

Adaptive Fuzzy Neural Tree Network

R. G. Dutra, M. Martucci Jr.

Resumo - O objetivo deste trabalho é combinar métodos de redes neurais artificiais, árvores de decisão adaptativas e sistemas de inferência baseado em lógica nebulosa (*fuzzy*) para prospecção de padrões de classificação de clientes, como parte de um processo interativo de descobrimento e análise de regularidades, regras e associações em uma base de dados proveniente de processos de negócios de vendas e distribuição.

Como resultado dessa combinação de métodos de tecnologia adaptativa, espera-se maximizar as vantagens dos diferentes métodos, em um modelo híbrido denominado de *Adaptive Fuzzy Neural Tree Network (AFNTN)*.

Palavras Chave - Sistemas Adaptativos, Árvores de Decisão, Redes Neurais Artificiais, Lógica Difusa.

I. INTRODUÇÃO

As corporações brasileiras sentiram nos últimos anos os sintomas de uma febre mundial que contaminou quase todas as empresas do globo, denominado de efeito ERP (*Enterprise Resource Planning*) ou simplesmente pacotes de gestão empresarial. Fenômeno típico da década de noventa, que sucedeu a redução (*downsizing*) dos sistemas de grande porte em plataforma *mainframe*, essas soluções resolvem apenas o dia a dia operacional das companhias, isto é, os dados transacionais, gerando continuamente enormes quantidades de informação em estado “bruto”. Com o propósito de garimpar e lapidar tais dados criou-se o conceito de Inteligência de Negócios [1].

Existem várias ferramentas para implementação do conceito de Inteligência de Negócios, variando desde planilhas eletrônicas até sofisticados sistemas de suporte à decisão baseados em Repositórios de Dados (*Data Warehouse*) com ferramentas analíticas de prospecção de dados (*Data Mining*), cuja complexidade depende fundamentalmente da aplicação.

Este trabalho objetiva a exploração das ferramentas de *Data Mining*, visando a aplicação em gerenciamento de relações de empresas com seus clientes, através da combinação de métodos de Inteligência Artificial, tais como *Redes Neurais Artificiais*, *Árvores de Decisão Adaptativas e Lógica Fuzzy* um modelo denominado de *Adaptive Fuzzy Neural Tree Network (AFNTN)*.

O problema alvo consiste em classificar clientes através dos dados reais provenientes da realização de processos de vendas e distribuição, de forma não supervisionada, ou seja, o número de classes e os atributos utilizados para definição das mesmas não é conhecido a priori, cabendo ao modelo *AFNTN* sua determinação.

Apesar do número de clientes que compõem a base de dados pertencerem a um conjunto aberto, neste artigo o mesmo será considerado como um conjunto finito e

completamente conhecido dentro de um intervalo de tempo determinado.

O conteúdo deste artigo está organizado nas seguintes seções: *II – Utilização de Dispositivos Adaptativos*, descreve o mecanismo adaptativo para indução de árvores de decisão e uma proposta para melhorá-lo. *III-Redes Neurais Artificiais*, descrevendo o algoritmo de aprendizado não supervisionado. *IV-Lógica Nebulosa* descreve como tratar incerteza nos dados. *V-O modelo da AFNTN* propõe a modelagem para implementação da ferramenta. *VI – Resultados Obtidos* e *VII – Conclusão* descrevem os resultados e conclusões obtidos através da aplicação da *AFNTN* em uma base de dados reais.

II. UTILIZAÇÃO DE DISPOSITIVOS ADAPTATIVOS

Bases de dados provenientes de processos empresariais podem atingir uma quantidade de registros elevadíssima, tornando impossível a realização de atividades de *data mining* por parte de um especialista humano. Face a esta necessidade, diferentes métodos foram criados para automatizar e sistematizar o processo de prospecção, análise de padrões e relacionamento dos mesmos com seus respectivos dados, denominados de *machine learning (ML)* [2].

Dentre os diversos métodos de ML, tais como: métodos baseados em regras, método do discriminante de Bayes entre outros; escolheu-se neste artigo, trabalhar com métodos partitivos recursivos denominados de árvores de decisão ou *Decision Trees (DT)*, devido às características citadas a seguir:

- Permitem a redução do volume de dados através da transformação para um formato mais compacto, porém sem perder as principais características e relacionamento dos mesmos.
- Permitem descobrir se os conjuntos de dados contém agrupamentos de objetos, que podem ser úteis para simplificações e particionamento dos mesmos.
- Permitem mapear o relacionamento entre variáveis independentes e dependentes, objetivando a construção de um modelo classificatório preditivo.

Segundo [3], o problema de indução incremental de árvores de decisão, para classificação de atributos discretos, pode ser resolvido através da aplicação da tecnologia adaptativa utilizando um dispositivo adaptativo, com descrito no algoritmo *AdapTree*.

Um dispositivo adaptativo [4] é constituído de duas partes: a primeira é usualmente algum dispositivo usual, definido através de regras (em particular, algum autômato, gramática ou qualquer outro dispositivo descrito através de um conjunto finito de regras estáticas), denominado seu dispositivo subjacente (tipicamente não-adaptativo); a segunda é um

mecanismo adaptativo, cuja conexão ao formalismo subjacente proporciona todos os recursos complementares necessários para a realização das propriedades responsáveis pela auto-modificação autônoma que caracteriza os dispositivos adaptativos.

Um autômato adaptativo [5] de estados finitos (AAF) é um dispositivo adaptativo, que estende o poder de expressão do autômato de estados finitos (AF), através da capacidade de modificar a sua própria estrutura com a aplicação de regras adaptativas citadas anteriormente.

O dispositivo adaptativo no qual o *AdapTree* se baseia pode ser visto como um AAF classificador, estendido para trabalhar com mais de duas classes. Para isso, a cada estado final é associado um elemento, que corresponderá a uma das classes possíveis. Ao receber um exemplo de treinamento o *AdapTree* cria um caminho ligando o estado inicial do autômato ao estado final correspondente a classificação deste exemplo.

Caso não seja possível determinar, sintaticamente, a classe de uma cadeia de entrada, o *AdapTree* utiliza o mecanismo estatístico ID3 [6] que oferecerá como resposta uma estimativa baseado no ganho de entropia, que define a ordem dos atributos utilizada neste cassificação

Porém percebeu-se [7] que, um atributo com muitos valores possíveis, teria uma dispersão maior na distribuição de probabilidades desses valores, conseqüentemente maior ganho de informação. Para evitar esta distorção, que favorece o atributo com maior número de valores, introduziu-se no algoritmo C4.5[8], sucessor do ID3, o conceito de Razão do Ganho de Informação.

