

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
DE PRODUÇÃO E SISTEMAS**

WELVIS DOUGLAS DA SILVA MORETTO

**PREVISÃO DE INADIMPLÊNCIA USANDO
CLASSIFICAÇÃO DE DADOS BASEADA EM REDES
NEURAIS ARTIFICIAIS**

CURITIBA

2009

WELVIS DOUGLAS DA SILVA MORETTO

**PREVISÃO DE INADIMPLÊNCIA USANDO
CLASSIFICAÇÃO DE DADOS BASEADA EM REDES
NEURAIS ARTIFICIAIS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Sistemas da Pontifícia Universidade Católica do Paraná como requisito para obtenção de título de Mestre em Engenharia de Produção e Sistemas.

Área de concentração: Automação e Controle de Processos.

Orientador: Prof. Dr. Leandro dos Santos Coelho.

CURITIBA

2009



TERMO DE APROVAÇÃO

WELVIS DOUGLAS DA SILVA MORETTO

PREVISÃO DE INADIMPLÊNCIA USANDO CLASSIFICAÇÃO DE DADOS BASEADA EM REDES NEURAIS ARTIFICIAIS

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre no Curso de Mestrado em Engenharia de Produção e Sistemas, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, do Centro de Ciências Exatas e de Tecnologia da Pontifícia Universidade Católica do Paraná, pela seguinte banca examinadora:

Prof. Dr. Leandro dos Santos Coelho (PUCPR)
Orientador

Prof. Dr. Luiz Marcio Spinosa (PUCPR)
Membro Externo

Prof. Dr. Julio Cesar Nievola (PPGIA / PUCPR)
Membro Externo

Curitiba, 20 de agosto de 2009.



A Deus, por ter me sustentado em todos os momentos.

Porque certamente te livrarei, e não cairás à espada;

mas a tua alma terá por despojo,

porquanto confiaste em mim, diz o SENHOR.

AGRADECIMENTOS

Em primeiro lugar agradeço a Deus, que me proporcionou a graça e viver, e aprender e crescer. Por ter me dado força e por sua presença em todos os momentos durante a elaboração desta dissertação.

A minha mãe, Luzia Eunice da Silva Moretto, foi quem me incentivou a fazer este mestrado, e que também sempre dedicou com amor e com quem aprendi a ter simplicidade.

A meu pai, Jair Moretto, que me deu o apoio necessário para meus estudos e com quem aprendi a ter humildade o meu muito obrigado.

A minha esposa, Crislaine Lucas Delfino Moretto, pela dedicação, paciência apoio e carinho, para que juntos pudéssemos provar o sabor desta conquista.

À minha família pelo apoio, amor e compreensão por minhas ausências.

Ao Prof. Leandro dos Santos Coelho, meu orientador, agradeço a grande e positiva influência que efetivamente exerceu em minha formação acadêmica e pela contribuição neste trabalho.

Ao LACTEC (Instituto de Tecnologia para o Desenvolvimento) pela bolsa de estudos o meu muito obrigado.

Aos colegas de trabalho, por todo o apoio dado.

E são tantas as pessoas que, das mais diferentes formas, participaram desta história. A elas, o meu mais sincero agradecimento.

RESUMO

A concessão de crédito é uma decisão importante para as instituições financeiras. Devido a isso, cada vez mais as empresas têm buscado sistemas computacionais para fazer a pontuação de crédito (*credit score*). O objetivo da pontuação de crédito é prever a probabilidade da concessão de crédito para um determinado cliente, podendo o cliente ser “adimplente” ou “inadimplente”. Para o desenvolvimento de um sistema computacional de pontuação de crédito podem ser usados diversos métodos, tais como programação linear, análise discriminante e árvore de decisão. Recentemente, os modelos baseados em redes neurais artificiais (RNs) tem despertado interesse para as instituições financeiras. Uma vantagem da utilização das RNs é a capacidade de detectar relações não lineares entre os diversos atributos que podem influenciar no comportamento do cliente. Neste trabalho, os modelos de pontuação de crédito foram gerados através da utilização de RNs, visando a previsão da probabilidade de um cliente ser (ou não) inadimplente. Na construção de um modelo adequado de classificação de dados para a realidade da empresa foram utilizados para a geração da pontuação de crédito as seguintes RNs: i) perceptron multicamadas (MLP, *Multilayer Perceptron*) com treinamento baseado no método de Levenberg-Marquardt; ii) RN parcialmente recorrente de Elman usando método de aprendizagem quase-Newton do tipo Broyden, Fletcher, Goldfarb e Shanno (BFGS) e iii) RN de função de base radial (RBF) treinada pelo algoritmo de descida de encosta. A utilização das arquiteturas de RNs mencionadas visa verificar qual das RNs apresenta melhor desempenho para a classificação dos dados, estes disponibilizados pela empresa DM-Inf S/A. O estudo proposto visa a classificação dos clientes, para obter a informação dos clientes que tem a probabilidade de ser (ou não) inadimplente. Os resultados obtidos nesta dissertação foram satisfatórios, tanto no treinamento, como também na validação dos resultados das diferentes RNs. A RN com arquitetura MLP obteve melhor desempenho na classificação dos clientes do que as RNs de Elman e RN-RBF para o caso estudado.

Palavras Chave: Pontuação de Crédito, Redes Neurais Artificiais, Inadimplência, Classificação de dados.

ABSTRACT

The granting of credit is an important decision for financial institutions. Because of this, more companies have sought to make the computer systems of credit scores. The purpose of credit scores is to predict the probability of granting credit to a client, the client may be “accepted” or “rejected”. For the development of a computer system for credit scores can be used several mathematical methods, such as linear programming, discriminant analysis, and decision tree. Recently, models based on artificial neural networks (ANN) has attracted interest for financial institutions. One advantage of the use of neural networks is the ability to detect non-linear relationships between the various attributes that can influence the behavior of the customer. In this work, models of credit scores were generated through the use of neural networks, to indebted the probability of a customer is or not too. In the construction of an appropriate model for the reality of the company were used the following neural networks for the generation of credit scores, partially recurrent ANN: i) multilayer perceptron (MLP, Multilayer Perceptron) with training based on the method of Levenberg-Marquardt, ii) partially ANN of Elman type method using the quasi-Newton method in Broyden, Fletcher, Goldfarb and Shanno (BFGS) approach and iii) an ANN radial basis function (RBF) trained by the descent gradient algorithm. The use of ANN architectures to verify which ANN shows better performance for data classifying, the data provided by DM-Inf S/A. This study proposes the customers classification, to obtain the customer information about their in default risk. Results in this dissertation were satisfactory, both in training and validation phases of validation ANNs. The ANN with MLP architecture obtained best performance in the customer’s classification than the ANNs of Elman and RBF neural networks for the evaluated case study.

Key-words: Credit Score, Artificial Neural Networks, Default, Data Classification.

LISTA DE FIGURAS

Figura 1.1: Estruturação do encadeamento lógico da dissertação	13
Figura 2.1: Distribuição dos dados utilizados para o processo de estimação e validação das RNs.....	18
Figura 3.1: Modelo de um neurônio artificial.....	23
Figura 3.2: a) Rede neural <i>feedforward</i> acíclica, b) Rede neural recorrente.....	23
Figura 3.3: Rede parcialmente recorrente de Elman.....	27
Figura 3.4: Representação de RN <i>feedforward</i> com uma camada escondida.....	29
Figura 3.5: Arquitetura de uma RN-RBF.	40

LISTA DE TABELAS

Tabela 2.1: Descrição das variáveis utilizadas.....	17
Tabela 4.1: Resultado de treinamento da RN-MLP.....	44
Tabela 4.2: Resultado de validação da RN-MLP.....	45
Tabela 4.3: Resultado de treinamento da RN de Elman.....	46
Tabela 4.4: Resultado da validação da RN de Elman.....	47
Tabela 4.5: Resultado de treinamento da RN-RBF.....	48
Tabela 4.6: Resultado de validação da RN-RBF.....	49
Tabela 4.7: Resultado de treinamento das RNs.....	50
Tabela 4.8: Resultado de validação das RNs.....	51

LISTA DE ABREVIATURAS

AG	– Algoritmo Genético
AR	– <i>Auto-Regressive</i>
ARIMA	– <i>Auto-Regressive Integrate Moving Average</i>
BD	– Banco de Dados
BFGS	– Broyden, Fletcher, Goldfarb e Shanno
BPTT	– <i>Back Propagation Through Time</i>
CART	– <i>Classification and Regression Trees</i>
CISP	– Central de Informação de São Paulo
DFP	– Davidson, Fletcher e Powell
MLP	– <i>MultiLayer Perceptron Network</i>
MSE	– <i>Mean Square Error</i>
RBF	– <i>Radial Function Neural Network</i>
RNs	– Redes Neurais
RTRL	– <i>Real Time Recurrent Learning</i>
SG	– <i>Stocked Generalization</i>
SOM	– <i>Self-Organizing Map</i>
SPC	– Central de Proteção ao Crédito
SVM	– <i>Support Vector Machines</i>
UCI	– University of California

LISTA DE SÍMBOLOS

D	– Saída desejada apresentada durante o treinamento
E	– Sinal de erro
$e(n)$	– Erro da rede no passo n do treinamento
$e_j(n)$	– Sinal de erro da unidade de saída j na iteração n
$f_j(\cdot)$	– Função de ativação associada à unidade j
i, j	– Índices referentes a diferentes neurônios da rede
M	– Número de camadas
Nn	– m -ésimo vetor de entrada (iteração)
N	– Número de amostras (padrões de treinamento)
S	– Matriz de dados de saída (saídas desejadas)
$s_j(n)$	– m -ésimo elemento do vetor de saídas
$u_j(n)$	– Ativação da unidade j na iteração n ; sinal a ser aplicado à não-linearidade
$w(n)$	– Valor de ajuste a ser acrescentado ao peso w_{ij}
$w_{ij}(n)$	– Peso sináptico conectando a saída da unidade i à entrada da unidade j na iteração n
X	– Matriz de dados de entrada (amostras de treinamento)
$x_j(n)$	– m -ésimo elemento do vetor de entradas
Y	– Saída real da rede após a apresentação do estímulo de entrada
$y_j(n)$	– Sinal de saída da unidade j na iteração n

SUMÁRIO

AGRADECIMENTOS	iv
RESUMO	v
ABSTRACT	vi
LISTA DE FIGURAS	vii
LISTA DE TABELAS	viii
LISTA DE ABREVIATURAS	ix
LISTA DE SÍMBOLOS	x
SUMÁRIO	ix
CAPÍTULO 1 - INTRODUÇÃO	2
1.1 Contextualização	2
1.2 O problema de pesquisa	4
1.3 Objetivos de Pesquisa	4
1.3.1 Objetivo geral	5
1.3.2 Objetivos específicos	5
1.4 Justificativa de pesquisa	6
1.4.1 Justificativa teórica	6
1.4.2 Justificativa prática	6
1.4.3 Revisão da Literatura	8
1.5 Caracterização da pesquisa	10
1.5.1 Estratégia de Pesquisa	12
1.6 Estrutura desta dissertação	14
CAPÍTULO 2 - DESCRIÇÃO DO PROBLEMA	15
2.1 Limitações da pesquisa	15
2.2 Levantamento dos dados	16
2.3 Atributos considerados no problema de classificação pelas RNs	16

2.4 Dados utilizados	17
CAPÍTULO 3 - FUNDAMENTAÇÃO TEÓRICO-EMPÍRICA	19
3.1 A pontuação de crédito	19
3.1.1 Limitações da pontuação de crédito	20
3.2 Fundamentos das redes neurais artificiais	21
3.2.1 Arquitetura de RNs	22
3.3 Redes neurais parcialmente recorrentes	24
3.4 Perceptron de múltiplas camadas	28
3.4.1 Algoritmo de retropropagação do erro	29
3.4.2 Método quase-Newton	33
3.4.3 Método de otimização Levenberg-Marquardt	36
3.5 Rede neural de função de base radial (RN-RBF)	37
3.5.1 Arquitetura de uma RN-RBF	39
CAPÍTULO 4 - APRESENTAÇÃO E ANÁLISE DOS RESULTADOS	42
4.1 Coleta e tratamento dos dados	42
4.2 Configuração dos métodos	43
4.3 Processo de aprendizado e validação das RNs	44
CAPÍTULO 5 - CONCLUSÃO E RECOMENDAÇÕES PARA TRABALHOS FUTUROS	52
5.1 Conclusão	52
5.2 Recomendação para trabalhos futuros	53
REFERÊNCIAS	55

Capítulo 1

INTRODUÇÃO

Esta dissertação relata uma pesquisa desenvolvida no âmbito do Programa de Mestrado em Engenharia de Produção e Sistemas (PPGEPS) da Pontifícia Universidade Católica do Paraná, estando esta vinculada a área de automação e controle de processos.

Esta dissertação foca-se na classificação de clientes adimplentes e inadimplentes que ocorrem no domínio de conhecimento do setor alimentício da empresa DM-Inf S/A.

Este capítulo está estruturado em seis seções. A seção 1.1 refere-se à contextualização desta dissertação. A seção 1.2 apresenta o problema de pesquisa. Na seção 1.3 apresenta os objetivos gerais e específicos abordados. Na seção 1.4, a justificativa teórica, prática e a revisão da literatura são comentadas. Na seção 1.5 refere-se à caracterização da dissertação, enquanto a seção 1.6 define a organização do restante deste documento.

1.1 Contextualização

Muitas empresas sofrem com consumidores inadimplentes, fazendo com que muitos empresários sintam-se desprotegidos. Lembrando que a ocorrência da inadimplência é comum em tempos de crise (SEBRAE, 2008).

O risco de crédito é uma entidade presente em mercados financeiros. O risco de crédito é tão antigo quanto os empréstimos em si, e a inadimplência é o principal fator que aflige qualquer instituição com fins lucrativos. Nos últimos anos, em razão do crescimento e globalização do mercado mundial, o assunto passou a merecer atenção e preocupação, notadamente em razão do acentuado aumento dos índices de inadimplências (GUIMARÃES, 2006).

A pontuação de crédito (*credit scoring*) pode ser definida como uma abordagem de avaliar o risco em empréstimos, podendo incluir características dos clientes, dados

históricos e/ou técnicas estatísticas. O método de pontuação de crédito é utilizada frequentemente pelos credores para formar um *ranking* de risco dos empréstimos (MESTER, 1997).

Recentemente, com o rápido crescimento da indústria de crédito, os modelos de pontuação para crédito têm sido cada vez mais utilizados para avaliar com rapidez e consistência o risco de conceder (ou não) crédito para novos clientes. Os modelos de pontuação de crédito são desenvolvidos para determinar quem são os bons pagadores e quais são os maus pagadores (ARNAUD *et al.*, 2005).

Neste contexto, faz-se necessário que as instituições financeiras detenham algum modelo de análise de pontuação de crédito, de modo que seja possível criar instrumentos que sejam capazes de auxiliar as instituições financeiras na tarefa de decidir sobre quando conceder (ou não) crédito e para quem.

Em um mercado competitivo é necessário prever de forma que sejam tomadas ações inovadoras, transformando a informação em vantagem competitiva. A procura de métodos que capturam as características de um comportamento no passado a fim de prever os acontecimentos com certo grau de confiança é uma área importante de pesquisa (BÓDIS, 2004).

A previsão é útil e possui exemplos de aplicações nas mais diferentes áreas de conhecimento, tais como Física (WEIGEND, 1994), Astrofísica (WEIGEND, 1990), Engenharia Elétrica (MARTIN, 2000), Ciência da Computação (LELAND, 1994; FAYYAD *et al.*, 1996), Governo e Indústria (MAQSOOD e ABRAHAM, 2006), entre outras.

Na pontuação de crédito sem a utilização de uma ferramenta computacional eficiente no auxílio à tomada de decisão, dois ou mais grupos de analistas podem chegar a conclusões distintas para a concessão (ou não) de crédito. Logo, com a existência de um processo decisório, sem abordar avaliações e conceitos subjetivos do tomador de decisões. Neste contexto, as redes neurais artificiais (RNs) podem ser um método útil no apoio à tomada de decisão (MATARAZZO, 1995).

As RNs são sistemas computacionais bio-inspirados que possuem características de processamento paralelo (BERRY e LINOFF, 2004; HAYKIN, 2001). Hand *et al.* (2001) mencionaram que as RNs possuem a habilidade de aprender a partir de dados de entrada (aprendizado supervisionado) ou sem um professor (aprendizado não-supervisionado).

Uma RN que pode ser aplicada com sucesso à previsão é o perceptron com

múltiplas camadas (MLP , *Multlayer Perceptron*) (PINO *et al.*, 2008). Uma segunda opção é a RN parcialmente recorrente de Elman que utiliza uma camada de contexto, guardando o último estado do sistema, a fim de que os estados anteriores interferindo nos estados futuros, gerando memória à RN (ELMAN, 1990). A RN-RBF também pode ser utilizada para a classificação de padrões e agrupamento de dados HAYKIN (1994).

