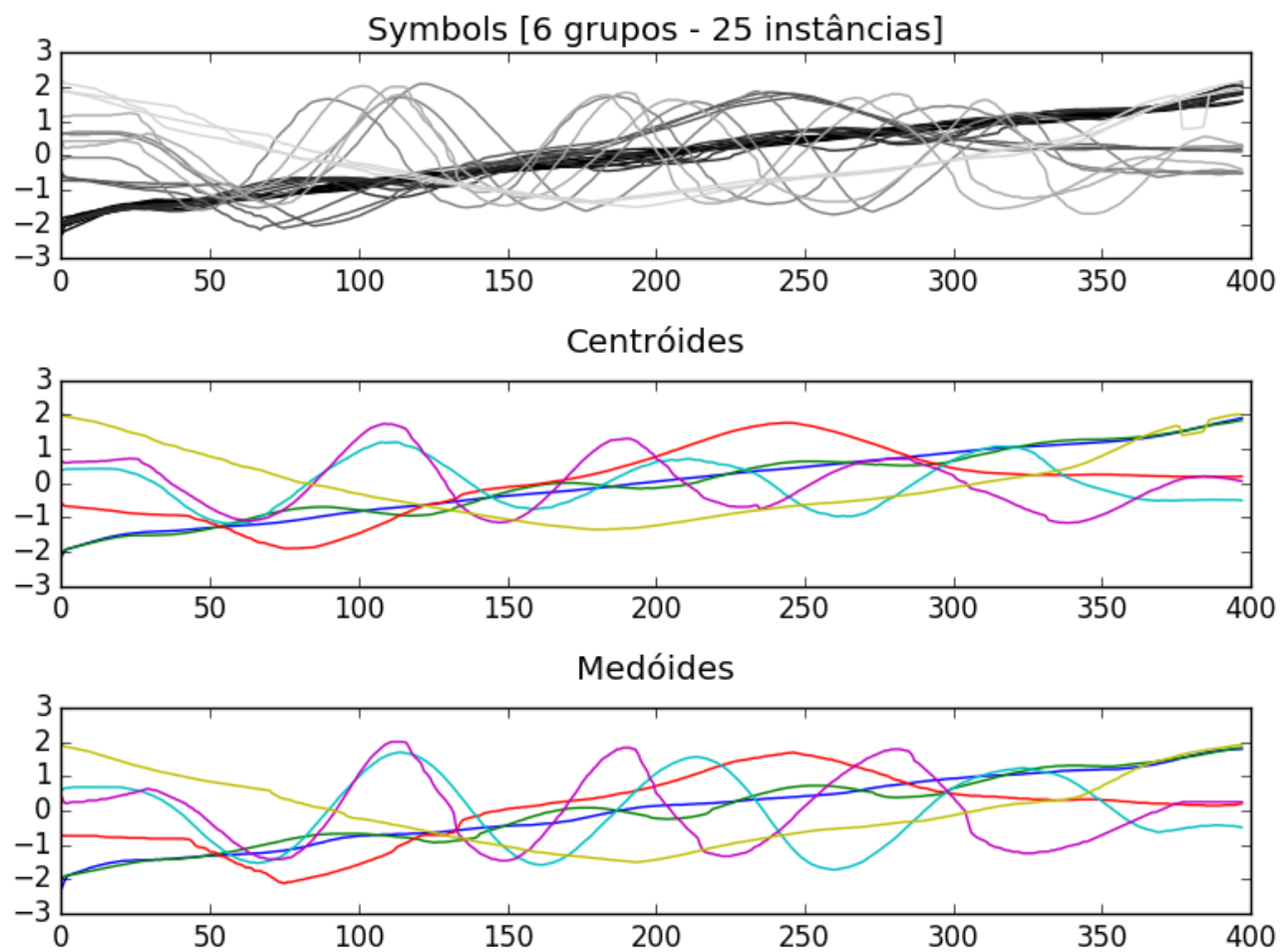


Figura A.69: Conjunto de dados 'Symbols'.

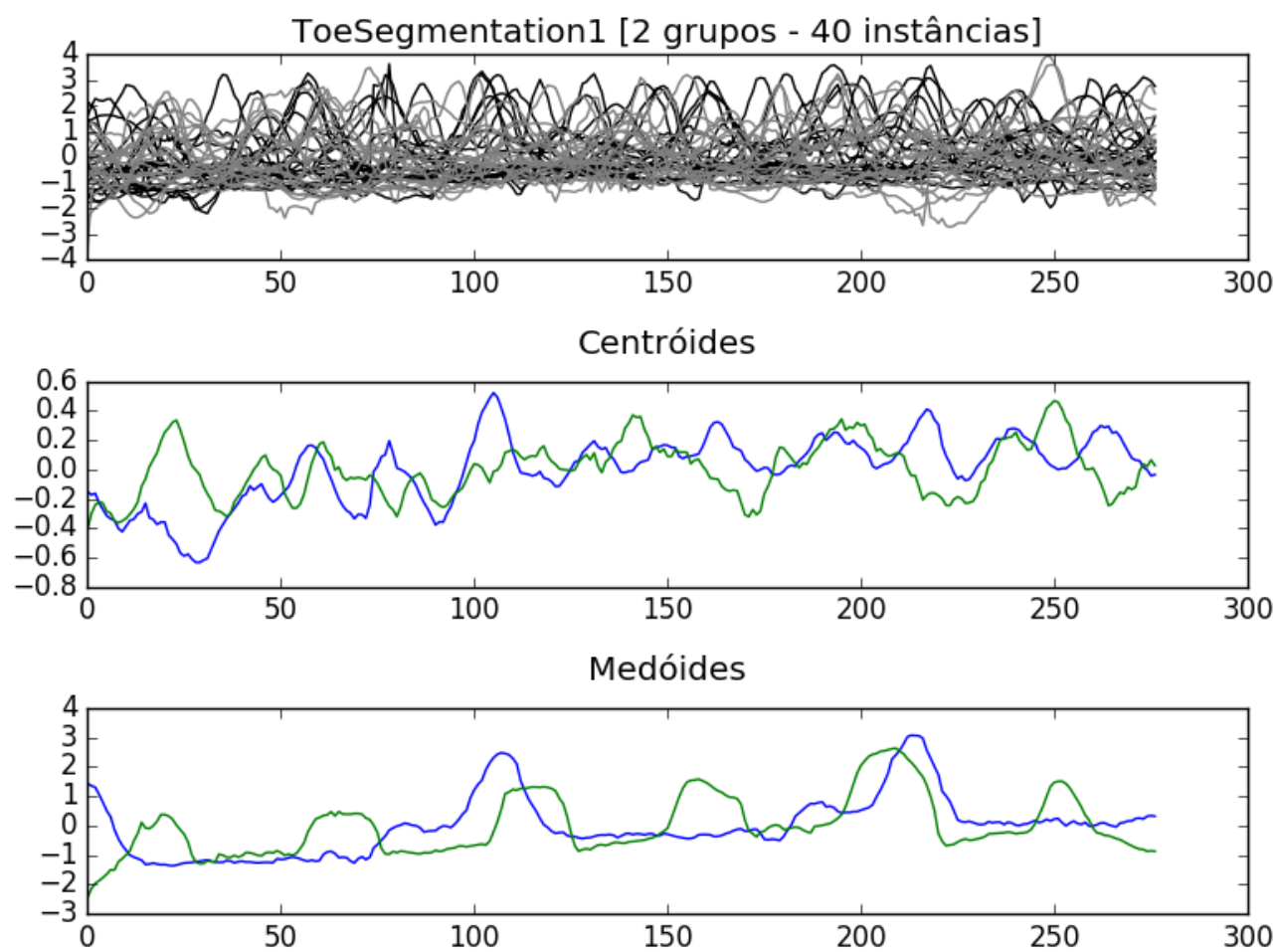


Figura A.70: Conjunto de dados 'ToeSegmentation1'.