O método C4.5 tem sido largamente empregado para construir DT que implementam classificadores de elevada performance. Contudo, este algoritmo só permite classes previamente definidas para classificação de atributos na fase de treinamento, não tendo a capacidade de interpolar ou deduzir novos padrões por inferência nesta fase ou tratar dados imprecisos ou incertos. Como demonstrado em [9] e [10], o C4.5 só pode delimitar hiperplanos paralelos aos eixos coordenados, fato que, em um espaço de atributos contínuos, implicaria em crescimento exponencial da DT resultante.

Diferentemente do C4.5, o *AdapTree* não particiona o conjunto de exemplos, ou seja, o princípio de otimização é global e não local. A árvores de decisão geradas pelo *AdapTree* não possuem um número excessivo de ramificações, devido a esta característica.

Adicionalmente, como o *AdapTree* é um algoritmo incremental, novos exemplos podem ser incorporados durante o processo de classificação, resolvendo adequadamente o problema de variação temporal de padrões, muito comum em atividades de *Data Mining*. A combinação do *AdapTree* e do mecanismo C4.5, permite maximar a vantagem de ambos algoritmos, minimizando as desvantagens citadas.

No entanto, ambos *AdapTree* e C4.5 necessitam de um conjunto de treinamento para geração de uma DT cujo número de nós entre a raiz e folhas não seja excessivamente grande. Dessa forma, o *AdapTree* e o C4.5 são algoritmos que trabalham em modo supervisionado durante a fase de

treinamento, ou seja, não sendo capazes de definir o número de classes que particionam o conjunto de treinamento.

Por outro lado, sabe-se que Redes Neurais Artificiais (RNA) tem sido empregadas em tarefas de classificação para determinação de padrões em modo supervisionado ou não supervisionado. Classes complexas podem ser prospectadas através dos dados a fim de gerar uma DT de forma mais rápida, eficiente e de simples visualização.

Evidentemente que a performance e qualidade dos padrões adquiridos estão diretamente correlacionados com o tipo e a arquitetura escolhida para a RNA, como demonstram os estudos realizados comparando-se a arquitetura de *Multi-Layered Perceptron* (MLP) com DT [7].

A seguir, será apresentado um algoritmo de RNA adaptativo, porém não baseado em Automatos Adaptativos Finitos, cuja vantagem é não necessitar de um conjunto de treinamento classificado a priori, como o *AdapTree*.

III. REDES NEURAIAS ARTIFICIAIS

O modelo *Self-Organizing Map* (SOM) desenvolvido por Teuvo Kohonen [11] é um dos modelos mais populares de RNA. O algoritmo da SOM é baseado em um aprendizado competitivo e não supervisionado, o que implica em um treinamento direcionado exclusivamente pelos dados, sendo que os neurônios que constituem o mapa competem entre si para adquirir padrões dos dados, se aproximando deles. Algoritmos supervisionados, como o *Multi-Layered Perceptron* (MLP), requerem uma classificação pré-definida para cada vetor de treinamento, além de depender fundamentalmente do número de camadas internas (*hidden units*) para um aprendizado com baixo erro de classificação e performance, limitações as quais não ocorrem na arquitetura SOM.

Entretanto, a presença de imprecisão, incerteza ou ruído nos dados pode reduzir drasticamente a performance do aprendizado competitivo do algoritmo SOM, da mesma forma que induz árvores excessivamente grandes utilizando os algoritmos *Adaptree* e C4.5. Para tratar de forma eficaz a questão da incerteza, torna-se necessária a utilização de um mecanismo de inferência baseado em lógica nebulosa.

IV LÓGICA NEBULOSA

A lógica nebulosa (*fuzzy*) possibilita que seja abordado de forma mais adequada um problema muito importante referente à representação e manipulação de conhecimentos em ML para representação da imprecisão e da incerteza.

Os sistemas baseados em lógica nebulosa foram criados por [12], fundamentando-se na representação e manipulação de informações incertas e imprecisas tão comuns no cotidiano humano. Expressões tais como “quase”, “muito” e “pouco” representam este tipo de imprecisão, que usualmente não pode ser tratada pelos sistemas da lógica clássica.

Os sistemas especialistas *fuzzy* utilizam um conjunto de regras do tipo “*If-Then*”, baseadas em variáveis nebulosas. Primeiramente as variáveis de entrada sofrem um processo de “*fuzzificação*”, ou seja, os conjuntos nebulosos das variáveis lingüísticas de entrada são ativados. Terminado este processo

efetua-se a inferência sobre o conjunto de regras nebulosas obtendo os valores dos termos das variáveis de saída. Finalmente, as variáveis de saída sofrem um processo de “defuzzificação”.

Este processo consiste em converter os dados nebulosos para valores numéricos precisos. Para isto são utilizadas várias técnicas, tais como valor máximo, média dos máximos, média local dos máximos, centro de gravidade, ponto central da área e o centro médio. Neste artigo, foi utilizado o método Takagi-Sugeno [13], ou simplesmente Sugeno, cuja saída representará um valor constante, representando uma classe definida pela RNA do tipo SOM.

V. O MODELO DA ADAPTIVE FUZZY NEURAL TREE NETWORK

Existem inúmeras formas de se combinar DT Adaptativa, RNA do tipo SOM e Lógica Fuzzy para formar o modelo da *Adaptive Fuzzy Neural Tree Network*, porém a arquitetura escolhida neste trabalho objetiva suportar as atividades recursivas de *Data Mining*, como é ilustrado na figura 5.1, a seguir:

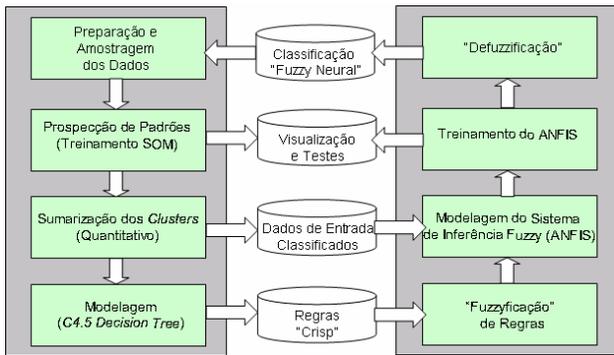


Fig. 5.1. Diagrama de Blocos da *Adaptive Fuzzy Neural Tree Network (AFNTN)*

A base de dados utilizada é composta de características inerentes ao processo de vendas e distribuição e indicadores (variáveis de conteúdo numérico) conforme ilustrado na tabela 5.1. A partir da tabela 5.1 construiu-se dois conjuntos de dados para treinamento do algoritmo da SOM, denominados de *Treinamento_Parcial.data* e *Treinamento_Completo.data*. O conjunto *Treinamento_Parcial.data* não contém as variáveis **Num_Rem** e **Prc_Med** para verificar o efeito do acréscimo, ou seja, a escalabilidade, de variáveis sobre o número de classes obtidas a partir do treinamento da SOM.

