O Método de Redes Neurais com Função de Ativação de Base Radial para Classificação em Data Mining

Ana Paula Scotti[1], Merisandra Côrtes de Matos[2], Priscyla Waleska T. A. Simões[2]

[1]Acadêmico do Curso de Ciência da Computação – Unidade Acadêmica de Ciências, Engenharias e Tecnologias – UNESC

[2]Professor do Curso de Ciência da Computação – Unidade Acadêmica de Ciências, Engenharias e Tecnologias – UNESC

anah.sour@gmail.com, {mem,pri}@unesc.net

Abstract. *The improvement about data storage entailed the increase of large databases, resulting in the necessity for knowledge discovery, distinguishing data mining among information analysis techniques. In this context, this article presents the modeling and development of the radial basis function network algorithm for the classification task in a free data mining tool called Shell Orion Data Mining Engine. This kind of neural network has as objective to classify non-linear databases according to the group of each element, performing a mapping of input data.*

Keywords: *Data Mining, Classification, Neural Networks, Radial Basis Function.*

Resumo. *Os avanços computacionais no que se refere ao armazenamento de dados ocasionaram a formação de grandes bases de dados resultando na necessidade de extração do conhecimento, destacando-se o data mining dentre as tecnologias para análise de informações. Deste modo, este artigo demonstra a modelagem e desenvolvimento do algoritmo de redes neurais com função de ativação de base radial para a tarefa de classificação em uma ferramenta gratuita de data mining denominada Shell Orion Data Mining Engine. Esta rede neural tem como objetivo classificar uma base de dados não-linear de acordo com o grupo a que cada registro pertence realizando um mapeamento dos dados de entrada apresentados.*

Palavras-chave*: Data Mining, Classificação, Redes Neurais, Radial Basis Function.*

1. Introdução

Analisar e extrair conhecimento útil de grandes bases de dados tornou-se um problema complexo para as organizações devido ao crescimento do volume de dados armazenados. Para facilitar esta análise são utilizadas ferramentas de *data mining* que são em sua maioria comerciais [Goldschimdt e Passos 2005].

A fim de disponibilizar uma ferramenta gratuita, o Grupo de Pesquisa em Inteligência Computacional Aplicada do Curso de Ciência da Computação da UNESC

mantém em desenvolvimento a *Shell* Orion *Data Mining Engine*, que já possui diferentes tarefas e métodos implementados.

Dentre as tarefas existentes, a classificação é uma das mais populares e consiste em encontrar propriedades comuns em um conjunto de registros de uma base de dados e relacioná-los a uma classe pré-definida. O método de redes neurais destaca-se nesta tarefa devido a sua capacidade de aprendizagem por experiência e classificação de dados não conhecidos.

Deste modo, neste artigo apresenta-se o desenvolvimento método de redes neurais com função de ativação de base radial para a tarefa de classificação na *Shell* Orion [Han e Kamber 2006].

2. Descoberta de Conhecimento em Bases de Dados

A descoberta do conhecimento em bases de dados auxilia na análise e extração de conhecimento útil de grandes bases. Este processo é dividido em três etapas [Han e Kamber 2006]:

a) **pré-processamento:** consiste na transformação dos dados para tornar possível a aplicação dos algoritmos;

b) **data mining:** refere-se efetivamente à busca por conhecimento e extração de padrões da base de dados, é considerada a etapa mais importante;

c) **pós-processamento:** realiza-se a análise e interpretação dos resultados obtidos com o *data mining,* para facilitar o entendimento do usuário.

2.1. *Data Mining*

Data mining é definido como um processo de reconhecimento de padrões no qual são aplicadas técnicas inteligentes a fim de extrair conhecimento implícito nas bases de dados e auxiliar no processo decisório. O uso desta técnica não é restrito as empresas, oferecendo vantagens também em áreas como saúde, economia, geologia dentre outras, devido à potencialização dos recursos computacionais e no constante aumento do volume de dados [Olson e Delen 2008].

As principais tarefas de *data mining* são:

a) **associação:** busca relações entre os dados que possam identificar uma tendência;

b) **clusterização:** agrupa elementos de uma base com características semelhantes entre si e diferentes de outros grupos;

c) **classificação:** associa cada registro de uma base de dados à uma classe;

d) **previsão:** prevê futuros valores de um índice por meio da análise do comportamento passado.