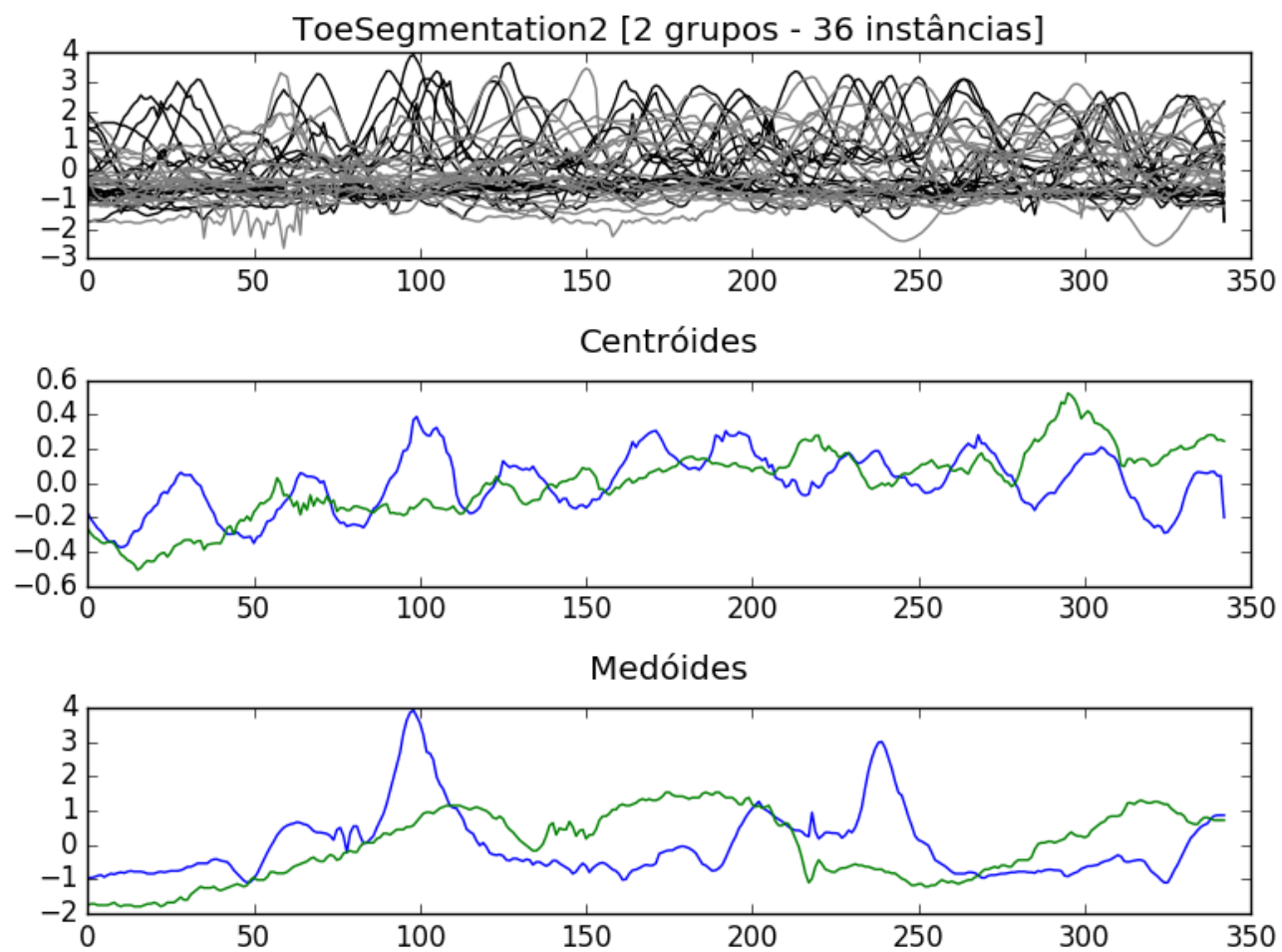


Figura A.71: Conjunto de dados 'ToeSegmentation2'.

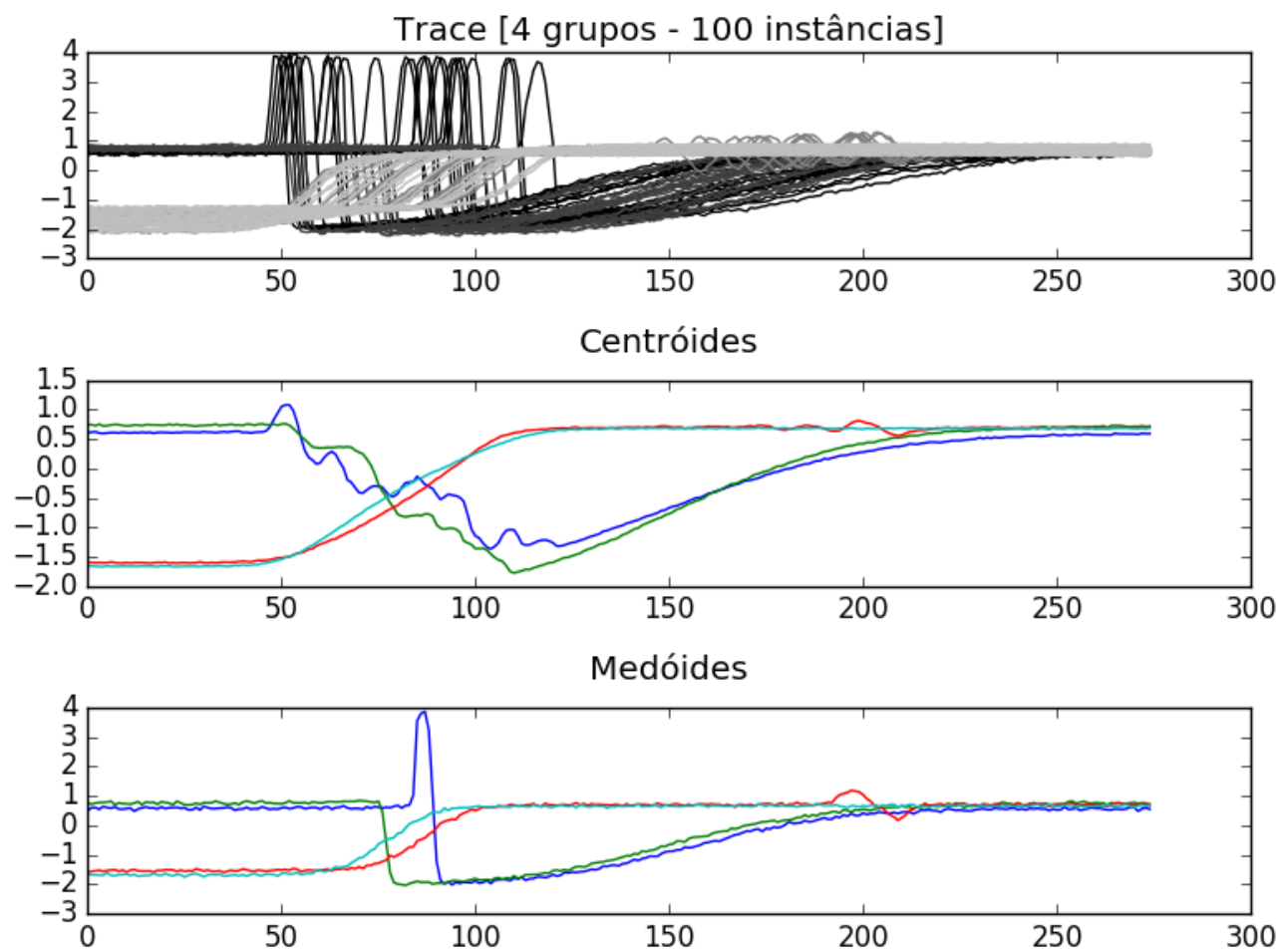


Figura A.72: Conjunto de dados 'Trace'.

