Pontifícia Universidade Católica
do Rio de Janeiro

José Luiz Do Nascimento De Aguiar

Medidas de Similaridade entre Séries Temporais

DISSERTAÇÃO DE MESTRADO

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Informática do Departamento de Informática, do Centro Técnico Científico da PUC-Rio.

Orientador: Prof. Eduardo Sany Laber

Rio de Janeiro
Março 2016

Pontifícia Universidade Católica
do Rio de Janeiro

José Luiz Do Nascimento De Aguiar

Medidas de Similaridade entre Séries Temporais

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Informática do Departamento de Informática, do Centro Técnico Científico da PUC-Rio. Aprovada pela comissão Examinadora abaixo assinada.

Prof. Eduardo Sany Laber
Orientador
Departamento de Informática — PUC–Rio

Prof. Hélio Côrtes Vieira Lopes
Departamento de Informática — PUC-Rio

Prof. Ruy Luiz Milidiú
Departamento de Informática — PUC-Rio

Prof. Márcio da Silveira Carvalho
Coordenador Setorial do Centro Técnico Científico — PUC-Rio

Rio de Janeiro, 11 de Março de 2016

José Luiz Do Nascimento De Aguiar

Luiz Aguiar obteve sua graduação em Analise e Desenvolvimento de Sistemas pela UNESA - RJ (Rio De Janeiro, Brazil) em 2009. Após completar a graduação ocupou a posição de Gerente de TI na empresa Koch Ind Do Brasil. Também participou do grupo de pesquisa que desenvolveu uma aplicação de Machine Learning para a Petrobrás. enquanto cursava seu mestrado na PUC-Rio.

Ficha Catalogràfica

Aguiar, José Luiz Do Nascimento

Medidas de Similaridade entre Séries Temporais / José Luiz Do Nascimento De Aguiar ; orientador: Eduardo Sany Laber. — 2016.
75 f. : il. ; 30 cm

Dissertação (Mestrado em Informática)-Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2016.
Inclui bibliografia

1. Informática – Teses. 2. Aprendizagem de maquina;. 3. similaridade;. 4. metodos de medida de similaridade;. 5. series temporais.. I. Laber, Eduardo Sany. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Agradecimentos

Para
Lynsey, Pelo suporte e compreensão durante estes anos de mestrado.
Megan e Noah, por todas as risadas e carinho.
Minha Mãe e Irmãos, por entender os longos períodos de ausência.
Minha Avó, por ter feito tudo isso ser possível.
Eduardo Laber, meu orientador, pela orientação.
A todos os meus amigos do mestrado, pelos bons tempos e toda ajuda.

Resumo

Aguiar, José Luiz Do Nascimento; Laber, Eduardo Sany. **Medidas de Similaridade entre Séries Temporais**. Rio de Janeiro, 2016. 75p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Atualmente, uma tarefa muito importante na mineração de dados é compreender como extrair os dados mais informativos dentre um número muito grande de dados. Uma vez que todos os campos de conhecimento apresentam uma grande quantidade de dados que precisam ser reduzidas até as informações mais representativas, a abordagem das séries temporais é definitivamente um método muito forte para representar e extrair estas informações. No entanto nós precisamos ter uma ferramenta apropriada para inferir os dados mais significativos destas séries temporais, e para nos ajudar, podemos utilizar alguns métodos de medida de similaridade para saber o grau de igualdade entre duas séries temporais, e nesta pesquisa nós vamos realizar um estudo utilizando alguns métodos de similaridade baseados em medidas de distância e aplicar estes métodos em alguns algoritmos de clusterização para fazer uma avaliação de se existe uma combinação (método de similaridade baseado em distância / algoritmo de clusterização) que apresenta uma performance melhor em relação a todos os outros utilizados neste estudo, ou se existe um método de similaridade baseado em distância que mostra um desempenho melhor que os demais.

Palavras–chave

Aprendizagem de maquina; similaridade; metodos de medida de similaridade; series temporais.

Abstract

Aguiar, José Luiz Do Nascimento; Laber, Eduardo Sany (Advisor). **Time Series Symilarity Measures**. Rio de Janeiro, 2016. 75p. MSc. Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Nowadays a very important task in data mining is to understand how to collect the most informative data in a very amount of data. Once every single field of knowledge have lots of data to summarize in the most representative information, the time series approach is definitely a very strong way to represent and collect this information from it (12, 22). On other hand we need to have an appropriate tool to extract the most significant data from this time series. To help us we can use some similarity methods to know how similar is one time series from another In this work we will perform a research using some distance-based similarity methods and apply it in some clustering algorithms to do an assessment to see if there is a combination (distance-based similarity methods / clustering algorithm) that present a better performance in relation with all the others used in this work or if there exists one distance-based similarity method that shows a better performance between the others.

Keywords

Machine Learning; Similarity; Distance Measure Methods; Time Series.

Sumário

1

Introdução

Atualmente a crescente dependência do uso da internet e a necessidade de acesso imediato a informação, nos permite ter ao alcance dos nossos dedos uma massiva quantidade de dados disponíveis em todos os mais diferentes tipos de usos possíveis. Em meio a esta imensa quantidade de dados, uma tarefa extremamente importante é possuir a habilidade de poder distinguir o que é similar e o quão similar dois ou mais tipos de dados são entre si. Pensando mais seriamente sobre isto, podemos verificar que existem muitos casos onde estes dados não possuem um rótulo. Desta forma, em um caso onde exista um grande número de diferentes *coisas* que não possuem um rótulo, se decidirmos agrupar estas *coisas* de uma forma tal que as *coisas* similares sejam agrupadas em um conjunto e as outras estejam em outro conjunto distinto, esta tarefa necessariamente vai requerer um longo tempo e será custoso operacionalmente. Para executar este tipo de tarefa em um tempo hábil felizmente temos os algoritmos de clusterização para nos ajudar. Uma vez que os problemas de clusterização são os mais importantes no *aprendizado não supervisionado*, isto nos leva a pensar como achar uma *estrutura* em meio a uma coleção de dados que não possuem um rótulo. Uma forma de alcançar este objetivo é determinar o grau de similaridade apresentado em cada *estrutura* e calcular o quão similares são entre si. Uma forma mais simples de descrever o que é uma tarefa de *clusterização* é dizer que se trata do processo de organizar objetos em grupos nos quais os objetos pertencentes a cada grupo são similares entre si de alguma forma. Necessariamente, devemos manter em mente que um *Cluster* é melhor definido como uma coleção de objetos similares entre si e diferentes de outros objetos que pertençam a outros *clusters* como ilustrado na Figura 1.1.

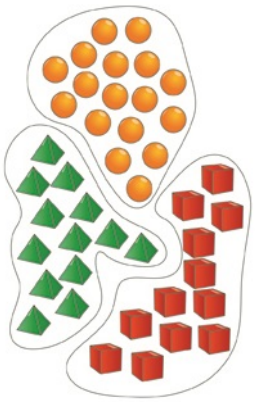


Figura 1.1: Clusters são separados de forma em que objetos similares estejam agrupados no mesmo conjunto segundo as suas formas geométricas

É comumente observado o uso de métodos de medidas de similaridade em algoritmos de clusterização para determinar o grau de similaridade entre dois objetos ou estruturas. Neste estudo utilizamos métodos de medidas de similaridades baseado na distância obtida uma vez que os dados são modelados como séries temporais. Para esta pesquisa, optamos por utilizar os seguintes métodos: *Distância Euclidiana (DE), Dynamic Time Warping (DTW) (22, 13, 11) e Complexity-Invariant distance (CID) (23)*. Nosso objetivo com este trabalho é tentar identificar se existe um método baseado em distância que seja superior aos demais utilizados neste estudo, e se existe uma combinação de algoritmo de clusterização e método de medida de similaridade baseado em distância que tenha um desempenho superior aos demais observados neste estudo. Para esta pesquisa estaremos aplicando a combinação *algoritmo de clusterização / método de medida de similaridade baseado em distância* nos datasets utilizados e que foram modelados como séries temporais, uma vez que é da natureza das séries temporais serem amplas e continuas, servindo, desta forma, adequadamente aos propósitos deste trabalho. Vale a pena mencionar nesta introdução, que uma série temporal é uma coleção de observações obtidas através do tempo e apresentada em um sistema de plano cartesiano que demonstra a variação obtida nos eixos tempo x variação ou nos eixos frequência x intensidade. Alguns exemplos de séries temporais que são observadas através do tempo, tipo tendências do mercado de ações ou resultados de eletrocardiograma são modelados nos eixos tempo x variação conforme a Figura 1.2 (10) ilustra.

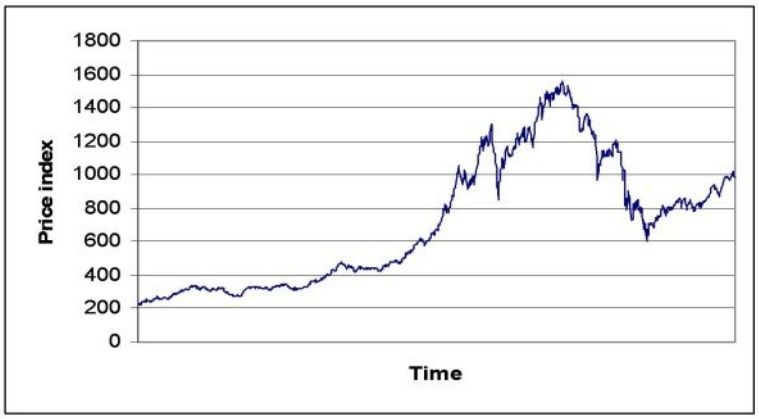


Figura 1.2: Índice de preços da bolsa do Mercado de ações da Holanda entre 1987 - 2005 - Um tipo de série temporal observada através do tempo

1.1 Nossos Resultados

Os resultados apresentados nesta seção, bem como a métrica utilizada para avaliar a tarefa de clusterização em relação às medidas de distância utilizadas nesta pesquisa, estão disponíveis na subseção *avaliação*. Na tabela 1.1 é possível observar os melhores resultados obtidos neste estudo segundo os parâmetros definidos para esta pesquisa.

1.1.1 Avaliação

Para a realização deste estudo, utilizamos o *Precision* e o *Recall* como ferramentas de avaliação dos resultados obtidos pelos algoritmos de clusterização. De posse destes resultados, obtivemos o *F-Measure*, que é a principal métrica utilizada neste trabalho. Maiores informações sobre as métricas de avaliação utilizadas nesta dissertação estão presentes na seção *Métricas de Avaliação para Algoritmos de Clusterização* (20, 24).

1.1.2 Resultados

Nossos resultados revelam que para cada tipo específico de tarefa cada uma das medidas de distância estudadas podem ser mais ou menos efetivas. Como é possível observar, os resultados obtidos não apontam para uma medida de distância em particular, podemos claramente afirmar que dentre as medidas e algoritmos utilizados neste estudo, não existe um método de medida de similaridade baseado em distância ou uma combinação de algoritmo de clusterização e método de medida de similaridade baseado em distância

que apresente uma performance melhor que os outros para todos os casos estudados nesta dissertação. A Tabela 1.1 apresenta os melhores resultados obtidos para cada dataset usado neste estudo. A tabela é composta pelo algoritmo usado, método de medida de similaridade utilizado que obteve o melhor resultado para o dataset relacionado (resultado obtido através do f-measure), nome do dataset, número de clusters encontrados, número real de clusters que o dataset contém, resultados do F-measure e tempo de execução. Os resultados demonstram que, como iremos perceber na seção sobre os datasets, que se o dataset estudado não apresenta uma variação significativa em sua complexidade relacionada a cada instância do dataset ou se não existe uma diferença significativa entre cada instância, a *Distância Euclidiana* apresentará uma performance melhor, devido a sua simplicidade, acurácia e tempo de execução. Uma vez que não existam as características mencionadas acima, tanto o DTW como o *CID* se reduzem a uma DE mais elaborada e custosa computacionalmente. Porém, se o dataset não esta completo (ex: valores faltando) ou as instâncias tem tamanhos diferentes, claramente o DTW terá uma imensa vantagem sobre a *distância euclidiana* uma vez que esta não é capaz de resolver o problema de dimensionalidade entre as duas séries temporais, uma vez que um exemplo irá apresentar um tamanho maior ou menor do que o outro que seja comparado. Da mesma maneira, se existe uma diferença significativa na complexidade de cada uma das instâncias ou se o dataset é muito desbalanceado, o *CID* apresentará uma performance superior perante os outros métodos de medida utilizados neste estudo.

Outro fato interessante a ser observado é que se o dataset é muito desbalanceado, os algoritmos usados neste estudo tendem a agrupar apenas o maior conjunto, juntamente com os demais, garantindo assim um alto número de verdadeiros positivos, em relação ao baixo número de falsos positivos devido ao fato de o dataset ser demasiadamente desbalanceado, perdendo assim a acurácia em relação ao número de clusters presentes no dataset. Em contrapartida, quando a combinação algoritmo de clusterização / Método de medida de similaridade baseado em distância encontra um melhor resultado referente ao número de clusters presente no dataset, isto sempre apresenta um declínio no resultado referente a quantidade de objetos agrupados adequadamente. talvez isto ocorra devido ao fato de que existam mais opções de grupos com objetos que estejam na interseção entre dois ou mais grupos. Devido à necessidade deste equilíbrio entre quantidade de clusters encontrados e número de objetos agrupados adequadamente, podemos realizar uma análise posterior levando em conta os resultados que melhor demonstram este equilíbrio.

Todas estas analises podem ser encontradas na seção análises e maiores in-

Algoritmo	Distância	Dataset	Total de pares	Precision	Recall	F-Measure	Tempo em seg
Xmeans	DE	flare	567.645	0.659	0.392	0.58	0.03
Xmeans	DTW	eeg	112.177.731	0.659	0.56	0.637	3.28
K means	CID	Arritmia	101.926	0.63	0.63	0.63	0.02
MDBC	DTW	Adult	530.093.080	0.69	0.363	0.585	0.02
Xmeans	CID	F Gen	51.040.356	0.509	0.25	0.422	0.13
Xmeans	DTW	G phase	48.733.128	0.584	0.27	0.474	279.14
Xmeans	DTW	Otto	1.913.917.515	0.701	0.658	0.692	374.19
MDBC	DE	Dw Jones	280.875	0.943	0.399	0.741	0.04
Xmeans	CID	HV	33.903.495	0.278	0.218	0.263	0.02
MDBC	DTW	Quake	2.370.753	0.538	0.488	0.527	0.04
Kmeans	DTW	Sido	80.359.503	0.817	0.917	0.835	120.09

Tabela 1.1: Melhores resultados

formações sobre os datasets estão disponíveis no apêndice desta dissertação.

Vale ressaltar que o F-measure é uma métrica apropriada para a avaliação deste trabalho, uma vez que estamos utilizando o parâmetro β como um invariante do número de clusters existentes nos datasets utilizados, Desta forma obtendo um resultado justo (14).

A tabela 1.2 apresenta a distribuição dos objetos nos clusters.

Algoritmo	Distância	Dataset	TP	FP	FN	TN
Xmeans	DE	flare	112015	57900	173700	224030
Xmeans	DTW	eeg	37765755	19517249	29717558	25177169
K means	CID	Arritmia	32112	18851	18851	32112
MDBC	DTW	Adult	73588839	33111325	129037561	294355355
Xmeans	CID	F Gen	7901047	7610776	23676964	11851570
Xmeans	DTW	G phase	8237802	5873396	22265227	12356703
Xmeans	DTW	Otto	603917532	257358995	314519560	738121429
MDBC	DE	Dw Jones	53290	3209	80297	144079
Xmeans	CID	HV	3744979	9746352	13457201	6954962
MDBC	DTW	Quake	552721	474638	580112	763282
Kmeans	DTW	Sido	2561700	573293	230303	76994207

Tabela 1.2: TP = Objetos que deveriam estar juntos e estão, FP = Objetos que deveriam estar separados e não estão, TN = Objetos que deveriam estar separados e estão e FN Objetos que não deveriam estar separados e estão

1.2 Organização da Tese

Esta dissertação foi estruturada com o objetivo de apresentar os resultados e técnicas usadas de forma fácil e inteligível, bem como uma visão geral dos algoritmos usados, os métodos de medida de similaridade baseados em distância e também apresentamos as séries temporais usadas neste estudo. Com isto em mente, apresentaremos os conceitos básicos, na seção de mesmo nome, onde introduziremos uma apresentação mais formal das técnicas utilizadas, bem como sua importância para o aprendizado não supervisionado. O Capítulo dois é dedicado as séries temporais, o que é e qual a sua importância para os problemas de clusterização . Uma explicação de como as combinações algoritmo de clusterização / métodos de medida de similaridade baseados em distância foram formados, e nos apresenta como estas duas importantes ferramentas trabalham juntas. Os datasets serão discutidos no Capítulo quatro. Alguns dos trabalhos relacionados estão disponíveis para consulta no Capítulo cinco. O Capítulo seis é dedicado a apresentar os resultados obtidos experimentalmente, segundo os parâmetros utilizados nesta dissertação. As conclusões e os trabalhos futuros podem ser encontrados no Capítulo sete desta dissertação.