3. A Tarefa de Classificação no Processo de Data Mining

A classificação é uma tarefa preditiva que realiza o mapeamento dos registros de uma base de dados em uma quantidade finita de conjuntos, atribuindo cada elemento a uma categoria pré-definida [Han e Kamber 2001].

Nesta tarefa o conjunto de dados é dividido em dois grupos: dados de treinamento, composto pelos registros utilizados na fase de aprendizagem, e dados de teste, utilizados na avaliação do modelo gerado Na aprendizagem os dados para os quais as classes são conhecidas são utilizados na criação de um modelo classificador. Posteriormente os dados de teste são utilizados para estimar a capacidade do modelo em classificar dados não conhecidos e a habilidade de atribuir cada registro à classe correta [Russel e Norvig 2004].

4. O Método de Redes Neurais na Tarefa de Classificação

Redes neurais são estruturas nas quais os neurônios estão dispostos em camadas e interligados por conexões conhecidas como pesos sinápticos que representam o conhecimento da rede. Estas estruturas possuem a capacidade de classificar padrões desconhecidos adequando-se à resolução de problemas onde se tem pouco conhecimento das relações entre atributos e classes. São também capazes de adquirir conhecimento por meio de um conjunto reduzido de exemplos e produzir respostas consistentes para dados não conhecidos [Haykin 2001].

O processo de aprendizagem de uma rede neural ocorre por meio do ajuste dos seus pesos sinápticos de acordo com a resposta da rede aos dados de entrada. O modo como é realizado este ajuste é que determina o tipo do aprendizado da rede que pode ser supervisionado quando a rede aprende utilizando exemplos fornecidos por um supervisor externo, ou não-supervisionado, quando utiliza apenas os dados de entrada [Haykin 2001].

As redes neurais de uma só camada são capazes de resolver apenas problemas linearmente separáveis, ou seja, que podem ser satisfeitos por uma reta ou hiperplano como fronteira de decisão (Figura 1), pois utilizam algoritmos de treinamento capazes de ajustar os pesos de somente uma camada. Já a resolução de problemas de classificação não-lineares exige a utilização de algoritmos que ajustem mais de uma camada, por isso redes neurais com uma ou mais camadas ocultas são aplicáveis neste tipo de problema [Bishop 1995].

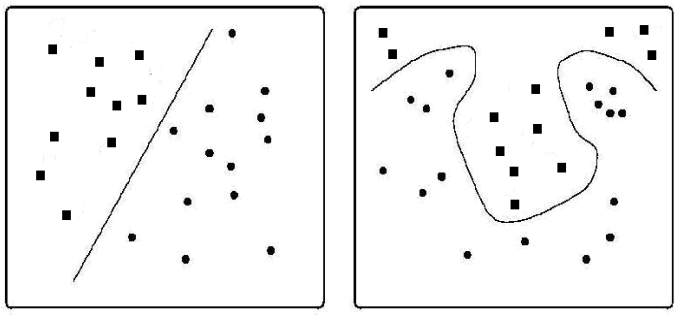

Figura 1. Classificação linear e não-linear

5. O Método de Redes Neurais com Função de Ativação de Base Radial

Uma rede neural com Função de Ativação de Base Radial (RBF) consiste em um modelo neural multicamadas, capaz de aprender padrões complexos e resolver problemas não-linearmente separáveis.

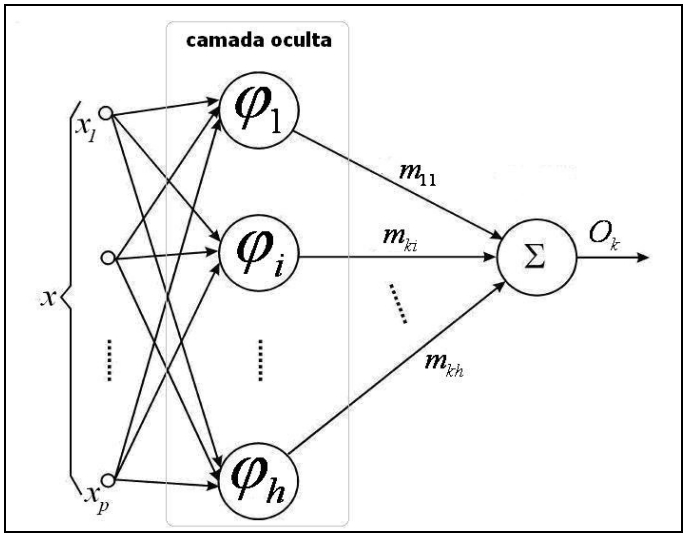

Figura 2. Arquitetura da rede RBF

A arquitetura de uma rede RBF está dividida em três camadas (Figura 2): camada de entrada, na qual os padrões são apresentados à rede; camada intermediária ou oculta que realiza o mapeamento não-linear do espaço de entrada utilizando função gaussiana; e camada de saída que fornece a resposta da rede ao padrão apresentado [Theodoridis e Koutroumbas 2006].

