Uma análise experimental de métricas de similaridade na classificação de séries temporais

Idilio Drago[1], **Flávio Miguel Varejão**[1]

[1]Departamento de Informática, Centro Tecnológico –
Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari s/n – CEP 29060-900 – Vitória – ES – Brasil

{idrago,fvarejao}@inf.ufes.br

Abstract. *This paper evaluates some techniques for the time series classification problem. Many distance measures have been proposed as an alternative to the Euclidean distance in the nearest neighbour classifier. We performed an experimental analysis and identified a list of those measures with some evidence of better classification accuracy than the Euclidean, based on the Wilcoxon ranks test for paired data.*

Resumo. *Neste trabalho, avaliamos alternativas para o problema de classificação de séries temporais. Com o objetivo de aumentar a precisão do classificador do vizinho mais próximo, diversas métricas têm sido propostas como opção à distância Euclidiana. Avaliamos algumas opções e, baseado no teste de Wilcoxon para dados pareados, produzimos uma relação daquelas em que há evidências de melhoria na precisão do classificador.*

1. Introdução

O problema de classificação é um dos mais comuns no contexto da aprendizagem supervisionada. Neste tipo de problema, deseja-se predizer a classe de um objeto desconhecido a partir de suas características e de exemplos anteriores, com classificação conhecida. Um caso especial de classificação ocorre quando, dentre as características que representam os objetos, algumas possuem relação temporal. Neste caso, os atributos com relação temporal são apresentados como uma seqüência de valores (geralmente números), cada qual simbolizando a característica em um determinado instante de tempo. Diversas soluções têm sido empregadas em problemas deste tipo. Em [Yamada et al. 2003], é proposta uma alteração do algoritmo de construção de árvores de decisão para tratamento especial de características temporais. Com maior freqüência, utiliza-se a combinação entre extração de características estáticas e treinamento de classificadores convencionais [Nanopoulos et al. 2001, Mörchen 2003].

Para o caso ainda mais restrito, em que os dados são compostos somente por uma única característica temporal (problemas unidimensionais), uma abordagem amplamente utilizada é classificar através do algoritmo do vizinho mais próximo, porém com métricas mais apropriadas à forma temporal dos dados. Várias métricas têm sido propostas como alternativa à distância Euclidiana e um resumo de opções pode ser encontrado em [Antunes and Oliveira 2001] e [Savary 2002]. Os artigos, porém, não avaliam a precisão do classificador construído com as métricas alternativas. Em [Keogh and Kasetty 2003], trabalho em que os autores defendem a criação de repositórios para reprodução de resultados em mineração de dados temporais, há uma avaliação de onze métricas com a conclusão de que nenhuma delas é superior à solução tradicional (distância Euclidiana) em dois problemas exemplos.

Neste trabalho, investigamos a precisão do classificador construído com algumas das métricas apresentadas em [Antunes and Oliveira 2001] e [Savary 2002]. No caso daquelas avaliadas também em [Keogh and Kasetty 2003], comparamos o nosso resultado, obtido a partir de um número maior de problemas exemplos, com o reportado anteriormente. Em um dos casos (distância de edição), obtivemos resultado diferente do reportado. O restante deste artigo está assim organizado: a seção 2 apresenta as diferentes métricas implementadas e avaliadas; a seção 3 descreve uma estratégia simples de extração e seleção de características que será contraposta ao resultado obtido com as métricas; a seção 4 descreve os dados, o método experimental empregado e os resultados obtidos; a seção 5 apresenta as conclusões e os possíveis trabalhos futuros.

2. Medidas de similaridade entre seqüências

Nesta seção apresentamos as métricas implementadas para determinar a similaridade entre seqüências. Como elas definirão o vizinho mais próximo de séries com classe desconhecida, interessam medidas que são minimizadas quanto mais "semelhantes" forem as seqüências. Nos experimentos realizados, as métricas foram comparadas à forma tradicional do algoritmo do vizinho mais próximo. Neste caso, convém apresentar a formulação da distância Euclidiana. Sejam duas seqüências $\vec{x}$ e $\vec{y}$ de tamanho igual a d[1]. A distância Euclidiana entre elas é definida como:

$$\mathcal{D}(\vec{x},\vec{y}) = (\sum_{i=0}^{n-1} |x_i - y_i|^2)^{1/2} \tag{1}$$

As principais críticas contra a abordagem tradicional estão relacionadas à sua alta sensibilidade a ruídos, a pequenas variações de fase entre as seqüências, a translações e escalamentos horizontais etc. [Agrawal et al. 1995]. A maior parte das métricas apresentadas a seguir se diferencia por tentar melhorar algum destes aspectos.

