

SUCESU 2005 – Tecnologias – Inteligência Artificial
O estado da arte em métodos para reconhecimento de padrões: Support Vector Machine

Bernardo Penna Resende de Carvalho¹
bpenna@gmail.com

Resumo

A área de reconhecimento de padrões consiste na classificação de diversos exemplos, existentes em uma determinada base de dados, como pertencentes a um tipo específico de padrão, dentre os possíveis padrões que essa base possa representar. Vários métodos são empregados nesta área, porém nos últimos anos um método vem se destacando entre os demais: Support Vector Machine (SVM).

As SVMs são máquinas de aprendizagem que se baseiam na Teoria da Aprendizagem Estatística, treinadas através de um algoritmo supervisionado. Elas foram propostas em 1992. Desde então são empregadas em diversos setores, obtendo resultados superiores a outros métodos. As SVMs possuem uma formulação teórica consistente, que aliada aos resultados práticos obtidos, as tornam o estado da arte em métodos para reconhecimento de padrões.

Neste trabalho, são descritos os princípios fundamentais que caracterizam as SVMs, bem como sua interpretação geométrica e exemplos de problemas em que são empregadas. Elas possuem algumas características particulares, como: a detecção automática dos exemplos mais relevantes nas bases de dados utilizadas, chamados vetores de suporte; a robustez aos exemplos das bases que são notadamente errôneos, conhecidos como outliers; e o mapeamento implícito dos exemplos em um espaço de dimensões elevadas, através das funções de kernel.

1) Introdução

Métodos de reconhecimento de padrões são pesquisados desde a década de 60 (Kanal 1968), época em que se iniciava o desenvolvimento da informática. Nos últimos 40 anos, não foram poucos os métodos criados para esta área, como Redes Neurais Artificiais (Rosenblatt 1958), Árvores de Decisão (Quinlan 1986), Algoritmos Genéticos (Goldberg 1989), entre outros.

Em 1992, um grupo de pesquisa da AT&T Bell Laboratories desenvolveu um método de classificação inovador, inicialmente conhecido como “Algoritmo para classificadores de margens ótimas” (Boser et al. 1992). No ano seguinte, foi publicado Boser & Guyon (1993), expandindo alguns conceitos contidos no trabalho inicial. Em Cortes & Vapnik (1995), os autores propuseram uma forma de se lidar de maneira eficiente com os outliers, como são conhecidos os exemplos que representam padrões notadamente incorretos, que interferem de maneira significativa nos métodos até então usados. A partir da publicação deste último artigo, o método passou a ser conhecido como Support Vector Machine, como é conhecido até hoje.

O objetivo deste trabalho é explicar o funcionamento das SVMs, descrevendo seus princípios fundamentais e as ferramentas das quais elas se baseiam, como treinamento supervisionado, aprendizagem estatística e otimização global. Além disso, serão indicadas várias áreas em que sua aplicação se mostrou bem sucedida e outras em que ainda podem vir a ser utilizadas, com grande expectativa de sucesso.

O trabalho está dividido em 6 seções. Na seção 2, é descrita e exemplificada a área de reconhecimento de padrões. Na seção 3, são abordados os princípios básicos que dão suporte à formulação das SVMs, que é descrita na seção 4. Na quinta seção, são discutidas algumas aplicações das SVMs. Por fim, a seção 6 contém algumas conclusões do trabalho. Para a correta compreensão, uma breve explicação da notação utilizada neste trabalho: os vetores, sempre do tipo coluna, são apresentados em negrito, e os valores numéricos com fonte normal.

2) Reconhecimento de padrões

Seja uma base de dados qualquer, constituída de diversos exemplos, i.e. seus elementos. Cada exemplo possui um tipo específico de padrão associado a ele. Um padrão nada mais é do que o tipo do exemplo, ou seja, um rótulo que o caracterize ou que o classifique. Os exemplos das bases de dados são geralmente medições ou observações sobre determinado assunto, definindo o domínio do processo de aprendizagem.