Indicadores						Classificação
Volume de Vendas (Vol_Ven)	Quantidade de Vendas (Qtd_Ven)	Volume Devoluções (Vol_Dev)	Quantidade Devoluções (Qtd_Dev)	Preço Médio (Prc_Med)	Número de Remessas (Num_Rem)	A, B ou C (ABC)

Tabela 5.1 – Estrutura Completa da Base de Dados para treinamento da SOM

A classificação a priori ABC, que foi obtida através da acumulação da variável Volume de Vendas, foi utilizada para fins de comparação da classificação resultante da RNA SOM, conforme figura 5.2 e tabela 5.2.

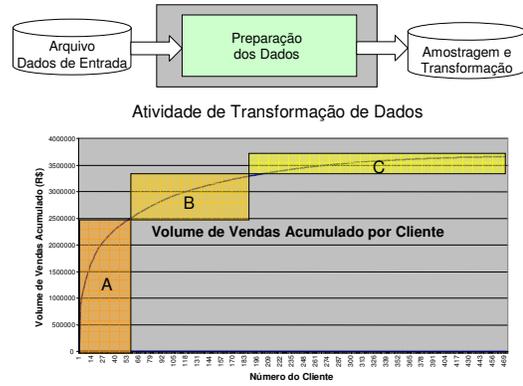


Fig. 5.2 – Classificação a priori ABC

Resultados da Classificação ABC, definindo a priori, os percentuais de separação entre classes em 70%, 20% e 10%:

Classificação	Clientes		Volume de Vendas Acumulado (%)
	Número	Percentual (%)	
Segmento A	62	13,22	70,03
Segmento B	124	26,44	20,00
Segmento C	283	60,34	9,97
Total	469	100	100

Tabela 5.2 – Segmentação da Base de Dados segundo classificação ABC

As próximas etapas resumem-se em preparar a base de dados para treinamento da rede neural artificial utilizando o algoritmo SOM e obter padrões qualitativos, através da visualização do formato e estrutura dos agrupamentos, mapa de componentes e projeções dos dados sobre o mapa, conforme ilustrado na figura 5.3.

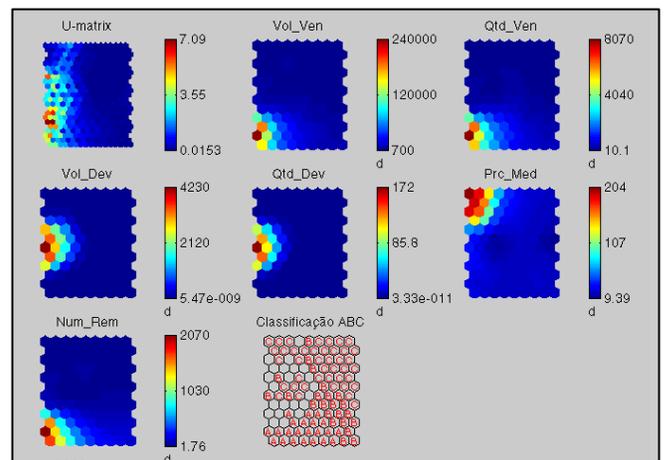


Fig. 5.3 – *U-Matrix* e mapas componentes após treinamento da RNA do tipo SOM

Após o treinamento da rede neural, realizou-se a quantificação do número de agrupamentos através da utilização do índice de *Davies-Boulding* [14]. Aplicando-se este algoritmo à base de dados *Treinamento_Completo.data*, o

índice de *Davies-Boulding* atingiu o valor mínimo para 4 classes, conforme figura 5.4. A cada amostra da base de dados foi associada uma das 4 classes, formando um novo conjunto de dados classificados.

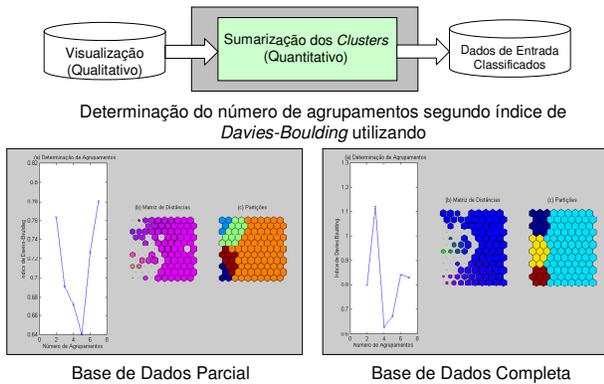


Fig. 5.4 . Sumarização Quantitativa dos Clusters encontrados

Todas as funções utilizadas no treinamento do algoritmo da SOM e resultados obtidos, basearam-se em funções previamente elaboradas em Matlab® versão 6.5 provenientes da SOMTOOLBOX 2.0 [15].

A partir desse novo conjunto foram efetuadas diferentes tentativas para a determinação de uma árvore de decisão que minimizasse o erro de classificação. A técnica de validação cruzada dos dados de treinamento e testes foi utilizada para obter tal árvore.

Os dados classificados foram utilizados para treinamento da árvore de decisão, conforme figuras 5.5 e 5.6 a seguir:

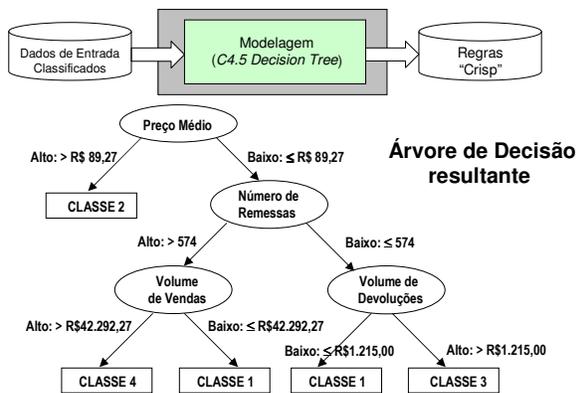


Fig. 5.5. Modelagem da árvore de decisão

A indução dos nós da árvore de decisão foi realizada até minimizar a razão do ganho de entropia $Info(X,T)$, onde X representa uma amostra e T o número total de amostras.