1.2 O problema de pesquisa

Dada a contextualização do tema proposto, faz-se necessário explicitar o problema de pesquisa, antes de abordar as questões envolvidas. De acordo com Lakatos e Marconi (1995), o problema indica exatamente qual a dificuldade que se pretende resolver. Logo, formular o problema significa dizer de forma clara e precisa qual a dificuldade existente. Nesse sentido, tem-se como problema de pesquisa: “Classificar os clientes da empresa DM-Inf S/A como inadimplente ou adimplente”.

1.3 Objetivos de pesquisa

A utilização de um estudo para a pontuação de crédito usando três topologias de RNs, incluindo as RN-MLP, RN de Elman e a RN-RBF é validada nesta dissertação. Neste contexto, é validado nesta dissertação a análise dos dados da empresa DM-Inf S/A em conjunto com os órgãos de proteção ao crédito foi adotada, classificando o resultado como “adimplente” ou “inadimplente”, no momento da concessão de crédito. Os objetivos são divididos em objetivo geral e em objetivos específicos, os quais são descritos nas duas subseções apresentadas seguir.

1.3.1 Objetivo geral

O objetivo geral desta dissertação é classificar os dados disponibilizados pela empresa DM-Inf S/A e pelos órgãos de proteção ao crédito, tais como CISP (Central de Informação de São Paulo) e o SPC (Central de Proteção ao Crédito), classificando o cliente como adimplente ou inadimplente.

1.3.2 Objetivos específicos

Os objetivos específicos são os seguintes:

- projetar uma RN-MLP, treinada com o método de otimização Levenberg-Marquardt em ambiente computacional MATLAB da Mathworks;
- projetar uma RN de Elman, treinada com algoritmo quase-Newton do tipo BFGS (Broyden, Fletcher, Goldfarb e Shanno) em ambiente MATLAB;
- projetar uma RN-RBF treinada pelo algoritmo de descida de encosta em ambiente MATLAB;
- utilizar da saída das RNs para a classificação do cliente, classificando como adimplente ou inadimplente.
- verificar qual das topologias obteve melhor desempenho na classificação do dados.
- verificar a quantidade de neurônios na camada oculta que terá um desempenho apropriado na resolução de um problema de classificação para pontuação de crédito.

1.4 Justificativa da pesquisa

As justificativas teóricas e práticas desta dissertação encontram-se estruturadas nas subseções mencionadas a seguir.

1.4.1 Justificativa teórica

Os prejuízos causados a uma empresa, devido à inadimplência podem ser significativos, e a adoção de abordagens computacional que vise a redução da inadimplência podem trazer benefícios.

A mensuração da pontuação de crédito evoluiu drasticamente, nas últimas décadas, em resposta a um número de pesquisas externas que tem obtido resultados importantes em termos de:

- aumento de falências a nível mundial;
- tendência no sentido da desintermediação pela melhor quantidade e maiores mutuários;
- aumento na competitividade em termos de crédito;
- diminuição no valor dos ativos (e, portanto, garantia) em muitos dos mercados (ALTMAN e SAUNDERS, 1998).

1.4.2 Justificativa prática

A previsão de riscos é um tema de estudos em finanças. A gestão de preços ou obrigações de preços de crédito representa um importante conjunto de procedimentos no

intuito de reduzir a estimativa de crédito, e pode dar suporte a decisão de conceder ou não créditos.

O crédito nas diversas modalidades financeiras, incluindo cartões de crédito, cheque especial e boletos, tem sido tratado empiricamente ou por sistemas computacionais de crédito baseado em análise discriminante ou regressão logística (HAND e HENLEY, 1997), (LIU e SCHUMANN, 2005), RN (CORRÊA e VELLASCO, 2009; MORIMITSU *et al.*, 2009; MENDES *et al.*, 2009; STEINER *et al.*, 2007), CART (*Classification and Regression Trees*) (HAND e HENLEY, 1997; KOMORÁD, 2002). Komorád (2002) apresentou diversos métodos para a pontuação de crédito. Entretanto, Hand e Henley (1997) afirmaram que nenhum modelo para pontuação de crédito é 100% eficiente.

Neste contexto, nos últimos anos, em razão do crescimento do mercado de crédito do Brasil, o assunto passou a merecer atenção e preocupação devido a crescente elevação das taxas de inadimplência, e também pelo interesse das instituições financeiras em desenvolver modelos eficientes para a pontuação de crédito.

Uma avaliação equivocada do percentual de um candidato pela pontuação de crédito classificada com “inadimplente” tem como conseqüências a geração de ganhos não realizado ao negar o crédito, enquanto que, a pontuação de crédito errônea de um cliente classificada como “adimplente” tem como conseqüência provável o prejuízo pela perda do valor envolvido na operação financeira.

Uma alta na quantidade do volume de crédito justifica que a instituição financeira adote um sistema computacional para fazer a pontuação de crédito. Neste contexto, a adoção de um sistema computacional pode levar a um ganho à instituição financeira. No entanto, para isto é necessário um bom projeto e avaliação dos sistemas computacionais para pontuação de crédito, onde são analisadas as informações disponibilizadas pela empresa, classificando o cliente com “adimplente” ou “inadimplente”.

Na maioria dos modelos computacionais de pontuação de crédito, os responsáveis pela análise de crédito trabalham com uma acentuada variedade de informações vindas das mais diversas fontes de informação. Muitas destas informações podem ser incompletas, ambíguas, parcialmente incorretas e/ou de relevância dúbia. Os gerentes analisam estas informações de maneira subjetiva, onde, às vezes não conseguem explicar os processos de tomada de decisão. Atualmente, existe uma crescente utilização de sistemas computacionais visando a resolução do problema de pontuação de crédito. Por exemplo, pode-se usar a

mineração de dados para o mapeamento das variáveis relevantes e a RNs para o mapeamento das irregularidades, fornecendo uma saída para a concessão de crédito. Este tipo de abordagem computacional pode ser projetada com mecanismos automáticos de atualização e validação possuindo vantagem sobre muitos métodos estatísticos utilizados em economia e em finanças, entre os métodos estatísticos usados neste contexto estão à predição, avaliação e aprovação de crédito (TRIPPI e TURBAN, 1993).

1.4.3 Revisão da Literatura

A seguir é apresentado um breve apanhado da literatura em termos da utilização de RNs para problemas de previsão, classificação e/ou pontuação de crédito.

Bastos (2008) utilizou árvore de decisão para resolver um problema de pontuação de crédito, utilizando para isto duas bases de dados disponíveis na internet para cartão de crédito do UCI (*University of California*), *Irvine Machine Learning Repository*. A previsão de árvore de decisão é utilizada por duas técnicas da mineração de dados, a RN-MLP e o SVM (*Support Vector Machines*).

Liu e Schumann (2005) utilizaram algoritmos de classificação para a resolução de um problema de pontuação de crédito. Na resolução do problema de classificação foram utilizados os algoritmos de árvore de decisão, regressão logística, *k*-médias e RN-MLP com o algoritmo de treinamento de retropropagação do erro.

Arnaud *et al.* (2005) utilizaram para a resolução de um problema de pontuação de crédito uma combinação do algoritmo *k*-médias e a RN-MLP. As simulações de Arnaud *et al.* (2005) mostram que as duas técnicas combinadas são mais eficientes para resolver o problema de pontuação de crédito do que a RN-MLP utilizada de forma isolada.

Outra técnica com pesquisas em previsão é a metodologia para construção de modelos ARIMA (Auto-Regressivo Integrado de Média Móvel, do inglês, *Auto-Regressive Integrated Moving Average*) que foi apresentada pelos pesquisadores BOX e JENKINS (1970), podendo ser utilizada junto com uma RN para a previsão.

Neste contexto, Hansen e Nelson (2000) propuseram uma abordagem para o

método SG (*Stacked Generalization*) para a previsão de rendimentos e impostos vinculados ao orçamento dos Estados Unidos da América, utilizando mineração de dados, modelo ARIMA e RNs.

A previsão por métodos baseados em RNs, ainda carece de uma análise detalhada em termos de desempenho e métodos de projeto nos trabalhos apresentados na literatura recente. Entretanto, a literatura recente é vasta.

Lam (2004) realizou uma série de experimentos para testar o poder da predição de RNs na área financeira. Os resultados obtidos mostram que a RN adotada supera uma estratégia baseada em investimento diversificado. A técnica adotada baseada em RNs incorpora séries históricas financeiras de dados financeiros de 364 companhias de S&P e os dados de variáveis macro econômicas. Neste contexto, foram coletadas algumas variáveis para a comparação de uma RN com aprendizado supervisionado.

Giordano *et al.* (2007) apresentaram uma metodologia para construir intervalos de predição de dados não-lineares de séries temporais utilizando RNs com *bootstrap* para classificação.

Thomassey e Happiette (2006) propuseram abordagens de RNs, sistemas nebulosos e modelos de regressão para previsão de vendas para a distribuidora *Apparel*, visando uma produção eficiente, a minimização de custos e a satisfação dos clientes.

Lai *et al.* (2006) apresentaram um estudo de pontuação de crédito, baseado em RN supervisionadas. Uma RN com aprendizado supervisionado foi proposta para diferentes modelos de aprendizagem avaliados, de modo a melhorar a qualidade da previsão de inadimplência.

Em termos da RN de Elman, a literatura recente também é ampla quanto a pesquisas em previsão.

Kremer (1995) comentou que apesar da simplicidade estrutural da RN de Elman, ela é capaz de modelar informações temporais utilizando o modelo de neurônio de MacCulloch-Pitts e o algoritmo de retropropagação do erro.

Gao (1996) propôs uma arquitetura de RN de Elman modificada, apresentando para o mesmo um modelo melhorado, onde foram adicionadas conexões entre a camada de contexto e a camada de saída e também foi proposta uma alteração no algoritmo de aprendizado. Zhang *et al.* (2007) propôs que o modelo de RN de Elman simplificado,

diferindo da proposta original de Elman (ELMAN, 1990).

Pham e Karaboga (1998) verificou que as RNs de Elman e Jordan apresentam melhores resultados, na identificação de sistemas dinâmicos quando treinadas com um AG (algoritmo genético), quando comparado com o algoritmo original de treinamento de retropropagação do erro.

Brunelli (2006) obteve melhores resultados na previsão da concentração de dióxido de enxofre, em construções próximas a áreas industriais, projetando uma RN de Elman e comparando suas previsões com outras obtidas com uma RN *feedforward*.

Ge *et al.* (2007) conseguiu resultados promissores na identificação de parâmetros de um motor ultrasônico. Para isso uma RN de Elman otimizada por um algoritmo de enxame de partículas modificado foi empregada. Os resultados foram comparado com a de uma RN de Elman treinada por um algoritmo baseado em algoritmo descida de encosta.

Souza (2008) afirmou que uma outra RN que pode ser aplicada com sucesso à previsão de séries temporais é a RN-RBF.

A RN-RBF apresenta potencialidades de generalização local e seu aprendizado é relativamente rápido. Tais potencialidades, úteis para a previsão de séries temporais, são amplamente difundidas na literatura (HUTCHINSON, 1994).

Recentemente, a utilização de diferentes abordagens de métodos de aprendizado para as RNs-RBF também tem sido estudadas. Os exemplos de tais estudos são a aplicação de métodos, tais como mínimos quadrados ortogonal (BILLINGS, 1998) e métodos de agrupamentos de dados (GUERRA, 2006).

1.5 Caracterização da pesquisa

O emprego de métodos e técnicas específicas de cada tipo de pesquisa é importante para viabilizar o processo de pesquisa, proporcionando a orientação necessária para a obtenção, análise e interpretação dos dados obtidos no decorrer do trabalho.

A pesquisa deve assegurar que o problema seja abordado de uma maneira válida,

confiável e adequado com relação aos conceitos e objetivos envolvidos na pesquisa (SANTOS, 2006).

Para se realizar uma pesquisa é preciso promover o confronto entre os dados, as variâncias, as informações coletadas sobre determinado assunto e o conhecimento teórico acumulado a respeito dele (LÜDKE e ANDRÉ, 1986). Um aspecto relevante à classificação das pesquisas diz respeito à forma de abordagem do problema que pode ser quantitativa (GIL, 1991) e/ou como uma pesquisa operacional (JUNG, 2004).

A pesquisa quantitativa considera que tudo pode ser quantificável, o que significa produzir em números, opiniões e informações para classificá-las e analisá-las (GIL, 1991).

Na pesquisa quantitativa os dados geralmente são originários de um grande número de pessoas que por meio de escalas numéricas, seguem a seguir com rigor um plano previamente estabelecido, baseado hipótese claramente indicadas e variáveis que são objeto de definição operacional (MATTAR, 1999).

A pesquisa pode ser classificada com quantitativa, pois baseiam-se em dados históricos para a detecção de padrões de comportamento utilizados modelos matemáticos para a obtenção do resultado.

A pesquisa também pode ser classificada do ponto de vista dos procedimentos técnicos, tais como, estudo de caso. Quando envolve o estado profundo e exaustivo de um ou poucos objetos de maneira que se permite o seu amplo e detalhado conhecimento.

Segundo Jung (2004) e Souza (2008), as pesquisas científicas também podem ser classificadas quanto à natureza, objetivos, procedimentos, fonte de referências, além de ambiente e tempo de aplicação. Em relação à natureza, esta pesquisa enquadra-se como tecnológica, uma vez que conhecimentos básicos são aplicados e conhecimentos são gerados como resultado do processo de pesquisa.

Quanto aos objetivos, esta pesquisa apresenta-se como explicativa. As pesquisas explicativas, segundo Jung (2004), visam à “identificação dos fatores que contribuem para a ocorrência dos fenômenos ou variáveis que afetam o processo”. Assim, nessa pesquisa, o objetivo é colaborar com a hipótese da influência de determinados fatores na previsão de pontuação de crédito.

Referente aos procedimentos, essa pesquisa caracteriza-se como operacional. Segundo Jung (2004), a pesquisa operacional “tem por princípio a investigação de forma sistemática e racional dos processos envolvidos” e utiliza ferramentas estatísticas e métodos

matemáticos para obtenção da melhor solução. Dessa forma, observa-se que essa pesquisa utiliza análises estatísticas e métodos matemáticos para a avaliação dos resultados.

A pesquisa também pode ser classificado como uma pesquisa operacional, porque tem como objetivo principal como a classificação de clientes inadimplentes e adimplentes. Para a aplicação de uma pesquisa operacional, pode-se abordar as seguintes fases: i) definição do problema; ii) a construção do modelo; iii) a validação do modelo; iv) e implementação do modelo.

O ambiente de aplicação desta pesquisa é laboratorial. A pesquisa em laboratório, “caracteriza-se pela possibilidade de se controlar as variáveis que possam interferir no experimento”. Neste contexto, esta pesquisa investiga os dados disponibilizados pela empresa DM-Inf S/A e pelos órgãos de proteção ao crédito para a pontuação de crédito.

Além disso, o fato de utilizar as informações dos clientes da empresa DM-Inf S/A dentro de um intervalo de tempo determinado, no caso entre Fevereiro de 2005 a Maio de 2008, faz com que esta pesquisa seja caracterizada quanto ao tempo de aplicação como longitudinal (JANG, 2004).

Com relação à fonte de referências essa pesquisa é bibliográfica, pois a pesquisa bibliográfica procura identificar as diferentes contribuições existentes na literatura sobre o tema. Neste estudo, revisa-se a literatura a respeito de previsão para pontuação de crédito e relativa ao uso de RNs para problemas de classificação.

1.5.1 Estratégia de pesquisa

Com o intuito de situar o leitor em relação aos passos utilizados para o desenvolvimento desta dissertação, este item da dissertação demonstra a construção da Estratégia de Pesquisa utilizada para alcançar os objetivos propostos para cada uma das etapas desta dissertação.

A Estratégia de Pesquisa utiliza-se das RNs para a resolução de um problema de pontuação de crédito. Nas seções 3 é descrito a pontuação de crédito e o referencial teórico das RNs. Na seção 4 são apresentados os passos utilizados para a resolução do problema de

classificação.

Considerando os objetivos que foram fixados nas seções 1.3.1 e 1.3.2 é apresentado o encadeamento lógico da estratégia de pesquisa, através da representação da Figura 1.1.