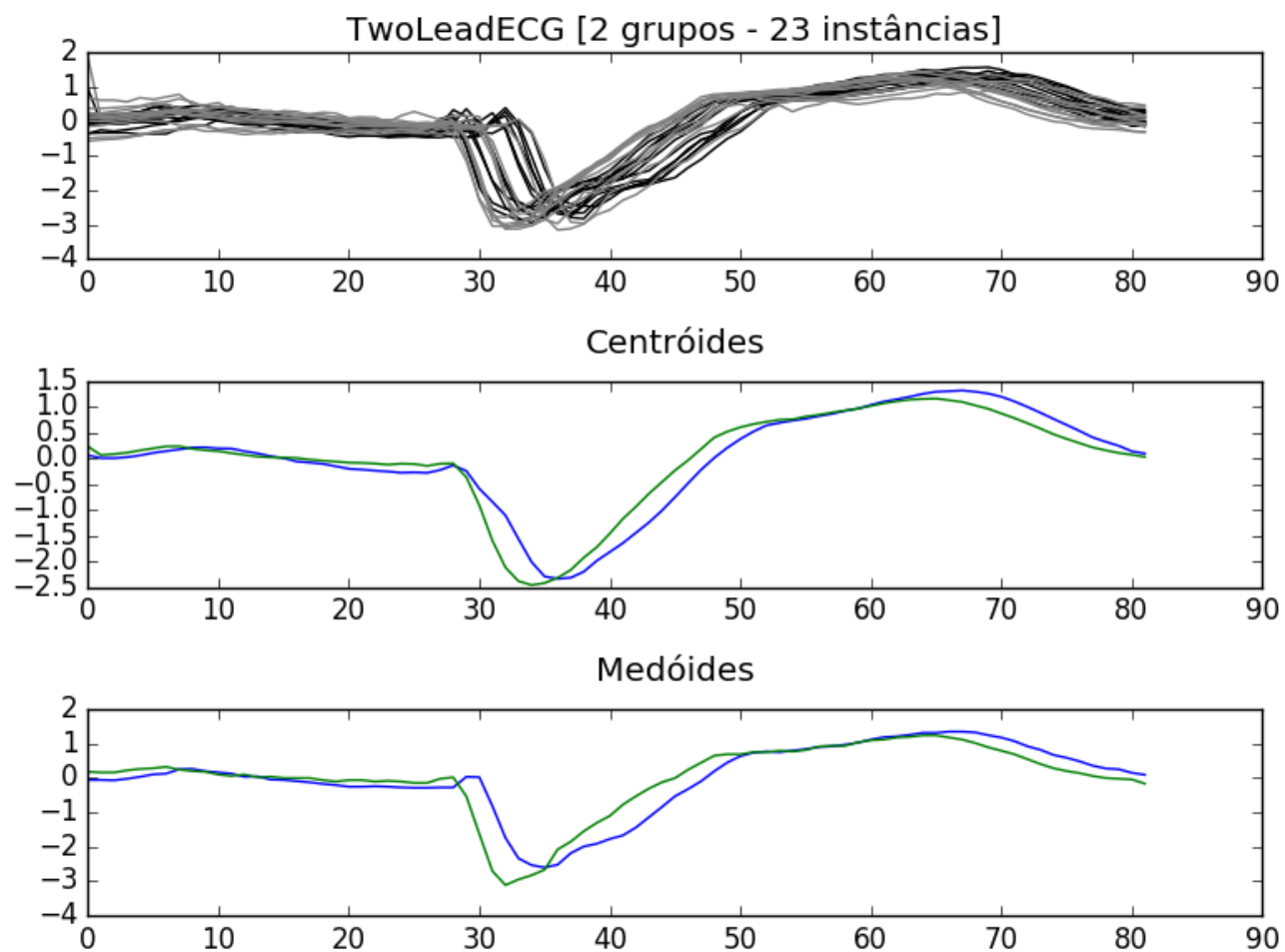


Figura A.73: Conjunto de dados 'TwoLeadECG'.

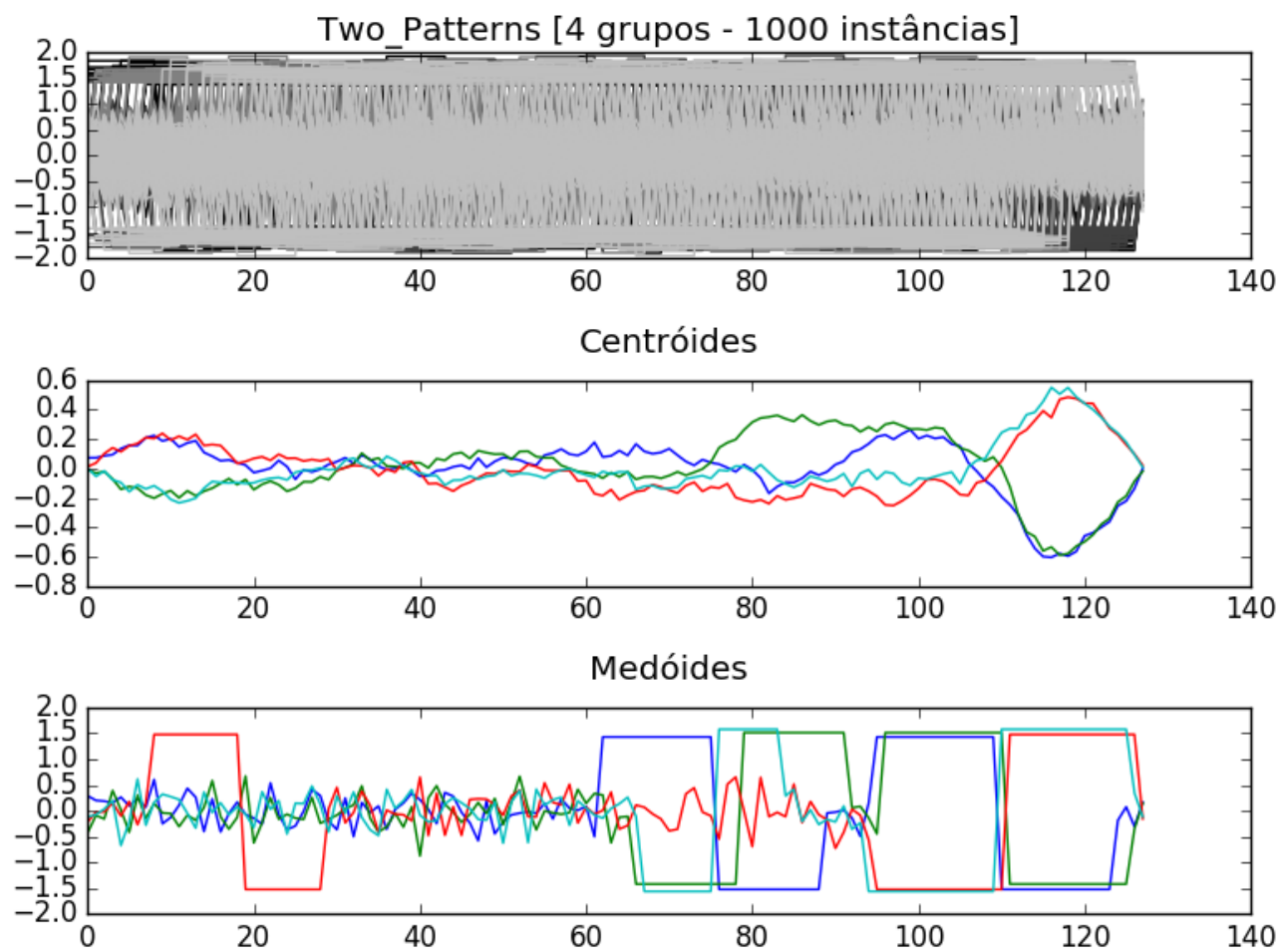


Figura A.74: Conjunto de dados 'Two_Patterns'.