2.1. *DTW - Dynamic time warping*

O algoritmo *DTW*, amplamente utilizado em problemas de reconhecimento de voz e aplicado pela primeria vez no contexto mais geral de seqüências temporais por [Berndt and Clifford 1994], se diferencia da distância Euclidiana por determinar a similaridade após alinhar as séries no eixo do tempo. A idéia central do algoritmo é alinhar as séries através de uma função de mapeamento não-linear representada pelo caminho W que minimiza a distância total entre $\vec{x}$ e $\vec{y}$. Tal caminho é definido em uma matriz quadrada de dimensões d como:

$$W = w_0, w_2, \ldots, w_k, \ldots, w_{K-1} \tag{2}$$

onde $w_k = (i,j)$, isto é, w_k são as coordenadas das células da matriz visitadas pelo caminho. Cada célula (i,j) da matriz é preenchida com $\delta(i,j) = (x_i - y_j)^2$ e o peso

[1]Nenhuma das medidas implementadas requer que as séries sejam de mesmo tamanho, porém este é o caso em todas as bases de dados avaliadas.

de um caminho W é definido como o somatório dos valores $\delta(w_k)$ para todas as células que o compõem. O valor da distância total entre $\vec{x}$ e $\vec{y}$ será o menor peso dentre todos os caminhos existentes, isto é:

$$\mathcal{D}(\vec{x},\vec{y}) = \min_{W} \sum_{k=0}^{K-1} \delta(w_k) \tag{3}$$

Um caminho W é considerado válido se atender as seguintes condições:

1. Se $w_k = (a,b)$ e $w_{k-1} = (a',b')$, então $a - a' \geq 0$ e $b - b' \geq 0$. Tal condição garante que o caminho apresentará os pontos sempre em ordem crescente ao longo do eixo do tempo.
2. Se $w_k = (a,b)$ e $w_{k-1} = (a',b')$, então $a - a' \leq 1$ e $b - b' \leq 1$. Tal condição exige que o caminho W, a partir de uma célula qualquer, prossiga para outra adjacente a esta (incluindo a diagonal).
3. Limites: $w_1 = (0,0)$ e $w_K = (d-1, d-1)$. Tal condição exige que o caminho comece e termine nas diagonais opostas da matriz.
4. Sendo $w_k = (i,j)$, então $|i-j| \leq r$. Tal condição exige que W não se afaste mais de r unidades da diagonal da matriz.

O caminho W de menor peso pode ser encontrado de maneira eficiente usando programação dinâmica para avaliar a seguinte recorrência:

$$\gamma(i,j) = \delta(x_i, y_j) + min \begin{cases} \gamma(i, j-1) \\ \gamma(i-1, j) \\ \gamma(i-1, j-1) \end{cases} \tag{4}$$

Após o preenchimento completo da matriz, a distância entre as seqüências será:

$$\mathcal{D}(\vec{x},\vec{y}) = \gamma(d-1, d-1) \tag{5}$$

É importante notar que nem todas as células da matriz precisam ser avaliadas na recorrência, dado que o parâmetro r limita os caminhos válidos a uma janela em torno da diagonal. Além disto, quando $r = 0$, a distância retornada pelo algoritmo *DTW* será a distância Euclidiana. Assim, o valor de r é o parâmetro livre do algoritmo, podendo influenciar no desempenho da classificação. Alguns trabalhos já têm explorado a comparação entre *DTW* e a distância Euclidiana, inclusive do ponto de vista do ajuste de r. Procuramos seguir os mesmos ajustes de [Xi et al. 2006], que reporta resultados positivos em favor do algoritmo *DTW*.

