

UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE CIÊNCIAS AGRONÔMICAS
CAMPUS DE BOTUCATU

**APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA ANÁLISE DE
DADOS DE MOLHAMENTO FOLIAR POR ORVALHO**

IVO MÁRIO MATHIAS

Tese apresentada à Faculdade de Ciências Agronômicas da Unesp - Campus de Botucatu, para obtenção do título de Doutor em Agronomia - Área de Concentração em Energia na Agricultura.

BOTUCATU – SP

Dezembro - 2006

UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE CIÊNCIAS AGRONÔMICAS
CAMPUS DE BOTUCATU

**APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS NA ANÁLISE DE
DADOS DE MOLHAMENTO FOLIAR POR ORVALHO**

IVO MÁRIO MATHIAS

Orientador: Prof. Dr. Angelo Cataneo

Co-orientadora: Prof. Dra. Alaine Margarete Guimarães

Tese apresentada à Faculdade de Ciências Agronômicas da Unesp - Campus de Botucatu, para obtenção do título de Doutor em Agronomia - Área de Concentração em Energia na Agricultura.

BOTUCATU – SP

Dezembro - 2006

FICHA CATALOGRÁFICA ELABORADA PELA SEÇÃO TÉCNICA DE AQUISIÇÃO E
TRATAMENTO DA INFORMAÇÃO - SERVIÇO TÉCNICO DE BIBLIOTECA E DOCUMENTAÇÃO
UNESP - FCA - LAGEADO - BOTUCATU (SP)

M426a Mathias, Ivo Mário , 1959-
Aplicação de redes neurais artificiais na análise de
dados de molhamento foliar por orvalho / Ivo Mário Mathias
. - Botucatu : [s.n.], 2006.
xi, 120 f. : il., color., gráfs, tabs.

Tese (Doutorado)-Universidade Estadual Paulista, Facul-
dade de Ciências Agrônômicas, Botucatu, 2006
Orientador : Angelo Cataneo
Co-orientador : Alaine Margarete Guimarães
Inclui bibliografia

1. Inteligência artificial. 2. Agricultura. 3. Umidade. 4.
Redes neurais. 5. Plantas - Efeito de umidade. I. Cataneo,
Angelo. II. Guimarães, Alaine Margarete. III. Universida-
dade Estadual Paulista "Júlio de Mesquita Filho" (Campus
de Botucatu). Faculdade de Ciências Agrônômicas. IV. Títu-
lo.

UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"
FACULDADE DE CIÊNCIAS AGRONÔMICAS
CAMPUS DE BOTUCATU
CERTIFICADO DE APROVAÇÃO

TÍTULO: "APLICAÇÃO DE REDES NEURAS ARTIFICIAIS NA ANÁLISE DE DADOS DE MOLHAMENTO FOLIAR POR ORVALHO"

ALUNO: IVO MÁRIO MATHIAS

ORIENTADOR: PROF. DR. ANGELO CATANEO

CO-ORIENTADORA: PROFª DRª ALAINE MARGARETE GUIMARÃES

Aprovado pela Comissão Examinadora



PROFª DRª ALAINE MARGARETE GUIMARÃES



PROFª DRª CÉLIA REGINA LOPES ZIMBACK



PROF. DR. MARCO ANTONIO MARTIM BIAGGIONI



PROF. DR. MANOEL HENRIQUE SALGADO



PROF. DR. MARCELO GIOVANETTI CANTERI

Data da Realização: 11 de dezembro de 2006.

“O que sabemos é uma gota, o que ignoramos é um oceano.”

Isaac Newton, 1642-1727

Dedicatória

Dedico esta tese a minha amada esposa Marilia, que sempre me estimulou nos momentos de incerteza e me compreendeu nos momentos de dedicação ao meu trabalho. Obrigado por seu amor, sua tolerância e sua paciência.

AGRADECIMENTOS

Agradeço a Deus, que está presente em todos os momentos de minha vida, fornecendo-me idéias e sabedoria. Sua energia, sempre delineou o meu desenvolvimento pessoal e profissional. E, sem a força deste, nada seria possível.

Agradeço aos meus pais Estanislau e Catarina, pelo empenho e ajuda que sempre me deram, estimulando-me e propiciando-me muitos conhecimentos e realizações, reerguendo sempre o meu ânimo e a minha vontade de aprender.

Meus orgulhos e razões de existência, minhas filhas Camila e Fernanda. Agradeço à vocês que sempre souberam me entender e me estimular durante todo o desenvolvimento do doutorado, mesmo quando eu estava ausente morando em Botucatu.

Meu orientador Prof. Dr. Angelo Cataneo, obrigado pela sua orientação, amizade e confiança. Sua sabedoria e integridade foram fundamentais em todas as fases do meu trabalho. Você foi para mim mais do que um orientador, sempre foi um grande amigo.

A amiga Elaine e também co-orientadora, obrigado por suas palavras de incentivo e por sempre estar disposta a colaborar no desenvolvimento da minha tese. Minha gratidão também, por ter sido a pioneira em Botucatu, suas dicas a respeito da cidade e da Unesp sempre foram muito importantes. Agradeço por sua amizade e ajuda.

Ao amigo Ariangelo, obrigado pela convivência harmoniosa durante o período que moramos em Botucatu e também, por compartilhar seus conhecimentos, os quais sempre foram de grande importância para o meu trabalho.

Agradeço a FABC – Fundação ABC na pessoa do Engenheiro Agrônomo Rodrigo Yoti Tsukahara, que me forneceu duas das bases de dados utilizadas neste trabalho, assim como, agradeço pelas orientações técnicas que foram muito importantes no desenvolvimento desta tese.

Aos meus colegas da UEPG – Ponta Grossa (PR) e da UNESP – Botucatu (SP), obrigado pelo apoio e incentivo.

Aos colegas do DEINFO/UEPG, obrigado pela força e incentivo durante esta jornada.

À UEPG, agradeço pela licença remunerada que viabilizou o meu doutorado.

Meus agradecimentos à Capes, pela aprovação do PQI – Programa de Qualificação Institucional, cujos recursos foram fundamentais para a realização do trabalho.

Aos acadêmicos da UEPG do Curso de Engenharia de Computação Douglas e Marcos, e do Curso de Bacharelado em Informática Carlos e Jones, pelo exemplar trabalho desenvolvido sob minha orientação, cujo trabalho é parte essencial desta tese.

Ao Professor Doutor Paulo César Sentelhas, pelas suas sugestões a respeito do meu trabalho e também por ter fornecido uma das bases de dados utilizadas nesta tese.

Agradeço ao Eduardo Alvarez Santos, na ocasião, orientado do Professor Dr. Paulo César Sentelhas, pela sua contribuição no envio e explicações a respeito da base de dados da ESALQ/USP de Piracicaba.

Ao Marcelo Giovanetti Canteri, agradeço pela amizade e pelo incentivo em realizar o meu doutorado na área da informática aplicada a agricultura.

Agradeço a todos os integrantes da banca examinadora por aceitarem o convite de participação.

A Márcia Eurich Belinsky da PROPESP/UEPG, agradeço por sempre estar disposta a colaborar e principalmente pela ajuda em inúmeras reuniões realizadas quando da elaboração do PQI/Capes.

A FCA/UNESP Botucatu, ao Departamento de Gestão e Tecnologia Agroindustrial - DGTA. As chefias do DGTA, obrigado pelo espaço e computadores cedidos ao desenvolvimento do meu trabalho. A todos os professores e funcionários pela acolhida sempre gentil e atenciosa. Em particular aos funcionários Marcos Norberto Tavares (Marquinhos), Anselmo Ribeiro e Mário Eduardo Bianconi Baldini, que sempre estiveram dispostos a auxiliar naquilo que fosse necessário.

A FCA/UNESP Botucatu, a equipe da Seção de Pós-Graduação, Jaqueline de Moura Gonçalves, Kátia Otomo Duarte, Marlene Rezende de Freitas e Marilena do Carmo Santos, em especial a Marlene pela sua atenção e cortesia. Meus agradecimentos.

Ao Luciano Frontino Medeiros, por permitir usar o componente TMLP (*Training Multilayer Perceptron*) de seu Livro Redes Neurais em Delphi.

SUMÁRIO

LISTA DE FIGURAS	XI
LISTA DE TABELAS	XIII
LISTA DE ABREVIATURAS.....	XIV
1 RESUMO	1
2 SUMMARY	3
3 INTRODUÇÃO.....	5
3.1 Contexto do trabalho.....	5
3.2 Especificação do problema	6
3.3 Objetivo do trabalho	7
4 REVISÃO DE LITERATURA	8
4.1 Redes Neurais Artificiais	8
4.1.1 O neurônio artificial e a rede neural	10
4.1.2 Modelo de neurônio.....	13
4.1.3 Tipos de função de ativação	15
4.1.4 Arquiteturas de RNAs	18
4.1.5 Tarefas de aprendizagem	20
4.1.6 Processo de aprendizagem.....	21
4.1.7 Modelos de aprendizagem	23
4.1.8 Perceptron de camada única	24
4.1.9 Perceptron de múltiplas camadas	28
4.1.10 Algoritmo de retropropagação de erro (<i>error back-propagation</i>).....	30

4.1.11	Algoritmo <i>resilient propagation</i> (Rprop).....	36
4.2	Molhamento foliar	38
4.2.1	Orvalho	39
4.2.2	Molhamento foliar por orvalho.....	39
4.2.3	Estimativa da DPMF	41
4.2.4	Principais fatores no desenvolvimento de doenças em plantas	43
4.2.5	Doenças do trigo.....	44
4.2.6	Cultura do trigo no Brasil	44
5	MATERIAL E MÉTODOS.....	46
5.1	Bases de dados	47
5.1.1	Especificação da base de dados 1	47
5.1.2	Especificação da base de dados 2.....	49
5.1.3	Especificação da base de dados 3.....	51
5.2	Metodologia computacional.....	52
5.2.1	Modelagem de redes neurais artificiais	53
6	RESULTADOS	54
6.1	Desenvolvimento do modelo em RNA	54
6.1.1	Definição do problema e coleta de dados.....	54
6.1.2	Pré-processamento dos dados.....	57
6.1.3	Projeto da RNA	59
6.1.4	Treinamento da RNA.....	61
6.2	Estudos de caso	62
6.2.1	Estudo de caso 1	62

6.2.2	Estudo de caso 2	64
6.2.3	Estudo de caso 3	65
6.3	Definição do sistema PMNeural	68
6.4	Funcionalidades do sistema PMNeural.....	71
6.5	Avaliação do sistema PMNeural.....	89
6.5.1	Estudo de caso 1	90
6.5.2	Estudo de caso 2	91
6.5.3	Estudo de caso 3	92
6.5.4	Estudo de caso 4	93
7	DISCUSSÕES	95
8	CONSIDERAÇÕES	100
9	CONCLUSÕES	102
10	REFERÊNCIAS BIBLIOGRÁFICAS	103
	APÊNDICE	116

LISTA DE FIGURAS

Figura 1 – Esquema de um neurônio biológico.....	11
Figura 2 – Modelo matemático de um neurônio biológico (McCulloch & Pitts, 1943)	12
Figura 3 – Modelo de neurônio base para projetos de RNA	13
Figura 4 – Gráfico função de limiar	15
Figura 5 – Gráfico função linear por partes.....	16
Figura 6 – Gráfico função sigmóide.....	17
Figura 7 – RNA representada em forma de Grafo.....	18
Figura 8 – Exemplos de arquiteturas de RNAs	19
Figura 9 – Modelo perceptron	25
Figura 10 – Gráfico perceptron 1 e 2 – classificação de frutos	27
Figura 11 – Arquitetura de RNA MLP com duas camadas ocultas.....	29
Figura 12 – Fluxo de treinamento de uma MLP <i>c/back-propagation</i>	32
Figura 13 – Gráfico erro global x peso.....	35
Figura 14 – Visão da EMA utilizada para coleta de dados da base 2. Pirai do Sul, PR. 2005..	50
Figura 15 – Detalhes de um sensor de molhamento Campbell modelo 237	52
Figura 16 – Etapas do desenvolvimento da RNA.....	53
Figura 17 – Regra de parada antecipada baseada na validação cruzada	63
Figura 18 – Tela SNNS – gráfico de evolução de treinamento/validação.....	66
Figura 19 – Tela SNNS – exibição de RNA com arquitetura 4-14-2.....	67
Figura 20 – Tela SNNS – painel de controle.....	68
Figura 21 – Tela de apresentação – PMNeural.....	72
Figura 22 – Tela de operações e <i>login</i>	72

Figura 23 – Tela de operações	73
Figura 24 – Tela – acesso ao cadastro de usuário.....	74
Figura 25 – Tela – cadastro de usuário.....	75
Figura 26 – Tela – cadastro de estações meteorológicas.....	76
Figura 27 – Tela – cadastro de cidades	77
Figura 28 – Tela – exemplos de variáveis	77
Figura 29 – Tela – cadastro de variável.....	78
Figura 30 – Tela – importação de dados	79
Figura 31 – Tela – iniciar importação na linha.....	80
Figura 32 – Tela – neurônios da camada de saída	81
Figura 33 – Tela – treinamento da RNA	82
Figura 34 – Tela – relatório de treinamento	84
Figura 35 – Tela – validação de RNAs.....	85
Figura 36 – Tela – resultados de validação	86
Figura 37 – relatório de validação - resumido.....	87
Figura 38 – Tela de ajuda – aba localizar.....	88
Figura 39 – Tela de ajuda – aba índice.....	89

LISTA DE TABELAS

Tabela 1 – Treinamento de perceptron para classificação de frutos.....	26
Tabela 2 – Conjunto de dados para classificação de frutos.....	27
Tabela 3 – Composição parcial da base de dados 1 (Tsukahara, 2004).....	49
Tabela 4 – Composição parcial da base de dados 2 (Tsukahara, 2005).....	50
Tabela 5 – Composição parcial da base de dados 3 (ESALQ, 2005).....	51
Tabela 6 – Escala de avaliação visual de orvalho Heldwein & Krzysch (1997).....	55
Tabela 7 – Escala de avaliação visual discretizada – base de dados 1	56
Tabela 8 – Neurônios de saída – base de dados 1	58
Tabela 9 – Detalhes - base de dados 1	62
Tabela 10 – Melhores resultados c/simuladores – base de dados 1.....	64
Tabela 11 – Melhores resultados c/simuladores – base de dados 2.....	65
Tabela 12 – Melhores resultados c/simuladores – base de dados 3.....	65
Tabela 13 – Avaliação PMNeural - estudo de caso 1.....	91
Tabela 14 – Avaliação PMNeural - estudo de caso 2.....	92
Tabela 15 – Avaliação PMNeural - estudo de caso 3.....	93
Tabela 16 – Avaliação PMNeural - estudo de caso 4.....	94
Tabela 17 – Épocas processadas em 10 horas de treino c/o PMNeural – algoritmo Rprop.....	98

LISTA DE ABREVIATURAS

ANN	Artificial Neural Networks
Backprop	Back-propagation
DARPA	Defense Advanced Research Projects Agency
DPM	Duração do Período de Molhamento
DPMF	Duração do Período de Molhamento Foliar
DPMV	Duração do Período de Molhamento Visual
EMA	Estação Meteorológica Automática
EMBRAPA	Empresa Brasileira de Pesquisa Agropecuária
IA	Inteligência Artificial
IMF	Índice de Molhamento Foliar
JavaNNS	Java Neural Network Simulator
MLP	Multilayer Perceptron
OCR	Optical Character Recognition
PRES	Pressão atmosférica do ar
RN	Rede Neural
RNA	Rede Neural Artificial
Rprop	Resilient Propagation
SNNS	Stuttgart Neural Network Simulator
TEMPMAX	Temperatura Máxima do ar
TEMPMED	Temperatura Média do ar
TEMPMIN	Temperatura Mínima do ar
TMLP	Training Multilayer Perceptron
UML	Unified Modeling Language
URMAX	Umidade Relativa Máxima do ar
URMED	Umidade Relativa Média Relativa do ar
URMIN	Umidade Relativa Mínima do ar
VENTMAX	Velocidade Máxima do vento
VENTMED	Velocidade Média do vento
VENTMIN	Velocidade Mínima do vento

1 RESUMO

O trabalho descrito nesta tese apresenta o desenvolvimento de um sistema computacional denominado PMNeural, baseado em Redes Neurais Artificiais (RNAs). A finalidade do sistema é o tratamento de dados climáticos e de molhamento foliar por orvalho, visando reconhecer padrões de comportamento de variáveis meteorológicas em relação ao molhamento foliar por orvalho. Para determinar as melhores arquiteturas e algoritmos de treinamento de RNAs, bem como, definir quais as variáveis climáticas que influenciam significativamente na ocorrência do molhamento foliar, foram utilizados dois simuladores: o simulador SNNS (Stuttgart Neural Network Simulator) versão 4.2, que utiliza plataforma operacional Linux e o simulador JavaNNS – Java Neural Network Simulator 1.1, com ambiente de execução Windows, o qual é baseado no SNNS. Foram utilizados dados climáticos de três estudos de caso, dois destes referentes à cultura do trigo, oriundos de locais e datas diferentes. Base de Dados 1 - Fazenda Capão do Cipó, em Castro – PR, safra de inverno de 2003. Base de Dados 2 - Campo Demonstrativo e Experimental da Fundação ABC - Fazenda Palmeirinha, em Piraí do Sul – PR., safra de inverno de 2005. Base de Dados 3 - Posto Agrometeorológico ESALQ/USP em Piracicaba - SP, período entre julho e setembro de 2005. Um quarto estudo de caso foi elaborado a partir dos arquivos dos estudos de casos 1, 2 e 3, utilizando-se as variáveis climáticas comuns, juntamente com seus respectivos índices de molhamento. Dentre os algoritmos de treinamento testados nos simuladores, o *Resilient*

Propagation (Rprop) foi o que apresentou as menores taxas de erro em relação aos outros, sendo eles: *Backpropagation Standard*, *Backpropagation for batch training*, *Backpropagation with momentum term*, *Backpropagation with chunkwise update*, *Backpropagation with Weight Decay* e *Quickprop*. Verificou-se, também, que dentre as variáveis climáticas utilizadas para classificação do molhamento foliar, a inclusão do horário influenciou na obtenção de melhores resultados das RNAs, durante a fase de treinamento e validação. Nos quatro estudos de caso o sistema PMNeural foi capaz de classificar o molhamento foliar com acertos superiores a 73,96%. O sistema PMNeural foi eficiente para reconhecer padrões em dados de variáveis meteorológicas visando estimar o molhamento foliar originado por orvalho.

Palavras Chave: Inteligência artificial, Agricultura, Índice de molhamento, Duração do período de molhamento foliar.

ARTIFICIAL NEURAL NETWORKS APLLIED TO LEAF WETNESS BY DEW DATA ANALYSIS. Botucatu, 2006. 120p. Tese (Doutorado em Agronomia/Energia na Agricultura) - Faculdade de Ciências Agronômicas, Universidade Estadual Paulista.

Author: IVO MÁRIO MATHIAS

Adviser: ANGELO CATANEO

Co-Adviser: ALAINE MARGARETE GUIMARÃES

2 SUMMARY

The work described in this thesis presents the development of a computational system named PMNeural based on Artificial Neural Networks (ANNs). The system has for purpose the handle of climatic and leaf wetness data, aiming to recognize patterns of behavior of meteorological variables in relation to the wetness from dew. Two simulators were used in order to determine the best architecture and ANNs training algorithms, as well as, to define which the climatic variables that influence significantly in the leaf wetness occurrence: the SNNS (Stuttgart Neural Network Simulator) version 4.2 for Linux platform, and the JavaNNS - Java Neural Network Simulator 1.1, for Windows platform, which is based on the SNNS. Climatic data of three case studies were used, two related to wheat culture, obtained from different places and dates. Dataset 1 - Capão do Cipó Farm, in Castro - PR, 2003. Dataset 2 - Palmeirinha Farm in Piraí do Sul - PR, 2005 winter crop. Dataset 3 - Meteorological Station of ESALQ/USP in Piracicaba - SP, from July to September, 2005. A fourth case study was elaborated from datasets of the case studies 1, 2 and 3, using the common climatic variables together with their respective wetness indexes. After testing the training algorithms in the simulators, the Resilient Propagation (Rprop) presented lower training errors than the others evaluated methods: Backpropagation Standard, Backpropagation for batch training, Backpropagation with momentum term, Backpropagation with chunkwise update, Backpropagation with Weight Decay and Quickprop. It was verified

also that, among the climatic variables used for classification of leaf wetness from dew, the inclusion of the schedule had influenced in the obtaining better ANNs results, during the training and validation phases. The PMNeural system was able to classify the wetness indexes, achieving higher than 73,96% accuracy. The PMNeural system was efficient to recognize patterns in meteorological variables data aiming to estimate the wetness from dew.

Keywords: Artificial intelligence, Agriculture, Wetness index, Wetness duration.

3 INTRODUÇÃO

3.1 Contexto do trabalho

A evolução da tecnologia da informação tem causado transformações sobre as organizações em geral e em particular no desenvolvimento de aplicações de computadores voltados a áreas específicas. Neste contexto, o cenário agrícola se faz presente, no qual o uso da informática tem importância fundamental em todas as fases do processo produtivo e organizacional.

Dentre as metodologias computacionais, a Inteligência Artificial (IA) contribui na busca de soluções e conhecimentos que auxiliem na tomada de decisões, através de programas e técnicas que manuseiem dados com o objetivo de obter informações úteis e implícitas. A IA tem como principal objetivo a representação do comportamento humano através de modelos computacionais, constituindo-se em um campo de pesquisa aberto e dinâmico que trata do estudo da solução de problemas através da distribuição de conhecimento entre diversas entidades.

Dentre os diversos paradigmas da IA, as Redes Neurais Artificiais (RNAs) são úteis para o reconhecimento de padrões, por oferecer método flexível e mais próximo dos processos biológicos. Na sua forma mais geral, a RNA é uma máquina projetada para modelar a maneira como o cérebro realiza uma tarefa particular ou função de interesse.

3.2 Especificação do problema

No cenário agrícola, um dos problemas que os produtores enfrentam para manter a produtividade das culturas é a ocorrência de doenças, as quais são influenciadas pelas condições climáticas, com destaque para a temperatura e a água livre sobre a superfície foliar.

Entre as variáveis climáticas possíveis de serem medidas ou estimadas, pode-se afirmar que a duração do período de molhamento foliar é o que mais se correlaciona com o início dos processos infecciosos, visto que, a maioria dos esporos¹ de fungos foliares patogênicos necessita de um filme de água sobre a superfície das folhas, por um determinado tempo, para que possam germinar e penetrar no hospedeiro, ocasionando o processo infeccioso. O molhamento foliar pode ser resultado de três fatores principais: orvalho, precipitação pluvial ou irrigação.

Sentelhas (1992) destaca vários trabalhos que evidenciam a importância do orvalho no estabelecimento e desenvolvimento de doenças, pois apesar de poder beneficiar as plantas, o orvalho quando persiste e é retido sobre as plantas por longos períodos favorece a germinação de esporos de fungos.

Atualmente o mecanismo mais utilizado para se medir o molhamento foliar são sensores eletrônicos, que tentam simular uma folha e/ou fruto específico. Contudo, o uso destes sensores em fazendas de um modo geral é dificultado pela necessidade de um conhecimento técnico especializado de instalação, utilização e manutenção constantes, situação geralmente encontrada em centros de pesquisa. Uma alternativa facilitadora seria a obtenção de medidas de molhamento foliar via variáveis meteorológicas, comumente obtidas de estações meteorológicas automáticas.

¹ Corpúsculos reprodutivos de fungos e algumas bactérias.

3.3 Objetivo do trabalho

O objetivo deste trabalho foi desenvolver e validar um sistema computacional baseado em RNAs, destinado ao tratamento de dados climáticos, visando reconhecer padrões de comportamento de variáveis meteorológicas em relação ao molhamento foliar por orvalho.

4 REVISÃO DE LITERATURA

Este capítulo tem dois focos distintos, o enfoque computacional (paradigma das RNAs) e o agrometeorológico (orvalho e molhamento foliar) os quais correspondem, respectivamente, à metodologia computacional aplicada e ao domínio de aplicação.

4.1 Redes Neurais Artificiais

Na literatura especializada (Azevedo et al., 2000; Barreto, 1997; Braga et al., 2000; Fischer et al., 2001; Haykin, 2001; Medeiros, 2003; Osório & Bittencourt, 2000; Neurais, 2003; Nievola, 2004; Sarle, 2004; Zell et al., 1998), várias são as definições que os autores atribuem às *Redes Neurais Artificiais* ou, simplesmente, *Redes Neurais* (RNs), ou ainda *métodos conexionistas*. Todos os autores citados concordam que as RNAs consistem em uma metodologia para solucionar problemas de IA, a partir de um sistema que possui circuitos que simulam o cérebro humano, inclusive seu comportamento, ou seja, aprendendo, errando e fazendo descobertas. São técnicas computacionais que apresentam um modelo inspirado na estrutura neural de organismos inteligentes, que adquirem conhecimento através da experiência.

A habilidade das RNAs em solucionar problemas complexos e variados tem as tornado uma abordagem interessante que pode ser aplicada em diversas áreas

de engenharia e ciências (Silva, 2001b). Os trabalhos têm sido motivados desde o começo, pelo reconhecimento de que o cérebro humano processa informações de uma forma diferente do computador digital convencional. O cérebro é um computador (sistema de processamento de informação) altamente complexo, não-linear e paralelo. Ele tem a capacidade de organizar seus constituintes estruturais, conhecidos por neurônios, de forma a realizar certos processamentos (reconhecimento de padrões, percepção, e controle motor) mais rapidamente que o mais rápido computador digital hoje existente (Haykin, 2001).

Apesar da complexidade das redes neurais não permitir uma única definição, adota-se a de Haykin (2001), que descreve as principais características do que seja uma RNA:

Uma rede neural é um processador maciçamente paralelamente distribuído constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro em dois aspectos:

- 1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.*
- 2. Forças de conexão entre neurônios, conhecidas como pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.*

A propriedade mais importante das redes neurais é a habilidade de aprender a partir de seu ambiente e com isso melhorar seu desempenho, isso é feito através de um processo de aprendizagem, que é um processo iterativo de ajustes aplicado a seus pesos sinápticos, o algoritmo de treinamento. O aprendizado ocorre quando a rede neural atinge uma solução generalizada para uma classe de problemas.

Essas características das redes neurais resultam em algumas atrativas habilidades que incluem (Balestrassi, 2000):

- encontrar relacionamentos entre conjuntos de dados aparentemente não correlacionados e depois construir um conjunto de informações a partir deles;
- encontrar relacionamentos entre os dados que analistas humanos não podem descobrir devido suas limitações intrínsecas quando, por exemplo, existe uma grande quantidade de pontos;

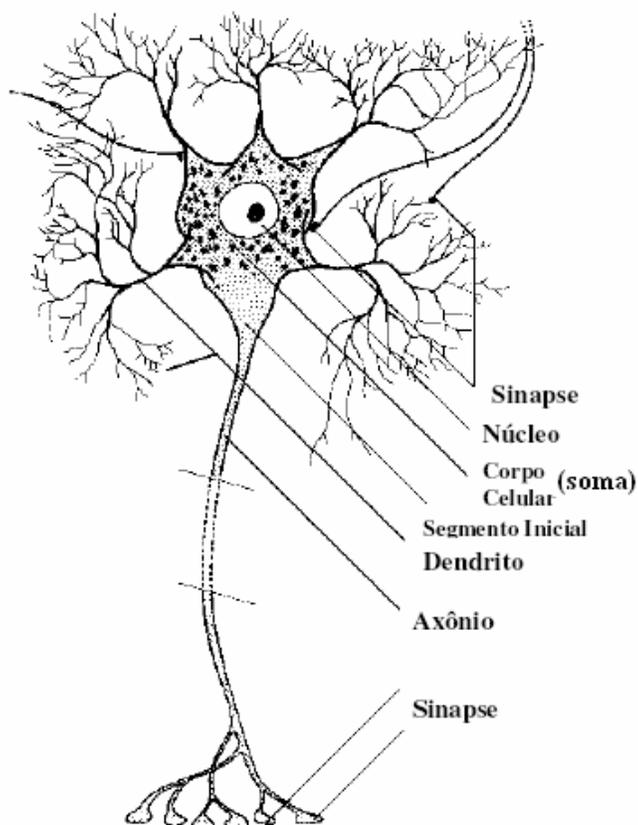
- ter a habilidade e velocidade de analisar os conjuntos de dados em uma fração de tempo inferior aos seres humanos.

Nas seções a seguir estão descritos os principais fundamentos das RNAs, no que se refere ao neurônio artificial, o mecanismo de funcionamento e a forma de representar geometricamente uma RNA.

4.1.1 O neurônio artificial e a rede neural

Considerando que o sistema nervoso é composto por bilhões de células nervosas, a idéia é que a RNA também seja constituída por unidades de processamento que simulem o funcionamento de um neurônio biológico. Estes módulos devem estar interconectados e funcionar de acordo com os elementos em que foram inspirados, recebendo, retransmitindo informações e gerando conhecimento.

Em 1911, Santiago Ramón y Cajal introduziu a idéia de neurônios como estrutura básica do cérebro (Figura 1), cujo funcionamento pode ser resumido basicamente do seguinte modo: os dendritos recebem sinais de entrada oriundos de outras células, o elemento soma e os próprios dendritos integram e processam estes sinais, a informação é transmitida ao longo do axônio para outros neurônios por meio das sinapses, que são as ligações entre os dendritos de neurônios diferentes.



