5CONTECSI June 04 - 06, 2008 São Paulo, Brazil

5th International Conference on Information Systems and Technology Management
5º Congresso Internacional de Gestão da Tecnologia e Sistema de Informação
De 04 a 06 de Junho de 2008 - São Paulo - Brasil

PS-919

KNOWLEDGE DISCOVERY VALIDITY BY HYBRID ARCHITECTURE (ROUGH SETS THEORY AND SELF-ORGANIZING MAPS) THROUGH MULTILAYER PERCEPTRON USING A COSTUMERS DATABASE

Renato José Sassi (University of São Paulo, São Paulo, Brasil) - sassi@lsi.usp.br
Leandro Augusto da Silva (University of São Paulo, São Paulo, Brasil) - leandro.augusto@poli.usp.br
Emilio Del Moral Hernandez (University of São Paulo, São Paulo, Brasil) - emilio_del_moral@ieee.org
http://www.lsi.usp.br/icone

The databases of real world contains a huge volume of data and among them there are hidden piles of interesting relations that are actually very hard to find out. The knowledge discovery databases (KDD) appear as a possible solution to find out such relations aiming at converting information into knowledge. However, not all data presented in the bases are useful to a KDD. Usually, data are processed before being presented to a KDD aiming at reducing the amount of data and also at selecting more relevant data to be used by the system. The purpose of this paper is to describe a validation methodology, through of a MLP neural network, to the knowledge discovery by a Hybrid Architecture composed by Rough Sets Theory used to pre-processing the data to be presented to Self-Organizing Maps neural network, which data cluster. The experimental results motivate the use of Hybrid Architecture in knowledge discovery databases.

Keywords: Hybrids Systems, Neural Network, Knowledge Discovery Systems, Rough Sets, Self-Organizing Maps, Multilayer Perceptrons.

VALIDAÇÃO DO CONHECIMENTO DESCOBERTO PELA ARQUITETURA HÍBRIDA (TEORIA DOS ROUGH SETS E REDE SELF-ORGANIZING MAPS) ATRAVÉS DE UMA REDE MULTILAYER PERCEPTRONS USANDO UMA BASE DE DADOS DE CONSUMIDORES

1 Introdução

Os avanços na área da Tecnologia da Informação têm possibilitado o armazenamento de grandes e múltiplas bases de dados. Esses dados produzidos e armazenados em larga escala são difíceis de serem analisados, interpretados e relacionados pelos métodos tradicionais, como planilhas de cálculo e relatórios informativos operacionais (PIATETSKY-SHAPIRO, 1991). Nesses grandes volumes de dados escondem-se diversas relações interessantes que, ao serem descobertas, acrescem conhecimento à tomada de decisão.

Por essa razão faz-se necessário o uso de sistemas que possam extrair o conhecimento dessas bases, viabilizando a análise dos dados, denominados de KDD (*Knowledge Discovery in Databases*) ou Descoberta de Conhecimento em Bases de Dados.

KDD é o termo criado pelo *Gartner Group* na década de oitenta para descrever todo o processo de extração de conhecimento dos dados devido ao crescimento vertiginoso das bases de dados das organizações.

Com toda essa informação armazenada, as organizações perceberam que não bastava apenas disponibilizá-la, era necessário interpretá-la, analisá-la e relacioná-la a fim de extrair conhecimento para apoiar a decisão (GOLDSCHMIDT e PASSOS, 2005).

O KDD pode ser definido como um processo de extração de conhecimentos válidos, novos, potencialmente úteis e compreensíveis para apoiar a tomada de decisão (FAYYAD, PIATETSKY-SHAPIRO e SMITH, 1996).

Entretanto, nem todos os dados que compõem as bases servem para um sistema descobrir conhecimento. Assim, é necessário pré-processar os dados antes de seguir com o processo de descoberta de conhecimento.

A fase de pré-processamento dos dados é importante, pois o sucesso dos resultados obtidos no processo de KDD depende diretamente da qualidade dos dados de entrada.

A redução de dados é uma forma de pré-processamento que visa obter uma representação reduzida dos dados, mas que produz os mesmos (ou quase os mesmos) resultados analíticos.

A redução de dados pode ser realizada por: agregação via cubo, redução de atributos (dimensão), compressão de dados, redução de casos, discretização e construção de hierarquias (LEE, 2001).

Em (SASSI, SILVA e HERNANDEZ, 2007), foi proposta uma Arquitetura Híbrida que combina a Teoria dos *Rough Sets* com uma rede neural artificial tipo Mapas Auto-Organizáveis ou rede SOM (*Self-Organizing Maps*). A Arquitetura Híbrida funciona da seguinte forma: a base de dados não rotulada (classificada) é apresentada ao *Rough Sets* realizando a redução dos atributos que provocam incerteza e em seguida, a base reduzida é apresentada à rede SOM para geração de *clusters* (agrupamentos), que são usados para rotulação da base. Utilizando medidas de desempenho aplicadas em SOM (a saber: quantização, topologia, número de *cluster* gerado (SASSI, SILVA e HERNANDEZ, 2007)), os experimentos comparativos mostraram melhores resultados quando usado o RS pré-processando os dados da rede SOM (Arquitetura Híbrida).

Vários trabalhos científicos descrevem as características vantajosas em utilizar redes neurais artificiais na solução de problemas, principalmente em classificação [BAETS, 94, HAYKIN, 99, ZHANG, 94]. Neste contexto, a rede neural artificial de arquitetura *Multilayer Perceptrons* (MLP) é a mais utilizada para dados rotulados, pela sua capacidade de separação não-linear.

Assim, a motivação deste trabalho reside em validar o conhecimento (rótulos gerados) por uma rede *Multilayer Perceptrons* (MLP), através da classificação, usando uma base de dados de consumidores.

O restante do paper é organizado da seguinte forma: na seção 2 discutem-se as fases e as tarefas da Descoberta de Conhecimento em Bases de Dados (KDD). Na seção 3 introduzem-se os conceitos fundamentais da Arquitetura Híbrida (Teoria dos *Rough Sets* (RS) e da rede *Self-Organizing Maps* (rede SOM)). Na seção 4 discute-se classificação e os conceitos fundamentais da rede *Multilayer Perceptrons* (MLP). A metodologia, experimentos realizados e os resultados são apresentados na seção 5 e finalmente na seção 6 a conclusão do trabalho.

2 Descoberta de Conhecimento em Bases de Dados (KDD)

O KDD é um processo composto por fases que devem ser desenvolvidas para atingir o objetivo final, que é a extração de conhecimento. As fases do KDD possuem numerosos passos, que envolvem um número elevado de decisões a serem tomadas, ou seja, é um processo interativo. É também um processo iterativo, pois ao longo do processo KDD, um passo será repetido tantas vezes quantas se fizerem necessárias para que se chegue a um resultado satisfatório (FAYYAD, PIATETSKY-SHAPIRO e SMITH, 1996).

No processo de KDD cada fase possui uma intersecção com as demais. Desse modo, os resultados produzidos numa fase são utilizados para melhorar os resultados das próximas. O KDD é composto das seguintes fases: seleção dos dados, pré-processamento dos dados, transformação dos dados, mineração de dados (*Data Mining*) e interpretação/avaliação do conhecimento.

A Figura 1 ilustra as fases do processo do KDD, com início na apresentação dos dados até chegar à obtenção do conhecimento. A iteração entre as fases pode ser observada pelas setas tracejadas.

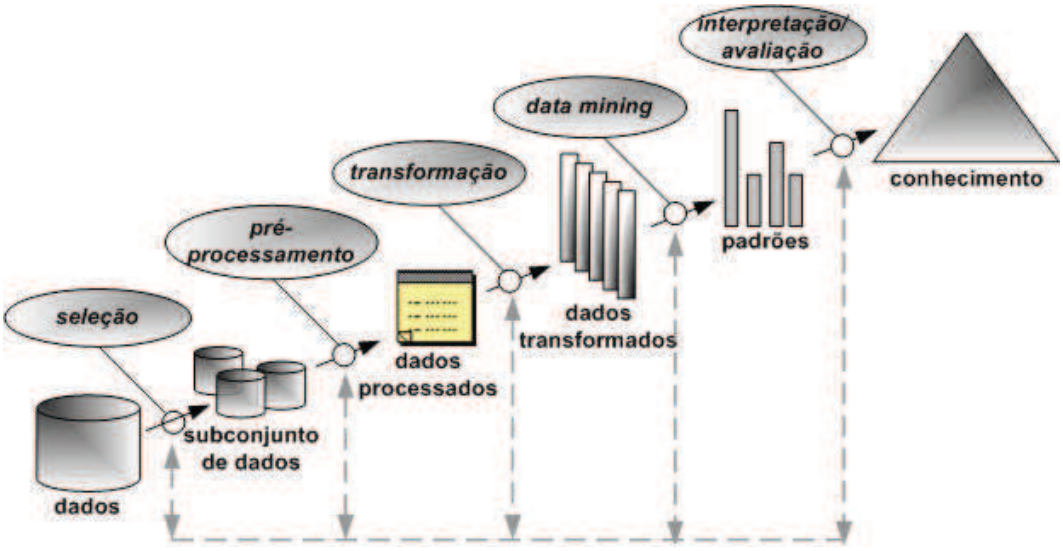


Figura 1: Fases do processo de KDD. Adaptado de Fayyad (FAYYAD, PIATETSKY-SHAPIRO e SMITH, 1996).