¹ Engenheiro Eletricista - UFMG, Mestrando em Engenharia Elétrica (Inteligência Computacional) – UFMG

Um método de reconhecimento de padrões deve, baseado no conhecimento extraído dos exemplos de uma base, classificar um exemplo novo, desconhecido até então, ao padrão que mais reflete as suas características. Problemas de reconhecimento de dígitos, reconhecimento de faces, predição de tendências em séries financeiras, predição de falhas em equipamentos, e muitos outros, englobam o universo do reconhecimento de padrões. Esta área é muito extensa e surgem freqüentemente novas aplicações, fazendo com que métodos poderosos sejam cada vez mais necessários.

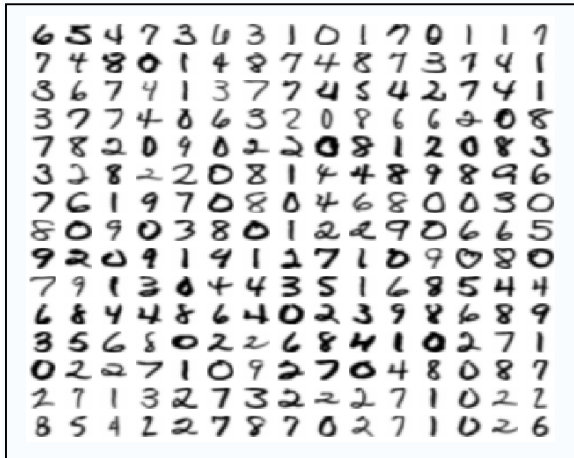


Figura 1 - Reconhecimento de dígitos

Na Figura 1, são representados exemplos para a tarefa de reconhecimento de dígitos usados em Le Cun et al. (1989). Os padrões desta base são os dígitos 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9. Dizer que um exemplo se refere ao dígito 3, equivale a dizer que a distribuição dos pixels deste exemplo representa, de modo geral, o padrão de distribuição encontrado no dígito 3.

Na Figura 2, pode-se observar alguns exemplos utilizados em Guodong et al. (2000) para o reconhecimento de faces. Neste caso, cada face constitui um exemplo e cada indivíduo um padrão da base de dados.

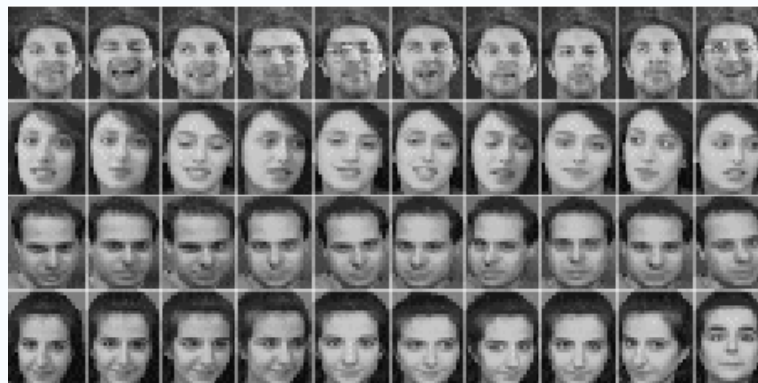


Figura 2 - Reconhecimento de faces

3) Aprendizagem de máquinas

Uma máquina de aprendizagem deve ter a propriedade de, após a observação de vários pares de entrada e saída $\{x_i, y_i\}_{i=1}^N$, imitar o comportamento do sistema, gerando saídas próximas de y_i a partir de entradas próximas de x_i (Vapnik 1995).

Quando o número de padrões - saídas ou classes - é finito, normalmente números naturais, a tarefa é denominada classificação de padrões. Se houver apenas duas classes possíveis, dá-se o nome de classificação binária. Já quando existe um número infinito de padrões possíveis (valores reais), eles são conhecidos como problemas de regressão.

