

Seleção de Protótipos

Cristiano Pereira {cdsp@cin.ufpe.br}

Roteiro

- Introdução
 - Classificação por aproximação local
 - k-NN
 - Definição de Protótipo
 - NPC
 - Taxonomia
 - Seleção x Síntese
 - Algumas técnicas puramente seletivas
 - CNN
 - Tomek Links
 - OSS
 - Algumas técnicas de síntese
 - LVQ
 - SNPC
 - SGP
 - Combinando modelos de síntese
-

Introdução

- Existem muitos modelos de classificadores de padrões baseados em classificação local
 - As fronteiras de classificação são aproximadas localmente
 - Características:
 - Baseados em similaridade e, em geral, são bastante intuitivos
 - Boas taxas de classificação
 - Aplicável nos mais diversos domínios
 - Biometria
 - Reconhecimento de caracteres
 - Predição de doenças
 - Classificação de imagens
 - ...
-

Introdução

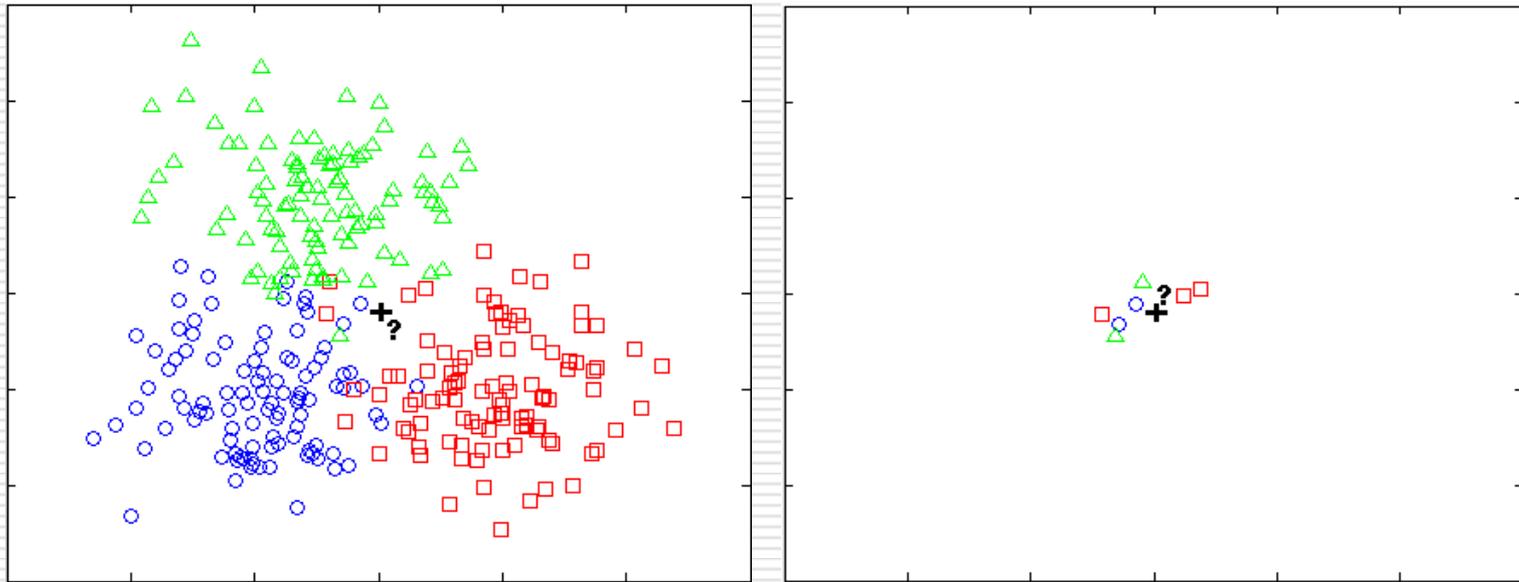
- Nearest Neighbor
 - K-NN é um dos mais clássicos modelos
 - Fix e Rodges (1951)
 - Estratégia de classificação
 - *“Diz-me com quem andas e dir-te-ei quem és”*.
 - Para prever a classe de um novo elemento seleciona-se os K vizinhos mais próximos e adota-se para este elemento a classe predominante nestes vizinhos.
 - Diferentes medidas de distância podem ser utilizadas
 - Mais comum: Euclidiana
 - Bons resultados de classificação
-