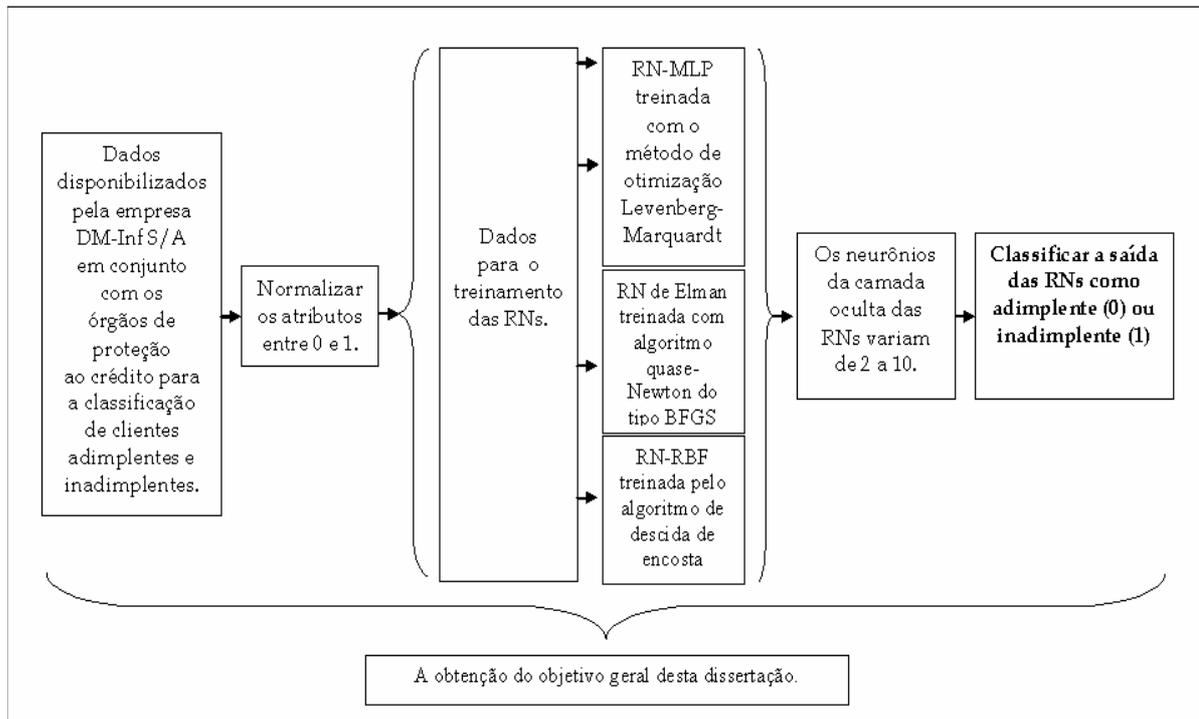


Figura 1.1: Estruturação do encadeamento lógico da dissertação.

A Figura 1.1 apresenta 6 etapas utilizadas para o desenvolvimento desta pesquisa, identificadas a seguir: i) o problema de pesquisa como motivador do estudo sendo considerado o ponto inicial à parte da metodologia; ii) trata-se da disponibilização dos dados pela empresa e órgãos de proteção ao crédito; iii) fazer a normalização dos atributos disponibilizados; iv) após a normalização dos atributos, os atributos vão servir como entradas nas RNs. As RNs vão variar a quantidade de neurônios na camada oculta de 2 até o 10 para cada uma das arquiteturas apresentadas na Figura 1.1; vi) a classificação do cliente, sendo este inadimplente e adimplente.

1.6 Estrutura desta dissertação

No primeiro capítulo, uma revisão sobre a pontuação de crédito de algumas das técnicas que podem ser utilizadas para a pontuação de crédito foi apresentada. O restante desta dissertação está organizada da seguinte forma. No segundo capítulo é apresentada a descrição do problema. No terceiro capítulo é descrita a fundamentação teórico-empírica da pontuação de crédito e das RNs para a classificação dos dados. No quarto capítulo são apresentados os resultados obtidos e a análise comparativa entre os modelos de RNs analisados. No quinto capítulo são mencionadas as conclusões e algumas propostas para continuidade da mesma.

Capítulo 2

DESCRIÇÃO DO PROBLEMA

A empresa DM-Inf S/A. com sede na cidade de Curitiba, Paraná. A empresa, atualmente, trabalha no ramo alimentício e possui diversos clientes no Brasil e exterior. A empresa DM-Inf S/A possui problemas vinculados à inadimplência, onde os diversos fatores e crises econômicas nacionais e internacionais contribuem para isto. Entre as quais, a empresa DM-Inf S/A, até o presente momento, não possui uma ferramenta computacional eficiente para fazer uma avaliação ou pontuação de crédito de seus clientes, isto é, um *software* para avaliar os atributos relevantes, classificar os dados para a pontuação de crédito, e inferir os possíveis riscos na concessão do crédito.

A inadimplência contribui para a elevação dos riscos na concessão de crédito da empresa DM-Inf S/A. No entanto, alguns dos clientes inadimplentes ou que possuem uma baixa pontuação de crédito acabam proporcionando lucro à empresa, desta forma a empresa DM-Inf S/A concede o crédito a estes clientes, que na compra dos produtos alguns dos clientes não efetivam o pagamento no prazo determinados pela empresa DM-Inf S/A. Neste contexto, um percentual destes clientes pagam seus títulos com juros gerando uma receita para a empresa. Uma outra pequena parte dos clientes pagam apenas o valor do título sem os eventuais juros. Entretanto, a DM-Inf S/A acaba recebendo mesmo que seja apenas o valor original do título, para que a empresa possa diminuir seus prejuízos.

A empresa DM-Inf S/A espera obter uma eficiente pontuação de crédito decorrente do resultado fornecido por uma ferramenta computacional, que pontua os clientes quanto a sugerir a concessão ou não de crédito ao cliente.

2.1 Limitações da pesquisa

Nesta dissertação, o resultado das RNs para a classificação será apresentado como

uma pontuação de crédito de 0 “adimplente” ou 1 “inadimplente”, sendo este um fator que pode ser considerado com uma limitação da pesquisa. Nesta dissertação, não é utilizado como as saídas das RNs um percentual, classificando a concessão de crédito, que poderia ser por intervalos, tais como “adimplente” onde os valores das saídas das RNs sejam menores que 0,5, “em verificação”, onde os valores das saídas das RNs estejam entre 0,5 e 0,8, e os valores das saídas acima de 0,8 seriam classificados como crédito “inadimplente”.

2.2 Levantamento dos dados

Não se deve projetar uma pesquisa sem antes planejar o que será feito com os dados coletados. Os dados utilizados para a validação das RNs mencionadas nesta dissertação foram disponibilizados pelo banco de dados da empresa DM-Inf S/A, contando também com dados oriundos de órgãos de proteção ao crédito, tais como CISP e o SPC.

No planejamento realizado, nesta dissertação, antes da coleta dos dados foram verificadas as variáveis relevantes para a análise da pontuação de crédito.

2.3 Atributos considerados no problema de classificação pelas RNs

Os atributos utilizados para a realização dos testes nesta dissertação são apresentados na Tabela 2.1, sendo que os atributos descritos são os resultados das movimentações de entradas e saídas realizadas na empresa DM-Inf S/A e de informações disponibilizadas pelos órgãos de proteção ao crédito, conforme mencionados na subseção 2.2.

Para este estudo, o valor mínimo que pode ser atribuído a cada uma das variáveis que foi descrita na Tabela 2.1 é 0 (0%) e o valor máximo é 1 (100%).

Atributos №	Descrição das variáveis
1	Valor de maior acúmulo
2	Valor de débito atual total
3	Valor limite de crédito
4	Média ponderada de atraso de pagamento
5	Média aritmética de dias de atraso do pagamento
6	Valor de débito atual a vencer
7	Média ponderada de títulos a vencer
8	Prazo médio de vendas
9	Média ponderada de atraso de títulos vencidos
10	Valor de débito atual com vencimento maior que 5 dias
11	Valor de débito atual com vencimento maior que 15 dias
12	Valor de débito atual maior que 30 dias
13	Média ponderada de atrasos de títulos vencidos
14	Valor da penúltima compra
15	Situação do cálculo limite para crédito
16	Venda sem crédito antecipado
17	Valor da última compra
18	Pontuação de crédito (0 ou 1)

Tabela 2.1: Descrição das variáveis utilizadas.

Os atributos são quantitativos e representam as características e históricos dos clientes cadastrados na empresa DM-Inf S/A. Os atributos da Tabela 2.1 têm como o tipo de dado *real*, podendo receber valores inteiros ou de ponto flutuante.

2.4 Dados utilizados

A amostragem dos dados utilizados para esta dissertação é constituída por 96.540 vetores (campos do banco de dados), onde apenas 7 % (por cento) deste valor corresponde aos vetores utilizados para o treinamento. Para o procedimento de treinamento das RNs, para a avaliação dos resultados serão gerados 30 experimentos, onde os experimentos foram

gerados aleatoriamente com distribuição uniforme. A quantidade de vetores selecionados para o processo de aprendizagem das RNs pode aumentar, pois a quantidade de vetores depende da quantidade de clientes que a empresa DM-Inf S/A possui disponível para as suas movimentações, ou seja, a quantidade de vetores é referente a quantidade de clientes disponíveis para as movimentações de entradas ou saídas. Entretanto, a quantidade de vetores disponíveis para o processo de aprendizagem não diminui, pois os dados são utilizados para o processo de aprendizagem das RNs.

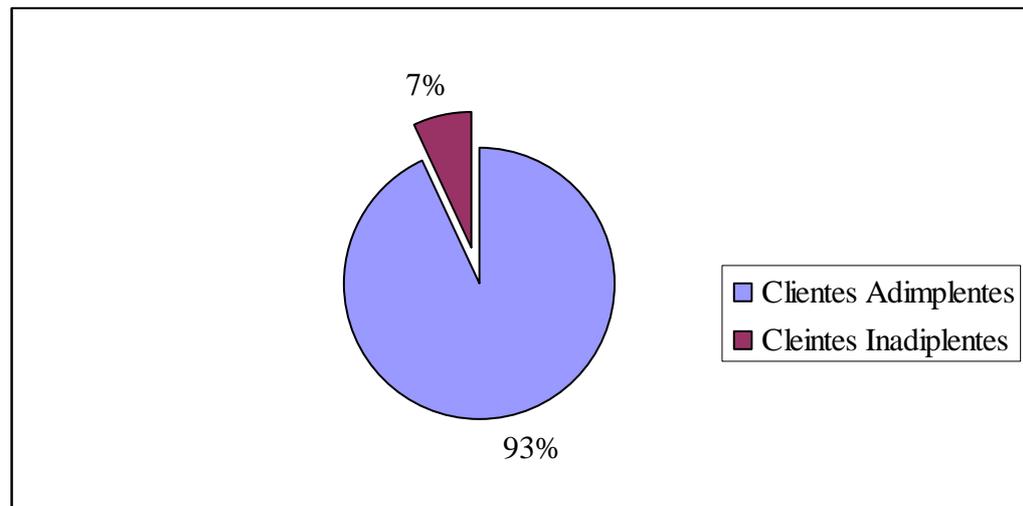


Figura 2.1: Distribuição dos dados utilizados para o processo de estimação e validação das RNs.

A figura 2.1 mostra que a maioria dos clientes estão adimplentes com (93%) da totalidade dos dados disponibilizados. Os clientes inadimplentes tem um total de (7%) dos dados disponibilizados. A amostragem adotada para o processo de estimação e validação das RNs foi de Fevereiro de 2005 a Maio de 2008, conforme descrito na subseção 1.5.

Capítulo 3

FUNDAMENTAÇÃO TEÓRICO-EMPÍRICA

Neste capítulo é apresentada a fundamentação teórico-empírica elaborada com o objetivo de dar suporte aos resultados obtidos a partir dos dados coletados, incluindo uma descrição da pontuação de crédito e suas limitações. Após, os fundamentos das RNs e os algoritmos de treinamento são discutidos.

3.1 A pontuação de crédito

A pontuação de crédito é um método estatístico usado para a previsão da possibilidade da concessão de crédito. O método, introduzido na década de 1950, é amplamente utilizado, recentemente, para a concessão de empréstimo, especialmente para cartões de crédito e concessão de crédito. Bastos (2008) afirma que estes modelos também podem ser utilizados em pequenas instituições, não apenas em instituições financeiras e bancos, onde a adoção da pontuação de crédito teve a sua origem.

Segundo Müller *et al.* (2002), a primeira etapa para construir uma pontuação de crédito eficaz, está na escolha dos dados necessários e relevantes para implementar a pontuação de crédito.

Mester (1997) menciona que a pontuação de crédito é um etapa importante na concessão de créditos em pequenos negócios.

Lai *et al.* (2006) comentam que o objetivo dos modelos de pontuação de crédito é o de atribuir aos requerentes de crédito, “bom crédito” ao grupo que provavelmente reembolsará a obrigação financeira ou “mau crédito” ao grupo que possui alta possibilidade de faltar com a obrigação financeira. Sendo assim, a pontuação de crédito é um método de avaliação dos riscos na concessão de crédito, que faz uso de dados históricos e/ou mesmo

técnicas estatísticas.

Para a construção de um bom modelo de pontuação é preciso ter-se dados históricos, que vão refletir no desempenho nos períodos de crise econômica. A avaliação do risco de crédito para decisões é relevante para as instituições financeiras, devido principalmente aos elevados riscos associados à decisão da concessão de crédito inadequada.

A pontuação de crédito acarretou maior atenção à indústria de crédito, podendo levar a uma melhoria no fluxo de fundos financeiros. Neste contexto, diferentes técnicas têm sido usadas como modelos de pontuação de crédito, tendo a sua origem nos bancos. Neste contexto, o problema da pontuação de crédito podem ser abordado como um problema de classificação de dados (JOHNSON e WICHERN, 2002).

A literatura de pontuação de crédito é vasta e é comum o uso de análise discriminante, regressão, métodos de programação matemática, LS-SVM (*Least Squares Support Vector Machines*) (LAI *et al.*, 2006), RNs (ATIYA, 2001; HAND e HENLEY, 1997; GUIMARÃES, 2006; CORRÊA e VELLASCO, 2009; MORIMITSU *et al.*, 2009; MENDES *et al.*, 2009) e mineração de dados (LIU e SCHUMANN, 2005; HUNG *et al.*, 2007), entre outros.

3.1.1 Limitações da pontuação de crédito

Os dados sobre os quais os modelos de pontuação de crédito baseiam-se para apoio à tomada de decisão devem ser constituídos de uma amostragem contendo dados relativos a clientes inadimplentes e adimplentes. Os dados utilizados para a construção dos modelos devem ser atualizados frequentemente, capturando as mudanças nos dados sempre que possível, tornando-se assim um conjunto de dados diversificado e completo. A pontuação de crédito tem a finalidade de aumentar a quantidade de clientes disponíveis para a venda; caso contrário, o modelo de pontuação de crédito não pode prever o comportamento de novos clientes.

Com a mudança da quantidade da concessão de crédito, maior será a precisão de um modelo que foi construído baseado em informações históricas.

Devem ser consideradas não só as características dos clientes a que foram concedidos crédito, mas também daqueles que foram negados. Caso contrário, um “viés na seleção” no processo de aprovação de empréstimo poderia conduzir a distorções na estimativa das ponderações na pontuação.

A precisão de um modelo deve ser, sempre que possível testada. Um bom modelo deve fazer previsões exatas em períodos favoráveis e também não favoráveis, por isso os dados sobre o modelo devem abranger tanto expansões quanto recessões. E o teste deve ser realizado usando amostras de concessões que não foram utilizados para desenvolver o modelo (MESTER, 1997).

Para Komorád (2002), devem ser escolhidos os dados de séries históricas, podendo ser tanto de fontes internas da empresa quanto de terceiros. As variáveis que entram na pontuação de crédito devem ser escolhidas com cuidado, como a quantidade de dados podem ser significativa e, assim computacionalmente problemática para os métodos utilizados. Dado que a maioria dos atributos da pontuação de crédito pode ser constituída de dados categóricos.

3.2 Fundamentos das redes neurais artificiais

As RNs têm sido freqüentemente utilizadas para a resolução de problemas que envolvam tarefas de classificação e regressão não linear. Neste contexto, as RNs podem ser ferramentas úteis se apropriadamente projetadas a partir de um procedimento do processo de aprendizagem de um conjunto de dados. Esta seção tem o intuito de introduzir os conceitos a respeito das RNs *feedforward* e recorrentes, além da descrição de alguns algoritmos de aprendizado.

Uma RN pode ser vista como um processador paralelamente distribuído constituído de uma unidade de processamento, também conhecida como neurônio ou nó. Uma RN tem seu funcionamento inspirado no cérebro e da aquisição do conhecimento a partir do ambiente, através de um processo de aprendizagem (HAYKIN, 2001).

Segundo Yao (1999), uma RN pode ser descrita como um grafo no qual cada nó i

realiza uma função de transferência, f_i da seguinte forma,

$$y_i = f_i \left(\sum_{j=1}^n w_{ij} x_j - \theta_i \right), \quad (3.1)$$

onde y_i é a saída do nó i , x_j é a j -ésima entrada para o nó, θ_i é um termo de polarização (*bias*), o w_{ij} é o peso da conexão entre os nós i e j . Normalmente, f_i é uma função não-linear, por exemplo, uma função degrau, sigmóide ou gaussiana.

Na equação (3.1), cada termo do somatório está relacionado com somente uma entrada x_j . As RNs de ordem mais alta, sendo aquelas que contém nós em mais de uma entrada são envolvidos na soma dos termos de um somatório. Um nó de segunda ordem pode ser representado da seguinte forma,

$$y_i = f_i \left(\sum_{j,k=1}^n w_{ijk} x_j x_k - \theta_i \right), \quad (3.2)$$

onde todas as variáveis tem a definição similares àqueles da equação (3.1).