3.1) Treinamento supervisionado

O processo de treinamento - ou aprendizagem - de uma máquina se refere à fase em que ela adquire o conhecimento, ou seja, retém as informações relevantes a respeito de um assunto específico, representado por uma base de dados, para fazer uso destas informações no futuro. O objetivo do treinamento é ajustar os parâmetros livres da máquina de forma a encontrar uma ligação entre os pares entrada e saída (Braga et al. 2000).

Um tipo particular de treinamento, o treinamento supervisionado, é realizado por meio de um supervisor externo. O supervisor é responsável por fornecer para a máquina as entradas - exemplos de treinamento - juntamente com as saídas desejadas para cada exemplo. Desta

forma, ações podem ser tomadas a fim de valorizar os acertos e punir os erros obtidos pela máquina, possibilitando que o processo de aprendizagem se efetue.

A utilização de um método de reconhecimento de padrões pode ser dividida em duas fases: treinamento e aplicação, cada uma utilizando um conjunto de dados específico. Na primeira é usado o conjunto de treinamento, composto pelos exemplos nos quais a máquina obtém o conhecimento. Na segunda fase é utilizado o conjunto de teste, constituído pelos exemplos no qual o método será efetivamente aplicado. O conjunto de treinamento deve ser estatisticamente representativo, para que seja possível à máquina reconhecer os exemplos de teste, propriedade conhecida como generalização.

3.2) Aprendizagem estatística

Seja $\{\mathbf{x}_i, y_i\}_{i=1}^N$ um conjunto de treinamento com N exemplos, uniformemente distribuídos em relação a uma função de densidade de probabilidade desconhecida $p(\mathbf{x})$. O objetivo do processo de aprendizagem estatística é obter uma função indicadora que minimize o risco funcional, por meio das relações extraídas deste conjunto (Vapnik 1995).

O risco funcional é a probabilidade da saída desejada ser diferente da saída obtida pela máquina, após a escolha de uma função indicadora. Como $p(\mathbf{x})$ é desconhecida, não se pode calcular diretamente este risco, utilizando-se um princípio indutivo para sua aproximação.

Os métodos de aprendizagem estatística devem apresentar as seguintes características:

- um conjunto flexível e grande o suficiente de funções indicadoras disponíveis, para representar o comportamento do conjunto de dados. As funções indicadoras são responsáveis por tentar simular o comportamento dos sistemas em que os métodos são utilizados.

- um princípio indutivo, capaz de associar o conjunto de treinamento à função que governa o sistema. São exemplos de princípios indutivos: regularização, minimização do risco empírico, minimização do risco estrutural, inferência Bayesiana.

- um algoritmo de aprendizagem, procedimento que indica como implementar o princípio indutivo e selecionar a melhor função dentro do universo das funções indicadoras existentes.

No processo de escolha da melhor função que se ajusta ao conjunto de treinamento, é necessária a criação de uma medida de discrepância ou perda, que sinaliza à máquina quando houve erros ou acertos durante a aprendizagem (Vapnik 1998).

Para problemas de classificação binária, a função de perda comumente empregada é

$$P(y, f(\mathbf{x}, \mathbf{z})) = \begin{cases} 1 & \text{se } f(\mathbf{x}, \mathbf{z}) \neq y \\ 0 & \text{se } f(\mathbf{x}, \mathbf{z}) = y \end{cases} \quad (1)$$

onde \mathbf{z} são parâmetros da função indicadora e $f(\mathbf{x}, \mathbf{z})$ a saída da máquina cuja entrada é \mathbf{x} .

Um princípio indutivo geralmente empregado pelas máquinas de aprendizagem existentes é a minimização do risco empírico. O risco empírico, calculado utilizando-se (1), é dado por

$$R_{empírico} = \frac{1}{N} \sum_{i=1}^N P(y_i, f(\mathbf{x}_i, \mathbf{z})). \quad (2)$$

Sua minimização nem sempre é suficiente para a obtenção de resultados adequados, pois ela não leva em consideração a complexidade das funções indicadoras. Quando a complexidade das funções é superior à necessidade do problema, ocorre o sobre-ajuste (overfitting – Fig. 3a) da função em relação ao conjunto de treinamento. Quando ela é inferior, ocorre o sub-ajuste (underfitting – Fig. 3b). Em ambos os casos, a capacidade de generalização é reduzida. A Fig. 3c apresenta uma função cuja complexidade é adequada ao problema.