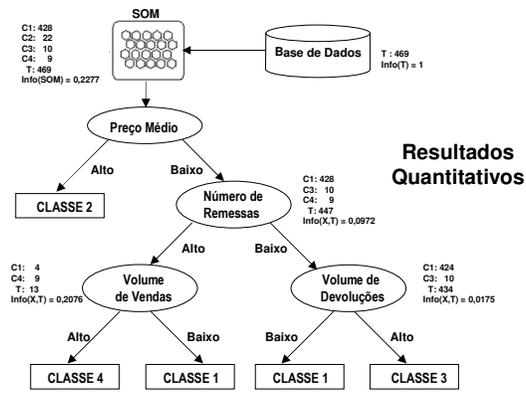


Fig. 5.6. Modelagem da árvore de decisão após treinamento

Como a árvore de decisão gera regras rígidas (crisp) ou booleanas (Alto ou Baixo), foi necessário “fuzzificar” as regras, como mostrado na figura abaixo:

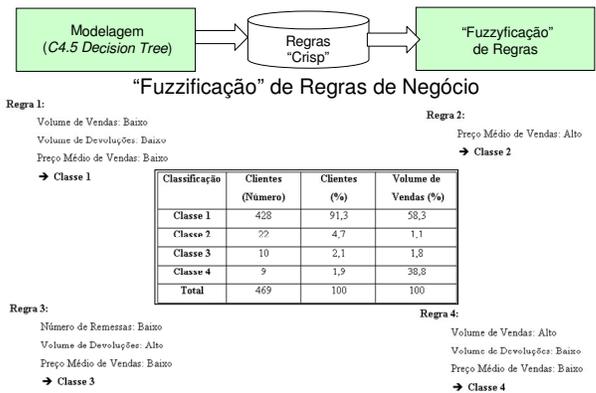


Fig. 5.6 – “Fuzzificação” das regras de negócio

Os dados classificados e as regras fuzzy foram utilizadas como entrada do sistema de inferência fuzzy do tipo Sugeno , fornecido pela FUZZYTOOLBOX [16] do Matlab®:

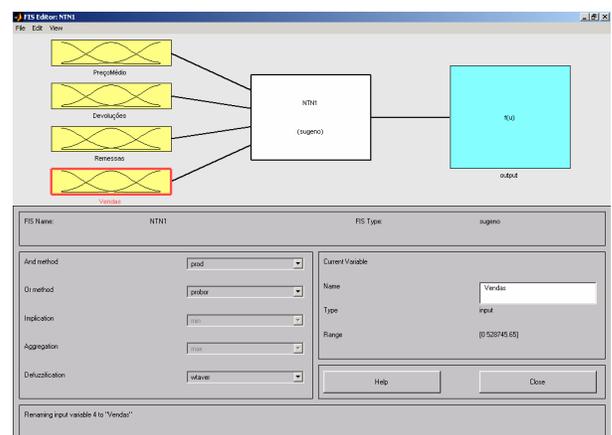


Fig. 5.7 – Modelagem do ANFIS

A funções de pertinência escolhidas foram do tipo gaussiana ou normal, conforme figura abaixo:

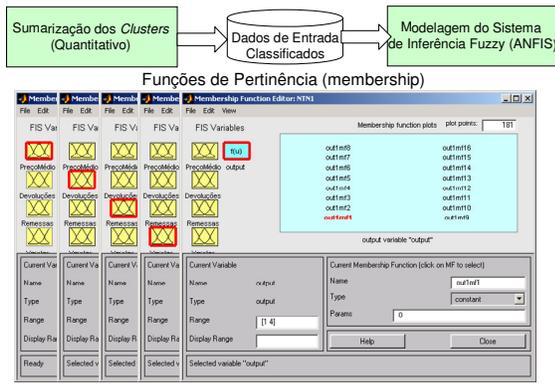


Fig. 5.8 – Funções de Pertinência

A figura a seguir ilustra a variável Preço Médio, cuja função de pertinência gaussiana, segundo uma distribuição normal, será ajustada após o treinamento do sistema de inferência fuzzy.

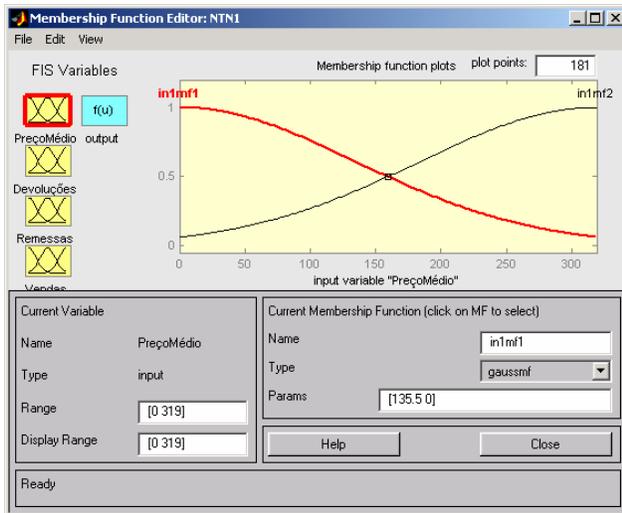


Fig. 5.9 – Função de Pertinência Gaussiana

O sistema de inferência fuzzy adaptativo do tipo Sugeno (ANFIS) [16] utiliza uma rede neural do tipo *Multi-Layer Perceptron* (MLP) para ajuste das funções de pertinência.

A combinação das variáveis de entrada com as possíveis classes de saída é realizado através da criação de regras fuzzy do tipo “AND”:

SE Variável_Entrada_1 pertence a Função_Pertinência_1 *E*
 Variável_Entrada_2 pertence a Função_Pertinência_1 *E*
 ...
 Variável_Entrada_n pertence a Função_Pertinência_1
 ENTÃO
 Saída PERTENCE A Função_Pertinência_Classe_1

A figura 5.10 ilustra a formação das regras fuzzy do tipo “AND” para a árvore de decisão obtida através do algoritmo *AdapTree* modificado.