As três fases iniciais do KDD (Figura 1), que envolvem a seleção, o pré-processamento e a transformação, também chamadas de preparação dos dados, exigem bastante tempo, aproximadamente entre 60 e 80% do tempo utilizado em todo o processo, sendo que a maior parte desse tempo é consumida com a limpeza dos dados (PYLE, 1999). O foco será dado no pré-processamento, no *Data Mining* e na Interpretação/Avaliação do Conhecimento.

O pré-processamento dos dados tem por objetivo assegurar a qualidade dos dados selecionados (BATISTA, 2003).

A limpeza dos dados envolve a verificação da consistência das informações, a correção de possíveis erros e o preenchimento ou a eliminação de valores nulos e redundantes. Nessa fase são identificados e removidos os dados, duplicados e/ou corrompidos. A execução dessa fase corrige a base de dados eliminando consultas desnecessárias que seriam executadas pelo algoritmo minerador e que afetariam o seu processamento.

A fase de Transformação de Dados ou Codificação dos Dados tem como objetivo principal converter o conjunto bruto de dados em uma forma padrão de uso. Os dados de um atributo podem ser padronizados (normalizados) para cair dentro de uma faixa de valores, como por exemplo: -1,0 a 1,0 ou 0,0 a 1,0.

A transformação de dados para o atributo idade poderia ocorrer da seguinte forma: {0...18}. Faixa 1; {19...25}. Faixa 2; {26...30}. Faixa 3 e assim por diante [DOUGHERTY, KOHAVI e SAHAMI, 1995].

A mineração de dados é considerada a etapa mais importante do processo de KDD. Caracteriza-se pela existência do algoritmo minerador (*Data Mining*), que diante da tarefa especificada será capaz de extrair de modo eficiente conhecimento implícito e útil de um banco de dados.

Segundo Berry (BERRY, 1997), *Data Mining* é a exploração e análise, por meios automáticos ou semi-automáticos, de grandes quantidades de dados para descobrir modelos e regras significativas, permitindo a uma empresa aumentar, por exemplo, suas operações de *marketing*, vendas e apoio aos clientes pela melhor compreensão da clientela. De uma forma mais simples, *Data Mining* é produzir conhecimento novo escondido em grandes bases de dados.

O *Data Mining* usa técnicas baseadas em descobertas por meio de procura de padrões dos dados, o que é feito com o emprego de algoritmos inteligentes para encontrar relações fundamentais entre os dados (PIATETSKY-SHAPIRO, MATHEUS e CHAN, 1993).

As técnicas de *Data Mining* permitem avaliar como as perguntas se relacionam com as respostas (padrões e relações) encontradas. Achadas essas relações e padrões, é fornecida uma base de regras que servem de apoio aos processos de tomada de decisão. Para tal, utilizam-se técnicas baseadas em Inteligência Artificial, como as Redes Neurais Artificiais, as Árvores de Decisões, a Teoria dos Conjuntos *Fuzzy*, os Algoritmos Genéticos ou, ainda, combinações entre essas técnicas, gerando as chamadas Arquiteturas Híbridas (HAN e KAMBER , 2001).

O *Data Mining* pode responder a questões de negócio que tradicionalmente demandariam muito tempo para resolver. Ele explora as bases de dados à procura de padrões escondidos, encontrando dados que permitem prever tendências e comportamentos futuros, que os especialistas podem não descobrir devido ao fato de essa informação sair do limite de suas expectativas e possibilitando a tomada de decisão.

Após a fase de mineração de dados é necessária a interpretação do conhecimento descoberto, ou algum processamento desse conhecimento. Essa interpretação deve ser incluída no algoritmo minerador, porém algumas vezes é vantajosa a implementação separadamente. Em geral, a principal meta dessa fase é melhorar a compreensão do

conhecimento descoberto pelo algoritmo minerador, validando-o através de medidas da qualidade da solução e da percepção de um analista de dados.

Esses conhecimentos serão consolidados em forma de relatórios demonstrativos, com a documentação e explicação das informações relevantes ocorridas em cada etapa do processo de KDD. Uma maneira genérica de obter a compreensão e interpretação dos resultados é utilizar técnicas de visualização (BIGUS, 1996).

Existem várias formas de interpretação dos dados pelo KDD denominadas tarefas. As tarefas mais comuns são (FAYYAD, PIATETSKY-SHAPIRO e SMITH, 1996): associação, classificação, clusterização (agrupamento) e visualização

A associação também chamada de Análise de Cestas de Mercado (*Market Basket Analysis*).

A idéia principal da associação é identificar clusters de itens tipicamente associados (o que vai com o quê?) [AGRAWAL, 2001].

Dada uma coleção de itens e um conjunto de registros, cada qual contendo um número de itens da coleção, através de técnicas de associação realizam-se operações sobre o conjunto de registros retornando afinidades entre a coleção de itens.

O objetivo da técnica é encontrar tendências, a partir de grande número de transações que possam ser usadas para entender e explorar padrões de compra.

Regras facilmente encontradas assumiriam a forma "80% das vendas de cerveja também correspondem à venda de batatas fritas", ou "30% das vendas de fornos elétricos também correspondem à venda de luvas térmicas" ou, ainda, quando se compram batatas fritas, 65% das vezes também é adquirida coca-cola, a não ser que haja uma promoção, e nesse caso a coca-cola é campeã em 85% das vezes.

Uma vez encontradas as regras, estas poderiam ser usadas de maneiras diversas, de acordo com o tipo de atividade analisada. Poderiam ser realizadas mudanças na disposição de produtos, na determinação de políticas de formação de estoques ou de pedidos de compra, na promoção de campanhas direcionadas de vendas, no estabelecimento de práticas comuns (ou incomuns) de vendas. Na área financeira poderiam ser avaliados serviços que freqüentemente são adquiridos em conjunto.

As técnicas de classificação criam automaticamente um modelo a partir de um conjunto inicial de registros, que serve de exemplo e é chamado de conjunto de treinamento. Os registros do conjunto de treinamento devem pertencer a um pequeno grupo de classes pré-definidas (pelo analista) [HAN, 2001].

O modelo é composto de padrões, essencialmente generalizações em relação aos registros, os quais são usados para diferenciar as classes. Uma vez obtido o modelo, este é usado para classificar automaticamente os demais registros.

A classificação pode ser utilizada com êxito, por exemplo, para aplicações típicas de cartões de crédito. Dada uma base de dados dos usuários de cartões e tomando-se como exemplo aqueles cujo histórico de crédito é conhecido, atribui-se a estes um grau de risco de acordo com seu histórico (alto, médio ou baixo). Desse modo, por meio das técnicas de classificação, podem ser obtidas regras para a caracterização do grau de risco de um usuário tendo em vista dados como a faixa de renda familiar, a idade, a área de moradia (por exemplo: usuários de 30 a 50 anos, com renda familiar superior a 20.000 reais anuais, moradores da região X apresentam baixo risco).

De maneira análoga, numa aplicação de detecção de fraudes seriam obtidos registros de operações fraudulentas e não-fraudulentas. Posteriormente seriam estabelecidos os atributos desses registros, que tipificariam uma determinada operação como fraudulenta ou não, permitindo a classificação dos demais registros. O método se ajusta a uma vasta gama de situações, tais como concessão de empréstimos, detecção de sonegação fiscal etc.

A clusterização transforma registros com grande número de atributos em conjuntos relativamente menores (segmentos).

Essa transformação é realizada, automaticamente, por meio de identificação das características que distinguem o conjunto de dados e pelo seu posterior particionamento (HAN e KAMBER, 2001). Não é necessário identificar os agrupamentos desejados nem os atributos que devem ser utilizados para a produção dos segmentos.

O objetivo nessa tarefa é maximizar similaridade intra-*cluster* e minimizar similaridade extra-*cluste*r (GOLDSCHMIDT e PASSOS, 2005).

As ferramentas de visualização não são propriamente tarefas de *Data Mining*, mas sim meios de analisar e observar os dados de uma determinada base de dados (BERRY, 1997).

A visualização fornece meios de obter sumários visuais dos dados de uma base de dados. No caso de técnicas de clusterização, podem ser usadas ferramentas de visualização para determinar qual ou quais *clusters* criados são úteis ou interessantes para os métodos de *Data Mining*. No caso específico da rede SOM, as formas de visualização mais utilizadas são a Matriz-U (ULTSCH, 1989) e o Mapa por Similaridade de Cor.

As ferramentas de visualização podem ainda ser usadas como um mecanismo de compreensão da informação extraída por meio das técnicas de *Data Mining*. Características difíceis de detectar pela simples observação de linhas e colunas com valores numéricos podem se tornar óbvias se forem observadas graficamente.

Por meio dessas ferramentas podem ser encontrados características ou fenômenos pouco comuns ou interessantes sem que se esteja diretamente procurando por eles (LEE, 1995).

3 A Arquitetura Híbrida (Teoria dos *Rough Sets* e Rede *Self-Organizing Maps*)

Nesta sessão serão discutidos os principais conceitos da Teoria dos *Rough Sets* e rede *Self-Organizing Maps* (rede SOM) que formam a Arquitetura Híbrida.