3.2.1 Arquitetura de RNs

As RNs podem ser divididas em duas arquiteturas, a *feedforward*, tal como apresentado na Figura 3.2 a, e recorrente, conforme mostrado na Figura 3.2 b.

As RNs *feedforward* podem constituir de múltiplas camadas, onde a entrada de cada camada é a saída da denominada predecessora e a interconexão dos neurônios é acíclica.

As RNs *feedforward* são aproximadores universais de função, isto é, são capazes de aproximar mapeamentos entrada-saída não-lineares com certo grau de precisão. Sendo esta a propriedade motivadora para o uso das RNs em tarefas de reconhecimento de padrões

e aproximação de funções (CYBENKO, 1992; HORNIK, 1991).

Uma RN é *feedforward* se as saídas dos neurônios de uma determinada camada forem propagadas para uma camada posterior. As RNs *feedforward* podem ter uma ou mais camadas ocultas, onde a RN com uma ou mais camadas ocultas possui a capacidade de reconhecer padrões fora de seu conjunto de treinamento.

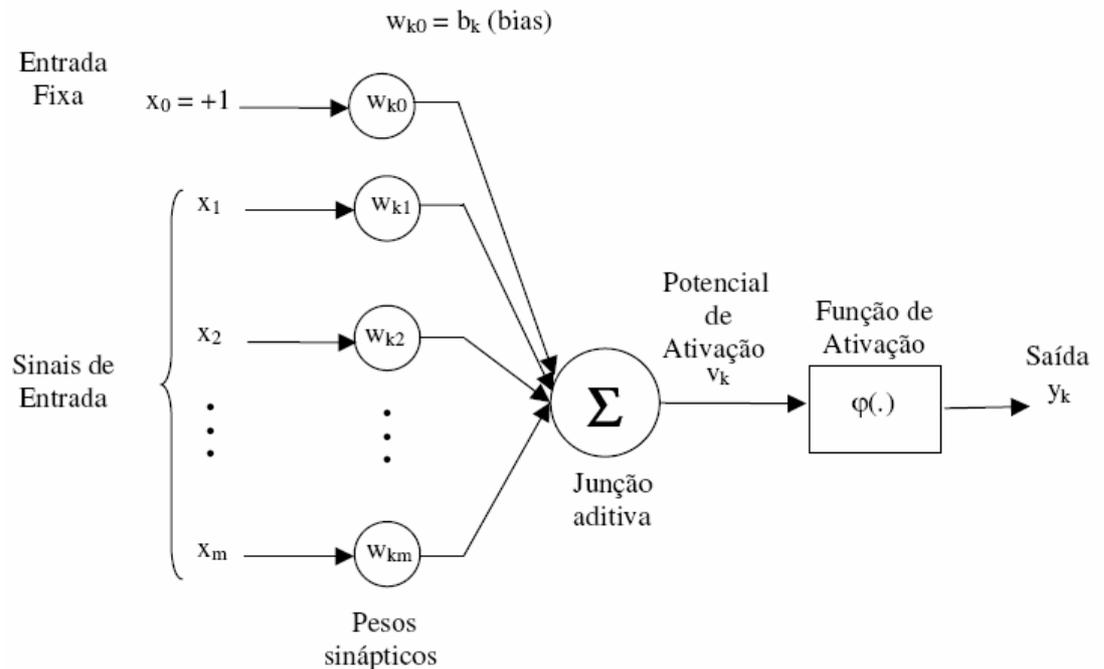


Figura 3.1: Modelo de um neurônio artificial.

Um exemplo de RN recorrente se é uma RN *feedforward* com pelo menos um laço de realimentação. As RNs parcialmente recorrente clássicas são as de Jordan (JORDAN, 1986) e a de Elman (ELMAN, 1990).

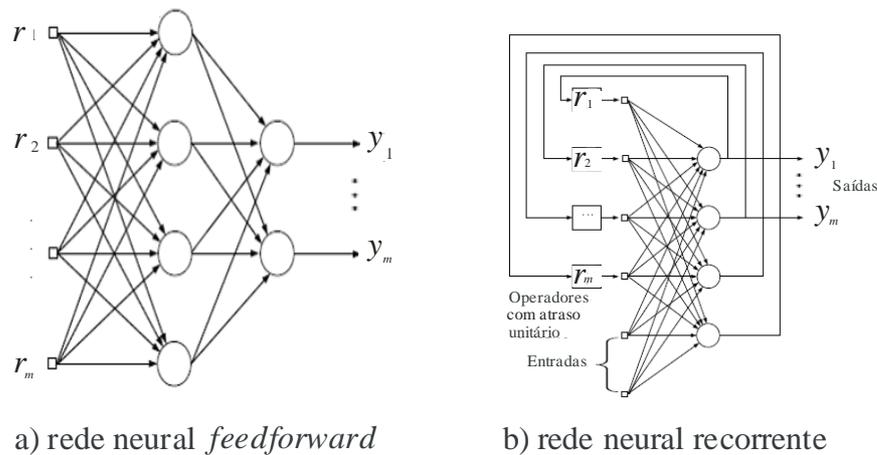


Figura 3.2: a) Rede neural *feedforward* acíclica, b) Rede neural recorrente.

O aprendizado ou treinamento é tipicamente realizado através de exemplos no caso do aprendizado supervisionado. O processo de aprendizado é obtido através do ajuste dos pesos das conexões presentes na RNs, onde uma vez treinada pode executar certas tarefas.

A essência de um algoritmo de aprendizagem é a regra de aprendizagem, isto é, a forma de ajuste dos pesos. A regra de aprendizagem determina como são modificados os pesos de cada conexão. Exemplos de regras de aprendizagem incluem a regra delta, regra de aprendizado Hebbiano e a regra de aprendizado competitiva (YAO, 1999).

3.3 Redes neurais parcialmente recorrentes

As RNs recorrentes contém a realimentação das saídas para as entradas, sendo suas saídas determinadas pelas entradas atuais e pelas saídas anteriores. Além disso, suas estruturas não são obrigatoriamente organizadas em camadas. Quando o são, estas RNs podem possuir interligação entre os neurônios da mesma camada ou mesmo entre camadas não consecutivas, gerando interconexões mais complexas que as presentes em RNs não recorrentes. Para Haykin (2001), as RNs recorrentes apresentam um ou mais laços de realimentação, podendo estes serem locais e/ou globais.

As RNs totalmente recorrentes, também conhecidas como redes simétricas,

caracterizam-se por apresentar conexões entre os neurônios nos dois sentidos. As RNs parcialmente recorrentes surgiram a partir de abordagens que utilizam ligações recorrentes em alguns neurônios pela necessidade de resolver problemas de instabilidade e complexidade do aprendizado das RNs totalmente recorrentes (SCHÜHLI, 2005).

As RN recorrentes tendem a possuir um processo mais complexo de aprendizado que as RNs *feedforward*, principalmente em função dos ciclos de treinamento (DE OLIVEIRA, 2003).

As RNs recorrentes são assim denominadas por apresentarem pelo menos um laço de realimentação. As conexões de realimentação podem originar tanto dos neurônios de saída quanto dos neurônios escondidos. A presença de laços de realimentação pode alterar significativamente a capacidade de aprendizagem e do desempenho de uma RN (HAYKIN, 2001).

Pode-se fundamentar que o modelo de RNs dinâmicas recorrentes sob a forma de equações de variáveis de estado. Assim, considerando um caso especial de uma RNs em que um vetor $x(n)_{p \times 1}$ representado por um vetor de entrada e o vetor $v(n)_{q \times 1}$ represente a saída da camada oculta no tempo n , pode-se então descrever o comportamento dinâmico dos modelos de RNs dinâmicas recorrentes podem ser descritas da seguinte forma,

$$v(n+1) = \phi(v(n), x(n)), \quad (3.3)$$

$$y(n) = \phi(v(n)), \quad (3.4)$$

onde $\phi(\cdot)$ é uma função não-linear que caracteriza a função de ativação das camadas ocultas e de saída da RN (HAYKIN, 1994).

Existem muitos algoritmos propostos na literatura para o processo de aprendizado de uma RN, o algoritmo mais utilizado é algoritmo de retropropagação através do tempo (BPTT, *Back Propagation Through Time*) (WILLIAMS e ZIPSER, 1995).

De Oliveira (2003) menciona que um dos algoritmos mais utilizados é o de aprendizagem recorrente em tempo real (RTRL, *real time recurrent learning*). Este algoritmo é baseado no cálculo do gradiente de uma média de erro de saída em relação aos pesos da RN.

A diferença entre a RN de Elman e a de Jordan é que, na RN de Jordan, a realimentação ocorre dos nós de saída para os nós de entrada, enquanto que na RN de Elman a realimentação sai dos nós das camadas escondidas. Existe também a RN de Elman estendida, na qual a realimentação sai dos nós da camada escondida e também dos neurônios de saída (DE OLIVEIRA, 2003).

A RN de Elman pode ser treinada com qualquer algoritmo usado nas RNs do tipo MLP, tal como o algoritmo da retropropagação do erro. Sendo assim, a RN de Elman opera de forma semelhante à RN de Jordan. A RN de Elman contém conexões de realimentação e as ativações para cada camada de neurônio são computadas, uma só vez, a cada passo de tempo (LIMA, 2004).

Para Gao (1996), a estrutura básica de uma RN Elman pode ser representada conforme a Figura 3.3. A RN de Elman possui uma estrutura de quatro camadas: camada de entrada, camada escondida, camada de contexto e camada de saída. Além disso, existem pesos de conexões ajustáveis para cada duas camadas vizinhas. A RN de Elman é geralmente considerada um tipo especial de RN *feedforward* com neurônio adicionais de memória com realimentação local. A realimentação da camada de contexto para a camada de entrada faz a RN de Elman ser sensível às entradas históricas o que é importante em problemas de previsão de séries temporais e identificação de sistemas dinâmicos.

Na RN de Elman, a camada de contexto armazena informação da camada oculta. As unidades de contexto são conectadas com as unidades escondidas de maneira que existe um peso de toda unidade de contexto para toda unidade na camada escondida. As unidades de camada de contexto têm função de ativação linear, sendo assim funcionam como memória, fixando o peso entre a camada escondida e a camada de contexto, possibilitando a utilização do algoritmo de retropropagação de erro.

As entradas da RN são: $u_{i,(k)}$, $y_{j(k)}$, $x_{i(k)}$ e as saídas de camada podem ser dadas pelas equações

$$o_c_i^{(k)} = rede_c_i^{(k)} \quad (3.5)$$

$$rede_h_j^{(k)} = \sum_{i=1}^M w_{1,i,j} x_i^{(k)} + \sum_{i=1}^L w_{3,i,j} o_c_i^{(k)} \quad (3.6)$$

$$o_h_j^{(k)} = f(rede_h_j^{(k)}) \quad (3.7)$$

$$y_j^{(k)} = \sum_{i=1}^L w_{2,i,j} o_{-} h_i^{(k)}. \quad (3.8)$$

onde f e g são funções com saídas lineares ou não-lineares da camada escondida e da camada de saída, respectivamente.

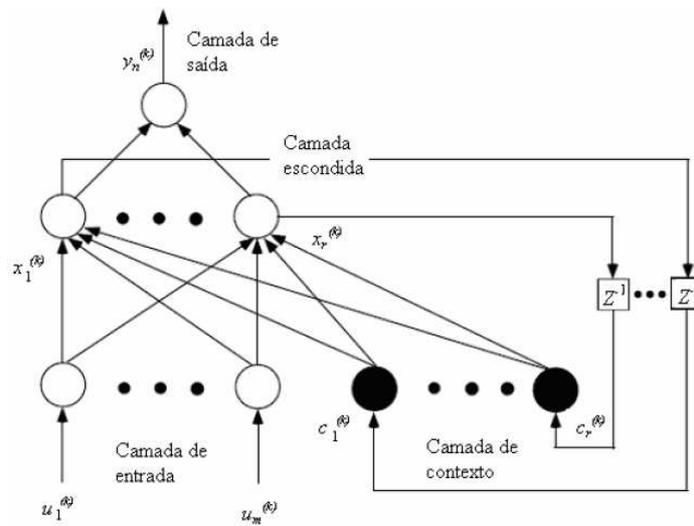


Figura 3.3: Rede parcialmente recorrente de Elman.

$w_{1_{ij}}$: peso que conecta o nó i da camada de entrada com o nó j na camada escondida;

$w_{2_{ij}}$: peso que conecta o nó i da camada escondida com o nó j na camada de saída;

$w_{3_{ij}}$: peso que conecta o nó i da camada de contexto com o nó j na camada escondida;

$rede_h_i$: entrada do nó i na camada oculta;

o_h_i : saída do nó i na camada oculta;

$rede_c_i$: entrada do nó i na camada de contexto;

o_c_i : saída do nó i na camada de contexto;

$f(x)$: função de ativação na camada escondida em $f(x) = \frac{1}{1 + e^{-ax}}$;

m, n, r : número de nós na camada de entrada, saída e camadas escondida, respectivamente;

$u_{i,(k)}, y_{j,(k)}$: entradas e saídas da RN de Elman, onde $i=1,2,\dots,m$, e $j=1,2,\dots,n$;

$x_{i,(k)}$: saída do nó escondido i , onde $i=1,2,\dots,r$;

$c_{i,(k)}$: saída do nó de contexto i , isto é, a saída do nó escondido i da última iteração;

Z^{-1} : operador de atraso unitário.

Devido a característica dinâmicas da RN de Elman provida por meio das conexões internas, ela não precisa apresentar o estado como entrada ou sinal do aprendizado. Esta é uma vantagem da RN de Elman em relação às RN *feedforward* estáticas.

3.4 Perceptron de múltiplas camadas

O MLP é uma RN *feedforward* é constituída por conjuntos de nós, os quais formam a camada de entrada da RN, uma (ou mais) camadas escondidas (intermediárias) e a primeira camada de saída. Desta forma, o MLP é capaz de resolver problemas complexos e não-linearmente separáveis (CASTRO, 2007; ROSELLI, 2007).

Haykin (2001) menciona que o MLP tem sido utilizado com sucesso para a solução de vários problemas envolvendo não-linearidades. O aprendizado do MLP é do tipo supervisionado. O algoritmo mais popular de aprendizado (treinamento) do MLP é denominado algoritmo de retropropagação do erro. Este algoritmo é baseado em uma regra de aprendizagem que “corrige” o erro durante o treinamento (HAYKIN, 2001). Entretanto para Maji (2007) e Pino *et al.* (2008) e Giordano *et al.* (2007) o MLP com aprendizado baseado no algoritmo de retropropagação do erro é o modelo mais utilizado em áreas de conhecimento, tais como reconhecimento de padrões, processamento de fala e na previsão de séries temporais.

3.4.1 Algoritmo de retropropagação do erro

Segundo Haykyn (2001), a RN-MLP é uma topologia importante de RNs pela sua capacidade de aproximação. É de conhecimento que o perceptron só pode resolver problemas separáveis linearmente ou linearmente independentes.

Os problemas não linearmente separáveis requerem que as RNs tenham uma representação intermediária apropriada dos padrões de entrada pela introdução de uma (ou mais) camada(s) oculta(s). Uma RN-MLP *feedforward* de uma única camada escondida é representada na Figura 3.4. Entretanto, Schiffmann *et al.* (1994) e Wilamowski *et al.* (2001) comentam que o algoritmo de retropropagação do erro é comumente utilizado no processo de aprendizado nas RNs-MLP. Entretanto, a RN-MLP possui uma lenta taxa de convergência.

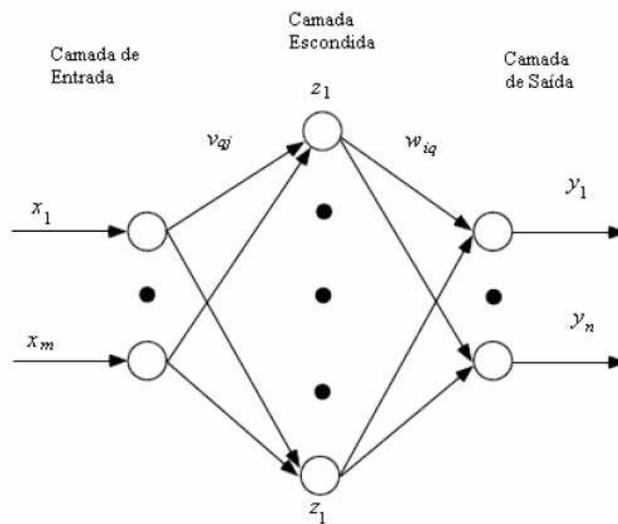


Figura 3.4: Representação de RN *feedforward* com uma camada escondida.

O algoritmo de aprendizagem por retropropagação do erro (do inglês, *error backpropagation*) é um método de estimação dos pesos sinápticos de uma RN-MLP. O algoritmo de retropropagação do erro foi desenvolvido por RUMELHART *et al.* (1986), apesar de ter sido proposto antes por WERBOS (1974), no entanto, com resultados pouco difundidos.