2

Conceitos Básicos

Os conceitos básicos que se fazem necessários para o bom entendimento desta dissertação serão apresentados neste capítulo. As definições do que são algoritmos de clusterização, Métodos de medida de similaridade baseados em distância, séries temporais e como as combinações destas técnicas se fazem importantes para as tarefas de clusterização, serão examinadas para que se possa ter um melhor entendimento deste estudo.

2.1 Séries Temporais

Uma *série temporal* é qualquer observação feita através do tempo e também ordenada pelo tempo em que cada observação foi efetuada, o que significa que uma série temporal é uma sequência de dados que tiveram suas medidas efetuadas durante um intervalo de tempo. Sua onipresença em quase todos os campos do conhecimento produziu um grande interesse no campo da mineração de dados na última década. Devido a sua fácil obtenção as séries temporais constituem uma classe muito importante dos objetos temporais (10). Existem inúmeros exemplos de séries temporais como pode ser observado pelos exemplos abaixo:

(i) **Temperatura diária da cidade do Rio De Janeiro**

(ii) **Média anual das manchas solares**

(iii) **Registro das marés da Baia de São Francisco**

Os exemplos (i) e (ii) apresentam séries temporais discretas, ou seja, exemplos obtidos a partir de amostras de uma série temporal continua, como a presente no exemplo (iii).
A Figura 2.1 Ilustra como uma série temporal se apresenta.

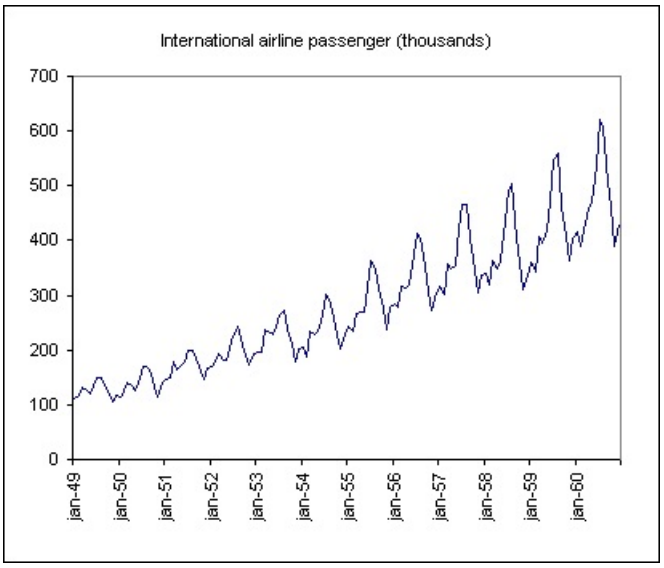


Figura 2.1: Série temporal que representa o número de passageiros de uma companhia aérea entre os anos de 1949 - 1960.

Em uma série temporal multivariada, cada observação é um ponto em R^p e $p \geq 2$ em vez de um escalar. A Figura 2.2 demonstra um exemplo de uma série temporal multivariada.

person	year	income	age	sex
1	2001	1300	27	1
1	2002	1600	28	1
1	2003	2000	29	1
2	2001	2000	38	2
2	2002	2300	39	2
2	2003	2400	40	2

Figura 2.2: Exemplo de uma série temporal multivariada, onde cada ponto é composto de mais do que duas informações.

Nesta dissertação estamos usando 10 datasets de séries temporais multivariadas, que, por sua vez, estão descritos na seção datasets.
É importante mencionar que uma vez dada uma série temporal T, seu tamanho será dado pelo seu número de pontos em T. Também, uma série temporal apresenta marcações de tempo, estas marcações geralmente apresentam diferentes números de aspectos, como por exemplo, um número diferente de estados (um estado é um valor observado em um determinado tempo específico), o número de transições (por exemplo as mudanas de direções), e a distribuição da duração de cada evento (quanto tempo dura um evento) através de cada estado. Cada um destes aspectos contribui para agregar uma propriedade específica a cada série temporal, e esta propriedade é denominada *complexidade*

(26). Podemos ilustrar a complexidade existente em cada instância através do exemplo de formas geométricas que foram modeladas como séries temporais conforme a Figura 2.3 demonstra. Para obtermos estes modelos, simplesmente calculamos a distância do ponto central de cada figura até o seu contorno, desta maneira podemos representar uma observação, que não ocorre no tempo, como uma série temporal utilizando a intensidade e a frequência obtida com o cálculo do contorno.

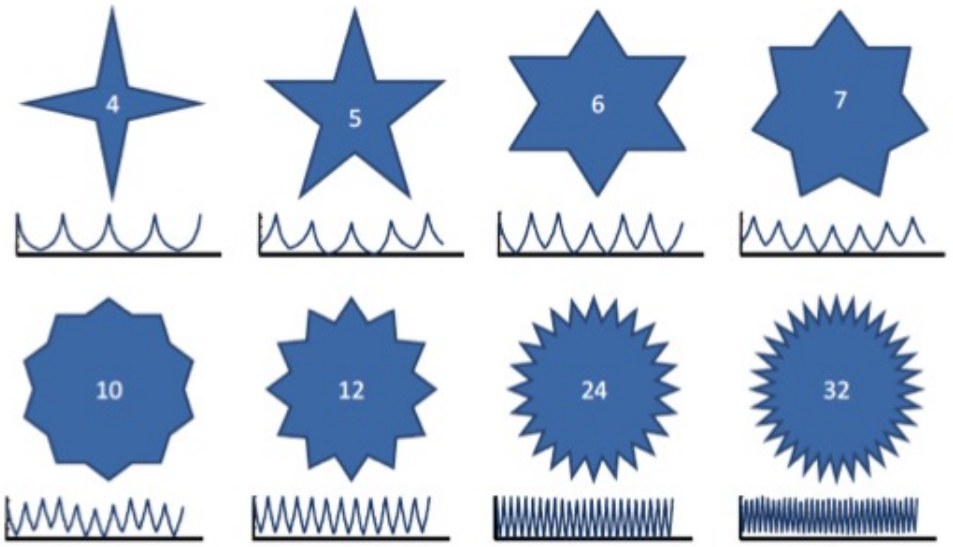


Figura 2.3: Figuras geométricas que foram modeladas como uma série temporal - este é um exemplo de série temporal que não ocorre durante um intervalo de tempo

Conforme a Figura 2.3 demonstra, existe uma grande variação nas complexidades obtidas entre a imagem onde a figura apresenta 4 pontas e a que apresenta 32 pontas. Desta forma podemos ver claramente que a figura que apresenta 4 pontas tem uma maior *similaridade* com a figura que apresenta 5 pontas, do que com a figura que apresenta 32 pontas. Como é possível ver pelas imagens, a complexidade de cada instância se mantém quando modelamos o exemplos em uma série temporal.

2.2 Domínio das séries estudadas

- **Flare dataset** Científico
- **eeg dataset** Médico
- **Arritmia dataset** Médico

- **Adult census dataset** Ciências Sociais
- **Final General dataset** Social
- **Gesture phase dataset** Físico e Médico
- **Otto group dataset** Comercial
- **Dow Jones dataset** Financeiro
- **HV dataset** Comercial
- **Quake dataset** Científico
- **Sido dataset** Médico

2.3 Métodos de Similaridade Baseados em Distância

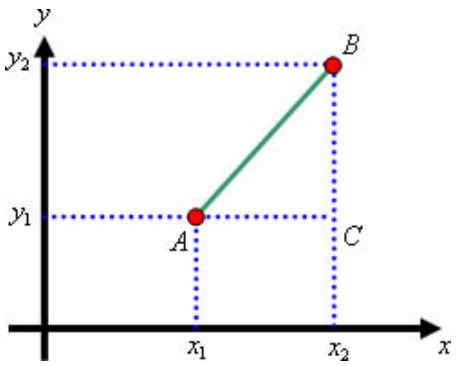
O problema de clusterização é um dos, se não o, mais importantes no campo de aprendizagem de máquina, mais específicamente em aprendizagem não supervisionada. O ato de organizar objetos similares em *grupos* onde todos os objetos são similares entre si é chamado de *clusterização*. Devido a sua importância, existem diversos métodos para definir o quanto um objeto é similar a outro objeto, e quais objetos dentro de um grupo são similares entre si. Nesta dissertação iremos pesquisar os métodos de medida de similaridade baseados no cálculo das distâncias destes objetos, uma vez devidamente modelados como séries temporais. Escolhemos estes métodos para este estudo, uma vez que o objetivo deste estudo é pesquisar a relação entre distância e similaridade, ou dissimilaridade.

Uma vez que podemos representar cada série temporal como um ponto multidimensional em um espaço euclidiano, podemos tirar vantagem deste fato e usar métodos baseados em medidas de distância para saber o quanto distante ou perto duas séries temporais estão entre si. Quanto mais perto duas séries temporais se encontrarem no espaço euclidiano, mais similares elas devem ser, de outra forma, quanto mais distante estiverem, elas devem apresentar um menor grau de similaridade entre si. Porém, em alguns casos, alguns métodos tendem a *clusterizar* uma série temporal extremamente simples (levando-se em conta a complexidade) com outra série extremamente complexa. Isto ocorre uma vez que a distância entre uma série temporal pequena, porém significativamente complexa é extremamente similar a de uma série temporal simples, porém longa. Devido a este tipo de comportamento apresentado por alguns métodos (podemos citar como exemplo a distância euclidiana), faz se

necessário o uso de outros métodos para que se possa obter um resultado com uma melhor definição e mais próximo da realidade.
Na próxima seção, introduziremos uma pequena visão geral dos métodos de medida de similaridade baseado em distância utilizados neste estudo e discutiremos seu uso. (9).

2.3.1 Distância Euclidiana (DE)

A *Distância Euclidiana* é a mais básica, e talvez a mais importante, medida de distância existente, dizemos que talvez seja a mais importante uma vez que existem vários outros métodos, mais sofisticados, de medir distâncias e que são uma derivação da distância euclidiana. Basicamente a DE nos retorna a distância entre dois pontos (A, B), onde A é composto das coordenadas x_1 e y_1 e B pelas coordenadas x_2 e y_2. A Figura 2.4 exemplifica o cálculo da DE.



$$DE(A,B) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2-1}$$

Figura 2.4: Distância euclidiana entre dois pontos bidimensionais.

De uma forma mais geral, digamos que seja $T = (t_1, t_2, ..., t_n)$ e $S = (s_1, s_2, ..., s_n)$, podemos calcular a distância euclidiana entre T e S como o somatório de todos os pontos que compõem T e S.

$$DE(T,S) = \sqrt{\sum_{i=1}^{n}(t_i - s_i)^2} \tag{2-2}$$

A Figura 2.5 nos mostra que cada ponto de uma série temporal T tem sua distância calculada com apenas um ponto da série temporal S e vice versa, mantendo assim uma relação de 1 para 1 não havendo a possibilidade de um

ponto ter a sua distância calculada com mais de um ponto da outra série temporal.

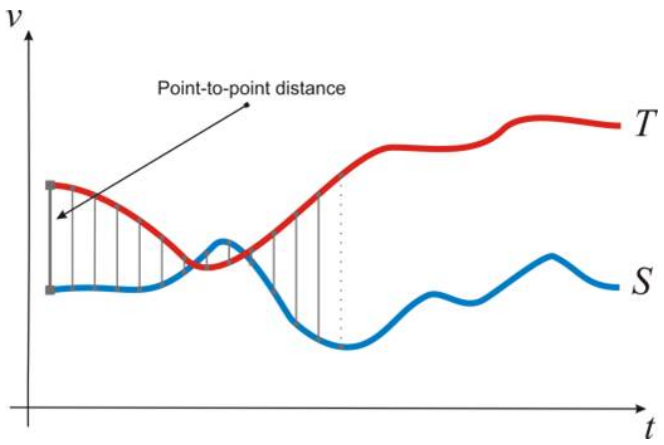


Figura 2.5: Séries temporais T e S, para cada ponto de T, o cálculo da distância é realizado apenas uma única vez para apenas um único ponto em S.

Conforme podemos intuir, para séries temporais que apresentam o mesmo tamanho, a DE vai proporcionar um bom resultado final. Porém em casos que as séries temporais apresentem tamanhos diferentes a *distância euclidiana* vai ignorar os pontos restantes da série temporal de maior tamanho, conforme ilustrado na Figura 2.8, e desta forma não irá retornar um resultado confiável.

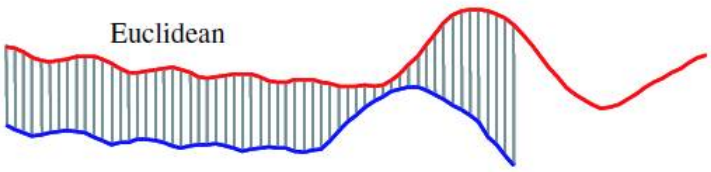


Figura 2.6: DE não é a melhor alternativa para séries temporais de tamanhos diferentes.

Uma vez que a DE não é a mais apropriada para este tipo de problema, devemos utilizar um outro método de medida de similaridade baseado em distância para podermos comparar com os resultados obtidos, sob os mesmo parâmetros, com a DE e um destes métodos é o DTW *Dynamic Time Warping*.

2.3.2 Dynamic Time Warping (DTW)

Quando duas séries temporais apresentam tamanhos diferentes (Figura 2.7), a DE não é o método mais apropriado para se usar nestes casos, uma vez

que a DE ignora os pontos da série temporal de maior tamanho que excedem os pontos da série temporal de menor tamanho (Figura 2.8).

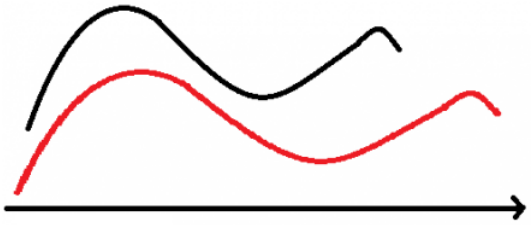


Figura 2.7: Séries temporais com tamanhos diferentes.

Para resolver este problema, o *Dynamic Time Warping (DTW)* nos apresenta uma abordagem interessante.
Basicamente o DTW vai calcular todas as distâncias entre todos os pontos de cada uma das séries apresentadas, garantindo assim que todos os pontos estejam ligados a pelo menos 1 ponto da outra série apresentada e vice versa. De outra forma teremos que, sejam X e Y duas séries temporais onde o objetivo do DTW é comparar duas séries temporais, desta forma temos que:

$$X = (x_1, x_2, ..., x_N) \text{ de tamanho } N \in \mathbb{N} \text{ e } Y = (y_1, y_2, ..., y_M) \text{ de tamanho } M \in \mathbb{N} \quad (2\text{-}3)$$

Neste ponto, faz-se necessária uma pequena explicação sobre *alinhamento*: *Alinhamento* é o processo de organizar objetos em uma linha reta ou de uma forma apropriada considerando-se as posições relativas de dois ou mais objetos. Uma vez que nesta dissertação estamos utilizando a comparação entre pontos de uma série temporal X e os pontos de uma série temporal Y de tal forma que todos os pontos estejam ligados a pelo menos um ponto da outra série.
O alinhamento global de sequência é o tipo de alinhamento utilizado pelo DTW uma vez que cada ponto de uma série temporal se apresenta de forma sequencial e é necessário que todo o comprimento de todas as séries sejam cobertos pelo alinhamento. O DTW utiliza uma matriz de custo, sendo que uma matriz de custo nada mais é do que uma matriz preenchida com todos os valores calculados entre cada ponto das duas séries temporais em questão (Figura 2.9). O objetivo de preencher esta matriz é encontrar um alinhamento entre X e Y que tenha o menor custo, ou seja, apresente um somatório que tenha o menor valor possível. Para podermos obter o alinhamento que apresente o somatório de menor valor, utilizamos *programação Dinâmica* (5, 16) (18), A Figura 2.10 ilustra o alinhamento de valor mínimo obtido pela matriz de custo.

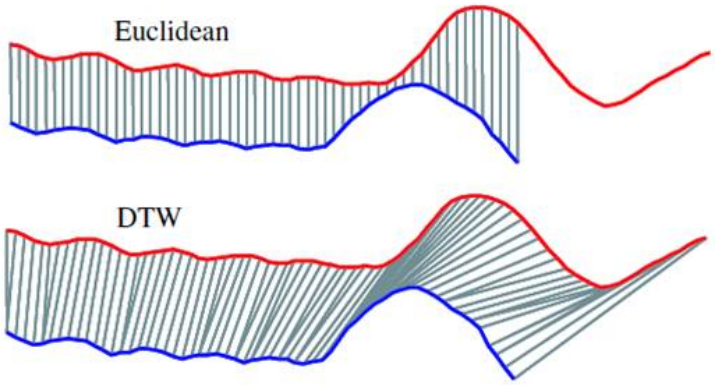


Figura 2.8: Diferença de alinhamento entre a DE e o DTW.