Introdução

□ K-NN

■ 7-NN

■ A classe adotada para o novo ponto seria “quadrado”



Introdução

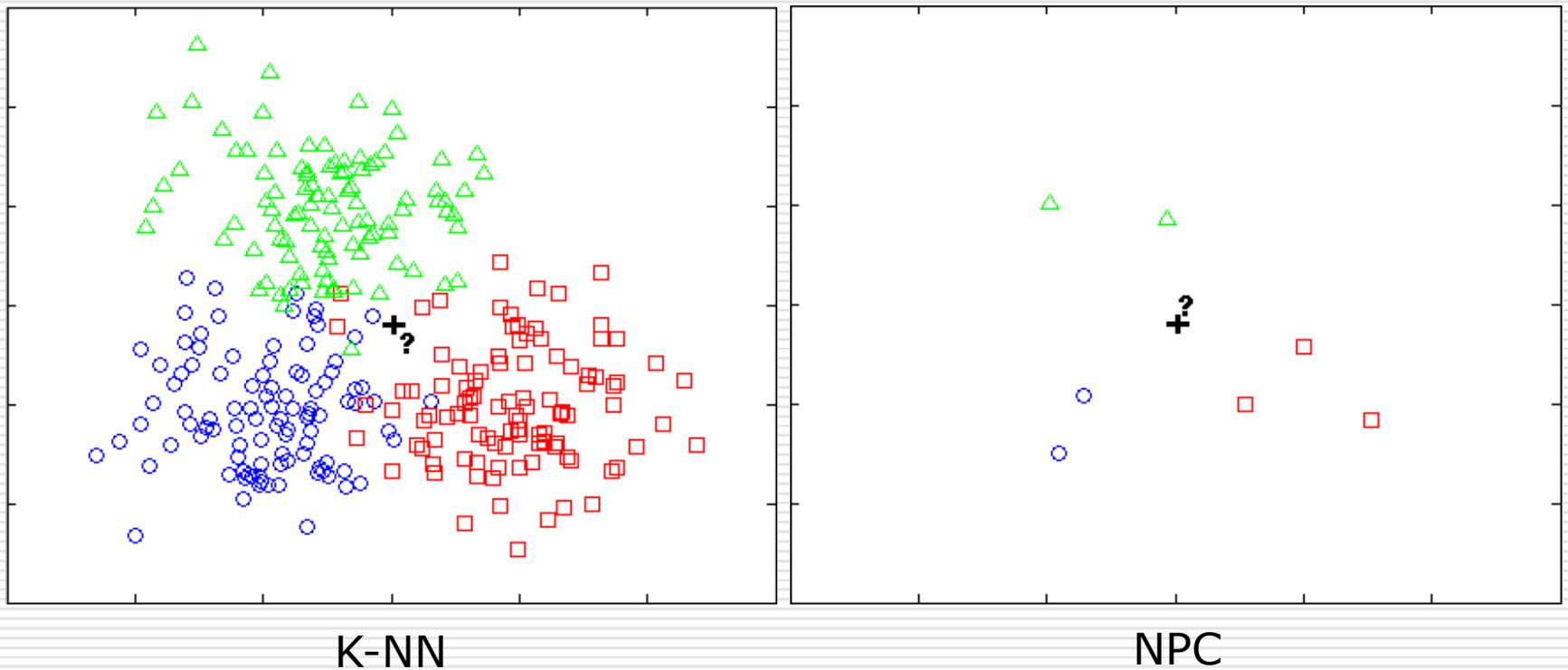
- ❑ O K-NN é dispendioso visto que para achar os k-vizinhos mais próximos de um determinado elemento, é necessário utilizar todos os elementos do conjunto de treinamento.
 - ❑ O tamanho do conjunto de dados pode tornar a aplicação desta técnica inviável em termos de performance.
 - ❑ Idéia: Que tal eleger representantes para as classes ao invés de utilizar todos os elementos?
 - ❑ Estes “escolhidos” são os protótipos.
 - ❑ A escolha deve ser criteriosa para que estes protótipos tenham uma relação com os grupos de elementos (padrões) representados por eles.
-