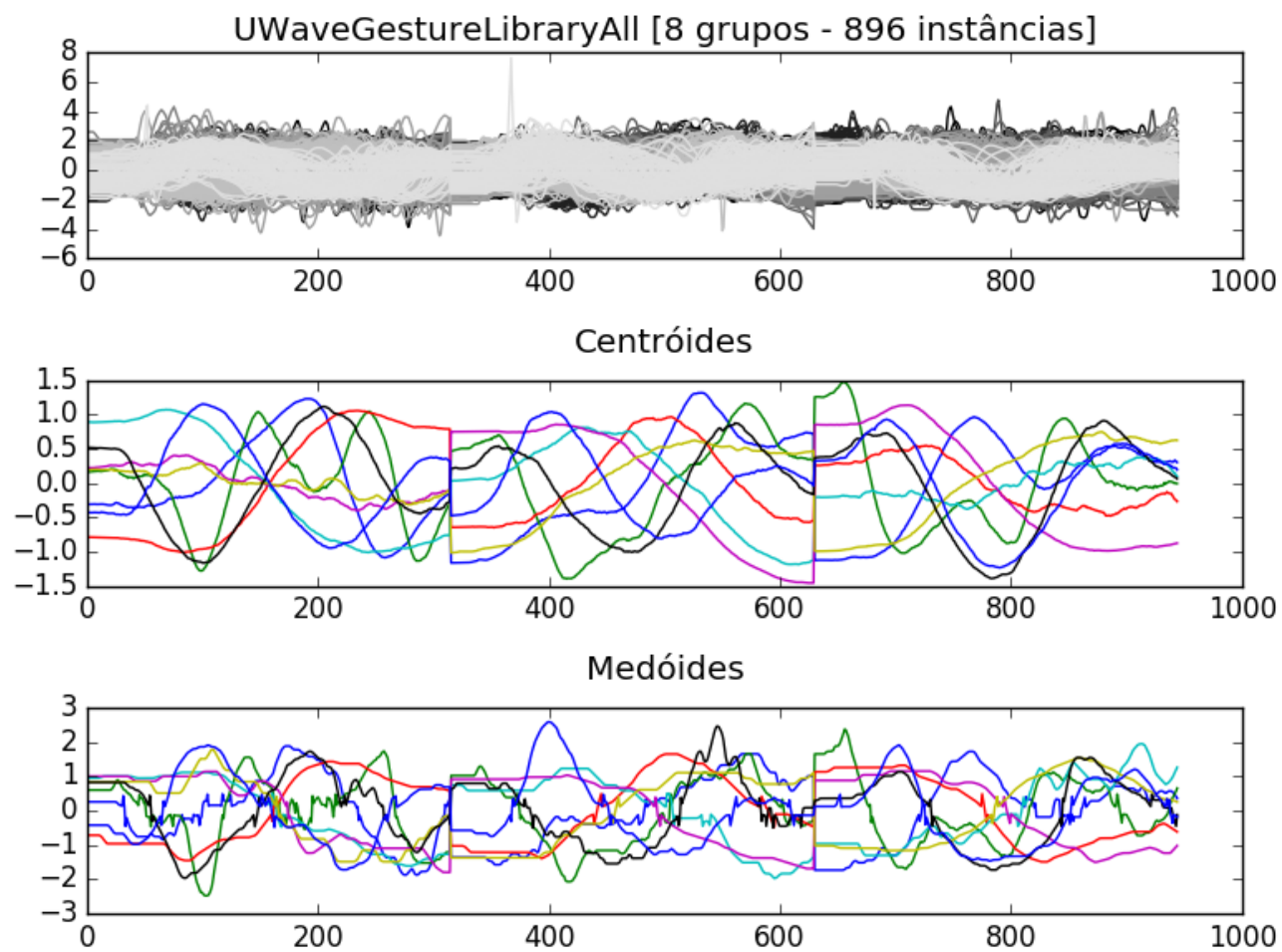


Figura A.75: Conjunto de dados 'UWaveGestureLibraryAll'.

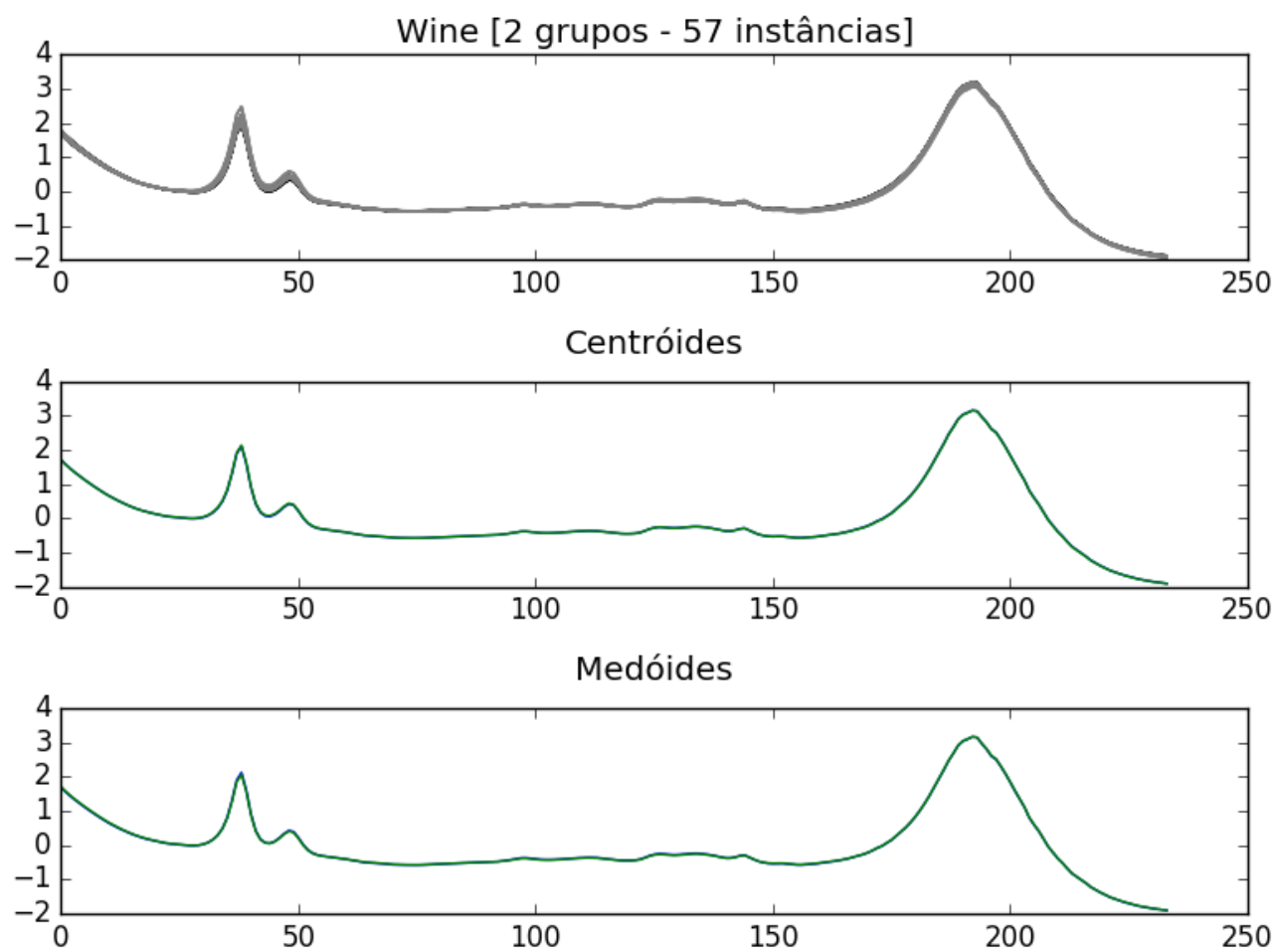


Figura A.76: Conjunto de dados 'Wine'.