6. O Método de Redes Neurais com Função de Ativação de Base Radial na *Shell* Orion *Data Mining Engine*

A modelagem do módulo de classificação com redes RBF iniciou-se com a construção dos diagramas de caso de uso, atividades e seqüência utilizando os padrões UML. Posteriormente foi desenvolvida a demonstração matemática do funcionamento da rede a fim de facilitar o entendimento e a implementação.

O processo de aprendizado da rede RBF desenvolvida transforma um problema de classificação não-linear em um problema linear, e é dividido nas seguintes etapas:

a) **seleção dos centros (*c*):** um subconjunto dos dados de treinamento é atribuído aos vetores centro das funções de base radial;

b) **definição do raio de abrangência (σ):** calcula-se a área de sensibilidade da função de base em relação ao seu centro utilizando a seguinte equação:

$$\sigma = \frac{dist_{\max}(c_i, c_j)}{\sqrt{2H}}, \qquad \forall i \neq j \tag{1}$$

c) **cálculo da ativação dos neurônios ocultos (*u*):** define-se o grau de ativação de cada neurônio da camada oculta utilizando distância euclidiana conforme a equação (2)

$$u_i(t) = \| x(t) - c_i(t) \|, \qquad i = 1, \ldots H \tag{2}$$

d) **mapeamento do espaço não-linear (φ):** na camada oculta da rede, as funções gaussianas definidas pela equação (3) realizam a transformação dos dados de entrada não-lineares;

$$\varphi_i(t) = \exp\left(-\frac{{u_i}^2(t)}{2{\sigma_i}^2}\right) \quad (3)$$

e) **cálculo das saídas (*O*):** os pesos de saída da rede são atualizados de acordo com a regra do *perceptron* simples e utilizados na próxima iteração.

$$o_k(t) = \left\{\frac{1, \quad U_k(t) \geq 0}{0, \quad U_k(t) < 0}\right\} \quad (4)$$

Onde $U_k(t)$ é definido pela equação (5):

$$U_k(t) = \sum_{i=1}^{H} m_{ki}(t)\varphi_i(t) \quad (5)$$

f) **cálculo do erro (*e*):** diferença entre a saída desejada e a saída real da rede, onde:

$$e_k(t) = d_k(t) - o_k(t) \quad (6)$$

g) **ajuste das sinapses (*m*):** a atualização dos pesos sinápticos descrita na equação (7) ocorre somente quando o erro for diferente de zero.

$$m_{ki}(t+1) = m_{ki}(t) + \eta e_k \varphi_i(t) \quad (7)$$

h) **condição de parada (*E*):** o algoritmo atinge a convergência quando a rede não apresentar mudanças significantes nas sinapses. Essa condição pode ser verificada por meio da equação (8).

$$E = \frac{1}{N}\sum_{i=1}^{N}\left(e_k(t)\right)^2 \quad (8)$$

A apresentação de todos os vetores de treinamento à rede define uma época de treinamento, nesta fase a condição de parada é testada e se não for satisfeita, o conjunto de treinamento é embaralhado e a rede continua seu processamento iterativamente.

6.1. Implementação e Realização de Testes

A rede RBF foi implementada no módulo de classificação da Shell Orion Data Mining Engine por meio da linguagem de programação Java e ambiente de programação Netbeans 6.8.

A *Shell* Orion possibilita a conexão com *drivers* de diferentes bancos de dados, sendo que nos testes realizados nesta pesquisa optou-se pelo uso do MySQL 5.1, disponível gratuitamente para download em: http://dev.mysql.com/downloads/mysql.

Para executar a tarefa de classificação por meio de redes RBF é necessário definir alguns parâmetros da rede:

a) **quantidade de classes:** número de classes que o algoritmo irá identificar, o valor informado não pode ser maior que a quantidade real de classes;

b) **quantidade de épocas:** quantidade máxima de épocas executadas. Esta é uma condição de parada utilizada somente para casos em que o algoritmo não atinge a convergência por meio do cálculo do erro médio;

c) **taxa de aprendizagem:** taxa de atualização dos pesos sinápticos que corresponde ao grau de aprendizagem da rede;

d) **quantidade de centros:** quantidade de funções de base radial na camada oculta, este valor não pode ser muito alto para não ocasionar *overfitting*, nem muito baixo que gere *underfitting*;

e) **atributos de entrada:** atributos da base de dados que serão utilizados como valores de entrada da rede neural.