2.2. Alinhamento de *strings*

Uma segunda abordagem consiste em converter as séries para seqüências compostas por símbolos de um alfabeto pré-definido e calcular a similaridade a partir desta nova representação. Alguns trabalhos exploram a distância de edição como métrica de similaridade. Por exemplo, [Bozkaya et al. 1997] emprega uma alteração da medida para casamento dos valores em séries de números inteiros. Ao contrário desta alternativa, preferimos discretizar as seqüências em N intervalos iguais, com o valor original da seqüência sendo convertido para o valor simbólico igual a parte inteira de

$N * (x_i - MIN)/(MAX - MIN)$, onde MAX e MIN são, respectivamente, o maior e o menor valor em todas as séries da etapa de treinamento.

Considerando que os valores das séries são simbólicos (*strings*), a distância de edição é definida como a quantidade mínima de operações necessárias para transformar uma série em outra qualquer. As operações permitidas para tal fim são a substituição de um caractere e a adição ou remoção de caracteres [Theodoridis and Koutroumbas 2006]. Pode-se, arbitrariamente, definir o custo mais conveniente para cada operação. Usamos custo unitário nos três casos. O algoritmo de cálculo da distância de edição é bastante semelhante ao algoritmo *DTW*. O cálculo da distância de edição é realizado por meio de uma matriz quadrada de dimensões $d + 1$, na qual as mesmas condições impostas ao algoritmo *DTW* são válidas. O preenchimento da matriz é feito através de programação dinâmica, avaliando a seguinte recorrência:

$$\begin{aligned} \gamma(i,0) &= \gamma(0,j) = j \\ \gamma(i,j) &= min \begin{cases} \gamma(i-1,j)+1 \\ \gamma(i,j-1)+1 \\ \gamma(i-1,j-1)+t(i,j) \end{cases} \end{aligned} \tag{6}$$

com $t(i,j) = 0$, se $x_i = y_j$ e $t(i,j) = 1$, caso contrário. Após o preenchimento completo da matriz, a distância entre as seqüências será o valor de $\gamma(d,d)$.

Outra abordagem semelhante é calcular a distância entre as seqüências usando o algoritmo da maior sub-seqüência comum (*LCSS* - *Longest common subsequence*). Neste caso, deseja-se maximizar o total de casamento de caracteres entre as duas seqüências. A *LCSS* é calculada através da seguinte recorrência:

$$\begin{aligned} \gamma(i,0) &= \gamma(0,j) = 0 \\ \gamma(i,j) &= \begin{cases} \gamma(i-1,j-1)+1 & \text{se } x_{i-1} = y_{i-1} \\ \gamma(i-1,j) & \text{se } \gamma(i-1,j) \geq \gamma(i,j-1) \\ \gamma(i,j-1) & \text{caso contrário} \end{cases} \end{aligned} \tag{7}$$

Porque queremos uma métrica para minimização, a distância entre duas seqüências será:

$$\mathcal{D}(\vec{x},\vec{y}) = d - \gamma(d,d) \tag{8}$$

2.3. Filtro de ruídos e coeficiente de correlação

Coeficiente de correlação é o nome dado a uma medida estatística que determina a relação linear entre duas variáveis aleatórias. É definido como:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E((X-\mu_X)(Y-\mu_Y))}{\sqrt{E(X^2)-E^2(X)}\sqrt{E(Y^2)-E^2(Y)}} \tag{9}$$

onde $\mu_X = E(X)$ e $\mu_Y = E(Y)$. O coeficiente de correlação possui valor no intervalo $[-1, 1]$, sendo que os extremos implicam relação linear perfeita (no sentido positivo ou negativo) e zero implica que não há relação linear entre as variáveis. No contexto de casamento de padrões, a medida é usada com freqüência quando deseja-se encontrar um padrão pré-estabelecido em um conjunto de dados que pode conter o padrão (por exemplo, um objeto em uma imagem) [Theodoridis and Koutroumbas 2006]. Como queremos métricas a minimizar, definimos a similaridade entre duas seqüências como:

$$\mathcal{D}(\vec{x}, \vec{y}) = (1 - \rho_{\vec{x},\vec{y}}) \tag{10}$$

Notar que a métrica considera boa apenas a correlação positiva. Tal decisão justifica-se porque, em alguns problemas, o padrão a ser reconhecido pode estar relacionado à correlação negativa - por exemplo, em casos em que há tendências de crescimento e decaimento associadas às classes em questão. Para tornar a medida menos sensível a ruídos, foi aplicado um filtro removedor de ruídos de alta freqüência antes do cálculo da correlação. Utilizamos um filtro de médias móveis com janela de tamanho 7. Para efeito de comparação, na seção de resultado estão listados também os resultados da classificação usando o filtro de ruídos com a distância Euclidiana.