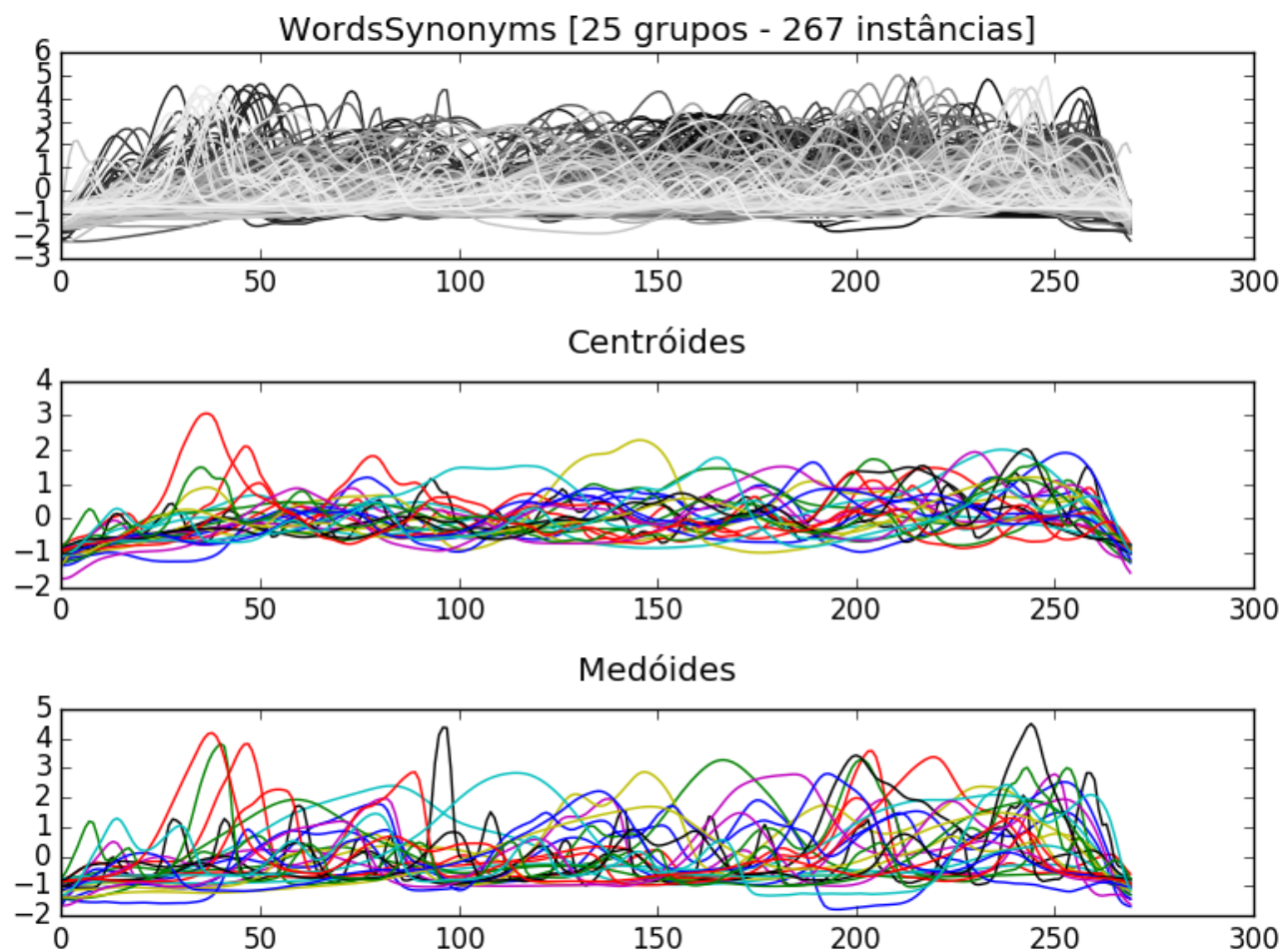


Figura A.77: Conjunto de dados 'WordsSynonyms'.

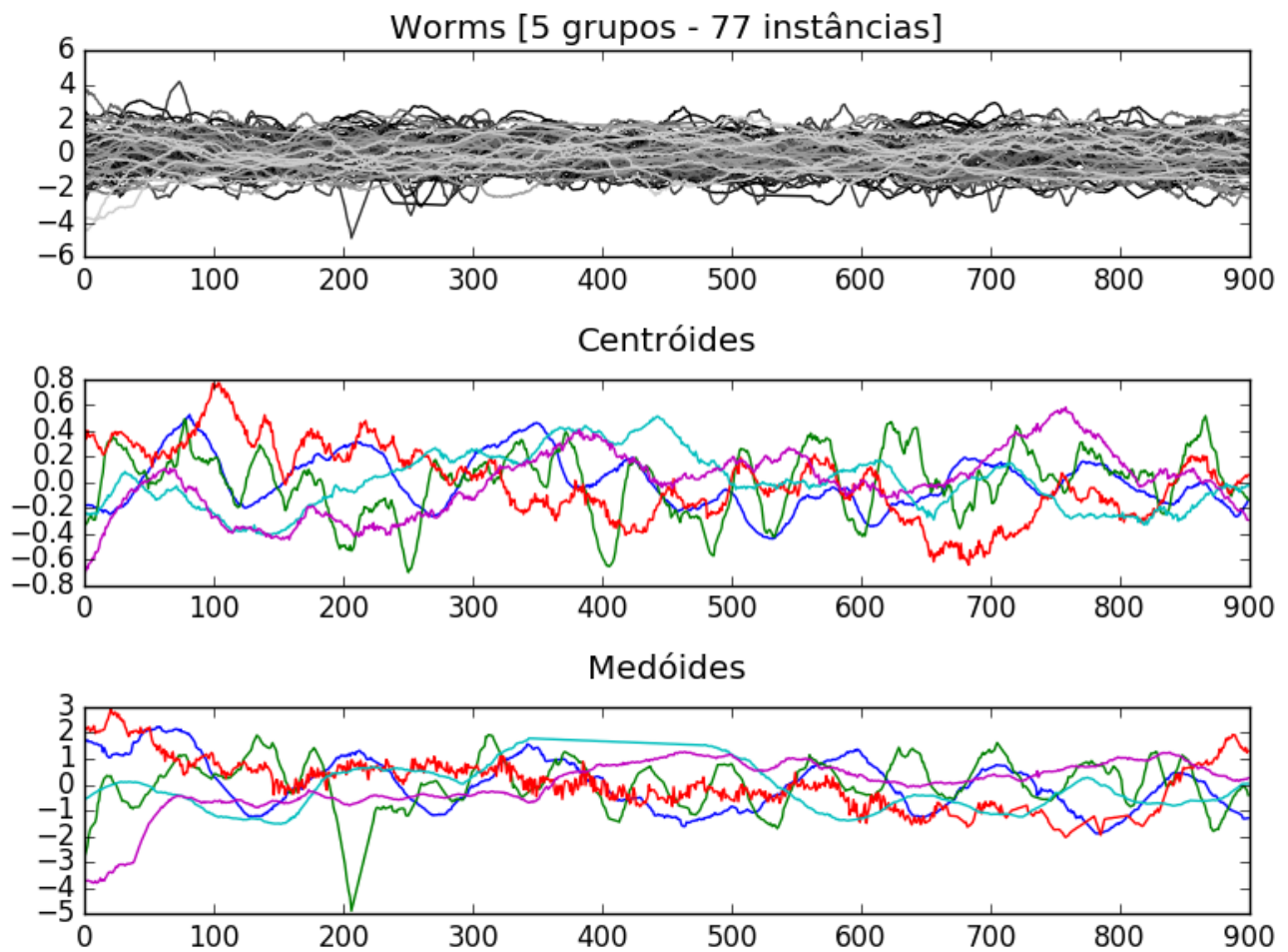


Figura A.78: Conjunto de dados 'Worms'.