A *Shell* Orion permite que os resultados gerados pelo algoritmo possam ser analisados por meio de resumo, árvore e gráfico. Na Figura 3 observa-se o relatório gerado pelo algoritmo contendo um resumo da classificação

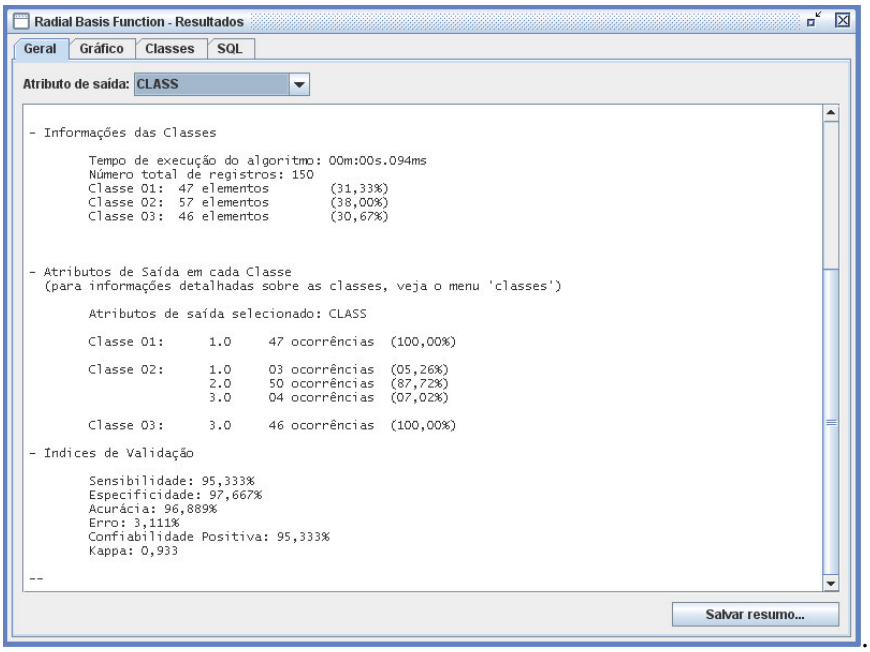

Figura 3. Resumo da classificação por meio da rede RBF

A atribuição dos registros para cada classe pode ser facilmente visualizada também em forma gráfica como mostra a Figura 4, onde as classes identificadas são representadas por meio de *Principal Component Analysis* (PCA). O método PCA transforma uma base de dados de n dimensões em uma matriz de duas dimensões, possibilitando a projeção dos dados graficamente.

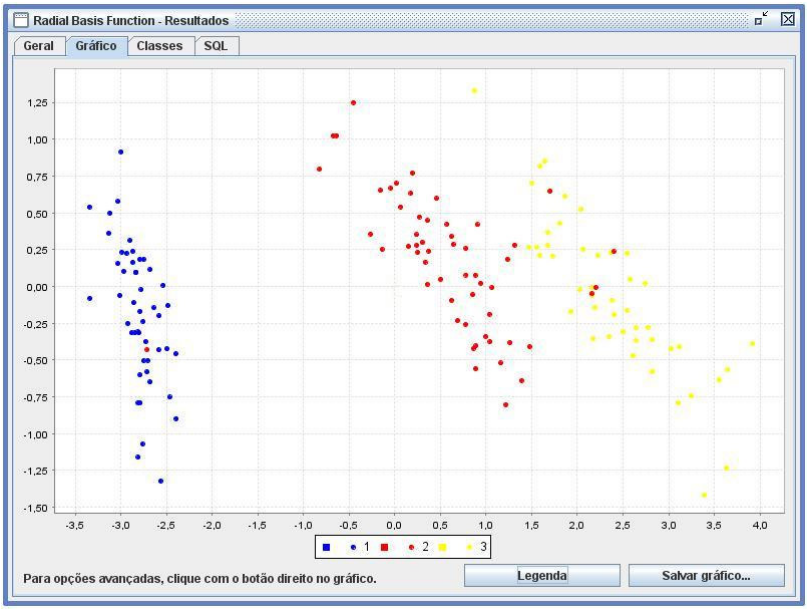

Figura 4. Gráfico gerado pelo classificador RBF

Detalhes dos elementos contidos em cada classe podem ser analisados individualmente por meio de uma estrutura em árvore (Figura 5).