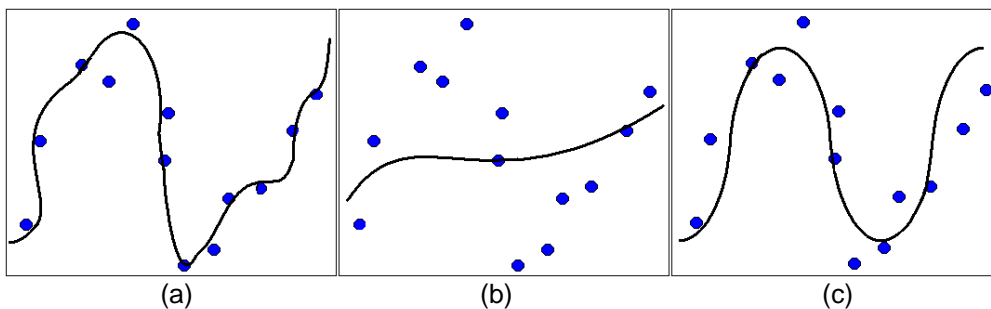


Figura 3 - Funções indicadoras ou de aproximação

Com o uso do conceito de dimensão VC (Vapnik 1995), foi desenvolvida uma expressão, com probabilidade $(1 - \eta)$ de ocorrer, que indica que o limite superior do risco funcional é

$$R_{funcional} \leq R_{empirico} + R_{bound}(h, \eta, N) \quad (3)$$

dado um valor de $\eta \in [0,1]$. A dimensão VC é h e N o número de exemplos de treinamento.

A minimização do risco estrutural tem como objetivo minimizar $R_{bound}(h, \eta, N)$, o fator somado ao risco empírico em (3). Este princípio usa a dimensão VC para controlar a complexidade das funções indicadoras, de forma a adequá-las a cada problema.

3.3) Otimização global e teoria do Lagrangeano dual

Qualquer problema de otimização pode ser descrito como: Ache os valores dos parâmetros $\mathbf{v} = [v_1, \dots, v_M]^T$ que minimizem a função $c_p(\mathbf{v})$, sujeita às restrições $g(\mathbf{v}) \leq 0$ e $h(\mathbf{v}) = 0$.

Quando a função de custo $c_p(\mathbf{v})$ é uma função convexa, quadrática em \mathbf{v} , e as restrições $g(\mathbf{v})$ e $h(\mathbf{v})$ são lineares em \mathbf{v} , dá-se o nome de QP (quadratic programming). Um problema QP tem a propriedade de possuir uma única solução global, ou seja, existe um único conjunto de valores de \mathbf{v} que torna $c_p(\mathbf{v})$ a mínima possível. Esta característica, como indica a Figura 4, diferencia as SVMs de outros métodos, como as Redes Neurais Artificiais (RNAs).

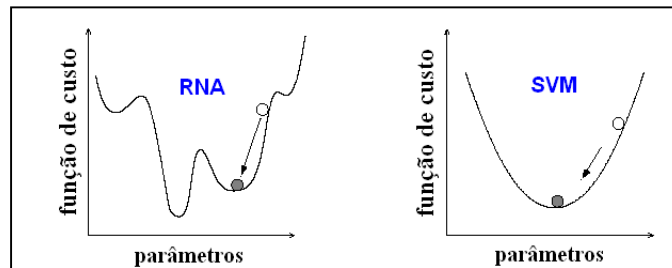


Figura 4 - Otimização local RNAs x Otimização global SVMs

O problema de otimização descrito anteriormente, também chamado de problema primal, pode apresentar dificuldade na obtenção da solução, devido principalmente à natureza das restrições $g(\mathbf{v})$ e $h(\mathbf{v})$. Por este motivo, é comum a utilização da Teoria do Lagrangeano (Fletcher 1987) para que seja obtida uma formulação dual para o problema de otimização, mais simples de se resolver que a primal. O problema dual possui a mesma solução do primal, quando obedecidas certas condições (Luenberger 1984). Uma implicação desta propriedade é que se pode resolver indiretamente o problema primal por meio da resolução direta do dual.