Introdução

- NPC (Nearest Prototype Classification)
 - Método de classificação baseado em protótipos
 - Cada protótipo representa um conjunto de padrões com características relacionadas
 - As classes são representadas por um ou mais protótipos
 - Uma das vantagens das técnicas baseadas em protótipos é a redução da demanda de espaço para armazenamento e de recursos computacionais para classificação
 - É mais eficiente que o K-vizinhos mais próximos, pois, o número de elementos a serem comparados para a classificação é menor.
-

Introdução

- ❑ Comparação realizada apenas com os protótipos



Introdução

- Por que reduzir?
 - Diminuir a demanda por espaço de armazenamento
 - Reduzir o esforço computacional do processo de classificação
 - Eliminar ruídos e *outliers*
 - Melhorar a capacidade de generalização
 - Acertar mais
-

Introdução

□ Abordagens

- [Brighton and Mellish 2002]
 - Preservação das competências (*Competence Preservation*)
 - Eliminação de informação redundante
 - Aprimoramento das competências (*Competence Enhancement*)
 - Eliminação de ruídos e *outliers*
 - Modelos Híbridos
 - Eliminação de informação redundante, ruídos e outliers.
-

Taxonomia

- Técnicas de seleção também são conhecidas na literatura como:
 - Prototype Reduction Schemes (PRS)
 - Editing Algorithms
 - [Kim e Oomen 2002]
 - As técnicas de seleção de protótipos podem ser enquadradas em dois grupos
 - Seletivas
 - De Síntese
-

Taxonomia

- Técnicas (puramente) Seletivas
 - O conjunto resultante do processo é totalmente composto por elementos selecionados do conjunto original.
 - Não se criam novos elementos
 - Os protótipos ou instâncias são alguns elementos já existentes.
 - Algumas técnicas clássicas:
 - *Edited Nearest Neighbor Rule* (ENN) [Wilson 1976]
 - *Condensed Nearest Neighbor Rule* (CNN) [Hart 1968]
 - *Tomek links* [Tomek 1976]
 - *One-Side Selection* (OSS)[Kubat 1997]
 - *Neighborhood Cleaning Rule* [Laurikkala 2001]
 - Tais técnicas são comumente aplicadas aos problemas com classes desbalanceadas.
-
- Em linhas gerais, quando há muito mais exemplos de uma classe do que elementos das demais.

Taxonomia

- Técnicas de síntese (“*Creative*”)
 - Novos elementos são criados durante o processo de redução
 - Combinação de outros elementos
 - Ajuste realizado por treinamento supervisionado
 - O treinamento é realizado para que os protótipos representem todos os elementos das melhor forma possível
 - Algumas técnicas de síntese
 - *Learning Vector Quantization* (LVQ) [Kohonem 1986]
 - *Soft Nearest Prototype Classification* (SNPC) [Seo et al. 2003]
 - *Self-Generating Prototypes* (SGP) [Fayed et al 2007]
-

Técnicas Seletivas

- ❑ *Edited Nearest Neighbor (ENN)* [Wilson 1976]
 - ❑ *Condensed Nearest Neighbor Rule (CNN)* [Hart 1968]
 - ❑ *Tomek links* [Tomek 1976]
 - ❑ *One-Side Selection (OSS)*[Kubat 1997]
-

ENN - Edited Nearest Neighbor

- [Wilson 1976]
- Projetado para ser um filtro de ruídos
 - Competence Enhancement
- Elimina ruídos e pontos da região de fronteira
 - Região de fronteira:
 - Alta susceptibilidade a erros. Região de indecisão.
- Baixa capacidade de redução
- Muito utilizado como método de pré-processamento do conjunto de dados.

ENN - Edited Nearest Neighbor

- Algoritmo
 - Para cada um dos pontos do conjunto de dados
 - Marque aqueles elementos classificados erroneamente pelos seus k -vizinhos mais próximos
 - Elimine os elementos marcados.
 - Tipicamente, $k = 3$
-

CNN – Condensed Nearest Neighbor

- [Hart 1968]
 - Eliminação de informação redundante
 - Competence preservation
 - Idéia
 - Encontrar o menor subconjunto S do conjunto original de dados E que, utilizando o primeiro vizinho mais próximo (1-NN), classifica corretamente todos os padrões de E .
 - Estratégia
 - Eliminar os elementos que estão mais afastados da fronteira de classificação.
-