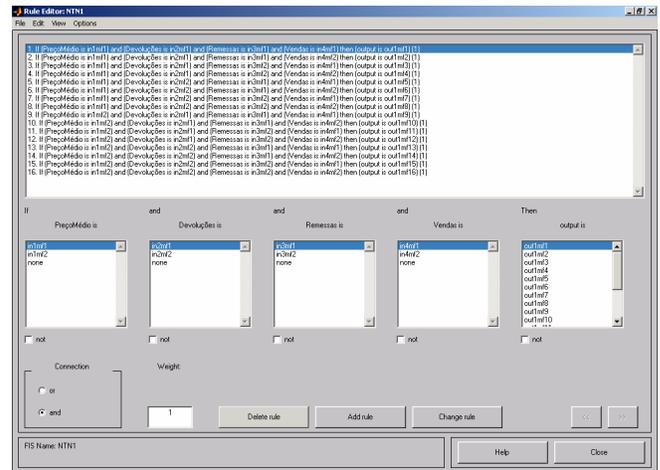


Fig. 5.10 – Definição de regras fuzzy do tipo “AND”

Para o treinamento, foram executados 500 ciclos de treinamento do ANFIS, conforme figura a seguir:



Fig. 5.11 – Carga de dados de treinamento do ANFIS

Aumentando o número de ciclos de treinamento de 500 para 1000, o erro de convergência reduziu apenas 5%, ou seja, um ganho absoluto muito baixo, para justificar um aumento ainda maior no número de ciclos de treinamento.

VI RESULTADOS OBTIDOS

O treinamento resultou em um ajuste das funções de pertinência, conforme figura 6.1.

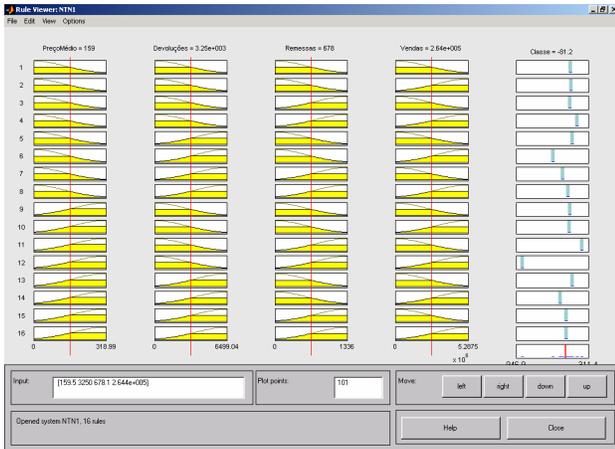


Fig 6.1. Superfície de decisão

As funções de pertinência delimitam superfícies de decisão, como no exemplo da figura 6.2, onde as variáveis de entrada Devoluções e Preço Médio delimitam uma superfície tridimensional que determina a função de pertinência da classificação determinada pela RNA do tipo SOM.

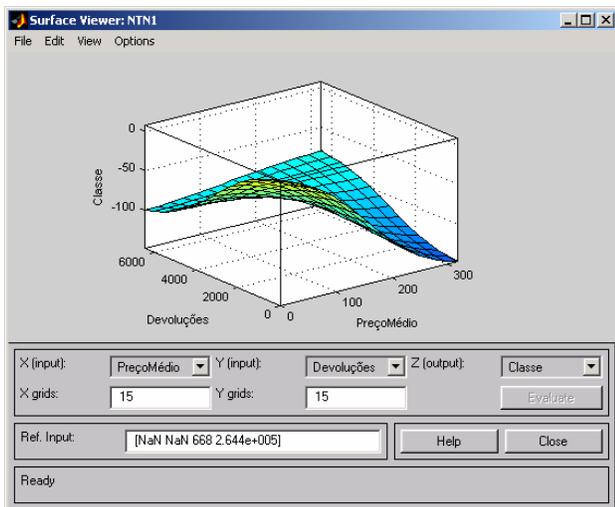


Fig 6.2. Superfície de decisão formada pelas variáveis de entrada Devoluções e Preço Médio

Uma vez treinado o sistema ANFIS, foram selecionados aleatoriamente 100 registros da base de dados para testes e 169 para checagem, conforme figuras a seguir:

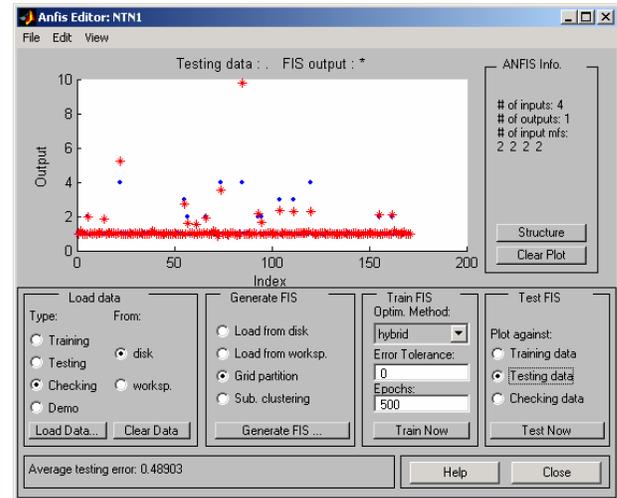


Fig 6.2 – Comparação dos dados utilizados no teste do ANFIS com a classificação conhecida a priori, resultando da árvore de decisão e SOM.

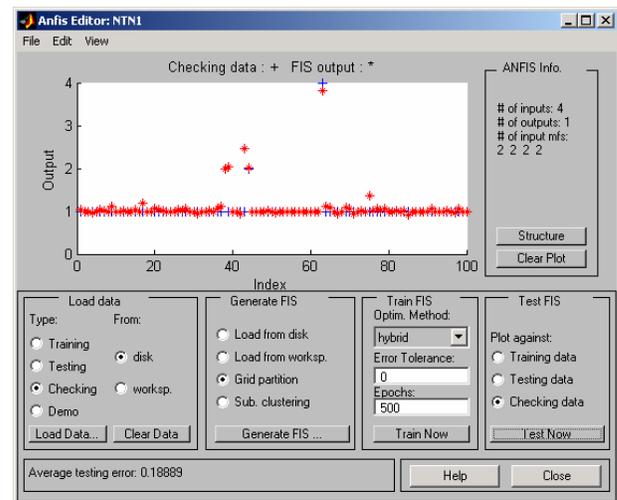


Fig 6.3 – Comparação dos dados utilizados no teste do ANFIS com a classificação conhecida a priori, resultando da árvore de decisão e SOM.

A partir da árvore de decisão ilustrada na figura 6.1 e de uma estatística efetuada sobre o número de clientes em cada classificação como mostrado na tabela 6.1, um especialista em negócios pode inferir um significado para cada classe dentro do contexto do processo de negócios de vendas e distribuição, a seguir.