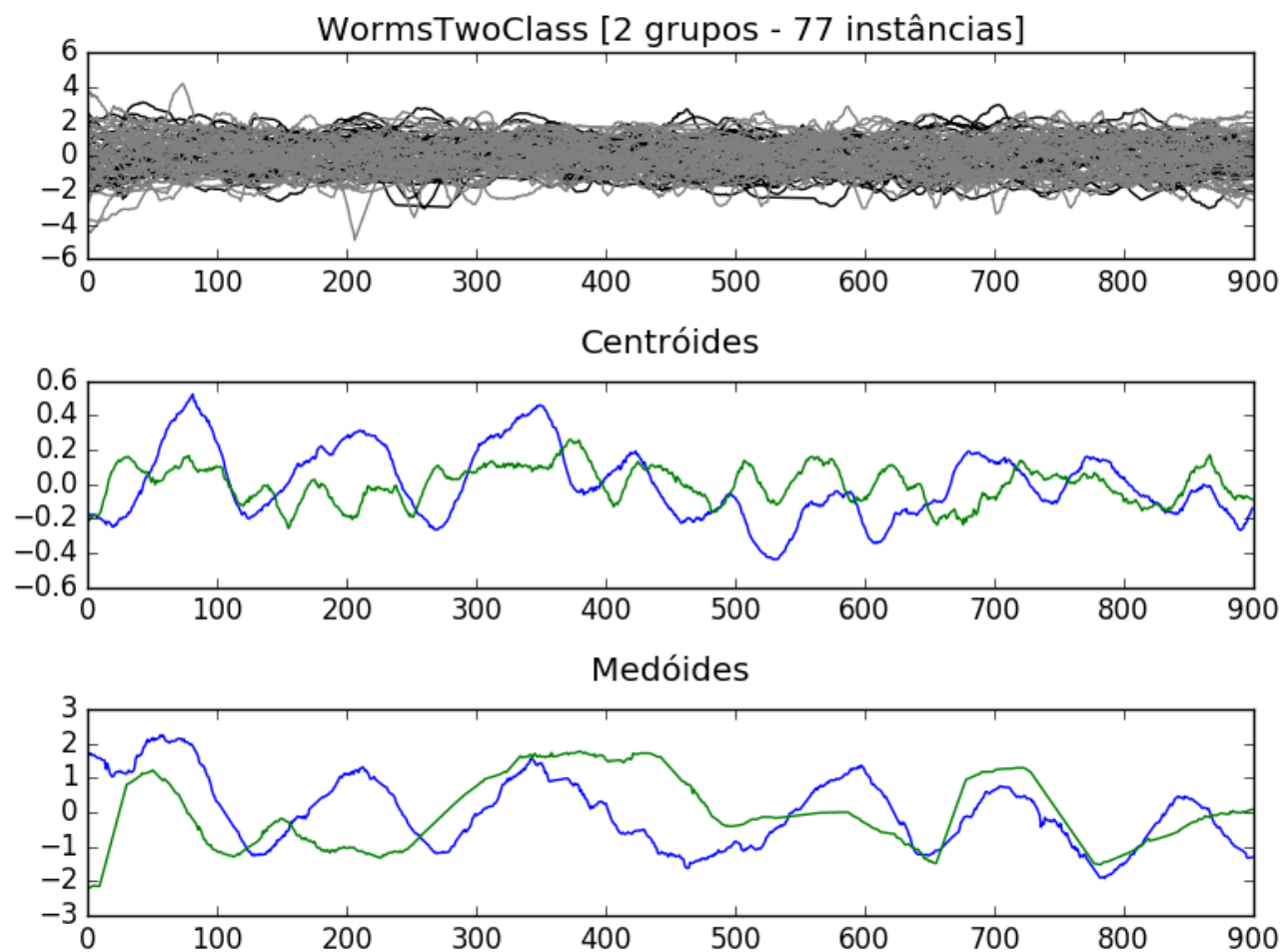


Figura A.79: Conjunto de dados 'WormsTwoClass'.

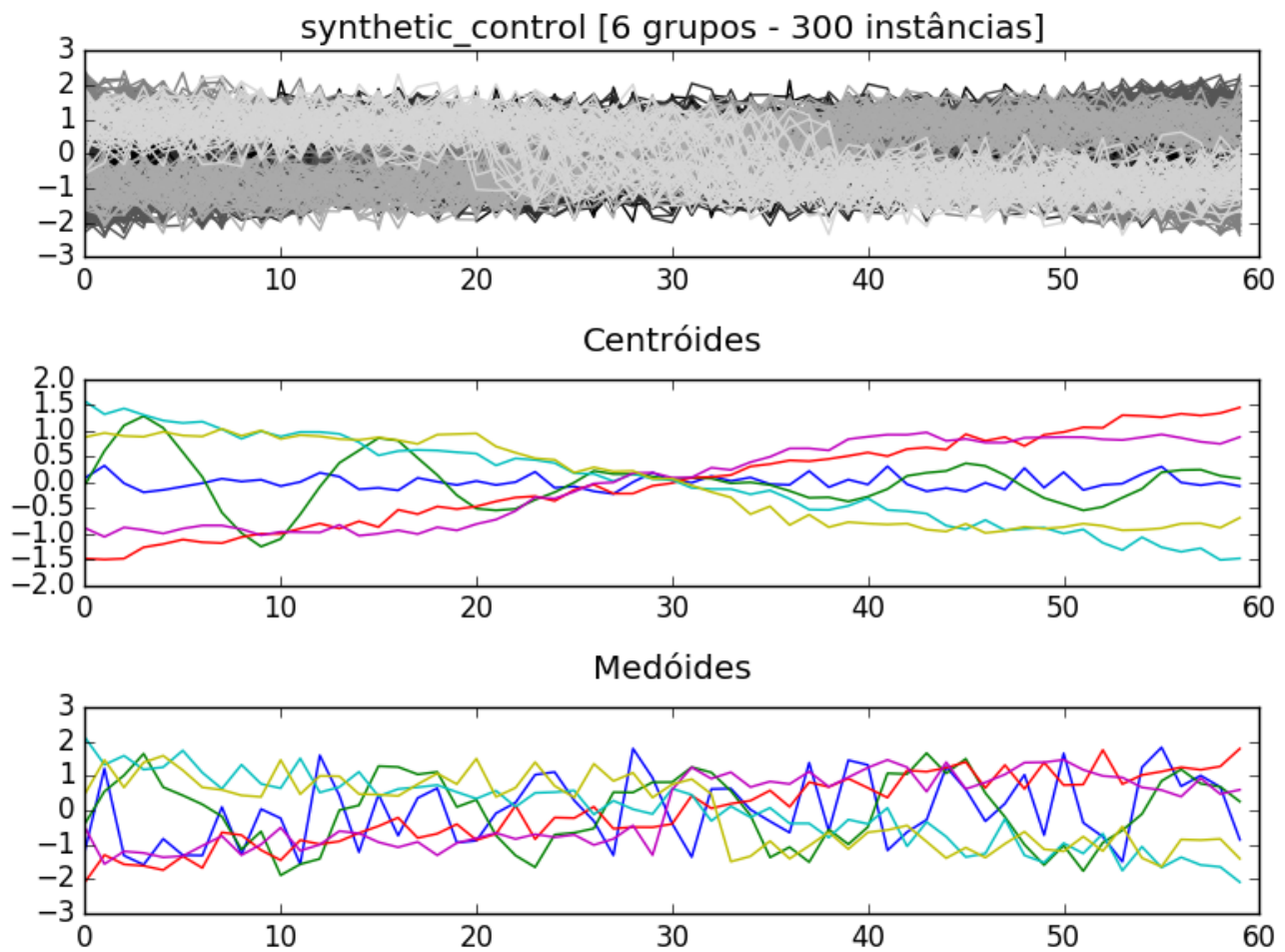


Figura A.80: Conjunto de dados 'synthetic_control'.

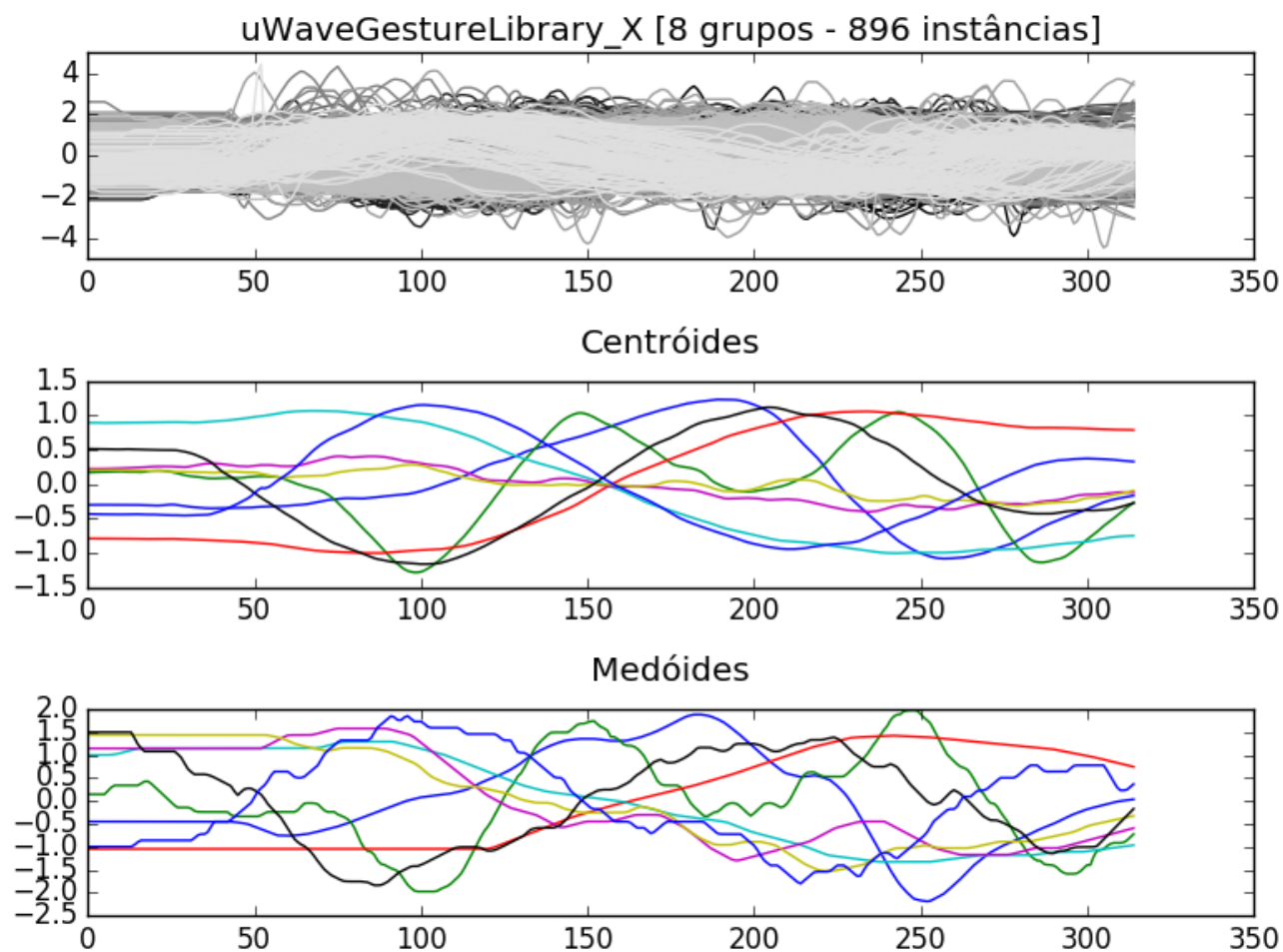


Figura A.81: Conjunto de dados 'uWaveGestureLibrary_X'.