O problema Lagrangeano dual pode ser obtido pelo acréscimo das restrições primais à função de custo primal, com o uso dos multiplicadores de Lagrange α e β . Ele é descrito como: Minimize a função $c_D(\mathbf{v}) = c_p(\mathbf{v}) + \alpha \cdot g(\mathbf{v}) + \beta \cdot h(\mathbf{v})$ em relação aos parâmetros \mathbf{v} e a maximize em relação aos parâmetros α e β . A única restrição do problema dual é $\alpha \geq 0$.

4) Support Vector Machines

Dado o conjunto de treinamento $\{\mathbf{x}_i, y_i\}_{i=1}^N$ com entradas $\mathbf{x}_i \in \mathfrak{R}^n$ e saídas correspondentes $y_i \in \{-1, +1\}$, a SVM foi desenvolvida para a aplicação em tarefas de classificação binária. Para isto, ela cria uma superfície linear de separação $f(\mathbf{x}) = 0$ descrito por

$$\mathbf{w}^T \varphi(\mathbf{x}) + b = 0 \quad (4)$$

onde \mathbf{w} é o vetor de pesos, b o termo de polarização e $\varphi(\cdot)$ o mapeamento realizado em um espaço, chamado espaço de características, cuja dimensão é superior a dos dados de entrada.

Durante o processo de treinamento de uma SVM, utiliza-se o produto da saída desejada y_i pela saída obtida $f(\mathbf{x}_i)$ para indicar se a classificação foi correta, como é mostrado por

$$y_i [\mathbf{w}^T \varphi(\mathbf{x}_i) + b] \geq 1 + \xi_i \text{ com } \xi_i \geq 0. \quad (5)$$

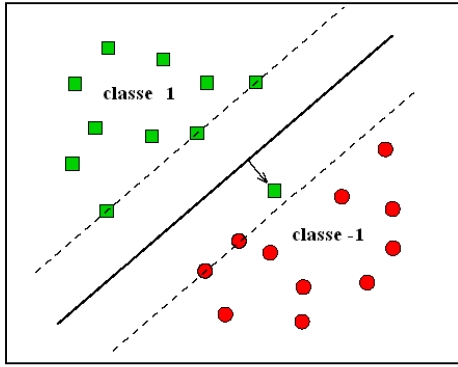


Figura 5 - Variáveis de folga

As variáveis de folga ξ possibilitam a correta classificação dos exemplos da base de dados que se encontram ligeiramente fora da região de sua classe, como mostrado na Figura 5.

Desta forma, as SVMs se tornam robustas a pequenas variações no conjunto de treinamento, diferentemente de outras máquinas de aprendizagem, possibilitando uma melhor generalização (Braga et al. 2000).

O processo de treinamento das SVMs consiste na obtenção de valores para \mathbf{w} e b , de forma a minimizar uma função de custo $J(\mathbf{w}, \xi)$. As SVMs têm como objetivo a construção de um hiperplano ótimo (Vapnik 1995), que maximiza a margem de separação, representada por

$$M = \frac{2}{\|\mathbf{w}\|^2}. \quad (6)$$

O primeiro termo da função de custo $J(\mathbf{w}, \xi)$ minimiza a norma do vetor de pesos $\|\mathbf{w}\|^2$, a fim de maximizar a margem. O outro termo minimiza as variáveis de folga $\xi = [\xi_1, \dots, \xi_N]^T$, para evitar que todos os exemplos incorretos sejam considerados outliers. Para a criação da função de custo, é utilizado um parâmetro de regularização C , que pondera estes dois termos.