CNN – Condensed Nearest Neighbor

- Algoritmo
 1. Inicialmente adote S como o conjunto contendo um elemento de cada classe escolhido aleatoriamente.
 2. A seguir, utiliza-se o primeiro vizinho mais próximo (1-NN) dos elementos de S para classificar todos os elementos de E .
 1. Aqueles que forem classificados corretamente são descartados.
 2. Os elementos de E que forem classificados incorretamente são incorporados ao conjunto S .
- Este critério de seleção não garante a obtenção do menor conjunto que classifica corretamente os elementos de E . Por que?
 - A ordem afeta a seleção.
- Promove drástica redução nas regiões internas dos agrupamentos
- Mantém pontos nas fronteiras e ruídos

CNN – Condensed Nearest Neighbor

- Pode ser utilizado como técnica de undersampling em problemas desbalanceados
 - Problemas desbalanceados: quando há muito mais elementos de uma classe do que outras
 - Adaptação:
 - S inicia-se com todos os elementos da classe minoritária e apenas um elemento da classe majoritária.
-

Tomek Links

- [Tomek 1976]
 - Idéia
 - Eliminar os elementos das regiões de fronteiras e ruídos (ou fortes candidatos a ruídos).
 - Tomek Link
 - Sejam E_i e E_j elementos de classes distintas e $d(E_i, E_j)$ a distância entre eles.
 - Este par de elementos E_i e E_j constitui um *Tomek link* se não há nenhum elemento E_l tal que $d(E_i, E_l) < d(E_i, E_j)$ ou $d(E_j, E_l) < d(E_i, E_j)$.
 - Estratégia
 - Identificar e eliminar os Tomek links
-

Tomek Links

- Como solução ao problema de desbalanceamento
 - Dos Tomek links apenas os elementos da classe majoritária são removidos.
 - Como limpeza dos dados
 - Os dois elementos do par são removidos.
 - Enquanto que o CNN promove uma eliminação drástica dos elementos que estão mais afastados das fronteiras de classificação, o método Tomek links corre em direção contrária:
 - os elementos próximos das regiões de dúvida são eliminados, causando o aparente afastamento das classes.
-

OSS – One-Sided Selection

- [Kubat 1997]
 - OSS é um método de redução surgido da aplicação do algoritmo CNN seguido da aplicação da estratégia Tomek Links.
 - Tomek links tem a função de remover ruídos e elementos nas fronteiras de classificação.
 - Elementos nas fronteiras são indesejados, visto que mesmo um pequeno ruído pode fazê-lo cair no lado errado da fronteira de classificação.
 - CNN tenta eliminar os padrões que estão distantes da fronteira de classificação.
 - Os padrões restantes da classe majoritária e todos os padrões da classe minoritária são utilizados no aprendizado.
-

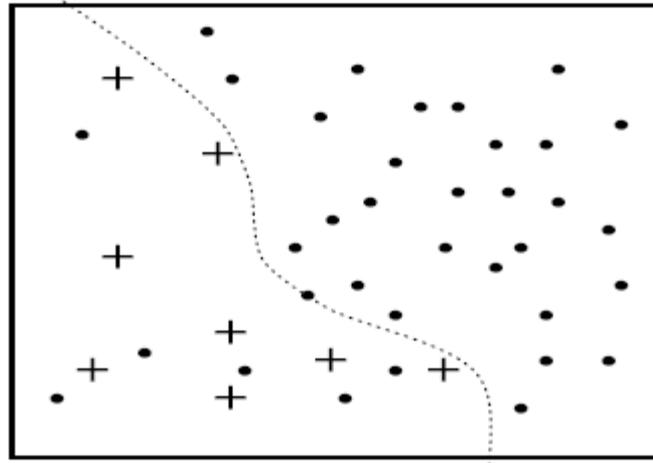
OSS – One-Sided Selection

□ Algoritmo

1. Defina S o conjunto original de treinamento
 2. Inicialmente, defina C como o conjunto que contém todos os elementos da classe minoritária de S e um elemento da classe majoritária escolhido aleatoriamente.
 3. Classifique todos os elementos de S utilizando 1-NN dos elementos de C .
 4. Mova todos os elementos classificados erroneamente para o conjunto C . C agora é o menor subconjunto de S consistente com este.
 5. Remova de C todos os elementos da classe majoritária participantes do Tomek Links.
 6. O conjunto T resultante desta eliminação em 5 é o novo conjunto de treinamento.
-