Fonte: Osório & Bittencourt (2000)

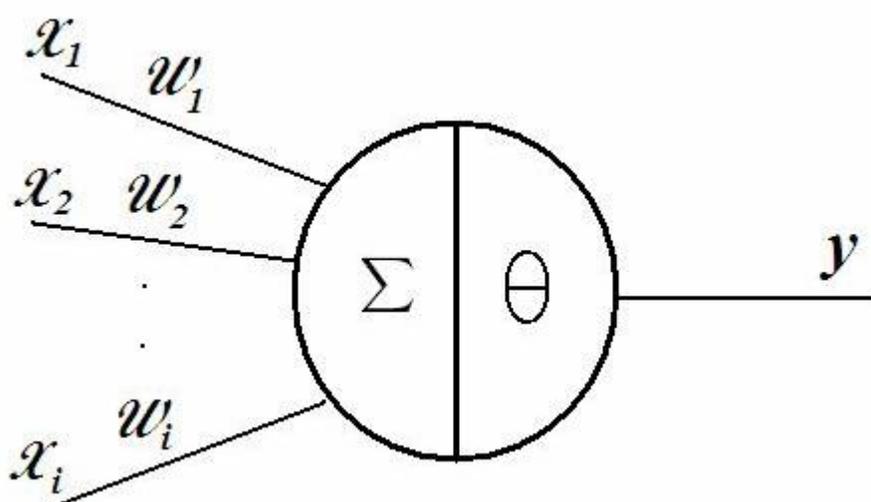
Figura 1 – Esquema de um neurônio biológico

Em 1943, o psiquiatra e neurofisiologista McCulloch e o matemático Pitts (McCulloch & Pitts, 1943) propuseram matematicamente, de forma simplificada, o funcionamento de um neurônio biológico. No modelo de McCulloch & Pitts (Figura 2), o neurônio possui i entradas (equivalente aos dendritos) X_1, X_2, \dots, X_i e apenas uma saída (equivalente ao axônio) y . Para simular a sinapse, cada entrada do neurônio tem um peso W_1, W_2, \dots, W_i cujos valores podem ser positivos (excitatórios) ou negativos (inibitórios). Os pesos têm como finalidade armazenar o conhecimento e determinar a intensidade com que cada entrada contribuirá no resultado do neurônio. O corpo celular é emulado simplesmente somando os valores do produto de suas entradas com seus respectivos pesos, $X_i W_i$ e, se a soma for maior ou igual ao seu limiar (*threshold*) a sua saída, y , é ativada com valor I .

Resumindo o neurônio será ativo quando:

$$\sum_{i=1}^m x_i w_i \geq \Theta \quad (1)$$

onde m é o número de entradas do neurônio, w_i é o peso associado à entrada x_i e Θ é o limiar do neurônio, conforme ilustrado na Figura 2.



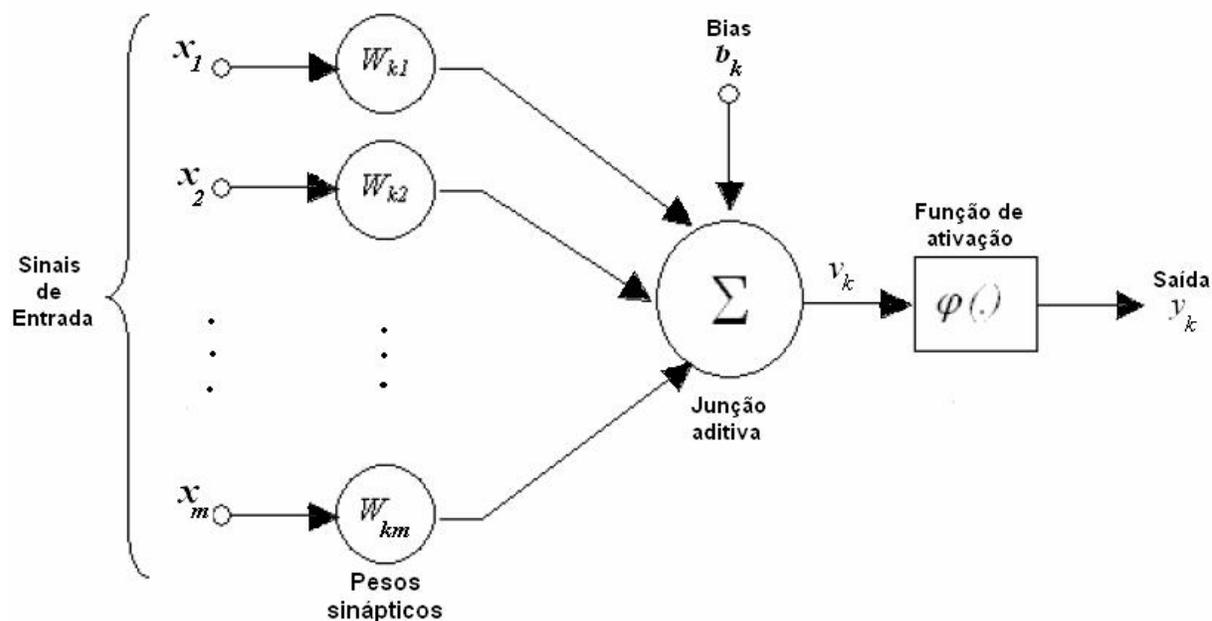
Fonte: Braga et al. (2000)

Figura 2 – Modelo matemático de um neurônio biológico (McCulloch & Pitts, 1943)

No decorrer dos anos seguintes, vários pesquisadores propuseram variações do modelo do neurônio McCulloch & Pitts. A principal diferença entre a idéia original e suas variações encontram-se na função que determina o estado de ativação da saída do neurônio (função ativação) e o acréscimo de um *bias*. *Bias* corresponde a um neurônio ou a uma entrada especial que serve para aumentar os graus de liberdade, permitindo uma melhor adaptação por parte da rede neural, ao conhecimento a ela fornecido; pode ser considerado como um ajuste fino.

4.1.2 Modelo de neurônio

Nesta seção, descreve-se o modelo de neurônio que conforme Haykin (2001) identifica a base para projetos de RNAs, sendo composto por três elementos básicos (Figura 3).



Fonte: Haykin (2001)

Figura 3 – Modelo de neurônio base para projetos de RNA

1. Conjunto de *sinapses* ou *elos de conexão*, cada uma se caracteriza por seu peso ou força própria. Um sinal x_j na entrada da sinapse j conectado a um neurônio k é multiplicado pelo peso sináptico w_{kj} , onde o índice k refere-se ao neurônio em questão e o índice j refere-se ao terminal de entrada da sinapse.
2. *Junção aditiva* – somador dos sinais de entrada, ponderados pelos respectivos pesos sinápticos. Estas funções constituem um combinador linear.
3. *Função de ativação* – tem por finalidade restringir a amplitude de saída de um neurônio, pode ser também referenciada como função restritiva já que restringe (limita) o intervalo permissível de amplitude do sinal de saída a um valor finito. Tipicamente, o

intervalo normalizado da amplitude de saída de um neurônio é escrito como o intervalo unitário fechado $[0,1]$ ou $[-1,1]$.

Neste modelo, pode ser notado o acréscimo de um *bias*, que tem o efeito de aumentar ou diminuir a entrada líquida da função de ativação, dependendo se ele é positivo ou negativo, respectivamente, identificado por b_k e representado matematicamente na equação 3.

Ao efetuar-se uma breve comparação com o modelo McCulloch & Pitts (Figura 2), o neurônio possui entradas similares, pesos sinápticos, e o somador dos valores do produto de suas entradas com os respectivos pesos. Contudo, a função de ativação difere por somente permitir que a saída de um neurônio seja ativada com valor 1 , ou não ativada com valor 0 , assim como, a não existência do bias.

Matematicamente, um neurônio k é descrito com as seguintes equações:

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (2)$$

$$y_k = \varphi(u_k + b_k) \quad (3)$$

onde $x_1, x_2, x_3, \dots, x_m$, são sinais de entrada; $w_{k1}, w_{k2}, w_{k3}, \dots, w_{km}$, são os pesos sinápticos do neurônio k ; u_k é a saída do combinador linear devido aos sinais de entrada; b_k é o bias; $\varphi(\cdot)$ é a função de ativação; e y_k é o sinal de saída do neurônio. O uso do bias b_k tem o efeito de aplicar uma transformação afim à saída u_k do combinador linear, demonstrado na equação 4.

$$v_k = u_k + b_k \quad (4)$$

4.1.3 Tipos de função de ativação

A função de ativação definida pelo termo j (.) é responsável pela saída do neurônio, conforme convenção já citada anteriormente. Identificam-se três tipos básicos de funções de ativação (Haykin, 2001):

1. *Função de Limiar* – definida pela equação 5 e seu gráfico é ilustrado na Figura 4.

$$\varphi(v) = \begin{cases} 1 & \text{se } v \geq 0 \\ 0 & \text{se } v < 0 \end{cases} \quad (5)$$

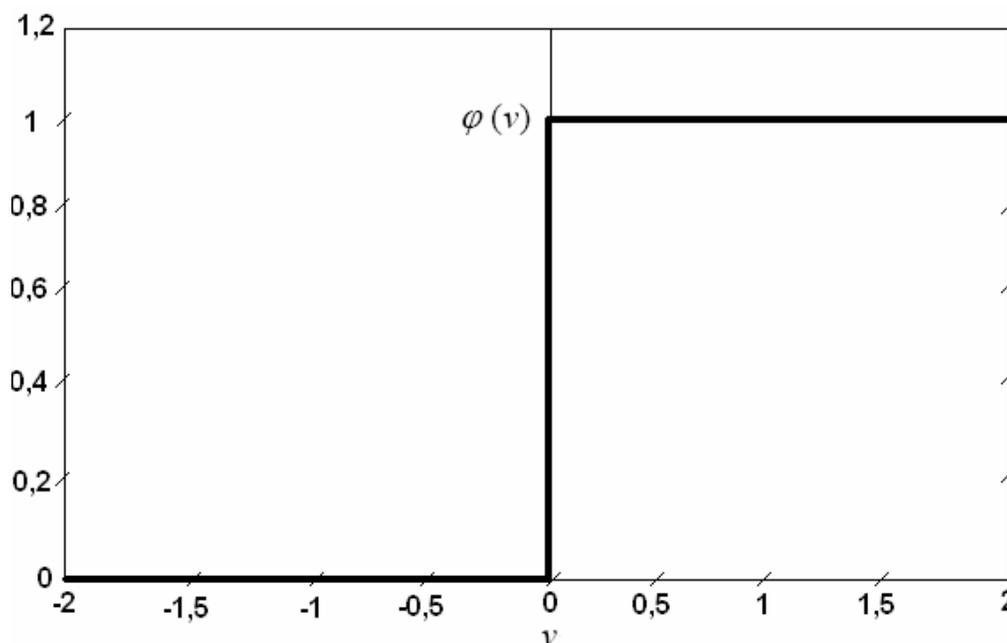


Figura 4 – Gráfico função de limiar

2. *Função Linear por Partes (piecewise)* – definida pela equação 6, ilustrada no gráfico da Figura 5. Assume-se que o fator de amplificação dentro da região linear de operação é a unidade. Esta forma de ativação pode ser vista como uma aproximação de um amplificador não-linear. As duas situações a seguir podem ser vistas como formas especiais da função linear por partes:

- Se a região de operação é mantida sem entrar em saturação, surge um combinador linear.
- A função linear por partes se reduz à função linear, se o fator de amplificação da região linear é feito infinitamente grande.

$$\varphi(v) = \begin{cases} 1, & v \geq +\frac{1}{2} \\ v, & +\frac{1}{2} > v > -\frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases} \quad (6)$$

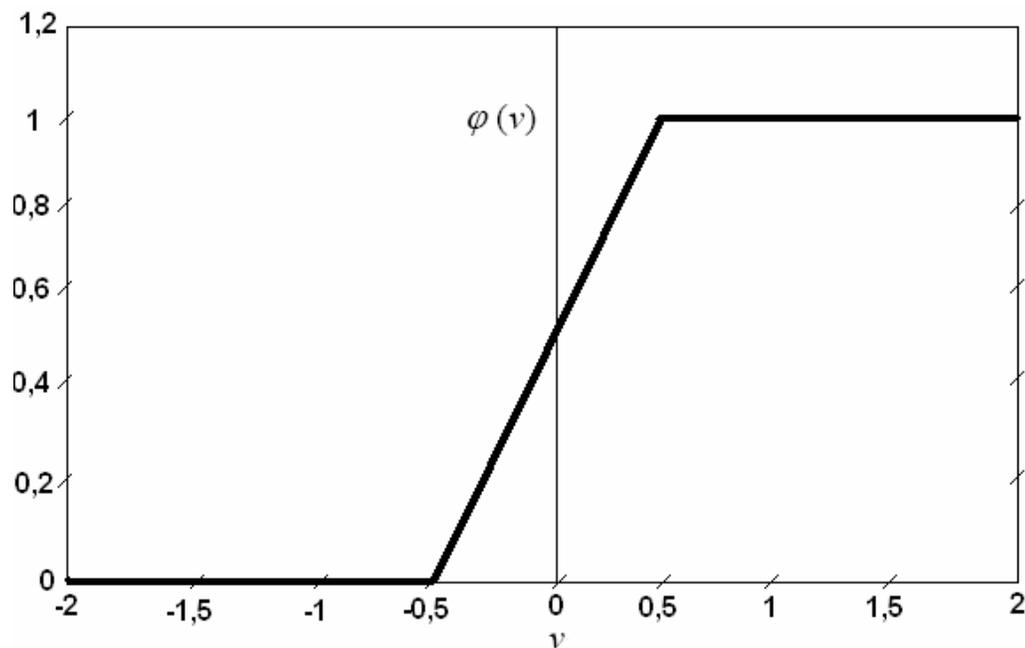


Figura 5 – Gráfico função linear por partes

3. *Função Sigmóide* – é o modelo de função de ativação mais utilizado em projetos de redes neurais. O gráfico desta função, ilustrado na Figura 6, forma um **S**, o que fornece um comportamento de balanceamento adequado entre linear e não-linear, sendo ainda

estritamente crescente. Um exemplo de função sigmóide é a função logística, definida pela equação 7.

$$\varphi(v) = \frac{1}{1 + \exp(-av)} \quad (7)$$

onde, a representa o parâmetro de inclinação da função sigmóide. Variando-se a , obtém-se diferentes inclinações desta função, com ilustrado na Figura 6.

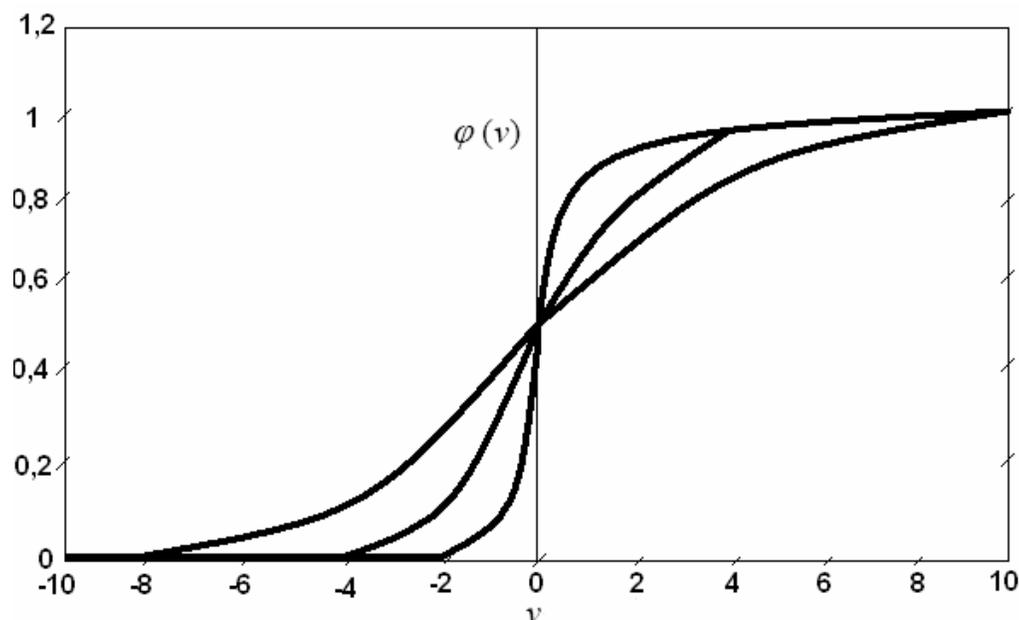


Figura 6 – Gráfico função sigmóide

Ante as abordagens até aqui desenvolvidas, em síntese, pode-se definir uma RNA como sendo um sistema de neurônios ligados por conexões sinápticas e dividido em **neurônios de entrada**, que recebem estímulos do meio externo, podendo haver, **neurônios internos ou ocultos** (*hidden*), organizados em camadas (*layers*) e **neurônios de saída**, que se comunicam com o exterior.

Um modo de representar a estrutura genérica de uma RNA é através de um grafo direcionado², pelo fato que uma rede neural é uma estrutura de processamento de

² Um grafo $D=(V,E)$ é dito **direcionado ou dirigido** (dígrafo) se ele é constituído de um conjunto finito não vazio V (os vértices) e um conjunto E (as arestas orientadas) de pares ordenados de vértices. Portanto, em um dígrafo cada aresta (v,w) possui uma única direção de v para w (Szwarcfiter, 1986).

informação distribuída paralelamente. Nesta representação, os neurônios podem ser simplificados como os nós deste grafo e as arestas (conexões) funcionam como as sinapses, sendo a elas atribuídos os pesos. Também é possível representar camadas de uma RNA, que compõem a sua estrutura: camada de entrada, camada(s) oculta(s) ou intermediária(s) e camada de saída, conforme ilustrado na Figura 7.

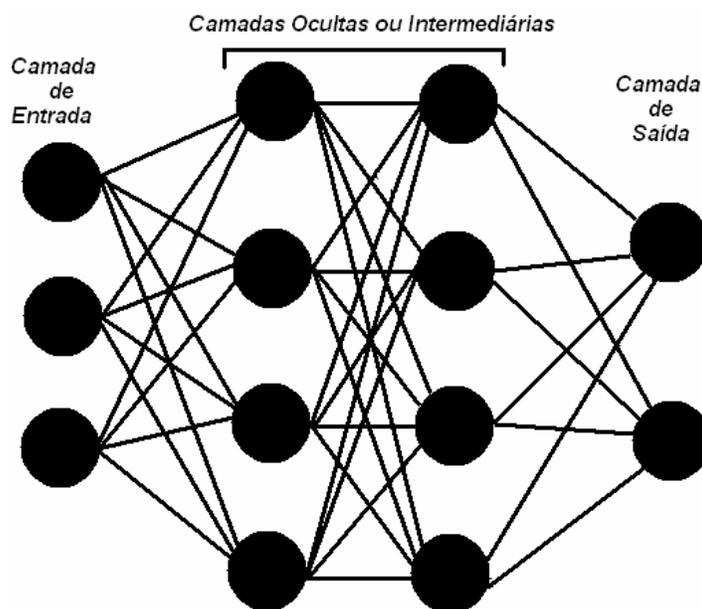


Figura 7 – RNA representada em forma de Grafo.

4.1.4 Arquiteturas de RNAs

Um fator que define diretamente a arquitetura ou topologia de uma RNA é o problema que será tratado pela rede e, neste contexto, devem ser considerados os seguintes componentes: quantidades de camadas, números de neurônios em cada camada e tipo de conexão entre os neurônios (Braga et al., 2000).

A seguir são destacados alguns dos principais exemplos de arquiteturas de RNA, que demonstram as possíveis maneiras de conectar os componentes de uma RNA, quanto ao número de camadas e tipos de conexões entre os neurônios (Haykin, 2001).

Quanto ao número de camadas:

- *Redes com camada única* – forma mais simples de uma rede em camadas, surge quando tem-se uma camada de entrada que se projeta para a camada de saída, como mostrado na Figura 8 (A) e (C);
- *Redes com múltiplas camadas* – distinguem-se de redes com camada única, pela presença de uma ou mais camadas ocultas. A função das camadas ocultas é extrair informações das amostras, conforme ilustrado na Figura 8 (B).

Quanto aos tipos de conexões entre neurônios:

- *Feedforward ou acíclica* – a saída do neurônio na i -ésima camada não pode ter entradas com neurônios em camadas de índice menor ou igual a i , como mostrado na Figura 8 (A) e (B);
- *Feedback ou cíclica* – a saída do neurônio na i -ésima camada tem entradas com neurônios em camadas de índice menor ou igual a i , como mostrado na Figura 8 (C).

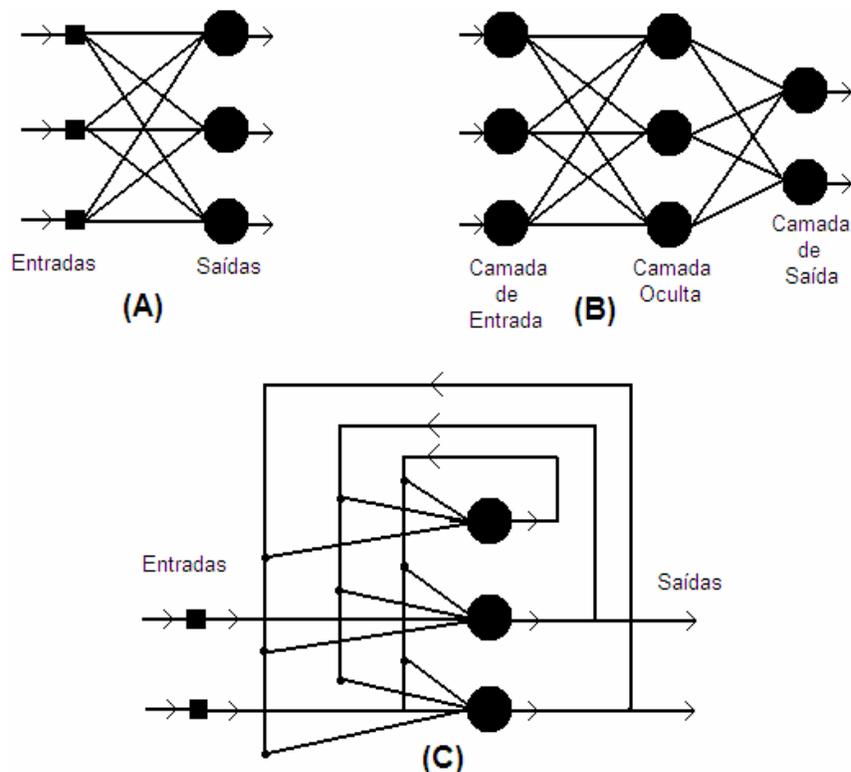


Figura 8 – Exemplos de arquiteturas de RNAs

O trabalho desta tese foi desenvolvido com RNAs de arquitetura de múltiplas camadas acíclica (*Feedforward*). O motivo da escolha desta arquitetura de RNA está relacionado ao método de aprendizado adotado – aprendizado supervisionado, conforme será abordado em seções posteriores e no capítulo de material e métodos.

4.1.5 Tarefas de aprendizagem

As RNAs podem ser aplicadas a diferentes tipos de tarefas, tais como: o reconhecimento de padrões, associação de padrões, transformação de dados, predição, controle de processos, aproximação de funções, entre outras. Exemplos de aplicações podem ser encontrados em UCI-ML, 2004 - Osório & Bittencourt (2000) e em Zuben (2003).

Neste trabalho, a tarefa de aprendizagem adotada é o reconhecimento de padrões (classificação), em função do principal objetivo ser a classificação do molhamento foliar em forma de índices de molhamento.

A classificação é uma tarefa de previsão onde um conjunto de atributos previsores (características) é usado para prever um atributo meta (rótulo). A tarefa consiste em descobrir o relacionamento entre os atributos previsores e o atributo meta, usando o conjunto de treinamento.

Segundo Haykin (2001), o reconhecimento de padrões é formalmente definido como o processo pelo qual um padrão/sinal recebido é atribuído a uma classe dentre um número predeterminado de classes (categorias). Uma RNA realiza o reconhecimento de padrões passando inicialmente por uma sessão de treinamento, durante a qual se apresenta repetidamente à rede um conjunto de padrões de entrada junto com a categoria à qual cada padrão particular pertence. Mais tarde, apresenta-se à rede um novo padrão que não foi visto antes, mas que pertence à mesma população de padrões utilizada para treinar a rede. A rede é capaz de identificar a classe daquele padrão particular em virtude da informação que ela extraiu dos dados de treinamento. O reconhecimento de padrões realizado por uma RNA é de natureza estatística, com os padrões sendo representados por pontos em um espaço de decisão multidimensional. O espaço de decisão é dividido em regiões, cada uma das quais associada a uma classe. As fronteiras de decisão são determinadas pelo processo de treinamento. A

construção dessas fronteiras é tornada estatística pela variabilidade inerente que existe dentro das classes e entre as classes.

Conforme já abordado, uma das propriedades mais importantes das RNAs é o seu aprendizado, e a escolha de um algoritmo de aprendizagem particular é influenciada pela tarefa de aprendizagem que uma rede neural deve executar (Haykin, 2001).

4.1.6 Processo de aprendizagem

Osório & Bittencourt (2000) definiram aprendizado natural como a capacidade de se adaptar, de modificar e melhorar seu comportamento e suas respostas sendo, portanto, uma das propriedades mais importantes dos seres ditos inteligentes, sejam eles humanos ou não.

Do ponto de vista computacional, o aprendizado de máquina tenta fazer com que os programas de computador “aprendam” com os dados que eles “estudam”, tal que esses programas tomem decisões diferentes baseadas nas características dos dados estudados, usando a estatística para os conceitos fundamentais, e adicionando heurística avançada da IA aos algoritmos para alcançar os seus objetivos (Guimarães, 2005; Guimarães, 2001).

Em se tratando de RNA, segundo Haykin (2001), aprendizagem é um processo pelo qual os parâmetros livres (pesos sinápticos) de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida. O tipo de aprendizagem é determinado pela maneira na qual a modificação dos parâmetros ocorre.

Esta definição do processo de aprendizagem implica a seguinte seqüência de eventos:

1. A rede neural é *estimulada* por um ambiente;
2. A rede neural *sofre modificações* nos seus parâmetros livres como resultado desta estimulação;
3. A rede neural *responde de uma maneira nova* ao ambiente, devido às modificações ocorridas na sua estrutura interna.

O aprendizado conexionista é em geral um processo gradual e iterado, onde os pesos são modificados várias vezes, pouco a pouco, seguindo-se uma regra de

aprendizado que estabelece a forma como estes pesos são alterados. O aprendizado é realizado utilizando-se um conjunto de dados de aprendizado disponível (base de exemplos). Cada iteração deste processo gradativo de adaptação dos pesos de uma rede neural, onde é feita uma apresentação completa do conjunto de dados, é chamada de época ou ciclo de aprendizado. Os métodos de aprendizado neural podem ser divididos em três grandes classes, segundo o grau de controle dado ao usuário (Osório & Bittencourt, 2000).

- *Aprendizado supervisionado* – o usuário dispõe de um comportamento de referência preciso, que ele deseja ensinar a rede. Sendo assim, a rede deve ser capaz de medir a diferença entre seu comportamento atual e o comportamento de referência, e então corrigir os pesos de maneira a reduzir este erro (desvio de comportamento em relação aos exemplos de referência). Exemplo de aplicação: reconhecimento de caracteres em uma aplicação do tipo OCR (*Optical Character Recognition* – Osório apud Osório & Bittencourt, 2000).
- *Aprendizado semi-supervisionado* – o usuário possui apenas indicações imprecisas (por exemplo: sucesso/insucesso da rede) sobre o comportamento final desejado. As técnicas de aprendizado semi-supervisionado são chamadas também de aprendizado por reforço (*reinforcement learning* - Sutton & Barto apud Osório & Bittencourt, 2000). Exemplo: aplicações em robótica autônoma, onde supondo uma situação hipotética, sabe-se que seguir em frente não é possível, pois existe um obstáculo, mas em compensação não se tem uma medida numérica que indique para que lado seguir e exatamente como se deve proceder para desviar deste obstáculo.
- *Aprendizado não-supervisionado* – os pesos da rede são modificados em função de critérios internos, tais como, a repetição de padrões de ativação em paralelo de vários neurônios. O comportamento resultante deste tipo de aprendizado é usualmente comparado com técnicas de análise de dados empregadas na estatística (*clustering*). Exemplo: diferenciar tomates de laranjas, sem ter, no entanto os exemplos com a sua respectiva classe etiquetada (*self-organizing feature maps* – Kohonen apud Osório & Bittencourt, 2000).

Rohn & Mine (2003) destacaram que, para o aprendizado não é necessário um conhecimento detalhado sobre as relações entre as variáveis envolvidas no problema, contudo, as redes necessitam de uma quantidade considerável de dados históricos para que consigam extrair satisfatoriamente as características relevantes existentes no conjunto de dados. Se treinada corretamente, a rede é capaz não somente de aproximar qualquer função, mas também de *generalizar*, proporcionando saídas corretas para entradas não apresentadas anteriormente.

Generalização é a capacidade de um modelo de aprendizado responder corretamente aos exemplos que lhe são apresentados, sendo que, estes exemplos não devem estar presentes na base de aprendizado. Ou seja, um modelo que tem uma boa generalização é aquele modelo que responde corretamente aos exemplos contidos na base de aprendizado, mas, também a outros exemplos diferentes daqueles da base de aprendizado, e que estão contidos em uma base de teste.

A capacidade de generalizar é a principal característica buscada nas tarefas que envolvem aprendizado. Contudo, na tentativa de generalizar, uma RNA pode se especializar demasiadamente em relação aos exemplos contidos na base de aprendizado. Este tipo de comportamento gera um problema de aprendizado conhecido como super-aprendizado (*over-training / over-fitting*), que pode ocorrer quando na arquitetura da RNA o número de camadas e/ou neurônios for muito grande. Outra situação, é a rede não conseguir generalizar, gerando o problema chamado de sub-ajuste (*under-fitting*), o qual pode ocorrer quando o número de camadas e/ou neurônios for muito pequeno.

4.1.7 Modelos de aprendizagem

Os modelos de aprendizagem estão relacionados ao ajuste dos pesos e podem ser implementados por diferentes métodos, segundo o tipo de regra de aprendizado que for empregado. As regras de aprendizado mais usadas são (Ribeiro, 2003; Osório & Bittencourt, 2000):

- *Aprendizagem por Correção de Erro* – a rede é estimulada por um vetor de entrada e sua saída desejada. A regra de ajuste consiste em achar o erro

subtraindo a resposta desejada com a resposta da rede, calculando o gradiente descendente da função do erro.

- *Aprendizagem Baseada em Memória* – usa-se arquitetura de redes alimentadas com camada única de modo supervisionado. A maioria das amostras de treino é armazenada em uma grande memória de exemplos de entrada-saída. Um exemplo simples de aprendizagem baseada em memória é a regra do vizinho mais próximo, onde a rede classifica de acordo com a distância euclidiana entre os vetores de treino.
- *Aprendizagem competitiva* – na camada de saída da rede, após receber as características dos objetos na camada de entrada, os neurônios competem entre si, e o neurônio que tiver o maior valor discriminante, se torna vencedor.

As regras de aprendizagem se constituem como elementos básicos para o projeto de redes neurais artificiais.