Há várias superfícies que separam os exemplos da Figura 6 (a, b, c, d), porém a melhor delas é a de máxima margem, Fig. 6-d. Resultados teóricos obtidos em Vapnik (1995) indicam que a maximização das margens de separação entre os exemplos das classes -1 e $+1$, como mostrado na Figura 7, implica em uma maior generalização de uma máquina de aprendizagem.

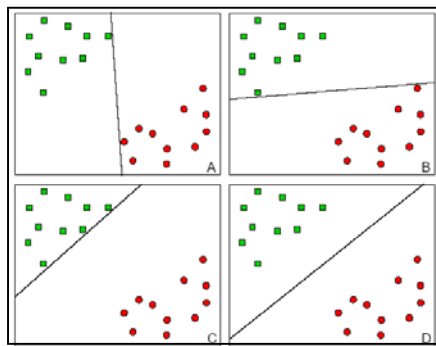


Figura 6 – Superfícies de separação

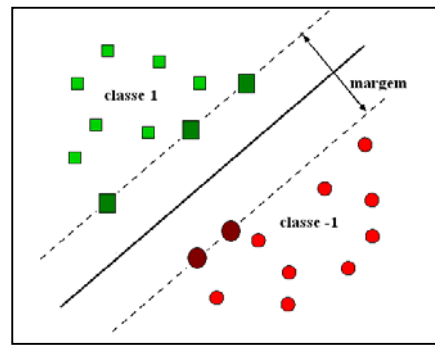


Figura 7 – Vetores de suporte

Pode-se definir o problema primal de uma SVM como

$$\min_{\mathbf{w}, b, \xi} J_P(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \quad \text{sujeito a} \quad \begin{cases} y_i [\mathbf{w}^T \phi(\mathbf{x}_i) + b] \geq 1 - \xi_i \\ \xi_i \geq 0, \quad i = 1, \dots, N \end{cases} \quad (7)$$

em que a primeira restrição é a condição apresentada em (5).

Como descrito na seção 3.3, é aplicado o Lagrangeano (Fletcher 1987) ao problema primal (7), resultando no problema Lagrangeano dual, que é então derivado em relação aos parâmetros primais. As derivadas são igualadas a zero, para minimizar o Lagrangeano em relação aos parâmetros primais. As expressões obtidas são então substituídas no próprio Lagrangeano dual para a obtenção do problema de otimização dual de uma SVM, dado por

$$\max_{\alpha} J_D(\alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad \text{suj. a} \quad \begin{cases} \sum_{i=1}^N \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C, i = 1, \dots, N \end{cases} \quad (8)$$

em que α_i é limitado pelo valor do parâmetro C , que deve ser sintonizado pelo usuário. A função $K(\mathbf{x}_i, \mathbf{x}_j)$ é chamada função de kernel.

As funções de kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)$, como as da Tabela 1, realizam um produto no próprio espaço de entrada, e não no espaço de características, que possui uma dimensão mais elevada. Graças às funções de kernel, problemas não linearmente separáveis podem ser resolvidos pelas SVMs, uma vez que a superfície de separação é linear apenas no espaço de características, e não no espaço de entrada, como mostra a Figura 8.

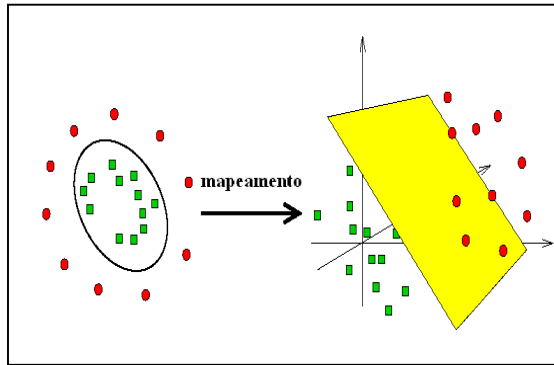


Figura 8 – Espaços: entrada x característica

| Kernel | Expressão | Parâmetro |
|------------|--|--------------------|
| Linear | $\mathbf{x}_i^T \cdot \mathbf{x}_j$ | |
| RBF | $e^{-\ \mathbf{x}_i - \mathbf{x}_j\ ^2 / 2\sigma^2}$ | σ^2 |
| Polinomial | $(\mathbf{x}_i^T \cdot \mathbf{x}_j + a)^b$ | a, b |
| Sigmóide | $\tanh(\beta_0 \mathbf{x}_i^T \cdot \mathbf{x}_j + \beta_1)$ | β_0, β_1 |