3.1 Teoria dos *Rough Sets*

A Teoria dos *Rough Sets* (RS) foi proposta por Zdzislaw Pawlak (PAWLAK, 1982) em 1982 como um novo modelo matemático para representação do conhecimento e tratamento de incerteza, tendo sido usada, posteriormente, para o desenvolvimento de técnicas para classificação aproximada em aprendizado de máquina. Devido a essas características, tem-se utilizado essa teoria em Inteligência Artificial, especialmente nas áreas de aquisição de conhecimento, raciocínio indutivo e descoberta de conhecimento em base de dados.

Os conceitos de RS têm se mostrado muito úteis quando aplicados a problemas do tipo: redução de atributos, descoberta de dependência entre atributos e na descoberta de padrões entre os dados (PAWLAK, 1996).

Devido às características comentadas, a crescente utilização de RS pode ser comprovada pelo número de aplicações em KDD (ADJEI 2001, JIAJIN 2003).

A redução de atributos em RS é feita através dos chamados Redutos (RED), que são subconjuntos de atributos capazes de representar o conhecimento da base de dados com todos os seus atributos iniciais (PAWLAK, 1982). Este procedimento de eliminação de atributos irrelevantes é uma das características da Teoria.

Um reduto de B sobre um sistema de informação S é um conjunto de atributos B' $\subseteq$ B tal que todos os atributos **a** $\in$ (B – B') são dispensáveis. Com isso, U/INDs(B') = U/INDs(B). O termo RED(B) é utilizado para denotar a família de redutos de B.

O cálculo de reduções para gerar os redutos é um problema *n-p* completo, e seu processamento em grandes bases de dados exige grande esforço computacional.

Essa redução é feita pela função de discernibilidade, a partir da Matriz de Discernibilidade. Considerando o conjunto de atributos B = {Experiência do Vendedor, Qualidade do Produto e Boa Localização} para o sistema de informação S, o conjunto de todas as classes de equivalência determinadas por B sobre S é dado por U/INDs(B) = {{e1}; {e2, e3}; {e4}; {e5}; {e6}}, que estão representadas na Tabela 1.

	Atributos Condicionais			Atributo de Decisão
Loja	**Experiência do Vendedor**	**Qualidade do Produto**	**Boa Localização**	**Retorno**
e1	Alta	Boa	**Não**	Lucro
e2	Média	Boa	Não	Prejuízo
e3	Média	Boa	Não	Lucro
e4	Baixa	Média	Não	Prejuízo
e5	Média	Média	Sim	Prejuízo
e6	Alta	Média	Sim	Lucro

Tabela 1: Sistema de Decisão (Sistema de Informação com o atributo de decisão Retorno)

A Matriz de Discernibilidade do sistema de informação S, denotada por MD(B), é uma matriz simétrica n x n com: mD(i, j) = {**a** $\in$ B $\mid$ **a**(Ei) $\neq$ **a**(Ej)} para i,j = 1, 2, ...,n.. sendo 1≤i, j≤n e n=$\mid$ U / INDs(B). Logo, os elementos da matriz de discernibilidade mD(i, j) é o conjunto de atributos condicionais de B que diferenciam os elementos das classes com relação aos seus valores nominais.

Considerando Experiência do Vendedor (EV), Qualidade do Produto (QP) e Boa Localização (BL), com a finalidade de construir a Matriz de Discernibilidade MD(B), temos na Tabela 2 a sua representação:

	e1	e2	e3	e4	e5	e6
e1	∅					
e2	EV	∅				
e3	EV	∅	∅			
e4	EV, QP	EV, QP	EV, QP	∅		
e5	EV, QP, BL	QP, BL	QP, BL	EV, BL	∅	
e6	QP, BL	EV, QP, BL	EV, QP, BL	EV, BL	EV	∅

Tabela 2: Matriz de Discernibilidade

A função de discernibilidade Fs(B) é uma função booleana com *m* variáveis, que determina o conjunto mínimo de atributos necessários para diferenciar qualquer classe de equivalência das demais, definida como:

$$F_S(\mathbf{a}_1^*, \mathbf{a}_2^*, \ldots, \mathbf{a}_m^*) = \wedge\left\{\vee\, m_D^*(i,j) \mid i,j = 1,2,\ldots,n, \quad m_D(i,j) \neq 0\right\}.$$

Sendo: $m_D^*(i,j) = \left\{\mathbf{a}^* \mid \mathbf{a} \in m_D(i,j)\right\}$

Utilizando o método de simplificação de expressões booleanas na função Fs(B), obtém-se o conjunto de todos os implicantes primos dessa função, o qual determina os redutos de S. A simplificação é um processo de manipulação algébrica das funções lógicas com a finalidade de reduzir o número de variáveis e de operações necessárias para a sua realização (PATRICIO, 2005).

A função de discernibilidade Fs(B) é obtida da seguinte forma: para os atributos contidos dentro de cada célula da Matriz de Discernibilidade MD(B) (Tabela 4), aplica-se o operador "soma", "*or*" ou "∨" e, entre as células dessa matriz, utiliza-se o operador "produto", "*and*" ou "∧", resultando em uma expressão booleana de "Produto da Soma". A Fs(B) da Tabela 4 é representada por:

Fs(B) = (EV) ∧ (EV) ∧ (EV ∨ QP) ∧ (EV ∨ QP) ∧ (EV ∨ QP) ∧ (EV ∨ QP ∨ BL) ∧ (QP ∨ BL) ∧ (QP ∨ BL) ∧ (EV ∨ BL) ∧ (QP ∨ BL) ∧ (EV ∨ QP ∨ BL) ∧ (EV ∨ QP ∨ BL) ∧ (EV ∨ BL) ∧ (EV)

Simplificando esta expressão, utilizando teoremas, propriedades e postulados da Álgebra Booleana, obtém-se a seguinte expressão minimizada:

Fs(B) = (EV ∧ (QP ∨ BL) ∧ (EV ∨ QP ∨ BL)), que ainda pode ser escrita na forma de "Soma do Produto", ou seja Fs(B) = (EV ∧ (QP ∨ BL)). Os redutos são RED(B) = {Experiência do Vendedor, Qualidade do Produto } e {Experiência do Vendedor, Boa Localização}.

A função de discernibilidade determinou o termo mínimo da função, ou seja, determinou o conjunto mínimo de atributos necessários para discernir quais as classes formadas por todas as classes de equivalência da relação INDs(B).

3.2 *Self-Organizing Maps* (rede SOM)

Um Mapa Auto-Organizável (*Self-Organizing Maps*, ou SOM) é uma arquitetura de rede neural artificial, com aprendizado não supervisionado, baseada em um mapa de neurônios, cujos pesos são adaptados para verificar padrões semelhantes em relação a um conjunto de treinamento (KOHONEN, 2001). Sua principal característica é o mapeamento ordenado dos padrões de entrada de elevada dimensão em reticulados de neurônios de saída com dimensão menor, comumente duas, o que facilita a visualização dos dados.

Para uma dada base de dados com N amostras com *d* atributos cada, em que *d* determina a dimensão dos padrões de entrada, ocorrerá um mapeamento desses padrões para um reticulado de neurônios de saída arranjados em 2D (Figura 2).

A rede SOM é uma arquitetura de rede neural artificial, estruturada em duas camadas, entrada e saída. Os neurônios da camada de saída são comumente dispostos em um mapa de duas dimensões, com uma dada relação de vizinhança.

A Figura 2 ilustra essa arquitetura, com *d* atributos na camada de entrada e um conjunto de unidades *u* (neurônios) arranjados na forma de um mapa em 2D na camada de saída. Cada *u* é caracterizado por sua posição *x* e *y* no mapa, que é representado por *ux* e *uy,* respectivamente, que resulta em um vetor 2D igual a *u=[ux uy]*. Cada *u* tem associado um vetor protótipo *mu = [m1u, m2u,..., mdu],* sendo *d* a dimensão do protótipo, a mesma do padrão de entrada.

O algoritmo de aprendizado da rede SOM é realizado em um processo iterativo, onde, no primeiro passo, t=0, inicializa o vetor protótipo (m) randomicamente. Porém, a inicialização do m, pode ser feita de outras maneiras (KOHONEN, 2001).

O algoritmo de treinamento da rede SOM é também chamado de competitivo. Em cada passo do processo (ou época), uma amostra *x* é randomicamente escolhida do conjunto de treinamento. A distância, geralmente euclidiana, entre x e todos os vetores protótipos m é calculada. A unidade com menor distância, chamada de BMU (*best-matching unit*) é o *u* com protótipo m mais próximo a *x*, conforme a Equação 1:

$$||x - m_{bmu}|| = \arg \min_{u} || x\text{-}mu|| \quad (1)$$

A seguir, os vetores protótipos são atualizados. O BMU e sua vizinhança topológica são movidos para próximos a *x*, como se fosse um "arrasto", como mostra a Figura 4, que ilustra a atualização do neurônio vencedor (BMU) e sua vizinhança em direção a x. Os círculos em preto e cinza correspondem às situações antes e depois da atualização, respectivamente. As linhas mostram a relação da vizinhança.