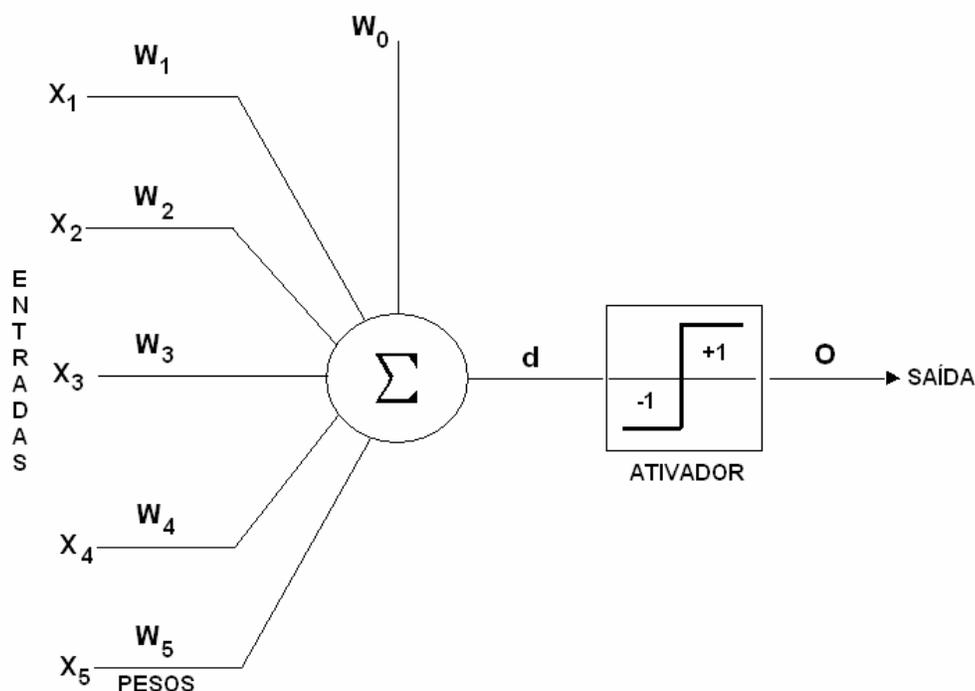
4.1.8 Perceptron de camada única

A partir do modelo de neurônio artificial de McCulloch & Pitts (McCulloch & Pitts, 1943), em 1957 Frank Rosenblatt (1958) propôs o perceptron como o primeiro modelo para aprendizagem supervisionada. O perceptron é a forma mais simples de uma RNA usada para classificação de padrões ditos linearmente separáveis, ou seja, padrões que se encontram em lados opostos de um hiperplano (Haykin, 2001).

Rosenblatt provou que, se os padrões (vetores) usados para treinar o perceptron são retirados de duas classes linearmente separáveis, então o algoritmo do perceptron converge e posiciona a superfície de decisão na forma de um hiperplano entre as duas classes. A prova de convergência é conhecida como o *teorema de convergência do perceptron*.

Basicamente, o perceptron consiste de um único neurônio limitado a realizar classificação de padrões com apenas duas classes (hipóteses), com pesos sinápticos ajustáveis e um *bias*.

Na Figura 9, está ilustrado o modelo perceptron, onde foram adotadas 5 entradas representadas por X_1, \dots, X_5 , cada entrada do neurônio tem um peso (sinapse) W_1, \dots, W_5 , W_0 representa o bias, d é a saída intermediária e O corresponde a saída ativada da rede.



Fonte: Medeiros (2003)

Figura 9 – Modelo perceptron

Os pesos têm como finalidade armazenar o conhecimento e determinar a intensidade com que cada entrada contribuirá no resultado do neurônio. Para emular um neurônio biológico, é realizado um somatório dos valores do produto de suas entradas com seus respectivos pesos, $X_i W_i$, adicionando-se o bias, cujo resultado obtido corresponde à saída intermediária d , posteriormente este valor é submetido à função de ativação, ou seja,

$$d = X_1 W_1 + X_2 W_2 + X_3 W_3 + X_4 W_4 + X_5 W_5 + W_0 \quad (8)$$

Para calcular a saída ativada O (*output*) a função de ativação tem por finalidade avaliar o resultado da saída intermediária (d) do seguinte modo: se o resultado

obtido em d for maior ou igual a 0 (zero), O (saída ativada) será igual a +1 (um); se d for menor que 0 (zero), O será igual a -1 (menos 1).

No sentido de estabelecer uma abordagem prática de uso do modelo perceptron, a seguir, desenvolve-se um exemplo simples de aplicação, adaptado de Ramos (2003), cujo domínio de aplicação é a classificação de frutos.

Segundo Ramos (2003), no processo de classificação artificial de frutos, faz-se necessário o desenvolvimento de modelos que permitam estabelecer o relacionamento entre a entrada de padrões de classificação (entradas), análise/processamento dessa informação e convergência para uma saída de classificação definida (saída). A rede neural deve aprender a reconhecer padrões de entrada e definir a saída segundo classes definidas, ou seja, dado um determinado padrão de entrada, escolher em que categoria esse se enquadra melhor.

Para exemplificar, os padrões de entrada considerados são valores que correspondem às propriedades físicas do fruto, peso e diâmetro. O modelo perceptron deve aprender a relação entre os padrões de entrada e as classes desejadas na saída. Hipoteticamente, considera-se que, em média, os frutos que tenham no máximo 5 cm de diâmetro sejam os maiores e os próximos de 1 cm sejam os menores, havendo evidentemente, tamanhos intermediários dentro desta faixa. Quanto ao peso, frutos que tenham 50 gramas de peso, são os maiores e os de 5 gramas, os menores. Do mesmo modo, como em relação ao diâmetro, há pesos diferentes dentro desta faixa. Os perceptrons terão que classificar os frutos com certo peso e diâmetro de acordo com duas classes, A ou B.

Assume-se que os pesos foram calculados mediante um processo de treinamento e que se têm dois pesos para cada perceptron, conforme Tabela 1.

Tabela 1 – Treinamento de perceptron para classificação de frutos

Perceptron	W_1 (diâmetro)	W_2 (peso)	W_0
1	0,021	0,022	-1
2	-0,023	-0,019	1

Uma vez que os pesos já tenham sido definidos, pode-se classificar um conjunto de frutos que supostamente não estiveram no conjunto de treinamento (conjunto de

teste), calculando as saídas de cada perceptron e definindo que, se a saída ativada O para o perceptron 1 for +1, o fruto pertence a classe A, se a saída ativada O para o perceptron 2 for +1, o fruto é de classe B, como pode ser observado na Tabela 2.

Tabela 2 – Conjunto de dados para classificação de frutos

Fruto	X_1 (diâmetro)	X_2 (peso)	D_1	d_2	O_1	O_2	Classe
1	15	7	-0,53	0,52	-1	1	B
2	30	15	-0,04	0,02	-1	1	B
3	50	25	0,60	-0,62	1	-1	A
4	20	8	-0,40	0,38	-1	1	B
5	40	22	0,32	-0,33	1	-1	A

Neste exemplo é demonstrado como se pode utilizar uma RNA do tipo perceptron para obter a classificação de padrões. Foram consideradas duas classes e duas variáveis de entrada para os perceptrons. A partir destes dados, pode-se representar os perceptrons num sistema cartesiano com as variáveis diâmetro e peso para cada eixo, e representar os pontos do conjunto de teste da Tabela 2, conforme o gráfico da Figura 10.

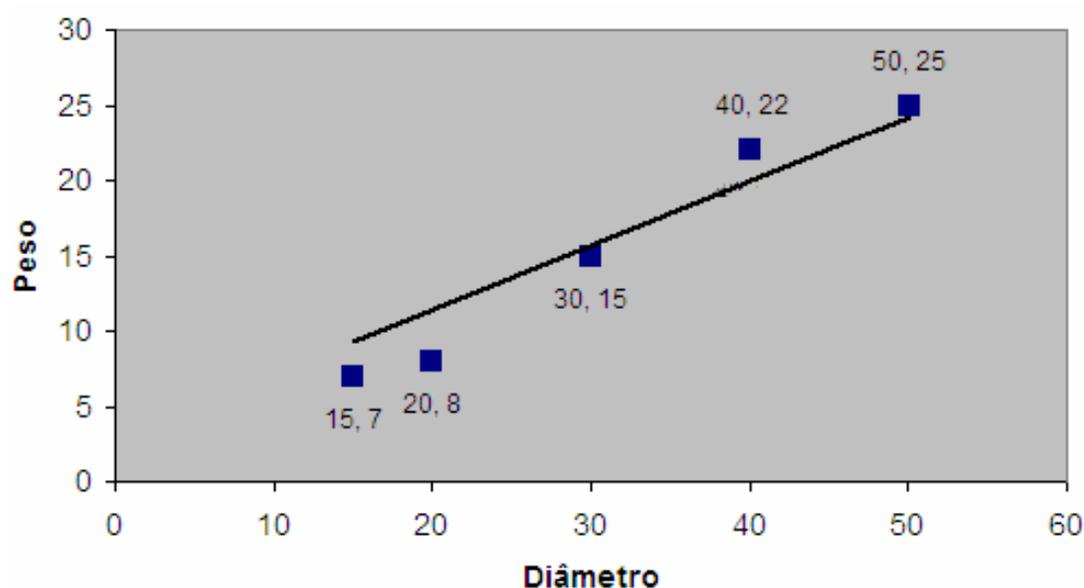


Figura 10 – Gráfico perceptron 1 e 2 – classificação de frutos

A classificação dos padrões desta rede exemplo, de acordo com os valores dos pesos assumidos anteriormente, pode ser representada através de retas cortando o gráfico (Figura 10), cujas linhas podem ser interpretadas como um mapa das regiões de decisão no espaço de sinal bidimensional, abrangido pelas 2 variáveis de entrada X_1, X_2 , onde, um ponto (X_1, X_2) que se encontra acima da linha de fronteira é atribuído à classe A, e um ponto (X_1, X_2) que está abaixo da linha de fronteira é atribuído à classe B. O efeito do bias W_0 é deslocar a fronteira de decisão em relação à origem. Deste modo, conforme os resultados da Tabela 2, os frutos 3 e 5 pertencem a classe A e os 1, 2 e 4 à classe B.

Percebe-se neste exemplo que, para o perceptron funcionar adequadamente, as duas classes estabelecidas para o problema devem estar linearmente separáveis, ou seja, os padrões a serem classificados devem estar suficientemente separados entre si para assegurar que a superfície de decisão consista de um hiperplano (neste caso uma linha reta) como fronteira de decisão (Haykin, 2001).

Haykin (2001) e Medeiros (2003) destacaram as limitações do perceptron para problemas que não são linearmente separáveis. Uma limitação é quando um ponto (X_1, X_2) que se encontra entre as linhas de fronteira, ou seja, fora das classes definidas. Uma possível solução seria aumentar o conjunto de treinamento para refinar mais os pesos obtidos, contudo, pode não haver uma divisão bem definida das classes, o que pode provocar erros de classificação, ou ainda, as características do problema considerado, não permite uma divisão clara entre duas ou mais classes, mesmo através de linhas bem definidas.

Outra situação é o fato de determinado problema ser tratado com três ou mais classes, o que também caracteriza um problema não separável linearmente. Para problemas não separáveis linearmente, utiliza-se mais camadas na rede, para obter uma divisão do espaço de maneira mais precisa e neste contexto as redes perceptron de múltiplas camadas são utilizadas.

4.1.9 Perceptron de múltiplas camadas

Estas RNAs são normalmente chamadas de perceptron de múltiplas camadas ou *multilayer perceptron* (MLP), as quais se caracterizam como uma generalização do perceptron de camada única (Haykin, 2001).

Considerando como estrutura básica da rede o modelo de neurônio perceptron (Figura 9), tipicamente uma rede MLP possui uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Na Figura 11, está ilustrada a arquitetura de uma rede MLP com uma camada de entrada composta por três neurônios, duas camadas ocultas ou intermediárias, cada uma composta por quatro neurônios, e uma de saída, com dois neurônios. Percebe-se neste modelo, que qualquer neurônio em qualquer camada da rede está conectado a todos os neurônios das camadas anteriores.

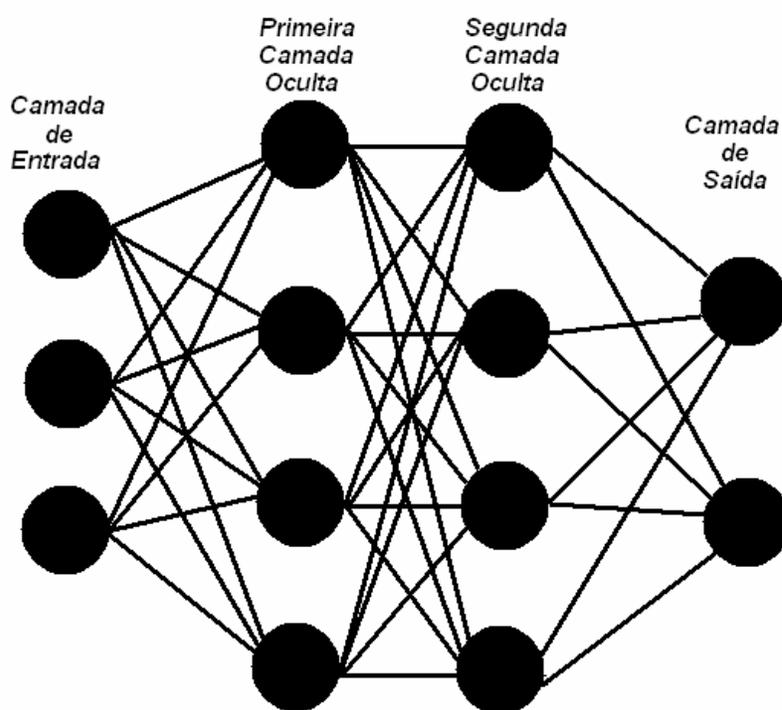


Figura 11 – Arquitetura de RNA MLP com duas camadas ocultas

Os perceptrons de múltiplas camadas têm sido aplicados com sucesso para resolver diversos problemas difíceis, através do seu treinamento de forma supervisionada com retropropagação de erro (Haykin, 2001). Exemplos dessas aplicações são citados em Braga et al. (2000), a saber: reconhecimento de caracteres, previsão do comportamento de ações na bolsa, verificação de assinaturas, segurança em transações com cartões de crédito, diagnóstico médico entre outros. Alguns trabalhos específicos estão descritos em Abelém (1994), Andrade et al. (2003), Bocanegra (2002), Dias et al. (2004), Dias et al. (2004a),

Diverio et al. (2003), Franco & Martins (1999), Freitas et al. (2002), Galo (2000a), Galo et al. (2000b), Giovanini & Coury (1999), Gleriani (2004), Gonçalves (2003), Mathias et al. (2004), Medeiros (1999), Muller & Fill (2003), Pernomian & Trindade al. (2003), Rohn et al. (2003), Ramos (2003), Ribeiro (2003), Santos et al. (1999), Segatto et al. (2003), Senger & Caldas Junior (2001), Tinós (1999), Tápia et al. (2000), Weinert & Lopes (2003).

Rumelhart, Hinton e Williams (Rumelhart et al., 1986) apresentaram em 1986 a descrição do Algoritmo de Retropropagação de erro (*error back-propagation*) para arquitetura do perceptron de múltiplas camadas. Este algoritmo é baseado na regra de aprendizagem por correção de erro, tema da próxima seção.

4.1.10 Algoritmo de retropropagação de erro (*error back-propagation*)

Conforme abordagens encontradas em Haykin (2001), Medeiros (2003), Paula (2000), Pessoa & Freire (1998) e Ribeiro (2003), nas linhas seguintes descreve-se as principais características do algoritmo de aprendizagem *back-propagation* (Backprop) para o treinamento de redes MLP.

Enquanto Rosenblatt (1958) trabalhava no perceptron, Bernard Widrow da Universidade de Stanford com a ajuda de alguns estudantes desenvolveram um novo modelo de processamento de redes neurais chamado de Adaline (*ADaptive LINear Element*), o qual se destacava pela sua poderosa lei de aprendizado. O princípio de treinamento para as redes Adalines, ficou conhecido como a *Regra Delta*, que foi mais tarde generalizada para redes com modelos neurais mais sofisticados. Mais tarde, Widrow criou a Madaline, que era uma generalização multidimensional do Adaline.

Nos anos seguintes, muitos artigos foram publicados e várias previsões exageradas e pouco confiáveis para a época foram anunciadas. A maioria destas suposições falava de computadores com um poder de raciocínio e/ou processamento igual ou superior ao do cérebro humano. Desta forma, a credibilidade de futuros estudos das RNAs foram fortemente comprometidos.

No início da década de 80, muitos pesquisadores publicaram inúmeras propostas para a exploração de desenvolvimento e pesquisa em redes neurais. Foi quando o administrador de programas da DARPA (*Defense Advanced Research Projects Agency*) Ira

Skurnick resolveu dar atenção às proposições da neurocomputação, contrariando todos os preceitos, fundando em 1983 as pesquisas em neurocomputação da DARPA.

Este fato acabou abrindo novos horizontes para a neurocomputação. O físico e biólogo de reputação mundial John Hopfield também se interessou pela neurocomputação e escreveu vários artigos em 1982 que levaram outros cientistas a se unirem nesta nova área emergente. Hopfield reascendeu as pesquisas em neurocomputação, criticando fortemente as teorias apresentadas por Minsky e Papert, na década de 50.

Este campo de pesquisa iniciou, realmente, em 1986, quando o professor de psicologia da Universidade de Stanford, David E. Rumelhart, e seu colega James L. McClelland, professor de psicologia da Universidade de CarnegieMellon, publicaram o livro *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (vol.1: Foundations, vol.2: Psychological and Biological Models). Nesse livro, eles apresentam um modelo matemático e computacional que propicia o treinamento supervisionado dos neurônios artificiais. Surgia, então, o algoritmo *back-propagation*, um algoritmo de otimização global sem restrições.

O método de aprendizado *back-propagation* pode ser aplicado a qualquer rede que usufrui de uma função de ativação diferencial e aprendizado supervisionado. Assim como a regra delta, sua otimização é baseada no gradiente descendente, que ajusta os pesos para reduzir o erro da rede.

Durante a fase de treinamento, os sinais/padrões de entrada são apresentados à RNA, cada padrão de treinamento é propagado adiante, camada após camada, até a produção do sinal/padrão de saída. A saída computada pela rede é então comparada com uma saída desejada. Esta comparação gera um valor que determina o erro. Este erro é utilizado como uma realimentação para as conexões, que resultam no ajuste dos pesos sinápticos de cada camada num sentido oposto à propagação dos sinais de treinamento. Os acoplamentos retrógrados somente existirão na fase de treinamento, considerando que as conexões adiante (sentido entrada saída) são usadas durante a fase de treinamento e uso da rede.

Fazendo o uso do *back-propagation*, as camadas ocultas terão os seus pesos ajustados de acordo com as camadas subseqüentes, i.e, com as camadas seguintes. Deste modo, os erros computados na camada de saída serão usados para ajustar os pesos entre a última camada escondida ou oculta com a camada de saída. Assim, o erro calculado de uma

camada oculta será usado para ajustar os pesos da camada oculta anterior. Este processo será repetido até que a primeira camada oculta seja ajustada. Desta forma, os erros serão retropropagados camada a camada com as devidas correções. Rotina esta que será realizada de uma maneira repetitiva, ajustando os pesos das respectivas camadas.

O processo é repetido para cada padrão de dados durante o treinamento até que o erro total da saída tenha convergido a um valor mínimo, ou até que algum limite predeterminado de iterações tenha sido completado.

Pode-se, então, criar duas fases para o algoritmo de treinamento do *back-propagation*. Cada fase percorre um sentido da rede. A primeira fase, chamada de propagação (*forward*), define a saída da rede para um determinado padrão de dados de entrada. A segunda e última retropropagação (*backward*) está incumbida de utilizar a saída desejada/esperada e a saída fornecida pela última camada da rede para ajustar os pesos sinápticos da rede neural. Na Figura 12, observa-se, esquematicamente, a direção destes dois fluxos básicos de sinais em uma rede MLP.

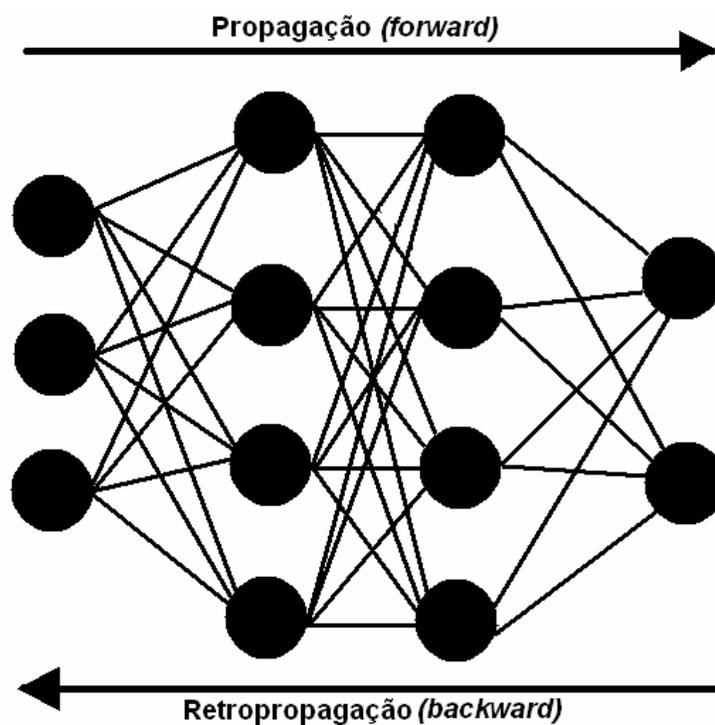


Figura 12 – Fluxo de treinamento de uma MLP c/*back-propagation*

Conforme descrito em Paula (2000), segundo Carvalho & Ludemir (1998), pode-se definir os passos do seguinte modo:

Propagação (**Forward**):

1. Os padrões de entrada são apresentados à primeira camada c_1 que compõe a rede;
2. Para cada camada c_i a partir da camada de entrada:
 - 2.1. Os sinais de saída do neurônio da camada c_i irão alimentar a entrada da camada c_{i+1} , i.e, serão os sinais de entrada para a próxima camada;
3. Os sinais provenientes da última camada serão comparados com os sinais desejados.

Retropropagação (**Backward**):

- 1 Da última camada da rede até a primeira [$n \rightarrow 1$];
 - 1.1 Os neurônios artificiais da camada c_a (camada atual) devem ajustar seus pesos sinápticos de forma a reduzir seus erros;
 - 1.2 O erro de um neurônio das camadas intermediárias, $c_{[2, n-1]}$, por exemplo, c_i , será calculado utilizando os erros retropropagados dos neurônios que pertencem às camadas seguintes conectados a ele, no caso, c_{i+1} , os quais serão ponderados pelos pesos das conexões entre eles.

Back-propagation:

1. Inicializar a rede (pesos sinápticos);
2. Repita
 - 2.1. Para cada padrão/dados de treinamento $P[x]$, para todo $x \in [1, n]$, sendo n o número total de amostras do conjunto de treinamento.
 - 2.1.1. Calcular a saída (S) da rede utilizando a fase *forward*;
 - 2.1.2. Comparar a saída (S), calculada no item 2.1.1, com as saídas desejadas;
 - 2.1.3. Realizar atualização dos pesos sinápticos fazendo o uso da fase *backward*;
3. Até o erro ser mínimo ou até x épocas.

Dentre as características que norteiam o funcionamento do algoritmo de aprendizado *back-propagation*, um detalhe importante é a repetição (ciclos) em um determinado número de vezes (x) que o par amostra-resultado (saída) é submetido para que a rede possa, efetivamente, aprender. Estas repetições são chamadas *épocas* de treinamento (ciclos).

Na atualização dos pesos sinápticos, o algoritmo *back-propagation* fornece também uma aproximação, cujo parâmetro é um índice variando entre 0 (zero) e 1 (um), chamado de *taxa de aprendizagem*, o qual tem por finalidade determinar quanto do valor atual da sinapse será alterado na próxima época. Quanto menor for este parâmetro, menor serão as variações dos pesos sinápticos da rede, de uma iteração para a outra, e mais suave será a trajetória no espaço de pesos. Esta melhoria, entretanto, é obtida à custa de uma evolução de aprendizagem lenta. Por outro lado, caso utilize-se um parâmetro muito alto, para acelerar a aprendizagem, as grandes modificações nos pesos resultantes podem tornar a rede instável (oscilatória) (Haykin, 2001).

Outra forma de atualização dos pesos é não utilizar uma taxa fixa, mas variável. Em alguns casos, pode-se orientar o algoritmo para que, quanto mais próximo estiver do valor ideal do peso, menor a taxa de atualização ou aprendizagem. Também pode ser utilizado, na atualização do peso, uma parcela proporcional à atualização anterior, denominado *momento* ou *inércia* (Medeiros, 2003).

O processo de atualização dos pesos é chamado de *convergência*, ou seja, uma rede converge seus pesos para determinados valores quando o erro global diminui a cada época de treinamento.

Como já visto, para diminuir os erros da rede, o algoritmo *back-propagation* é baseado no gradiente descendente (*gradient descent*), o qual se baseia no erro existente entre o resultado ideal ou desejado e o resultado obtido. A relação entre o erro existente e os pesos são feitos de forma que o algoritmo calcula a direção do menor erro no espaço de pesos. Porém, não os atualiza de uma vez, mas em pequenas diferenças (positivas ou negativas). Estas diferenças são passadas para trás (retropropagadas) na rede, atualizando para cada par amostra-resultado apresentado. Efetuando-se o processo para o conjunto de treinamento por várias épocas, a expectativa é que o erro seja diminuído gradativamente. Se o

erro é diminuído, a obtenção de dados próximos aos desejados é a consequência e a rede fica apta, então, a executar o processo para o qual foi projetada (Medeiros, 2003).

Neste mecanismo observa-se que o cálculo dos valores dos pesos sinápticos da rede não é um processo exato, ou seja, os valores são obtidos por meio de um processo de tentativa e erro, até que se obtenha um resultado adequado ao problema proposto.

Medeiros (2003) descreveu graficamente de forma simplificada a evolução do processo de convergência dos pesos para um valor de mínimo erro. Na Figura 13, o eixo vertical representa o erro global E (diferença média quadrática entre as saídas desejadas e saídas calculadas). No eixo horizontal, são representados os valores que um peso w pode assumir. Reproduz-se, então, a curva referente ao erro apresentado, conforme o valor do peso.

O processo da descida de gradiente direciona o valor do peso para onde está o mínimo erro e, assim, o algoritmo executa passo a passo a atualização do valor do peso até aonde o erro minimiza (E_{min}). Nesta ilustração, a convergência do valor do peso é representada como $W_{inicial}$, para chegar próximo a W_{final} , as setas ao lado da curva representam a direção do gradiente dando o sentido do mínimo erro global.

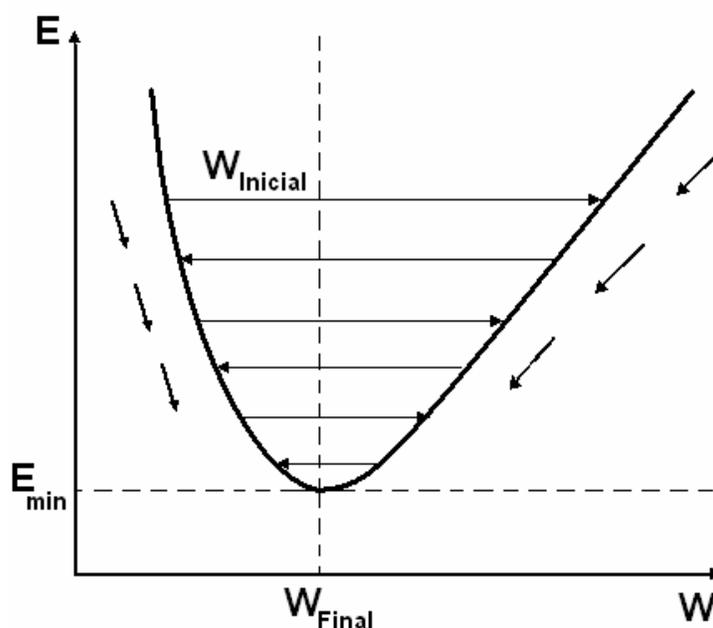


Figura 13 – Gráfico erro global x peso

O *Back-propagation* é um algoritmo de aprendizagem eficiente e utilizado por diversos autores, porém, sua implementação normalmente exige muitos passos e, assim, um tempo de treinamento longo. Há muitos estudos, baseados em heurísticas e em técnicas de otimização, para apressar a convergência do algoritmo, com o objetivo de melhorar sua eficiência em relação ao tempo de treinamento (Andrade et al., 2003).

Dentre estes estudos, destaca-se o de Riedmiller & Braun (1993) que propuseram o algoritmo *Resilient Propagation* (Rprop), o qual corresponde a um método que efetua uma adaptação direta na alteração dos pesos, baseada em informações de gradiente locais.

4.1.11 Algoritmo *resilient propagation* (Rprop)

Conforme Riedmiller & Braun (1993), a idéia principal é eliminar a influência do tamanho das derivadas parciais nos cálculos dos pesos e, deste modo, considerar somente a indicação do sinal da derivada parcial na atualização dos pesos. Dessa forma, se a derivada for pequena ele não dá passos pequenos, o que causaria uma lentidão no processo de treinamento.

Cada peso tem seu valor individual de atualização Δ_{ij} que determina apenas o tamanho da alteração no respectivo peso. Esse valor individual de atualização evolui durante o processo de aprendizado, com base na observação local da função de erro E , de acordo com a seguinte regra de aprendizado:

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^+ * \Delta_{ij}^{(t-1)} & , \text{ se } \frac{\partial E^{(t-1)}}{\partial w_{ij}} * \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^- * \Delta_{ij}^{(t-1)} & , \text{ se } \frac{\partial E^{(t-1)}}{\partial w_{ij}} * \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)} & , \text{ senão} \end{cases} \quad (9)$$

onde $0 < \eta^- < 1 < \eta^+$

Esta regra de aprendizado expressa a seguinte situação: cada vez que a derivada parcial do peso correspondente w_{ij} troca seu sinal, o qual indica que a última atualização foi muito grande e o algoritmo pulou sobre um mínimo local, o valor de atualização Δ_{ij} é decrementado pelo fator h . Se a derivada retém seu sinal, o valor de atualização é ligeiramente incrementado para acelerar a convergência nas regiões rasas.