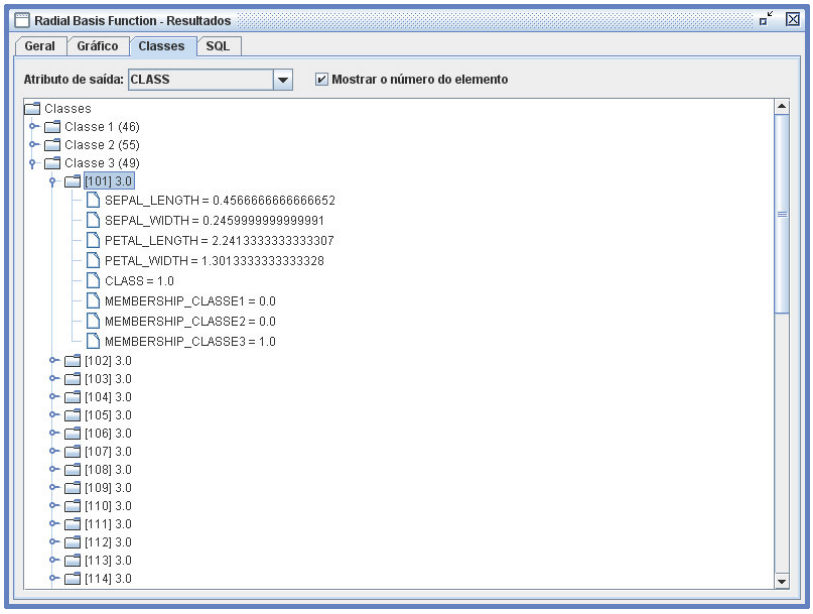

Figura 5. Árvore das classes identificadas pela rede RBF

Além disso, a ferramenta permite a exportação dos resultados gerados em formato de arquivo SQL, esta funcionalidade permite uma posterior aplicação dos resultados da classificação como entrada para outras tarefas de DM. É possível também executar diversas vezes o algoritmo com parâmetros de entrada diferentes, possibilitando a comparação dos resultados encontrados. Além disso, um arquivo de

ajuda disponibiliza a documentação necessária para auxiliar o usuário na utilização do classificador RBF.

6.2. Resultados Obtidos

Nos testes realizados no módulo desenvolvido utilizou-se a base de dados das Iridáceas, composta por entradas não-lineares, contendo dados referentes a três tipos de plantas da família das Iridáceas: setosa, versicolor e virgínica, totalizando 150 registros e 4 atributos (*sepal_length, sepal_width, petal_length* e *petal_width*) referentes a largura e comprimento das sépalas e pétalas das plantas.

O algoritmo foi executado com os seguintes parâmetros de entrada: 3 para quantidade de classes que se refere à quantidade real de classes existentes; 2000 para quantidade máxima de épocas, sendo que o algoritmo somente executa o número máximo de épocas se não atingir a convergência por meio do erro médio; taxa de aprendizagem de 0,1 e 20 funções de base radial, ambos definidos por tentativa e erro. Os resultados gerados pelo classificador RBF são descritos na Tabela 1.

Tabela 1. Resultados do classificador RBF para a base de dados das iridáceas

Classe	Quantidade de elementos	Porcentagem de elementos	Classe
1 (*íris-setosa*)	49	32,67%	1
2 (*íris-versicolor*)	55	36,67%	1 (1 ocorrência) 2 (50 ocorrências) 3 (4 ocorrência)
3 (*íris-virgínica*)	46	30,67%	3

Os resultados mostram que o algoritmo obteve desempenho satisfatório, identificando apenas cinco registros em classes incorretas e confirmou-se que os parâmentros de entrada selecionados influenciam.

6.3. Avaliação do Desempenho

A análise de desempenho do classificador desenvolvido realizou-se por meio da uma matriz de confusão, que combina os valores reais com os valores preditos pelo modelo (Tabela 2).