A regra para a atualização dos vetores protótipos da unidade *u* é dada pela Equação 2:

$$mi(t+1) = mi(t) + \alpha(t)\, h_{bi}(t)\, [x - mu(t)] \quad (2)$$

onde *t* é o número de épocas, α (t) é a taxa de aprendizado e h_{bi} (t) é o *kernel* da vizinhança centrado no neurônio vencedor. O *kernel* pode ser Gaussiano, como na Equação 3:

$$h_{bi}(t) = e^{-\frac{\|r_b - r_i\|^2}{2\sigma^2(t)}} \quad (3)$$

onde r_b e r_i são as posições do neurônio vencedor *b* e do neurônio *i* no mapa da rede SOM e σ(t) é o raio da vizinhança. Conforme a distância (b,i) aumenta e o número de épocas (t) aumenta, $h_{bi} \rightarrow 0$. A taxa de aprendizado α(t) e o raio da vizinhança σ(t) diminuem monotomicamente com o tempo.

Devido às características da rede SOM de capacidade de quantização vetorial e de projeção vetorial, ele também pode ser utilizado na análise dos dados (KASKI & KOHONEN, 1996), (CURRY, 2003). A quantização vetorial é feita com a projeção de N amostras de entrada para m protótipos, que representam todo o conjunto de dados original.

A partir dos protótipos, realizam-se a formação de clusters e visualização das amostras em duas dimensões (JIN, SHUM, LEUNG et al; 2004). A Figura 2 traz, como exemplo, três *clusters* diferentes (dimensão d=3).

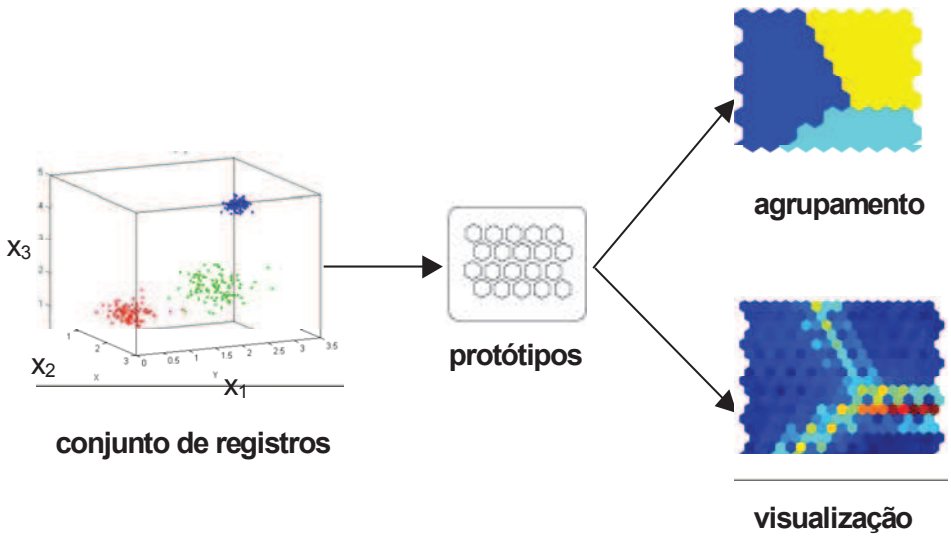


Figura 2: Passos da utilização de uma rede SOM na análise de dados.

O primeiro passo da rede SOM na análise de dados é a redução do conjunto de registros para os m protótipos, os quais são utilizados no agrupamento ou na visualização dos dados (Figura 2).

A motivação para o uso dos protótipos é que a complexidade computacional do passo subseqüente, por exemplo, o agrupamento de dados, é reduzida.

Quando utilizado na tarefa de clusterização, a rede SOM, em seu processamento das amostras, descarta ruídos com média zero e efeito de amostras discrepantes (*outliers*) na geração dos vetores protótipos (VESANTO e ALHONIEMI, 2000). Com os vetores protótipos, outros algoritmos de clusterização podem ser utilizados, como a própria rede SOM ou o K-médias. Um estudo comparativo entre a rede SOM e K-médias é apresentado por (ULTSCH, 1995).

Após o treinamento do SOM, o mapa se torna uma interessante ferramenta para a visualização dos dados, encerrando a última fase do KDD. Para tanto, existem duas propostas principais. A primeira é o uso da Matriz-U, ou matriz unificada (ULTSCH, 1993) e a segunda é o Mapa por Similaridade de Cor (Figuras 7 e 9).

A Matriz-U (ULTSCH, 1993) é um método que permite a construção de um mapa 3D (três dimensões), o qual considera a diferença entre os neurônios adjacentes. Assim, neurônios adjacentes com valor de protótipo semelhantes, resultarão em distâncias pequenas e conseqüentemente em cores claras, as quais representam vales (Figura 2). Do contrário, ou seja, valores altos de distâncias entre os protótipos dos neurônios adjacentes, resultarão em cores escuras, as quais representam montanhas. Dessa maneira, visualmente consegue-se inferir sobre o número de *clusters* da base, de forma visual, através dos vales separados pelas montanhas exibidos. Acontece que nem sempre a visualização é nítida, na maioria dos casos é impossível inferir o número de clusters com uso da Matriz-U.

A segunda proposta para o uso dos protótipos do SOM na visualização de dados é segmentar a grade bidimensional do mapa por similaridade de cores que destacam duas propostas.(Figura 2). Na primeira proposta, Costa e Netto (COSTA e NETTO, 2001) propuseram o uso da técnica de processamento de imagens conhecida como *watershed*, para segmentar a Matriz-U, e o resultado é a rotulação do mapa em clusters semelhantes. Na segunda proposta, feita por Vesanto e Alhoniemi (VESANTO e ALHONIEMI, 2000), utiliza o k-médias, que é um método tradicional de agrupamento de dados, para segmentar o mapa de neurônios do SOM. O esquema desta segmentação está ilustrado na Figura 3 com a presença da Arquitetura Híbrida proposta.

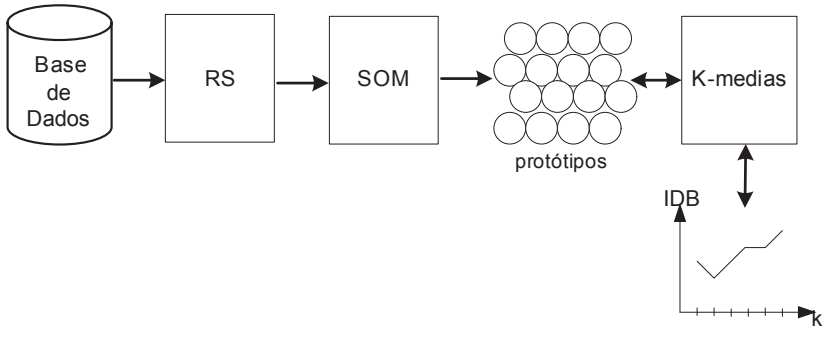


Figura 3: Esquema para segmentação do mapa SOM (VESANTO e ALHONIEMI, 2000) com a presença da Arquitetura Híbrida proposta.

O esquema da Figura 3 mostra resumidamente que a rede SOM é treinada a partir de uma base de dados pré-processada pelo *Rough Sets* gerando os vetores protótipos. Com base nos vetores protótipos, aplica-se o k-médias, variando o número de k e, a cada variação, o índice de Davies-Bouldin (IDB) (DAVIES e BOULDIN, 1979) é calculado. O IDB é uma medida que considera a relação da dispersão intra-cluster e a dispersão extra-*cluster*. Assim, o menor IDB que indica o número de *clusters* no protótipo ocorrerá quando a dispersão intra-cluster for pequena e a dispersão extra-cluster for grande. A partir desta informação, é aplicado o k-médias, com k que resultou em menor IDB, e assim rotula-se o mapa. Para os experimentos apresentados a seguir, usa-se esta metodologia para rotular o mapa SOM e assim definir os potenciais *clusters* da base de dados.

4 Classificação

Uma rede neural artificial (RNA) pode ser treinada para, a partir de exemplos (registros) de uma mesma classe ou de um conjunto de classes e usada para separar os elementos determinando se algum registro pertence a uma classe ou não.

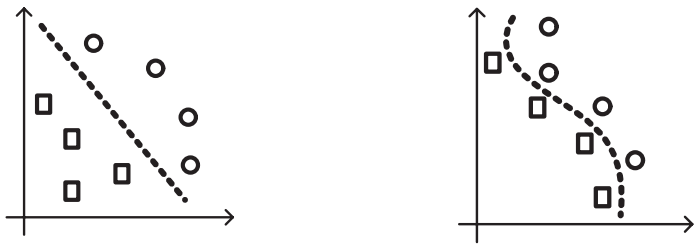
A RNA pode, por exemplo, classificar os itens comprados pelas pessoas. Essa classificação pode ser utilizada para verificar a relação entre itens de compra e fornecer para a empresa o perfil dos seus clientes, além de definir novas estratégias de negócios, realizar promoções, determinar a necessidade de aquisição de novos produtos para vender ou até trocar de atividade comercial. Enfim, a utilização de RNA para fazer classificações pode ser uma ferramenta muito útil para uma empresa gerenciar o seu relacionamento com os clientes [BERRY, 1997].