2.4. Medidas em domínios transformados e outras alternativas

Nesta seção apresentamos, resumidamente, alternativas que podem ser consideradas inferiores de modo geral, apesar de terem superado a distância Euclidiana em alguns casos. A primeira delas, chamada *Landmarks* [Perng et al. 2000], tenta medir a semelhança entre séries a partir dos *pontos de referência* das seqüências. Dos vários tipos de pontos existentes, apenas os de máximos e mínimos locais foram usados. A solução ainda engloba um método de suavização, chamado de *Princípio da Mínima Distância/Percentual* e medidas de similaridade entre vetores de *Landmarks*. Listaremos apenas os resultados da melhor alternativa: usar o vetor de pontos suavizados como entrada para o cálculo da distância de edição.

O trabalho de [Agrawal et al. 1993], o mais citado em relação a recuperação eficiente de séries, apresenta uma forma de indexação de seqüências baseada na transformada de Fourier. A idéia central é usar apenas os primeiros coeficientes do domínio transformado para construção de um índice e executar as consultas em duas etapas: a primeira usa apenas os primeiros coeficientes e retorna um super-conjunto da resposta desejada; a segunda filtra o resultado no espaço Euclidiano. A abordagem acima é exata, ou seja, produz o mesmo resultado que a distância Euclidiana, porém de maneira eficiente. Executamos experimentos considerando apenas os primeiros coeficientes de Fourier como representação das seqüências. Gostaríamos de verificar se, do ponto de vista da precisão da classificação, a segunda etapa das consultas poderia ser negligenciada. Como mostrado na seção 4, a idéia não foi bem sucedida. Tentamos a mesma abordagem usando a transformada *Wavelet* de *Daubechies* com resultados semelhantes.

3. Extração e seleção de características

A principal estratégia de classificação alternativa às métricas especializadas ao caso temporal é a junção da extração (e seleção) de características estáticas a um classificador convencional. Como forma de estabelecer mais um parâmetro de comparação para o resultado obtido com as métricas apresentadas nas seções anteriores, um classificador semelhante ao proposto em [Nanopoulos et al. 2001] foi implementado. Tal estratégia permite que técnicas mais sofisticadas sejam treinadas com os dados. Porém, como deseja-se apenas um padrão de comparação, o classificador do vizinho mais próximo foi novamente utilizado. Além disso, o conjunto de características extraídas das séries foi formado somente por características relativamente triviais (como informações estatísticas de média, desvios etc.) ou por aquelas amplamente encontradas na literatura (como os coeficientes de Fourier). Como não há certeza que estas características são úteis ao algoritmo de classificação, uma etapa de seleção foi realizada. O Algoritmo 1 descreve o processo de treinamento e consulta realizado nesta abordagem.

Algoritmo 1 Estimativa do erro do classificador construído com características extraídas.

Pré-condição: $X \neq \emptyset$ {conjunto de instâncias}

Faça $Y \leftarrow X_1, X_2, ..., X_n$ {sub-conjuntos disjuntos e aleatórios de X para validação cruzada.}
para todo $Y_i \in Y$ **faça**
  $T \leftarrow Y - \{Y_i\}$ {Conjunto de treinamento da rodada.}
  $E \leftarrow extract(T)$ {*extract* extrai características das instâncias.}
  $N \leftarrow nomalize(E)$ {*nomalize* normaliza as características.}
  $S \leftarrow select(N)$ {*select* seleciona somente as características relevantes.}
  Execute o algoritmo de treinamento em S.
  Use Y_i para aferir o desempenho do classificador treinado.
fim para

No Algoritmo 1, o conjunto de teste em cada etapa da validação não é usado na seleção das características - dessa forma não há super-ajuste. De cada instância de validação são extraídas apenas as características selecionadas durante a etapa de treinamento. Na seleção de características foi implementada a heurística da busca seqüencial incremental (*SFS - Sequential Forward Selection*) [Theodoridis and Koutroumbas 2006]. Além disso, o procedimento de normalização transforma os valores das características para o intervalo $[0, 1]$ de maneira linear, conforme descrito em [Pyle 1999], e os mesmos limites de máximo e mínimo são usados para normalização das instâncias no conjunto de teste. Os seguintes procedimentos de extração foram implementados:

1. Tendência linear - gera os coeficientes $(c0, c1)$ da regressão linear do modelo $Y = c0 + c1X$, usando a seqüência de entrada como valores Y e números naturais como coordenadas X.
2. Estatísticas - produz as estatísticas básicas sobre as séries contendo: média, desvio padrão, desvio absoluto, assimetria, curtose e autocorrelação (apenas de intervalo 1).
3. Estatísticas de área - produz estatísticas descritivas sobre a concentração do módulo dos valores das curvas em relação ao eixo X.
4. Histograma - produz o histograma (de 10 intervalos) das seqüências.
5. Wavelets - extrai os coeficientes da transformada *wavelet*, usando a base de Daubechies (8 como tamanho do filtro).
6. Fourier - extrai os coeficientes da transformada de Fourier.

Todas as funções acima estão disponíveis na biblioteca *GNU Scientific Library*. Em [Galassi et al. 2006] estas funções são apresentadas juntamente com a documentação da biblioteca.

4. Avaliação experimental

O objetivo principal da avaliação experimental é comparar a precisão do classificador construído com cada uma das métricas à precisão do classificador tradicional (distância Euclidiana). O foco dos experimentos realizados não é determinar a melhor alternativa para um problema especial e sim verificar a expectativa de que cada métrica classificará melhor do que a distância Euclidiana. Como queremos comparar o desempenho de duas alternativas em diversos problemas distintos, utilizamos a metodologia experimental recomendada por [Demsar 2006]. O procedimento de avaliação consiste em desenvolver um teste para verificar a hipótese nula de que as taxas de erro de dois classificadores seguem a mesma distribuição de probabilidade. Como não há nenhuma indicação de que o erro siga uma distribuição especial, optamos pelo teste não-paramétrico de *Wilcoxon* para dados pareados.

O teste de *Wilcoxon* é realizado arranjando as observações das diferenças entre as taxas de erro em uma lista ordenada de acordo com os seus valores absolutos. Para cada elemento na lista, é atribuído um número de ordem (posto). Se a hipótese nula for verdadeira, as taxas de erro seguem a mesma distribuição de probabilidade e é correto assumir que os postos das diferenças positivas e das diferenças negativas estarão igualmente distribuídos, ao invés de concentrados entre os menores (ou maiores) valores. Para realização do teste é necessária uma amostra aleatória de problemas de classificação, na qual o valor esperado da taxa de erro deve ser estimado para cada problema. Notar que o teste de *Wilcoxon* não utiliza a variância do erro em um problema específico, assim, a forma tradicional de validação cruzada já é suficiente para estimar a média dos erros. O valor mínimo de significância no qual é possível rejeitar a hipótese nula (*p-value*) é determinado por tabelas de probabilidades, através da soma dos postos de uma das alternativas e do tamanho da amostra.

Montamos o nosso experimento a partir de 11 problemas de classificação disponibilizados por outros autores. Por limitações de espaço, listaremos apenas o nome da base e a referência para o local no qual é possível obter mais informações sobre os dados. As bases usadas no experimento foram: *Control Chart* de [Hettich and Bay 1999]; *Two patterns*, *Cbf-tr* e *Cbf* de [Geurts 2002]; *Leaf*, *Gunx*, *Face Norm.* e *Trace* de [Keogh et al. 2006]; *Wafer* e *Ecg-svdb* de [Olszewski 2001]; *Motor* de [Povinelli et al. 2004]. Na tabelas 1, 2 e 3 estão listadas as taxas de erro obtidas pelas diversas alternativas avaliadas. Em cada tabela é apresentado o resultado da distância Euclidiana e o *p-value* da métrica em comparação com a Euclidiana.

Na Tabela 1 são mostradas as taxas de erro das métricas que podem ser consideradas melhores que a distância Euclidiana com significância de 5%. A partir deste resultado, é possível concluir que, dado um problema qualquer, é muito provável que as três medidas serão mais bem sucedidas do que a distância Euclidiana. Não é correto, porém, comparar as métricas entre si de acordo com o *p-value* obtido.