Uma vez que o valor de atualização para cada peso é adaptado, a adaptação do peso segue uma regra simples: se a derivada é positiva (erro crescente), o erro é decrementado pelo seu valor de atualização, se a derivada é negativa, o valor de atualização é somado:

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)} & , \text{ se } \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)} & , \text{ se } \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0 & , \text{ senão} \end{cases} \quad (10)$$

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)} \quad (11)$$

Entretanto, há uma exceção à essa regra, que ocorre quando a derivada parcial troca o sinal, i.e. quando o passo prévio é muito largo e o algoritmo pula sobre um mínimo local, a atualização prévia do peso é revertida:

$$\Delta w_{ij}^{(t)} = -\Delta w_{ij}^{(t-1)}, \text{ se } \frac{\partial E^{(t-1)}}{\partial w_{ij}} * \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \quad (12)$$

Devido a essa reversão no peso do passo, a derivada deverá trocar novamente de sinal durante o próximo passo somente uma vez. Para evitar uma dupla modificação do valor de atualização, não deve haver nenhuma atualização do valor no passo

seguinte. Isto se faz atribuindo zero na derivada parcial do passo $t - 1$, na regra da Equação 12 em Δ_{ij} .

Os valores de atualização e os pesos são trocados toda vez que o conjunto padrão todo for apresentado uma vez para a rede (aprendizado por época).

Neste contexto, o tamanho da atualização dos pesos depende apenas da seqüência de troca de sinais das derivadas parciais, não dependendo da magnitude das mesmas. Dessa forma, os pesos distantes da camada de saída têm a mesma oportunidade de aprendizado daqueles próximos à camada de saída, resultando em um modo de distribuição de aprendizado homogêneo pela RNA, melhorando a eficiência no processo de aprendizado.

Diferentemente do algoritmo *back-propagation*, onde um dos aspectos da descida do gradiente é que o valor da derivada decresce exponencialmente com a distância entre o peso e a camada de saída da RNA. Deste modo, pesos muito distantes da camada de saída são menos modificados e, conseqüentemente, o aprendizado torna-se mais lento.

Nas próximas seções deste capítulo, serão abordados temas que norteiam o foco agrônômico desta tese.

4.2 Molhamento foliar

A umidade constitui-se num dos fatores mais importantes que influenciam no desenvolvimento das doenças das plantas. A umidade, medida em termos de quantidade e duração, é essencial para que haja infecção pela maioria dos fungos e bactérias, agentes causais de doenças (Silva et al., 2001a).

Vários autores evidenciam a importância da Duração do Período de Molhamento Foliar (DPMF), como um dos fatores relacionados a germinação de esporos e penetração nos hospedeiros, desencadeando processos infecciosos nas culturas em geral e, conseqüentemente, a evolução de epidemias em escala regional (Buzzerio, 2001; Leite, 2002; Monteiro, 2002; Noronha, 2002; Pinto et al., 2002; Pivonia & Yang, 2004; Rajvanshi, 1991; Sentelhas, 1992; Sentelhas et al., 1993; Sentelhas, 2004; Silva et al., 2001a; Silveira et al., 2001; Tsukahara, 2004).

Uma vez que o orvalho é dificilmente medido em estações meteorológicas e a DPMF das folhas é de grande importância para o estudo da relação

patógeno³ – hospedeiro, torna-se importante a sua estimativa para cada localidade, a fim de se compreender a dinâmica e evolução de epidemias regionalmente, objetivando o uso racional de defensivos agrícolas, com redução dos custos de produção e, também, dos impactos ambientais (Tsukahara, 2004).

4.2.1 Orvalho

Orvalho é o nome que se dá às gotas de água que se formam na superfície de objetos que permanecem ao relento, durante a noite, resultantes da condensação do vapor de água contido no ar. Forma-se nas noites claras, porque nelas as superfícies descobertas irradiam calor para a atmosfera. A maior parte dos objetos, inclusive as folhas de plantas e pétalas de flores, irradiam calor melhor do que o ar. Durante a noite, quando essa perda de temperatura não é compensada por uma produção eficiente de calor no interior do objeto, sua superfície fica mais fria que o ar. As superfícies resfriadas esfriam o ar que está a sua volta e este, se for suficientemente úmido, baixa a temperatura até o ponto de orvalho. O vapor de água contido no ar se condensa e adere às superfícies resfriadas, transformado em orvalho (Benton, 1970).

Ponto de orvalho é a designação dada à temperatura de saturação do ar úmido, abaixo da qual o vapor contido ali se condensa e precipita-se em forma de gotículas.

Podem ocorrer dois tipos de orvalho: o primeiro se forma quando o vapor de água se difunde no ar em direção ao solo, este processo é conhecido como precipitação de orvalho; o outro, pelo vapor de água que se difunde da superfície do solo, cujo processo é conhecido como destilação de orvalho (Sentelhas, 1992).

4.2.2 Molhamento foliar por orvalho

A formação de orvalho depende da temperatura e da umidade relativa do ar, de forma que, quanto maior a temperatura, maior a quantidade de vapor de água que o ar

³ É o microorganismo causal da doença que vive numa relação parasitária com o hospedeiro ou planta cultivada. As doenças causadas por microorganismos envolvem fungos, bactérias, vírus, protozoários e nematóides (Melo, 2004).

pode reter (Roman, 1999), sendo que, os fluxos de vapor da superfície são influenciados pela radiação solar, temperatura, umidade e velocidade do vento (Burrage apud Sentelhas, 1992).

A temperatura como um fator de ponto de orvalho, é um parâmetro geofísico importante que indica o estado do conteúdo de umidade no ar sob determinadas condições, e é vital para calcular vários parâmetros agrometeorológicos, inclusive a evapotranspiração (Hubbard et al., 2003).

Um aspecto importante na duração do molhamento foliar é o fato de que a quantidade e a duração do orvalho sobre as folhas de uma planta não dependem somente das condições meteorológicas locais, mas também da estrutura da planta, estágio fenológico⁴, posição e geometria da folha, além de suas propriedades térmicas, dificultando assim o registro deste parâmetro em estações meteorológicas convencionais (Monteiro, 2002; Sutton et al., 1984).

Com relação a estrutura da planta, conforme um estudo realizado por Burrage (apud Sentelhas, 1992), sobre o orvalho na cultura do trigo, o mesmo verificou uma maior quantidade de orvalho formada nas partes mais altas da cultura, contudo a DPMF nas diferentes alturas da planta foi semelhante.

Para se medir a DPMF, de acordo com Miranda et al. (2000), os mecanismos mais utilizados são os sensores mecânicos, representados pelo aspergígrafo e orvalhógrafo, além dos sensores eletrônicos, que tentam simular uma folha e/ou fruto específico.

Aspergígrafo é um registrador de DPMF que se caracteriza por possuir um elemento sensível formado por fios de cânhamo que se contraem pelo molhamento e se distendem pelo secamento. As variações de comprimento desses fios são registradas por uma pena num diagrama localizado sobre um mecanismo de relojoaria.

O orvalhógrafo registra a DPMF por medidas da variação do peso do orvalho acumulado sobre um pequeno prato metálico exposto, que funciona como elemento sensível. O prato aciona, através de um sistema de alavancas, uma pena que registra o movimento num diagrama situado no sistema de relojoaria.

⁴ Referente a fenologia que corresponde a parte da botânica que estuda vários fenômenos periódicos das plantas, como a brotação, a floração, e a frutificação, marcando-lhes as épocas e os caracteres.

Com o advento da microeletrônica e o desenvolvimento e a popularização dos sistemas automáticos de aquisição de dados, atualmente os sensores eletrônicos são os mais empregados (Sentelhas, 2004).

Monteiro (2002) citou uma crescente demanda por sensores para determinação do molhamento, destacando-se os que detectam a mudança de resistência ou impedância em função do molhamento do sensor, porém o mesmo afirma que estes sensores de aquisição automática de dados demonstram grandes divergências nas determinações da duração do período de molhamento, mesmo em curtos intervalos de tempo.

Para minimizar estas divergências, Sentelhas (2004) destacou que a pintura dos sensores com duas ou três camadas de tinta látex branca, aumenta a habilidade do sensor em detectar pequenas quantidades de molhamento, ou seja, gotículas com menos de 1 mm de diâmetro, e também submeter os sensores a um tratamento térmico em estufa, com temperatura entre 60 e 70°C por 12 horas, para remoção ou desativação de componentes higroscópicos⁵ da tinta. São procedimentos importantes para a estimativa da duração do período de molhamento através de sensores eletrônicos.

4.2.3 Estimativa da DPMF

As dificuldades de medição do período de molhamento através de aparelhos registradores levaram ao emprego crescente de dados de estações meteorológicas padrões em modelos de estimativa da duração do período de molhamento, através de regressões lineares simples e múltiplas, para uso em programas de manejo de doenças em plantas (Tsukahara, 2004).

Conforme descrito em Tsukahara (2004), Crowe et al. (1978) utilizaram dados de uma estação meteorológica padrão para a estimativa da DPMF na cultura do trigo. Desenvolveram equações de regressão múltipla, envolvendo variáveis como umidade relativa do ar, velocidade do vento e temperatura mínima, obtendo acurácia de 71%, a qual foi elevada para 85% com a inclusão da umidade do solo. Esses resultados foram contestados por Getz (1979), uma vez que o padrão de medida da DPMF utilizado pelos autores foi o número

⁵ Material ou substância que tem grande afinidade pelo vapor de água, sendo capaz de retirá-lo de uma atmosfera ou eliminá-lo de uma mistura gasosa.

de horas de umidade relativa maior ou igual a 90% obtido em registrador tipo Taylor, ambos não aconselháveis como padrão devido aos erros que apresentam.

Tsukahara (2004) concordou com Sentelhas (1992) e Amador (1987), em que a estimativa da DPMF através da simples soma do número de horas com umidade relativa do ar maior ou igual a 90%, obtida em termohigrógrafo⁶ localizado no posto meteorológico, 1,5 metros acima do plano horizontal, não apresenta resultados confiáveis.

Amador (1987), através de regressão linear simples, desenvolveu uma equação para estimativa do período de molhamento para a cultura do feijoeiro tendo como base parâmetros observados no posto meteorológico. Assim, o molhamento foi estimado em função do número de horas com umidade relativa do ar maior ou igual a 90%, alcançando um valor de $R^2 = 0,55$ a 1% de probabilidade. Acréscimos dos registros de velocidade do vento e temperatura mínima não melhoraram o desempenho das equações.

Sentelhas (1992) avaliou o molhamento na cultura do trigo através de medidas obtidas em posto meteorológico padrão e obteve valores de $R^2=0,34$, a 5% de probabilidade, quando realizou regressões lineares simples entre a Duração do Período de Molhamento Visual (DPMV) e o número de horas com umidade relativa do ar maior ou igual a 90%. Quando foram utilizados os registros obtidos no dossel da cultura (microclima), o coeficiente de determinação (R^2) aumentou para 0,89.

Sentelhas (2004), em sua tese de livre docência, fez uma profunda análise a respeito da DPMF, especialmente com relação aos aspectos operacionais de sua medida, de sua variabilidade espacial em diferentes culturas e de sua estimativa por diversos métodos, de modo a se determinar uma condição de referência para sua medida e a relação dessa DPMF de referência com a DPMF em diferentes culturas. Dentre suas conclusões, pode-se destacar que, a medida da DPM em diversas posições de diferentes culturas mostrou que essa variável é afetada não somente pelas condições climáticas, mas também pela estrutura, arquitetura, arranjo e altura das plantas, fatores esses que controlam o microclima.

⁶ Aparelho que registra simultaneamente a temperatura e a umidade relativa do ar, cujo sensor é um fio de cânhamo ligado a uma pena que faz os registros em papel colocado e um sistema de relógio.

4.2.4 Principais fatores no desenvolvimento de doenças em plantas

Zahler et al. (1991) apontaram as variáveis meteorológicas, como sendo os fatores que exercem maior influência na ocorrência de doenças em plantas, onde são citadas como principais, a DPMF e a temperatura do ar durante o período de molhamento, seguidos da umidade relativa do ar, precipitação pluvial e vento.

Para o entendimento e quantificação da ocorrência de doenças em plantas, a observação contínua dessas variáveis, assim como da cultura e do patógeno, são necessárias (Sutton et al., 1984).

A temperatura e a umidade na superfície da planta são os fatores ambientais que afetam mais intensamente o início e o progresso de doenças infecciosas em plantas. Os patógenos diferem em suas preferências por alta ou baixa temperatura, uma vez que a mesma afeta a germinação de esporos e o número de esporos formados (Silveira et al., 2001). Esses fatores do ambiente influenciam o hospedeiro e o patógeno e, às vezes, os vetores dos patógenos (Sentelhas, 1992).

De modo geral, o efeito da temperatura sobre a atividade do patógeno é menos marcante que aquele exercido pela umidade. A maioria dos patógenos, particularmente aqueles presentes em regiões tropicais e subtropicais, é capaz de crescer em uma ampla faixa de temperatura, o que significa que a relação temperatura-molhamento varia conforme o patógeno (Monteiro, 2002).

Como exemplo, Canteri et al. (1999), em estudos da ocorrência de mancha angular (*Phaeoisariopsis griseola*) nos cultivares de feijão Rosinha e Carioca, observaram a máxima severidade de infecção na faixa de temperatura entre 21 e 25°C, e o menor período de incubação a 25°C, não tendo sido observado o desenvolvimento de sintomas em temperaturas inferiores a 6°C e superiores a 29°C.

De acordo com pesquisas realizadas por Burrage (apud Sentelhas, 1992), em estudos de fitopatossistemas, a DPMF é mais importante em um processo de infecção, do que a quantidade de água depositada sobre as folhas.

Um outro fator de grande importância em processos epidêmicos de plantas são as chuvas, porque as mesmas dispersam o inóculo, reduzem a luminosidade e

provocam quedas de temperatura, aumentando a probabilidade de formação de orvalho por dias seguidos (Eversmeyer & Burleigh apud Sentelhas, 1992).

Entre os diversos parâmetros climáticos possíveis de serem medidos ou estimados, pode-se afirmar que o orvalho (ou a DPMF) é o que mais se correlaciona com o início dos processos infecciosos na cultura do trigo, visto que a maioria dos esporos de fungos foliares patogênicos necessita de um filme de água sobre as folhas, por um determinado tempo, para que possam germinar e penetrar no hospedeiro, ocasionando o processo infeccioso (Tsukahara, 2004).

4.2.5 Doenças do trigo

Muitas são as doenças que ocorrem na cultura do trigo em todo o mundo, desde doenças radiculares até as que atacam as glumas. Porém, dentre todas, as mais afetadas pelas condições microclimáticas da cultura são as doenças foliares como: ferrugens, septorioses, helmintosporiose e oídio que necessitam de condições ótimas para sua ocorrência e desenvolvimento, principalmente, de temperatura e umidade (Sentelhas, 1992).

Segundo Barros (1988), as doenças mais comuns que ocorrem na cultura do trigo são a ferrugem do colmo (*Puccinia graminis*) e a da folha (*Puccinia recôndita*), consideradas importantes pela frequência e intensidade de ocorrência na região sul do país. Outras doenças como a helmintosporiose (*Cochiobolus sativus*), septoriose (*Septoria nodorum*), oídio (*Erysiphe graminis*), giberela (*Gibberela zae*) e a brusone (*Pyricularia sp.*) chegam a causar maiores danos que as ferrugens quando as condições meteorológicas são altamente favoráveis.

4.2.6 Cultura do trigo no Brasil

A cultura do trigo sempre mereceu destaque mundial, pois se trata de um dos cereais mais importantes no fornecimento de carboidratos essenciais à dieta humana. Seu cultivo abrange praticamente todas as regiões do mundo, sendo as maiores produtividades observadas entre as latitudes 20° e 55° nos hemisférios Norte e Sul (Tsukahara, 2004).

No Brasil, há relatos que o cultivo do trigo tenha se iniciado em 1534, na antiga Capitania de São Vicente. A partir de 1940, a cultura começou a se expandir

comercialmente no Rio Grande do Sul. Nessa época, colonos do Sul do Paraná plantavam sementes de trigo trazidas da Europa em solos relativamente pobres, onde as cultivares de porte alto, tolerantes ao alumínio tóxico, apresentavam melhor adaptação (Embrapa, 2004).

De acordo com a Embrapa Trigo (Embrapa, 2006), no Brasil, as principais áreas produtoras deste cereal se concentram nas regiões Sul e Sudeste, devido, principalmente, às condições climáticas e tecnológicas necessárias para sua produção econômica, sendo representada pelos estados do Rio Grande do Sul, Paraná e São Paulo. Estes estados são, hoje, responsáveis por cerca de 91% da produção nacional, representando, aproximadamente, 2,6 milhões de toneladas.

Segundo a Embrapa (2004), a adaptação do trigo para as condições de clima e solo do Paraná tem sido realizada, pela soma de fatores genéticos e culturais, envolvendo inúmeras pesquisas, dentre elas a busca pela resistência às ferrugens. Ao longo das duas últimas décadas, a pesquisa tem aprimorado outras tecnologias, tais como, rotação de culturas, manejo adequado do solo, controle integrado de pragas, controle químico de doenças e zoneamento agroclimático.

Ironicamente, o aumento do potencial produtivo e a diminuição dos riscos de perdas, proporcionados pelas cultivares melhor adaptadas e pelo controle de doenças e pragas, via manejo da cultura e utilização de fungicidas eficientes, não têm sido suficientes para sustentar o aumento da produção (Embrapa, 2004).

De acordo com um levantamento realizado pela Embrapa Trigo (Embrapa, 2004), durante a safra de 2003 da região norte do Rio Grande do Sul, os itens de maior oneração no custo total de produção foram os fertilizantes de base e cobertura (24,75%), sementes (15,46%) e defensivos agrícolas (10%). Em relação ao ano anterior (2002), observou-se um aumento de, aproximadamente, 29% nos custos totais, com destaque para os reajustes de preços de semente (81%) e de fertilizantes (46%) (De Mori & Filho, 2003).

Considerando que o custo total de produção no Paraná não se diferencia expressivamente dos custos estimados pela Embrapa Trigo, e que o manejo conservacionista priorizando a manutenção da palha sobre o solo intensifica a ocorrência de processos epidêmicos, o tratamento fitossanitário em sistemas de plantio direto se torna indispensável para a obtenção de produtividades econômicas (Tsukahara, 2004).

5 MATERIAL E MÉTODOS

Os dados utilizados para atingir os objetivos dessa tese foram variáveis climáticas obtidas de estações meteorológicas, instaladas na cultura do trigo.

A escolha da cultura do trigo teve como motivo principal, o fato de tratar-se de um dos principais cereais de inverno para a região dos Campos Gerais. Além da produção de grãos, contribui com a rotação de culturas e com a manutenção do sistema de plantio direto.

Como metodologia computacional para elaboração do trabalho, foram utilizadas RNAs tipo Perceptron de Múltiplas Camadas (*Multilayer Perceptron*), com os algoritmos de treinamento *Back-Propagation* e *Resilient Propagation*.

Para determinar as melhores arquiteturas e os respectivos algoritmos de treinamento de RNAs, bem como, definir quais as variáveis climáticas que influenciam significativamente a ocorrência do molhamento foliar, foram utilizados os simuladores SNNS (Stuttgart Neural Network Simulator) versão 4.2 (Zell et al., 1998) que utiliza plataforma operacional Linux e o simulador JavaNNS – Java Neural Network Simulator 1.1 (Fischer et al., 2001) para ambiente de execução Windows, o qual é baseado no SNNS.

O sistema computacional resultante foi codificado na linguagem de programação Borland Delphi Enterprise versão 7.0 (Borland, 2003; Cantu, 2003), sendo executado na plataforma Windows. Para a implementação das RNAs foi utilizado o componente TMLP (*Training Multilayer Perceptron*) obtido em Medeiros (2003), cujo

componente, foi modificado visando as funcionalidades exigidas pelos estudos de caso, assim como, o desenvolvimento do algoritmo de treinamento *Resilient Propagation*.

5.1 Bases de dados

Os dados utilizados neste trabalho têm como origem três locais diferentes, a saber:

1. Base de Dados 1 – Campo Demonstrativo e Experimental da Fundação ABC - Fazenda Capão do Cipó, em Castro – PR (Tsukahara, 2004).
2. Base de Dados 2 – Campo Demonstrativo e Experimental da Fundação ABC - Fazenda Palmeirinha em Piraí do Sul – PR (Tsukahara, 2005).
3. Base de Dados 3 – Posto Agrometeorológico ESALQ/USP em Piracicaba - SP (ESALQ, 2005).

A principal diferença na composição das bases de dados foi a coleta visual de informações relativas ao molhamento foliar para a base de dados 1. Esta mesma variável, nas bases de dados 2 e 3, foram obtidas por meio de sensores eletrônicos de molhamento.

A seguir são descritos detalhes sobre cada uma das bases de dados utilizadas.

5.1.1 Especificação da base de dados 1

São dados meteorológicos resultantes de um experimento conduzido no Campo Demonstrativo e Experimental de Castro, Fazenda Capão do Cipó, em Castro - PR, durante a safra de inverno de 2003, com a cultura de trigo (Tsukahara, 2004). Tais dados correspondem aos valores da duração do período de molhamento e dados meteorológicos monitorados em uma Estação Meteorológica Automática (EMA), Squitter™, instalada na área experimental, com dados cotados em intervalos de 5 minutos.

Os valores da duração do período de molhamento foliar foram obtidos visualmente. Para tanto foi utilizada uma área cultivada com trigo de 5.000 m², sendo que,

dentro desta demarcou-se uma sub-área de 20 x 50 metros, localizada nas coordenadas geográficas 24° 51' 40" sul e 49° 56' 01" oeste e a 1.044 metros de altitude.

A avaliação da DPMV foi feita após as plantas de trigo terem atingido o estágio fenológico 3 da escala de crescimento desenvolvida por Feekes (2004), foram selecionadas 24 plantas, distribuídas homoganeamente dentro da parcela. Para a avaliação visual, utilizou-se a escala proposta por Heldwein & Krzysch (1997), ajustada aos aspectos relacionados à morfologia da cultura do trigo. Para proceder à observação, utilizou-se lanterna e lupa de 20x de aumento, permitindo uma melhor visualização da formação das primeiras gotículas do orvalho.

Para obtenção da DPMV, foram efetuadas observações visuais do início e fim de duração do período de molhamento (DPM), durante o período de 10 de agosto a 15 de outubro de 2003, num total de 14 observações. Estas observações foram realizadas a cada 30 minutos, utilizando a escala visual adaptada de Heldwein & Krzysch (1997), repetida nas mesmas 24 plantas demarcadas no início do experimento.

O critério de avaliação foi semelhante ao usado por Amador (1987) e Sentelhas (1992), baseando-se na observação da presença (início) ou da secagem (fim) do orvalho em cerca de 60% das folhas situadas no terço superior da cultura.

Na Tabela 3, está descrita de forma resumida a composição desta base de dados, a qual, originalmente, contém 3.888 registros, organizados em 10 variáveis meteorológicas, e a estas foram acrescentadas mais três colunas correspondentes ao índice de classificação do molhamento foliar (IMF), a data e horário. Os demais parâmetros são: PRES (pressão atmosférica do ar), URMED (umidade média relativa do ar), URMIN (umidade relativa mínima), URMAX (umidade relativa máxima), TEMPMED (temperatura média do ar), TEMPMIN (temperatura mínima do ar), TEMPMAX (temperatura máxima do ar), VENTMED (velocidade média do vento), VENTMIN (velocidade mínima do vento) e VENTMAX (velocidade máxima do vento).

Tabela 3 – Composição parcial da base de dados 1 (Tsukahara, 2004)

HORÁRIO (hh:mm:ss)	DATA (dd/mm/aaaa)	IMF	PRES (mb)	URMED (%)	URMIN (%)	URMAX (%)	TEMPMED (°C)	TEMPMIN (°C)	TEMPMAX (°C)	VENTMED (m/s)	VENTMIN (m/s)	VENTMAX (m/s)
18:05:00	11/8/2003	1,00	903,66	76,78	76,36	77,24	9,99	9,44	10,63	0,10	0,00	1,47
18:10:00	11/8/2003	1,00	903,72	74,92	74,92	75,65	10,17	9,68	10,63	0,78	0,00	1,96
18:15:00	11/8/2003	1,00	903,72	75,87	75,62	76,29	9,99	9,50	10,42	0,98	0,00	2,06
18:20:00	11/8/2003	1,00	903,72	77,97	77,76	78,22	9,56	9,07	9,99	1,67	0,59	2,06
18:25:00	11/8/2003	1,00	903,72	78,69	78,28	79,44	9,32	9,01	9,71	1,76	1,18	1,96
18:30:00	11/8/2003	1,00	903,74	80,75	80,35	81,12	8,80	8,46	9,19	1,67	0,98	2,06
18:35:00	11/8/2003	1,20	903,74	81,94	81,42	82,43	8,52	8,03	8,95	1,86	1,18	2,16
18:40:00	11/8/2003	1,20	903,75	82,95	82,55	83,28	8,28	7,88	8,71	0,10	0,00	1,08
18:45:00	11/8/2003	1,20	903,75	82,43	81,76	83,01	8,43	7,97	8,83	1,47	1,08	1,96
18:50:00	11/8/2003	1,20	903,77	82,15	81,60	82,73	8,65	8,28	9,10	1,57	0,10	1,96

Como houve diferença do intervalo de leitura da EMA (5 minutos) em relação a coleta visual de dados (30 minutos), a adequação da coluna IMF, Tabela 3, foi realizada relacionando os horários, e considerando que dentro de cada parcela de horário de 30 minutos o IMF é o mesmo.

5.1.2 Especificação da base de dados 2

Os dados desta base foram oriundos de um experimento conduzido na Fazenda Palmeirinha em Piraí do Sul - PR, durante a safra de inverno de 2005, cultura de trigo (Tsukahara, 2005).

A Tabela 4, apresenta parcialmente o conjunto de dados, com 21.610 registros, organizados com os seguintes campos: HORÁRIO, DATA, IMF, TEMPMED e CHUVA (índice pluviométrico – milímetros).

A coleta destes dados meteorológicos foi realizada nas coordenadas geográficas 24° 22' sul e 50° 61' oeste a 980 metros de altitude, Figura 14, com dados coletados em intervalos de 5 minutos entre 22 de agosto de 2005 a 26 de outubro de 2005. Neste local foram instalados um sensor de molhamento plano marca Davis® (Agrosystem, 2005), um termohigrômetro⁷ Squitter™ e um pluviômetro Texas Campbell. O sensor de

⁷ Aparelho que registra simultaneamente a temperatura e a umidade relativa do ar, cujo sensor é um termopar, ligado a um sistema registrador de sinais, que posteriormente os converte em valores.

molhamento teve sua altura ajustada conforme o desenvolvimento da cultura, com inclinação de 45° em relação ao solo e voltado para a face norte.

Tabela 4 – Composição parcial da base de dados 2 (Tsukahara, 2005)

HORÁRIO (hh:mm)	DATA (dd/mm/aaaa)	IMF	URMED (%)	TEMPMED (°C)	CHUVA (mm)
16:30	29/07/2005	152,0	65,8	22,8	0,0
18:30	29/07/2005	151,0	77,5	17,5	0,0
04:00	30/07/2005	35,0	82,9	14,2	0,0
08:45	30/07/2005	32,0	72,8	18,8	0,0
17:20	09/08/2005	60,0	80,3	12,2	3,8
23:35	09/08/2005	55,0	87,9	7,0	3,8
00:00	05/09/2005	30,0	93,8	13,9	0,0
23:15	05/09/2005	148,0	91,3	10,7	0,0
05:25	25/10/2005	32,0	90,7	17,3	0,0
12:00	25/10/2005	142,0	79,3	22,2	0,3



Figura 14 – Visão da EMA utilizada para coleta de dados da base 2. Pirai do Sul, PR. 2005.

5.1.3 Especificação da base de dados 3

Os dados desta base foram fornecidos pelo Professor Doutor Paulo César Sentelhas, pesquisador na área de Área de Física e Meteorologia na Escola Superior Agricultura Luis Queiroz da Universidade do Estado de São Paulo – ESALQ/ USP. São dados do Posto Agrometeorológico da ESALQ/USP – Estação Automática - Latitude: 22° 43' sul - Longitude de 47° 25' oeste - Altitude de 580 metros - Piracicaba - SP – Brasil (ESALQ, 2005).

Os dados que compuseram esta base foram: VENTMED, URMED, TEMPMED, data, horário e o IMF, conforme Tabela 5. O intervalo de coleta dos dados foi de 15 minutos no período compreendido entre 31 de julho de 2005 a 30 de setembro de 2005.

O IMF foi obtido por meio de um sensor de molhamento Campbell modelo 237 (Campbell, 2005), instalado sobre o gramado a altura de 30 cm com ângulo de exposição de 30° em relação à horizontal e voltado para o norte, detalhes deste sensor podem ser observados na Figura 15.

Tabela 5 – Composição parcial da base de dados 3 (ESALQ, 2005)

HORÁRIO (hh:mm)	DATA (dd/mm/aaaa)	IMF	URMED (%)	TEMPMED (°C)	VENTO (m/s)
22:45	31/07/2005	0,889	87,7	14,0	1,0
08:30	01/08/2005	0,861	94,3	13,8	1,3
22:15	01/08/2005	0,022	75,7	14,7	0,7
03:15	02/08/2005	1	99,0	12,2	0,9
08:45	03/08/2005	0,606	92,8	13,2	0,8
12:00	05/08/2005	0	41,3	18,1	1,3
09:45	03/09/2005	0,661	91,9	18,2	2,5
01:45	10/09/2005	1	98,5	17,3	0,3
09:00	15/09/2005	0,983	89	20,2	1,4
08:30	30/09/2005	0,144	87,8	17,5	1,5



Fonte: Sentelhas (2004)

Figura 15 – Detalhes de um sensor de molhamento Campbell modelo 237

Os dados descritos anteriormente foram submetidos à metodologia computacional, descrita na próxima seção.

5.2 Metodologia computacional

Neste trabalho foram utilizadas Redes Neurais Artificiais tipo Perceptron de Múltiplas Camadas (*Multilayer Perceptron*) com os algoritmos de treinamento *Back-Propagation* e *Resilient Propagation*, os quais seguem o paradigma de aprendizado supervisionado.

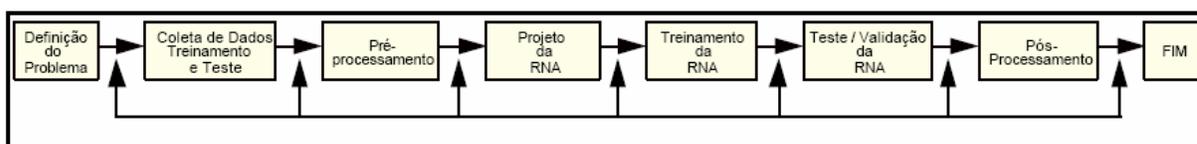
Para determinação dos algoritmos de treinamento a serem implementados, foram realizados experimentos utilizando os simuladores: SNNS (Stuttgart Neural Network Simulator) versão 4.2 (Zell et al., 1998) que utiliza plataforma operacional Linux, e o simulador JavaNNS – Java Neural Network Simulator 1.1 (Fischer et al., 2001) com plataforma de execução Windows, o qual é baseado no SNNS, tendo sido testados os seguintes algoritmos de treinamento: *Backpropagation Standard*, *Backpropagation for batch training*, *Backpropagation with momentum term*, *Backpropagation with chunkwise update*, *Backpropagation with Weight Decay*, *Quickprop* e *Resilient Propagation*.