Na Arquitetura Híbrida (SASSI, SILVA e HERNANDEZ, 2007), o RS pré-processa dados para a rede SOM organizar, segmentar e rotular a base de dados. Agora, a rede MLP valida os *clusters* gerados pela rede SOM através da classificação dos registros da base rotulada. A taxa de classificação, medida pelo número de acertos, indicará a discernibilidade dos *clusters* formados. Uma vez validado, quando usado na prática, apenas a rede MLP se fará necessária.

4.1 Rede *Multilayer Perceptrons* (MLP)

A *MultiLayer Perceptrons* é uma das redes neurais artificiais mais utilizadas em classificação. Ela permite classificar dados não-linearmente separáveis, usando uma função de ativação não linear, geralmente a função sigmoidal (Figura 6).

Na Figura 4 pode-se observar a diferença entre dados linearmente separáveis (Figura 4a) e dados não-linearmente separáveis (Figura 4b).



a) dados linearmente separáveis b) dados não-linearmente separáveis

Figura 4: Dados linearmente separáveis (a) e dados não-linearmente separáveis (b).

As redes neurais de arquitetura MLP tipicamente consistem de uma especificação do número de camadas, tipo de função de ativação de cada unidade e pesos de conexões entre

as diferentes unidades que devem ser definidas para a construção desta arquitetura neural. A Figura 5 ilustra uma arquitetura do tipo MLP com Ne_m neurônios na camada de entrada (neurônios sensores), No_n neurônios na camada escondida e com um único neurônio de saída (Ns_1).

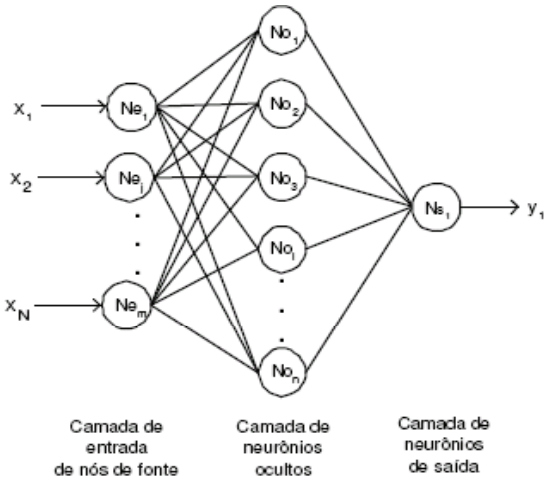


Figura 5: Ilustração de uma rede MLP

O algoritmo de treinamento usado na MLP é o *error backpropagation* (HAYKIN, 1999). Ele funciona da seguinte maneira: primeiramente apresenta-se um padrão - nesse trabalho um padrão será um vetor protótipo e o respectivo rotulo - à camada de entrada da rede. Esse padrão é processado camada por camada até que a camada de saída forneça a resposta processada, $f_{MLP,}$ calculada como mostrado a seguir:

$$f_{MLP}(\mathbf{x}) = \varphi\left(\sum_{l=1}^{N_{on}} v_l \cdot \varphi\left(\sum_{j=1}^{N_{em}} w_{lj} x_l + b_{l0}\right) + b_0\right), \quad (4)$$

onde v_l e w_{lj} são pesos sinápticos; b_{l0} e $_{b0}$ são os biases; e φ a função de ativação, comumente especificada como sendo a função sigmóide (Figura 6).

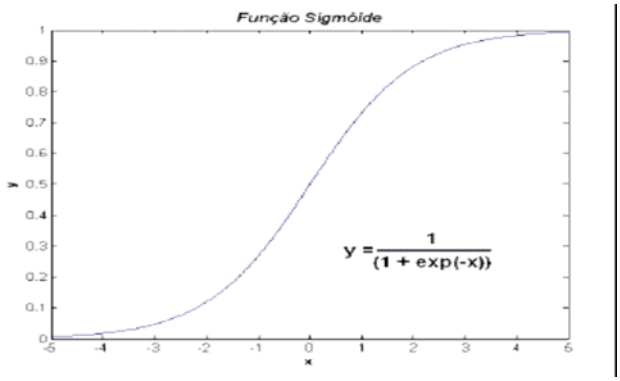


Figura 6: Exemplo da função sigmóide.

O objetivo do processo de treinamento é escolher parâmetros adequados para minimizar uma função de custo pré-determinada. Esta função é dependente da resposta desejada y_i – no contexto do trabalho será o rotulo gerado pela Arquitetura Híbrida - e havendo erro, esse é calculado. A função da soma do erro quadrático é a mais usual:

$$E(X) = \sum_{i=1}^{N} \frac{1}{2} [f_{MLP}(x_i) - y_i]^2 . \quad (5)$$

onde N é o número de padrões do conjunto de treinamento.

O erro calculado é retropropagado da camada de saída para a camada de entrada e conforme isso acontece, os pesos são ajustados e o processamento é feito novamente, até que se obtenha um erro mínimo.

A arquitetura (conexões, número de neurônios na camada escondida e função de ativação) é usualmente fixada como um conhecimento a priori; por outro lado, os pesos são ajustados durante o processo de treinamento pela retropropagação do erro.

Pela capacidade em lidar com dados não lineares, o que é uma característica interessante para aplicações em *marketing* que lidam com dados como vendas e preços, as redes MLP são muito utilizadas [ZHANG, 1994].

Assim, a motivação deste trabalho reside em validar o conhecimento (rótulos gerados) por uma rede *Multilayer Perceptrons* (MLP), através da classificação, usando uma base de dados de consumidores.

Após o treinamento da rede SOM, os neurônio do mapa são rotulados. Cada neurônio do mapa tem um vetor protótipo. A rede MLP será treinada com este padrão, vetor protótipo e rótulo. Desta maneira, todo o conhecimento gerado é incorporado pelos pesos da MLP. A validação do conhecimento gerado será feito após o treinamento da MLP com o uso dos registros da base de dados. Esta metodologia será melhor discutida na próxima sessão.

5 Metodologia, Experimentos e Discussões

O experimento realizado comparou o conhecimento (*clusters*) gerado pelo RS e rede SOM (Arquitetura Híbrida) com a rede SOM, analisando-se os resultados obtidos em termos de taxa de acerto na classificação realizada pela rede MLP

Para os experimento, utilizou-se a base de dados Consumidor (FERNANDEZ, 2003; SAS, 2005). Esta é uma base que contém informações do consumo de 1.968 consumidores (registros) com 48 atributos, sendo 47 atributos condicionais e um atributo de decisão dividido em duas classes (M = masculino; F = feminino). O atributo que identificava o número da conta do consumidor foi eliminado da base. O atributo prefixo do nome é nominal e portanto foi transformado em numérico para poder ser processado pela rede SOM. A transformação foi realizada da seguinte forma: Ms = 0; Mr = 1; Mrs = 2 e None = 3.

Para realização dos experimentos com a rede SOM foi utilizada a ferramenta SOM *Toolbox* (SOM, 1999), uma implementação do Mapa Auto-Organizável de Kohonen em *Matlab*. A escolha dessa ferramenta baseou-se no grande número de trabalhos publicados que relatam a sua utilização (DUTRA, 2001; VESANTO, 2000). Esta ferramenta pode ser considerada como uma plataforma padrão que tem sido adotada em grande parte das pesquisas atuais com Mapas Auto-Organizáveis. Para a realização dos experimentos com RS foi utilizada a ferramenta chamada Rosetta (A *Rough Sets* Toolkit *for Analisys of Data*) (ROSETTA, 1997), de domínio público. A escolha da ferramenta Rosetta foi embasada no grande número de publicações que relatam a sua utilização (KOMOROWSKI, 1997; ØHM, 1999). A rede MLP foi implementada em rotinas *Matlab*.

A plataforma de hardware utilizada nos experimentos foi um Pentium IV com 2.4 MHZ, 512 MB de memória RAM e 40 GB de disco rígido.

A escolha dos parâmetros da rede SOM aplicadas no trabalho, tanto de inicialização quanto de treinamento do mapa, ainda não segue regras bem definidas. Mesmo lançando mão das heurísticas existentes para definição dos parâmetros, é consenso entre os

pesquisadores que se devem efetuar alguns testes com diferentes configurações de mapas antes de decidir-se qual representa melhor o conjunto de dados em questão.

Em geral, são usadas heurísticas baseadas no comportamento do mapa e em medidas de qualidade como erro de quantização e erro topográfico. O erro de quantização (EQ) corresponde à média das distâncias entre cada vetor de dados e o correspondente vetor de pesos do neurônio BMU. A medida corresponde à resolução do mapa. O erro topográfico (ET) quantifica a capacidade do mapa em representar a topologia dos dados de entrada. Para cada vetor de dados de entrada são calculados seu primeiro BMU e o seu segundo BMU e toda vez que eles não forem adjacentes (vizinhos, próximos), aumenta-se o erro em uma unidade, tirando-se depois a média pelo número total de vetores [KIVILUOTO, 1995].

Os parâmetros que regulam a rede SOM podem ser agrupados em dois conjuntos: Parâmetros de estrutura: Dimensões (tamanho do mapa, número de neurônios), Vizinhança (hexagonal ou retangular) e Formato do Arranjo (Folha, Cilindro ou Toróide) e Parâmetros de treinamento que correspondem ao número de épocas de treinamento.