O algoritmo de retropropagação do erro utiliza pares de dados de entrada e saída

(desejada) para ajustar os pesos da RN-MLP, por meio de um mecanismo de correção de erro baseado na informação do erro médio quadrático de previsão, classificação ou modelagem. O treinamento através do algoritmo retropropagação do erro ocorre em duas etapas:

Fase de propagação: Esta fase é utilizada para definir a saída da RN-MLP para um dado padrão de entrada. Nesta fase, os pesos da RN-MLP são mantidos fixos. O fluxo segue no sentido de entrada para a saída da RN-MLP;

Fase de retropropagação: Esta fase utiliza a saída desejada e a saída calculada pela RN-MLP na fase de propagação, para a realização do ajuste dos pesos das conexões da RN-MLP. Nesta fase, o fluxo do sinal do erro é inverso daquele na fase de propagação, ou seja, o fluxo segue da camada de saída para a camada de entrada.

Na representação apresentada na Figura 3.4, v_{qj} é o peso conectando o nó j na camada de entrada com o nó q na camada escondida, e w_{iq} o peso que conecta o nó q da camada escondida com o nó i na camada de saída. Existem m , l e n nós na camada de entrada, camada oculta, e camada de saída, respectivamente. Assuma que $[x, d]$ representa um par de exemplo de treinamento; $d = \{x_1, x_2, \dots, x_m\}$ é o padrão (vetores) de entrada, e $d = \{d_1, d_2, \dots, d_m\}$ é a saída desejada. O nó q na camada oculta recebe a entrada e obtém a saída transformada, conforme calculado pela equação,

$$rede_q = \sum_{j=1}^m v_{qj} x_j. \quad (3.9)$$

Sendo assim:

$$Z_q = a(rede_q) = a \sum_{j=1}^m v_{qj} x_j, \quad (3.10)$$

onde $a(\cdot)$ é a função de ativação de um nó na RN-MLP. O cálculo da entrada do nó i na camada é representada pela equação,

$$rede_i = \sum_{q=1}^l w_{iq} z_q = \sum_{q=1}^l w_{iq} a \left(\sum_{j=1}^m v_{qj} x_j \right). \quad (3.11)$$

Entretanto, a saída correspondente é representada pela seguinte equação,

$$y_i = a(rede_i) = a \left(\sum_{q=1}^l w_{iq} Z_q \right) = a \left[\sum_{q=1}^l w_{iq} a \left(\sum_{j=1}^m v_{qj} x_j \right) \right]. \quad (3.12)$$

As equações (3.10) a (3.12) apresentam o procedimento de propagação do erro desde a entrada à saída da RN-MLP. O algoritmo básico de retropropagação do erro é baseado nestas equações e no seguinte procedimento de retropropagação do erro. A função de custo $E(W)$ para minimizar o erro da RN-MLP pode ser dado pela equação,

$$E(W) = \frac{1}{2} \sum_{i=1}^n (d_i - y_i)^2 = \frac{1}{2} \sum_{i=1}^n [d_i - a(rede_i)]^2 = \frac{1}{2} \sum_{q=1}^l [d_i - a \left(\sum_{q=1}^l w_{iq} Z_q \right)], \quad (3.13)$$

onde W representa os pesos das conexões v_{qj} e w_i . Os pesos w_i são utilizados pelo princípio da descida de encosta (*descent gradient*), sendo que,

$$\Delta w_{iq} = \eta \frac{\partial E}{\partial w_{iq}}, \quad (3.14)$$

onde η é a taxa de aprendizado da RN-MLP. Aplicando a regra da cadeia para $\frac{\partial E}{\partial w_{iq}}$, obtém-se,

$$\Delta w_{iq} = -\eta \frac{\partial E}{\partial y_i} \frac{\partial y_i}{\partial rede_i} \frac{\partial rede_i}{\partial w_{iq}} = \eta (d_i - y_i) a'(rede_i) z_q = \eta \delta_{oi} z_q, \quad (3.15)$$

onde a' é a derivada de primeira ordem da função a . Com base na equação (3.14), a definição

do termo geral de retropropagação δ_{oi} para a camada de saída pode ser dada pela equação,

$$\delta_{iq} = -\frac{\partial E}{\partial rede_i} = -\frac{\partial E}{\partial y_i} \frac{\partial y_i}{\partial rede_i} = (d_i - y_i)a(rede_i). \quad (3.16)$$

Para utilizar os pesos conectados a camada de entrada com a camada de saída, aplica-se a abordagem de descida de encosta e novamente a regra da cadeia, tal que

$$\Delta v_{qj} = \eta \frac{\partial E}{\partial y_{qj}} = -\eta \frac{\partial E}{\partial rede_q} \frac{\partial rede_q}{\partial v_{qj}} = -\eta \frac{\partial E}{\partial z_q} \frac{\partial z_q}{\partial rede_q} \frac{\partial rede_q}{\partial v_{qj}}. \quad (3.17)$$

Uma vez que z_q contribui com cada termo de erro da função de custo, gera-se,

$$\Delta w_{qj} = \eta \sum_{i=1}^n [(d_i - y_i)a(rede_i)w_{iq}] a(rede_q)x_j. \quad (3.18)$$

Substituindo a equação (3.15) em (3.17), obtém-se

$$\Delta w_{qj} = \eta \sum_{i=1}^n (\delta_{oi}w_{iq})a(rede_q)x_j = \eta \delta_{hq}x_j. \quad (3.19)$$

De forma similar a δ_{oi} , como erro geral de retropropagação para a camada escondida é dada pela equação,

$$\delta_{hq} = -\frac{\partial E}{\partial rede_q} = -\frac{\partial E}{\partial z_q} \frac{\partial z_q}{\partial rede_q} = a(rede_q) \sum_{i=1}^n \delta_{oi}w_{iq}, \quad (3.20)$$

onde o gradiente local δ_{hq} para o neurônio de saída. Comparando-se a equação (3.19) com a equação (3.15), conclui-se que δ_{hq} é determinado por δ_{oi} , que obtido a partir da camada de

saída para a camada escondida. Portanto, fica aparente que esses erros de retropropagação, δ_{hq} e δ_{oi} , são propagados através da camada de saída em direção a camada de entrada para a implementação do algoritmo de aprendizado de retropropagação do erro. O procedimento pode ser generalizado e aplicado as RNs com mais de uma camada oculta. Está é a versão do algoritmo de treinamento por retropropagação do erro baseado em padrões (SANTOS FILHO, 2008).

O método de aprendizado de retropropagação do erro foi descrito nesta dissertação, pois este método é o mais utilizado para o processo de aprendizagem de uma RN *feedforward*. Nesta dissertação são utilizados também o método de otimização de Levenberg-Marquardt para também a RN-MLP, o métodos de quase-Newton para a RN de Elman e para a RN-RBF é utilizado para o treinamento pelo método de descida em encosta.

3.4.2 Método quase-Newton

O algoritmo quase-Newton pertence à classe de algoritmos que implementam o método de Newton, mas não calculam as derivadas de segunda ordem, daí a origem do nome quase-Newton. Os algoritmos do tipo quase-Newton apresentam-se com melhor eficiência, pois são mais rápidos. Entretanto, estes algoritmos precisam armazenar a matriz Hessiana, das segundas derivadas (CARVALHO, 2005; DE ANDRADE *et al.*, 2001).

Os métodos quase-Newton são considerados eficientes. Entretanto, o seu armazenamento é de acordo com os requisitos computacionais de tamanho da estrutura da RN. Para RN com estruturas grandes, as exigências de memória para a utilização do algoritmo quase-Newton tornam impraticáveis. No entanto, para RN com algumas centenas de pesos, o algoritmo de quase-Newton é eficiente quando comparado com técnicas de gradiente conjugado e suas variantes (HAGAN e MENHAJ, 1994).

Segundo Haykin (2001), os métodos de quase-Newton são basicamente métodos de gradiente descritos pela equação de atualização

$$w(n+1) = w(n) + \eta(s)s(n), \quad (3.21)$$

onde o vetor de direção $s(n)$ é definido em termos de vetor gradiente $g(n)$ pela equação

$$s(n) = -S(n)g(n). \quad (3.22)$$

A matriz $S(n)$ é definida positiva e é ajustada a cada iteração de modo que o vetor de direção $s(n)$ aproxime à direção de Newton. Neste caso,

$$-\left(\frac{\partial^2 \mathcal{E}_{med}}{\partial w^2}\right)^{-1} \left(\frac{\partial \mathcal{E}_{med}}{\partial w}\right), \quad (3.23)$$

onde \mathcal{E}_{med} é dado pela média do erro médio.

Os métodos quase-Newton utilizam informações de segunda ordem acerca da superfície de erro, sem requerer o conhecimento da matriz Hessiana. O processo utilizado pelo algoritmo quase-Newton utiliza dois vetores iterativos sucessivos $w(n)$ e $w(n+1)$, juntamente com os respectivos vetores de gradiente $g(n)$ e $g(n+1)$. Considerando as seguintes equações,

$$q(n) = g(n+1) \cdot g(n) \quad (3.24)$$

e

$$\Delta w(n) = w(n+1) - w(n), \quad (3.25)$$

pode-se então obter a informação da curvatura usando as equação aproximada,

$$q(n) \equiv \left(\frac{\partial}{\partial w} g(n)\right) \Delta w(n). \quad (3.26)$$

Dado o valor de w incrementos de peso linearmente independentes

$\Delta w(0), \Delta w(1), \dots, \Delta w(w-1)$ e os pesos dos respectivos incrementos de gradiente $q(0), q(1), \dots, q(w-1)$, pode-se aproximar a matriz Hessiana pela seguinte equação,

$$H \equiv [q(0), q(1), \dots, q(w-1)] [\Delta w(0), \dots, \Delta w(w-1)]^T, \quad (3.27)$$

e a matriz inversa é apresentada pela seguinte equação,

$$H^{-1} \equiv [\Delta w(0), \Delta w(1), \dots, \Delta w(w-1)]^T [q(0), q(1), \dots, q(w-1)]^{-1}. \quad (3.28)$$

Quando a função de custo $\varepsilon_{med}(w)$ é quadrática, as equações (3.24) e (3.25) são exatas. Na classe mais popular de métodos quase-Newton, a matriz $S(n+1)$ é obtida a partir do seu valor prévio $S(n)$ e dos vetores $\Delta w(n)$ e $q(n)$, utilizando a equação (FLETCHER, 1987):

$$S(n+1) = S(n) + \frac{\Delta w(n)\Delta w^T(n)}{q^T(n)q(n)} - \frac{S(n)q(n)q^T(n)S(n)}{q^T(n)S(n)q(n)} + \xi(n)[q^T(n)S(n)q(n)]^{-1} [v(n)v^T(n)], \quad (3.29)$$

onde

$$v(n) = \frac{\Delta w(n)}{\Delta w^T(n)\Delta w(n)} - \frac{S(n)q(n)}{q^T(n)S(n)q(n)}. \quad (3.30)$$

e

$$0 \leq \xi(n) \leq 1, \text{ para todo } n. \quad (3.31)$$

O algoritmo de quase-Newton é iniciado com a matriz definida positiva arbitrária $S(n)$, onde a forma particular do algoritmo é parametrizada de acordo com a definição de $\eta(n)$, tal que:

Caso $\xi(n) = 0$ para todo n , obtém-se o algoritmo de Davidson, Fletcher, Powell (DFP), sendo este historicamente o primeiro algoritmo de quase-Newton;

Caso $\zeta(n) = 1$ para todo n , obtém-se o algoritmo BFGS (Broyden, Fletcher, Goldfarb, Shanno), sendo esta considerada a uma forma eficiente do método de quase-Newton (FLETCHER, 1987).

3.4.3 Método de otimização Levenberg-Marquardt

O algoritmo de Levenberg-Marquardt foi introduzido para o treinamento de RNs *feedforward*, visando obter uma melhor taxa de convergência que o clássico método de RN-MLP retropropagação do erro que possui uma lenta taxa de convergência.

O algoritmo de Levenberg-Marquardt pode ser visto como uma modificação do método de quase-Newton. O algoritmo é considerado um método mais rápido para o aprendizado de RNs *feedforward* que o algoritmo de retropropagação do erro (DA SILVA, 2006), desde que a RN possua uma quantidade moderada de pesos sinápticos a fim de tornar o problema computacionalmente tratável. Este algoritmo baseia-se na determinação da derivada de segunda ordem de erro quadrático em relação aos pesos, diferindo neste aspecto do algoritmo de retropropagação do erro. Além disso, o método Levenberg-Marquardt converge com menos iterações. Entretanto, o método de Levenberg-Marquardt requer mais cálculos por iteração, isso se dá pelo cálculo das matrizes inversas (SANTOS, 2008; CHAN, 1996).

Suratgar *et al.* (2005), apresentaram modificações no algoritmo Levenberg-Marquardt, com um melhor desempenho no processo de aprendizagem de uma RN-MLP e reduzindo a quantidade de oscilações no processo de aprendizagem.

Segundo De Oliveira (2007), Levenberg (1944) propôs um algoritmo baseado na regra de atualização que pode ser dada pela seguinte equação,

$$w_{t+1} = w_t - [H + \lambda I]^{-1} \nabla f(w_t), \quad (3.32)$$

onde I é a matriz de identidade, $\nabla f(w)$ é uma função linear e o H é a matriz Hessiana calculada em w_t . Se o erro decresce após uma atualização, isso implica na suposição de

forma quadrática para $f(w)$, onde é reduzido o λ para reduzir a influencia de descida de encosta.

O algoritmo proposto por Leavenberg é basicamente composto por 4 passos. No primeiro passo, é realizado uma atualização descrita pela equação (3.32). No segundo passo avalia-se o erro no novo vetor de parâmetros. No terceiro passo é verificado se o erro diminuiu como resultado da atualização, e então retorna-se ao passo anterior e aumenta-se o λ de um determinado fator, usualmente um fator de 10, e após retornar-se para o primeiro passo. No último passo, é avaliado se o erro diminuiu como resultado da avaliação, e então aceitar o passo, ou seja, manter os pesos em seus novos valores e diminui-se o λ de um determinado fator.

O algoritmo de Levenberg-Marquardt tem a desvantagem de que se o valor de λ é alto, a matriz Hessiana calculada não é utilizada totalmente. Pode-se obter vantagem do uso da segunda derivada, em tais casos, pelo escalonamento de cada componente do gradiente de acordo com a curvatura. Neste caso, resultando em acentuados movimentos ao longo das direções onde o gradiente é menor. Neste contexto, Marquardt (1963) sugeriu substituir o valor da matriz identidade I em (3.32) pela diagonal da matriz Hessiana, resultando na regra de atualização de Levenberg-Marquardt, dada pela seguinte equação,

$$w_{t+1} = w_t - [H + \lambda \text{diag}[H]]^{-1} \nabla f(w_t). \quad (3.33)$$

Para De Oliveira (2007), os problemas de minimização utilizado nas RNs frequentemente são mal e considerados, isto torna o problema de minimização de difícil resolução.

3.5 Rede neural de função de base radial (RN-RBF)

As RNs-RBF possuem aprendizado supervisionado ou híbrido de supervisionado e não supervisionado, e teve sua origem na área de Análise Numérica com o propósito da aproximação de funções (HARDY, 1971).

Uma RN-RBF é uma RN *feedforward* com uma única camada oculta e uma camada de saída (CELIKOGU e CIGIZOGLU, 2007).

O aprendizado de uma RN-RBF é equivalente a ajustar uma superfície não-linear ao conjunto de dados, em um espaço multi-dimensional, considerando o critério estatístico ou o índice de desempenho. O processo de generalização equivale a usar esta superfície multi-dimensional para interpolar outros pontos que não pertençam ao conjunto de treinamento, mas que estejam em sua vizinhança (HAYKIN, 2001; CASTRO, 2001). A justificativa para tal é conhecida como Teorema de Cover. Em síntese, um problema complexo de classificação de padrões de ser linearmente separável quando analisado em um espaço de muitas dimensões (COVER, 1965).

Haykin (1994) menciona que a RN-RBF tornou-se um dos tipos mais poderosos dentre as RNs, isso devido a sua capacidade de lidar bem com aplicações em aproximação, classificação de padrões e agrupamento de dados.

Maqsood e Abraham (2006) apresentaram o aprendizado de dois métodos básicos para configurar uma RN-RBF. Um destes métodos é a otimização conjunta de todos os parâmetros da RN-RBF. Este método normalmente resulta em uma apropriada aproximação. No entanto, possui alguns inconvenientes, tais como o alto custo computacional e a presença de muitos parâmetros ajustáveis. O segundo método de aprendizado para a RN-RBF consiste em dividir o aprendizado de uma RN-RBF em duas etapas. A primeira etapa é selecionar os centros μ em termos agrupamentos (*clusters*) por meio de um algoritmo não-supervisionado, por exemplo o algoritmo *k*-médias. Após, escolhe-se as aberturas σ pelo algoritmo do vizinho mais próximo. Na segunda etapa os pesos da camada de saída são atualizadas, mantendo-se os μ fixos. O algoritmo de duas etapas possui rápida taxa de convergência e demanda pequeno custo computacional.