O DTW resolve o problema de dimensionamento suprimindo o espaço existente entre duas séries temporais de tamanhos diferentes. Para isto o DTW vai realizar o cálculo entre todos os pontos de cada uma das séries temporais em uma relação de m para n onde m e n são os tamanhos de cada uma das séries. Exemplificando, podemos dizer que o ponto x_1 de uma série temporal X terá sua de *distância euclidiana* calculada para cada ponto da série Y, exemplo $\{y_1, y_2, ..., y_M\}$, uma vez concluído o processo para o ponto x_1 o mesmo processo será repetido para os N pontos de X.

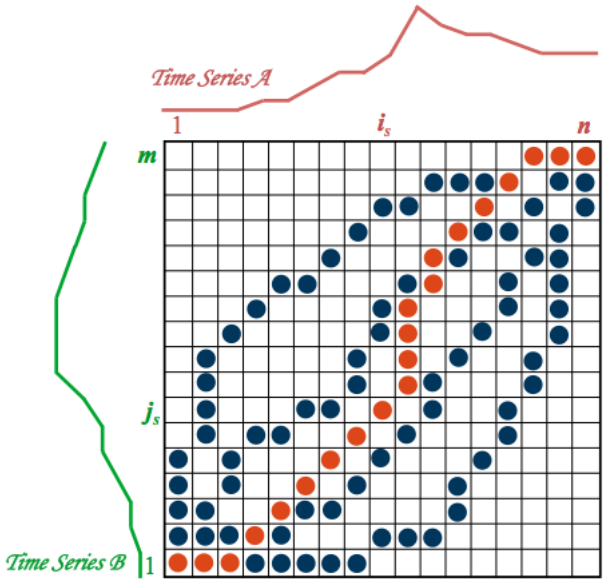


Figura 2.9: Ilustração de uma Matriz de custo

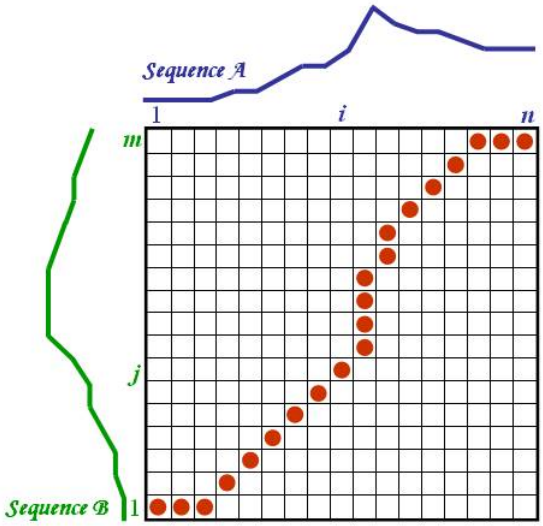


Figura 2.10: Custo mínimo obtido por programação dinâmica aplicada nos resultados da Matriz de custo da Figura 2.9

Também é necessária a aplicação de algumas restrições para que exista um alinhamento efetivo:
Monotocidade: sejam $i_{s-1} \leq i_s$ e $j_{s-1} \leq j_s$. Não é permitido o retrocesso no índice tempo (Figura 2.11).

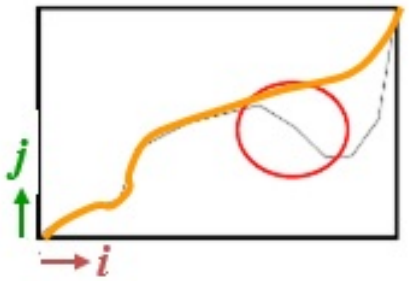


Figura 2.11: Monotocidade: a linha preta demonstra um alinhamento inválido enquanto a laranja o alinhamento válido

A monotocidade vai garantir que não existam cruzamentos no alinhamento (Figura 2.12)

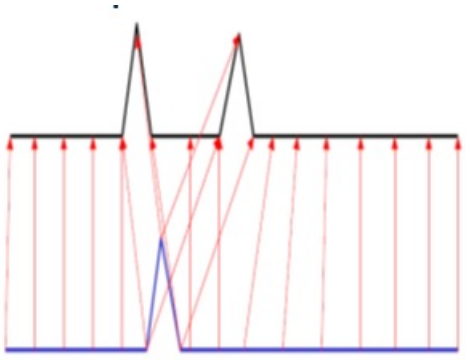


Figura 2.12: A monotocidade não permite o cruzamento durante o alinhamento

Continuidade: Sejam $i_s - i_{s-1} \leq 1$ e $j_s - j_{s-1} \leq 1$ Veremos que não é permitido que exista falha de tempo no alinhamento (Figura 2.13).

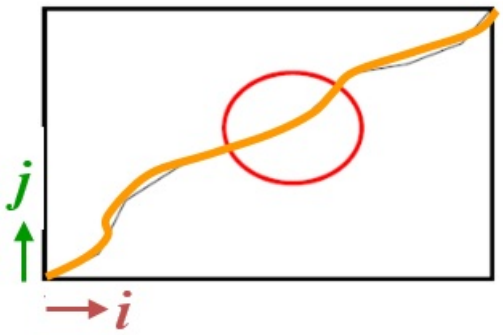


Figura 2.13: A continuidade não permite a falha temporal durante o alinhamento

A continuidade não permite que características importantes da série não seja omitida durante o alinhamento(Figura 2.14).

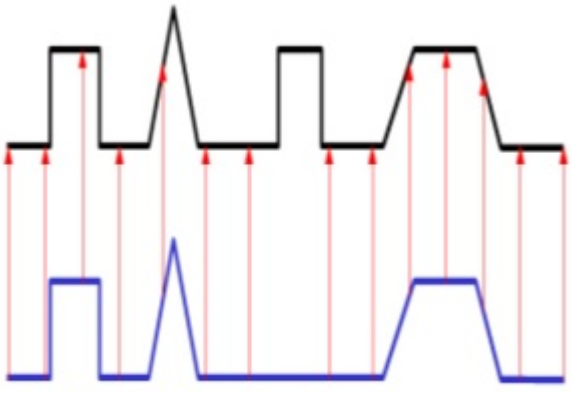


Figura 2.14

É fácil perceber que serão necessários realizar os cálculos $N.M$ vezes para preencher a matriz de custo. Isto se apresenta como um problema, uma vez que para séries longas, será demasiadamente custoso realizar todas estas operações. Para resolver este problema, utilizaremos limites mínimos para prevenir todas estas operações, uma vez que a maioria delas pode ser descartada pois haverá um alinhamento que apresentará um custo menor e necessariamente devemos prevenir que cálculos de pontos cruzados não existam, (Ex: tentar conectar o ponto x_3 com o ponto y_1 uma vez que os pontos x_1 e x_2 apresentam o melhor alinhamento entre os pontos y_1 e y_2). Para atingir este objetivo, utilizaremos uma técnica chamada faixa de Sakoe-Chiba que por sua vez irá assegurar um bom limite para o preenchimento da matriz de custo. Vale a pena mencionar que existem outras técnicas para garantir um limite inferior para a matriz de custo, e uma bastante utilizada é o paralelogramo de Itakura.

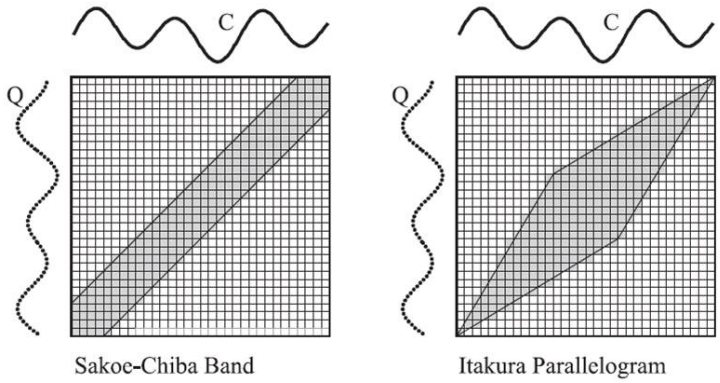


Figura 2.15: Faixa de Sakoe-Chiba e o paralelogramo de Itakura. As duas técnicas mais usadas para garantir um limite inferior. Apenas as áreas escurecidas serão preenchidas.

Também existem algumas restrições pertinentes as funções, sendo elas: *Limitantes:* Sejam $i_1 = 1$, $i_k = n$ e $j_1 = 1$, $j_k = m$, o alinhamento se inicia na célula (1,1) e terminará na célula (m,n)(Figura 2.16)

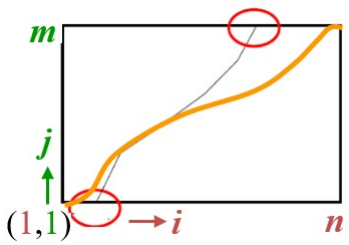


Figura 2.16: O alinhamento se inicia na celula (1,1) e termina na celula (m,n)

Estes limitantes garante que a função não considere apenas parcialmente uma das séries (Figura 2.17)

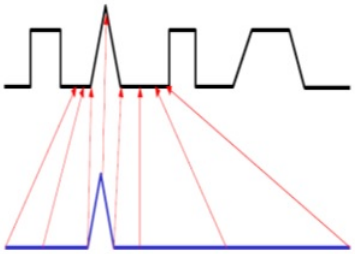


Figura 2.17: Garante que o primeiro ponto de ambas as séries estejam alinhados, assim como o últimos pontos de cada série também estejam alinhados, a figura 2.17 demonstra um alinhamento inválido.

Janela de Warping: Seja $|i_s - j_s| \leq r$, tal que $r > 0$ sendo este o tamanho da janela. Um alinhamento ideal e válido dificilmente estará fora de uma diagonal (Figura 2.18).

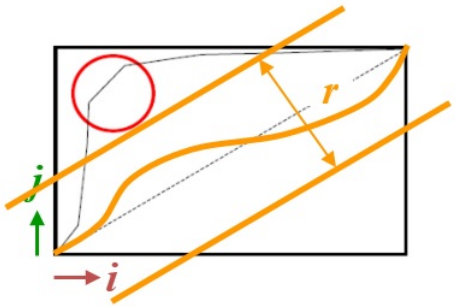


Figura 2.18: Alinhamento de menor custo, dentro da janela de warping

A janela de warping evita que o alinhamento ignore características importantes das séries e fique preso em características similares (Figura 2.19).

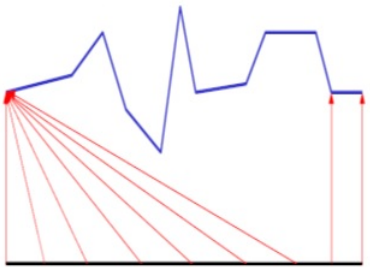


Figura 2.19: Alinhamento preso em características similares

Podemos agora passar para o cálculo realizado pelo DTW para preencher a matriz de custo.

$$DTW(Q, C) = \min \left\{ \sqrt{\sum_{k=1}^{K} w_k} \right.$$

Utilizando programação dinâmica poderemos obter o alinhamento com o menor custo para as séries apresentadas. Este caminho pode ser encontrado utilizando programação dinâmica para avaliar a recorrência que define a distância cumulativa $\gamma(i, j)$ como a distância $d(i, j)$ encontrada na célula atual e a distância mínima entre os elementos adjacentes.

$$\gamma(i, j) = d(q_i, c_j) + min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \qquad (2\text{-}4)$$

A DE entre duas sequencias pode ser entendida como um caso especial do DTW onde k-ésimo elemento de W é uma delimitante tal que $w_k = (i, j)_k$, $i = j = k$.

Precisamos notar que este caso especial é definido apenas quando duas séries temporais apresentam o mesmo tamanho. Assim a complexidade de tempo e espaço do DTW é de $O(nm)$.

O DTW resolve o problema de dimensionamento entre séries temporais, porém ele não resolve o problema de diferença de complexidade significativa entre as instâncias. Uma vez que o DTW não leva em consideração as diferenças entre as complexidades das instâncias, isto necessariamente, uma vez que o DTW é uma derivação da DE, a agrupar séries grandes e simples com séries pequenas e complexas.(13). Uma vez que o DTW não é provido do devido fator de correção o DTW irá apenas *suprimir* o problema da dimensionalidade e retornar um resultado não satisfatório.

Para o leitor mais é possível achar uma descrição mais apurada do DTW nos trabalhos de Kruskall e Liberman (1983) e Rabiner e Juang (1993).

É importante demonstrar o funcionamento do DTW como se segue:

- **Inicia calculando** $g(1, 1) = d(1, 1)$
- **Realiza cálculo da primeira linha** $g(i, 1) = g(i-1, 1) + d(i, 1)$
- **Calcula a primeira coluna** $g(1, j) = g(1, j) + d(1, j)$
- **realiza o cálculo da segunda linha** $g(i, 2) = min(g(i, 1), g(i-1, 1), g(i-1, 2)) + d(i, 2)$
- **Mantem o registro do índice de cada célula para recuperar o melhor caminho posteri**

• **Refaz o melhor caminho através da matriz iniciando em $g(n, m)$ e avançando em direç**

A Figura 2.20 ilustra o melhor caminho encontrado (círculos vermelhos) e os possíveis caminhos recuperados (setas vermelhas).

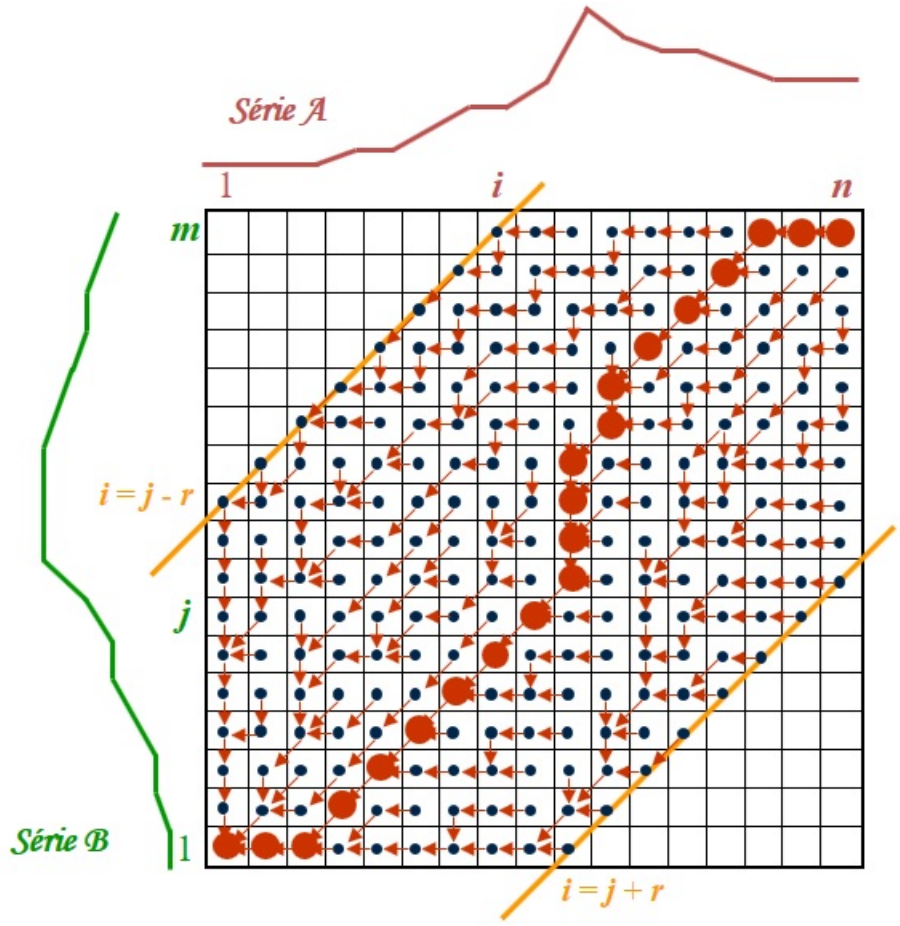


Figura 2.20: Caminhos recuperados na Matriz de custo

A Figura 2.21 demonstra um exemplo dos valores encontrados para cada célula e a recuperação do melhor caminho através da matriz de custo.