Tabela 1 – Algumas funções de kernel

Existem vários algoritmos capazes de resolver problemas QP rapidamente, que podem ser utilizados para a resolução do problema dual das SVMs (Platt 1998, Mangasarian & Musicant 1999). Para a utilização das SVMs, após o processo de aprendizagem, não é necessário realizar o mapeamento $\varphi(\mathbf{x}_i)$ diretamente. Basta usar as funções de kernel, juntamente com os exemplos de treinamento, os multiplicadores de Lagrange e o termo de polarização ótimos, de modo a calcular a saída da SVM para um exemplo \mathbf{x}_k qualquer, utilizando

$$f(\mathbf{x}_k) = \text{sign} \left[\sum_{i=1}^N \alpha_i^* y_k K(\mathbf{x}_k, \mathbf{x}_i) + b^* \right]. \quad (9)$$

Uma característica das soluções da SVM é o fato de que vários valores de α_i^* são nulos após o processo de treinamento. Como indicado em (9), quando estes valores são nulos, os exemplos aos quais eles são associados não influenciam na saída da SVM, ou seja, são irrelevantes para o problema. Os exemplos que possuem multiplicadores de Lagrange não nulos são conhecidos como vetores de suporte, e normalmente estão próximos da superfície de separação, como mostra a Figura 7.

5) Aplicações de SVM

As SVMs foram aplicadas com sucesso em diversas áreas. Na medicina, foram usadas para a identificação de proteínas em Zien et al. (2000) e de células cancerígenas em Cristianini et al. (2000). Na área de segurança, elas foram utilizadas para o reconhecimento de impressões digitais em Pontil et al. (2001), além do seu emprego tanto na detecção (Osuna 1997) quanto no reconhecimento de faces (Guodong et al. 2000).

As tarefas de reconhecimento de textos (Joachims 1998) e de assinaturas (Bortolozzi et al. 2003) por meio das SVMs também obtiveram resultados significativos. Análises de crédito através de SVMs foram abordados em Mangasarian & Musicant (1999).

Uma modificação das SVMs, chamada Support Vector Regression (SVR), capaz de lidar com problemas de predição ou regressão de funções, foi desenvolvida em Vapnik et al. (1996), e utilizada com sucesso em Muller et al. (1997).

Existem ainda áreas em que as SVMs tendem a contribuir de modo significativo nos próximos anos, mas que ainda são utilizados apenas métodos mais conhecidos, como as Redes Neurais Artificiais ou Algoritmos Genéticos. Alguns exemplos destas áreas são as indústrias de mineração (Carvalho & Monteiro 2003) e siderurgia (Carvalho et al. 2004).

6) Conclusão

A área de reconhecimento de padrões é muito extensa e surgem freqüentemente novas aplicações, fazendo com que métodos poderosos sejam cada vez mais necessários. Neste contexto se inserem as SVMs, que possuem uma formulação teórica consistente, e têm obtido resultados práticos de sucesso em diversas áreas.

As SVMs utilizam o princípio de minimização do risco estrutural, que resulta em uma alta capacidade de generalização, mesmo que o conjunto de treinamento não seja muito representativo. Além disso, elas possuem outras características, descritas com detalhes neste trabalho, que justificam a denominação de estado da arte em métodos de reconhecimento de padrões.