Classificação	Clientes (Número)	Clientes (%)	Volume de Vendas (%)
Classe 1	428	91,3	58,3
Classe 2	22	4,7	1,1
Classe 3	10	2,1	1,8
Classe 4	9	1,9	38,8
Total	469	100	100

Tabela 6.1 – Estatística do número de clientes e percentual de faturamento por classe

a) *Classe 1*

Representa clientes regulares, isto é, os clientes que compram produtos de baixo preço médio, em pequenas quantidades e normalmente realizam poucas devoluções. Estes clientes regulares representaram 91,3% em número, porém respondem por 58,3% do faturamento em vendas.

b) *Classe 2*

Representa clientes que compram produtos de alto valor agregado, isto é, produtos de elevado preço médio em baixas quantidades. Estes clientes, apesar de representarem 4,7% em número, respondem por somente 1,1% do faturamento em vendas.

c) *Classe 3*

Representa clientes com elevada ocorrência de devoluções. Apesar do baixo percentual em número, ou seja 2,1%, representaram um faturamento em vendas de 1,8%.

d) *Classe 4*

Representam clientes ótimos, isto é, clientes que compram produtos de baixo preço médio, porém em quantidades elevadas. Respondem por um faturamento de vendas de 38,8%, apesar de representarem somente 1,9% em número.

Baseado no significado de cada classe, um especialista de negócios utilizando o modelo da *AFNTN* como ferramenta para estratificação de clientes dentro do conceito de gerenciamento de relações com clientes, chegaria aos seguintes resultados finais:

Os clientes segmentados através da *Classe 4* deverão receber tratamento personalizado. Todos os esforços de interação com os mesmos deverão buscar o melhor atendimento em função de suas necessidades e o aumento de satisfação.

Os clientes segmentados através da *Classe 1* devem continuar fazendo parte do processo de prospecção e análise através da *AFNTN*, aguardando uma possível modificação para a *Classe 4*, porém sem aumento de esforço operacional para a empresa para que a mudança ocorra.

Os clientes segmentados através da *Classe 2* representam clientes potenciais cujo aumento da quantidade de vendas poderá promovê-los à *Classe 4* com menor esforço operacional que os clientes da *Classe 1* e portanto justificam tratamento personalizado.

Os clientes segmentados através da *Classe 3* deverão ter seu processo de vendas e distribuição revisto para que os possíveis problemas por parte da empresa sejam resolvidos resultando em uma mudança de classificação para a *Classe 1*. Caso contrário deverão ser eliminados do cadastro de clientes.

VII. CONCLUSÃO

Segundo a teoria da informação [7], utiliza-se do termo entropia para quantificar o conteúdo de informação de um determinado agrupamento. Um valor de entropia elevado implica em conteúdo baixo de informação, ou seja, mais informação será necessária para identificar que uma determinada amostra de dados pertence a um determinado agrupamento ou classificação.

Baseado nessa afirmação, o modelo da *AFNTN* foi capaz de reduzir a entropia da base de dados inicial do valor máximo até seu valor mínimo com um grau de precisão médio, como será descrito a seguir. Observando-se cada etapa do modelo, a etapa de Preparação de dados não reduz a entropia do sistema, ao passo que a *SOM* foi responsável pela maior redução, isto é, um ganho de aproximadamente 72%, cabendo os 28% restantes à árvore de decisão. O *ANFIS* tem por função tratar dados incertos ou com ruído e não redução de entropia do sistema.

A tabela a seguir sintetiza as principais características avaliadas nos modelos *Adaptive Fuzzy Neural Tree Network (AFNTN)*, *Self-Organizing Map (SOM)*, *Árvore de Decisão (DT)* Adaptativa e *Sistema de Inferência Fuzzy*, através da qual conclui-se que a *AFNTN* maximizou as vantagens de ambas as técnicas.

Característica	<i>SOM</i>	<i>Decision Tree</i>	<i>Fuzzy Logic</i>	<i>Desejável</i>	<i>AFNTN</i>
Precisão	Média	Alta	Média	Alta	Média
Inteligibilidade	Média	Alta	Alta	Alta	Alta
Tempo de Resposta	Médio	Baixo	Médio	Baixo	Médio
Escalabilidade	Alta	Alta	Médio	Alta	Alta
Tolerância a ruído nos dados	Média	Baixa	Alta	Alta	Alta
Tolerância a dados esparsos	Alta	Baixa	Alta	Alta	Alta
Curva de Aprendizado	Média	Baixa	Baixa	Baixa	Baixa
Independência de Especialistas	Alta	Alta	Média	Alta	Alta

Tabela 7.1 – Síntese de características do modelo *AFNTN* em comparação com as técnicas adaptativas utilizadas

Nesta tabela, a definição das características da RNA do tipo *SOM* foram obtidas através de [17] e da *Decision Tree* através de [2].

Através da tabela 7.1, conclui-se que o modelo proposto da *AFNTN* contribuiu para aumentar a inteligibilidade da RNA do tipo *SOM*, aumentar a escalabilidade do modelo de *Fuzzy Logic*, aumentar a tolerância a ruído da RNA do tipo *SOM* e *Decision Tree*, aumentar a tolerância a dados esparsos da *Decision Tree*, reduzir a curva de aprendizado da RNA do tipo *SOM*, reduzir a independência de especialistas do modelo de *Fuzzy Logic*.

As limitações do modelo da *AFNTN* ocorreram em características pouco tangíveis, tais como Curva de Aprendizado e Independência de Especialistas, que dependem fundamentalmente da experiência e familiaridade do especialista em negócios com o problema e ferramenta em questão. As características *Precisão* e *Tempo de Resposta*, relacionadas à qualidade e construção do modelo, foram avaliadas de forma conservadora devido a falta de padrões para uma avaliação mais quantitativa e menos qualitativa.

Com relação aos resultados obtidos em relação ao significado das classes de cliente prospectados da *Adaptive Fuzzy Neural Tree Network* como ferramenta de inteligência de negócios, deverão ser compatíveis com o planejamento estratégico de médio e longo prazo da empresa para manter ou aumentar sua participação de mercado para os produtos rentáveis e diminuir sua participação nos produtos pouco rentáveis.

SUGESTÕES PARA MELHORIAS FUTURAS

Baseado nos resultados obtidos e conclusões realizadas nas seções anteriores sugerem-se as seguintes melhorias futuras:

- Definição de um método de avaliação quantitativa da qualidade do mapa resultante após treinamento da SOM, para avaliar quantitativamente a precisão da SOM e conseqüentemente a precisão da *AFNTN*;
- Introduzir a capacidade de modelar e classificar variáveis alfanuméricas no algoritmo de treinamento da SOM;
- Testar e avaliar outros métodos de agrupamento e sumarização além do algoritmo *k-means* e o índice de *Davies-Boulding*.
- Testar e avaliar o efeito da variação temporal sobre os dados, afetando a captura de padrões variantes no tempo sobre o *AdapTree*.
- Testar a performance da *AFNTN* com um grande volume de dados.
- Comparar e avaliar as vantagens do modelo proposto para a *AFNTN* com outras ferramentas de classificação estatísticas, como determinante de Bayes.
- Avaliar a utilização da *AFNTN* em sistemas distribuídos e telecomunicações.
- Avaliar a utilização da *AFNTN* para prospecção de atributos funcionais e não funcionais de serviços web.