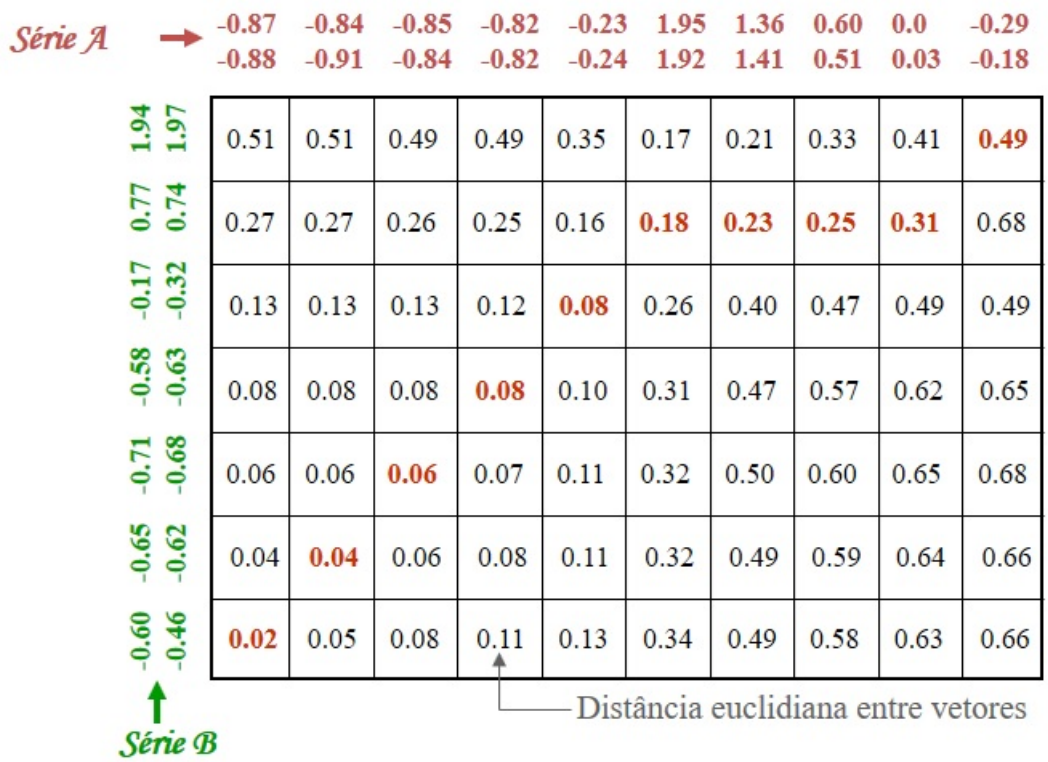


Figura 2.21: Matriz Preenchida

2.3.3 Complexity Invariant Distance (CID)

Nos casos onde existe uma diferença significativa nas complexidades das instâncias faz-se necessário a utilização de um fator de correção que possa forçar uma série temporal complexa ser clusterizada com outras também complexas, deixando de fora as séries mais simples, independente de seus tamanhos (23). O fenômeno de agrupar séries simples com séries mais complexas, se deve ao fato de que frequentemente a distância entre duas séries extremamente complexas é maior do que a distância de uma delas para uma série menos complexa. O *complexity invariant distance (CID)* utiliza a informação obtida pelo cálculo da complexidade de cada série temporal como um *fator de correção (CF)* garantindo assim que a complexidade seja um parâmetro importante no cálculo da distância. Introduzindo este fator de correção nos métodos de medida de similaridade baseados em distância nos tornamos hábeis a alcançar a *invariância de complexidade* que irá suprimir o problema de clusterizar séries simples com séries mais complexas.

Tornando mais claro, convencionamos que sejam Q e C duas séries temporais e CF é o recém introduzido *fator de correção*, teremos que:

$$CID(Q,C) = DE(Q,C)XCF(Q,C) \tag{2-5}$$

O CF é extremamente importante uma vez que ele *força* as séries temporais com complexidades diferentes a se agruparem com séries temporais mais similares a elas ao invés de se agruparem com séries menos complexas. Vale ressaltar que, nos casos onde as séries não apresentam variações consideráveis em suas instâncias, o CID simplesmente se reduz a uma DE custosa. Podemos obter o CF através do cálculo:

$$CF(Q,C) = \frac{max(CE(Q), CE(C)}{min(CE(Q), CE(C)} \tag{2-6}$$

Neste ponto necessitamos de explicar o que é a *estimativa de complexidade (CE)*. Nossa proposta estima que a complexidade de uma série temporal é baseada no fato de que se *esticarmos* uma série temporal em uma linha reta, uma série mais complexa resultará, necessariamente, em uma linha mais longa do que uma série temporal menos complexa. Podemos formular a ideia matematicamente como:

$$CE(Q) = \sqrt[2]{\sum_{i=1}^{n-1}(q_i - q_{i+1})^2} \tag{2-7}$$

Aqui $CE(Q)$ é a estimativa de complexidade de uma série temporal Q. Basicamente, estamos medindo a distância entre cada ponto da série temporal Q para saber como esta variação se comporta durante o tempo de observação (23), e adicionando este resultado ao cálculo, forçamos a medida de distância a considerar a complexidade existente em cada uma das séries temporais submetidas ao cálculo.

O CID, quando existem diferenças significativas nas séries temporais, vai agrupar as séries de uma maneira bem mais efetiva do que a DE e o DTW. Para fins de comparação em termos de complexidade, a Figura 2.22 apresenta a matriz obtida com a distância euclidiana e a Figura 2.23 a matriz obtida com o CID para o mesmo dataset composto das figuras geométricas apresentadas na Figura 2.3.

	4	5	6	7	10	12	24	32
4		1.000	1.122	1.231	1.181	1.048	1.155	1.170
5			1.318	1.068	1.103	1.153	1.165	1.180
6				1.088	1.097	1.103	1.186	1.200
7					1.217	1.199	1.198	1.191
10						1.263	1.195	1.214
12							1.135	1.199
24								1.191
32								

Figura 2.22: Matriz de distância para a Figura 2.3 utilizando a distância euclidiana

	4	5	6	7	10	12	24	32
4		1.000	1.061	1.229	1.707	1.839	3.843	5.042
5			1.307	1.138	1.700	2.159	4.135	5.422
6				1.096	1.599	1.953	3.982	5.214
7					1.651	1.977	3.744	4.819
10						1.439	2.580	3.396
12							2.018	2.761
24								1.446
32								

Figura 2.23: Matriz obtidas com o CID para os exemplos da Figura 2.3

A Figura 2.24 ilustra como a distância euclidiana (esquerda) agrupa as formas geométricas pela sua similaridade obtida pelo seu cálculo e como o CID (direita) executa a mesma tarefa para o dataset composto pelas imagens da Figura 2.3.

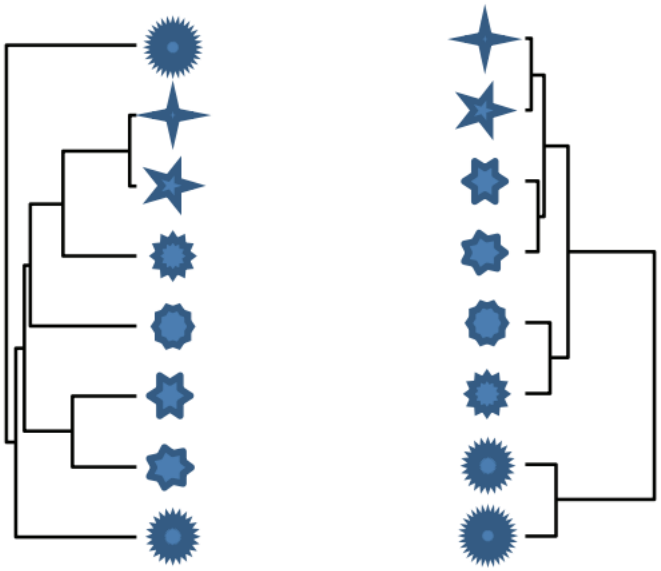


Figura 2.24: Similaridade obtida pela DE (esquerda) e CID (direita) utilizando o mesmo dataset da Figura 2.3

O CID vai clusterizar as séries temporais com diferentes complexidades em clusters diferentes devido ao CF forçar este acontecimento, enquanto a DE não leva em conta a complexidade existente e assim não resultará em um resultado confiável. No exemplo a seguir ilustramos a utilização do CID na clusterização de folhas. A Figura 2.25 ilustra uma série de folhas que foram convertidas em séries temporais para serem utilizadas no CID.

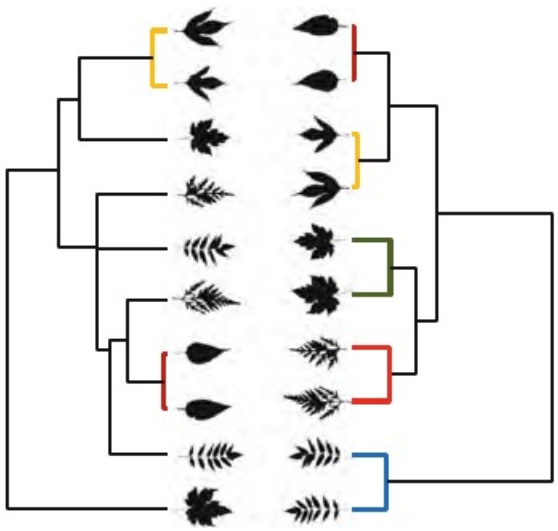


Figura 2.25: imagens de folhas que foram convertidas em s eries temporais

A Figura 2.26 demonstra o valor obtido com o cálculo do CF para cada folha da Figura 2.25, assim como a transformação de cada imagem em uma série temporal.

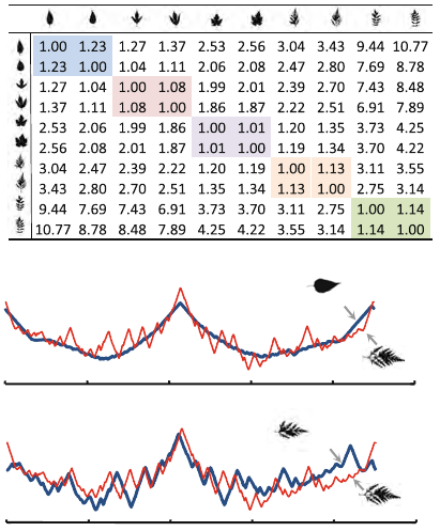


Figura 2.26: Acima Valores obtidos para o CF de cada folha. No Centro uma comparação entre uma folha de forma simples(em azul) e uma complexa (vermelha) e abaixo a comparação entre uma folha de forma complexa (azul) e a mesma folha utilizada no exemplo acima em vermelho

Tendo em mãos os valores de CF para cada folha, só nos resta aplicar os valores na forma e obter o valor para cada célula a ser preenchida.

2.4 Algoritmos de Clusterização

Aqui apresentaremos e discutiremos os algoritmos considerados nesta dissertação.

2.4.1 K-Means

K-means talvez seja o mais popular algoritmo de clusterização conhecido no campo da mineração de dados. O K-means tem sua origem enraizada no processamento de sinais e seu alvo é particionar n observações em k grupos. Cada observação será inserida em um cluster que contenha o *mean* mais próximo, ou seja, que contenha um centroide significativo mais próximo da observação analisada. O K-mean é um método de *quantização vetorial* que foi inicialmente proposto por Stuart Lloyd em *pulse-code modulation* (3). Se um dado conjunto de observações é descrito na forma $S = (x_1, x_2, ...x_n)$ onde cada observação é um vetor real com d dimensões, o algoritmo *K-Means* vai particionar estas n observações em $k(\leq n)$ conjuntos disjuntos $S_1, S_2, ..., S_k$ objetivando minimizar a soma dos quadrados dos erros.

$$SSE(s_1, s_2, ..., s_k) = \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2 \quad (2\text{-}8)$$

Onde μ_i é o *mean* dos pontos em S_i.

Dado um conjunto de k means $m_1^{(1)}, ..., m_k^{(1)}$ o algoritmo vai alternar entre duas etapas como explicado abaixo.

Etapa de designação: nesta etapa cada observação será designada a um cluster, tal cluster, neste momento, irá apresentar o *mean* mais próximo nesta etapa.

Etapa de atualização step: Basicamente, nesta etapa o algoritmo vai calcular um novo *mean* pela média dos pontos que foram designados para cada cluster.

O algoritmo irá convergir quando as designações não mudarem mais.

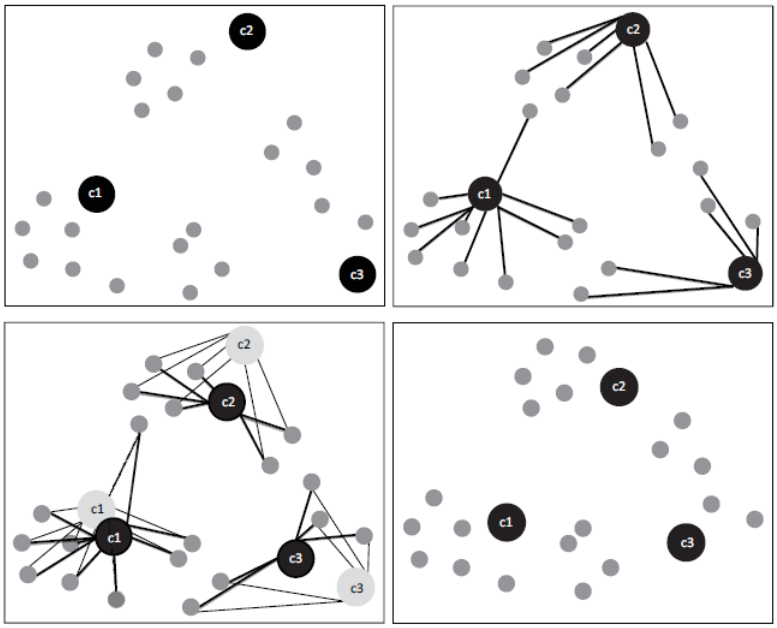


Figura 2.27: O Algoritmo K-means

A complexidade do K-means é aproximadamente de O($l.k.m.d$) , assumindo que o algoritmo irá convergir depois de l interações e k sendo o número de cluster, m número de objetos e d o número de atributos existentes.

A Figura 2.27 nos fornece uma boa ideia de como o algoritmo K-means funciona. Como podemos ver, as imagens na parte superior da figura ilustram a etapa de designação e as duas imagens na parte inferior da figura a etapa de atualização.

Podemos demonstrar o algoritmo K-means também através do seu pseudocódigo.

Data: Um conjunto D contendo m objetos em um espaço euclidiano.
Result: Particionamento destes m objetos entre k clusters $C_1, C_2, ...C_k$

1 Selecione k pontos como centroides iniciais (aleatoriamente)
2 Repetir
3 Formar k clusters designando cada objeto para o centroide mais próximo
4 Re-computar o centroide de cada cluster
5 Repetir até os centroides não mudarem mais.

Dados este pseudocódigo, nós temos que:

- **Passo 1** A Figura 2.27 Na primeira imagem do canto superior esquerdo - K pontos são selecionados aleatoriamente para desempenharem a função de centroide para iniciar o processo de clusterização.
- **Passo 2** Repetir o passo 1 se necessário.
- **Passo 3** A Figura 2.27 primeira imagem no canto superior direito - os objetos estão sendo designados para o centroide mais próximo, formando os primeiros clusters.
- **Passo 4** A Figura 2.27, no canto inferior esquerdo - O algoritmo recalcula os centroides de cada cluster, visando obter o melhor resultado.
- **Passo 5** Canto inferior direito - A Figura 2.27 nos demonstra que os centroides não mudaram, desta forma o algoritmo para e retorna o resultado obtido.

Apesar da popularidade do algoritmo k-means para clusterização em geral, ele apresenta alguns problemas que devem ser seriamente considerados. O K-means desenvolve muito mal computacionalmente falando, a quantidade de clusters esperados devem ser fornecidos pelo usuário como entrada ao algoritmo e a busca tende ao mínimo local (8). Portanto, estaremos utilizando outros algoritmos de clusterização nesta dissertação para podermos obter uma avaliação mais justa (2).

2.4.2 X-Means

O algoritmo X-Means não necessita que o número esperado de clusters seja fornecido pelo usuário como o K-means necessita. Isto é importante, uma vez que o usuário pode não saber quantos grupos existem dentro da massa de dados que esta sendo analisada. Adicionalmente, o X-means é mais rápido e é um algoritmo baseado em estatísticas.

Quando iniciamos o algoritmo, fornecemos um limite inferior e um superior para o número de clusters que o algoritmo deve trabalhar, isto é, um número mínimo de clusters desejados e um número máximo com a mesma intenção. Desta forma, o algoritmo vai ser executado e vai verificar o melhor número de clusters que esteja dentro de valor mínimo e máximo fornecido como limites superiores e inferiores.

Basicamente, o X-means inicia com um valor K de clusters, que é igual ao limite inferior fornecido, e vai continuamente adicionando centroides enquanto o algoritmo calcular necessário, ou até atingir o limite superior fornecido. Enquanto este processo esta em execução, o conjunto de centroides que obtiver o melhor resultado é guardado e será este que será retornado como resultado ao final do algoritmo.