Nos experimentos realizados variaram-se alguns dos parâmetros de estrutura para conhecer a sua influência nos resultados. Dessa forma, todos os parâmetros descritos nesta seção foram utilizados nos experimentos tanto na primeira fase como na segunda, e são os seguintes:

Parâmetros de estrutura:

-Dimensões: número de neurônios 15X15 = 225 neurônios.

-Vizinhança: hexagonal (KOHONEM, 1997).

-Formato do Arranjo: folha.

Parâmetros de treinamento: o treinamento de um mapa na SOM *Toolbox* é dividido em 2 etapas: "*rough phase*" e "*fine tune*". Para cada uma destas etapas são definidos diferentes números de épocas. Os valores *default* (VESANTO, 2000) são: para a "*rough phase*" = 10 x mpd épocas, e para a "*fine tune*" = 40 x mpd épocas, onde mpd = neurônios/dados. A taxa de aprendizado foi de 0,5 para a fase inicial e 0,05 para a fase de convergência (KASKI, 1997). O número de épocas foi calculado da seguinte forma: *rough phase* = quantidade de neurônios (225) / tamanho de dados (1968) x 10 = 12,0 e a *fine tune* = quantidade de neurônios (225) / tamanho de dados (1968) x 40 = 58,0. Somando as duas fases (12,0 + 46,0) temos 58 épocas.

A mesma observação feita anteriormente na rede SOM sobre a escolha dos parâmetros, serve para MLP, a não existência de regras bem definidas. Os parâmetros da rede MLP adotados nos experimentos foram os seguintes: número de neurônios na camada escondida igual a 15, o erro máximo permitido igual a 0.01 e o número de iterações igual a 20000.

Os experimentos foram realizados em duas fases distintas. Na primeira fase apresentou-se à rede SOM a base de dados consumidor com todos os atributos condicionais (47). Após o treinamento da rede SOM, rotulo-se o mapa com método K-médias e IDB, descrito na sessão 3.2 (Figura 3). Este rotulo pode ser usado para classificar a base. O número de cluster gerado foi igual a 15 clusters (Figura 7).

Para a validação dos clusters, experimentos de classificação com a rede MLP serão conduzidos. O treinamento da MLP será feito usando os 225 (15 x 15) vetores protótipos rotulados de 1 a 15 (veja barra de cores Figura 7). Para a classificação, os registros são projetados no mapa, afim de, encontrar o neurônio vencedor que fornecerá rotulos a base. Na Figura 8, o Histograma mostra a quantidade de registros que compõem cada cluster.

Na tabela 3 observa-se a matriz de confusão que contém dados de acerto da rede MLP em classificar os registros nos 15 clusters gerados pela rede SOM A diagonal principal (em destaque) mostra a taxa de acerto por classe (cluster) da rede MLP na tarefa de classificação e a soma total dos valores da diagonal principal mostra a taxa de acerto total da rede. Assim, a taxa de acerto total da rede MLP ficou em 56%. Os resultados fora da diagonal principal indica a taxa de erro em relação aos demais *clusters*.

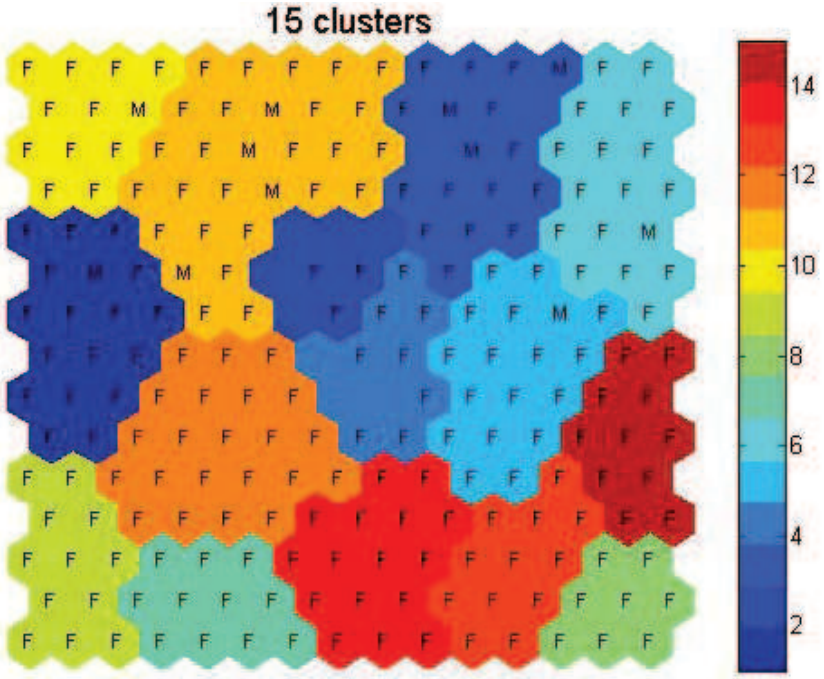


Figura 7: Visualização dos 15 *clusters* gerados pela rede SOM sem redutos, rotulados de acordo com as classes da base de dados Consumidor (primeira fase do Experimento).

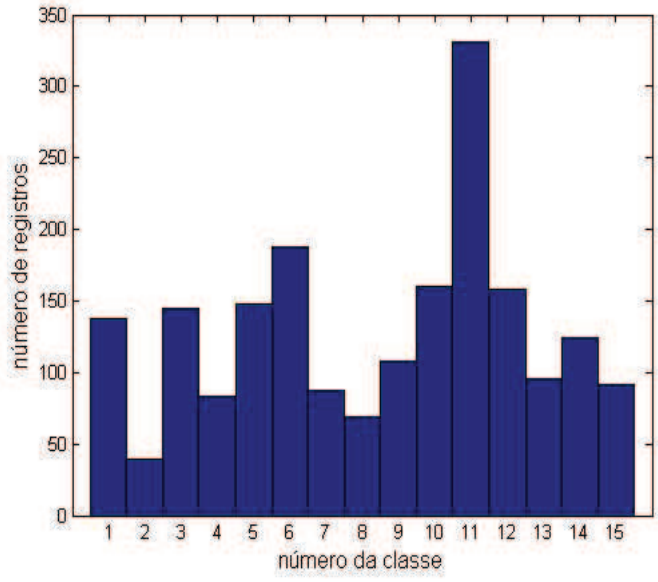


Figura 8: O Histograma mostra a quantidade de registros que compõem cada um dos 15 *clusters* gerados pela rede SOM sem RS.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	**0.00**	0.08	0.02	0.05	0.03	0.07	0.04	0.00	0.04	0.12	0.17	0.27	0.00	0.10	0.00
2	0.00	**0.79**	0.00	0.00	0.00	0.08	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.08	0.00
3	0.00	0.01	**0.65**	0.04	0.08	0.05	0.03	0.01	0.00	0.02	0.03	0.06	0.00	0.01	0.01
4	0.00	0.00	0.05	**0.55**	0.08	0.08	0.01	0.00	0.01	0.01	0.00	0.08	0.01	0.06	0.04
5	0.00	0.00	0.03	0.05	**0.55**	0.04	0.05	0.01	0.00	0.04	0.08	0.07	0.00	0.05	0.03
6	0.00	0.00	0.06	0.01	0.03	**0.69**	0.05	0.00	0.01	0.00	0.09	0.03	0.00	0.03	0.01
7	0.00	0.01	0.03	0.00	0.03	0.10	**0.60**	0.00	0.00	0.02	0.03	0.03	0.01	0.10	0.01
8	0.00	0.00	0.00	0.01	0.00	0.01	0.00	**0.54**	0.06	0.00	0.00	0.00	0.17	0.01	0.19
9	0.00	0.00	0.01	0.00	0.01	0.17	0.06	0.06	**0.48**	0.07	0.01	0.06	0.02	0.01	0.05
10	0.00	0.03	0.03	0.02	0.12	0.00	0.03	0.00	0.03	**0.63**	0.11	0.01	0.00	0.01	0.00
11	0.00	0.02	0.07	0.00	0.04	0.09	0.01	0.00	0.00	0.04	**0.67**	0.05	0.00	0.01	0.00
12	0.00	0.00	0.00	0.03	0.00	0.09	0.01	0.00	0.00	0.01	0.04	**0.73**	0.00	0.08	0.00
13	0.00	0.00	0.01	0.00	0.05	0.00	0.06	0.09	0.03	0.10	0.08	0.06	**0.36**	0.09	0.04
14	0.00	0.06	0.00	0.02	0.06	0.04	0.03	0.02	0.00	0.01	0.02	0.05	0.06	**0.60**	0.03
15	0.00	0.00	0.02	0.09	0.02	0.13	0.02	0.05	0.04	0.00	0.00	0.02	0.03	0.07	**0.50**

Tabela 3: Taxa de acerto por classe da rede MLP

Ainda na Tabela 3 pode-se observar na primeira linha, uma taxa baixa de acerto (0) e um alto valor no erro de classificação (0,27) entre o *cluster* 1 e o *cluster* 12. Pode-se ver na barra de cores da Figura 7 que o *cluster* 1 e *cluster* 12 fazem fronteira. A mesma análise pode ser feita nas outras linhas da Tabela 3.