- [1] DUTRA, R.; CABRAL E. Aplicação de métodos de inteligência artificial em inteligência de negócios. Dissertação de mestrado, Poli, USP, 2001.
- [2] DHAR, V.; STEIN, R. Seven Methods for transforming corporate data into business intelligence. Prentice-Hall Press. 1997.
- [3] PISTORI, H. e NETO, J.J. AdapTree - Proposta de um Algoritmo para Indução de Árvores de Decisão Baseado em Técnicas Adaptativas. Anais Conferência Latino Americana de Informática - CLEI 2002. Montevideo, Uruguai, Novembro, 2002
- [4] PISTORI, H.; NETO, J. J.; PEREIRA, M.C.; Tecnologia Adaptativa em Engenharia da Computação. Estado da Arte e aplicações. Edição Revisada, 174p. São Paulo, 2003.
- [5] PISTORI, H.; NETO, J. J.; PEREIRA, M. C. Adaptive Non-Deterministic Decision Trees: General Formulation and Case Study. INFOCOMP Journal of Computer Science, Lavras, MG, 2006 (accepted).
- [6] CHENG, J.; FAYYAD, U. M.; IRANI, K. B.; QIAN, Z.; Improved decision trees: A generalized version of ID3. Proceedings of the Fifth International Conference on Machine Learning (pp. 100-106). Ann Arbor, MI: Morgan Kaufman, 1988.
- [7] QUINLAN, J. R.; Comparing Connectionist and Symbolic Learning Methods. Basser Department of Computer Science; University of Sydney; Sydney NSW 2006; Australia. 1990.
- [8] QUINLAN, J. R. C4.5 Programs for Machine Learning. Morgan Kaufmann. 1992.
- [9] BRODLEY, C. E.; UTGOFF, P. E. Multivariate versus univariate decision trees. Technical report, Department of Computer Sciences University of Massachusetts. 1992.
- [10] MITCHELL, T. Machine Learning. McGraw Hill. 1997.
- [11] KOHONEN, T. Self-Organizing Maps. Springer-Verlag. 1995.
- [12] ZADEH, L.A., "Fuzzy sets," Information and Control, Vol. 8, pp. 338-353, 1965.
- [12] SUGENO, M., Industrial applications of fuzzy control, Elsevier Science Pub. Co., 1985.
- [14] DAVIES, D. L.; BOULDING, D.W. A Cluster Separation Measure. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-1, no. 2, pp. 224-277. 1979.
- [15] VESANTO, J.; ALHONIEMI, E.; HIMBERG, J.; PARHANKANGAS, J. Som Toolbox 2.0 BETA online documentation. Internet address [http:// www.cis.hut.fi/projects/somtoolbox](http://www.cis.hut.fi/projects/somtoolbox). 1999.
- [16] JANG, J.-S. R., ANFIS: Adaptive-Network-based Fuzzy Inference Systems, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 3, pp. 665-685, May 1993.
- [17] VESANTO, J. Using SOM in Data Mining. Thesis for the degree of Licentiate of Science in Technology. Helsinki University of Technology. Finland.2000.

Dutra, R., Engenharia Eletrônica pelo Instituto Tecnológico de Aeronáutica (1995), mestrado em Engenharia Elétrica pela Universidade de São Paulo (2001) e atualmente cursando doutorado na Politécnica da Universidade de São Paulo. Atualmente é gerente de projetos SAP, com ênfase em projetos de implementação de soluções de sistemas integrados e Inteligência de Negócios. Atuando principalmente nos seguintes temas: Sistemas de Automação, Sistemas distribuídos, Sistemas Abertos, Arquitetura Orientada a Serviços, Arquitetura Distribuída.

Martucci Jr., M. Engenharia Elétrica pela Universidade de São Paulo (1973), graduação em Bacharelado Em Física pela Universidade de São Paulo (1975), mestrado em Engenharia Elétrica pela Universidade de São Paulo (1977) e doutorado em Engenharia Elétrica pela Universidade de São Paulo (1982) . Atualmente é professor titular da Universidade de São Paulo. Tem experiência na área de Engenharia Elétrica , com ênfase em Eletrônica Industrial, Sistemas e Controles Eletrônicos. Atuando principalmente nos seguintes temas: Sistemas de Automação, Sistemas distribuídos, Sistemas Abertos, Arquitetura Hierarquizada, Arquitetura Distribuída.

RESPOSTAS EM NEGRITO ÀS REVISÕES PROPOSTAS

reviewer #1

O formato do artigo não está totalmente de acordo com o padrão IEEE: nomes, filiação, e-mail, etc; falta do rodapé padronizado

Solicitamos aos autores que o adequem para que siga o modelo padrão usado pela revista.

O padrão foi seguido conforme instruções do arquivo Modelo_Template.doc (http://www.ewh.ieee.org/reg/9/etrans/files/autor_revista_latim_america_r4.zip) para IEEE América Latina. Não existe rodapé padronizado neste template e a filiação está definida conforme este padrão.

A tabela do pé da página 6 tem seu título na página 7.

OK. Resolvido.

Restam muitos erros de linguagem no texto, que devem ser eliminados por uma revisão cuidadosa. **OK. Resolvido todos os erros, conforme informações do revisor #2.**

O artigo apresenta-se como uma aplicação prática da adaptatividade, como um interessante exemplo de uso dos conceitos utilizados.

Faltou, no entanto, um pouco mais de fundamentação teórica (ao menos da indicação das fontes onde se podem encontrar os fundamentos matemáticos necessários), que poderia melhorar a sustentação aos argumentos apresentados na utilização dos conceitos.

As fundamentações teóricas e matemáticas de cada modelo encontram-se nas seguintes referências:

Redes Neurais do tipo SOM:

- [1] DUTRA, R.; CABRAL E. Aplicação de métodos de inteligência artificial em inteligência de negócios. Dissertação de mestrado, Poli, USP, 2001.
- [17] VESANTO, J. Using SOM in Data Mining. Thesis for the degree of Licentiate of Science in Technology. Helsinki University of Technology. Finland.2000.
- [11] KOHONEN, T. Self-Organizing Maps. Springer-Verlag, 1995.
- [14] DAVIES, D. L.; BOULDING, D.W. A Cluster Separation Measure. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. PAMI-1, no. 2, pp. 224-277. 1979.
- [15] VESANTO, J.; ALHONIEMI, E.; HIMBERG, J.; PARHANKANGAS, J. Som Toolbox 2.0 BETA online documentation. Internet address [http:// www.cis.hut.fi/projects/somtoolbox](http://www.cis.hut.fi/projects/somtoolbox). 1999.