Uma RN-RBF é constituída por três camadas: uma camada de entrada de neurônios, onde é realizada a alimentação dos padrões (vetores) de entrada da RN-RBF, uma camada oculta e uma camada de saída, conforme apresentado na Figura 3.5.

Em uma RN-RBF, as unidades da camada oculta fornecem um conjunto de “funções” que consistem de uma “base” (que dão o nome a RN-RBF) que consiste em uma base arbitrária para os padrões (vetores) de entrada, quando eles são expandidos sobre o espaço da camada oculta. Cada uma dessas funções concentra-se em uma coordenada particular do espaço multi-dimensional dos pontos que compõem o espaço de dados de

entrada. Cada uma destas coordenadas particulares caracteriza-se por definir o centro de uma (entre várias possíveis) região de maior aglomeração de pontos ou grupos (*clusters*), do espaço de dados de entrada (HAYKIN, 1996; BORS, 2001).

A consequência imediata do uso de funções de ativação da base radial na RN-RBF está na forma como as entradas são processadas pelos neurônios da camada intermediária. Ao invés da ativação interna de cada neurônio da camada intermediária se dar pelo emprego do produto escalar entre os padrões de entrada (vetores) e o vetor de pesos, como no caso do MLP, ela é obtida a partir de uma norma ponderada da diferença entre ambos os vetores.

3.5.1 Arquitetura de uma RN-RBF

A Figura 3.5 apresenta um exemplo de arquitetura de uma RN-RBF. A RN-RBF é composta por uma camada de nós fonte (que conectam a RN ao ambiente externo), a qual é apresentado o vetor de entrada $\underline{u}(n) \in \mathcal{R}^M$. Na camada intermediária, a distância entre o vetor de entrada e o centro da função de base radial associada, constitui os componentes inerentes à camada oculta e esta distancia é usada na função φ .

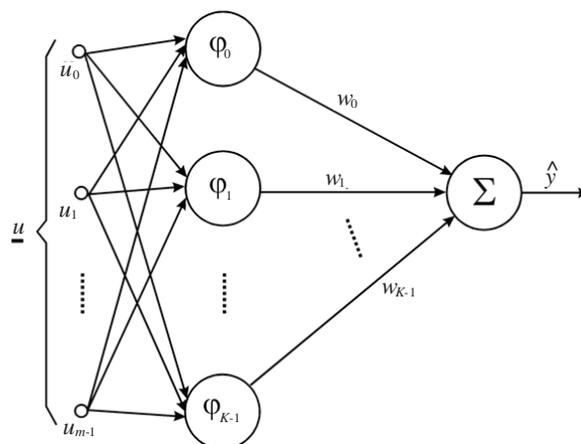


Figura 3.5: Arquitetura de uma RN-RBF.

Na Figura 3.5, o mapeamento não linear é expresso por funções de ativação Gaussianas, e representando e isso pode ser expresso por

$$\varphi_k(n) = \varphi_k(\underline{u}(n), \underline{t}_k(n), \sigma_k^2(n)) = \exp\left[-\frac{1}{\sigma_k^2(n)} \|\underline{u}(n) - \underline{t}_k(n)\|^2\right], \quad (3.34)$$

onde $\underline{u}(n) \in \mathfrak{R}^M$ representa o vetor de entrada u no instante n , $\underline{t}_k(n) \in \mathfrak{R}^M$ representa o vetor centro da k -ésima função de base radial $k = 0, 1, \dots, K-1$, K é o número de funções de base radial, e $\sigma_k^2(n) \in \mathfrak{R}$ é a variância associada a cada uma das funções no instante de tempo n .

A camada de saída da RN-RBF é formada por neurônios, geralmente com função de ativação linear. A camada de saída é composta pelas combinações lineares das saídas das funções de ativação de base radial. A saída y de uma RN-RBF é obtida a soma das saídas composta por funções de ativação da camada oculta, geralmente funções Gaussianas, estas ponderadas pelos seus respectivos pesos sinápticos w_k , de tal forma que a combinação linear é representada na seguinte equação,

$$\hat{y}(n) = \sum_{k=0}^{K-1} w_k \varphi_k(\underline{u}, \underline{t}_k, \sigma_k^2). \quad (3.35)$$

Na equação (3.35), o termo $\varphi_k(\underline{u}, \underline{t}_k, \sigma_k^2)$ é a k -ésima função de base radial. Note que φ_k calcula o quadrado da distância Euclidiana $D_k^2 = \|\underline{u} - \underline{t}_k\|^2$ entre um vetor de entrada \underline{u} e o centro \underline{t}_k da k -ésima função de base radial. O sinal de saída produzido pelo k -ésimo neurônio escondido é, portanto, devido à função $\exp(\cdot)$ e ao operador $(\cdot)^2$, uma função não-linear da distância D_k . O fator de escala w_k representa uma ponderação do caminho que conecta o k -ésimo neurônio escondido ao nó de saída da rede. À equação (3.35) pode, em alguns casos, ser ainda acrescido de um termo constante de polarização ou *bias* (HAYKIN, 1996).

A transformação não-linear acima referida é definida pelo conjunto de funções de base radial φ_k e a transformação linear é definida pelo conjunto de pesos w_k , $k = 0, 1, \dots, K-1$.

O mapeamento da entrada e saída de uma RN-RBF com função de ativação Gaussiana na camada oculta é semelhante à técnica estatística denominada de Mistura de Modelos (*Mixture Models*), que são misturas de distribuição de probabilidades. Em

particular, os Modelos de Misturas de Gaussianas têm sido usados como modelos em uma variedade de aplicações onde os dados de interesse provêm de dois ou mais vetores, misturadas entre si, com parâmetros estáticos distintos. A resposta φ_k do neurônio k da camada oculta de uma rede RN-RBF representa a densidade probabilística de u dado o centro \underline{t}_k , então é, $\varphi_k(\underline{u}|\underline{t}_k)$. O coeficiente de w_k representa a probabilidade *a priori* de u no contexto da densidade condicional $\varphi_k(\underline{u}|\underline{t}_k)$. Sendo assim o conjunto K densidades probabilísticas condicionais modela a função de densidade de probabilidade, respectivamente, do mecanismo estático subjacente que gerou os dados, sendo que o modelo definido através de $\sum_{k=0}^{K-1} w_k \varphi_k(\underline{u}|\underline{t}_k)$. Neste sentido, a RN-RBF pode ser referida como um Estimador *Bayesiano*.

O procedimento para a implementação de uma RN-RBF compreende a determinação, por meio de um processo de aprendizagem, dos valores adequados aos parâmetros livres da RN-RBF, que são as variâncias σ_k^2 , os centros \underline{t}_k e os pesos sinápticos w_k . O aprendizado consiste em determinar estes parâmetros de tal forma que, dado um conjunto de estímulos \underline{u} na entrada, as saídas y se aproximem os valores presentes no conjunto de valores desejado.

A camada oculta, sempre única na RN-RBF, possui funções de ativação de base radial é responsável pela transformação não-linear do espaço das entradas no espaço da camada escondida, geralmente de alta dimensionalidade (geralmente, quanto maior a dimensão do espaço escondido melhor sua aproximação).

Capítulo 4

APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Neste capítulo têm-se os resultados obtidos a partir dos dados descritos nas subseções 2.3 e 2.4. Este capítulo encontra-se estruturado da seguinte forma. A coleta e tratamento aplicado aos dados utilizados para a resolução do problema de classificação pelas RNs, são descritos inicialmente. Após, os resultados de classificação obtidos pelas RNs são representados e discutidos.

4.1 Coleta e tratamento dos dados

Os dados utilizados nesta dissertação estão disponíveis em um banco de dados Oracle 10g (versão 10.2.0.3), rodando em um processador Intel® Pentium® 4-3.0 HT GHz com 2 GB de memória. Para a implementação computacional das RNs foi utilizado o *software* MATLAB (versão 7.0), da Mathworks, utilizando a caixa de ferramentas de RNs 6.0.2 (MATHWORKS, 2008).

Para uma classificação eficiente, os padrões de entrada (vetores) devem receber valores obedecendo a determinados limites. Os valores dos padrões de entrada (vetores) variam em uma ordem de magnitude grande, podendo inclusive inviabilizar o uso de alguns modelos de RNs se não forem adequadamente padronizados. Devido a estas preocupações, os valores de entrada foram normalizados entre 0 e 1.

As variáveis utilizadas nesta dissertação estão descritas na Tabela 2.1 e como foram obtidas essas variáveis que esta descrito nas subseções 2.2 e 2.3.

4.2 Configuração dos métodos

Para os processos de estimação e validação das RNs foram gerados 30 experimentos de geração aleatória com diferentes sementes para a geração de números aleatórios usando uma distribuição uniforme, conforme descrito na subseção 2.4. Na execução dos experimentos são verificados os valores do mínimo, máximo, a média e o desvio padrão das variáveis “Acerto de Clientes Adimplentes” e “Acerto de Clientes Inadimplentes” tanto para o processo de estimação quanto para o processo de validação dos resultados obtidos pelas RNs.

Neste contexto, serão executados todos os experimentos para todas as RNs testadas, variando o número de neurônios da camada oculta. Os resultados serão apresentados em tabelas. Para a análise de qual das arquiteturas das RNs obteve um melhor desempenho serão verificados os valores da média e do desvio padrão obtido. Sendo a média próxima de 100% e o desvio padrão próximo de 0.

A RN-MLP foi configurada com 50 épocas (iterações) para a finalidade do processo de estimação e validação. Para o processo de treinamento foi utilizado o método de otimização de Levenberg-Marquardt sendo este um método de rápida convergência (DA SILVA, 2006; SANTOS, 2008).

A RN de Elman também foi configurada com 50 épocas (iterações) para o processo de estimação e validação. Entretanto, para o processo de treinamento a RN de Elman difere da RN-MLP, pois as conexões da RN de Elman foram ajustadas por um algoritmo quase-Newton do tipo BFGS. Para a RN-RBF foi utilizado para o processo de estimação e validação o algoritmo de descida de encosta.

Para as arquiteturas de RNs validadas nesta dissertação são adotados 17 neurônios para a camada de entrada e os neurônios da camada oculta variam de 2 a 10. Para a saída das RNs é utilizado 1 neurônio na camada de saída.

4.3 Processo de aprendizado e validação das RNs

Foram executados os processos de estimação e validação para as arquiteturas de RNs apresentadas nesta dissertação. Para comparar quais das arquiteturas obteve um melhor desempenho na classificação dos dados descritos nas subseções 2.3 e 2.4, os resultados serão apresentados nas tabelas 4.1 a 4.8, onde são apresentados os valores mínimo, máximo, média e desvio padrão das variáveis “Acerto de Clientes Adimplentes” e “Acerto de Clientes Inadimplentes”, sendo a variável “Acerto de Clientes Inadimplentes” dependente da variável “Acerto de Clientes Adimplentes”.

Os resultados descritos nas Tabelas 4.1 a 4.6 são os resultados do processo de estimação e validação das arquiteturas de RNs, tendo como objetivo, saber qual a quantidade de neurônio na camada oculta das arquiteturas obtiveram uma melhor média com o menor desvio padrão.

Uma análise comparativa inicial entre os modelos de RNs apresentou um desempenho superior nos processo de validação e estimação para a RN-MLP. Os resultados obtidos nas Tabelas 4.1 e 4.2 são apresentados em termos de percentuais.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	100	100	100	0	86,1464	95,3151	93,8388	1,5364
3	99,9752	100	99,9983	0,0063	86,1464	95,3151	93,8125	1,5290
4	99,9752	100	99,9968	0,0084	92,9594	95,3151	94,0135	0,5031
5	99,8255	100	99,9933	0,0320	91,1847	95,2036	93,8715	0,6865
6	100	100	100	0	92,8370	95,3151	93,9654	0,5329
7	100	100	100	0	92,9785	95,3151	94,0251	0,4884
8	100	100	100	0	92,8449	95,0363	94,0132	0,4812
9	100	100	100	0	92,7520	95,1478	93,9882	0,5127
10	100	100	100	0	92,7520	95,3151	93,9487	0,5094

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.1: Resultado de treinamento da RN-MLP.

Nota-se pela Tabela 4.1 que no processo de estimação da RN-MLP para a classificação dos clientes adimplentes e inadimplentes, o processo de estimação da RN-MLP apresentou melhor desempenho quando a RN-MLP foi configurada com 8 neurônios na camada oculta. A RN-MLP obteve 100% de acerto para os clientes adimplentes, com a utilização de 8 neurônios na camada oculta, com o resultado da RN foi obtido uma média de 100% e o desvio padrão foi de 0. Entretanto, a variável “Acerto de Clientes Inadimplentes” quando a RN foi configurada com 8 neurônios na camada oculta da RN-MLP obtiveram uma média de 94,0132 e o desvio padrão de 0,4812.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	99,9857	100	99,9992	0,0027	85,8847	94,0899	93,7323	1,4829
3	99,9667	100	99,9932	0,0104	85,8847	94,0899	93,6813	1,4757
4	99,9444	100	99,9889	0,0157	93,5229	93,5229	93,8858	0,1458
5	99,8666	100	99,9872	0,0257	91,7329	94,0544	93,7732	0,4537
6	99,9523	100	99,9929	0,0136	92,8795	94,0432	93,8309	0,2161
7	99,9619	100	99,9908	0,0125	93,5213	94,0544	93,8853	0,1304
8	99,9635	100	99,9910	0,0110	93,4923	94,0304	93,8246	0,1501
9	99,9603	100	99,9895	0,0120	93,5339	94,0133	93,8284	0,1164
10	99,9333	100	99,9923	0,0155	93,5001	94,0201	93,8097	0,1425

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.2: Resultado de validação da RN-MLP.

A Tabela 4.2 apresenta os dados obtidos pelo processo de validação da RN-MLP. Conforme apresentado na Tabela 4.1, onde a RN-MLP obteve um melhor desempenho quando foi configurada com 8 neurônios na camada oculta para o processo de estimação, pode-se observar que para o processo de validação da RN-MLP obteve o mesmo desempenho obtido pelo processo de estimação com a RN configurada com 8 neurônios na camada oculta.

Nota-se que para a validação da RN-MLP configurada com 2 neurônios na camada oculta, a RN-MLP apresentou 99,9992 de média e um desvio padrão de 0,0027. Entretanto, para a variável “Acerto de Clientes Inadimplentes” obteve uma média de 93,7323 e um desvio padrão de 1,4829. As colunas “Acerto de Clientes Inadimplentes” e “Acerto de

Clientes Adimplentes” são dependentes, analisando os valores da média e desvio padrão de clientes inadimplentes e adimplentes, observa-se que a RN-MLP configurada com 8 neurônio na camada oculta poderá obter melhor desempenho.

Nas Tabelas 4.3 e 4.4 são apresentados os resultados da estimação e validação da RN de Elman.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	99,3002	100	99,9020	0,1674	91,0220	94,7016	93,1367	0,7554
3	98,9285	100	99,7531	0,2866	90,8267	94,5901	92,7250	0,6986
4	99,1984	99,9755	99,7606	0,1943	91,4049	94,3670	92,4112	0,8131
5	99,4981	100	99,8333	0,1035	90,1302	94,4228	92,5686	0,8814
6	99,6769	100	99,8466	0,0763	91,6195	94,4785	92,9823	0,5083
7	99,1785	100	99,7933	0,2345	91,9618	94,1078	93,1082	0,5244
8	99,5249	100	99,9002	0,1109	90,9966	93,1958	92,2758	0,5183
9	99,0682	100	99,7541	0,2475	90,5640	94,0881	92,5866	0,7537
10	99,2793	100	99,8006	0,2008	90,8504	93,5862	92,2066	0,6761
		Média	99,8160			Média	92,6668	

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.3: Resultado de treinamento da RN de Elman.

Na estimação da RN de Elman para classificar os clientes adimplentes e inadimplentes, a RN de Elman apresentou um melhor desempenho quando a RN foi configurada com 6 neurônios na camada oculta. A RN de Elman obteve uma média de 99,8466% e o desvio padrão foi de 0,0763. Entretanto, a variável “Acerto de Clientes Inadimplentes” que utilizou 6 neurônios na camada oculta da RN de Elman obtiveram uma média de 92,9823 e um desvio padrão de 0,5083.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	99,4520	100	99,8943	0,1431	90,5199	93,6151	93,0095	0,7583
3	99,0184	99,9984	99,7575	0,2652	91,4049	92,9953	92,6142	0,4433
4	98,9684	99,9873	99,7359	0,2437	90,9101	92,9450	92,2410	0,6266
5	99,2745	100	99,8180	0,1486	90,4444	92,9343	92,4136	0,6680
6	99,7206	99,9873	99,8283	0,0791	92,5454	92,9438	92,8563	0,0817
7	99,2456	100	99,7992	0,2189	92,0138	93,8777	93,0121	0,4020
8	99,5554	100	99,8876	0,1298	91,3104	92,8008	92,0878	0,4087
9	99,1443	100	99,7258	0,2661	90,4247	93,0013	92,4085	0,6903
10	99,2915	100	99,8042	0,1894	90,8618	92,8767	92,0731	0,5350

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.4: Resultado da validação da RN de Elman.