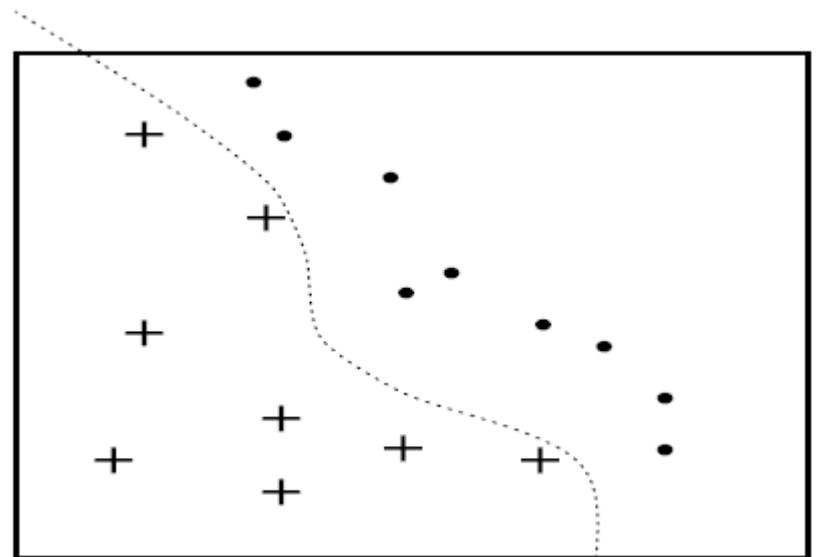
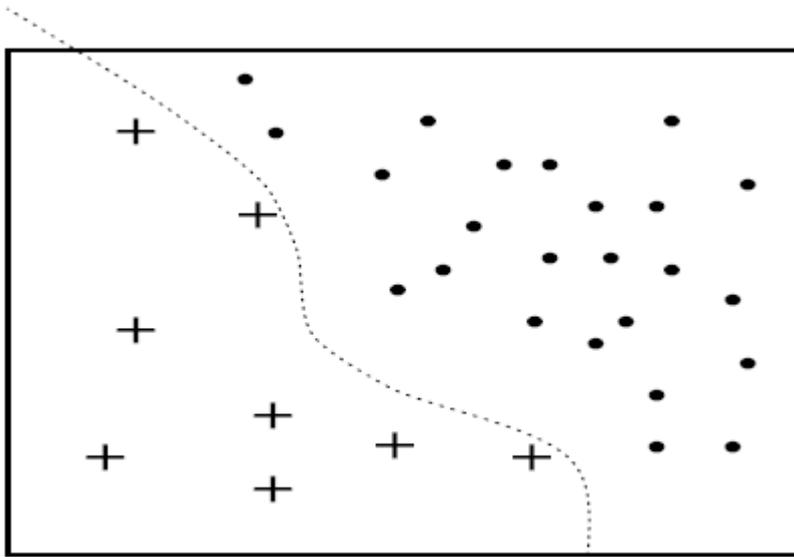
OSS – One-Sided Selection

- Aplicações:
 - Problema com positivos esparsos



OSS – One-Sided Selection

□ Exemplo



Técnicas de Síntese de Protótipos

- *Learning Vector Quantization (LVQ)*
[Kohonen 1986]

LVQ – Learning Vector Quantization

□ Learning Vector Quantization

- Utilizado como algoritmo de aprendizagem para NPC
- Classificação “*winner-takes-all*”

$$c = \arg \min_i \{ \|x - m_i\| \}$$

- A classe atribuída a um novo elemento é a mesma do protótipo mais próximo a este.
 - Redes SOM (Self-Organized Maps)
-

LVQ

- ❑ A precisão do classificador depende do conjunto de protótipos.
 - ❑ Dado um conjunto inicial de protótipos, o LVQ promove um ajuste de forma a estabelecer uma função discriminante baseada nestes protótipos.
 - Por exemplo, o conjunto inicial de protótipos pode ser formado por pontos localizados em torno dos centróides de suas respectivas classes.
 - ❑ O objetivo é minimizar o erro de classificação.
 - ❑ Serão mostrados o LVQ convencional e duas melhorias propostas por Kohonen.
-

LVQ 1

□ Os ajustes seguem a regra:

- Se o elemento x for da mesma classe do protótipo mais próximo:

$$m_c(t+1) = m_c(t) + \alpha(t)[x(t) - m_c(t)]$$

- Casos eles sejam de classes distintas:

$$m_c(t+1) = m_c(t) - \alpha(t)[x(t) - m_c(t)]$$

- Os outros protótipos não são ajustados:

$$m_i(t+1) = m_i(t) \quad i \neq c$$

LVQ 1

□ Características

- Robusto
 - Estabiliza durante o treinamento
 - Treinamento requer um grande número de passos
 - Para a maioria dos problemas a sua aplicação pode ser satisfatória.
-

OLVQ

- ❑ *Optimized-learning-rate* LVQ
 - ❑ Proposto para aumentar a velocidade de convergência do LVQ 1
 - ❑ Modelo:
 - Cada protótipo tem taxas de aprendizado individuais
 - Dinâmica da taxa de aprendizado:
 - ❑ Diminuição da taxa de aprendizado se o protótipo está classificando corretamente e aumentando na situação inversa.
-

OLVQ

- Modelo:

$$\alpha_c(t) = \frac{\alpha_c(t-1)}{1 + s(t)\alpha_c(t-1)} \quad 0 < \alpha(t) < 1$$

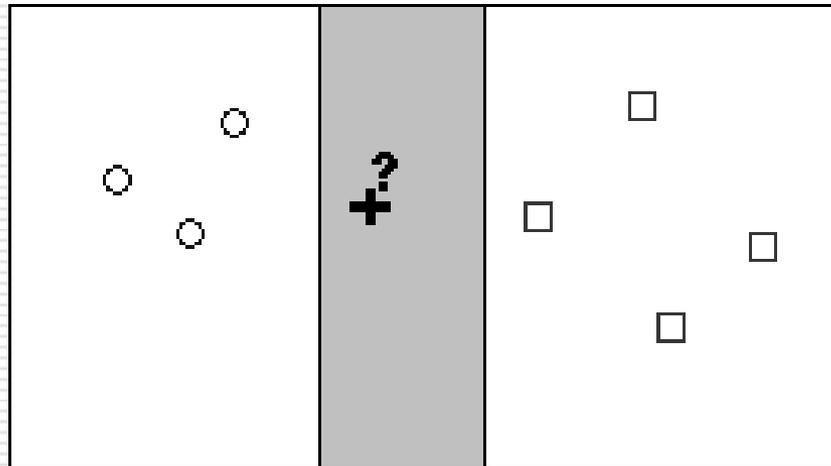
- $s(t) = +1$, se x é classificado corretamente,
 - $s(t) = -1$, caso contrário.
-

LVQ - A Regra da Janela

- As demais variações do LVQ propostas por Kohonen fazem atualização nos dois protótipos mais próximos desde que atendidas certas condições do modelo.
 - Para evitar que os protótipos divirjam foi introduzida a Regra da Janela
 - A idéia é ajustar os protótipos apenas para os elementos do treinamento que caírem próximos às fronteiras de classificação.
 - Região de indecisão.
-

Regra da Janela

- A região sombreada fica em torno das fronteiras de classificação. É naturalmente uma região de dúvida.



Regra da Janela

- Dado um elemento do conjunto de treinamento x , ele terá “caído na janela”, se:

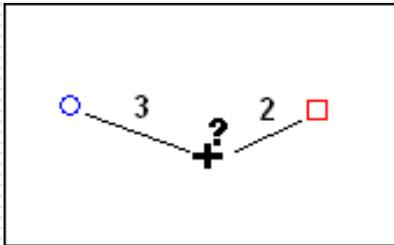
- $\min\left(\frac{d_i}{d_j}, \frac{d_j}{d_i}\right) > s$ onde $s = \frac{1-w}{1+w}$

- m_i m_j são os protótipos mais próximos de x
 - d_i e d_j são as distâncias de x aos protótipos m_i m_j respectivamente.
 - w é a largura relativa e o único parâmetro da regra.
-