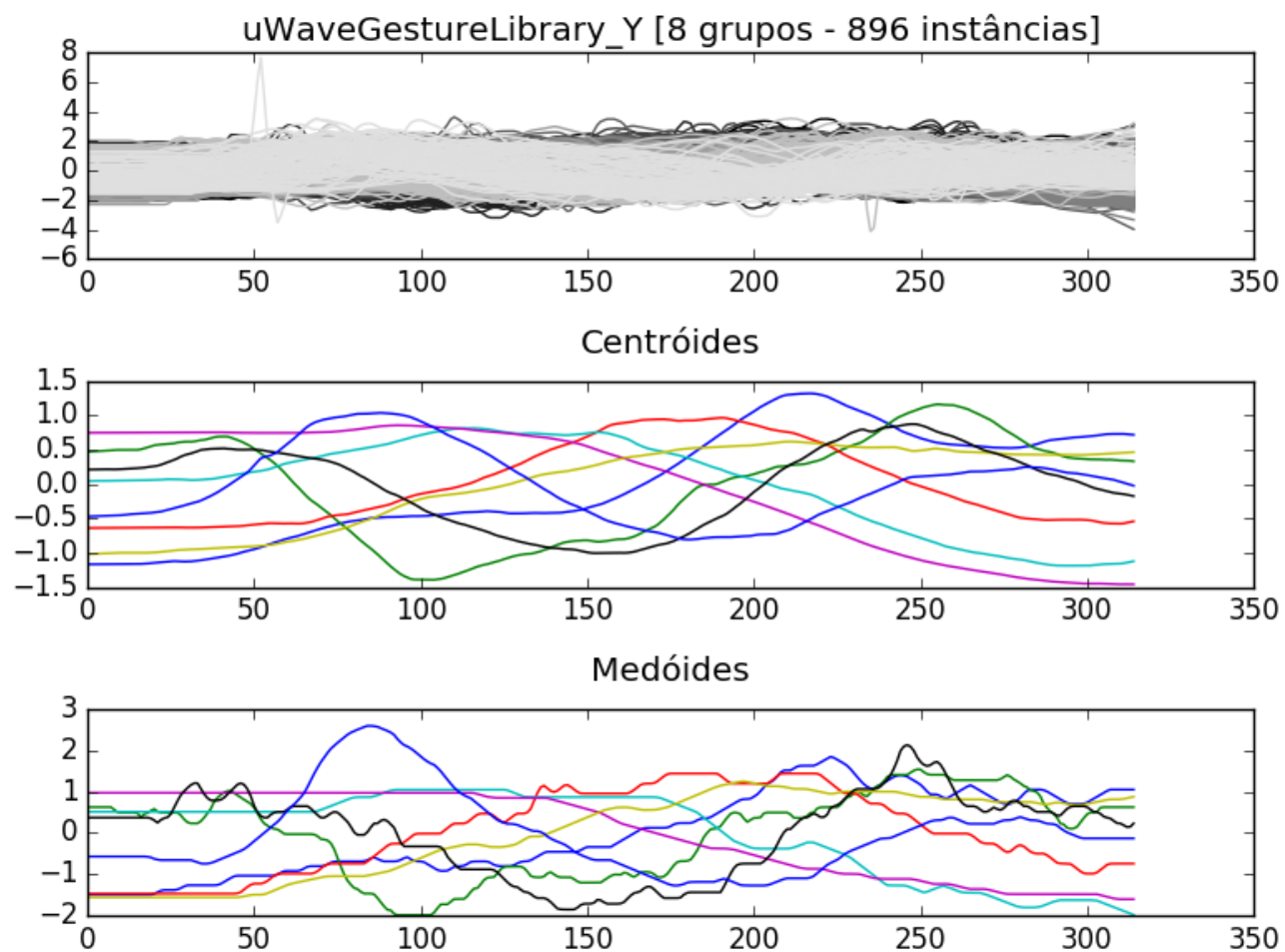


Figura A.82: Conjunto de dados 'uWaveGestureLibrary_Y'.

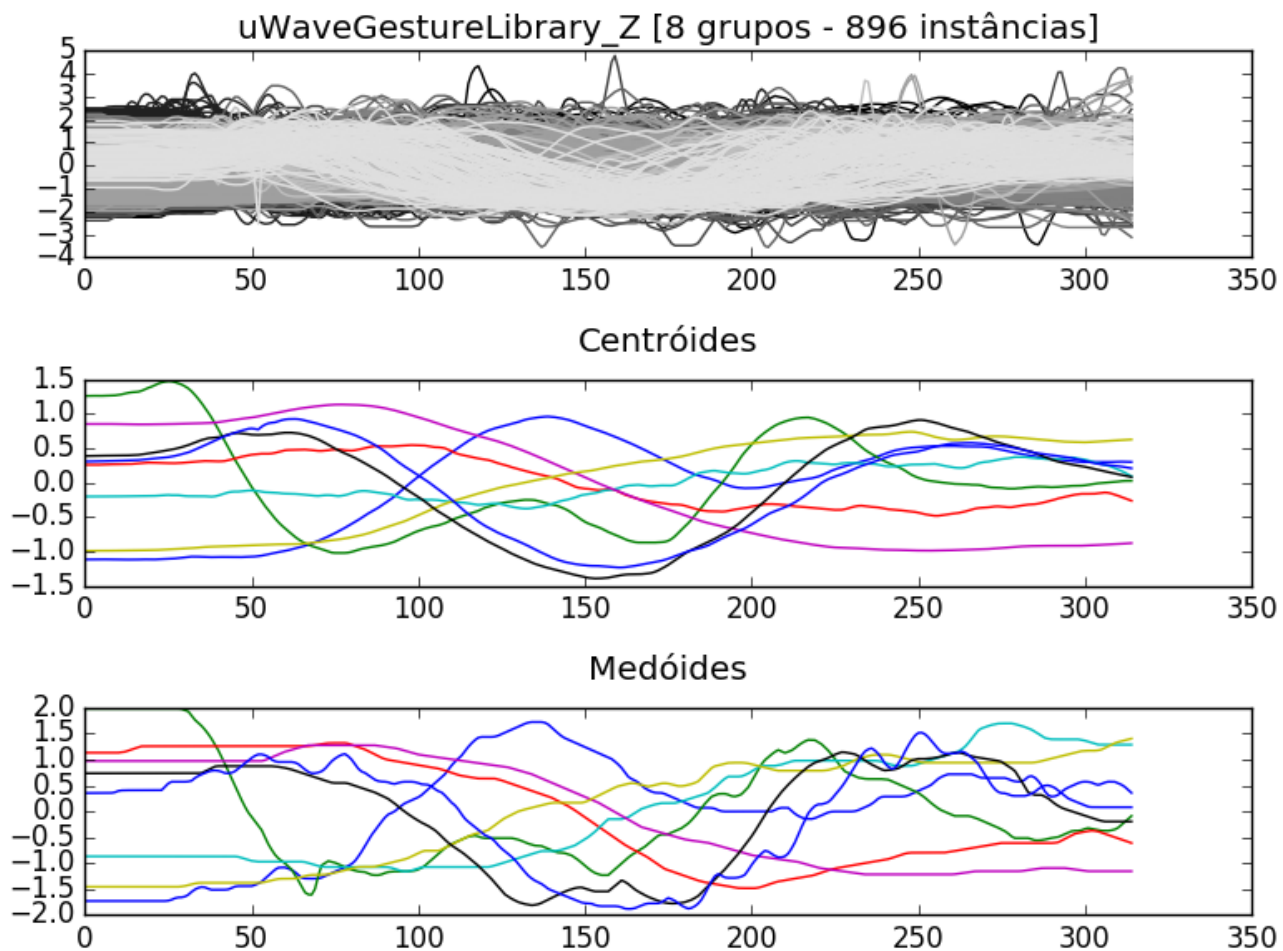


Figura A.83: Conjunto de dados 'uWaveGestureLibrary_Z'.