Toda a metodologia discutida para a primeira fase do experimento será repetida na segunda fase. Nesta, os atributos condicionais (47) foram reduzidos pelo RS, gerando 87 redutos, sendo 16 redutos com 3 atributos, 63 redutos com 4 atributos e 8 redutos com 5 atributos. Escolheram-se, então, os 16 redutos que apresentaram um número menor de atributos (3) com base no menor esforço computacional (MITCHELL, 1997). Dentre os 16 redutos a escolha foi aleatória, já que apresentaram resultados semelhantes de EQ e de ET. O reduto escolhido foi o formado pelos seguintes atributos: valor da casa do consumidor, freqüência de pedidos e idade do consumidor. Finalmente a base de dados reduzida foi apresentada à rede SOM gerando 8 clusters (Figura 9).

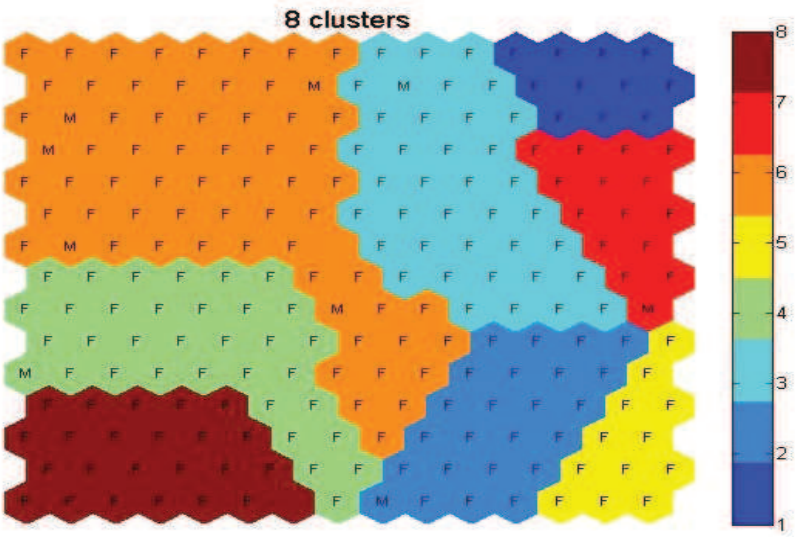


Figura 9: Visualização dos 8 *clusters* gerados pela rede SOM com redutos rotulados de acordo com as classes da base de dados Consumidor (segunda fase do experimento).

O próximo passo é validar este conhecimento gerado pela rede SOM classificando os registros da base de dados através de uma rede MLP. Na Figura 10 o Histograma mostra a quantidade de registros que compõem cada *cluster* gerado pela rede SOM com RS. Pode-se observar que agora os registros estão alocados em apenas 8 clusters, ou seja, quase a metade dos *clusters* gerados no primeiro experimento.

Na tabela 4 observa-se a matriz de confusão que contém dados de acerto da rede MLP em classificar os registros nos 8 clusters gerados pela rede SOM com RS (Arquitetura Híbrida). A diagonal principal (em destaque) mostra a taxa de acerto por classe (*cluster*) da rede MLP na tarefa de classificação e a soma total dos valores da diagonal principal mostra a taxa de acerto total da rede. A taxa de acerto total da rede MLP ficou em 91,38%, ou seja, a rede MLP conseguiu classificar corretamente 91,38% dos registros alocados nos *clusters*.

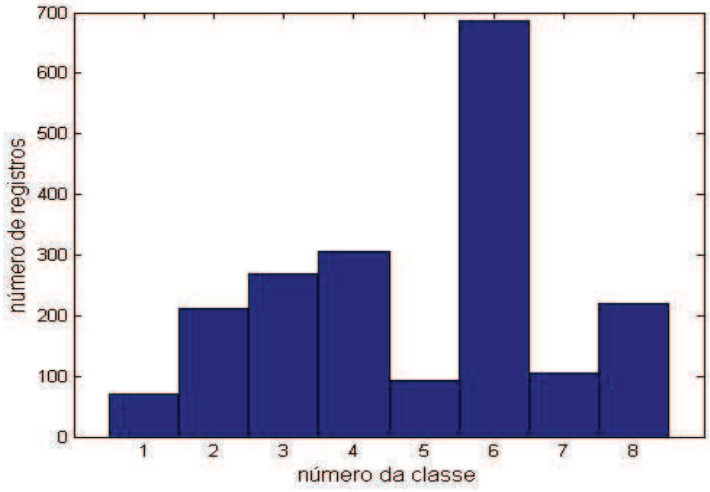


Figura 10: O Histograma mostra a quantidade de registros que compõem cada um dos 8 *clusters* gerados pela rede SOM com RS.

	1	2	3	4	5	6	7	8
1	**0.92**	0.00	0.07	0.00	0.00	0.00	0.00	0.01
2	0.00	**0.78**	0.02	0.01	0.08	0.11	0.00	0.00
3	0.01	0.00	**0.96**	0.01	0.00	0.01	0.01	0.00
4	0.00	0.00	0.01	**0.96**	0.00	0.02	0.00	0.01
5	0.00	0.10	0.00	0.00	**0.88**	0.00	0.02	0.00
6	0.00	0.01	0.00	0.03	0.00	**0.96**	0.00	0.00
7	0.02	0.00	0.05	0.00	0.02	0.00	**0.91**	0.00
8	0.04	0.00	0.00	0.01	0.00	0.00	0.00	**0.95**

Tabela 4: Taxa de acerto por classe da rede MLP

Ainda na Tabela 4 pode-se observar que o valor mais alto no erro de classificação ocorreu na linha 2 entre o *cluster* 2 e o *cluster* 6 (0,11). Pode-se ver na barra de cores da Figura 9 que o *cluster* 2 e o *cluster* 6 fazem fronteira.

A taxa de acerto mostra que a Arquitetura Híbrida gera conhecimento mais valido que a rede SOM. O motivo da melhora pode ser justificado pela utilização do RS no pré-processamento da base de dados consumidor. O RS, eliminou informação desnecessária ou redundante apresentada à rede SOM que gerava dificuldade na formação dos *clusters*. A indiscernibilidade entre os *clusters* da rede SOM foi revelada na classificação feita pelo MLP, que trabalha bem com dados não-linearmente separáveis.

Esta conclusão motiva o uso da Arquitetura Híbrida na descoberta de conhecimento em bases de dados. Pode-se então, explorar os resultados obtidos em *marketing* para direcionar de forma mais eficaz e eficiente o envio de mala direta e de *e-mail marketing* direcionando mensagens apenas aos consumidores que se interessam ou que possam se interessar pelo produto ou serviço auxiliando inclusive no desenvolvimento de técnicas adequadas de *telemarketing* ao público-alvo.Os resultados obtidos também podem ser utilizados como fonte de informação para o CRM analítico ajudando a transformar o conhecimento do cliente em oportunidade gerando valor para empresa e aumento de receita. A descoberta de diferentes hábitos e perfis de consumo pode auxiliar na alteração de produtos ou serviços já existentes e no desenvolvimento de novos produtos ou serviços.

6 Conclusão

Como citado em Goldschmidt (GOLDSCHMIDT e PASSOS, 2005), técnicas podem ser combinadas para gerar as chamadas Arquiteturas Híbridas. A grande vantagem desse tipo de sistema deve-se ao sinergismo obtido pela combinação de duas ou mais técnicas. Este sinergismo resulta na obtenção de um sistema mais poderoso (em termos de interpretação, de aprendizado, de estimativa de parâmetros, de generalização, dentre outros) e com menos deficiências.

Foi o que ocorreu com o desempenho da Arquitetura Híbrida (rede SOM com redutos, segunda fase do experimento).

Como os *clusters* são obtidos por intermédio da aplicação dos conceitos de similaridade e distância (JOHNSON e WICHERN, 1998), um número menor de *clusters* indica melhor similaridade entre os dados e maior diferença de separação dos *clusters*, ou seja, alta similaridade intra-*cluster* e baixa similaridade extra-*cluster*.

A redução de atributos realizada pelo RS fez com que informação considerada incerta não fosse apresentada à rede SOM, melhorando as fronteiras entre os *clusters*. Essa informação, quando submetida à rede SOM, gerava incerteza, ocasionando nos *clusters* uma definição de fronteira ruim, prejudicando a separação dos *clusters*.

Assim, com base nos experimentos realizados e nos resultados apresentados, pode-se concluir que a Arquitetura Híbrida teve desempenho superior ao da rede SOM sem redutos. Isso resultou em melhor taxa de acerto na classificação da base consumidor pela rede MLP.

Finalmente, com base nos resultados obtidos, pode-se considerar que a aplicação da Arquitetura Híbrida pode ser vantajosa em *marketing*: na detecção de perfil do consumidor, nas necessidades de consumo, na análise de clientes que podem abandonar, na fidelidade aos produtos ou serviços e nas tendências mercadológicas.

Os estudos aqui realizados não têm a pretensão de esgotar o assunto, pelo contrário, buscou-se realizar uma validação mais efetiva da Arquitetura Híbrida na descoberta de conhecimento em bases de dados. Sabe-se que existe uma clara demanda por estudos sistematizados que possam estabelecer outros domínios de aplicação ainda mais adequados para a Arquitetura Híbrida. Este cenário oferece, portanto, amplo espaço para trabalhos de continuidade.