LVQ

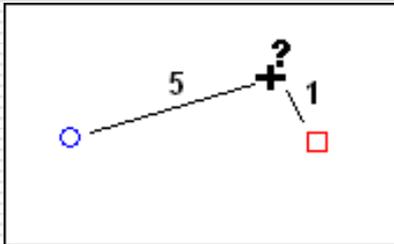
□ Regra da Janela

$$\text{Para } w = 0.60 \Rightarrow s = \frac{1 - 0.60}{1 + 0.60} = 0.25$$



$$\min\left(\frac{3}{2}, \frac{2}{3}\right) = 0.67$$

Protótipos são ajustados



$$\min\left(\frac{5}{1}, \frac{1}{5}\right) = 0.20$$

Não há ajuste

LVQ 2.1

- ❑ O LVQ 1 provoca o afastamento dos protótipos das fronteiras de decisão segundo o modelo de Bayes.
 - ❑ O LVQ 2.1 foi projeto pensando em reduzir este afastamento atuando sobre os protótipos vizinhos pertencentes a classes adjacentes.
 - ❑ O LVQ 2.1 não é utilizado sobre o conjunto inicial de protótipos, mas sim sobre um conjunto previamente ajustado pelo LVQ 1
 - ❑ Refinar os ajustes em busca de maior precisão nos resultados de classificação
 - ❑ A regra de classificação é a mesma do LVQ1
-

LVQ 2.1

□ Regra

- m_i e m_j são os protótipos mais próximos de x , sendo m_i da mesma classe e m_j de uma classe diferente.

- Ajustes:

$$m_i(t+1) = m_i(t) + \alpha(t)[x(t) - m_i(t)]$$

$$m_j(t+1) = m_j(t) - \alpha(t)[x(t) - m_j(t)]$$

- Aproxima o protótipo de mesma classe e repele o protótipo que "tenta" classificar o padrão erroneamente.
-

LVQ 2.1

- Não haverá ajustes quando:
 - Os dois protótipos forem da mesma classe, não importando se forem da classe x ou de qualquer outra classe.
 - Nenhum dos dois protótipos for da classe do elemento x .
 - O x não cair na janela.
-

LVQ3

- ❑ A aplicação do LVQ 2.1 pode sobre-ajustar os protótipos às fronteiras de classificação.
 - ❑ Com o objetivo de buscar uma maior proximidade com a distribuição real das classes foi introduzido o LVQ 3.
 - ❑ Este método também atua quando o elemento já está sendo classificado corretamente pelos dois protótipos mais próximos.
 - ❑ Foi introduzido um fator de estabilização ϵ .
-

LVQ 3

□ Regras

- Se m_i e m_j são os protótipos mais próximos de x , sendo m_i da mesma classe e m_j de uma classe diferente, os ajustes serão:

$$m_i(t+1) = m_i(t) + \alpha(t)[x(t) - m_i(t)]$$

$$m_j(t+1) = m_j(t) - \alpha(t)[x(t) - m_j(t)]$$

- Se m_i e m_j forem da mesma classe de x , então:

$$m_k(t+1) = m_k(t) + \varepsilon\alpha(t)[x(t) - m_k(t)]$$

- $k \in \{i, j\} \quad 0 < \varepsilon < 1$
-

Referências

- ❑ Fayed, H. A., Hashem, S. R. and Atiya, A. F. (2007). "Self-generating prototypes for pattern classification". Pattern Recognition, vol. 40, pp 1498-1509.
 - ❑ Hart, P. E. (1968) "The Condensed Nearest Neighbor Rule". IEEE Transactions on Information Theory IT-14, pp. 515-516.
 - ❑ Kim, S-W. and Oommen, B. J. (2003). "A Brief Taxonomy and Ranking of Creative prototype Reduction Schemes, Pattern Analysis and Applications Journal, pp. 232-244.
 - ❑ Kohonen, T. (1986) "The Learning Vector Quantization". Helsinki Univ. Technol., Tech. Rep.
 - ❑ Kubat, M., and Matwin, S. (1997) "Addressing the Course of Imbalanced training Sets: One-sided Selection". In ICML, pp. 179-186.
 - ❑ Laurikkala, J. (2001) "Improving Identification of Difficult Small Classes by Balancing Class Distribution". Tech. Rep. A-2001-2, university of Tampere.
 - ❑ Seo, S., Bode, M. and Obermayer, K. (2003) "Soft Nearest Prototype Classification", IEEE Transactions On Neural Networks, vol. 14, no. 2, pp. 390-398.
 - ❑ Tomek, I. (1976) Two Modifications of CNN. IEEE Transactions on Systems Man and Communications SMC-6, pp. 769-772.
-