A Tabela 2 lista a taxa de erro das alternativas em que não foi possível rejeitar a hipótese nula. Na coluna *Extração* é mostrado o resultado do classificador construído com características extraídas das séries. Em relação ao desempenho desta abordagem, fica claro que é necessário extrair mais características para obter desempenho satisfatório em problemas variados. Isso valoriza a alternativa das métricas, principalmente porque a complexidade computacional da seleção de características é maior. Sobre os outros

Tabela 1. Taxa de erro por base. As três medidas podem ser consideradas melhores que a distância Euclidiana com nível de significância de 5%.

Base	Euclidiana	*DTW*	D. Edição	*LCSS*
Cbf-tr	1.56	0.44	1.28	0.74
Cbf	0.46	0.14	0.48	0.30
Control Chart	1.67	0.17	1.00	1.00
Trace	11.50	2.00	2.50	2.00
Wafer	2.43	1.93	1.18	1.68
Ecg-svdb	12.00	11.00	11.50	16.50
Motor	6.67	18.10	5.71	2.14
Face Norm.	6.29	4.55	0.91	1.82
Gunx	6.50	1.00	3.00	4.00
Leaf	34.62	8.85	2.50	5.21
Two patterns	1.42	0.00	0.02	0.00
p-value		4.09%	0.44%	2.62%

resultados, vale destacar que houve melhoria, aparentemente, devido ao filtro de ruídos. É possível que as duas medidas passem a ter resultado consistentemente superior com mais alguns aperfeiçoamentos, como os de [Agrawal et al. 1995], por exemplo.

Tabela 2. Não é possível rejeitar a hipótese nula de que as métricas são equivalentes à distância Euclidiana.

Base	Euclidiana	Euc. + Filtro	Correlação	Extração
Cbf-tr	1.56	0.70	0.20	1.94
Cbf	0.46	0.20	0.02	0.37
Control Chart	1.67	0.67	3.50	0.67
Trace	11.50	9.50	10.50	0.00
Wafer	2.43	1.59	1.84	1.17
Ecg-svdb	12.00	13.50	10.50	19.00
Motor	6.67	3.10	28.10	6.51
Face Norm.	6.29	11.74	9.02	9.92
Gunx	6.50	6.00	6.00	10.00
Leaf	34.62	30.57	31.21	18.08
Two patterns	1.42	0.42	0.42	1.18
p-value		18.23%	65.66%	65.66%

Na Tabela 3 estão listadas as métricas que foram inferiores à distância Euclidiana com 5% de significância. Exceto em situações especiais, tais medidas são piores que a solução padrão e não devem ser utilizadas se o desejado for minimizar a taxa de erro.

5. Conclusões

Neste trabalho apresentamos um estudo experimental do desempenho de diversas métricas de similaridade em um problema de classificação com dados temporais. Avaliamos a expectativa de que as alternativas à distância Euclidiana classificam melhor em um problema qualquer. Mostramos que *DTW*, *LCSS* e *distância de edição* são métricas com desempenho, em geral, superior. No caso da última, o resultado contraria o apresentado

Tabela 3. Casos em que é possível afirmar que a distância Euclidiana é melhor com significância de 5%.

Base	Euclidiana	*Landmarks*	k-Wavelets	k-Fourier
Cbf-tr	1.56	3.50	1.92	1.72
Cbf	0.46	1.44	0.16	0.50
Control Chart	1.67	2.67	0.33	4.17
Trace	11.50	11.00	15.50	19.00
Wafer	2.43	2.01	1.26	1.59
Ecg-svdb	12.00	29.00	13.50	14.00
Motor	6.67	18.10	21.19	10.48
Face Norm.	6.29	23.11	19.85	20.46
Gunx	6.50	8.50	9.50	16.00
Leaf	34.62	25.32	41.62	40.95
Two patterns	1.42	3.30	4.78	4.72
p-value		5.05%	2.62%	0.76%

em [Keogh and Kasetty 2003]. Mostramos ainda que a estratégia de filtragem de ruído e uso da correlação melhora o desempenho em alguns problemas, mas não é possível dizer que ela é, em geral, superior. Para esta alternativa, outras formas de tratamento das séries deverão ser consideradas em trabalhos futuros. Mostramos que o desempenho das métricas é competitivo com a abordagem de extração e seleção de características estáticas. Apresentamos alguns exemplos de métricas com desempenho inferior ao da distância Euclidiana. Exceto em casos especiais, tais alternativas não devem ser utilizadas. Listamos como alguns trabalhos futuros: repetir a avaliação com novas métricas, como as derivadas de modelos de *Markov*, e construir classificadores que aproveitem simultaneamente as qualidades das características extraídas e das medidas de similaridade entre exemplos.