Nos obtemos o melhor resultado através de uma técnica conhecida como medida *Bayesian Information Criterion (BIC)*, que, dado um conjunto de dados D e um conjunto de modelos alternativos M_j, onde diferentes modelos correspondem a diferentes soluções com diferentes valores de k. Então, nos usamos as probabilidades posteriores $Pr[M_j|D]$ para atribuir valores a cada modelo, aquele com o melhor valor necessariamente é o que apresenta o melhor resultado. Para calcular as probabilidades posteriores até a normalização, nos usamos a formula de Kass e Wasserman(1995):

$$BIC(M_j) = i_j(D) - \frac{p_j}{2}.logR \tag{2-9}$$

Onde $i_j(D)$ é a log-verossimilhança dos dados para com o j-ésimo modelo e p_j é o número de parâmetros em M_j e R é o número total de pontos pertencentes ao centroide em questão, uma descrição mais detalhada sobre o cálculo das probabilidades posteriores utilizado neste estudo pode ser encontrado em (8).

O algoritmo X-Means pode ser escrito na forma do pseudocódigo a seguir:

Data: Um conjunto D contendo m objetos com n atributos em um espaço euclidiano; K_{min} e K_{max}

Result: Particionamento destes m objetos em k-clusters $C_1, C_2, ...C_k$

1 Melhorar-Parâmetros

2 Melhorar-Estrutura

3 Se o algoritmo achou um k que obteve o melhor resultado é tal que k ¿ K_{max} o algoritmo para e retorna o melhor resultado encontrado durante a busca.

4 Senão, Repete 1.

No caso do dataset apresentar um número de clusters inferior ao K_{min} o X-means irá necessariamente agrupar os dados em K_{min} grupos, da mesma forma, se o dataset apresentar uma quantidade de clusters maior que K_{max} os dados serão clusterizados em no máximo K_{max} grupos.

A operação ***Melhorar-parâmetros*** é simples: Consiste apenas de executar o K-means simples ate ele convergir.

A operação ***Melhorar-estrutura*** é a etapa a qual o algoritmo vai calcular se, e onde os novos centroides devem ser colocados. Para realizar esta tarefa, deixamos alguns centroides se dividirem em dois. Existem algumas técnicas que nos dão as diretivas de como dividir estes centroides. Para um melhor entendimento desta dissertação, daremos uma breve explicação de duas delas abaixo:

Divisão 1: Uma por vez A primeira ideia é escolher um centroide aleatoriamente, produzir um novo centroide perto deste e executar o K-means ate convergir e comparar se o modelo resultante é melhor que o anterior.

Divisão 2: Duplicar metade dos centroides A essência é usar um sistema de SPLITLOOP para identificação de modelos de mistura gaussianos (7), simplesmente escolhendo metade dos centroides, de acordo com uma heurística pré-definida, dividi-los e executar o K-means ate convergir e comparar os resultados. Esta ideia é a utilizada nesta dissertação (8).

A primeira ideia requer O (K_{max}) etapas de Melhorar-estrutura ate que o algoritmo X-Means atinja a completude. A segunda ideia é muito mais agressiva, porém requer apenas O ($logK_{max}$) etapas de melhorar-estrutura ate que o algoritmo X-Means tenha completado sua tarefa (15, 21).

2.4.3 Make Density Based Cluster (MDBC)

No Density based clustering algorithm, os clusters são regiões densas no espaço, separadas por regiões onde a densidade de objetos é menor. A partir deste ponto, algumas definições são necessárias.

Para o MDBC nos temos as seguintes definições:

- $\in$ **-neighbourhood** é todo objeto que esta dentro de um raio $\in$ a um core object.
- **Core object** é um objeto com no mínimo MinPts objetos em sua "$\in -neighbourhood$".
- **Border object** é um objeto que esta na borda, ou perímetro que define uma $\in$-neighbourhood.

E existem dois parâmetros globais que são:

- $\in$**(Eps)** é o raio máximo de uma $\in$ -neighbourhood.
- **MinPts** número mínimo de objetos em uma $\in$ - neighbourhood.

A Figura 2.28 ilustra estas definições.

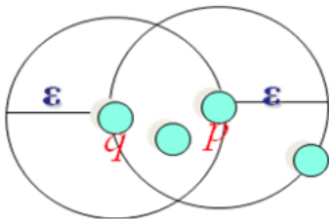


Figura 2.28: A $\in$-neighbourhood de q é composta de 3 objetos, e a $\in$-neighbourhood de p é composta de 4 objetos

O core object ilustrado na Figura 2.28 é o objeto p com um MinPts = 4. Também podemos ver que q é um border object para p e p é um border object para q.

A figura 2.29 demonstra os parâmetros globais.

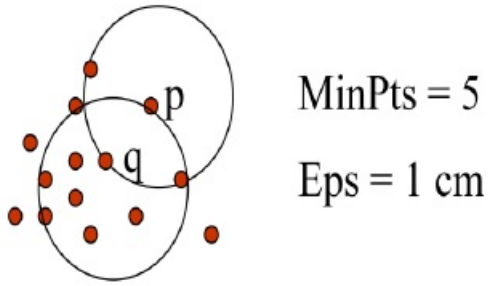


Figura 2.29: Ilustração dos parâmetros globais do make density based clustering algorithm.

.

2.4.4 Densidade por alcance e Densidade por conexão

Um objeto q como o mostrado na Figura 2.30 é apresenta densidade de alcance se a partir de um objeto p, q esta dentro de uma $\in$-neighbourhood de p e p é um core object.

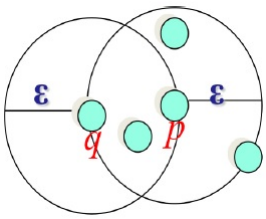


Figura 2.30: Densidade de alcance.

Um objeto p é *conectado por densidade* se para um objeto q existe um objeto o que esteja em uma posição tal qual p e q sejam densamente alcançáveis a partir de o, o objeto o serve como um tipo de ponte que conecta p e q como a Figura 2.31 demonstra (17, 1).

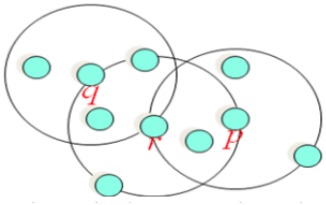


Figura 2.31: p e q não são diretamente alcançáveis, porém eles são conectados pela densidade por r uma vez que r é diretamente alcançável por p e q

Uma vez que temos as definições em mente, nos precisamos entender a intuição para a formalização da ideia básica que é:

- **I** Para qualquer ponto dentro de um cluster ser considerado um core object, a densidade em volta deste ponto deve exceder a um limiar
- **II** O conjunto de pontos dentro de um cluster deve estar espacialmente conectado

Basicamente, podemos determinar o make density-based cluster algorithm como:

```
1 Arbitrariamente selecionar um ponto "p"
2 Se um ponto "q" é density reachable a partir de "p"
3 Formar Clusters rotulado Clusters
4 Senão
5 Selecionar outro ponto "q" + 1.
```

2.5 Métricas de Avaliação para Algoritmos de Clusterização

Para a avaliação dos algoritmos de clusterização, existem dois tipos principais de métricas, a *interna* e *externa*. Para este estudo a métrica que será usada é a externa.

Uma avaliação externa é feita baseando-se nos dados que não foram usados na clusterização.

Para esta tarefa, nos precisamos ter em mãos classes rotuladas e um *benchmark* externo que consiste de um conjunto de itens que foram previamente classificados. Para a avaliação dos algoritmos desta dissertação, estamos usando o *F-Measure* assim como o *Recall e precision* (24, 20, 14).

Convencionando que TP seja o número de *true positives*, que é quando um objeto e designado para o clusters correto, FP, o número de *False positives*(Objetos que foram agrupados em um cluster ao qual ele não pertence), e FN e TN, o número de *false negatives (Objetos que não foram clusterizados no grupo correto) e true negatives respectivamente (Objetos que corretamente não foram agrupados em um grupo específico)*, A Figura 2.32 ilustra estas definições.

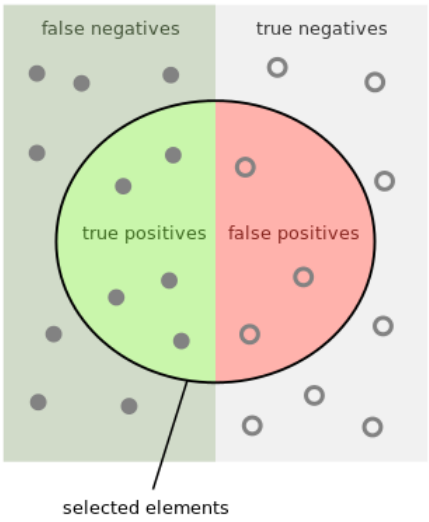


Figura 2.32: FN = 7, TP = 5, FP = 3 e TN = 7

Uma vez que estamos realizando uma tarefa de clusterização, e uma vez que possuímos os rótulos de cada clusters previamente, devemos considerar as possíveis combinações de cada elemento pertencente a cada clusters. Desta forma estaremos considerando:

- **Número de objetos que deveriam estar juntos e estão**
- **Número de objetos que deveriam estar juntos e não estão**
- **Número de objetos que não deveriam estar juntos e estão**
- **Número de objetos que não deveriam estar juntos e não estão**

A Figura 2.33 apresenta a clusterização de 3 formas geométricas *Círculos, Xs e Losângulos*

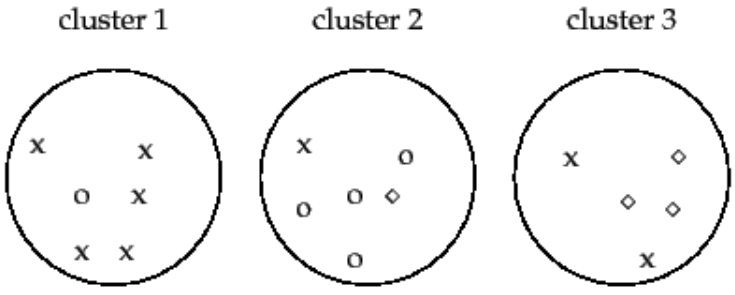


Figura 2.33

Para exemplificar, utilizando a Figura 2.33 como exemplo, inicialmente necessitamos de considerar todos os possíveis pares de objetos, desta forma

teremos para $N = 17$:

$$\text{Total de pares} = \frac{N(N-1)}{2} = 136 \tag{2-10}$$

Desta forma, considerando o primeiro cluster, poderemos calcular o número de objetos que deveriam estar juntos e estão como:

$$\text{TP} = \binom{5}{2} + \binom{4}{2} + \binom{3}{2} + \binom{2}{2} = 20 \tag{2-11}$$

Para os objetos que não deveriam estar juntos porém estão teremos:

$$\text{TP} + \text{FP} = \binom{6}{2} + \binom{6}{2} + \binom{5}{2} = 40 \tag{2-12}$$

subtraindo o valor de TP previamente calculado, temos que $FP = 20$.
Uma vez que temos o total de positivos, torna-se trivial obter o valor do total negativo, uma vez que a soma do total positivo com o total negativo deve ser igual a N.

2.5.1 Recall

Podemos calcular o recall como:

$$Recall = R = \frac{TP}{TP + FN} \tag{2-13}$$

Utilizando o exemplo da Figura 2.32, Temos que o cálculo do recall será:

$$R = \frac{5}{5+7} = 0.417 \tag{2-14}$$

2.5.2 Precision

O precision é definido como:

$$Precision = P = \frac{TP}{TP + FP} \tag{2-15}$$

E, novamente, usando o exemplo da Figura 2.32, teremos que:

$$P = \frac{5}{5+3} = 0.625 \tag{2-16}$$

F-Measure

A aplicação do F-Measure neste estudo é necessária para balancear a contribuição de false negatives através do peso do parâmetro $\beta \geq 0$ do Recall.

O F-Measure será calculado da seguinte forma:

$$F_\beta = \frac{(\beta^2 + 1).P.R}{\beta^2.P + R} \tag{2-17}$$

$$\text{onde}\beta = 1 - E \tag{2-18}$$

O cálculo de β é baseado na medida de efetividade de *Van Rijsbergen* onde E é obtido como:

$$E = 1 - (\frac{\alpha}{P} + \frac{1-\alpha}{R})^{-1} \tag{2-19}$$

e α por sua vez é obtido pelo cálculo:

$$\alpha = \frac{1}{1 + \beta^2} \tag{2-20}$$

3

Trabalhos Relacionados

A mineração de dados é uma complexa, interessante e importante tarefa para a aprendizagem de máquina e para os problemas de clusterização. Neste capítulo estamos apresentando algumas das pesquisas relacionadas no campo da aprendizagem de máquina, mais específicamente nos problemas de clusterização de séries temporais.

3.1 Encontrando Séries Temporais Similares

Para inúmeras aplicações, a busca por objetos similares esta definitivamente na posição de tarefa extremamente importante incluindo a mineração de dados. Para objetos complexos, esta tarefa efetivamente não é trivial. No paper *Finding Similar Time Series* o autor apresenta um modelo intuitivo para medir a similaridade entre duas séries temporais. Como nossa dissertação é exatamente sobre como os métodos medidas de distância influenciam os resultados em modelos que comparam a similaridade entre séries temporais, o trabalho apresentado pelos autores é intimamente ligado ao tema desta dissertação (6).

3.2 Busca de Subsequências em Séries Temporais

A abordagem clássica para buscar combinações de subsequências tem o seu princípio baseado nas exaustivas pesquisas sobre encontrar similaridades, onde todos os possíveis candidatos são gerados e avaliados ou todos os termos das séries temporais de um dataset são examinados. No estudo *Subsequence searching in time series* o autor apresenta a noção de *eHaar* para aparar as partes da série temporal que não contém subsequências que poderiam combinar com uma outra sequência. Isto é bastante similar com o que é realizado pelo DTW apresentado e discutido nesta pesquisa, e a combinação de subsequências pode tirar vantagem dos métodos de medida de similaridades

baseados em distância (25) .

3.3 Medidas de Distância usando Invariância de Complexidade em Séries Temporais

O *CID* Foi amplamente utilizado nesta pesquisa como parte ativa do processo de demonstrar como a invariância de complexidade pode ter uma performance melhor em certas ocasiões onde existam diferenças significativas nas complexidades de cada instância do dataset. A pesquisa (23) foi exaustivamente consultada para a realização deste trabalho.

3.4 Dynamic Time Warping:

Para a avaliação dos métodos de medida de similaridades baseados em distância, o DTW não poderia ter sido omitido, uma vez que atualmente o DTW é provavelmente o segundo mais popular método de medida de similaridade baseado em distância, atrás apenas da distância euclidiana. O DTW trouxe uma perspectiva totalmente nova para os problemas de clusterização de séries temporais (13, 4, 19).

4

Resultados Experimentais

4.1 Ambiente do Experimento

Para a implementação, foi utilizado um computador com as seguintes configurações.

- **S.O Windows 7 64 Bits**
- **Processor Intel Core i7-3960X - CPU 3.30GHz X 8**
- **Ram 32Gb**

A linguagem de programação utilizada nesta implementação foi o JAVA em sua versão 1.8.0_31.
TA DE usada para a implementação foi o NetBeans em sua versão 8.0.1.
Partes do código foram reutilizadas do WEKA em sua versão 3.6.
Uma adaptação significativa foi necessária aos algoritmos de clusterização fornecidos pelo WEKA, uma vez que não havia suporte para o DTW e o CID e estes são partes cruciais do projeto.

4.2 Datasets

Nesta seção introduziremos uma pequena descrição dos datasets usados neste experimento, assim como seus atributos. Uma visão mais detalhada dos datasets utilizados pode ser encontrada no apêndice desta dissertação.
Para este estudo, tentamos manter o dataset o mais fiel possível a formatação ao qual encontramos os mesmos. Porém em alguns casos algumas alterações foram feitas para atender aos requisitos dos algoritmos usados, uma vez que, como o X-Means por exemplo, não permite o uso de outro tipo de data que não seja numérico. Nestes casos, o atributo nominal foi convertido em binário. Para o K-Means e o Make Density Based Cluster Algorithm nenhuma alteração foi necessária nos datasets.

Cada Dataset foi testado com os três algoritmos, cada algoritmo foi executado com os três métodos de medida de similaridade baseados em distância e cada método de medida de similaridade baseado em distância foi executado com três números de K diferentes, sendo K o número de clusters fornecido.
Convencionamos que o primeiro K_1 fornecido seria igual ao número real de clusters existentes no dataset, uma vez que tínhamos este número previamente conhecido. Na segunda execução no mesmo dataset, com o mesmo método e com o mesmo algoritmo, aumentamos o valor de K_1 em 2 ($K_2 = K_1 + 2$), e na terceira vez , subtraímos o valor de K_1 em 2, assim temos ($K_3 = K_1 - 2$). Nos casos onde o número de clusters existentes era igual a 2 ou igual a 10 ($K_1 = 2$) ou ($K_1 = 10$) a variação aplicada foi de +1 e - 1.
A Tabela 4.1 Descreve os tamanhos, atributos e distribuição, também fornece outras informações como os tipos de parâmetros que foram utilizados para clusterizar os dados.