5.2.1 Modelagem de redes neurais artificiais

Para obter os resultados esperados de um sistema implementado com RNAs e o paradigma de aprendizado supervisionado, torna-se necessária a análise preliminar dos dados com o objetivo de identificar fatores fundamentais para dar início ao processo. Dentre estes fatores, foram considerados a disponibilidade e o volume dos dados a serem tratados e se as suas características permitiam uma modelagem no sentido da obtenção de conhecimento. Outro fator fundamental, também considerado, foi a qualidade do processo de amostragem, sendo esta, uma condição essencial para o sucesso de todo o processo.

Em um segundo momento, duas fases distintas ocorreram no desenvolvimento das redes neurais: o treinamento através dos algoritmos de treinamento escolhidos, *Back-Propagation* e *Resilient Propagation*, e a validação (ou teste). Na validação, foram utilizados valores de entrada não utilizados no treinamento e, a partir do processamento destes, as redes retornaram valores conforme o seu conhecimento implícito.

O desenvolvimento do modelo seguiu proposta de Caudill (apud Medeiros, 1999), a saber: definição do problema, coleta dos dados de treinamento e de teste, pré e pós-processamento dos dados, projeto da estrutura da rede, treinamento, teste e validação. A representação gráfica do fluxo destas etapas pode ser visualizada na Figura 16.



Fonte: Medeiros (1999)

Figura 16 – Etapas do desenvolvimento da RNA

6 RESULTADOS

6.1 Desenvolvimento do modelo em RNA

As etapas para o desenvolvimento do modelo em Rede Neural Artificial serão descritas a seguir:

6.1.1 Definição do problema e coleta de dados

Diante dos dados disponíveis, buscou-se um modo de classificar o molhamento foliar em índices, sendo, então, necessária uma discretização inicial dos dados a serem trabalhados.

A discretização dos dados tem como objetivo formatar e ajustar os dados de forma a adequá-los a uma determinada metodologia de classificação, neste caso, especificamente, RNAs.

Para o caso da Base de Dados 1 (Tsukahara, 2004), tomou-se por base a variável DPMV, a qual foi obtida utilizando-se a escala de avaliação visual de orvalho desenvolvida por Heldwein & Krzysch (1997), cuja escala foi ajustada em doze níveis, dos quais, seis são para a identificação do início de formação do orvalho e seis para o final, conforme pode ser observado na Tabela 6.

Tabela 6 – Escala de avaliação visual de orvalho Heldwein & Krzysch (1997)

Escala visual para avaliar o <u>INÍCIO</u> do molhamento por orvalho	
1.0	folha seca, sem indício algum de formação de orvalho
1.2	nuança da cor da folha na sua face adaxial, visualizada facilmente quando a folha é observada no sentido longitudinal
1.5	início da formação do orvalho, aspecto aveludado fino e sem brilho, gotículas em menos de 50% da superfície da folha
2.0	confirmação da formação do orvalho, aspecto aveludado, porém brilhante, gotículas em até próximo de 50% da superfície da folha
2.5	gotículas proeminentes de fácil visualização em mais de 50% da folha, início de formação de gotas de orvalho na ponta das folhas
3.0	molhamento em mais de 50% da superfície da folha, apresentando formação de filme contínuo de água com diâmetro de gotas > 1mm
Escala visual para avaliar o <u>FINAL</u> do molhamento por orvalho	
3.0	molhamento em mais de 50% da superfície da folha, apresentando filme contínuo de água com diâmetro de gotas > 1mm
2.5	gotas de orvalho em torno de 50% da superfície da folha, com diâmetro < 1mm
2.0	gotas de orvalho sobre 50 a 20% da superfície da folha
1.8	gotas isoladas de orvalho sobre 20 a 10% da superfície da folha
1.5	Final do molhamento, menos de 10% da folha coberta com algumas gotas isoladas
1.0	ausência completa de molhamento

Para adequar a escala exibida na Tabela 6, de forma tal que não houvesse redundância nos valores dos índices, a escala foi modificada visando adaptá-la aos aspectos relativos à classificação de padrões e sua aplicabilidade as RNAs.

Neste sentido, valores da escala que são repetidos e possuem o mesmo significado, foram unificados, como no caso dos índices 1 (ausência completa de molhamento) e 3 (molhamento em mais de 50% da superfície da folha). Pode-se observar na Tabela 6, que estes dois índices aparecem duas vezes.

Para identificar o final do molhamento e também evitar as repetições dos índices 1.5, 2.0 e 2.5, os valores correspondentes foram representados como negativos, resultando em 10 padrões distintos conforme ilustrado na Tabela 7.

Tabela 7 – Escala de avaliação visual discretizada – base de dados 1

Escala visual para avaliar o <u>INÍCIO</u> do molhamento por orvalho	
1.0	folha seca, sem indício algum de formação de orvalho
1.2	nuança da cor da folha na sua face adaxial, visualizada facilmente quando a folha é observada no sentido longitudinal
1.5	início da formação do orvalho, aspecto aveludado fino e sem brilho, gotículas em menos de 50% da superfície da folha
2.0	confirmação da formação do orvalho, aspecto aveludado, porém brilhante, gotículas em até próximo de 50% da superfície da folha
2.5	gotículas proeminentes de fácil visualização em mais de 50% da folha, início de formação de gotas de orvalho na ponta das folhas
3.0	molhamento em mais de 50% da superfície da folha, apresentando formação de filme contínuo de água com diâmetro de gotas > 1mm
Escala visual para avaliar o <u>FINAL</u> do molhamento por orvalho	
-2.5	gotas de orvalho em torno de 50% da superfície da folha, com diâmetro < 1mm
-2.0	gotas de orvalho sobre 50 a 20% da superfície da folha
-1.8	gotas isoladas de orvalho sobre 20 a 10% da superfície da folha
-1.5	final do molhamento, menos de 10% da folha coberta com algumas gotas isoladas

Os índices definidos na tabela acima tiveram por objetivo definir o número de neurônios da camada de saída, ou seja, 10. Portanto, as RNAs geradas a partir da Base de Dados 1, especificamente, terão como objetivo classificar o molhamento foliar em 10 padrões distintos. Esta definição corresponde a um dos itens das arquiteturas das RNAs.

Com relação à definição do molhamento foliar das Bases de Dados 2 e 3, cuja coleta de dados foi obtida através de sensores de molhamento, o IMF foi determinado com base na proporção de tempo em que os sensores permaneceram secos ou molhados.

Para estimar a condição do sensor de seco ou molhado, são considerados os valores de resistência ou impedância elétrica observados em laboratório, e a partir destes valores é estabelecido um limiar acima do qual o sensor é considerado molhado, e abaixo deste seco. Vale ressaltar que mesmo com a utilização de sensores de diferentes marcas, os procedimentos para determinação do IMF foram os mesmos, de acordo com Sentelhas (2004).

Para a Base de Dados 2 (Tsukahara, 2005), foi utilizado o sensor Davis®, cujos valores da resistência elétrica oscilam entre 0 (totalmente seco) e 15 (totalmente saturado-molhado). Conforme Tsukahara (2005), através de testes realizados em laboratório,

definiu-se que a transição entre seco e molhado acontecia quando o sensor apresentava valores de resistência superiores a 5. Desta forma, adotou-se como uma condição de molhado (presença de orvalho) quando os valores estão acima de 5 em pelo menos metade do tempo do intervalo de leitura (5 minutos).

No caso do sensor modelo 237 da Campbell® usado no experimento da Base de Dados 3 (ESALQ, 2005), os valores de resistência elétrica variaram entre 0 (totalmente seco) e 1 (totalmente saturado), onde foram consideradas como presença do orvalho as leituras que apresentaram valores de resistência superiores a 0,51 em pelo menos metade do tempo do intervalo de leitura (5 minutos) (Sentelhas, 2004).

Diante deste contexto, as RNAs geradas a partir das Bases de Dados 2 e 3, tiveram como objetivo classificar o molhamento foliar em 2 padrões distintos, seco ou molhado e, conseqüentemente, o número de neurônios da camada de saída destas RNAs foi 2.

Uma vez que o problema tenha sido definido e os dados estejam disponíveis, os dados devem ser preparados para aplicação em uma RNA. Este procedimento é definido como o pré-processamento dos dados.

6.1.2 Pré-processamento dos dados

Esta etapa corresponde à preparação dos dados conforme o formato requerido pela metodologia computacional adotada. No caso das RNAs, estas processam dados de tipo numérico normalmente no intervalo entre -1 e 1.

Deste modo, dados de entrada devem ser tratados visando converter dados simbólicos, quando houver, em dados numéricos, assim como normalizar atributos que possuem valores muito discrepantes em relação aos demais.

Dentre as atividades desenvolvidas nesta etapa de preparação dos dados, fez-se a conversão dos arquivos, originalmente planilhas eletrônicas do tipo Microsoft Excel, para o formato texto com um padrão pré-definido, tendo em vista o formato requerido pelos simuladores SNNS e JavaNNS. Todas as atividades que envolveram a preparação dos dados foram realizadas com programas computacionais desenvolvidos especificamente para este propósito.

6.1.3 Projeto da RNA

Na etapa que consiste na escolha da arquitetura da RNA, foram considerados os seguintes componentes: determinação do número de neurônios da camada de entrada, quantidades de camadas e o número de neurônios da(s) camada(s) ocultas e o número de neurônios da camada de saída.

Na elaboração do projeto de uma RNA ideal para solucionar um determinado problema, uma das tarefas mais difíceis é determinar o número de elementos de processamento da camada oculta, bem como o número de camadas ocultas (Medeiros, 1999).

É necessário obter um modelo que não seja muito rígido a ponto de não modelar fielmente os dados. A idéia é que a rede responda de acordo com as características presentes nos dados de entrada e não exatamente igual aos dados de entrada.

- Determinação da camada de entrada

A capacidade de mapeamentos complexos das redes neurais cresce com o número de camadas e neurônios. Contudo, este aumento acarreta maior tempo de processamento, bem como, uma considerável soma de dados (Ballini et al., 2003).

Para a determinação do número de neurônios e quais as variáveis climáticas seriam representadas em cada neurônio da camada de entrada, foram realizados diversos experimentos para descobrir qual a melhor combinação que proporcionava a menor taxa de erro no treinamento da RNA. Este resultado também buscou definir quais as variáveis climáticas que influenciavam significativamente a ocorrência do molhamento foliar, do ponto de vista das RNAs.

No desenvolvimento destes experimentos, a quantidade de neurônios da camada de entrada variou entre 2 a 13 parâmetros, correspondendo às combinações realizadas com as variáveis meteorológicas: umidade relativa média do ar (%), umidade relativa mínima do ar (%), umidade relativa máxima do ar (%), temperatura média do ar (°C), temperatura mínima do ar (°C), temperatura máxima do ar (°C), temperatura do bulbo seco (°C), temperatura do bulbo úmido (°C), velocidade média do vento (m/s=metros por segundo), velocidade mínima do vento (m/s), velocidade máxima do vento (m/s), pressão atmosférica local (kPa=pressão barométrica) e precipitação pluviométrica (mm).

- Determinação das camadas ocultas

Esta é uma etapa que envolve subjetividade. Autores como Caudill (1990), assumem que o número de elementos das camadas ocultas é uma incógnita e o seu tamanho deve ser escolhido usando o bom senso, portanto, trata-se de um procedimento com forte apelo empírico.

Neste sentido, deve-se considerar que, se o número de camadas for muito grande, assim como, a quantidade de neurônios destas, a rede estará memorizando os padrões e com isso perdendo a capacidade de generalização (*over-training / over-fitting*). Por outro lado, um número pequeno de elementos pode levar a rede a um *underfitting*, quando a rede não consegue convergir durante o treinamento e, também, a mesma exigirá mais iterações, ou seja, um esforço computacional maior para atingir a precisão desejada. Segundo Weigend et al. (1990), a rede que melhor generaliza os dados é a menor rede apta a realizar o treinamento com os dados disponíveis.

Na resolução dos problemas deste trabalho, foram implementadas redes neurais com apenas uma camada oculta. A quantidade de neurônios da camada foi calculada, multiplicando-se por três o número de neurônios da camada de entrada e somando-se este resultado ao número de neurônios da camada de saída. Por exemplo, se na rede que está sendo projetada, estão sendo consideradas as variáveis climáticas umidade relativa média do ar, temperatura média do ar e velocidade média do vento, o que corresponde a três neurônios de entrada, e o IMF considerado é seco ou molhado, ou seja, dois neurônios de saída, o cálculo para determinar o número de neurônios é: $3 * 3 + 2 = 11$.

Assim, como foi destacado anteriormente, determinar o número de neurônios das camadas ocultas é um procedimento empírico, o método adotado neste trabalho não corresponde a uma técnica padrão nem a um modo que tenha sido matematicamente provado e considerado bom.

O motivo do uso desta padronização está relacionado à obtenção dos melhores resultados, os quais foram obtidos usando este cálculo, mas, possivelmente, isto está relacionado com a composição das bases de dados utilizadas e, talvez, com outros dados estes resultados possam não se repetir.

- Determinação da camada de saída

O número de neurônios da camada de saída está condicionado aos padrões de classificação de molhamento foliar, conforme as bases de dados utilizadas neste trabalho. Portanto, para as RNAs da base de dados 1 foram definidos 10 neurônios de saída e 2 neurônios para as bases de dados 2 e 3.

A partir da definição das arquiteturas das RNAs, o próximo passo é o treinamento, a partir do qual pode-se obter os resultados esperados.

6.1.4 Treinamento da RNA

Pode-se dizer que a partir desta etapa é que se obtém o sucesso ou o fracasso da rede, pois, aqui, submete-se a rede ao aprendizado, onde vários fatores são relevantes, dentre eles destacam-se: o algoritmo de treinamento, parâmetros de aprendizagem e número de repetições (ciclos ou épocas).

Todos estes fatores devem ser aplicados a partir de um conjunto de dados que possua alto grau de pertinência em relação ao escopo do problema a ser resolvido, assim como, um número elevado de dados para que, ao serem submetidos à rede, ofereçam um bom treinamento. É válida a regra que diz, quanto maior a experiência, melhor o desempenho. Se a rede for treinada por uma quantidade considerável de dados, ela será não somente capaz de aproximar qualquer função, como também estará apta a generalizar, proporcionando resultados coerentes para entradas jamais vistas (Rohn et al., 2003).

Conforme já destacado, dentre os algoritmos de treinamento testados, foram selecionados o *Back-Propagation* e o *Resilient Propagation* que seguem o paradigma de aprendizado supervisionado. A escolha destes algoritmos de treinamento deu-se em função de que o *Back-Propagation* é o mais utilizado e o *Resilient Propagation* foi o que apresentou os melhores resultados em mais de 100 experimentos realizados, onde os parâmetros climáticos foram combinados de formas diferentes para se determinar os melhores resultados.

Estes experimentos foram desenvolvidos com os simuladores SNNS e JavaNNS, objetivando avaliar a eficiência da metodologia computacional adotada neste trabalho, antes da definição e desenvolvimento do sistema PMNeural.

A seguir, estão descritos os três estudos de caso e seus melhores resultados, que correspondem, respectivamente, a cada uma das bases de dados especificadas na seção Material e Métodos.

6.2 Estudos de caso

6.2.1 Estudo de caso 1

Para a realização dos treinamentos e testes das RNAs, a base de dados foi dividida em dois conjuntos. Neste sentido, dos 3.888 registros desta base, 89,7% foram utilizados para o treinamento, equivalente a 3.488 registros, o restante 10,3% foram usados para teste, correspondendo a 400 registros (Mathias, 2005).

A escolha dos registros do conjunto de teste foi o método da amostragem sistemática (Soares et al., 1991). Embora a probabilidade de seleção tenha sido a mesma para todos os registros, na amostra houve uma maior ocorrência das leituras correspondentes a classe 3 (molhamento em mais de 50% da superfície da folha - ocorrência de molhamento), tendo em vista ocorrerem em 71,2% do total da base original. É um modo de manter a proporcionalidade em relação às ocorrências presentes nos dados originais. A distribuição dos registros pode ser observada parcialmente na Tabela 9.

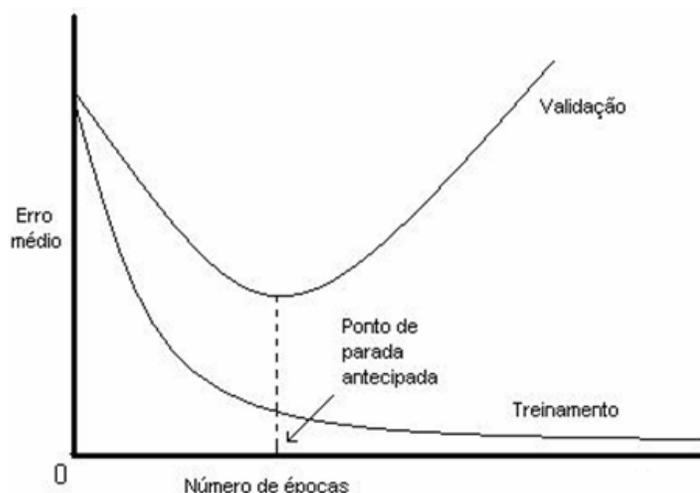
Tabela 9 – Detalhes - base de dados 1

Datas	Horário das leituras (registros)	Número de leituras (registros)
11/08/2003	18:05/23:55	71
19/08/2003	00:00/23:55	216
20/08/2003	00:00/23:55	216
05/09/2003	00:00/12:00	145
11/09/2003	18:05/23:55	71
12/09/2003	00:00/12:00	145
...
08/10/2003	18:05/23:55	71
09/10/2003	00:00/12:00	145
Total de registros		3888

O objetivo desta divisão é a obtenção de um modelo que tenha uma boa generalização, pois o mesmo deve responder corretamente aos exemplos contidos na base de aprendizado, mas, também, a outros exemplos diferentes daqueles usados durante o processo de treinamento.

Esta técnica de divisão da base de dados em dois subconjuntos é uma ferramenta padrão da estatística conhecida como validação cruzada (Haykin, 2001), através da qual, pode-se definir o número ideal de épocas de treinamento de RNAs.

Este método consiste em acompanhar em um gráfico, conforme ilustrado na Figura 17, a evolução do aprendizado nas curvas correspondentes aos subconjuntos de dados de treinamento e validação.



Fonte: Haykin (2001)

Figura 17 – Regra de parada antecipada baseada na validação cruzada

O objetivo é identificar um ponto de parada de aprendizado, para que a rede não fique excessivamente ajustada aos dados de treinamento (super-aprendizado), e obtenha-se a melhor generalização da rede. Deste modo, o treinamento é interrompido quando a curva de validação decresce a um erro mínimo, e antes de começar a crescer conforme o treinamento continua.

Na Tabela 10, estão descritos quatro dos melhores resultados obtidos com esta base de dados, onde estão destacadas: as variáveis de entrada, arquiteturas das

RNAs, épocas de treinamento, e os erros obtidos com os algoritmos de treinamento *Rprop* e *Back-propagation*.

Tabela 10 – Melhores resultados c/simuladores – base de dados 1

Variáveis de Entrada	Arquiteturas	Épocas	Erros de Algoritmos de Treinamento		
			Etapas	<i>Rprop</i>	<i>Backprop</i>
HORÁRIO + URMED + TEMPMED + VENTMED	4-22-10	11400	Treino	0.19753	0.36773
			Validação	0.20639	1.62107
HORARIO + URMAX + TEMPMIN	3-19-10	12000	Treino	0.21425	0.36773
			Validação	0.20475	1.62107
HORARIO + URMAX	2-16-10	7600	Treino	0.25845	0.36773
			Validação	0.25969	1.62107
URMAX + TEMPMIN	2-16-10	10000	Treino	0.34398	0.41638
			Validação	0.26467	1.34183
<i>Resilient Propagation (Rprop) - Back-Propagation (Backprop)</i>					

Com relação ao formato de exibição das arquiteturas testadas (Tabela 10), por exemplo, 4-22-10, o valor 4, corresponde ao número de neurônios da camada de entrada, 22 são os neurônios da camada oculta, e a camada de saída é formada por 10 neurônios.

6.2.2 Estudo de caso 2

Nesta base também foi utilizada a técnica da validação cruzada, onde dos 21.610 registros, 19.610 equivalente a 90,7% do total da base, foram usados para o treinamento e 2.000 registros - 9,3% utilizados para os testes. A seleção dos registros foi feita utilizando-se o método da amostragem aleatória simples (Soares et al., 1991) e não houve priorização em registros que indicassem a maioria, visto que, 12.035 ocorrências - 55,7% do total – indicavam a condição de seco, o restante dos registros, 9.575 registros, 44,3%, molhado. Na Tabela 11 são apresentados os melhores resultados deste estudo de caso.

Tabela 11 – Melhores resultados c/simuladores – base de dados 2

Variáveis de Entrada	Arquiteturas	Épocas	Erros de Algoritmos de Treinamento		
			Etapas	<i>Rprop</i>	<i>Backprop</i>
HORÁRIO + URMED + TEMPMED + CHUVA	4-14-2	1600	Treino	0.21493	0.14808
			Validação	0.32596	0.50315
HORARIO + URMED + TEMPMED	3-11-2	750	Treino	0.25724	0.17282
			Validação	0.27444	0.56399
HORARIO + URMED	2-8-2	600	Treino	0.27089	0.17282
			Validação	0.30798	0.56399
URMED + TEMPMED	2-8-2	500	Treino	0.29080	0.18500
			Validação	0.28944	0.58500
<i>Resilient Propagation (Rprop) - Back-Propagation (Backprop)</i>					

6.2.3 Estudo de caso 3

Assim como nos estudos de caso 1 e 2, o uso da técnica da validação cruzada foi mantida, e dos 5.856 registros desta base, 5.356 equivalentes a 91,5% do total, foram usados para o treinamento, e 500 registros, 8,5% utilizados para os testes. A amostragem aleatória simples na escolha dos registros, também foi o modo operacional de seleção, onde, a distribuição os dados corresponde a: 3.667 registros – 62,6% - com indicação de seco e 2.189 – 37,4% - molhado. Na Tabela 12 estão ilustrados os melhores resultados obtidos.

Tabela 12 – Melhores resultados c/simuladores – base de dados 3

Variáveis de Entrada	Arquiteturas	Épocas	Erros de Algoritmos de Treinamento		
			Etapas	<i>Rprop</i>	<i>Backprop</i>
HORÁRIO + URMED + TEMPMED + VENTMED	4-14-2	1800	Treino	0.07209	0.32747
			Validação	0.06330	0.43402
URMED + TEMPMED + VENTMED	3-11-2	1500	Treino	0.08594	0.32747
			Validação	0.06655	0.43402
HORARIO + URMED + TEMPMED	3-11-2	1200	Treino	0.09919	0.34155
			Validação	0.08502	0.43427
HORARIO + URMED	2-8-2	700	Treino	0.11123	0.37679
			Validação	0.07135	0.44444
URMED + TEMPMED	2-8-2	1350	Treino	0.10323	0.39879
			Validação	0.08481	0.45099
<i>Resilient Propagation (Rprop) - Back-Propagation (Backprop)</i>					

Apesar de inesperado, dentre as diversas variáveis climáticas utilizadas para classificação do molhamento foliar, a inclusão do horário influenciou a obtenção dos resultados das RNAs. Isto pode ser explicado pelo fato de que, nas bases de dados utilizadas, a formação do orvalho apresentou horários diários semelhantes para seu início e final, ou seja, foi observado um padrão nos horários de ocorrência de orvalho. Este padrão foi captado pelas RNAs, porém é possível que este mesmo resultado não se repita com outros dados climáticos.

Na Figura 18 está ilustrada uma das telas do simulador SNNS, onde podem ser observados detalhes das evoluções do treinamento, linha preta, e validação, linha vermelha, de um dos ensaios da Tabela 12. Este exemplo corresponde a RNA com arquitetura 4-14-2 (Figura 19), variáveis climáticas – HORÁRIO + URMED + TEMPMED + VENTMED –, algoritmo de treinamento Rprop.

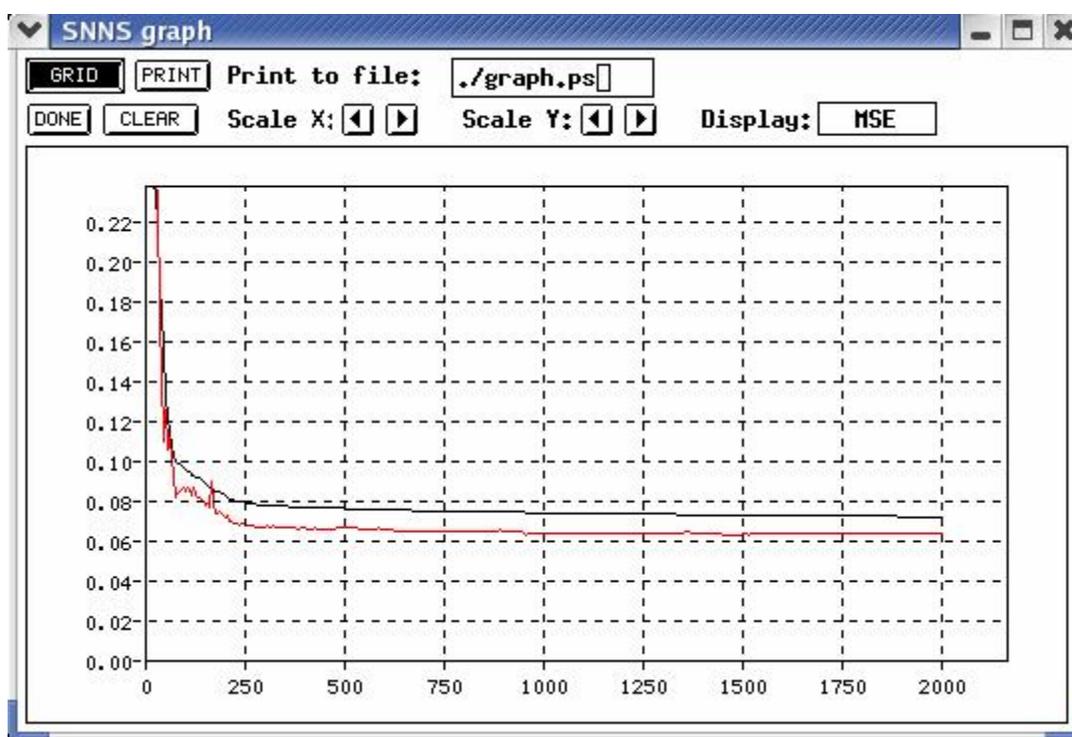


Figura 18 – Tela SNNS – gráfico de evolução de treinamento/validação

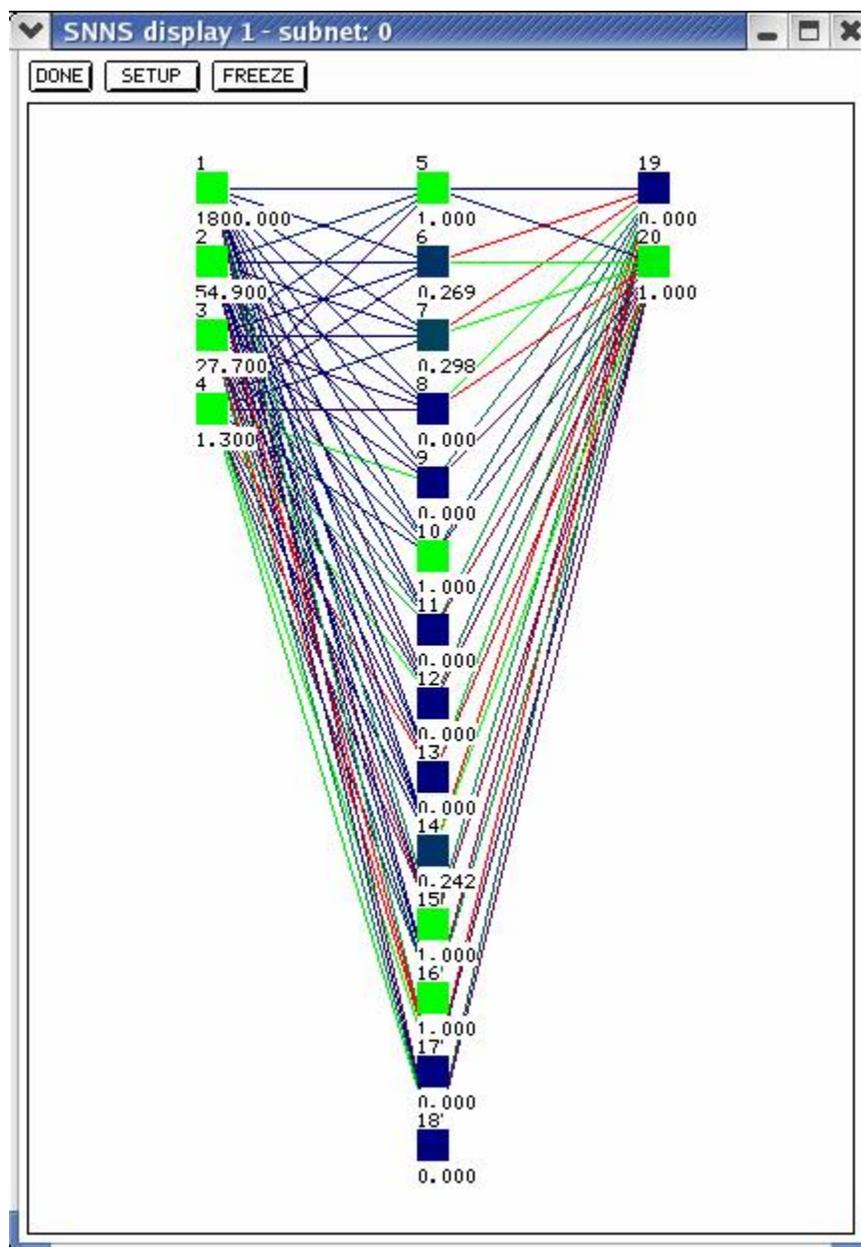


Figura 19 – Tela SNNS – exibição de RNA com arquitetura 4-14-2

Na Figura 20, pode-se observar detalhes do painel de controle do SNNS, que corresponde a uma das etapas do processo de treinamento de uma RNA, onde são informados e selecionados diversos itens para dar início ao aprendizado. Os principais itens são: CYCLES - número de épocas de treinamento; LEARN - taxa de aprendizagem; os arquivos de padrões para o treinamento e validação; SEL FUNC - algoritmo de aprendizado.