Conforme apresentado na Tabela 4.4, onde a RN de Elman obteve um melhor desempenho com 6 neurônios na camada oculta para o processo de estimação, pode-se observar que para o processo de validação a RN continua obtendo o mesmo desempenho obtido pelo processo de estimação configurado com 6 neurônios na camada oculta. A RN de Elman configurada com 6 neurônios na camada oculta teve uma média de 99,8283 e um desvio padrão de 0,0791. Entretanto, para a variável “Acerto de Clientes Inadimplentes” obteve uma média de 92,8563 e um desvio padrão de 0,0817. Comparando os resultados obtidos, verificou-se que a RN de Elman não obteve superioridade nos processo de estimação e validação. Pode-se observar que a RN-MLP obteve resultado superior, tanto na estimação quanto na validação, embora a RN de Elman possa ser utilizada com uma excelente solução para a classificação de dados.

Embora a RN de Elman tenha apresentado desempenho eficiente quanto a classificação de clientes, a RN de Elman possui unidades de contexto para armazenar os resultados das camadas ocultas, isso faz com que a quantidade de conexões entre os neurônios cresça rapidamente quando são adicionados neurônios a essa camada e isso exige mais tempo para o processo de estimação da RN de Elman (SANTOS FILHO, 2008).

A Tabela 4.5 apresenta os dados obtidos pela fase de estimação da RN-RBF.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	99,9749	100	99,9958	0,0095	88,3918	91,6000	90,6098	0,6279
3	99,9253	100	99,9896	0,0191	79,8415	82,9704	81,5485	0,8337
4	100	100	100	0	86,6365	90,2315	88,4615	0,7775
5	99,9250	100	99,9893	0,0221	87,1461	90,4317	88,7927	0,7636
6	99,9500	100	99,9926	0,0161	87,6984	90,7727	89,3220	0,8561
7	99,9002	100	99,9842	0,0289	87,8873	91,0466	89,9032	0,7811
8	99,8747	100	99,9875	0,0277	88,3918	91,3714	90,2231	0,7254
9	99,9501	100	99,9917	0,0177	88,6183	91,5226	90,4202	0,7110
10	99,9749	100	99,9958	0,0095	88,3918	91,6000	90,6098	0,6279

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.5: Resultado de treinamento da RN-RBF.

No processo de estimação da RN-RBF para classificar os clientes adimplentes e inadimplentes, onde a RN-RBF apresentou um melhor desempenho quando a RN-RBF foi configurada com 4 neurônios na camada oculta. A RN-RBF obteve 100% de acerto para os clientes adimplentes, com a utilização de 4 neurônios na camada oculta, a RN teve uma média de 100% e o desvio padrão foi de 0. Entretanto, a variável “Acerto de clientes inadimplentes” que utilizou 4 neurônios na camada oculta obtiveram uma média de 88,4615 e um desvio padrão de 0,7775. Comparando o resultado da estimação e validação obtido na Tabela 4.1 com a Tabela 4.3, verificou-se a superioridade significativa na estimação da RN-MLP utilizando algoritmo de treinamento de Levenberg-Marquardt.

A Tabela 4.6 apresenta os dados obtidos pelo processo de validação da RN-RBF.

N	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
2	99,9190	100	99,9842	0,0269	91,3823	89,3597	90,4671	0,4543
3	99,9412	100	99,9872	0,0168	80,8195	82,4232	81,6130	0,5180
4	99,9762	100	99,9981	0,0054	87,5649	89,7123	88,4868	0,6109
5	99,9363	100	99,9898	0,0165	88,0125	90,2915	88,8056	0,6576
6	99,8920	100	99,9861	0,0256	87,5193	90,4172	89,3081	0,7304
7	99,8491	100	99,9744	0,0404	88,1832	90,4991	89,8324	0,5436
8	99,8809	100	99,9840	0,0294	89,1475	91,1875	90,1588	0,3858
9	99,9047	100	99,9817	0,0303	89,3201	90,9385	90,3424	0,3935
10	99,9190	100	99,9842	0,0269	89,3597	91,3823	90,4671	0,4543

Convenção: N = Quantidade de neurônio na camada oculta.

Tabela 4.6: Resultado de validação da RN-RBF.

Conforme apresentado na Tabela 4.5, onde a RN-RBF obteve um melhor desempenho com 4 neurônios na camada oculta para o processo de estimação, pode-se observar que para o processo de validação a RN-RBF obteve o mesmo desempenho obtido pelo processo de estimação configurado com 4 neurônios na camada oculta. A RN-RBF apresentou melhor desempenho quando foi configurada com 4 neurônios na camada oculta a média de 99,9981 e um desvio padrão de 0,0054, entretanto para a variável “Acerto de Clientes Inadimplentes” obteve uma média de 88,4868 e um desvio padrão de 0,6109.

Quando comparado os resultados de estimação e validação das RN-MLP, RN de Elman e RN-RBF, pode-se observar que os resultados apresentados nas Tabelas 4.5 e 4.6 obtidos pela RN-RBF possui desempenho superior quando os resultados são comparados com os resultados obtidos pela RN de Elman. Entretanto, quando os resultados são comparados com os resultados obtidos pela RN-MLP, a RN-MLP possui um melhor desempenho na estimação e classificação conforme os resultados foram apresentado nas Tabelas 4.1 e 4.2.

As Tabelas 4.1 e 4.2, respectivamente, mostram os resultados para os processo de estimação e validação, a RN-MLP foi treinada com o algoritmo de Levenberg-Marquardt para a execução da classificação dos clientes, pode-se adotar como padrão 8 neurônios na camada oculta da RN-MLP, onde obteve uma média de 100% e desvio padrão de 0 na

estimação e a validação configurado com 8 neurônios na camada oculta a RN-MLP obteve média de 99,9910% e desvio padrão de 0,0110.

De uma forma geral, a RN-RBF apresentou resultados satisfatórios. Para a classificação dos clientes utilizando a RN-RBF com algoritmo de descida de encosta, observando as Tabelas 4.5 e 4.6 pode-se adotar como padrão a quantidade de 4 neurônios para os processos de estimação e validação. A RN-RBF pode ser utilizada com sucesso no processo de classificação devido seu poder de generalidade e sua capacidade de aprendizado ser relativamente rápida (HUTCHINSON, 1994).

As Tabelas 4.3 e 4.4, respectivamente, apresentam resultados positivos na classificação dos clientes. Entretanto, os resultados obtidos pelas RNs RN-MLP e RN-RBF são superiores aos apresentados pela RN de Elman.

A Tabela 4.7 apresenta os resultados do processo de aprendizagem das RNs validadas. Para os dados apresentados na Tabela 4.7 foram consolidados os resultados do treinamento para cada uma das arquiteturas de RNs. Na apresentação dos resultados na Tabela 4.7 foram obtidos os valores mínimo, máximo, média, e desvio padrão, quantos aos resultados obtidos no processo de classificação para as arquiteturas de RNs os neurônios da camada oculta variaram de 2 até 10.

RN	Acerto de clientes adimplente (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
MLP	99,8255	100	99,9986	0,0114	86,1464	94,7016	93,9421	0,8522
Elman	98,9285	100	99,8160	0,1974	90,1302	94,7016	92,6668	0,7574
RBF	99,8747	100	99,9915	0,0197	79,8415	91,6000	88,3755	2,9917

Tabela 4.7: Resultado de treinamento das RNs.

Verificando a Tabela 4.7 pode-se observar que a RN-MLP com o processo de estimação baseado no método de otimização de Levenberg-Marquardt obteve uma média de 99,9986, onde a RN-MLP apresentou uma média acima das médias apresentada pelas RN de Elman 99,8160 e o valor da média da RN-RBF é de 99,9915.

RN	Acerto de clientes adimplentes (em %)				Acerto de clientes inadimplentes (em %)			
	Mínimo	Máximo	Média	Desvio Padrão	Mínimo	Máximo	Média	Desvio Padrão
MLP	99,8666	100	99,9917	0,0146	85,8847	94,0899	93,8057	0,7164
Elman	98,9684	100	99,8057	0,2024	90,4247	93,8777	92,5240	0,6432
RBF	99,8491	100	99,9859	0,0260	80,8195	91,3823	88,3642	2,8837

Tabela 4.8: Resultado de validação das RNs.

De uma forma geral, com os resultados obtidos na Tabela 4.8, pode-se notar tanto no processo de estimação quanto no processo de validação que a RN-MLP obteve melhor média. A média para o processo de validação da RN-MLP é de 99,9917, onde o processo de validação da RN-MLP apresentou uma média acima da média apresentada pela RN de Elman 99,8057 e o valor da média da RN-RBF que é de 99,9859. No entanto, observa-se que os resultados são próximos.

Verificando as Tabelas 4.1 a 4.8 pode se observar que as arquiteturas descritas nesta dissertação são eficientes na classificação de clientes inadimplentes e adimplentes.

Capítulo 5

CONCLUSÃO E RECOMENDAÇÕES PARA TRABALHOS FUTUROS

Neste capítulo é apresentada as conclusões desta dissertação, analisando o desempenho das RNs para a classificação de clientes da empresa DM-Inf S/A. Além disso, Apresentam-se também os possíveis trabalhos futuros vinculados a esta dissertação.

5.1 Conclusão

O objetivo principal desta dissertação foi apresentar métodos baseados em diferentes RNs para realizar uma tarefa de classificação de dados.

No entanto, para que as RNs tenham um desempenho aceitável na realização de determinada classificação, os parâmetros e a arquitetura das RNs devem ser estudadas com critério e cuidado, até que se consiga atingir resultados promissores de classificação.

Esta dissertação apresentou um estudo comparativo entre os métodos baseados nas seguintes RNs: i) a RN-MLP com treinamento usando otimização pelo método Levenberg-Marquardt; ii) a RN de Elman usando método de quase-Newton do tipo BFGS; e iii) a RN-RBF treinada pelo algoritmo de descida de encosta na classificação das variáveis, estas mencionadas nas subseções 2.3 e 2.4.

Com base nos resultados apresentados no capítulo 4, nota-se que as RNs são eficientes na classificação de sistemas não-lineares com variáveis de entrada e saída. No entanto, os resultados obtidos são dependentes da arquitetura da RN, ou seja, o número de entradas da RN, o número de neurônios na camada oculta e método de treinamento utilizando no modelo de RNs.

Outro comentário em relação aos resultados é o desempenho superior da RN-MLP sobre as RNs de Elman e RN-RBF na classificação. Pode-se observar nas tabelas 4.7 e 4.8 onde foi apresentado uma análise do desempenho da classificação da RN-MLP, foram analisados a média e o desvio padrão das variáveis “Acerto de clientes Adimplentes” e “Acerto de Clientes Inadimplentes”. Nota-se que a RN-MLP com algoritmo de treinamento Levenberg-Marquardt apresentou um melhor desempenho na classificação que os modelos de RNs de Elman e RN-RBF.

Entretanto, a Tabela 4.1 são apresentados os valores do mínimo, máximo, média e desvio padrão das variáveis “Acerto de Clientes Adimplentes” e “Acerto de Clientes Inadimplentes”. Pelos resultados obtidos tanto na estimação quanto na validação, a RN-MLP obteve um melhor desempenho da variável média e na variável de desvio padrão que as outras RNs analisadas nesta dissertação. A RN-MLP obteve um melhor desempenho que o resultado apresentado na Tabela 4.3 pela RN de Elman e na Tabela 4.5 pela RN-RBF. Na fase de estimação conforme apresentado na Tabela 4.1, a RN-MLP obteve uma melhor média quando a RN foi configurada com 8 neurônios na camada oculta.

Para a fase de validação da Tabela 4.2 é apresentado um melhor desempenho na variável média e na variável de desvio padrão das variáveis “Acerto de Clientes Adimplentes” e “Acerto de Clientes Inadimplentes” que a média e desvio padrão apresentados na Tabela 4.4 onde foi utilizado a RN de Elman e para a Tabela 4.6 da RN-RBF. Neste caso, a RN-MLP obteve bons resultados na classificação quando foi configurada com 8 neurônios na camada oculta.

Deste modo, conclui-se que a RN-MLP com método de aprendizado de Levenberg-Marquardt mostrou-se uma técnica mais eficiente que as RNs de Elman e RN-RBF ao tratar da classificação das variáveis descritas nas subseções 2.3 e 2.4, sendo a classificação obtida pela RN-MLP mais satisfatória que as outras arquiteturas apresentadas nesta dissertação em termos de percentuais de acerto da classificação.

5.2 Recomendação para trabalhos futuros

Nesta dissertação validou modelos de classificação baseados em RNs para a

classificação dos dados descritos nas subseções 2.3 e 2.4.

Para pesquisa futura pode-se aplicar métodos de análise de correlação para medir a correlação entre os atributos disponibilizados pela empresa DM-Inf S/A em conjunto com a CISP e SPC, com a correlação pode-se diminuir a quantidade variáveis utilizadas para a classificação.

A empresa DM-Inf S/A não irá utilizar o software computacional desenvolvido neste estudo, devido ao seu alto custo do ambiente computacional MATLAB, mesmo a implementação em MATLAB apresentou bons resultados.

Para pesquisa futura, o autor deseja testar novos métodos de treinamento para as RNs-MLP, Elman e RN-RBF usando métodos híbridos com paradigmas da computação evolutiva.

Podem ser realizadas novas análises comparativas levando-se em conta outros critérios diferentes. Não apenas verificando a média e o desvio padrão da variável “Acerto de Clientes Adimplentes” obtida como saída das RNs, mas também podem ser considerado o MSE em conjunto com a média e o desvio padrão obtido na variável “Acerto de Clientes Adimplentes”.

REFERÊNCIAS

- ALTMAN, E. I.; SAUNDERS, A. **Credit Risk Measurement: Developments over the Last 20 years.** Journal of Banking & Finance, vol. 21, p. 1721 - 1742, 1998.
- ARNAUD, A. L., CUNHA, R. C., LUDERMIR, T. B., ADEODATO, P. J. L. **Modelo Híbrido de Redes Neurais Artificiais e Técnicas Não-Supervisionadas para problema de Credit Scoring.** In: Anais do Encontro Nacional de Inteligência Artificial, (ENIA) São Leopoldo, RS, p. 922 - 931, 2005.
- ATIYA, A. F. **Bankruptcy Prediction for Credit Risk Using Neural Networks: A Survey and New Results.** IEEE Transactions on Neural Networks, vol. 12. no. 4, p. 929 - 935, 2001.
- BASTOS, J. **Credit Scoring with Boosted Decision Trees.** Technical Report, CEMAPRE, School of Economics and Management (ISEG), Technical University of Lisbon, Lisboa, Portugal, 2008.
- BERRY, A. J. M.; LINOFF, F. G. **Data Mining Techniques for Marketing, Sales, and Customer Relationship Management.** 2ª ed., Canada, Wiley & Publishing, 2004.
- BILLINGS, S. A.; XONG, X. **Dual-orthogonal Radial Basis Function Networks for Nonlinear Time Series Prediction.** Neural Networks, vol. 11, no. 3, p. 479 - 493, 1998.
- BÓDIS, L. **Financial Times Series Forecasting Using Artificial Neural Networks.** Master Thesis, “Babes-Boluai” University, Faculty of Mathematics and Computer Science, Department of Computer Science, Cluj-Napoca, Romania, 2004.
- BORS, A. G. **Introduction of the Radial Basis Function (RBF) Networks.** Online Symposium for Electronics Engineers, DSP Algorithms, York, UK, vol. 1, p. 1 - 7, 2001. Disponível em: <<http://axiom.anu.edu.au/~daa/courses/GSAC6017/rbf.pdf>>. Acesso em: 03 de Fevereiro de 2009.
- BOX, G.; JENKINS, G. **Time Series Analysis: Forecasting and Control.** Holden Day: San Francisco, CA, USA, 1976.
- BRUNELLI, U.; PIAZZA, V.; PIGANTO, L.; SORBELLO, F.; VITABILE, S. **Three Hours ahead Prevision of SO2 Pollutant Concentration using an Elman Neural based Forecaster.** Building and Environment, vol. 43, no. 3, p. 304 - 314, 2006.
- CARVALHO, J. V. **Modelagem Temporal das Médias de Vazão de Drenos na Barragem de Funil (RJ) utilizando Redes Neurais e Métodos Estatísticos.** Tese de Doutorado, Programa de Pós-Graduação em Engenharia Civil, Pontifical Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, 2005.
- CASTRO, M. C. F. **Predição Não-Linear de Séries Temporais. Usando Redes Neurais RBF por Decomposição em Componentes Principais.** Tese de Doutorado, Programa de Pós-Graduação em Engenharia Elétrica, Faculdade de Engenharia Elétrica e de Computação de Campinas – UNICAMP, Campinas, SP, 2001.