4.3 Implementação

4.3.1 Medidas de Distância

A *Complexity-Invariant Distance (CID)*, *Euclidean Distance (DE)* e o *Dynamic Time Warping(DTW)* usados nesta dissertação foram implementados como explicado na seção *conceitos básicos*. A Distância Euclidiana foi implementada seguindo as instruções encontradas na literatura (23). Para as implementações do DTW e CID seguimos as direções apontadas nos trabalhos (13, 23, 19).

4.3.2 Algoritmos de Clusterização

Os algoritmos de clusterização utilizados neste projeto foram adaptados do WEKA, permitindo assim que os novos métodos de medida de similaridade baseados em distância pudessem ser executados no WEKA. A motivação para o uso do WEKA é o reuso de um algoritmo bem implementado, robusto e confiável, evitando com isto erros que poderiam ocorrer derivados de uma nova implementação, o que custaria tempo para verificar, testar e reparar.

Dataset	Objetos	Tamanho	Atributos	Classes	Distribuição	Completo?	Grupos
Dow Jones	750	7	16	10	75 Cada	Sim	Segmento
Solar flare	1066	24	11	6	B=147, C=211, D=239, E=95, F=43, H=331	Sim	Região solar
Quake	2178	3 - 27	4	3	Fraco=1209, Médio=851, Forte=118	Sim	Força
HV	8.235	30	7	5	A=2175, B-E=1515 Cada	Não	Local
EEG	14.979	256	15	2	1=8257, 2=6723	Sim	Pre Disposto
Arritmia	452	60 - 120	279	3	A=68, B=285, C=99	Não	Idade
Adult	32.561	13 - 24	15	9	A=1.298, B=2.541, C=22.696, D=960, E=2.093, F=1.836, G=1.116, H=14, I=7	Não	Ocupação
F Gen	10.104	24	72	5	A=577, B=3.910, C=3.218, D=1.921, E=478	Não	Idade
Gesture	9.873	128	33	5	D=2.741, H=998, P=2.097, R=1.087, S=2.850	Sim	Gesto
Otto Group	61.870	5 - 19	95	9	1=1.929, 2=16.122, 3=8.004, 4=2.691, 5=2.739, 6=14.135, 7=2.839, 8=8.464, 9=4.955	Sim	Produto
Sido	12.678	4932	1	2	1=452, -1=12226	Sim	Ativo

Tabela 4.1

4.4 Resultados

As tabelas seguintes são compostas dos resultados obtidos com cada dataset testado. Também é possível encontrar informações relevantes a esta dissertação, tais como quantos clusters foram encontrados, a distribuição correta dos objetos em cada cluster, o precision, assim como o recall e o f-measure, bem como o tempo de execução medido em segundos. Também pode ser encontrado o número de clusters fornecidos em cada execução, esta informação pode ser encontrada na coluna K Max.

É importante mencionar que o método de medida de similaridade baseado em distância usado também foi informado na tabela de resultados, ao lado do algoritmo usado. Cada método foi executado três vezes com um número diferente de K em cada algoritmo.

Cada teste foi executado com 500 iterações.

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	6	567645	109877	62212	175802	219754	0.638	0.385	0.564	0.03	6
K means	DE	3	567645	99054	84041	186442	198108	0.541	0.347	0.487	0.03	3
K means	DE	6	567645	79167	124150	205993	158335	0.389	0.278	0.360	0.04	9
K means	CID	5	567645	53604	175708	231124	107209	0.234	0.188	0.223	0.01	6
K means	CID	3	567645	64068	154604	220837	128136	0.293	0.225	0.276	0.01	3
K means	CID	4	567645	53604	175708	231124	107209	0.234	0.188	0.223	0.01	9
K means	DTW	4	567645	75799	130943	209304	151599	0.367	0.266	0.341	0.11	6
K means	DTW	3	567645	68685	145292	216298	137370	0.321	0.241	0.301	0.08	3
K means	DTW	5	567645	72772	137049	212280	145544	0.347	0.255	0.324	0.17	9
Xmeans	DE	3	567645	112015	57900	173700	224030	0.659	0.392	0.580	0.03	6
Xmeans	DE	3	567645	112015	57900	173700	224030	0.659	0.392	0.580	0.03	3
Xmeans	DE	3	567645	112015	57900	173700	224030	0.659	0.392	0.580	0.03	9
Xmeans	CID	2	567645	67095	148498	217861	134191	0.311	0.235	0.292	0.01	6
Xmeans	CID	2	567645	67095	148498	217861	134191	0.311	0.235	0.292	0.01	3
Xmeans	CID	2	567645	67095	148498	217861	134191	0.311	0.235	0.292	0.01	9
Xmeans	DTW	4	567645	63368	156016	221525	126736	0.289	0.222	0.273	5.89	6
Xmeans	DTW	3	567645	64787	153154	220130	129574	0.297	0.227	0.280	3.12	3
Xmeans	DTW	4	567645	63368	156016	221525	126736	0.289	0.222	0.273	3.12	9
MDBC	DE	6	567645	109517	62937	176155	219035	0.635	0.383	0.561	0.03	6
MDBC	DE	3	567645	88401	105527	196915	176802	0.456	0.310	0.417	0.04	3
MDBC	DE	6	567645	79338	123807	205825	158675	0.391	0.278	0.361	0.04	9
MDBC	CID	4	567645	66736	149223	218214	133472	0.309	0.234	0.290	0.04	6
MDBC	CID	2	567645	70653	141323	214363	141305	0.333	0.248	0.312	0.02	3
MDBC	CID	3	567645	66736	149223	218214	133472	0.309	0.234	0.290	0.02	9
MDBC	DTW	4	567645	79867	122738	205304	159735	0.394	0.280	0.364	0.14	6
MDBC	DTW	3	567645	79338	123807	205825	158675	0.391	0.278	0.361	0.09	3
MDBC	DTW	5	567645	75799	130943	209304	151599	0.367	0.266	0.341	0.18	9

Tabela 4.2: Flare dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	2	112177731	37139803	20123408	30154652	24759868	0.649	0.552	0.627	1.2	2
K means	DE	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.08	1
K means	DE	2	112177731	32562952	24555541	33350604	21708634	0.570	0.494	0.553	1.36	4
K means	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.09	2
K means	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.09	1
K means	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.08	4
K means	DTW	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.44	2
K means	DTW	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.15	1
K means	DTW	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.69	4
Xmeans	DE	2	112177731	37139803	20123408	30154652	24759868	0.649	0.552	0.627	0.43	2
Xmeans	DE	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.28	1
Xmeans	DE	2	112177731	35006183	22189564	31644529	23337455	0.612	0.525	0.592	0.41	4
Xmeans	CID	1	112177731	37139803	20123408	30154652	24759868	0.649	0.552	0.627	0.05	2
Xmeans	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.04	1
Xmeans	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.05	4
Xmeans	DTW	2	112177731	37092688	20169033	30187551	24728459	0.648	0.551	0.626	2.84	2
Xmeans	DTW	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.63	1
Xmeans	DTW	2	112177731	37765755	19517249	29717558	25177169	0.659	0.560	0.637	3.28	4
MDBC	DE	2	112177731	36359046	20879478	30699843	24239364	0.635	0.542	0.614	1.14	2
MDBC	DE	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.09	1
MDBC	DE	2	112177731	30853363	26211073	34544386	20568908	0.541	0.472	0.525	1.38	4
MDBC	CID	1	112177731	36278278	20957692	30756243	24185518	0.634	0.541	0.613	0.09	2
MDBC	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.08	1
MDBC	CID	1	112177731	37099419	20162515	30182851	24732946	0.648	0.551	0.626	0.1	4
MDBC	DTW	1	112177731	36278278	20957692	30756243	24185518	0.634	0.541	0.613	0.96	2
MDBC	DTW	1	112177731	36278278	20957692	30756243	24185518	0.634	0.541	0.613	0.2	1
MDBC	DTW	1	112177731	36278278	20957692	30756243	24185518	0.634	0.541	0.613	0.74	4

Tabela 4.3: egg Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	3	101926	22083	28880	28880	22083	0.433	0.433	0.433	0.32	3
K means	DE	2	101926	25353	25610	25610	25353	0.497	0.497	0.497	0.28	2
K means	DE	3	101926	22083	28880	28880	22083	0.433	0.433	0.433	0.12	4
K means	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.08	3
K means	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.02	2
K means	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.04	4
K means	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.08	3
K means	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	1.25	2
K means	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	2.17	4
Xmeans	DE	3	101926	22873	28090	28090	22873	0.449	0.449	0.449	0.13	3
Xmeans	DE	2	101926	25353	25610	25610	25353	0.497	0.497	0.497	0.11	2
Xmeans	DE	3	101926	21411	29552	29552	21411	0.420	0.420	0.420	0.11	4
Xmeans	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.03	3
Xmeans	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.02	2
Xmeans	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.04	4
Xmeans	DTW	3	101926	22313	28650	28650	22313	0.438	0.438	0.438	814.8	3
Xmeans	DTW	2	101926	24676	26287	26287	24676	0.484	0.484	0.484	301.83	2
Xmeans	DTW	3	101926	20622	30341	30341	20622	0.405	0.405	0.405	305.52	4
MDBC	DE	3	101926	28057	22906	22906	28057	0.551	0.551	0.551	0.04	3
MDBC	DE	2	101926	28394	22569	22569	28394	0.557	0.557	0.557	0.28	2
MDBC	DE	3	101926	22873	28090	28090	22873	0.449	0.449	0.449	0.12	4
MDBC	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.27	3
MDBC	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.03	2
MDBC	CID	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.04	4
MDBC	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	1.95	3
MDBC	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	0.03	2
MDBC	DTW	1	101926	32112	18851	18851	32112	0.630	0.630	0.630	2.44	4

Tabela 4.4: Arritmia Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	9	530093080	21897167	196031671	224575575	87588667	0.100	0.089	0.098	1.77	9
K means	DE	9	530093080	18170214	207778175	231463835	72680856	0.080	0.073	0.079	1.52	11
K means	DE	7	530093080	24462746	187945551	219833799	97850984	0.115	0.100	0.112	1.01	7
K means	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.17	9
K means	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.19	11
K means	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.17	7
K means	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	3.55	9
K means	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	4.34	11
K means	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	3.11	7
Xmeans	DE	4	530093080	33268068	160193188	203559554	133072270	0.172	0.140	0.165	5.88	9
Xmeans	DE	4	530093080	33268068	160193188	203559554	133072270	0.172	0.140	0.165	5.56	11
Xmeans	DE	4	530093080	33268068	160193188	203559554	133072270	0.172	0.140	0.165	5.4	7
Xmeans	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.3	9
Xmeans	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.47	11
Xmeans	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.47	7
Xmeans	DTW	3	530093080	40626951	136999667	189958656	162507806	0.229	0.176	0.216	142.14	9
Xmeans	DTW	3	530093080	40626951	136999667	189958656	162507806	0.229	0.176	0.216	147.51	11
Xmeans	DTW	3	530093080	40626951	136999667	189958656	162507806	0.229	0.176	0.216	146.76	7
MDBC	DE	9	530093080	21854935	196164776	224653629	87419740	0.100	0.089	0.098	1.73	9
MDBC	DE	9	530093080	18676995	206180917	230527188	74707980	0.083	0.075	0.081	1.54	11
MDBC	DE	7	530093080	24642231	187379856	219502070	98568924	0.116	0.101	0.113	1.08	7
MDBC	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.19	9
MDBC	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.19	11
MDBC	CID	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.19	7
MDBC	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	3.68	9
MDBC	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	0.2	11
MDBC	DTW	1	530093080	73588839	33111325	129037561	294355355	0.690	0.363	0.585	4.56	7

Tabela 4.5: Adult Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	5	51040356	5910473	12122834	24141340	8865709	0.328	0.197	0.289	4.64	5
K means	DE	3	51040356	8427783	6416815	23554083	12641674	0.568	0.264	0.461	1.93	3
K means	DE	5	51040356	4258807	15866686	24526652	6388210	0.212	0.148	0.195	5.54	7
K means	CID	1	51040356	7894922	7624659	23678393	11842382	0.509	0.250	0.422	0.21	5
K means	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.2	3
K means	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.23	7
K means	DTW	1	51040356	7894922	7624659	23678393	11842382	0.509	0.250	0.422	7.41	5
K means	DTW	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	5.64	3
K means	DTW	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	11.7	7
Xmeans	DE	4	51040356	6933322	9804330	23902722	10399982	0.414	0.225	0.355	0.87	5
Xmeans	DE	3	51040356	7249772	9087029	23828898	10874657	0.444	0.233	0.376	0.74	3
Xmeans	DE	4	51040356	6933322	9804330	23902722	10399982	0.414	0.225	0.355	0.77	7
Xmeans	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.19	5
Xmeans	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.13	3
Xmeans	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.13	7
Xmeans	DTW	3	51040356	7231397	9128678	23833185	10847095	0.442	0.233	0.375	1517.43	5
Xmeans	DTW	3	51040356	7231397	9128678	23833185	10847095	0.442	0.233	0.375	1572.17	3
Xmeans	DTW	3	51040356	7231397	9128678	23833185	10847095	0.442	0.233	0.375	1567.78	7
MDBC	DE	5	51040356	6294296	11252817	24051799	9441444	0.359	0.207	0.313	4.58	5
MDBC	DE	3	51040356	8470657	6319633	23544081	12705985	0.573	0.265	0.465	2.07	3
MDBC	DE	5	51040356	4669172	14936508	24430920	7003757	0.238	0.160	0.217	5.48	7
MDBC	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.19	5
MDBC	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.27	3
MDBC	CID	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.27	7
MDBC	DTW	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	7.95	5
MDBC	DTW	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	0.22	3
MDBC	DTW	1	51040356	7901047	7610776	23676964	11851570	0.509	0.250	0.422	6.53	7

Tabela 4.6: Final general Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	5	48733128	7259478	8069651	22514783	10889216	0.474	0.244	0.398	2.31	5
K means	DE	3	48733128	7571294	7369650	22435243	11356941	0.507	0.252	0.422	2.78	3
K means	DE	5	48733128	7148393	8319027	22543119	10722590	0.462	0.241	0.390	3.24	7
K means	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.29	5
K means	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.22	3
K means	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.29	7
K means	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	1.66	5
K means	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	1.22	3
K means	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	2.18	7
Xmeans	DE	4	48733128	8048763	6297772	22313448	12073145	0.561	0.265	0.459	0.47	5
Xmeans	DE	3	48733128	7912343	6604023	22348247	11868515	0.545	0.261	0.448	0.46	3
Xmeans	DE	4	48733128	8048763	6297772	22313448	12073145	0.561	0.265	0.459	0.47	7
Xmeans	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.06	5
Xmeans	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.06	3
Xmeans	CID	2	48733128	6187608	10475907	22788200	9281413	0.371	0.214	0.324	0.06	7
Xmeans	DTW	4	48733128	7399795	7754650	22478990	11099693	0.488	0.248	0.409	276.97	5
Xmeans	DTW	3	48733128	8237802	5873396	22265227	12356703	0.584	0.270	0.474	279.14	3
Xmeans	DTW	4	48733128	7399795	7754650	22478990	11099693	0.488	0.248	0.409	280.13	7
MDBC	DE	5	48733128	7187370	8231527	22533176	10781055	0.466	0.242	0.393	2.46	5
MDBC	DE	3	48733128	7631709	7234024	22419832	11447563	0.513	0.254	0.426	2.99	3
MDBC	DE	5	48733128	6934019	8800278	22597802	10401029	0.441	0.235	0.375	3.28	7
MDBC	CID	2	48733128	6976894	8704028	22586865	10465341	0.445	0.236	0.378	0.32	5
MDBC	CID	2	48733128	6976894	8704028	22586865	10465341	0.445	0.236	0.378	0.23	3
MDBC	CID	2	48733128	6976894	8704028	22586865	10465341	0.445	0.236	0.378	0.24	7
MDBC	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	1.77	5
MDBC	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	1.28	3
MDBC	DTW	1	48733128	5823173	11294034	22881162	8734759	0.340	0.203	0.300	2.29	7