- [16] JANG, J.-S. R., ANFIS: Adaptive-Network-based Fuzzy Inference Systems, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 23, No. 3, pp. 665-685, May 1993.
- [17] VESANTO, J. Using SOM in Data Mining. Thesis for the degree of Licentiate of Science in Technology. Helsinki University of Technology. Finland.2000.

Machine Learning (ML) e Árvores de Decisão:

- [6] CHENG, J.; FAYYAD, U. M.; IRANI, K. B.; QIAN, Z.; Improved decision trees: A generalized version of ID3. Proceedings of the Fifth International Conference on Machine Learning (pp. 100-106). Ann Arbor, MI: Morgan Kaufman, 1988.
- [7] QUINLAN, J. R.; Comparing Connectionist and Symbolic Learning Methods. Basser Department of Computer Science; University of Sydney; Sydney NSW 2006;Australia. 1990.
- [8] QUINLAN, J. R. C4.5 Programs for Machine Learning. Morgan Kaufmann. 1992.
- [9] BRODLEY, C. E.; UTOFF, P. E. Multivariate versus univariate decision trees. Technical report, Department of Computer Sciences University of Massachusetts. 1992.
- [10] MITCHELL, T. Machine Learning. McGraw Hill. 1997.

Sistema adaptativos

- [3] PISTORI, H. e NETO, J.J. AdapTree - Proposta de um Algoritmo para Indução de Árvores de Decisão Baseado em Técnicas Adaptativas. Anais Conferência Latino Americana de Informática - CLEI 2002. Montevideo, Uruguai, Novembro, 2002
- [4] PISTORI, H.; NETO, J. J.; PEREIRA, M.C.;Tecnologia Adaptativa em Engenharia da Computação. Estado da Arte e aplicações. Edição Revisada, 174p. São Paulo, 2003.
- [5] PISTORI, H.; NETO, J. J.; PEREIRA, M. C. Adaptive Non-Deterministic Decision Trees: General Formulation and Case Study. INFOCOMP Journal of Computer Science, Lavras, MG, 2006 (accepted).

Lógica Fuzzy

- [12] ZADEH, L.A., "Fuzzy sets," Information and Control, Vol. 8, pp. 338-353, 1965.
- [12] SUGENO, M., Industrial applications of fuzzy control, Elsevier Science Pub. Co., 1985.

Sugerimos que seja adicionada nas conclusões uma informação sobre a contribuição que este artigo representa em cada uma das áreas a que se refere.

Foi incluído o seguinte parágrafo no item de conclusão:

Através da tabela 7.1, concluí-se que concluí-se que o modelo proposto da *AFNTN* contribuiu para aumentar a inteligibilidade da RNA do tipo SOM, aumentar a escalabilidade do modelo de *Fuzzy Logic*, aumentar a tolerância a ruído da RNA do tipo SOM e Decision Tree, aumentar a tolerância a dados esparsos da Decision Tree, reduzir a curva de aprendizado da RNA do tipo SOM, reduzir a independência de especialistas do modelo de *Fuzzy Logic*.

reviewer #2

Legenda: P1.C1=página 1, coluna 1.

- 1. P1.C1 (resumo): retirar o ponto duplo no final do resumo. **OK.**
- 2. P1.C1 (final): reescrever o

trecho "Apesar o número..." **OK**.

3. P1.C2 (início): corrigir no

"melhora**rá**-lo" **ok**.

4. P1.C2: corrigir a conjugação do

verbo em "modelo da AFTNFN

propõe**m**..." **ok**.

5. P1.C2: corrigir a conjugação em

"VI - Resultados Obtidos e VII ...

descreve**m**..." **ok**.

6. P1.C2: corrigir a conjugação em

"Bases de dados proveniente**s**..." **ok**.

7. P1.C2: corrigir concordância em

"...o problema de indução

incremental de árvores de decisão

para atributo**s**..." **ok**.

8. P2.C2: corrigir concordância em

"... Para tratar de forma eficaz a

incerteza, torna-se

necessári**o**..." **ok**.

9. P3.C1: corrigir a frase "...

Neste artigo, foi utilizado o método

de chamado..." **ok**

10. P3.C1: acentuar "cont**e**m". **ok**.

11. P3.C1: corrigir concordância em

"A classificação **o** priori **ok**.

ABC..." seguido de "... foi obtida **ok**

através...Vendas, **foi** utilizada

para..." **ok**.

12. P3.C2 (final): corrigir grafia

em "Após o tr**ie**namento..." **ok**.

13. P4.C1: corrigir concordância em

"algoritmo da SOM e **o**

resultados..." **ok**.

14. P4.C1: corrigir concordância em

"... determinação de uma árvore**s**

de decisão..." **ok**.

15. P5.C1 (início): corrigir grafia

em "... foram do tipo

gau**ssi**na..." **ok**.

16. P5.C2: corrigir concordância em

"... foram executad**a**s 500

ciclos..." **ok**.

17. P6.C1: corrigir a frase "...

seleccio**nou**-se **,**

aleatoriamente **,** 100

registros..." **ok**.

18. P7.C1: agrupar a legenda com a

tabela. **ok**.

19. P7.C1: corrigir concordância em

"... efetua pequenos volum**e** de

devoluções..." **ok**.

20. P7.C1: agrupar o símbolo de

porcentagem ao valor 38,8. **ok**.

21. P7.C1: corrigir a grafia em

"Baseado no signifi**ca** de cada

classe..." **ok**.

22. P7.C1: corrigir concordância em

"... mais informação será

necessá**rio** para identificar..." **ok**

23. P7.C2 (final): corrigir

acentuação em "...relacionadas **a**

qualidade e construção do modelo..." **ok**

24. P7.C2 (final): corrigir

concordância em "... seções

anteriores suge**re**-se as

seguintes..." **ok**.

25. P8.C1 (início): corrigir

acentuação em "conseq**u**entemente" **ok**

26. P8.C1: corrigir a frase em

"Comparar e avaliar **e**

vantagens..." seguido de "...modelo

propost**a** da AFNTN..." **ok**

27. P8.C1 (final): corrigir grafia

em "sis**t**mas". **ok**.

No parágrafo acrescentado às conclusões

sobrou uma duplicação de palavra:

"concluí-se que concluí-se" **ok**.