7) Bibliografia

- BORTOLOZZI, F. & JUSTINO, E.J.R. & SABOURIN, R. 2003. An Off-Line Signature Verification Method Based on SVM Classifier and Graphometric Features . *In* Fiith International Conference on Advances in Pattern Recognition (1):134-141, Calcutá.
- BOSER, B. & GUYON, M. 1993. Automatic Capacity Tuning of Very Large VC-dimension Classifiers. *Advances in Neural Information Processing Systems*.
- BOSER, B. & GUYON, M. & VAPNIK, V. 1992. A Training Algorithm for Optimal Margin Classifiers. *Computational Learning Theory*. 144–152. Pittsburgh, PA.
- BRAGA, A.P. & CARVALHO, A.P.L.F. & LUDERMIR, T.B. 2000. *Redes Neurais Artificiais: Teoria e aplicações*. LTC. 262p.
- CARVALHO, B.P.R. & MONTEIRO, A.M. 2002. Modelagem Neural de um Processo de Produção de Pelotas de Minério de Ferro. VII Sem. Autom. de Processos ABM, Santos.
- CARVALHO, B.P.R. & MORAIS, F.M. & SENNA & A.L. 2004. Predição do teor de silício no ferro-gusa utilizando técnicas de inteligência artificial. *ABM Internat. Meeting Ironmaking*.
- CORTES, C. & VAPNIK, V. 1995. Support-Vector Networks. *Machine Learning*.
- CRISTIANINI N. & DUFFY N. & SCHUMMER M. 2000. Support vector classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*. 16(10).
- FLETCHER, R. 1987. *Practical Methods of Optimization*. 2 ed. John Wiley and Sons.
- GOLDBERG, D.E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, USA.
- GUODONG, G. & LI, S. & KAPLUK, S. 2000. Face recognition by support vector machines. *In Proc. IEEE International Conf. on Automatic Face and Gesture Recognition*, 196-201.
- JOACHIMS, T. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *In Proceedings of European Conf. Machine Learning*, 137-142.
- KANAL, L.N. 1968. *Pattern Recognition*. Thompson Book. Library of Congress No.68-31794.
- LE CUN, Y. & BOSER, B. & DENKER, J. S. & HENDERSEN, D. & HOWARD, R.E. & HUBBARD, W. & JACKEL, L.D. 1989. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation* (1):541-551.
- LUENBERGER, D.G. 1984. *Linear and Nonlinear Programming*, Addison-Wesley, CA.
- MANGASARIAN, O. L. & MUSICANT, D. R. 1999. Successive overrelaxation for support vector machines. *IEEE Trans. Neural Networks* 10 (5), 1032-1037.
- MULLER K.R. & SMOLA A. & RÄTSCH G. & SCHÖLKOPF B. & VAPNIK, V. 1997. Predicting Time Series with Support Vector Machines. *In Proceedings ICANN'97*, p.999.
- OSUNA, E. 1997. Training Support Vector Machines: an Application to Face Detection.
- PLATT, J. 1999. Fast training of support vector machines using sequential minimal optimization. *In Advances in kernel methods-support vector learning*. MA, 185–208.
- PONTIL, M. & YAO, Y. MARCIALIS, G. & FRASCONI, P. & ROLI F. 2001. A new machine learning approach to fingerprint classification. *In 7th Congress of the Italian Association for Artificial Intelligence*, 57-63.
- QUINLAN, J.R. 1986. Induction of decision trees. *Machine Learning*, 1:81-106.
- ROSENBLATT, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev.* 65(6):386-408.
- VAPNIK, V. 1995. *The nature of statistical learning theory*. Springer-Verlag, New York.
- VAPNIK, V. 1998. *Statistical learning theory*. John Wiley and Sons, New York.
- VAPNIK, V. & GOLOWICH, E. & SMOLA, A. 1996. Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing. *In Advances in Neural Information Processing Systems*. (9):281-287, Cambridge, MA.
- ZIEN, A. & RÄTSCH, G. & MIKA, Z. & SCHÖLKOPF, B. & LENGAUER, T. & MULLER, K.R. 2000. Engineering support vector machine kernels that recognize translation initiation sites. *Bioinformatics*, 16(9):799--807.