Tabela 4.7: Gesture phase Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	9	1913917515	378955667	517962900	553830910	463168038	0.423	0.406	0.419	53.06	9
K means	DE	7	1913917515	383003603	513273626	549524771	468115515	0.427	0.411	0.424	55.85	7
K means	DE	8	1913917515	379386299	517464041	553372810	463694366	0.423	0.407	0.420	55.17	11
K means	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	1.73	9
K means	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	1.98	7
K means	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	1.93	11
K means	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	83.51	9
K means	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	67.41	7
K means	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	102.45	11
Xmeans	DE	4	1913917515	477570267	403724205	448926049	583696994	0.542	0.515	0.536	11.83	9
Xmeans	DE	4	1913917515	477570267	403724205	448926049	583696994	0.542	0.515	0.536	11.94	7
Xmeans	DE	4	1913917515	477570267	403724205	448926049	583696994	0.542	0.515	0.536	11.67	11
Xmeans	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	0.03	9
Xmeans	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	0.83	7
Xmeans	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	0.84	11
Xmeans	DTW	4	1913917515	603917532	257358995	314519560	738121429	0.701	0.658	0.692	374.19	9
Xmeans	DTW	4	1913917515	603917532	257358995	314519560	738121429	0.701	0.658	0.692	384.5	7
Xmeans	DTW	4	1913917515	603917532	257358995	314519560	738121429	0.701	0.658	0.692	384.6	11
MDBC	DE	9	1913917515	361041399	538715431	572887862	441272822	0.401	0.387	0.398	159.06	9
MDBC	DE	7	1913917515	382142340	514271343	550440971	467062861	0.426	0.410	0.423	57.32	7
MDBC	DE	9	1913917515	370945923	527241676	562351566	453378350	0.413	0.397	0.410	63	11
MDBC	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	1.89	9
MDBC	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	2.05	7
MDBC	CID	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	2.28	11
MDBC	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	87.98	9
MDBC	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	72.19	7
MDBC	DTW	1	1913917515	224272854	697153027	718380368	274111266	0.243	0.238	0.242	107.06	11

Tabela 4.8: Otto Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	10	280875	52380	5552	81324	141619	0.904	0.392	0.717	0.04	10
K means	DE	10	280875	51166	8677	82694	138338	0.855	0.382	0.685	0.04	11
K means	DE	9	280875	49650	12583	84405	134238	0.798	0.370	0.648	0.02	9
K means	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.01	10
K means	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0	11
K means	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.01	9
K means	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.06	10
K means	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.08	11
K means	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.05	9
Xmeans	DE	4	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.02	10
Xmeans	DE	4	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.02	11
Xmeans	DE	4	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.01	9
Xmeans	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.01	10
Xmeans	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0	11
Xmeans	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0	9
Xmeans	DTW	2	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.03	10
Xmeans	DTW	2	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.04	11
Xmeans	DTW	2	280875	46813	19886	87606	126569	0.702	0.348	0.583	0.03	9
MDBC	DE	10	280875	53290	3209	80297	144079	0.943	0.399	0.741	0.04	10
MDBC	DE	10	280875	51568	7642	82240	139425	0.871	0.385	0.696	0.04	11
MDBC	DE	8	280875	49346	13364	84748	133417	0.787	0.368	0.641	0.03	9
MDBC	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0	10
MDBC	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.01	11
MDBC	CID	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.01	9
MDBC	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.06	10
MDBC	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.06	11
MDBC	DTW	1	280875	46715	20140	87717	126303	0.699	0.347	0.581	0.06	9

Tabela 4.9: Dow Jones Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	4	33903495	2681766	11791990	14449316	4980423	0.185	0.157	0.179	0.01	5
K means	DE	3	33903495	3034193	11113916	14120456	5634930	0.214	0.177	0.206	0.13	3
K means	DE	5	33903495	2361378	12408421	14748279	4385417	0.160	0.138	0.155	0.22	7
K means	CID	5	33903495	3709381	9814844	13490419	6888850	0.274	0.216	0.260	0.04	5
K means	CID	3	33903495	3744979	9746352	13457201	6954962	0.278	0.218	0.263	0.04	3
K means	CID	5	33903495	3707007	9819411	13492634	6884443	0.274	0.216	0.260	0.04	7
K means	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.21	5
K means	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.14	3
K means	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.27	7
Xmeans	DE	4	33903495	3036566	11109350	14118242	5639337	0.215	0.177	0.206	0.05	5
Xmeans	DE	3	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.03	3
Xmeans	DE	4	33903495	3036566	11109350	14118242	5639337	0.215	0.177	0.206	0.04	7
Xmeans	CID	3	33903495	3744979	9746352	13457201	6954962	0.278	0.218	0.263	0.02	5
Xmeans	CID	3	33903495	3744979	9746352	13457201	6954962	0.278	0.218	0.263	0.02	3
Xmeans	CID	3	33903495	3744979	9746352	13457201	6954962	0.278	0.218	0.263	0.02	7
Xmeans	DTW	3	33903495	2972488	11232636	14178034	5520336	0.209	0.173	0.201	0.91	5
Xmeans	DTW	2	33903495	2948756	11278298	14200180	5476261	0.207	0.172	0.199	0.92	3
Xmeans	DTW	3	33903495	2972488	11232636	14178034	5520336	0.209	0.173	0.201	0.94	7
MDBC	DE	4	33903495	2933330	11307978	14214574	5447613	0.206	0.171	0.198	0.1	5
MDBC	DE	3	33903495	3122003	10944968	14038518	5798006	0.222	0.182	0.213	0.14	3
MDBC	DE	5	33903495	2763643	11634458	14372914	5132480	0.192	0.161	0.185	0.25	7
MDBC	CID	5	33903495	3247785	10702962	13921148	6031601	0.233	0.189	0.223	0.03	5
MDBC	CID	3	33903495	3214559	10766888	13952151	5969896	0.230	0.187	0.220	0.03	3
MDBC	CID	5	33903495	3243038	10712094	13925577	6022786	0.232	0.189	0.222	0.05	7
MDBC	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.21	5
MDBC	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.15	3
MDBC	DTW	1	33903495	3133869	10922137	14027446	5820043	0.223	0.183	0.214	0.28	7

Tabela 4.10: HV Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	3	2370753	372596	706259	777360	514538	0.345	0.324	0.341	0.04	3
K means	DE	2	2370753	511598	527518	625144	706493	0.492	0.450	0.483	0.03	2
K means	DE	3	2370753	361146	720983	789899	498725	0.334	0.314	0.330	0.03	4
K means	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0.02	3
K means	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0.01	2
K means	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0.02	4
K means	DTW	1	2370753	552721	474638	580112	763282	0.538	0.488	0.527	0.04	3
K means	DTW	1	2370753	552721	474638	580112	763282	0.538	0.488	0.527	0.04	2
K means	DTW	1	2370753	552721	474638	580112	763282	0.538	0.488	0.527	0.04	4
Xmeans	DE	2	2370753	511598	527518	625144	706493	0.492	0.450	0.483	0.04	3
Xmeans	DE	2	2370753	511598	527518	625144	706493	0.492	0.450	0.483	0.01	2
Xmeans	DE	2	2370753	511598	527518	625144	706493	0.492	0.450	0.483	0.01	4
Xmeans	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0.02	3
Xmeans	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0.01	2
Xmeans	CID	2	2370753	472267	578093	668214	652179	0.450	0.414	0.442	0	4
Xmeans	DTW	3	2370753	490987	554022	647715	678030	0.470	0.431	0.462	0.09	3
Xmeans	DTW	2	2370753	512494	526365	624163	707731	0.493	0.451	0.484	0.07	2
Xmeans	DTW	3	2370753	490588	554534	648151	677480	0.469	0.431	0.461	0.08	4
MDBC	DE	3	2370753	376280	701521	773326	519626	0.349	0.327	0.345	0.03	3
MDBC	DE	2	2370753	53072	1117132	1127260	73289	0.045	0.045	0.045	0.01	2
MDBC	DE	3	2370753	365328	715605	785320	504500	0.338	0.317	0.334	0.02	4
MDBC	CID	2	2370753	471769	578733	668759	651492	0.449	0.414	0.442	0.03	3
MDBC	CID	2	2370753	471769	578733	668759	651492	0.449	0.414	0.442	0.01	2
MDBC	CID	2	2370753	471769	578733	668759	651492	0.449	0.414	0.442	0.04	4
MDBC	DTW	1	2370753	552721	474638	580112	763282	0.538	0.488	0.527	0.04	3
MDBC	DTW	1	2370753	552721	474638	580112	763282	0.538	0.488	0.527	0.04	2
MDBC	DTW	1	33903495	552721	16241009	16346483	763282	0.033	0.033	0.033	0.04	4

Tabela 4.11: Quake Dataset

Algoritmo	Distância	Clusters	Pares	TP	FP	FN	TN	Precision	Recall	F measure	Tempo em seg	K Max
K means	DE	3	80359503	2557722	586839	234281	76980661	0.813	0.916	0.832	99.8	3
K means	DE	2	80359503	2547803	602841	244200	76964659	0.808	0.912	0.827	97.0	2
K means	DE	4	80359503	2560111	574583	231892	76992917	0.816	0.916	0.834	112.3	4
K means	CID	2	80359503	2555702	597184	236301	76970316	0.810	0.915	0.829	79.0	3
K means	CID	2	80359503	2538146	578093	253857	76924331	0.797	0.909	0.817	60.7	2
K means	CID	2	80359503	1971713	673891	820290	76893609	0.745	0.706	0.737	88.8	4
K means	DTW	2	80359503	2559102	574820	232901	76992680	0.816	0.916	0.834	115.05	3
K means	DTW	2	80359503	2561700	573293	230303	76994207	0.817	0.917	0.835	120.09	2
K means	DTW	2	80359503	2558199	575627	233804	76991873	0.816	0.916	0.834	132.09	4
Xmeans	DE	2	80359503	2463133	808736	328870	76758764	0.752	0.882	0.775	56.09	3
Xmeans	DE	2	80359503	2463028	808693	328975	76758807	0.752	0.882	0.775	23.0	2
Xmeans	DE	2	80359503	2463007	808818	328996	76758682	0.752	0.882	0.775	33.02	4
Xmeans	CID	2	80359503	2459702	12932283	332301	64635217	0.159	0.881	0.191	133.08	3
Xmeans	CID	2	80359503	2459939	12932273	332064	64635227	0.159	0.881	0.191	130.01	2
Xmeans	CID	4	80359503	2459939	12932273	332064	64635227	0.159	0.881	0.191	128.07	4
Xmeans	DTW	3	80359503	2555811	632291	236192	76935209	0.801	0.915	0.822	100.8	3
Xmeans	DTW	2	80359503	2555811	632291	236192	76935209	0.801	0.915	0.0822	95.09	2
Xmeans	DTW	4	80359503	2538455	1362508	253548	76204992	0.650	0.909	0.689	101.03	4
MDBC	DE	2	80359503	2528352	618960	263651	76948540	0.803	0.905	0.821	45.08	3
MDBC	DE	2	80359503	2528352	618960	263651	76948540	0.803	0.905	0.821	45.0	2
MDBC	DE	2	80359503	2528352	618960	263651	76948540	0.803	0.905	0.821	45.08	4
MDBC	CID	2	80359503	2526297	632294	668759	651492	0.799	0.904	0.818	53.09	3
MDBC	CID	2	80359503	2526297	632294	668759	651492	0.799	0.904	0.818	53.09	2
MDBC	CID	2	80359503	2526297	632294	668759	651492	0.799	0.904	0.818	54.1	4
MDBC	DTW	1	80359503	2545991	590302	246012	76977198	0.811	0.911	0.830	78.08	3
MDBC	DTW	1	80359503	2545991	590302	246012	76977198	0.811	0.911	0.830	78.08	2
MDBC	DTW	3	80359503	2532999	605310	259004	76962190	0.807	0.907	0.825	78.08	4

Tabela 4.12: Sido Dataset

Os resultados acima são todos os resultados obtidos após avaliarmos todos os datasets executando todos os algoritmos com suas respectivas medidas de distância.
Na próxima seção iremos apresentar uma análise detalhada de cada resultado obtido.

4.5 Análise

Nesta seção iremos apresentar uma analise detalhada dos resultados obtidos depois da tarefa de clusterizar ter sido executada nas séries temporais multivariadas apresentadas nesta dissertação. Para uma melhor compreensão dos melhores resultados obtidos, iremos fazer uma análise inicial dos melhores resultados obtidos em cada dataset usado. Como é possível verificar na tabela 1.1 não houve uma medida que sempre obteve o melhor resultado em todos os casos. Iremos analisar o por que de não haver uma medida de distância que sempre tenha uma performance superior as demais em todos os casos e para todos os datasets.
Como argumentamos na seção *nossos resultados*, a análise dos acertos da quantidade de clusters encontrados seria uma boa abordagem a ser realizada, porém uma vez que o algoritmo K-means necessariamente sempre distribui os objetos nos K clusters fornecidos como entrada. Para esta pesquisa a abordagem de analisar o número correto dos clusters encontrados não funcionaria como uma boa métrica, uma vez que o K-means sempre acertaria o número de clusters, uma vez que o K correto for fornecido, tendo em mente que o X-means recebe uma faixa de valores e então ele calcula o número adequado de clusters, a abordagem descrita acima não seria justa. Após esta pequena explicação, iremos proceder a analise aceitando a distribuição dos objetos nos clusters corretos como fator da análise.

Flare dataset: Tabela 4.2
Este dataset apresenta uma distribuição desbalanceada por objetos em cada classe, uma vez que temos duas classes com poucos exemplos. Este dataset apresenta um bom número de exemplos e é um dataset completo, ou seja, não existem valores faltando em seus atributos. A tarefa de clusterização é definida pela distribuição de cada objeto relativo a região solar onde ocorreu a observação. Como pode ser visualizado na tabela 1.1, o melhor resultado foi alcançado utilizando a distância euclidiana. Uma vez que neste dataset não existem valores faltantes, não apresenta uma variação nos tamanhos da

cada série temporal, a DE apresenta uma melhor performance uma vez que é reconhecidamente pela literatura, que uma vez ausentes as idiossincrasias citadas acima como ausentes, tanto o CID como o DTW se reduzem a uma DE custosa.

eeg Dataset: Tabela 4.3
Neste dataset apresentamos a primeira melhor performance alcançada pelo DTW. Trata-se de um dataset grande, com 14.979 exemplos e sua distribuição é quase totalmente balanceada, porém alguns atributos tem um valor relativamente grande enquanto outros um valor bem pequeno contribuindo assim para um grau de complexidade significativo o que inicialmente deveria favorecer ao CID. Porém a capacidade de resolver o problema da dimensionalidade *envelopando* a série temporal e calculando o seu alinhamento de custo mínimo, neste caso, supera a performance do CID neste dataset quando levado em conta a ausência de valores faltantes apresentados.

Arritmia Dataset: Tabela 4.4
Devido a este dataset ser extremamente desbalanceado, o CID apresentou a peculiaridade de clusterizar apenas o maior grupo e ignorar os demais, uma vez que isto representa mais que 50% dos exemplos. Devemos observar também que este dataset apresenta uma grande variação nos tamanhos das séries, bem como os valores faltantes como é possível observar na tabela 4.1 e a grande diferença de complexidade entre os valores apresentados. Vale a pena ressaltar que o DTW também clusterizou apenas o maior grupo de exemplos, porém devido à necessidade de preencher uma matriz de custo, seu tempo de execução foi muito superior ao CID. É pertinente informar que o CID obteve o mesmo desempenho tanto com o K-means, quanto com o X-means.

Adult Census Dataset: Tabela 4.5
Este dataset apresenta uma variação significativa nos tamanhos das séries apresentadas (tabela 4.1) e é extremamente desbalanceado. Com uma classe representando 69% dos exemplos apresentados, a diferença dimensional entre os exemplos é significante o suficiente para que o DTW tenha uma performance superior aos demais.

Internet usage dataset: Tabela 4.6
Como as séries temporais apresentadas neste dataset variam significativamente em sua complexidade, uma vez que um adulto necessariamente tem

um comportamento com uma complexidade diferente do apresentado por um adolescente, o que pode estar diretamente ligado a ausência de valores, o CID é capaz de reconhecer esta variação e clusterizar de forma mais apropriada as séries apresentadas.

Gesture phase Dataset: Tabela 4.7
O DTW obteve uma performance superior, a presença de um dataset desbalanceado, porém não em demasia, esta diretamente ligado ao tamanho das séries, uma vez que alguns tipos de movimentos tem atributos que outros não possuem. Desta forma, a abordagem do DTW de encontrar o alinhamento com o menor custo retorna o melhor resultado para este dataset. Uma vez que o CID e a DE não lançam mão desta tática apenas retornando o cálculo ponto a ponto e assim cirando uma grande diferença entre o tamanho das séries similares.

Otto group Kaggle Dataset: Tabela 4.8
Este dataset consiste de 61.870 exemplos, sendo assim um dataset grande e bastante desbalanceado. Não apresenta valores faltantes e possui um grande número de atributos (sendo 94 atributos e 1 adicional que é composto pela classe). O DTW , uma vez que a presença de vários atributos que são binários, tem a vantagem de ser executado em um dataset que pode ter uma instância de tamanho várias vezes menor do que o de maior tamanho. Desta forma, o DTW conseguiu uma performance considerávelmente alta.