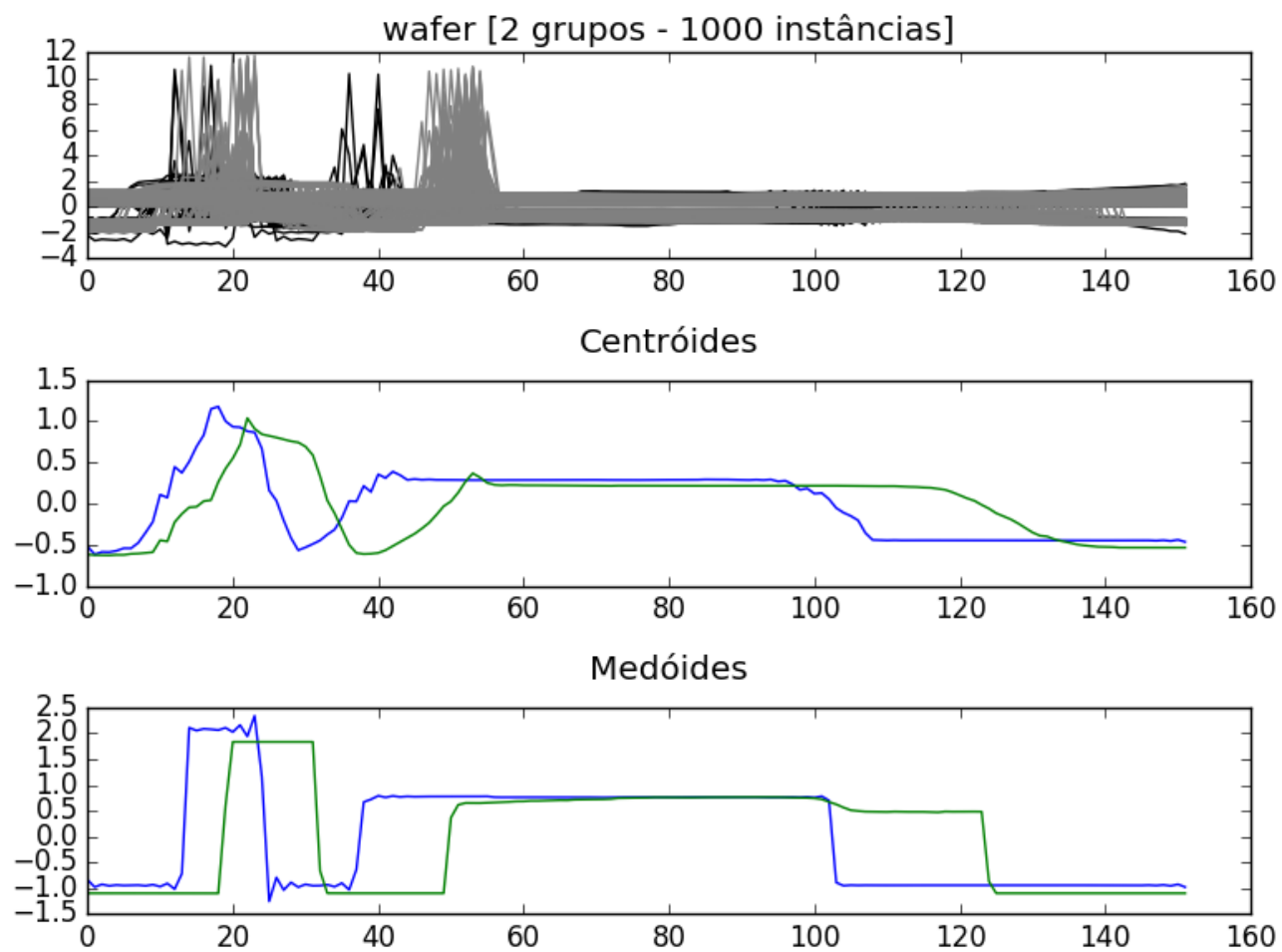


Figura A.84: Conjunto de dados 'wafer'.

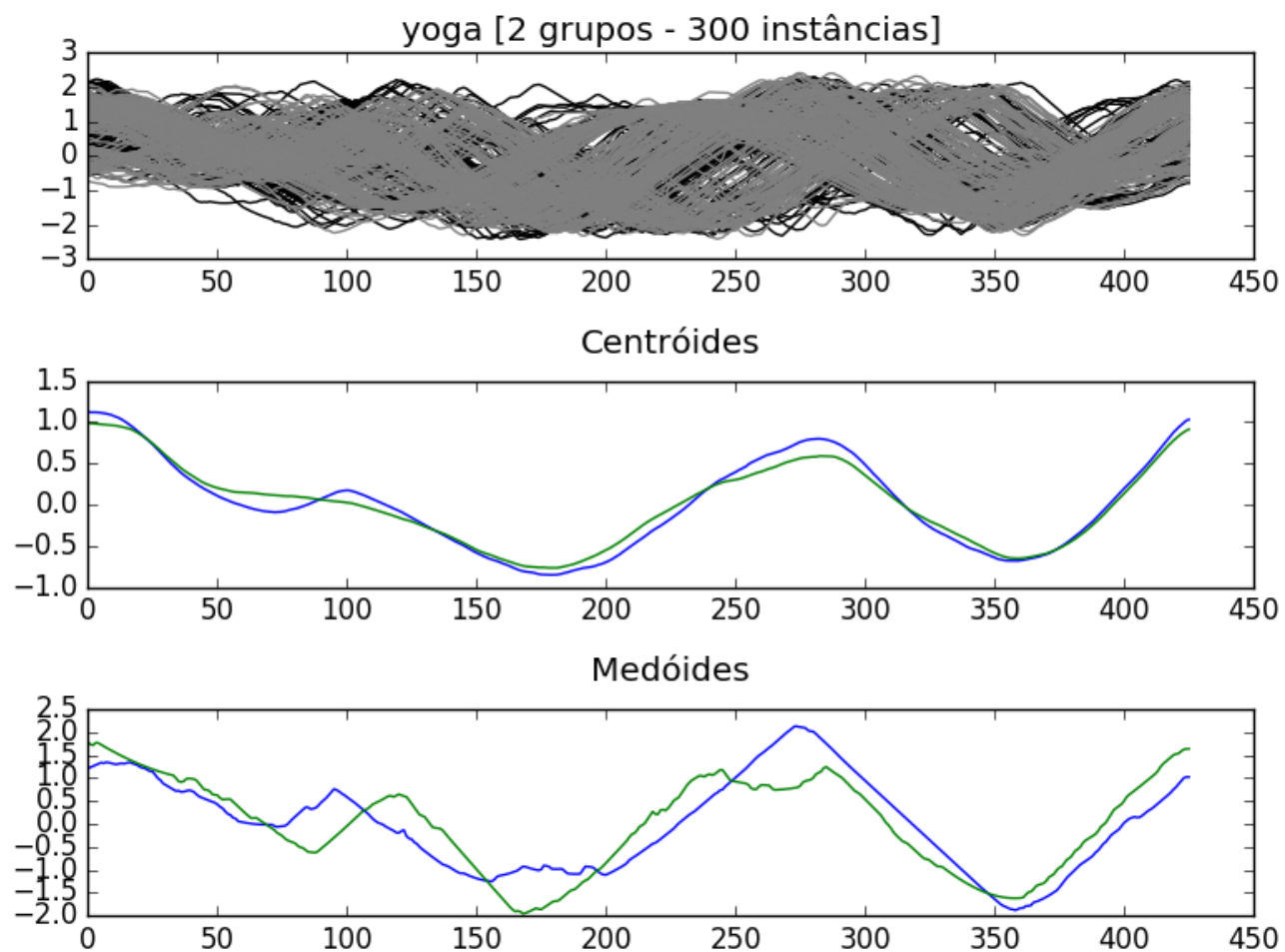


Figura A.85: Conjunto de dados 'yoga'.