Uma vez que todos os itens sejam informados, clica-se no botão ALL para o processo se iniciar.

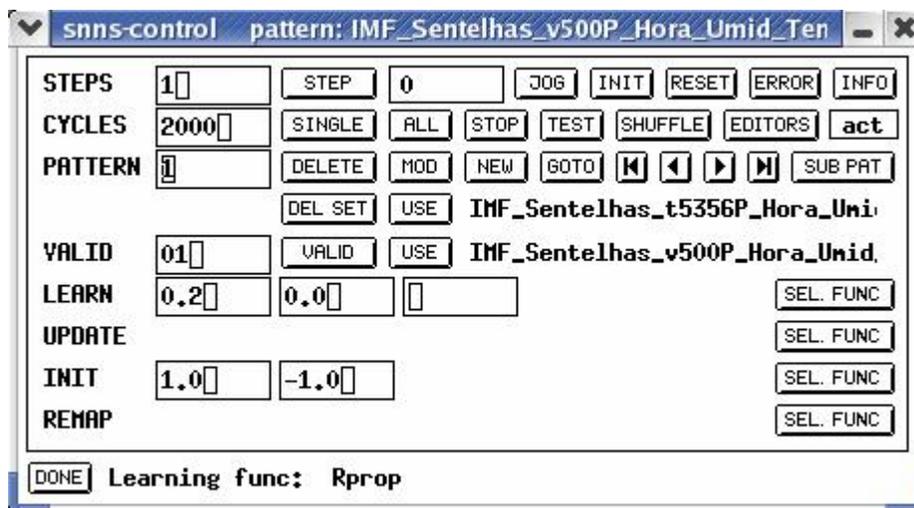


Figura 20 – Tela SNNS – painel de controle

Os resultados obtidos nestes estudos de caso foram satisfatórios para demonstrar a eficiência do uso de RNAs no tratamento de dados climáticos visando classificar o molhamento foliar. Apesar disso, como as arquiteturas das RNAs foram definidas com base nas diferentes combinações efetuadas com as variáveis climáticas, outras configurações, possivelmente, podem apresentar desempenho similar, ou até melhor. Contudo, todos os cenários testados foram decisivos para o desenvolvimento do sistema PMNeural, o qual está definido a seguir.

6.3 Definição do sistema PMNeural

Após a realização dos estudos com os simuladores, pode-se determinar a melhor forma de ajustar os dados climáticos ao paradigma das RNAs, bem como, a determinação dos melhores algoritmos de treinamento para os estudos de caso em questão, itens fundamentais para definição do sistema PMNeural, cuja modelagem foi baseada em Stadzisz (2002) e Fowler (2005).

Em um projeto de *software* onde são utilizados recursos de modelagem UML (*Unified Modeling Language*) (Furlan, 1998), dois componentes principais são destacados, os atores e os casos de uso. Os atores correspondem às pessoas que interagem com o sistema, caracterizados no PMNeural como *administrador* ou *técnico*, distinguindo os níveis de acesso ao sistema com permissões diferenciadas.

O *administrador* possui todas as permissões sobre o sistema, como:

- Cadastrar usuários do sistema (inclusão, exclusão e alteração);
- Cadastrar estações meteorológicas (inclusão, exclusão e alteração);
- Cadastrar cidades (inclusão, exclusão e alteração);
- Cadastrar variáveis que serão analisadas (inclusão, exclusão e alteração);
- Definição dos parâmetros da rede neural a ser utilizada (numero de camadas ocultas, taxa de inércia, taxa de aprendizado, número de épocas);

O *técnico* possui permissões limitadas:

- Executar treinamentos e testes com os dados coletados;
- Visualizar os resultados;
- Armazenar os resultados;
- Gerar relatórios;

Os casos de uso do sistema PMNeural correspondem a:

1. Realizar *login* no sistema;
2. Cadastrar usuário
 - a. Incluir usuário;
 - b. Excluir usuário;
 - c. Alterar usuário;

3. Cadastrar estações meteorológicas;
 - a. Incluir estação meteorológica;
 - b. Excluir estação meteorológica;
 - c. Alterar estação meteorológica;
4. Cadastrar cidades;
 - a. Incluir cidades;
 - b. Excluir cidades;
 - c. Alterar cidades;
5. Cadastrar variáveis;
 - a. Incluir variável;
 - b. Excluir variável;
 - c. Alterar variável;
6. Importar dados;
7. Treinar RNA
 - a. Escolher os dados de entrada;
 - b. Escolher a arquitetura da rede;
 - c. Escolher o algoritmo de treinamento;
 - d. Escolher os parâmetros de treinamento;
 - e. Realizar o treinamento;
 - f. Visualizar resultados;
 - g. Armazenar relatório de treinamento;
8. Executar a validação;
 - a. Escolher a RNA;
 - b. Escolher os dados de validação;
 - c. Realizar a validação da RNA;
 - d. Visualizar resultados;
 - e. Armazenar resultados;
 - f. Gerar relatórios.

A partir dos casos de uso o sistema PMNeural foi desenvolvido para classificar o molhamento foliar em índices, cujas funcionalidades e formas de execução estão descritas a seguir.

6.4 Funcionalidades do sistema PMNeural

O sistema desenvolvido, denominado PMNeural, foi codificado no ambiente de programação de computadores Borland Delphi Enterprise versão 7.0 (Borland, 2003; Cantu, 2003), o qual pode ser executado na plataforma Windows. Para a implantação das RNAs foi utilizado o componente TMLP (*Training Multilayer Perceptron*) obtido em Medeiros (2003), cujo componente, foi modificado visando as funcionalidades exigidas pelos estudos de caso, assim como, o desenvolvimento do algoritmo de treinamento *Resilient Propagation*.

Ao ser iniciada a execução do sistema a tela ilustrada na Figura 21 é exibida, após algum tempo esta interface se modifica, solicitando a identificação do usuário para a entrada do *login* e da senha de acesso, conforme pode ser observado na Figura 22.

O tempo de exibição da primeira tela do sistema pode variar conforme o computador em uso. O que pode motivar esta situação é a utilização do sistema em computadores com diferentes velocidades de processador e capacidade de memória RAM.



Figura 21 – Tela de apresentação – PMNeural



Figura 22 – Tela de operações e login

Conforme já visto, existem dois tipos de usuários, *administrador* ou *técnico*, cujas permissões são distintas. A diferença está em que o *técnico* não pode efetuar

nenhum tipo de cadastro, já o *administrador* tem acesso a todas as tarefas do sistema, situação que será descrita a seguir.

Uma vez que o *administrador* faça sua identificação, a tela resultante corresponde a exibida na Figura 23, na qual tornam-se disponíveis as opções da barra de menu: Cadastros, Importação, Ensaios, Ajuda, Sobre e Sair.

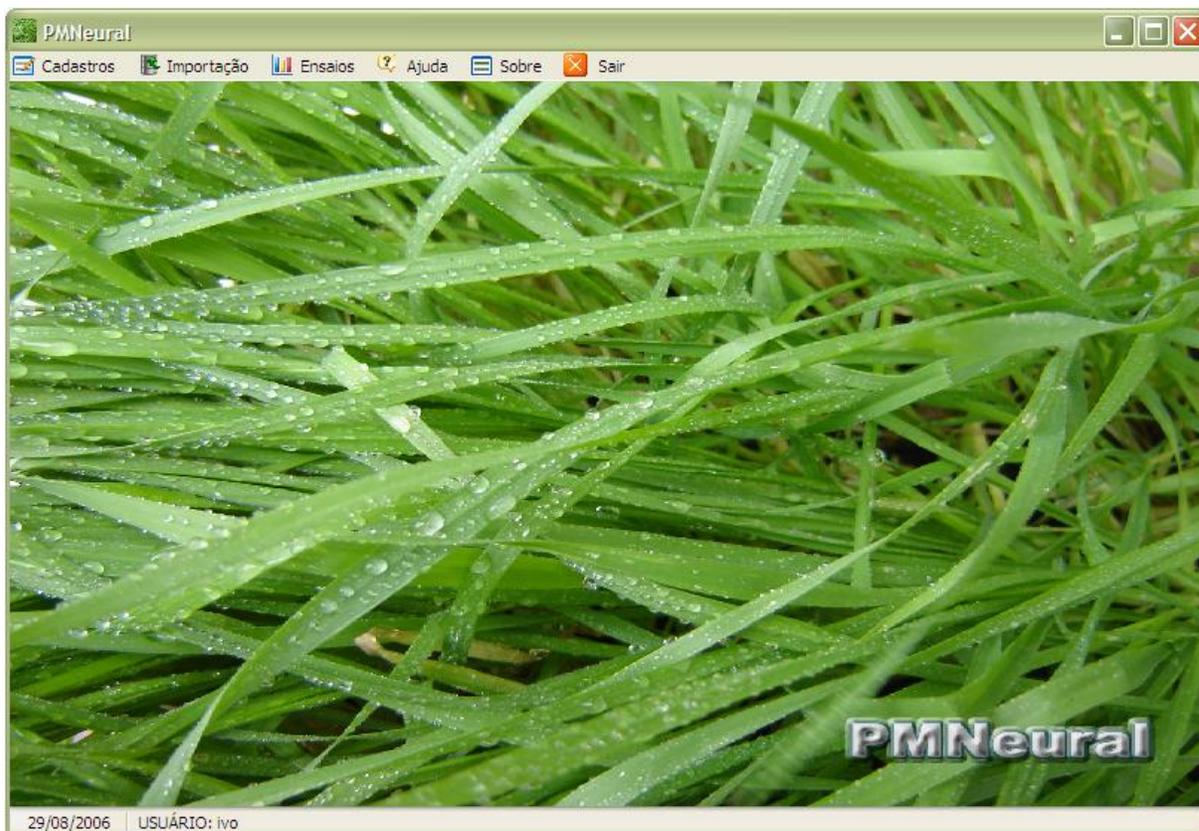


Figura 23 – Tela de operações

A partir da opção *Cadastros* podem ser cadastrados: usuários, estações meteorológicas, cidades e variáveis. Para todos os quatro casos também são disponibilizadas opções para: Incluir, Excluir, Alterar, Ajuda e Sair. As telas de cadastros são padronizadas conforme o exemplo ilustrado na Figura 24, que corresponde ao cadastro de usuário.

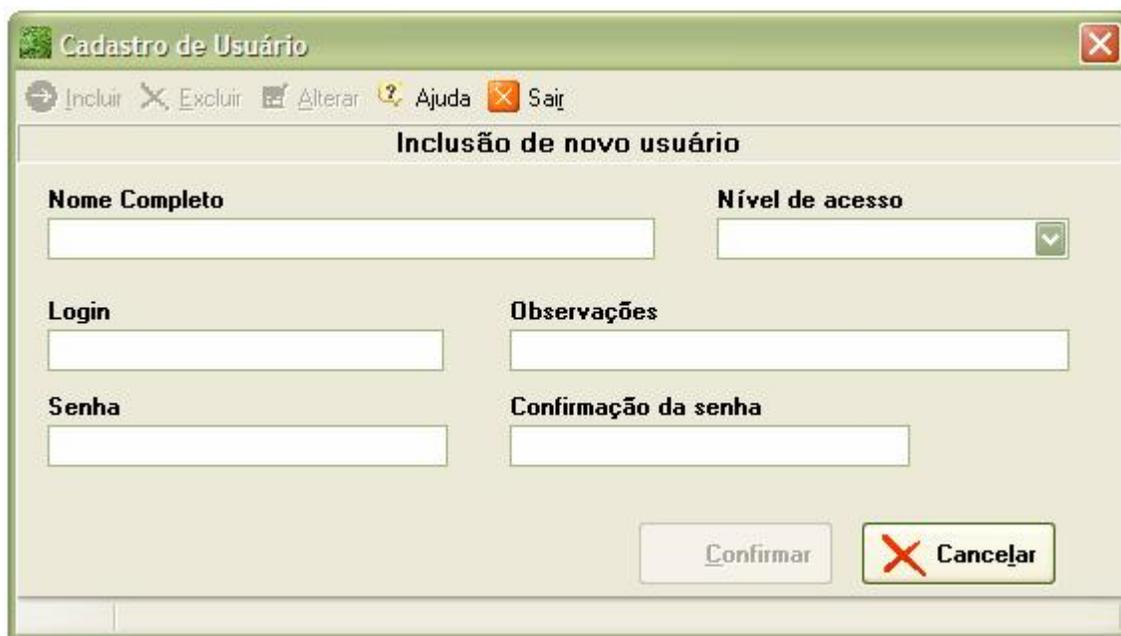


Login	Nome	Descrição	Observação
admin	administrador	Administrador	
ivo	Ivo Mário Mathias	Administrador	Administrador do sistema

Figura 24 – Tela – acesso ao cadastro de usuário

A tela de *Cadastro de usuário* é composta pelos seguintes dados (Figura 25):

- **Nome completo;**
- **Nível de acesso** – Administrador ou Técnico;
- **Login** – identificação de acesso do usuário;
- **Observações** – anotações gerais sobre o usuário;
- **Senha** – para efetuar o *login* no sistema;
- **Confirmação da senha.**



The image shows a software window titled "Cadastro de Usuário" (User Registration). The window has a menu bar with the following items: "Incluir" (Add), "Excluir" (Delete), "Alterar" (Edit), "Ajuda" (Help), and "Sair" (Exit). The main content area is titled "Inclusão de novo usuário" (Add new user). It contains several input fields: "Nome Completo" (Full Name), "Nível de acesso" (Access Level) which is a dropdown menu, "Login", "Observações" (Observations), "Senha" (Password), and "Confirmação da senha" (Confirm Password). At the bottom right, there are two buttons: "Confirmar" (Confirm) and "Cancelar" (Cancel).

Figura 25 – Tela – cadastro de usuário

Na Figura 26, pode-se visualizar a tela para o cadastro das estações meteorológicas, onde são cadastradas as informações relativas a origem dos dados climáticos, que são utilizados no treinamento e validação das RNAs. Este cadastro é constituído pelas seguintes informações:

- **Código da estação** – composto por 8 dígitos numéricos que correspondem aos graus e minutos (4 dígitos) da latitude mais a longitude;
- **Número da estação** – conforme numeração utilizada para identificar a estação, ou quando existe mais de uma estação no mesmo local;
- **Modelo** – de acordo com o marca/modelo;
- **Identificação** – qualquer informação que auxilie na identificação da estação;
- **Proprietário** – do imóvel ou da estação;
- **Localização** – local onde esta instalada a estação, por exemplo: Fazenda Capão do Cipó;
- **Culturas** – culturas existentes na área de abrangência da estação;
- **Outros equipamentos** – informação adicional sobre outros equipamentos usados na estação;

- **Observações** – quaisquer informações complementares;
- **Estado** – unidade da federação;
- **Cidade** – município;
- **Altitude**;
- **Longitude**;
- **Latitude**;
- **Área total** – hectares.



The image shows a software window titled "Cadastro de Estações Meteorológicas" with a standard menu bar containing "Incluir", "Excluir", "Alterar", "Ajuda", and "Sair". The main area is titled "Inclusão de Nova Estação" and contains a form with the following fields:

Código da estação	<input type="text"/>	Estado	<input type="text"/>
Número da estação	<input type="text"/>	Cidade	<input type="text"/>
Modelo	<input type="text"/>	Altitude	<input type="text"/>
Identificação	<input type="text"/>	Latitude	<input type="text"/>
Proprietário	<input type="text"/>	Longitude	<input type="text"/>
Localização	<input type="text"/>	Área total (ha)	<input type="text"/>
Culturas	<input type="text"/>		
Outros equipamentos	<input type="text"/>		
Observações	<input type="text"/>		

At the bottom right of the form, there are two buttons: "Confirmar" and "Cancelar".

Figura 26 – Tela – cadastro de estações meteorológicas

O próximo cadastro é o das cidades, o qual está vinculado ao cadastro das estações meteorológicas, pois tem por objetivo indicar o município de localização de

determinada estação. Na Figura 27 pode-se visualizar a tela de acesso à manutenção do cadastro de cidades.

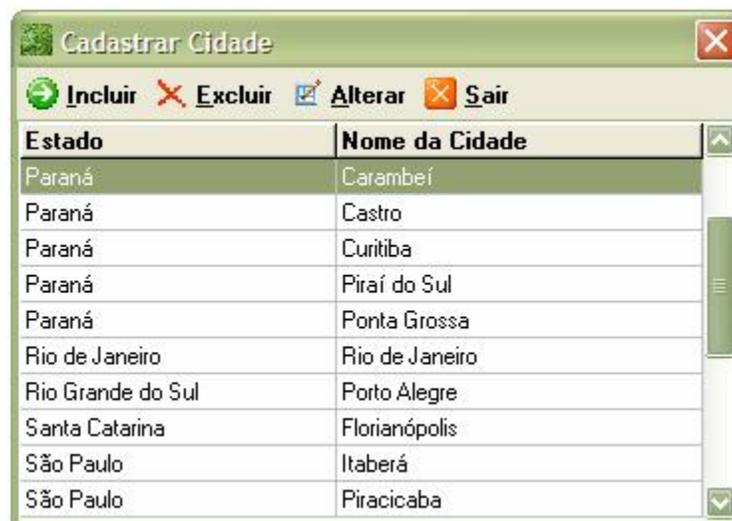


Figura 27 – Tela – cadastro de cidades

O cadastro subsequente corresponde as variáveis utilizadas pelo sistema, as quais correspondem a identificação dos dados utilizados para o treinamento e validação das RNAs.

O objetivo principal dessas variáveis é identificar os diversos tipos de dados, bem como, o tipo de dado associado à mesma e conseqüentemente a informação relacionada. Na Figura 28, pode-se observar um quadro com alguns exemplos de variáveis.

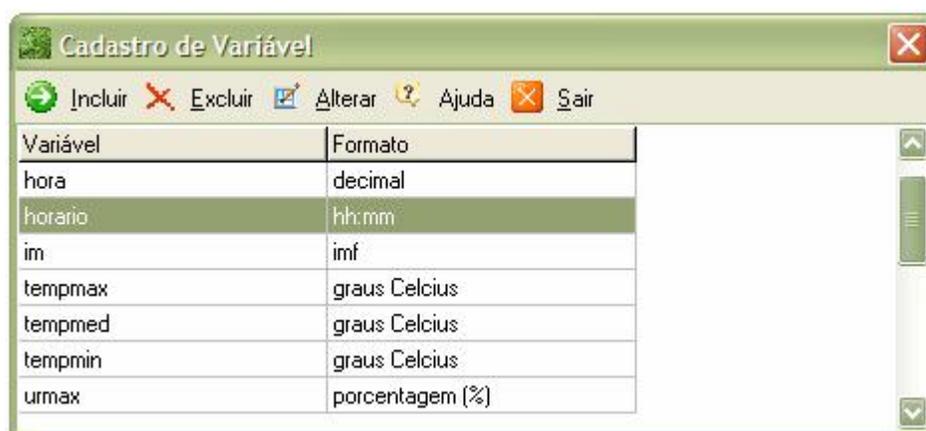


Figura 28 – Tela – exemplos de variáveis

Assim como, nos demais cadastros, existe uma tela correspondente para sua manutenção, Figura 29. Este cadastro é constituído por duas informações.

- **Nome** – é a identificação da variável e as regras para criação de nomes de variáveis são as seguintes: deve começar necessariamente com uma letra e não deve conter nenhum símbolo especial exceto sublinha (*underline*);
- **Formato** – corresponde ao formato do dado numérico associado a variável.

A imagem mostra uma janela de software intitulada "Cadastro de Variável". No topo, há uma barra de menu com ícones e o texto "Incluir", "Excluir", "Alterar", "Ajuda" e "Sair". Abaixo disso, o título "Inclusão de nova variável" está centralizado. O formulário contém dois campos principais: "Nome", que é um campo de texto vazio, e "Formato", que é um menu suspenso com uma seta verde apontando para baixo. À direita do campo "Formato", há dois botões: "Confirmar" e "Cancelar" (com um ícone de X vermelho).

Figura 29 – Tela – cadastro de variável

O cadastro de variáveis tem importância fundamental na funcionalidade do sistema, pois, as variáveis cadastradas estão diretamente relacionadas aos neurônios das RNAs e também a importação dos dados, onde, os dados são normalizados quando necessário.

Por exemplo, no caso da variável *horario*, foi pré-definido o seu formato como *hh:mm* (hora e minutos), contudo, este é um formato não decimal e portanto não é possível ser utilizado como entrada em uma RNA. Sendo assim, quando o sistema PMNeural identifica uma variável com este formato (hh:mm) durante a importação, automaticamente transforma este dado em decimal, substituindo os dois pontos (:) por vírgula (,) exemplificando: 18:15 (dezoito horas e quinze minutos) é transformando no valor 18,15 (dezoito vírgula quinze).

Para realizar a importação de dados basta escolher a opção correspondente na barra de menu. A Figura 30 ilustra a tela correspondente e um exemplo deste procedimento.

hora	urmed	tempmed	ventmed	im
18,05	76,78	9,99	0,1	1
18,1	75,32	10,17	0,78	1
18,15	75,87	9,99	0,98	1
18,2	77,97	9,56	1,67	1
18,25	78,89	9,32	1,76	1

Figura 30 – Tela – importação de dados

- **Código de Importação** – é um valor de controle inserido automaticamente pelo sistema;
- **Estação Meteorológica** – é uma escolha que deve ser realizada com base nas estações meteorológicas cadastradas e conseqüentemente relacionadas a origem dos dados;
- **Data** – pode ser a corrente ou conforme a data de origem dos dados;
- **Arquivo de importação** – o sistema utiliza como padrão, arquivos do tipo planilha eletrônica no formato de gravação CSV (campos separados por vírgulas). Estes arquivos podem ser gerados pelos programas Microsoft Office

Excel ou BrOffice Calc. O local de gravação destes arquivos pode ser qualquer pasta do sistema, ou conforme determinado na instalação, recomendável;

- **Finalidade dos dados** – conforme a metodologia das RNAs, os dados podem ser utilizados para o treinamento ou validação (teste) das RNAs.

Após a inserção correta dos dados de importação, uma lista parcial dos dados é exibida, conforme exemplo ilustrado na Figura 30, onde, cada uma das colunas, previamente rotuladas com a identificação de cada atributo, deve estar diretamente relacionada ao cadastramento das variáveis. Situação que exige exatidão da nomenclatura das variáveis conforme os cabeçalhos dos arquivos de importação.

Durante o processo de importação, um quadro de dialogo como o ilustrado na Figura 31 é exibido, o qual tem por finalidade solicitar ao usuário para indicar a partir de qual linha do arquivo de origem os dados devem ser importados. Esta linha deve corresponder aquela que possui a identificação de cada atributo e cujos valores estarão expressos na referida coluna. Por uma questão de operacionalização, sempre a ultima coluna do arquivo deve corresponder ao *índice de molhamento*. Portanto, as colunas anteriores estarão representando as ocorrências das variáveis que formarão os neurônios da camada de entrada.

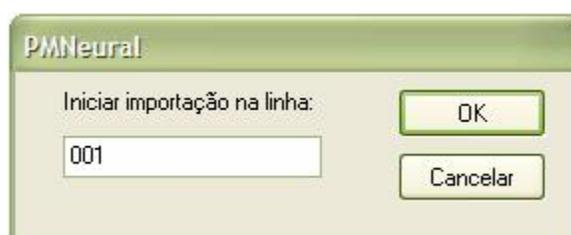


Figura 31 – Tela – iniciar importação na linha

Outro quadro de diálogo que é exibido durante a importação dos dados, Figura 32, é resultado de um processo automático que efetua uma varredura no arquivo, identificando os diferentes padrões de valores que serão usados para a determinação dos índices de molhamento, e automaticamente o número de neurônios da camada de saída da RNA.

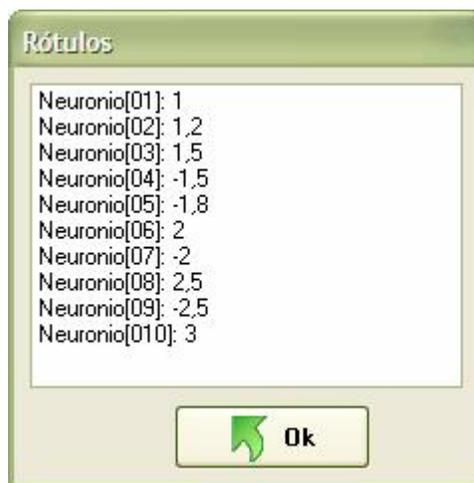


Figura 32 – Tela – neurônios da camada de saída

A próxima opção da barra de menu são os ensaios, os quais podem ser: treinamento e teste. A Figura 33 exemplifica uma tela com parâmetros para a realização de um treinamento de uma RNA, utilizando o algoritmo *Resilient Propagation*. Pode-se observar nesta tela, que a interface exibida é interativa e disponibiliza ao usuário todas as informações necessárias ao treinamento, bem como, um gráfico onde se pode acompanhar a evolução do erro em relação ao número de iterações (épocas).

Treinamento

Número do treinamento: 1

Data: 29/08/2006 Estação meteorológica: 22434725 - Estudo de caso 3

Importações

- 27/08/2006 - 1
- 26/08/2006 - 3
- 26/08/2006 - 1
- 18/08/2006 - 1
- 01/08/2006 - 15
- 01/08/2006 - 14
- 01/08/2006 - 13
- 01/08/2006 - 12
- 01/08/2006 - 11
- 31/07/2006 - 6
- 31/07/2006 - 5
- 31/07/2006 - 3
- 31/07/2006 - 2
- 31/07/2006 - 1
- 27/07/2006 - 3

Variáveis

- hora
- urmed
- im

Selecionar todas as variáveis

hora	urmed	im
1215	39,4	0
1230	39	0
1245	38,3	0
1300	39,1	0
1315	37,9	0
1330	39,2	0
1345	38,5	0
1400	37,4	0
1415	37	0
1430	37,2	0
1445	37,5	0

Escolha um Algoritmo

Resilient Propagation Back Propagation

Resilient Propagation

Número de Épocas: 700

Neurônios Camada Oculta: 8

Fator Crescente: 1,2

Fator Decrescente: 0,5

Delta Inicial: 0,1

Delta Mínimo: 0,00001

Delta Máximo: 50

Número de Observações: 5356

Botões: Treinar, Interromper, Fechar

Evolução do Erro de Treinamento

Épocas: 700 Erro Total: 0,055746

Figura 33 – Tela – treinamento da RNA

O procedimento inicial é definir qual estação meteorológica será usada, em seguida selecionar qual conjunto de dados (Importações) que se quer utilizar para treinar. Posteriormente deve-se escolher com quais *Variáveis* do conjunto de dados se deseja realizar o treinamento, podem ser, todas ou optar por algumas especificamente. Neste procedimento da escolha das variáveis de treinamento, esta sendo definido o número de neurônios da camada de entrada da RNA.

O próximo passo é a escolha do algoritmo de treinamento: *Resilient Propagation* ou *Back Propagation*.

Para o algoritmo *Back-propagation* os parâmetros são:

- **Número de Épocas;**
- **Neurônios Camada Oculta;**
- **Taxa de Aprendizagem;**
- **Taxa de Inércia.**

O algoritmo *Resilient Proagation* necessita dos seguintes parâmetros:

- **Número de Épocas;**
- **Neurônios Camada Oculta;**
- **Fator Crescente** – fator de atualização crescente dos pesos sinápticos;
- **Fator Decrescente** – fator de atualização decrescente dos pesos sinápticos;
- **Delta Inicial** – determina o acréscimo inicial dos pesos sinápticos;
- **Delta Mínimo** – determina o valor mínimo de atualização dos pesos sinápticos;
- **Delta Máximo** – determina o valor máximo de atualização dos pesos sinápticos.

Para todos os parâmetros do processo de treinamento existem valores recomendados, mas podem ser alterados pelo usuário conforme as suas necessidades. O número de épocas é o parâmetro que exige maior atenção do usuário, por ser o número de iterações que vai determinar a convergência da rede treinada. Neste sistema esta escolha é empírica. O número de observações é um valor fornecido automaticamente pelo sistema, conforme a quantidade de registros existentes no arquivo de importação correspondente.

O próximo requisito para o treinamento é a construção da RNA, que é obtida clicando-se no botão **Treinar**, o qual, ao ser clicado emite uma mensagem solicitando uma confirmação, se o usuário realmente deseja iniciar o treinamento da rede. O tempo de treinamento das redes é influenciado diretamente pelo número de épocas, número de observações e número de neurônios nas diversas camadas da RNA, assim como, pela capacidade de processamento do computador em uso e quantidade de memória RAM, sendo assim, o tempo de treinamento de uma RNA pode variar entre segundos, minutos e horas.

Uma vez que o treinamento seja realizado com sucesso, um relatório como o da Figura 34 é exibido.



Figura 34 – Tela – relatório de treinamento

Neste relatório são destacados todos os parâmetros utilizados no treinamento, o gráfico do erro de treinamento e também o erro obtido. A partir de um treinamento em que o usuário considere satisfatório o erro obtido, este relatório pode ser gravado em disco, bastando para isso clicar no botão *Salvar*. Este procedimento se constitui no armazenamento da RNA treinada no banco de dados do sistema.

Uma vez que, se tenha obtido uma RNA treinada para um determinado conjunto de dados, é necessário validar este treinamento. Isto pode ser feito a partir da opção *Teste em Ensaios* da barra de menu. A Figura 35 exibe uma tela de exemplo.