CELIKOGU, H. B.; CIGIZOGLU, H. K. **Modelling Public Transport Trips by Radial Basis Function Neural Networks**. Mathematical and Computer Modelling, vol. 45, no. 3 - 4, p. 480 - 489, 2007.

CHAN, L. W. **Levenberg-Marquardt Learning and Regularization**. Progress in Neural Information Processing, p. 139 - 144, 1996.

CORRÊA, M.; F.; VELLASCO, M. **Análise de Risco de Crédito em Correspondentes Bancários através de Redes Neurais**. Laboratório de Inteligência Computacional – Departamento de Engenharia Elétrica da Pontifícia Universidade Católica do Rio de Janeiro, RJ, 2009. Disponível em: <http://publique.rdc.puc-rio.br/rica/media/ICA01_CorreiaVellasco.pdf>. Acesso em: 18 de Janeiro de 2009.

COVER, T. M. **Geometrical and Statistical Properties of System of Linear Inequalities with Applications in Recognition**. IEEE Transactions on Electronic Computers, vol. 14, no. 3, p. 126 - 334, 1965.

CYBENKO, G. **Approximation by Superpositions of a Sigmoidal Function**. Journal Mathematics of Control, Signals, and Systems, vol. 5, no. 4, p. 303 - 314, 1992.

DA SILVA, G. B. **Sistemas Híbridos de Previsão de Carga Elétrica em Curto Prazo utilizando Redes Neurais Artificiais e Lógica Fuzzy**. Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Pernambuco, Recife, PE, 2006.

DE ANDRADE, M. C.; RIOSNETO, A.; ROSA, R. R. **Estimativa de Padrões Espaço-Temporais Utilizando Redes Neurais Artificiais do Tipo Multilayer Perceptron com Aplicação em Física Solar**. Anais do I Workshop dos Cursos de Computação Aplicada do INPE, São José dos Campos, SP, p. 6 - 8, 2001.

DE OLIVEIRA, M. A. **Previsão de Sucessões Cronológicas Econômico-Financeiras por meio de Redes Neurais Artificiais Recorrentes de Tempo Real e de Processos ARMA-GARCH: Um Estudo Comparativo quanto à Eficiência de Previsão**. Dissertação de Mestrado, Programa de Pós-Graduação em Administração, Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, São Paulo, SP, 2003.

DE OLIVEIRA, M. A. **Aplicação de Redes Neurais Artificiais na Análise de Séries Temporais Econômicas-Financeiras**. Tese de Doutorado, Programa de Pós-Graduação em Administração, Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo, São Paulo, SP, 2007.

ELMAN, J. L. **Finding Structure in Time**. Cognitive Science, vol. 14, no. 2, p. 179 - 211, 1990.

FAYYAD, U.; PIATETSKY-SHAPIRO, P. S. **The KDD Process for Extracting useful Knowledge from Volume of Data**. Communications of the ACM, vol. 39, no. 11, p. 27 - 34, 1996.

FLETCHER, R. **The Newton Method Converges Fast but is Costly.** Practical Methods of Optimization. New York, USA, Chapter 3, 1987.

GAO, X. Z.; GAO, X. M.; OVASKA, S. J. **A Modified Elman Neural Network Model with Application to Dynamical System Identification.** Proceedings of IEEE International Conference, Systems, Man, and Cybernetics, Beijing, China, vol. 2, no. 14-17, p. 1376-1381, 1996.

GE, H.; LIANG, Y.; MARCHESE, M. **A Modified Particle Swarm Optimization-based Dynamic Recurrent Neural Network for Identifying and Controlling Nonlinear Systems.** Computers and Structures, vol. 85, no. 21-22, p. 1611-1622, 2007.

GIL, A. C. **Como Elaborar Projetos de Pesquisa.** 3. Ed. São Paulo: Atlas S/A., 1991.

GIORDANO, F.; LA ROCCA, M.; PERNA, C. **Forecasting Nonlinear Time Series with Neural Network Sieve Bootstrap.** Computational Statistics & Data Analysis, vol. 51, no. 8, p. 3871 - 3884, 2007.

GODOY, A. S. **Introdução à Pesquisa Qualitativa e suas Possibilidades.** Revista de Administração de Empresas, São Paulo, vol. 35. no. 2, p. 35 – 37, 1995.

GUERRA, F. A. **Análise de Métodos de Agrupamento para o Treinamento de Redes Neurais de Base Radial Aplicadas à Identificação de Sistemas.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, Pontifícia Universidade Católica do Paraná, Curitiba, PR, 2006.

GUIMARÃES, L. R. **Previsão de Inadimplência e Redes Neurais Artificiais.** Dissertação de Mestrado, Programa de Pós-Graduação em Modelagem Computacional, Rio de Janeiro, RJ, 2006.

HAGAN, M. T.; MENHAJ, M. B. **Training Feedforward Networks with the Marquardt Algorithm.** IEEE Transactions on Neural Networks, vol. 5, no. 6, p. 989 - 993, 1994.

HAND, D.; HENLEY, W. **Statistical Classification Methods in Consumer Credit Scoring: A review.** Journal Of The Royal Statistical Society, vol. 160, no. 3, p. 523 - 541, 1997.

HAND, D.; MANNILA, H.; SMYTH, P. **Principles of Data Mining.** The MIT Press, Cambridge, Massachusetts, London, England, 2001.

HANSEN, V. J.; NELSON, D. R. **Data Mining of Time Series Using Stacked Generalizers.** Neurocomputing, vol. 43, no. 2, p. 173 - 184, 2002.

HARDY, R. L. **Multiquadric Equations of Topography and other Irregular Surfaces.** Journal of Geophysics Research, vol. 76, no. 8, p. 1905 - 1915, 1971.

HAYKIN, S. **Neural Networks: A Comprehensive Foundation.** Macmillan College Publishing, New York, NY, USA, 1994.

HAYKIN, S. **Adaptive Filter Theory.** 3nd, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.

- HAYKIN, S. **Redes Neurais: Princípios e Prática**. 2ª ed., Porto Alegre/RS, Editora Bookman, 2001.
- HORNIK, K. **Approximation Capabilities of Multilayer Feedforward Networks**. Neural Networks, vol. 4, no. 2, p. 251 - 257, 1991.
- HUNG, S.; CHEN; MU-CHEN; W. **Credit Scoring with a Data Mining Approach Based on Support Vector Machines**. Expert Systems with Applications, vol. 33, no. 4, p. 847 - 856, 2007.
- HUTCHINSON, J. M. **A Radial Basis Function Approach to Financial Time Series Analysis**. PhD Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts, England, 1994.
- JOHNSON, R.A., WICHERN, D.W. **Applied Multivariate Statistical Analysis**. Upper Saddle River, NJ, USA, 2002.
- JORDAN, M. I. **Attractor Dynamics and Parallelism in a Connectionist Sequential Machine**. Proceedings of the Eighth Annual Conference of the Cognitive Science Society, Piscataway, NJ, USA, p. 531 - 546, 1986.
- JUNG, C. F. **Metodologia Para a Pesquisa & Desenvolvimento**. Rio de Janeiro: Axcel Books do Brasil Editora, 2004.
- KOMORÁD, K. **On Credit Scoring Estimation**. Master's Thesis, Institute for Statistics and Econometrics, Berlin, Heidelberg, Germany, 2002.
- KREMER, S. C. **On the Computational Power of Elman-Style Recurrent Networks**. IEEE Transactions on Neural Networks, vol. 6, no. 4, p. 1000 - 1004, 1995.
- LAI, K. K.; YU, L.; WANG, S. ZHOU, LIGANG. **Neural Network Metalearning for Credit Scoring**. Lecture Notes in Computer Science, Berlin, Heidelberg, Germany, vol. 4113-2006, p. 403 - 408, 2006.
- LAI, K. K.; YU, L.; ZHOU, L.; WANG, S. **Credit Risk Evaluation with Least Square Support Vector Machine**. Springer-Verlag, Berlin, Heidelberg, Germany, vol. 4062, p. 490 - 495, 2006.
- LAKATOS, E. M.; MARCONI, M. de A. **Metodologia do Trabalho Científico**. São Paulo: Atlas, 1995.
- LAM, M. **Neural Network Techniques for Financial Performance Prediction: Integrating Fundamental and Technical Analysis**. Decision Support Systems, vol. 37, no. 4, p. 567 - 581, 2004.
- LELAND, W. E.; TAQQU, M. S.; WILLINGER, W.; WILSON, D. V. **On the Self-similar Nature of Ethernet Traffic**. IEEE/ACM Transactions on Networking, vol. 2, no. 1, p. 1 - 15, 1994.
- LEVENBERG, K. **A Method for the Solution of Certain Problems in Least Squares**.

Quarterly of Applied Mathematics, vol. 2, p. 164 - 168, 1944.

LIMA, F. G. **Um Método de Análise e Previsão de Sucessões Cronológicas Unidimensionais Lineares e Não-Lineares.** Tese de Doutorado, Programa de Pós-Graduação em Administração, Faculdade de Economia, Administração e Contabilidade de São Paulo, São Paulo, SP, 2004.

LIU, Y.; SCHUMANN M. **Data Mining Feature Selection for Credit Scoring Models.** Journal of the Operational Research Society, vol. 56, no. 9, p. 1099 - 1108, 2005.

LÜDKE, M; ANDRÉ, M. E. **Pesquisa em Educação: Abordagens Qualitativas.** São Paulo: EPU, 1986.

MAJI, P. **Efficient Design of Neural Network Tree using a New Splitting Criterion.** Neurocomputing, vol. 71, no. 4-6, p. 787 - 800, 2007.

MAQSOOD, I.; ABRAHAM, A. **Weather Analysis using Ensemble of Connectionist Learning Paradigms.** Applied Soft Computing, vol. 7, no. 3, p. 995 - 1004, 2006.

MARQUARDT, D. **An Algorithm for Least-squares Estimation of Nonlinear Parameters.** SIAM Journal on Applied Mathematics, vol. 11, p. 431 - 441, 1963.

MARTIN, C. **Aplicação de Redes Neurais para Prognóstico com Base em Séries Temporais.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, SC, 2000.

MATARAZZO, D. C. **Análise Financeira de Balanços.** São Paulo - SP: Atlas, 1995.

MATHWORKS, INC. **MATLAB (MATrix LABoratory).** version 7. www.mathworks.com, 2008.

MATTAR, F. N. **Pesquisa de Marketing: Metodologia, Planejamento.** 5.ed. São Paulo: Atlas, 1999.

MENDES, F. E. F.; CARVALHO, A. C. P. L. F.; MATIAS, A. B. **Utilização de Redes Neurais Artificiais na Análise de Risco de Crédito a Pessoas Físicas,** 2009. Disponível em: <http://www.cepefin.org.br/publicados_pdf/Utilizacao_rna_analise_risco_credito_pf.pdf>. Acesso em: 08 de Janeiro de 2009.

MESTER, L. J. **What's the Point of Credit Scoring?** Business Review, Philadelphia, USA, p. 1 - 16, 1997.

MORIMITSU, H.; JUNIOR, M. P. R.; SANTOS, R. C.; HONDA W. Y. **Tratamento de Dados e Aplicação de uma Rede Neural MLP para Credit Scoring,** 2009. Disponível em: <<http://www.ime.usp.br/~cpq/home/arquivos/Henrique%20Morimitsu%20e%20Rosiane%20Correia%20Santos.pdf>>. Acesso em: 23 de Janeiro de 2009.

MÜLLER, M.; KRAFT, H.; KROISANDT, G. **Assessing Discriminatory Power of Credit Rating.** 2002. Disponível em: <<http://edoc.hu-berlin.de/series/sfb-373-papers/2002-67/PDF/67.pdf>>. Acesso em: 23 de Janeiro de 2009.

PHAM, D. T.; KARABOGA, D. **Training Elman and Jordan Networks for System Identification using Genetic Algorithms.** Artificial Intelligence in Engineering, vol. 13, no. 2, p. 107-117, 1998.

PINO, R.; PARRENO, J.; GOMEZ, A.; PRIORE, P. **Forecasting Next-day Price of Electricity in the Spanish Energy Market using Artificial Neural Networks.** Engineering Applications of Artificial Intelligence, vol. 21, no. 1, p. 53 - 62, 2008.

ROSELLI, M. A. **Correção da Saturação de Transformadores de Corrente Via Redes Neurais Artificiais e seus Efeitos na Proteção de Distância.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ, 2007.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning Internal Representations by Error Propagation.** In: Parallel distributed processing: Explorations in the Microstructure of Cognition. Foundations, eds. Rumelhart, D. E., McClelland, J. L. MIT Press, vol. 1, Cambridge, Massachusetts, England, 1986.

SANTOS FILHO, E. L. **Previsão dos Retornos do Índice Bovespa Usando Redes Neurais Artificiais.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, Pontifícia Universidade Católica do Paraná, Curitiba, PR, 2008.

SANTOS, G. S. M. **Aplicação de Redes Neurais para Diagnóstico de Poluição em Isoladores de Alta Tensão.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Pernambuco, Recife, PE, 2008.

SANTOS, M. A. **Representando a Gestão Estratégica da Inovação na Indústria de Software Brasileira: Uma Abordagem Orientada pela Engenharia Ontológica.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, Pontifícia Universidade Católica do Paraná, Curitiba, PR, 2006.

SCHIFFMANN, W.; JOOST, M.; WERNER, R. **Optimization of the Backpropagation Algorithm for Training Multilayer Perceptrons.** University of Koblenz Institute of Physics vol. 2, 1994. Disponível em: <<http://www.cs.bham.ac.uk/~pxt/NC/schiffmann.bp.pdf>>. Acesso em: 10 de Dezembro de 2008.

SCHÜHLI, E. S. **Reconhecimento de Gestos de Maestro Utilizando Redes Neurais Artificiais Parcialmente Recorrentes.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal do Paraná, Curitiba, PR, 2005.

SEBRAE, 2008. Acesso em 20 de agosto de 2008. Disponível em: <http://www.sebraesp.com.br/principal/abrindo%20seu%20neg%C3%B3cio/produtos%20sebrae/artigos/listadeartigos/inadimplencia.aspx>.

SOUZA, R. C. T. **Previsão de Séries Temporais Utilizando Rede Neural Treinada por Filtro de Kalman e Evolução Diferencial.** Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia de Produção e Sistemas, Pontifícia Universidade Católica do Paraná, Curitiba, PR, 2008.

STEINER, M. T. A.; NIEVOLA, J. C.; SOMA, N. Y; SHIMIZU, T.; STEINER, P. J. N. **Extração de Regras de Classificação a Partir de Redes Neurais para Auxílio à Tomada de Decisão na Concessão de Crédito Bancário.** Pesquisa Operacional, vol. 27, no. 3, p. 407 - 426, 2007.

SURATGAR, A. A.; TAVAKOLI M. B.; HOSEINABADI, A. **Modified Levenberg-Marquardt Method for Neural Networks Training.** Proceedings of World Academy of Science, Engineering and Technology, vol. 6, p. 46 - 48, 2005.

THOMASSEY, S.; HAPPIETTE, M. **A Neural Clustering and Classification System for Sales Forecasting of New Apparel Items.** Applied Soft Computing, vol. 7, no. 4, p. 1177 – 1187, 2007.

TRIPPI, R.; TURBAN, E. **Neural Network in Finance & Investing.** McGraw-Hill, Chicago, USA, 1992.

WEIGEND, A. S.; GERSHENFELD, N. A. **Time Series Prediction: Forecasting the Future and Understanding the Past.** Addison-Wesley Publishing Company, Menlo Park, CA, USA, 1994.

WEIGEND, A. S.; HUBERMAN, B. A.; RUMELHART, D. E. **Predicting the Future: A Connectionist Approach.** International Journal of Neural Systems, vol. 1, no. 3, p. 193 - 209, 1990.

WERBOS, P. J. **Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.** PhD Thesis, Harvard University, Cambridge, Massachusetts, England, 1973.

WILAMOWSKI, B.M.; IPLIKCI, S.; KAYNAK, O.; EFE, M. O. **An Algorithm for Fast Convergence in Training Neural Networks.** Proceedings of International Joint Conference on Neural Networks, Washington, DC, USA. vol. 3, p. 1778 - 1782, 2001.

WILLIAMS, R. J. E ZIPSER, D. **Gradient-based Learning Algorithms for Recurrent Networks and their Computational Complexity.** Backpropagation: theory, architectures, and applications, L. Erlbaum Associates Inc, Hillsdale, NJ, USA, p. 433 - 486, 1995.

YAO, X. **Evolving Artificial Neural Networks.** Proceedings of IEEE, vol. 87, no. 9, p. 1423-1439, 1999.

ZHANG, L.; GU, J.; QIN, Z. **Elamn Network using Simulated Annealing Algorithm and its Application in Thermal Processes Modeling.** Proceedings of IEEE Machine Learning and Cybernetics, Hong Kong, China, vol. 2, no. 19-22, p. 929 - 933, 2007.