Tabela 2. Matriz de confusão para a classificação da base das iridáceas

Predita\Verdadeira	Classe 1	Classe 2	Classe3	Total
Classe 1	49	0	0	49
Classe 2	1	50	4	55
Classe 3	0	0	46	46
Total	50	50	50	150

Os elementos marcados em cinza representam a diagonal principal da matriz de confusão, que representam as concordâncias da classificação. Já os elementos de fora desta diagonal, descrevem as discordâncias da classificação, ou seja, elementos classificados incorretamente. A partir desta matriz é possível calcular os índices de avaliação de desempenho do classificador (Tabela 3).

Tabela 3. Índices de avaliação

Índice	Valor
sensibilidade	96,6%
especificidade	98,33%
accuracy	97,78%
erro	2,3%
confiabilidade	96,6%
kappa	0,95

6.4. Tempos de Processamento

A fim de analisar o desempenho do módulo no que se refere a tempo de processamento, utilizou-se uma base de dados gerada aleatoriamente contendo 6000 registros e 4 atributos.

Nesta avaliação foram testados diferentes valores para os seguintes parâmetros: quantidade de classes, quantidade de atributos de entrada, quantidade de centros e taxa de aprendizagem da rede. Observou-se que quanto maior a quantidade de classes e atributos de entrada, maior é o tempo de processamento da rede. Para taxa de aprendizagem com valores altos a rede apresentou melhores tempos no entanto mostrou pior desempenho na identificação de classes assim como a quantidade de funções de base.

6.4. Comparação com outra Aplicação

Os resultados gerados pelo classificador RBF para a base de dados das iridáceas na *Shell* Orion foi comparados com os resultados obtidos com a aplicação da mesma base de dados no classificador RBF da ferramenta gratuita Weka 3.6.2 disponível em: http://www.cs.waikato.ac.nz/ml/weka/. A Tabela 3 demonstra a comparação entre os índices de avaliação de desempenho de ambas as ferramentas.

Tabela 4. Tempos de processamento

	***Shell* Orion**	**Weka**
Registros classificados corretamente	145	146
Registros classificados incorretamente	5	4
Sensibilidade	96,6%	97,3%
Especificidade	98,3%	98,6%
Accuracy	97,7%	98,2%
Erro	2,3%	2%
Confiabilidade	96,6%	97,3%
Kappa	0,95	0,96
Tempo de processamento	00m:00s.187ms	00m:00s.140ms

Analisando os resultados obtidos pode-se concluir que o classificador RBF da Shell Orion está funcionando de maneira satisfatória, considerando que apresentou 96,6% de registros classificados corretamente e índices de avaliação muito próximos aos da ferramenta Weka que classificou os registros com 97,3% de acerto, porém a Weka teve um desempenho melhor perante os índices de avaliação e tempo de processamento pouco menor.

Ambas as ferramentas obtiveram valores excelentes (entre 0,8 e 1) para o coeficiente kappa, demonstrando alto grau de concordância entre os dois modelos classificadores.

. Considerando todos os testes realizados, o algoritmo apresentou resultados satisfatórios que confirmam o correto funcionamento do módulo desenvolvido.

7. Conclusão

Este artigo apresentou um modelo classificador implementado pelo método de redes neurais com função de ativação de base radial na Shell Orion *Data Mining* Engine, contribuindo com o desenvolvimento da ferramenta.

Diante dos resultados obtidos pode-se confirmar a aplicabilidade de redes RBF para a tarefa de classificação em bases de dados das quais já se possui algum conhecimento prévio, por utilizarem uma abordagem supervisionada e campos receptivos locais como fronteira de decisão. Concluiu-se que o modelo foi desenvolvido com sucesso, pois apresentou funcionamento correto e resultados satisfatórios na classificação e em tempos de processamento.

Referências

Bishop, C. (1995), Neural Networks for Pattern Recognition, Oxford University Press.

Goldschmidt, R., e Passos, E. L. (2005), Data mining: uma guia prático: conceitos, técnicas, ferramentas, orientações e aplicações, Elsevier.

Haykin, S. (2001). Redes neurais: princípios e prática, Bookman, 2. ed.

Russel, S. e Norvig, P. (2004), Inteligência Artificial, Elsevier.

Olson, D. e Delen, D. (2008), Advanced Data Mining Techniques, Springuer.

Kantardzic, M. (2003) Data mining: Concepts, Models, Methods, and Algorithms, John Wiley & Sons.

Han, J. e Kamber, M. (2006), Data Mining: Concepts and Techniques, Morgan Kaufmann, 2. ed.

Theodoridis, S. e Koutroumbas, K. (2006), Pattern recognition, Elsevier.