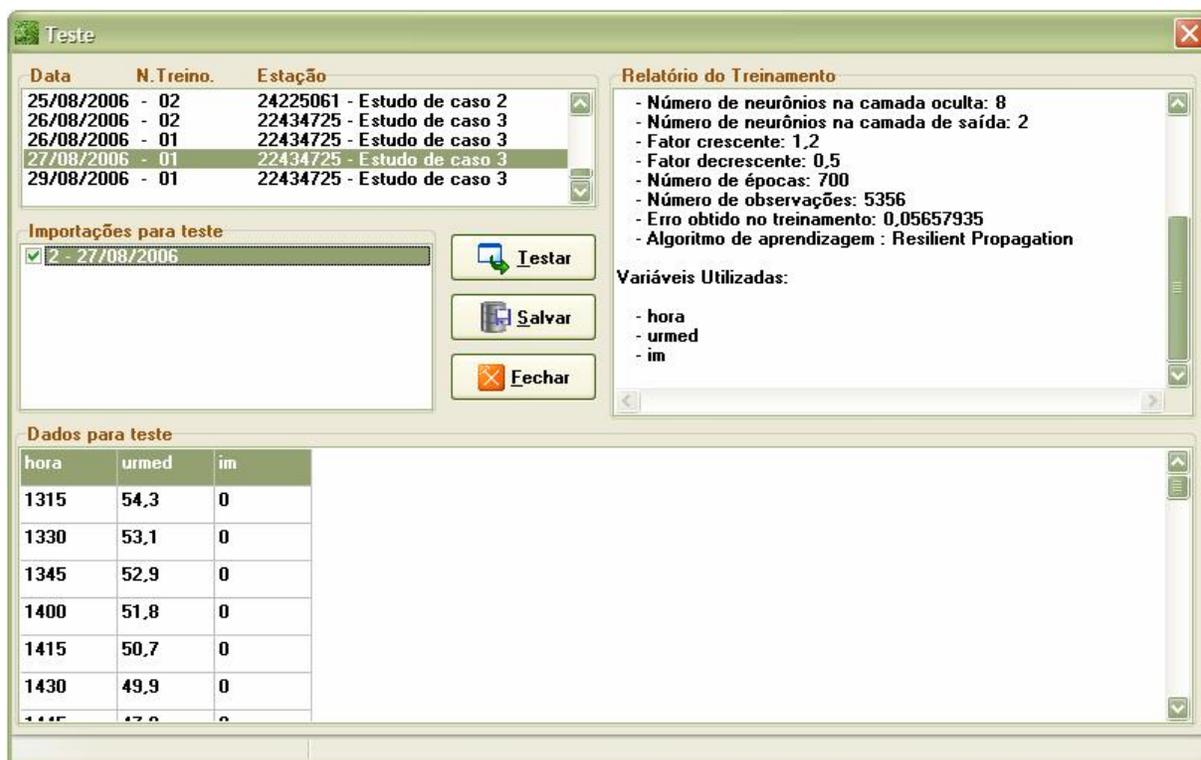


Figura 35 – Tela – validação de RNAs

Para realizar um teste é preciso escolher a RNA que será testada. Isto é feito ao selecionar uma determinada *Data* com o respectivo *N.Treino* e a *Estação*. Cada uma dessas opções corresponde a uma RNA previamente treinada. Ao clicar-se duas vezes sobre a RNA escolhida, são exibidos os dados do *Relatório do Treinamento* e no quadro *Importações para teste*, são disponibilizados os dados previamente importados para testes relativos à RNA em uso. Uma vez que se escolha um dos arquivos para teste, uma lista dos dados é exibida (Figura 35). Neste ponto pode-se clicar o botão *Testar* e o processo têm início. Assim como no treinamento, o tempo de execução é variável, mas geralmente é inferior a um minuto, evidentemente, proporcional à quantidade de registros do conjunto de dados usados para o teste.

Após a conclusão do teste, um relatório como o ilustrado na Figura 36 é exibido. Para uma visualização geral deve-se usar a barra de rolagem na lateral direita da janela. Neste relatório podem ser visualizados todos os erros e/ou acertos de cada um dos neurônios de saída estimados pela RNA em relação aos IMF da base de testes. No final do

relatório, são computados os totais e percentuais de acertos e erros obtidos. Para uma posterior consulta o relatório pode ser gravado em arquivo, clicando-se no botão *Salvar*.

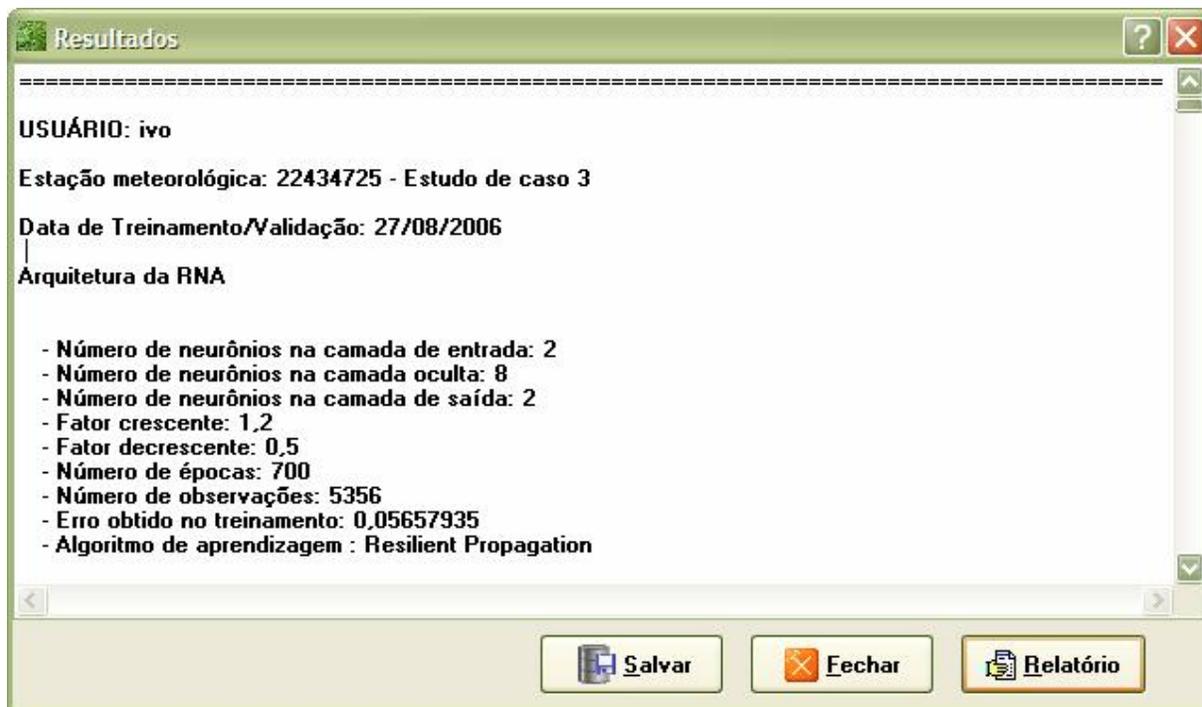


Figura 36 – Tela – resultados de validação

Além do relatório completo, pode ser obtido um relatório resumido (Figura 37), clicando-se no botão *Relatório*, o qual pode ser impresso para facilitar ao usuário avaliar a eficiência da rede em relação a dados não utilizados na fase de treinamento.

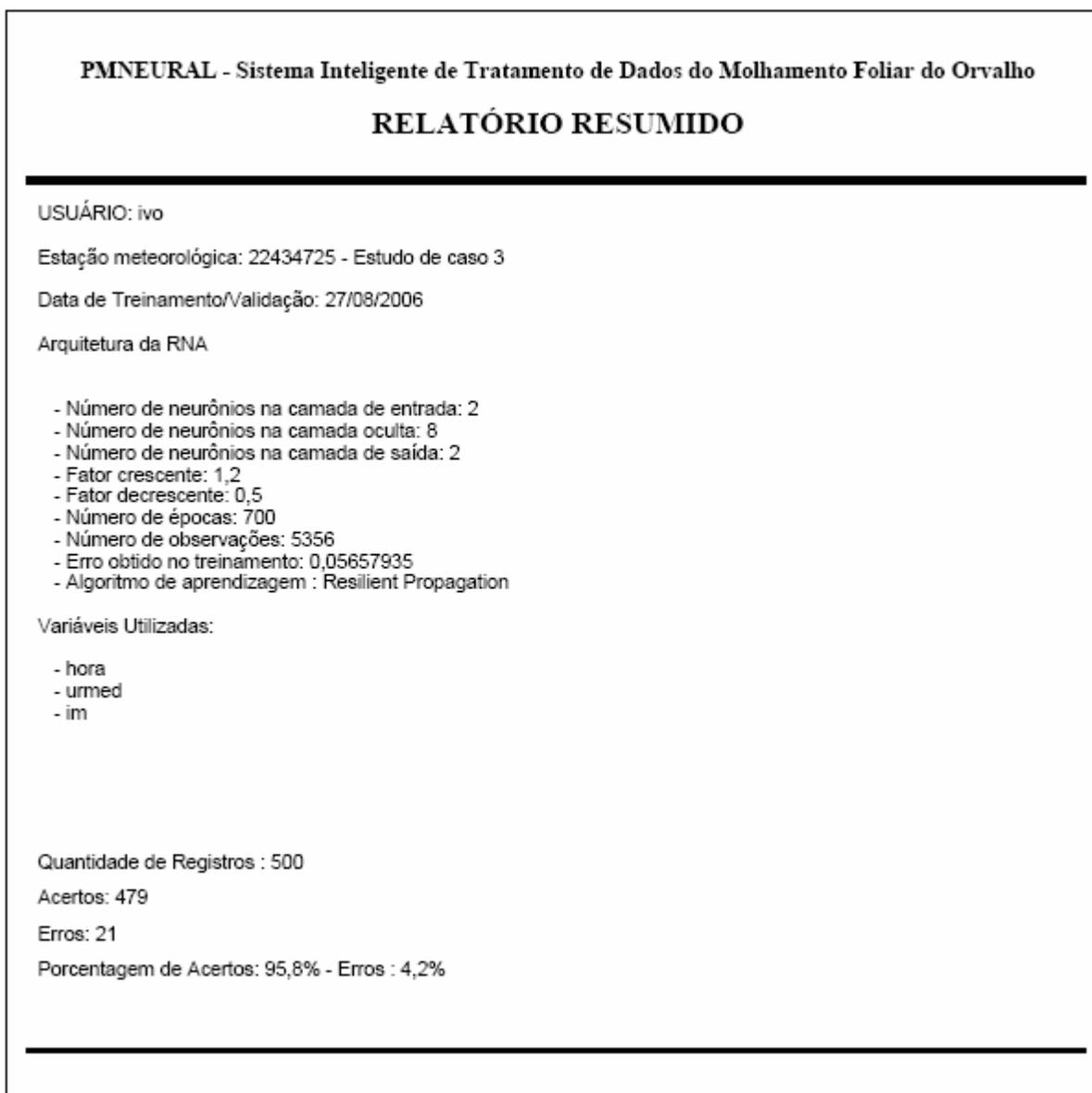


Figura 37 – relatório de validação - resumido

O próximo item da barra de menu é a *Ajuda*, esta opção também pode ser acessada a partir de qualquer módulo do sistema através da tecla de função *F1*. Permite ao usuário obter orientações sobre a operação do sistema, bem como, sobre a metodologia usada no desenvolvimento e implementação. Na ajuda são disponibilizadas duas formas de consulta, as quais podem ser acessadas através das abas: Localizar e Índice.

Em *Localizar* é possível efetuar uma consulta relativa ao sistema digitando uma palavra chave (Figura 38).

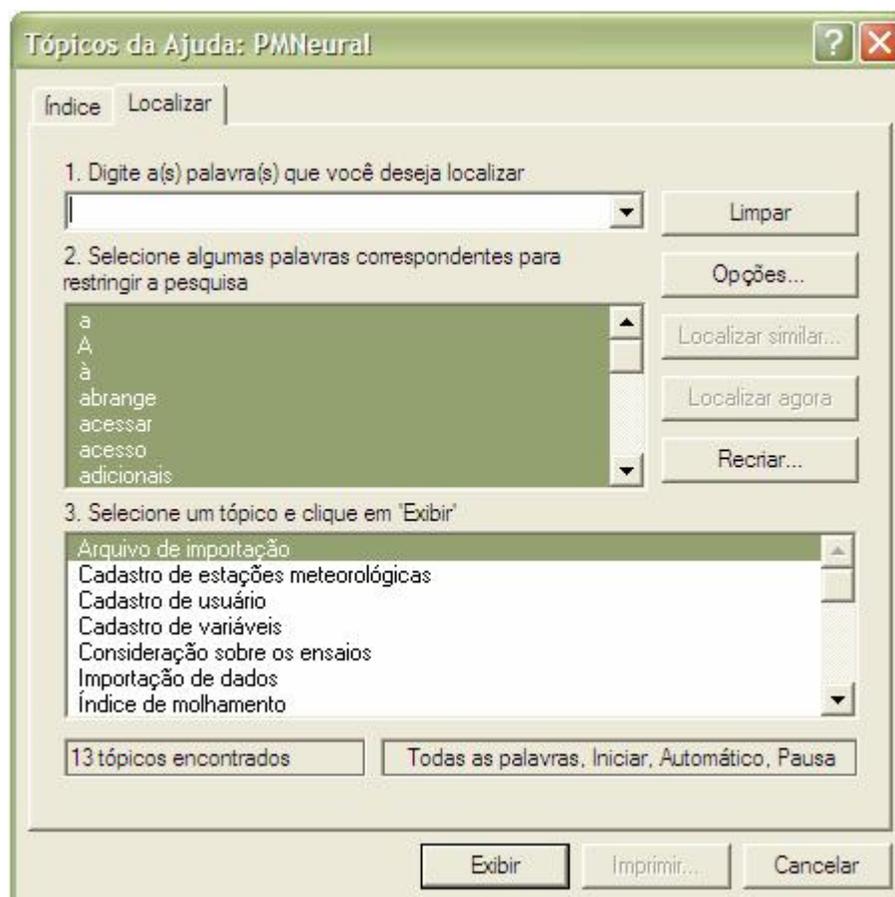


Figura 38 – Tela de ajuda – aba localizar

Na opção de *Índice* pode-se digitar as primeiras letras do índice disponível ou escolher diretamente um item na lista exibida (Figura 39).

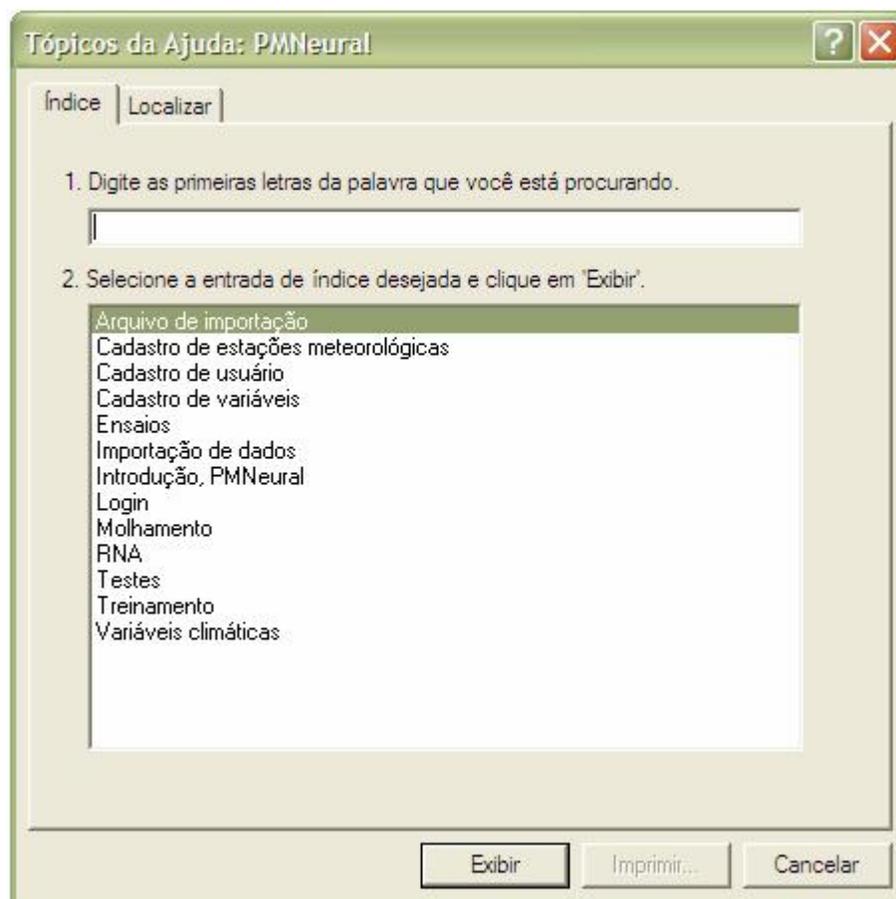


Figura 39 – Tela de ajuda – aba índice

O que foi descrito nesta seção corresponde às funcionalidades do sistema PMNeural. A seguir são apresentados resultados que têm por finalidade avaliar o potencial do sistema para classificação do molhamento foliar.

6.5 Avaliação do sistema PMNeural

Para avaliação do sistema foram utilizados os três estudos de caso descritos em Material e Métodos e também um quarto estudo de caso, cuja base de dados foi gerada unindo arquivos das bases de dados 1, 2 e 3. Com relação às variáveis climáticas, número de épocas, número de observações, bases de treinamento e validação, foram mantidos os mesmos dados e configurações. A única exceção foi nas arquiteturas das RNAs do Estudo de Caso 1, no qual originalmente foram considerados 10 índices de molhamento foliar, e na

avaliação do sistema foram convertidos para 2 índices (seco ou molhado). O motivo desta conversão visou uma padronização em relação aos outros estudos de caso.

No que se refere aos métodos de treinamento, foram destacados apenas os resultados com o algoritmo *Resilient Propagation*, visto que, os resultados do algoritmo *Backpropagation* não foram satisfatórios.

A principal diferença entre os procedimentos de avaliação descritos nas próximas seções, em relação aos da fase preliminar do trabalho com os simuladores SNNS e o JavaNNS, são os seguintes:

- § com os simuladores, os conjuntos de validação foram utilizados para determinar o momento de parada de treinamento, ou seja, a validação cruzada;
- § na avaliação do sistema, os conjuntos de validação foram utilizados para determinar os acertos e erros e seus respectivos percentuais, obtidos a partir das respostas apresentadas pelas RNAs previamente treinadas no próprio PMNeural, com os conjuntos de treinamento.

Portanto, assim como ocorreu nos experimentos iniciais com os simuladores, a avaliação do sistema PMNeural também foi realizada com dados diferentes daqueles utilizados durante a fase de aprendizado.

6.5.1 Estudo de caso 1

Na base de dados deste estudo de caso, a obtenção de dois índices de molhamento foi realizada considerando-se os índices de molhamento 1.0 e 1.2 como *seco* e os demais (1.5; 2.0; 2.5; 3.0; -2.5; -2.0; -1.8; -1.5 – Tabela 7) como *molhado*. Essa conversão de índices resultou em 525 (15,05%) ocorrências indicando a condição de *seco* e o restante, 2.963 (84,95%), como *molhado*, de um total de 3.488 registros.

Na Tabela 13, estão ilustrados os percentuais de erros e acertos de diversas arquiteturas de RNAs selecionadas a partir dos experimentos realizados com os simuladores.

Tabela 13 – Avaliação PMNeural - estudo de caso 1

Variáveis de Entrada	RNA	Épocas	Erro de Treino	Validação (%)	
				Acertos	Erros
HORÁRIO + URMED + TEMPMED + VENTMED	4-14-2	1400	0,03386	94,75	05,25
HORARIO + URMED + TEMPMED	3-11-2	1200	0,07819	92,50	07,50
HORARIO + URMAX + TEMPMIN	3-11-2	1200	0,04939	96,00	04,00
HORARIO + URMED	2-8-2	600	0,10450	85,25	14,75
HORARIO + URMAX	2-8-2	600	0,10450	85,25	14,75
URMED + TEMPMED	2-8-2	1000	0,09530	94,50	05,50
URMAX + TEMPMIN	2-8-2	1000	0,09413	96,00	04,00
MÉDIAS (%)				92,04	07,96

Percebe-se, nesta tabela, que o menor percentual de acerto foi de 85,25% e o maior 96%, em uma média de acertos de 92,04%. Também estão ilustrados as épocas e os erros de treinamento, bem como, as variáveis climáticas utilizadas como entradas para as RNAs.

6.5.2 Estudo de caso 2

Neste estudo de caso, manteve-se o objetivo de classificar o molhamento foliar em 2 padrões distintos, *seco* ou *molhado*, conforme pode ser observado na Tabela 14, onde as arquiteturas das RNAs demonstram que o número de neurônios da camada de saída é 2.

Tabela 14 – Avaliação PMNeural - estudo de caso 2

Variáveis de Entrada	RNA	Épocas	Erro de Treino	Validação (%)	
				Acertos	Erros
HORÁRIO + URMED + TEMPAMED + CHUVA	4-14-2	1600	0,10922	77,00	23,00
HORARIO + URMED + TEMPAMED	3-11-2	750	0,13282	82,70	17,30
HORARIO + URMED	2-8-2	600	0,13624	80,65	19,35
URMED + TEMPAMED	2-8-2	500	0,14606	83,10	16,90
MÉDIAS (%)				80,86	19,14

Pode-se observar, na Tabela 14, que esta base de dados proporcionou 77% como o menor percentual de acertos e 83,1% o maior, resultando em uma média de acertos de 80,86%.

6.5.3 Estudo de caso 3

De acordo com os resultados exibidos na Tabela 15, percebe-se que este estudo de caso apresentou resultados melhores que os anteriores, ou seja, o menor percentual de acertos foi de 94,4% e o maior 96,4% e uma média de acertos de 95,64%.

Tabela 15 – Avaliação PMNeural - estudo de caso 3

Variáveis de Entrada	RNA	Épocas	Erro de Treino	Validação (%)	
				Acertos	Erros
HORÁRIO + URMED + TEMPMED + VENTMED	4-14-2	1800	0,04251	95,80	04,20
URMED + TEMPMED + VENTMED	3-11-2	1500	0,04352	96,00	04,00
HORARIO + URMED + TEMPMED	3-11-2	1200	0,05229	95,60	04,40
HORARIO + URMED	2-8-2	700	0,05653	96,40	03,60
URMED + TEMPMED	2-8-2	1350	0,05350	94,40	05,60
MÉDIAS (%)				95,64	04,36

6.5.4 Estudo de caso 4

Neste estudo de caso, foi gerada uma quarta base de dados a partir dos arquivos dos estudos de casos 1, 2 e 3. Foram utilizadas apenas as variáveis climáticas comuns em todos os estudos de casos juntamente com seus respectivos índices de molhamento foliar. As variáveis comuns são: URMED, TEMPMED e o HORÁRIO. Esta junção de dados resultou em 28.454 registros para as bases de treinamento e 2.900 registros para as bases de validação.

Deste total de registros, os índices de molhamento ficaram assim distribuídos:

§ nas bases de treinamento das 28.454 ocorrências, 14.960 (52,6%) indicam a condição de *seco*, o restante dos registros, 13.494 (47,4%) a condição de *molhado*;

§ nas bases de validação das 2.900 ocorrências, 1.286 (44,3%) correspondem a condição de *seco*, o restante dos registros, 1.614 (55,7%) a condição de *molhado*.

Na Tabela 16, estão ilustrados os resultados obtidos com este estudo de caso, destacando-se o menor percentual de acertos de 73,96%, o maior 81,86% e uma média de 79,13% de acertos.

Tabela 16 – Avaliação PMNeural - estudo de caso 4

Variáveis de Entrada	RNA	Épocas	Erro de Treino	Validação (%)	
				Acertos	Erros
HORÁRIO + URMED + TEMP MED	3-11-2	1000	0,13369	81,86	18,14
HORARIO + URMED	2-8-2	1000	0,13767	79,20	20,80
HORARIO + TEMP MED	2-8-2	1000	0,17692	73,96	26,04
URMED + TEMP MED	2-8-2	1000	0,14747	81,49	18,51
MÉDIAS (%)				79,13	20,87

Estes resultados, assim como os demais, são discutidos a seguir.

7 DISCUSSÕES

De acordo com o descrito no capítulo anterior, os resultados gerados foram satisfatórios visto que, os acertos obtidos por meio das RNAs treinadas com o sistema PMNeural foram superiores a 73,96% chegando a 96,4%. Em termos de média, a menor foi 79,13% e a maior 95,64%. Isto demonstra a potencialidade do sistema na classificação do molhamento foliar e, conseqüentemente, o reconhecimento de um padrão de comportamento das variáveis meteorológicas em relação à formação de orvalho. O sistema desenvolvido demonstrou ser um novo modo de analisar o molhamento foliar e, uma ferramenta de auxílio no entendimento deste fenômeno climático.

As diferenças entre os resultados exibidos anteriormente, levando-se em consideração, principalmente, os três estudos de caso principais – 1, 2 e 3 –, podem estar relacionadas a fatores como: quantidade de amostras, períodos e intervalos diferentes das coletas dos dados. Estes fatores definiram os conteúdos diversos das bases de dados. Esta situação também caracteriza um comportamento de padrões distintos associados as variáveis climáticas (atributos previsores), o que atribui a estas uma capacidade maior ou menor de generalização.

Uma outra situação que pode ser levada em consideração, e que possivelmente também contribuiu para causar as diferenças de resultados entre os três estudos de caso, são os diferentes modos de obtenção do IMF (atributo meta), a saber:

- § Na base de dados do estudo de caso 1, o IMF foi obtido de forma manual em cultura do trigo;
- § Na base de dados 2, o IMF foi obtido por meio de um sensor de molhamento plano marca Davis®, que teve sua altura ajustada conforme o desenvolvimento da cultura do trigo, com inclinação de 45° em relação ao solo e voltado para a face norte;
- § E na base de dados 3, um sensor de molhamento Campbell modelo 237, instalado sobre o gramado a uma altura de 30 cm com um ângulo de exposição de 30° em relação à horizontal e voltado para o norte.

Estes diferentes modos de coleta dos dados, possivelmente, contribuíram para que os resultados obtidos no estudo de caso 3 tenham sido os melhores em relação aos estudos de caso 1 e 2, em função das condições mais estáveis para obtenção dos dados.

Evidentemente, deve-se levar em conta que, pelo fato dos resultados do sistema terem sido obtidos a partir de arquiteturas de RNAs baseadas em combinações com as variáveis climáticas, é possível que resultados ainda melhores possam ser obtidos com estes mesmos dados, ou ainda, com novos dados, em outros experimentos.

Diante desta situação, a implementação de um Módulo de Validação Cruzada (MVC) para o PMNeural, pode tornar este sistema ainda mais eficiente, uma vez que, o usuário do PMNeural terá a sua disposição uma ferramenta que possibilitará experimentar um número maior de arquiteturas de RNAs, em um tempo menor de processamento.

Outra vantagem que o MVC poderá trazer, é o fato que as RNAs geradas a partir do PMNeural, possivelmente, alcançarão uma melhor generalização, e, conseqüentemente, poderão gerar resultados com maior grau de acurácia.

O sistema PMNeural, embora utilize técnicas de RNAs que são complexas e exigem um conhecimento especializado, possui interfaces e mecanismos de operação que exigem do usuário apenas conhecimentos superficiais sobre a metodologia computacional. Estes conhecimentos podem ser obtidos na opção de ajuda, o que possibilita ao usuário obter noções básicas sobre RNAs, assim como, informações sobre as operações necessárias para a execução do sistema.

Outros mecanismos de operação que facilitam o uso do sistema estão refletidos, principalmente, nos valores *default* (padrão) de parâmetros que são requeridos durante a fase de treinamento das RNAs. O usuário tem a possibilidade de ajustar os parâmetros conforme suas necessidades, ou manter os valores *default* assumidos automaticamente pelo sistema, situação que é a mais recomendável, diante dos estudos que foram realizados.

Na questão operacional, ainda, o sistema resultante foi projetado para ser executado em computadores pessoais do tipo PC. É sabido que os PCs possuem diferentes velocidades de processador e capacidade da memória RAM, o que pode interferir no desempenho do sistema. Considerando que o desenvolvimento de RNAs é um processo de aprendizado de máquina em que o número de iterações, combinações de parâmetros e o tamanho das RNAs resultam em diferentes velocidades e tempo de processamento, foi realizado um estudo para avaliar o desempenho do PMNeural em diferentes configurações de equipamentos. A finalidade deste estudo foi avaliar apenas o número de épocas que cada computador poderia processar em dez horas de treinamento com o algoritmo *Resilient Propagation*.

Para este experimento foram utilizados dados de dois estudos de caso:

- § **EC_3** (Estudo de Caso 3) – variáveis meteorológicas: horário, urmed, tempmed e ventmed. Número de observações 5.356. Arquitetura das RNAs: 4-14-2;
- § **EC_4** (Estudo de Caso 4) – variáveis meteorológicas: horário, urmed, tempmed. Número de observações 28.454. Arquitetura das RNAs: 3-11-2;

Os computadores utilizados foram:

- § **Notebook_1** – *notebook* com processador Intel Pentium 725 Centrino, clock de 1.6 Giga Hertz (GHz), 2 Megabytes (MBytes) de memória cachê e 512 Mbytes de memória RAM;

- § **Notebook_2** – *notebook* com processador Intel Pentium M Centrino, clock de 1.86 Giga Hertz (GHz), 2 Megabytes (MBytes) de memória cachê e 1 Gibabyte (GByte) de memória RAM;
- § **Desktop_1** – *desktop* com processador AMD Athlon™ XP, clock de 1.8 GHz, 1 MByte de memória cachê e 256 MBytes de memória RAM;
- § **Desktop_2** – *desktop* com processador Intel Pentium 4 HT, clock de 3.2 GHz, 1 MByte de memória cachê e 1 Gibabyte de memória RAM.

Na Tabela 17, estão descritos os resultados obtidos com cada um dos computadores e respectivo estudo de caso.

Tabela 17 – Épocas processadas em 10 horas de treino c/o PMNeural – algoritmo Rprop

Estudos de Caso	Computadores	Épocas
EC_3	Notebook_1	80.500
	Notebook_2	90.370
	Desktop_1	69.230
	Desktop_2	49.140
EC_4	Notebook_1	22.640
	Notebook_2	26.050
	Desktop_1	20.040
	Desktop_2	11.845

Nos ensaios com os computadores Notebook_1, Notebook_2 e Desktop_1, os processamentos foram realizados sem nenhum aplicativo sendo executado em paralelo, ou seja, toda a capacidade de processamento de cada um dos processadores foi destinada apenas à execução do PMNeural. Foram mantidos apenas os processos essenciais ao funcionamento do sistema operacional (Windows XP). Com relação aos experimentos com o Desktop_2, por tratar-se de um processador com maior capacidade de processamento e maior memória RAM, neste caso, os processamentos das RNAs foram realizados com o computador sendo utilizado simultaneamente em outras tarefas, ou seja, o processamento foi em *background*.

Diante dos experimentos, confirma-se que, para se obter maior desempenho dos computadores no processamento das RNAs, a maior capacidade de memória RAM e o processamento dedicado são itens fundamentais.