Referências

Agrawal, R., Faloutsos, C., and Swami, A. (1993). Efficient Similarity Search in Sequence Databases. *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, pages 69–84.

Agrawal, R., lp Lin, K., Sawhney, H. S., and Shim, K. (1995). Fast Similarity Search in the Presence of Noise, Scaling, and Translation in Time-Series Databases. *Proceedings of the 21th International Conference on Very Large Data Bases*, pages 490–501.

Antunes, C. M. and Oliveira, A. L. (2001). Temporal Data Mining: An Overview. In *Proceedings of the Workshop on Temporal Data Mining*, San Francisco, EUA. Knowledge Discovery and Data Mining (KDD 01).

Berndt, D. J. and Clifford, J. (1994). Using Dynamic Time Warping to Find Patterns in Time Series. In *KDD Workshop*, pages 359–370.

Bozkaya, T., Yazdani, N., and Özsoyoglu, M. (1997). Matching and Indexing Sequences of Different Lengths. In *CIKM '97: Proceedings of the sixth international conference on Information and knowledge management*, pages 128–135, New York, NY, USA. ACM Press.

Demsar, J. (2006). Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7(1):1–30.

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., and Rossi, F. (2006). *GNU Scientific Library: Reference Manual*. Network Theory.

Geurts, P. (2002). *Contributions to decision tree induction: bias/variance tradeoff and time series classification*. PhD thesis, Department of Electrical Engineering and Computer Science, University of Liege, Belgium.

Hettich, S. and Bay, S. D. (1999). The UCI KDD Archive. http://kdd.ics.uci.edu.

Keogh, E. and Kasetty, S. (2003). On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration. *Data Mining and Knowledge Discovery*, 7(4):349–371.

Keogh, E., Xi, X., Wei, L., and Ratanamahatana, C. A. (2006). The UCR Time Series Classification/Clustering. http://www.cs.ucr.edu/˜eamonn/time_series_data.

Mörchen, F. (2003). Time series feature extraction for data mining using DWT and DFT. Technical report, Departement of Mathematics and Computer Science Philipps-University Marburg.

Nanopoulos, A., Alcock, R., and Manolopoulos, Y. (2001). Feature-based Classification of Time-series Data. pages 49–61.

Olszewski, R. T. (2001). *Generalized Feature Extraction for Structural Pattern Recognition in Time-series Data*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA. Co-Chair-Roy Maxion and Co-Chair-Dan Siewiorek.

Perng, C.-S., Wang, H., Zhang, S. R., and Parker, D. S. (2000). Landmarks: a new model for similarity-based pattern querying in time series databases. In *Proceedings 16th International Conference on Data Engineering*, pages 33–42, San Diego, CA.

Povinelli, R. J., Johnson, M. T., Lindgren, A. C., and Ye, J. (2004). Time series classification using Gaussian mixture models of reconstructed phase spaces. *IEEE Transactions on Knowledge and Data Engineering*, 16(6):779–783.

Pyle, D. (1999). *Data preparation for data mining*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Savary, L. (2002). Notion of Similarity in (Spatio-)Temporal Data Mining. In *ECAI'02 Workshop on Knowledge Discovery from (Spatio-)Temporal Data*, pages 63–71.

Theodoridis, S. and Koutroumbas, K. (2006). *Pattern Recognition*. Elsevier/Academic Press, Amsterdam.

Xi, X., Keogh, E., Shelton, C., Wei, L., and Ratanamahatana, C. A. (2006). Fast time series classification using numerosity reduction. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 1033–1040, New York, NY, USA. ACM Press.

Yamada, Y., Suzuki, E., Yokoi, H., and Takabayashi, K. (2003). Decision-tree Induction from Time-series Data Based on a Standard-example Split Test. In *Proceedings of the 12th International Conference on Machine Learning*, pages 840–847.