Dow Jones Dataset: Tabela 4.9
Este é um dataset pequeno, com apenas 750 exemplos e completamente balanceado com 75 exemplos em cada uma das 10 classes existentes, não apresenta valores faltantes. Este dataset não apresenta variação nos tamanhos das séries porém apresenta na complexidade, mesmo assim a DE teve uma performance superior aos demais. Isto pode ser explicado pelo fato do dataset ser extremamente balanceado e ser pequeno.

HV Dataset: Tabela 4.10
Trata-se de um dataset não completo, ou seja, há falta de valores e é bem pouco desbalanceado, porém há uma grande variação na complexidade de cada instância, uma vez que este dataset é composto pelo número de casas vagas e novos proprietários de casas em UK e sua tarefa consiste em clusterizar estes objetos pela geolocalização, podemos facilmente concluir que a diferença de complexidade de um tipo casa vaga, casa recém-comprada, e entre as loc-

alizações é consideravelmente grande, o CID, sem nenhuma surpresa, obtém uma boa performance de clusters encontrados.

Quake Dataset: Tabela 4.11
Não há valores faltantes neste dataset, e trata-se de um dataset que não é muito grande, desta forma, é válido informar que a diferença entre alguns valores é bastante grande e os tamanhos das séries variam significativamente, uma vez que um tremor pode durar 3 segundos, como pode durar ate 27 segundos (como observado na tabela 4.1) por exemplo, devido a trabalhar com faixas de valores, quando um tremor é considerado fraco, alguns de seus atributos podem ter um valor muito baixo, ate mesmo igual a zero, e um tremor considerado forte terá um valor bastante alto, neste caso podemos verificar tanto uma variação da complexidade como de tamanho da série, porém neste caso específico a técnica de obter o caminho de custo mínimo leva vantagem sobre o CID, uma vez que a diferença de tamanho é mais significativa do que a diferença de complexidade envolvida. O DTW obtém a melhor performance neste dataset.

Estes resultados nos fornecem uma boa visão do por que, neste estudo, com os datasets e parâmetros utilizados neste estudo, não existe um método de medida de similaridade baseado em distância, ou mesmo uma combinação deste com os algoritmos de clusterização estudados nesta dissertação, que sempre vai obter um resultado melhor do que os demais para qualquer dataset e em qualquer tarefa. Podemos concluir claramente que cada medida irá ter sua melhor performance quando aplicada nos problema e datasets para o qual eles foram desenvolvidos. Desta forma não podemos esperar que uma medida desenvolvida para trabalhar em datasets que apresentem uma variação considerável em sua complexidade, tenha uma performance melhor que a DE quando estas singularidades não existam no dataset, da mesma maneira, não podemos esperar que a DE tenha uma performance superior em um dataset onde exista uma variação considerável entre os tamanhos das séries temporais submetidas a tarefa de clusterização. Vale a pena ressaltar que em alguns casos, como observado neste estudo, quando o dataset for muito pequeno, ou houver peculiaridades adicionais poderemos obter resultados inesperados, como no caso do dataset da Dow Jones. De qualquer forma isto deve ser visto como uma exceção que ocorre em apenas um pequeno número de casos.

Sido Dataset: Tabela 4.12 Dataset extremamente desbalanceado (452 exemplos pertencentes à classe 1 e 12226 pertencentes à classe -1), não ap-

resenta valores faltantes e apesar da complexidade entre séries poder ser significativa, o cálculo do CF do CID não vai variar muito, uma vez que as séries são binárias. Desta forma o DTW por poder corrigir a variação espacial das séries apresenta um resultado superior, não apenas com o Kmeans, mas como pode ser observado na Tabela 4.12, o DTW sempre apresenta resultados significativos para esta série específica.

5

Conclusão e Trabalhos Futuros

Nosso objetivo no início desta dissertação, foi de apresentar um estudo no qual pudéssemos testar alguns métodos de medida de similaridade baseados em distância juntamente com alguns algoritmos de clusterização, aplicando estes aos datasets compostos por séries temporais multivariadas afim de avaliar se existia uma medida que se sobreporia as demais ou se haveria uma combinação de algoritmo de clusterização / medida de similaridade que sempre obteria uma performance superior aos demais para todos os casos e em todos os datasets. Pelas avaliações dos resultados encontrados nesta dissertação, podemos efetivamente concluir que não existe tal coisa. Dado que uma vez os datasets e suas propriedades, assim como a tarefa em questão necessitam de abordagens diferenciadas e diretamente orientadas para suas necessidades.
As características apresentadas por cada série temporal tem um impacto massivo no resultado obtido por cada medida utilizada. Pelos resultados obtidos podemos tranquilamente ter a ressalva de outros estudos existentes na literatura (23, 13) e ter uma avaliação positiva de qual método de medida de similaridade baseado em distância irá proporcionar uma melhor performance para cada tipo de dataset e problema envolvido.

5.1 Trabalhos Futuros

Um problema intensamente interessante de ser trabalhado é poder determinar para quais tipos de problemas de clusterização qual seria a medida de distância mais apropriada, podendo assim criar um conjunto p de problemas de clusterização que sempre teriam o melhor resultando utilizando-se a mesma medida de distância m.
Uma outra oportunidade de pesquisa, seria tentar desenvolver um novo método de medida de similaridade baseado em distância que pudesse herdar todas as características dos métodos estudados e pudesse, finalmente, sempre obter uma performance superior aos demais para qualquer dataset em qualquer tipo de problema.

Referências bibliográficas

[1] PRAJWALA, T.; SANGEETA, V.. **Comparative analysis of em clustering algorithm and density based clustering algorithm using weka tool.**

[2] HARTIGAN, J. A.; WONG, M. A.. **Algorithm as 136: A k-means clustering algorithm**. Applied statistics, p. 100–108, 1979.

[3] LLOYD, S. P.. **Least squares quantization in pcm**. Information Theory, IEEE Transactions on, 28(2):129–137, 1982.

[4] BERNDT, D. J.; CLIFFORD, J.. **Using dynamic time warping to find patterns in time series.** In: KDD WORKSHOP, volumen 10, p. 359–370. Seattle, WA, 1994.

[5] BERTSEKAS, D. P.; BERTSEKAS, D. P.; BERTSEKAS, D. P. ; BERTSEKAS, D. P.. **Dynamic programming and optimal control**, volumen 1. Athena Scientific Belmont, MA, 1995.

[6] DAS, G.; GUNOPULOS, D. ; MANNILA, H.. **Finding similar time series**. In: PRINCIPLES OF DATA MINING AND KNOWLEDGE DISCOVERY, p. 88–100. Springer, 1997.

[7] PELLOM, B. L.; HANSEN, J. H.. **An efficient scoring algorithm for gaussian mixture model based speaker identification**. Signal Processing Letters, IEEE, 5(11):281–284, 1998.

[8] PELLEG, D.; MOORE, A. W. ; OTHERS. **X-means: Extending k-means with efficient estimation of the number of clusters.** In: ICML, p. 727–734, 2000.

[9] KALPAKIS, K.; GADA, D. ; PUTTAGUNTA, V.. **Distance measures for effective clustering of arima time-series**. In: DATA MINING, 2001. ICDM 2001, PROCEEDINGS IEEE INTERNATIONAL CONFERENCE ON, p. 273–280. IEEE, 2001.

[10] BROCKWELL, P. J.. **Introduction to time series and forecasting**, volumen 1. Taylor & Francis, 2002.

[11] RATANAMAHATANA, C. A.; KEOGH, E.. **Making time-series classification more accurate using learned constraints**. SIAM, 2004.

[12] LIAO, T. W.. **Clustering of time series data a survey**. Pattern recognition, 38(11):1857–1874, 2005.

[13] KEOGH, E.; RATANAMAHATANA, C. A.. **Exact indexing of dynamic time warping**. Knowledge and information systems, 7(3):358–386, 2005.

[14] HRIPCSAK, G.; ROTHSCHILD, A. S.. **Agreement, the f-measure, and reliability in information retrieval**. Journal of the American Medical Informatics Association, 12(3):296–298, 2005.

[15] ISHIOKA, T.. **An expansion of x-means for automatically determining the optimal number of clusters**. In: PROCEEDINGS OF INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE, p. 91–96, 2005.

[16] KLEINBERG, J.; TARDOS, É.. **Algorithm design**. Pearson Education India, 2006.

[17] CAO, F.; ESTER, M.; QIAN, W. ; ZHOU, A.. **Density-based clustering over an evolving data stream with noise.** In: SDM, volumen 6, p. 328–339. SIAM, 2006.

[18] MÜLLER, M.. **Information retrieval for music and motion**, volumen 2. Springer, 2007.

[19] MÜLLER, M.. **Dynamic time warping**. Information retrieval for music and motion, p. 69–84, 2007.

[20] AMIGÓ, E.; GONZALO, J.; ARTILES, J. ; VERDEJO, F.. **A comparison of extrinsic clustering evaluation metrics based on formal constraints**. Information retrieval, 12(4):461–486, 2009.

[21] JAIN, A. K.. **Data clustering: 50 years beyond k-means**. Pattern recognition letters, 31(8):651–666, 2010.

[22] FU, T.-C.. **A review on time series data mining**. Engineering Applications of Artificial Intelligence, 24(1):164–181, 2011.

[23] BATISTA, G. E.; WANG, X. ; KEOGH, E. J.. **A complexity-invariant distance measure for time series.** In: SDM, volumen 11, p. 699–710, 2011.

[24] KRIEGEL, H.-P.; SCHUBERT, E. ; ZIMEK, A.. **Evaluation of multiple clustering solutions.** In: MULTICLUST@ ECML/PKDD, p. 55–66. Citeseer, 2011.

[25] KASHYAP, S.; LEE, M. L. ; HSU, W.. **Similar subsequence search in time series databases**. In: DATABASE AND EXPERT SYSTEMS APPLICATIONS, p. 232–246. Springer, 2011.

[26] FASANG, A. E.; LIAO, T. F.. **Visualizing sequences in the social sciences relative frequency sequence plots**. Sociological Methods & Research, p. 0049124113506563, 2013.

[27] MADEO, R. C.; LIMA, C. A. ; PERES, S. M.. **Gesture unit segmentation using support vector machines: segmenting gestures from rest positions**. In: PROCEEDINGS OF THE 28TH ANNUAL ACM SYMPOSIUM ON APPLIED COMPUTING, p. 46–52. ACM, 2013.

[28] BROWN, M. S.; PELOSI, M. J. ; DIRSKA, H.. **Dynamic-radius species-conserving genetic algorithm for the financial forecasting of dow jones index stocks**. In: MACHINE LEARNING AND DATA MINING IN PATTERN RECOGNITION, p. 27–41. Springer, 2013.

[29] WAGNER, P. K.; PERES, S. M.; MADEO, R. C. B.; LIMA, C. A. D. M. ; FREITAS, F. D. A.. **Gesture unit segmentation using spatial-temporal information and machine learning**. In: THE TWENTY-SEVENTH INTERNATIONAL FLAIRS CONFERENCE, 2014.

APÊNDICE A

Datasets

A.1
Maiores Informações sobre os Datasets usados

Neste apêndice, estaremos fornecendo algumas informações adicionais sobre os datasets utilizados.

Flare dataset:

- **Características** Multivariado, Série Temporal
- **Atributos** Categórico
- **Número de instâncias** 1066
- **Número de Atributos** 11
- **Valores faltando?** Não
- **Área** Científica
- **Data de publicação** 01-03-1989
- **Proprietário original** Gary Bradshaw

Dataset que apresenta as variações das manchas solares em cada região solar

eeg Dataset:

- **Características** Multivariado, Série Temporal
- **Atributos** Categórico, Integer, Real
- **Número de instâncias** 122
- **Número de Atributos** 4
- **Valores faltando?** Sim

- **Área** Médica
- **Data de publicação** 13-10-1999
- **Proprietário original** Henri Begleiter Neurodynamics Laboratory, State University of New York Health Center Brooklyn, New York

Este dataset foi composto após um longo estudo sobre as predisposições geneticamente correlatas com o alcoolismo. É composto pelas medidas obtidas com 64 eletrodos colocados na cabeça dos pacientes e cada exemplo foi inferido a 256 Hz (3.9-msec epoch) por cada 1 segundo.
Para estes testes havia dois grupos distintos, os alcoólatras e o grupo de controle, cada um foi submetido à estímulos simples (S1) ou estimulo duplo (S1 e S2).
Maiores informações podem ser encontradas em (Standard Electrode Position Nomenclature, American Electroencephalographic Association 1990). Zhang et al. (1995) onde todo o processo de como os dados foram coletados esta descrito.

Arritmia dataset:

- **Características** Multivariada, Série Temporal
- **Atributos** Categórico, Integer, Real
- **Número de instâncias** 452
- **Número de Atributos** 279
- **Valores faltando?** Sim
- **Área** Médica
- **Data de publicação** 01-01-1998
- **Proprietário original** H. Altay Guvenir, PhD., Bilkent University, Department of Computer Engineering and Information Science, 06533 Ankara, Turkey

Derivado do estudo H. Altay Guvenir, O objetivo é distinguir entre a presença ou ausência de arritmia cardíaca em 3 grupos etários distintos.

Adult Census dataset:

- **Características** Multivariada, Série Temporal
- **Atributos** Categórico, Integer
- **Número de instâncias** 48842
- **Número de Atributos** 14
- **Valores faltando?** Sim
- **Área** Social
- **Data de publicação** 01-05-1996
- **Proprietário original** Ronny Kohavi and Barry Becker Data Mining and Visualization, Silicon Graphics

Extraído do banco de dados do Census em 1994 por Barry Becker.

Final General dataset:

- **Características** Multivariado, Série Temporal
- **Atributos** Categórico, Integer, Real
- **Número de instâncias** 10104
- **Número de Atributos** 72
- **Valores faltando?** Não
- **Área** Social
- **Data de publicação** 30-06-1998
- **Proprietário original** The UCI KDD Archive, Information and Computer Science, University of California, Irvine, Irvine, CA 92697-3425

Este dataset foi usado pela American Statistical Association Statistical Graphics and Computing Sections em 1999. Este dataset demonstra o uso da internet por cada faixa etária.

Gesture Phase dataset:

- **Características** Multivariada, Sequencial, Série Temporal

- **Atributos** Real
- **Número de instâncias** 9900
- **Número de Atributos** 50
- **Valores faltando?** Não
- **Área** Médica
- **Data de publicação** 18-06-2014
- **Proprietário original** Renata Cristina Barros Madeo (Madeo, R. C. B.), Priscilla Koch Wagner (Wagner, P. K.), Sarajane Marques Peres (Peres, S. M.)

Dataset composto pelas características extraídas de 7 vídeos de pessoas em movimento, o alvo é estudar cada fase de segmentação dos gestos. (27, 29)

Otto Group Challenge Dataset:

- **Características** Multivariada, Série Temporal
- **Atributos** Integer
- **Número de instâncias** 61.870
- **Número de Atributos** 95
- **Valores faltando?** Não
- **Área** Business
- **Data de publicação** 17-03-2015
- **Proprietário original** Otto Group

Dataset apresentado como parte do desafio do Grupo Otto

Dow Jones Index Dataset:

- **Características** Série Temporal
- **Atributos** Integer, Real
- **Número de instâncias** 750

- **Número de Atributos** 16
- **Valores faltando?** Não
- **Área** Business
- **Data de publicação** 23-10-2014
- **Proprietário original** Dr. Michael Brown, University of Maryland University College

Preço de ações coletados durante um ano (28) .

HV dataset:

- **Características** Série Temporal
- **Atributos** Integer, Real
- **Número de instâncias** 8235
- **Número de Atributos** 7
- **Valores faltando?** Não
- **Área** Business
- **Data de publicação** 01-12-2012
- **Proprietário original** U.K. Census Bureau

Este dataset é constituído de parte das informações coletadas pelo censo UK.

Quake Dataset:

- **Características** Série Temporal
- **Atributos** Real, Integer
- **Número de instâncias** 2178
- **Número de Atributos** 4
- **Valores faltando?** Não
- **Área** Científica

- **Data de publicação** 13-03-2005
- **Proprietário original** Bilkent University Function Approximation Repository

Composto por informações que descrevem 2178 terremotos.

Sido Dataset:

- **Características** Série Temporal
- **Atributos** Integer, Binário
- **Número de instâncias** 12678
- **Número de Atributos** 1
- **Valores faltando?** Não
- **Área** Médica
- **Data de publicação** 27-05-2012
- **Proprietário original** Não declarado

SIDO (Simple Drug Operation Mechanism) Contém descrição de moléculas, que foram testadas contra o vírus HIV. Os grupos alvos para este dataset são: +1 (molécula ativa), -1 (inativa). A tarefa de clusterização para este dataset é identificar a causa da atividade ou inatividade das moléculas entre seus descritores. Desta forma auxiliando aos cientistas a criarem novos compostos, retendo a atividade molecular, porém adicionando outras características desejáveis, como por exemplo drogas menos tóxicas ou mais fáceis de administrar.