8 CONSIDERAÇÕES

Diante dos resultados obtidos e da operacionalidade atual do sistema PMNeural, como trabalhos futuros foram definidas quatro metas:

- § O desenvolvimento e implementação de um Módulo de Validação Cruzada, já discutido no capítulo anterior;
- § Aprimoramento do módulo de importação, para que durante este processo a base de dados seja dividida automaticamente em dados para treinamento e validação. No que se refere à validação, o sistema deverá automaticamente escolher um método de amostragem aleatória e selecionar os registros de forma tal que sejam priorizados índices que aparecem em maior quantidade, ou não, conforme a composição da base de dados;
- § Módulo de consolidação de bases de dados, a finalidade deste módulo é possibilitar a união de diversas bases comuns em uma base maior, ou seja, na lista de bases existentes, o usuário poder escolher quais bases quer unir. Por exemplo: reunir variáveis climáticas de safras de inverno e verão, em uma única base de dados;
- § Criação de um Sistema de Apoio a Decisão que calcule a DPMF em tempo real. Para se obter a DPMF, parte-se do princípio que o PMNeural esteja conectado *on-line* com o *dataloger* de uma estação meteorológica, o qual estaria enviando os valores correspondentes das variáveis climáticas previamente definidas para a consulta às RNAs, de acordo com os parâmetros de treinamento e num

determinado intervalo de tempo. Conforme o índice de molhamento da RNA consultada, e com base no intervalo de tempo das leituras, a DPMF poderia ser estimada.

9 CONCLUSÕES

Em relação aos algoritmos de aprendizagem testados, o *Resilient Propagation* foi o que apresentou melhor desempenho com taxas de erro de treinamento variando entre 0.07209 a 0.34398 e de validação entre 0.06330 a 0.26467, demonstrando ser eficiente e apresentando os resultados satisfatórios já mencionados.

O sistema PMNeural, baseado em RNAs foi eficiente, com grau de exatidão entre 73,96% a 96,40%, para reconhecer padrões em dados de variáveis meteorológicas visando estimar o molhamento foliar originado por orvalho, em estudos de caso com bases de dados obtidas em áreas onde a cultura alvo foi o trigo.

O sistema desenvolvido demonstrou que as RNAs podem ser utilizadas como um novo modo para definir relações entre variáveis meteorológicas e o molhamento foliar por orvalho.

10 REFERÊNCIAS BIBLIOGRÁFICAS

ABELÉM, A. J. G. **Redes neurais artificiais na previsão de séries temporais.** 1994. Dissertação. 100p. (Mestrado em Engenharia Elétrica), Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 1994.

AGROSYSTEM, Web:

http://agrosystem.locaweb.com.br/produto_detalhe.asp?id_linha=3&id_produto=39, acesso em novembro/2005.

AMADOR, P. A. **Duração do período de molhamento por orvalho: estimativa baseada em parâmetros meteorológicos e comparação do desempenho de instrumentos de medida.** 1987. Dissertação. 69p. (Mestrado), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 1987.

ANDRADE, M. C. de; NETO, A. R.; ROSA, R. R.; SAWANT, H. S.; FERNANDES, F. C. R. Spatiotemporal Patterns Estimation Using a Multilayer Perceptron Neural Network in a Solar Physics Application. **Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais**, v.1, n.2, p. 159-167, 2003.

AZEVEDO, F. M.; BRASIL, L. M. e OLIVEIRA, R. C. L. de. **Redes neurais com aplicações em controle e em sistemas especialistas**. Florianópolis: Visual Books Editora, 2000, 401p.

BALESTRASSI, P. P. **Identificação de padrões em gráficos de controle estatístico de processos, em tempo real, utilizando séries temporais e redes neurais artificiais**, 2000. Tese. 217p. (Doutorado em Engenharia de Produção), Universidade Federal de Santa Catarina, Florianópolis, 2000.

BALLINI, R.; SOARES, S.; ANDRADE, M. G. Previsão de vazões médias mensais usando redes neurais nebulosas. **Sba Controle & Automação**, Campinas, v.14, n.3, p.680-693. ISSN 0103-1759, 2003.

BARRETO, J. M. **Introdução às redes neurais artificiais**, In: V Escola Regional de Informática da SBC Regional Sul, anais, Santa Maria, Florianópolis, Maringá, p.41-71, 1997.

BARROS, B. C. **Avaliação da resistência varietal e da resposta de cultivares de trigo (*Triticum aestivum* L.) ao controle químico de oídio, mancha marrom e ferrugem da folha**. 1988. Tese. 122p. (Doutorado), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 1988.

BENTON, W. **Enciclopédia Barsa**. São Paulo: Encyclopaedia Britannica Editores Ltda., 1970. p.163. v.10. 468p.

BOCANEGRA, C. W. R. **Procedimentos para tornar mais efetivo o uso de redes neurais artificiais em planejamento de transportes**. 2000. Dissertação. 97p. (Mestrado em Engenharia Civil), Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.

BORLAND, Web: <http://www.borland.com.br/products/index.html>, acesso em setembro/2003.

BRAGA, A. P.; CARVALHO, A. C. P. L. F.; LUDERMIR, T. B. **Redes Neurais Artificiais Teoria e Aplicações**. Rio de Janeiro, LTC – Livros Técnicos e Científicos Editora SA, 2000, 262p.

BUZZERIO, N. F. **Ferrugem e mancha angular do feijoeiro: efeito de fungicidas no desenvolvimento do hospedeiro e no progresso das doenças**. 2001. Dissertação. 115p. (Mestrado em Agronomia), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 2001.

CAMPBELL, Web: <http://www.campbellsci.com/237-1>, acesso em outubro/2005.

CANTERI, M. G.; DALLA PRIA, M.; SILVA, O. C. **Principais doenças fúngicas no feijoeiro: orientações para manejo econômico e ecológico**. Ponta Grossa, UEPG, 1999. 178p.

CANTU, M. **Dominando o Delphi 7, a Bíblia**. Tradução de Kátia Aparecida Roque. São Paulo, Pearson Education do Brasil, 2003.

CARVALHO, A.; LUDEMIR, A. **Fundamentos de Redes Neurais Artificiais: 11ª Escola de Computação**. Imprinta Gráfica e Editora Ltda, 1998.

CAUDILL, M. **Neural Network Primer**. Miller Freeman Publications, San Francisco, CA, 1990.

CROWE, M. J.; COAKLEY, S. M.; EMGE, R. C. Forecasting dew duration at Pendleton, Oregon, using simple weather observations. **Journal of Applied Meteorology**. Boston, 17(10). 1482-87, 1978.

DE MORI, C.; FILHO, A. F. **Custo de produção de trigo e de aveia: estimativa safra 2003**. Passo Fundo: Embrapa Trigo, 2003. 5p. Embrapa Trigo. Comunicado Técnico Online, n.111. Web: http://www.cnpt.embrapa.br/biblio/co/p_co111.htm, acesso em out/2004.

DIAS, A. H.; MATHIAS, I. M.; BIAGGIONI, M. A. M.; CATANEO, A.; PAES, A. R.; ESPERANCINI, M. S. T.; VIRGENS FILHO, J. S. das; **Avaliação do ciclo produtivo da laranja através de redes neurais artificiais**, In: XI SIMEP - Simpósio de Engenharia de Produção, Anais...., Bauru, 2004.

DIAS, A. H.; MATHIAS, I. M.; VIRGENS FILHO, J. S. das; RICKLI, L. I. **Simulador de dados genéricos – aplicação de redes neurais topologia MLP**, In: SUCESU 2004 – Congresso nacional de tecnologia da informação e comunicação. 2004. Anais.... Florianópolis, SUCESU-NACIONAL, 2004a, CD-ROM.

DIVERIO, V. T.; FORMAGGIO, A. R.; SHIMABUKURO, Y. Identificação de áreas desflorestadas na Amazônia através de uma rede neural artificial utilizando imagens fração derivadas dos dados do IR-MSS/CBERS, In: Anais XI SBSR – Simpósio brasileiro de sensoriamento remoto, 2003, Anais..., p.2697–2704, Belo Horizonte, INPE – Instituto Nacional de Pesquisas Espaciais, 2003.

EMBRAPA - Empresa Brasileira de Pesquisa Agropecuária - Embrapa Soja. **Informações Trigo**, Web: http://www.cnpso.embrapa.br/index.php?op_page=39&cod_pai=54, acesso em out/2004.

EMBRAPA - Empresa Brasileira de Pesquisa Agropecuária - Embrapa Trigo. **Informações Trigo**, Web: <http://www.cnpt.embrapa.br/culturas/trigo/index.htm>, acesso em mar/2006.

ESALQ, Web: <http://www.lce.esalq.usp.br/posto.html>, acesso em setembro/2005.

FEEKES. Feekes Growth Stages for Small Grains. 1940. Web: <http://www.ces.ncsu.edu/resources/crops/ag580/feekes.html>. Acesso em outubro/2004.

FISCHER, I., HENNECKE, F., BANNES, C., ZELL, A. JavaNNS – Java Neural Network Simulator. **User Manual – Version 1.1**. University of Tübingen, Wilhelm-Schickard - Institute for Computer Science, Department of Computer Architecture, 2001, Web:

<http://www-ra.informatik.uni-tuebingen.de/software/JavaNNS/manual/JavaNNS-manual.pdf>, acesso em setembro/2004.

FOWLER, M. **UML Essencial**, Porto Alegre, Bookman, 3a. Edição, 2005.

FRANCO, R. B.; MARTINS, W. Avaliação prática de redes neurais para o acasalamento de gado nelore, In: IV Congresso Brasileiro de Redes Neurais, Anais, p.243-246, ITA, São José dos Campos, 1999.

FREITAS, A. A. C. de; SILVA, I. N. da; SOUZA, A. N. de. Aplicação de redes neurais na estimação da temperatura interna de transformadores de distribuição imersos em óleo. **Sba Controle & Automação**, Campinas, ISSN 0103-1759, v.13, n.3, p.266-274, 2002.

FURLAN, J. D. **Modelagem de Objetos através da UML**. São Paulo: Makron Books, 1998, 330p.

GALO, M. de L. B. T. **Aplicação de redes neurais e sensoriamento remoto na caracterização ambiental do parque estadual Morro do Diabo**, 2000. Tese. 207p. (Doutorado em Ciências da Engenharia Ambiental), Universidade de São Paulo – USP, São Carlos, 2000a.

GALO, M. de L. B. T.; NOVO, E. M. L. de M.; GALO, M. **Ambigüidade entre classes de vegetação resultantes da aplicação de uma rede neural artificial**. In: IX Simpósio latino americano de perceptron remota, Puerto Iguazú, Misiones, Argentina, 2000b. Web: http://www2.prudente.unesp.br/dcartog/galo/pdf/2000_selper.pdf. Acesso em janeiro/2005.

GETZ, R.R. Coments on “forecasting dew duration at pendleton, Oregon, using simple weather observations”. **Journal of Applied Meteorology**. Boston, 18(8). p.1087, 1979.

GIOVANINI, R.; COURY, D. V. **Classificação rápida de faltas em sistemas elétricos utilizando redes neurais artificiais**, In: IV Congresso Brasileiro de Redes Neurais, Anais, p.281-286, ITA, São José dos Campos, 1999.

GLERIANI, J. M. **Redes neurais artificiais para classificação espectro-temporal de culturas agrícolas**. 2004. Tese. 212p. (Doutorado em Sensoriamento Remoto), Instituto Nacional de Pesquisas Espaciais – INPE, São José dos Campos, 2004.

GONÇALVES, C. H. R. **Utilizando redes neurais artificiais para predição de falhas em links de redes ópticas**. 2003. Dissertação. 148p. (Mestrado em Ciência da Computação), Universidade Federal do Ceará, Fortaleza, 2003.

GUIMARÃES, A. M.; VRIESMANN, L. M.; TOMASI, A. L. B.; CANTERI, M. G.; MATHIAS, I. M. Aplicação de Técnicas de Data Mining na obtenção de padrões de características físico-químicas do solo em função de sua produtividade. In: X Encontro Anual de Iniciação Científica e I Encontro de Pesquisa da UEPG, Anais..., Ponta Grossa: Editora UEPG, v.1. p.292 – 293, 2001.

GUIMARÃES, A. M. **Aplicação de computação evolucionária na mineração de dados físico-químicos da água e do solo**, 2005. Tese. 127p. (Doutorado em Agronomia – Energia na Agricultura), Universidade Estadual Paulista – UNESP, Botucatu, 2005.

HAYKIN, S. **Redes neurais princípios e prática**. Porto Alegre, Bookman, 2001, 900p.

HELDWEIN, A.B.; KRZYSCH, G. Escala para a observação visual do molhamento por orvalho. **Revista Brasileira de Agrometeorologia**, Santa Maria, v.5, n.2, p.207-214, 1997.

HUBBARD, K. G.; MAHMOOD, R.; CARLOS C. Estimating Daily Dew Point Temperature for the Northern Great Plains Using Maximum and Minimum Temperature, **Agronomy Journal**, Stanford University Libraries, v.95, p.323-328, 2003.

LEITE, R. M. V. B. de C. **Avaliação de danos e efeito de variáveis ambientais na mancha de alternaria (*alternaria helianthi*) em girassol**. 2002. Tese. 107p. (Doutorado em Agronomia), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 2002.

MATHIAS, I. M.; DIAS, A.H.; CATANEO, A.; GUIMARÃES, A. M. **Sistema especialista para avaliação da aptidão agrícola das terras**, In: SUCESU 2004 – Congresso nacional de tecnologia da informação e comunicação. 2004. Anais... Florianópolis, SUCESU-NACIONAL, 2004, CD-ROM.

MATHIAS, I. M.; GUIMARÃES, A. M.; DIAS, A. H.; TSUKAHARA, R. Y.; CATANEO, A. **Avaliação de redes neurais artificiais na estimativa da duração do período de molhamento foliar por orvalho na cultura do trigo**, In: V Congresso brasileiro de agroinformática. 2005. Anais... Londrina, SBIAGRO, 2005, CD-ROM.

MCCULLOCH, W. S.; PITTS, W. **A Logical Calculus of the Ideas Immanent in Nervous Activity**. Bulletin of Mathematical Biophysics, Vol.5, p.115-133, 1943.

MEDEIROS, J. S. de. **Bancos de geográficos e redes neurais artificiais: tecnologias de apoio à decisão do território**. Tese. 217p. (Doutorado em Ciências: Geografia Física), Universidade de São Paulo, Faculdade de Filosofia, Letras e Ciências Humanas, Departamento de Geografia, São Paulo, 1999.

MEDEIROS, L. F. de. **Redes neurais em delphi**. Florianópolis: Visual Books Editora, 2003, 115p.

MELO, E. **Previsão de doenças de plantas**. Passo Fundo: Editora Universitária – Universidade de Passo Fundo, 2004, 316p.

MIRANDA, R. A. C.; DAVIES, T. D.; CORNELL, S. E. A Laboratory assessment of witness sensors for leaf, fruit and trunk surfaces. **Agriculture and Forestry Meteorology**, n.102, p.263-274, 2000.

MONTEIRO, J. E. B. de A. **Microclima e ocorrência de ramulose no algodoeiro em diferentes densidades populacionais**. 2002. Dissertação. 99p. (Mestrado em Agronomia), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 2002.

MULLER, M.; FILL, H. D. Redes neurais aplicadas na propagação de vazões. In: XV SIMPÓSIO BRASILEIRO DE RECURSOS HÍDRICOS, 2003. Anais.... Curitiba, Associação Brasileira de Recursos Hídricos, 2003, **CD-ROM**.

NEURAI, Web: <http://www.din.uem.br/ia/neurais>, acesso em setembro/2003.

NIEVOLA, J. C. **Redes neurais artificiais**. In: XII Escola regional de informática – ERI 2004, Anais, p.01-50, ISBN 858844287-6, Universidade Estadual do Centro-Oeste – UNICENTRO, Guarapuava, 2004.

NORONHA, M. de A. **Escala diagramática para avaliação da mancha preta em folhas de citros e efeito da temperatura e da duração do molhamento na pré-penetração dos conídios de *Guignardia citricarpa* Kiely [*Phyllosticta citricarpa* (McAlp.) Van der Aa]**. 2002. Dissertação. 67p. (Mestrado em Agronomia), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 2002.

OSÓRIO, F. S.; BITTENCOURT, J. R. **Sistemas Inteligentes baseados em Redes Neurais Artificiais aplicados ao Processamento de Imagens**, In: I Workshop de inteligência artificial, 2000, apostila-seminário, Santa Cruz do Sul, UNISC – Universidade de Santa Cruz do Sul - Departamento de Informática, 2000.

PAULA, M. B. **Reconhecimento de palavras faladas utilizando redes neurais artificiais**. 2000. Monografia. 97p. (Bacharelado em Informática), Instituto de Física e Matemática, Universidade Federal de Pelotas, Pelotas, 2000.

PERNOMIAN, V. A.; TRINDADE JR, O. **Sistema de identificação de plantas invasoras em tempo real**, In: VIII SIMPÓSIO E DISSERTAÇÕES. São Carlos, ICMC- Instituto de

Ciências Matemática e de Computação de São Carlos, 2003, Web:

<http://www.icmc.usp.br/~std-cd/Artigos/Computacao/SDPC/VivianeAP.pdf>, acesso em janeiro/2005.

PESSOA, J. D. C.; FREIRE L. Redes Neurais para predição de variáveis climáticas relevantes para agricultura. **EMBRAPA - Comunicado Técnico – ISSN 1413-6244**, São Paulo, n.23, p.1-7, 1998.

PINTO, A. C. S.; POZZA, E. A.; SOUZA, P. E.; POZZA, A. A. A.; TALAMINI, V.; BOLDINI, J. M.; SANTOS, F. S. Descrição da epidemia da ferrugem do cafeeiro com redes neuronais. **Fitopatologia Brasileira**, v.27, p.517-524, 2002.

PIVONIA, S.; YANG, X. B. Assessment of the potential year-round establishment of soybean rust throughout the world. **Plant Disease**, v.88, n.5, p.523-529, 2004.

RAJVANSHI, A. K. Large scale dew collection as a source of fresh water supply. **Desalination**, v.36, n.3, p.299-306, 1981.

RAMOS, J. P. S. Redes neurais artificiais na classificação de frutos: cenário bidimensional. **Ciência e Agrotecnologia**, Lavras. Universidade Federal de Lavras (UFLA) v.27, n.2, p.356-362, 2003.

RIBEIRO, L. N. **Rede Neural com retropropagação: uma aplicação na classificação de imagens de satélite**. 2003. Dissertação. 131p. (Mestrado em Estatística), Departamento de Computação e Estatística, Universidade Federal do Mato Grosso do Sul - UFMS, Campo Grande, 2003.

RIEDMILLER, M.; BRAUN, H. **A direct adaptive method for faster backpropagation learning: the RPROP algorithm**, In: Proceedings of the IEEE, Int. conf. Ond Neural Network (ICN), San Francisco – USA, 1993, p. 586-591

ROMAN, E. S. Tecnologia de aplicação de herbicida na dessecação de *Brachiaria plantaginea* (Link) Hitchc. **EMBRAPA Trigo – Circular Técnica on-line – ISSN 1518-6490**, n.1, 1999.
Web: http://www.cnpt.embrapa.br/biblio/p_ci01.htm, acesso em outubro/2004.

ROHN, M. da C.; MINE, M. R. M. **Uma aplicação de redes neurais artificiais à previsão de chuvas de curtíssimo prazo**. In: XV Simpósio brasileiro de recursos hídricos. 2003. Anais.... Curitiba, Associação Brasileira de Recursos Hídricos, 2003, CD-ROM.

ROHN, M. da C.; GILBERTONI, R. de F. C.; KAVISKI, E.; COSTA, M.; NASCIMENTO, L. V. do. **Sistema de apoio à decisão para controle de outorgas: uma solução baseada em redes neurais artificiais**. In: XV Simpósio brasileiro de recursos hídricos. 2003, Anais.... Curitiba, Associação Brasileira de Recursos Hídricos, 2003, **CD-ROM**.

ROSENBLATT, F. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, v.65, 1958.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Internal Representation by Error Propagation**, in *Parallel Distributed Processing*. MIT Press, Cambridge, 1986.

SANTOS, M. T. dos; ALMEIDA, A. R. de; CHING TU, C. C. Redes neurais artificiais aplicadas no monitoramento da condição de ferramentas de corte utilizando algoritmo de extração das características SFS, In: IV Congresso Brasileiro de Redes Neurais, Anais, p.292-297, ITA, São José dos Campos, 1999.

SARLE, W. S. Web: <ftp://ftp.sas.com/pub/neural/FAQ.html.zip>, acesso em setembro/2004.

SEGATTO, Ê. C.; COURRY, D. V.; TAVARES, M. C.; CAMPOS, P. E. G. Alto desempenho na proteção diferencial de transformadores de potência com a utilização de redes neurais artificiais. **Sba Controle & Automação**, São José dos Campos, ISSN 0103-1759, v.14, n.3, p.309-320, 2003.

SENGER, L. J.; CALDAS JUNIOR, J. Análise de risco de crédito utilizando redes neurais artificiais. **Revista do CCEI**, Bage, ISSN 1415-2061, v.5, n.8, p.19-27, 2001.

SENTELHAS, P. C. **Ocorrência de helmintosporiose (*helminthosporium sativum*) e de oídio (*erysiphe graminis*) na cultura do trigo (*triticum aestivum*, L.) sob diferentes condições microclimáticas**. 1992. Dissertação. 103p. (Mestrado em Agronomia), Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 1992.

SENTELHAS, P. C.; PEDRO JUNIOR, M. J.; FELÍCIO, J. C. Estimativa da duração do período de molhamento para o trigo. **Revista Brasileira de Agrometeorologia**, Santa Maria, v.1, n.1, p.117-122, 1993.

SENTELHAS, P. C. de. **Duração do período de molhamento foliar: aspectos operacionais da sua medida, variabilidade espacial em diversas culturas e sua estimativa através modelo de Penman-Monteith**. Tese. 161p. (Livre Docência em Agrometeorologia), Universidade de São Paulo, Escola Superior de Agricultura Luiz de Queiroz – ESALQ, Piracicaba, 2004.

SILVA, S. R.; RIOS, G. P.; SILVA, S. C. Influência da resistência e do período de molhamento na infecção e desenvolvimento de lesões de ferrugem no feijoeiro. **Fitopatologia Brasileira**, v.26, p.726-731, 2001a.

SILVA, I. N. da; ARRUDA, L. V. R.; AMARAL, W. C. do; BORDON, M. E. Projeto e análise de uma rede neural para resolver problemas de programação dinâmica. **Sba Controle & Automação**, v.12, n.1, p.1-10, 2001b.

SILVEIRA, N. S. S.; MICHEREFF, S. J.; MARIANO, R. L. R.; TAVARES, L. A.; MAIA, L. C. Influência da temperatura, período de molhamento e concentração do inóculo de fungos na incidência de podridões pós-colheita em frutos de tomateiro. **Fitopatologia Brasileira**, v.26, p.33-38, 2001.

SOARES, J. F.; FARIAS, A. A. de; CESAR, C. C. **Introdução à Estatística**. Rio de Janeiro: Editora Guanabara, 1991, 378p.

STADZISZ, P.C. **Projeto de Software usando a UML**, Curitiba: CEFET, 2002, 69p.

SUTTON, J.C.; GILLESPIE, T.J.; HILDEBRAND, P.D. Monitoring weather factors in relation to plant disease. **Plant Disease**, v.68, n.1, p.78-84, 1984.

SZWARCFITER, J. L. **Grafos e algoritmos computacionais**. Rio de Janeiro: Editora Campus, 1986, 216p.

TÁPIA, M.; ROISENBERG, M.; BARRETO, J. M. Aplicação de redes neurais diretas na previsão de preços de ovos. In: Congresso e mostra de agroinformática – INFOAGRO, 2000, Ponta Grossa, Edição 2000. versão 1.0. **CD-ROM**.

TINÓS, R. **Detecção e diagnóstico de falhas em robôs manipuladores via redes neurais artificiais**. 1999. Dissertação. 117p. (Mestrado em Engenharia Elétrica), Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 1999.

TSUKAHARA, R. Y. **Sistema de suporte à tomada de decisão para o manejo fitossanitário nos Campos Gerais do Paraná**, 2004. Dissertação. 113p. (Mestrado em Agronomia), Universidade Estadual de Londrina, Londrina, 2004.

TSUKAHARA, R. Y. **Controle das principais doenças do trigo em função do monitoramento das variáveis ambientais**, Projeto AGR05_34. Fundação ABC, Castro, 2005.

UCI-ML. Machine Learning Repository. UCI – University of California Irvine. Dept. of Computer Science. Web: <http://www.ics.uci.edu/~mllearn/MLRepository.html>, acesso em setembro/2004.

WEIGEND, A. S., RUMELHART, D. E., HUBERMAN, B. A. Predicting the future: A connectionist approach, **International Journal Neural System**, 1990.

WEINERT, W. R.; LOPES, H. S. **Aplicação de um sistema neural ao problema de classificação de proteínas**, In: VI Congresso Brasileiro de Redes Neurais, Anais, p.85-90, Centro Universitário da FEI, São Paulo, 2003.

ZAHLER, P. J. M.; MOTA, F. S. da; AGENDES, M. O. de O. **Previsão agrometeorológica no controle de doenças e pragas dos vegetais**. Ministério da Agricultura e da Reforma Agrária, Brasília, 1991, 60 p.

ZELL, A. et al. SNNS – Stuttgart Neural Network Simulator. **User Manual – Version 4.2**. University of Stuttgart, Institute for Parallel and Distributed High Performance Systems, 1998, Web: <http://www-ra.informatik.uni-tuebingen.de/downloads/SNNS/SNNSv4.2.Manual.pdf>, acesso em setembro/2004.

ZUBEN, F. J. V. Uma Caricatura Funcional de Redes Neurais Artificiais. **Learning and Nonlinear Models – Revista da Sociedade Brasileira de Redes Neurais**, v.1, n.2, p. 77-87, 2003.

APÉNDICE

Guia de instalação – Pmneural

A instalação do sistema se inicia automaticamente após inserir o CD na unidade correspondente. O processo demora poucos minutos para ser realizado. À medida que a instalação avança, surgem caixas de diálogo solicitando ao usuário para: Avançar (ou *Next*), Instalar (ou *Install*), OK ou Cancelar (ou *Cancel*). Para que a instalação seja realizada com sucesso, devem ser mantidas as configurações a seguir.

Local de destino:

C:\Arquivos de programas\PMNeural

Pasta do Menu Iniciar:

PMNeural

O processo de instalação como um todo, ocorre em duas etapas.

§ **Observação:** Antes de iniciar a instalação do PMNeural, recomenda-se fechar todos os aplicativos que estiverem em execução.

Primeira etapa – instalação do PMNeural

Será criada a seguinte estrutura de diretórios na unidade < C: > do computador:

C:\Arquivos de programas\PMNeural\Ajuda\AJUDA.HLP

Corresponde ao arquivo de *Ajuda* do sistema, que pode ser acessado pela tecla de atalho *F1* ou pela opção correspondente nas barras de opções do sistema.

C:\ Arquivos de programas\PMNeural\Conhecimento

Esta pasta se destina a armazenar RNAs que serão gravadas após os respectivos treinamentos. Estes arquivos são acessíveis pelo sistema.

C:\ Arquivos de programas\PMNeural\Dados\PMNEURAL.GDB

Este arquivo corresponde ao banco de dados, o qual armazenará todas as operações realizadas.

C:\ Arquivos de programas\PMNeural\Importações\Dados_Treinamento_2000p_HORARIO_URMED_TEMP MED.csv

C:\ Arquivos de programas\PMNeural\Importações\Dados_Validação_200p_HORARIO_URMED_TEMP MED.csv

Estes dois arquivos – do Estudo de Caso 2 –, são exemplos de bases de dados, os quais têm por finalidade auxiliar o usuário a entender o funcionamento do sistema.

Para utilizar estas bases de dados, deve-se seguir os seguintes passos:

1. Inicialmente, deve-se fazer a importação dos arquivos correspondentes, usando como referência a própria nomenclatura, ou seja, um é para *Treinamento* e outro para *Validação* (Teste).
2. Posteriormente pode-se fazer o treinamento usando os parâmetros padrões, inclusive o número de épocas (500), bem como, o algoritmo *Resilient Propagation*.
3. Fazer o Teste com o arquivo que foi previamente importado para esta finalidade. A referência é a data da importação. Ao testar, é gerado um relatório completo, mas, pode-se optar pelo resumido, onde, são descritas todas as informações sobre o treinamento da RNA, bem como, o total de acertos e erros.

§ **Observação:** Os dados do arquivo para validação **não** são os mesmos do Treinamento.

C:\ Arquivos de programas\PMNeural\PMNEURAL.exe

C:\ Arquivos de programas\PMNeural\leiname.txt

O arquivo executável *PMNeural.exe*, refere-se ao aplicativo em si, a partir do qual se inicia a execução do programa. Para isso, basta usar o atalho na *área de trabalho (Desktop)* ou a partir do botão *Iniciar/Programas/PMNeural*.

O arquivo *leiname.txt* tem um conteúdo similar a este guia e, é executado ao final da instalação do sistema.

Os demais arquivos dessa pasta são relativos ao processo de desinstalação do programa e bibliotecas do sistema. Caso opte por desinstalar, recomenda-se usar o recurso do Windows a partir do *Painel de Controle* em *Adicionar e remover programas*.

Segunda etapa – instalação do Interbase 6.0.1 – Borland

Durante o processo de instalação também será instalado o Interbase 6.0.1, caso ainda não exista no computador em uso. Este *software* corresponde ao Sistema Gerenciador de Banco de Dados e, é indispensável ao funcionamento do sistema PMNeural.

Detalhe importante, o status do Interbase deve ser RUNNING (Executando). Este status é assumido automaticamente quando instalado a partir do PMNeural. Caso contrário, para verificar esta condição, pode-se acessar o Interbase a partir do *Painel de Controle* do Windows, clicando no ícone correspondente ao Interbase.

Se por algum motivo qualquer a instalação do Interbase não foi concluída com sucesso, pode-se instalá-lo diretamente do CD de instalação do PMNeural executando o aplicativo *Setup*, que se encontra na pasta `\InterBaseV6.0.1-server`.

§ Atenção: Antes de iniciar a execução do PMNeural, deve-se verificar se as Configurações do Windows do computador em uso encontram-se com:

- i. Formato de data abreviada (dd/MM/aaaa);
- ii. Números – símbolo decimal (,) – vírgula;
- iii. Números – símbolo de agrupamento de dígitos (.) – ponto.

Caso as configurações não sejam as exibidas acima, é necessário efetuar estas configurações para que o sistema tenha um funcionamento normal. Isto pode ser feito a partir do *Painel de Controle* do Windows em *Opções regionais e de idioma*, em seguida clica-se no botão *Personalizar...* no quadro de diálogo que aparece podem ser feitos os ajustes necessários.

Para utilizar o sistema PMNeural pela primeira vez, deve-se utilizar:

§ Login = admin

§ Senha = admin