Referências Bibliográficas

ADJEI, O.; CHEN, L.; HENG-DA, C.; COOLEY, D.H.; CHENG, R. J.; TWOMBLY, X. In: *A fuzzy search method for Rough Sets in Data mining*. IFSA World Congress and 20th NAFIPS International Conference, 2001.

AGRAWAL R.; IMIELINSKI T.; SWAMI A. *Mining association rules between Sets of itens in large databases*. In: Proceed, ACM SIGMOD Conference on Management of Data. Washington, DC, 2001.

ALMEIDA, F.C. *L'Evaluation des risques de défaillance des entreprises à partir des réseaux de neurones insérés dans les systémes d'aide à la décision.* Grenoble: Universidade de Grenoble, Ecole Supérieure des Affaires, 1993. (Tese de doutorado em Ciências de Administração).

ALMEIDA, F. C. *Desvendando o Uso de Redes Neuronais em Problemas de Administração de Empresas*. Revista de Administração de Empresas EASP/FGV, volume 35, nº 1, páginas: 46-55, jan./fev. 1995.

BAETS, W.,R.,J.; VENUGOPAL, V. *Neural Networks and Statistical Techniques in Marketing Research: A Conceptual Comparison. Marketing* .Intelligence & Planning, MCB University Press, v.12, número 7, pgs. 30-38, 1994.

BATISTA, A. P. A. E. G. *Pré-processamento de Dados em Aprendizado de Máquina Supervisionado*. Tese de Doutorado. Instituto de Ciências Matemáticas e de Computação, USP São Carlos, 2003.

BERRY, M. J. A.; Linoff, G.: *Data Mining Techniques: For Marketing, Sales and Customer Support*. John Wiley & Sons 1997.

BIGUS, J. P. *Data Mining with Neural Network: Solving Business Problems from Applications Development to Decision Support*. Mcgraw-Hill, 1996.

COSTA, J. A. F.; DE ANDRADE NETTO, M. L. In: *Clustering of complex shaped data sets via Kohonen maps and mathematical morphology*. Proceedings of the SPIE, Data Mining and Knowledge Discovery. 2001.

CURRY, B. In: *The Kohonen Self-organizing Map as an Alternative to Cluster Analysis: An Application to Direct Marketing*. In: *International Journal of Market Research*, June, 2003.

DAVIES, D. L.; BOULDIN, D. W. In: *A cluster separation measure*. IEEE Transactions on Pattern Analysis and Machine Intelligence, v. PAMI-1, 1979.

DOUGHERTY, J.; KOHAVI, R.; SAHAMI, M. *Supervised and Unsupervised Discretization of Continuos Features*. Proc. 12 th Int. Conf. Machine Learning, 194-202, 1995.

DUTRA, R.G. *Aplicação de Métodos de Inteligência Artificial em Inteligência de Negócios*. Dissertação de Mestrado. Escola Politécnica da Universidade de São Paulo. São Paulo, 2001.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMITH, P. In *The KDD process for extracting useful knowledge from volumes of data*. *Comunications of the ACM*, volume 39, 1996.

FERNANDEZ, G. *Data Mining using SAS Applications*. New York: Chapman & Hall/CRC, 2003.

GOLDSCHMIDT, R.; PASSOS, E. *Data Mining um guia prático. Conceitos, Técnicas, Ferramentas, Orientações e Aplicações*. Rio de Janeiro: Ed. Campus, 2005.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. New York: Willey & Sons, 1999.

HAN, J.; KAMBER, M. *Data Mining: Concepts and Techniques*. San Francisco: Morgan Kaufmann Publishers, 2001.

JIAJIN, H.; CHUNNIAN, L.; CHUANGXIN, O.; YAO, Y. Y.; ZHONG, N. In: *Attribute reduction of Rough Sets in Mining Market value functions***.** WI 2003. Proceedings. IEEE/WIC International Conference on Web Intelligence, Oct. 2003.

JIN, H.; SHUM, W. H.; LEUNG, K. S.; WONG, M. L. In: *Expanding self-organizing map for data visualization and cluster analysis. Information Sciences*, 163, 2004.

JOHNSON, R.A; WICHERN D.W. *Applied Multivariate Statistical Analysis.* Prentice Hall, 1998.

KASKI, S.; KOHONEN, T. In: *Exploratory Data Analysis by the Self-organizing Map: Structures of Welfare and Poverty in the World*. In: Proceedings of the third International Conference on Neural Networks in the Capital Markets. *World Scientific*, Singapore, 1996.

KASKI, S.; KOHONEM, T. In: *Winner-Takes-All Networks*. Triennial Report 1994 – 1996, Neural Networks Research Centre & Laboratory of Computer and Information Science, Helsinky University of Technology, Finland, (1997).

KIVILUOTO, K. In: *Topology Preservation in Self-Organizing Maps*. Technical Report, Helsinki University of Technology, Laboratory of Computer and Information Science, Finland, 1995.

KOHONEN, T. In: *Exploration of very large databases by self-organizing maps*. International Conference on Neural Networks, (1997), volume 1, 1997.

KOHONEN, T. *Self-Organizing Maps*. *Springer Series in Information Sciences*, Vol. 30, third edition. Springer, Berlin, Heidelberg, New York, 2001.

KOMOROWSKI, J.; ØHRN, A . In: ROSETTA: *A Rough Set Toolkit for Analysis of Data*. Proc. Third International Joint Conference on Information Sciences, Fifth International Workshop on *Rough Sets* and Soft Computing (RSSC'97), Durham, NC, USA, March 1-5, Vol. 3.

LEE, H. Y.; ONG; H. L. In: *Exploriting visualization in knowledge discovery*. Proc. 1st Int. Conf. Knowledge Discovery and Data Mining (KDD-95), AAAI, 1995.

LEE, D. H. *Seleção e Construção de Features Relevantes para o Aprendizado de Máquina*. Dissertação de Mestrado, Instituto de Ciências Matemáticas e da Computação. Universidade de São Paulo, São Carlos, 2001.

MITCHELL, T. *Machine Learning*. Mcgraw-Hill, 1997.

ØHM, A. In: *Discernibility and Rough Sets in Medicine: Tools and Applications*, PhD thesis, Department of Computer and Information Science, Norwegian University of Science and Technology (NTNU), Trondheim, Norway. NTNU report 1999.

PATRÍCIO, C. M. M. M., PINTO, J. O. P., SOUZA, C. C. In: *Rough Sets*: Técnica de Redução de Atributos e Geração de Regras para Classificação de Dados. XXVIII – CNMAC, Congresso Nacional de Matemática Aplicada e Computacional, 2005.

PAWLAK, Z. In: *Rough Sets*, *International Jornal of Computer and information Sciences*, 1982.

PAWLAK, Z. In: *Rough Sets*, Rough relations and Rough functions. *Fundamenta Informaticae*, 27, 1996.

PIATETSKY-SHAPIRO, G. In: *Knowledge Discovery in Real Databases.* A report on the IJCAI-89 Workshop. *AI Magazine*, volume 11, nº 5, jan.1991.

PIATETSKY-SHAPIRO, G.; MATHEUS, C. J.; CHAN, P. K. In: *Systems for Knowledge Discovery in Data bases*. IEEE,1993.

PYLE, D. *Data Preparation for Data Mining*. San Francisco: Morgan Kaufmann Publishers, 1999.

ROSETTA (*A Rough Sets Toolkit for Analisys of Data*) disponível em: http://www.idi.ntnu.no/~aleks/rosetta.

SAS (*Statistical Analisys System*) disponível em www.sas.com

SASSI, R.J., SILVA, L.A ., HERNANDEZ, E. del M. *An Hybrid Architecture for the Knowledge Discovery in Databases: Rough Sets Theory and artificial neural nets Self-Organizing Maps"*. International Conference on Information Systems and Technology Management. Congresso Internacional de Gestão de Tecnologia e Sistemas de Informação (4º CONTECSI). 31 a 1 de junho, 2007. São Paulo, Brasil.

SOM *Toobox* disponível em http://www.cis.hut.fi/projects/somtoolbox/documentation/start.shtml.

ULTSCH, A. In: *Knowledge extraction from self-organizing neural networks*. In: Opitz, O. ed. *Information and Classification*. Springer, 1993.

ULTSCH,A.; SIEMON, H.P. *In: Kohonen's Self Organizing Feature Maps for exploratory Data Analysis*. Proceedings of the International Neural Networks Conference. Dordrecht. Netherlands, 1989.

ULTSCH, A. In: *Self-Organizing Neural Networks perform different from statistical K-means Clustering*. In Proc. GfKl, Basel, Swiss, 1995.

VESANTO, J. In: *Data Mining Techniques Based on the Self-Organizing Maps.* MSc Thesis. Helsinki University of Technology, Espoo, Finland, 1997.

VESANTO, J.; HIMBERG, J.; ALHONIEMI, E.; PARHANKANGAS, J. In: SOM *Toolbox for Matlab*. Technical report A 57. Helsinki University of Technology, Finland, (2000).

VESANTO, J.; ALHONIEMI, E. In: *Clustering of the Self-organizing Map*. IEEE Transaction on Neural Network, vol.11, May, 2000.

ZHANG, X. *Time series analysis and prediction by neural networks*. Optimization Methods and Software, v.4, pgs 151 – 170, 1994.